
Where Graph Meets Heterogeneity: Multi-View Collaborative Graph Experts

Zhihao Wu^{1*}, Jinyu Cai^{2*}, Yunhe Zhang³, Jielong Lu¹, Zhaoliang Chen⁴, Shuman Zhuang⁵,
Haishuai Wang^{1†}

¹Zhejiang Key Laboratory of Accessible Perception and Intelligent Systems,
College of Computer Science and Technology, Zhejiang University

²Institute of Data Science, National University of Singapore

³Department of Computer and Information Science, University of Macau

⁴Department of Computer Science, Hong Kong Baptist University

⁵College of Computer and Data Science, Fuzhou University

{zhihaowu1999, jinyuca11995, zhangyhannie}@gmail.com, jielonglu2022@163.com,
chenzl23@outlook.com, shumanzhuang@163.com, haishuai.wang@zju.edu.cn

Abstract

The convergence of graph learning and multi-view learning has propelled the emergence of multi-view graph neural networks (MGNNs), offering strong capabilities to address complex real-world data characterized by heterogeneous and interconnected information. While existing MGNNs exploit the potential of multi-view graphs, the inherent conflict persists between the two critical inductive biases of multi-view learning, consistency and complementarity. Consequently, the challenge of defining and resolving this tension in the new context of multi-view graphs remains largely underexplored. To bridge this gap, we propose **Multi-view Collaborative Graph Experts (MvCGE)**, a novel framework grounded in the Mixture-of-Experts (MoE) paradigm. MvCGE establishes architectural consistency through shared parameters while preserving complementarity via layer-wise collaborative graph experts, which are dynamically activated by a graph-aware routing mechanism that adapts to the structural nuances of each view. This dual-level design is further reinforced by two novel components: a load equilibrium loss to prevent expert collapse and ensure balanced specialization, and a graph discrepancy loss based on distributional divergence to enhance inter-view complementarity. Extensive experiments on diverse datasets demonstrate MvCGE’s superiority.

1 Introduction

With the advancement of data collection and analysis capabilities, modern data is no longer treated as isolated or singular. Instead, entities are now understood to be deeply interconnected through complex and pervasive relationships, which are captured with increasing fidelity. Graphs, due to their universality, are widely used to represent these relationships as a powerful modeling tool. They excel at capturing complex dependencies and have become a cornerstone of machine learning, leading to advanced techniques for learning representations from structured data [Liang *et al.*, 2024; Cai *et al.*, 2024; Dai *et al.*, 2016]. Beyond inter-entity relationships, the ‘non-isolation’ of data is also manifested in the heterogeneous information sources, referred to as different views. For instance, in social networks, a user may be associated with various types of interactions (e.g., friendship ties vs. behavioral interactions); With multimedia systems, objects can be described using multiple

*Equal contribution.

†Corresponding author: Haishuai Wang.

modalities (e.g., visual features and semantic tags). This diversity introduces unique challenges, giving rise to new paradigms like multi-view learning [Liang *et al.*, 2025; Zhang *et al.*, 2025; Chen *et al.*, 2023a; Xu *et al.*, 2015]. For such heterogeneous data, prior work in multi-view learning provides a foundational framework built upon two well-known key properties. Specifically, each view provides unique information that complements the others, which is the property known as complementarity. Since these views are rooted in the same underlying entities, there is an inherent agreement between them, referred to as consistency. These two distinct characteristics enable multi-view learning to outperform the single-view counterpart [Tao *et al.*, 2019]. But these characteristics may inherently conflict [Dong *et al.*, 2023]. Overemphasis on either distinctive or shared information will diminish the advantage of multi-view learning. Therefore, the effective promotion of a balance between these two properties has become a key focus.

Both the universal inter-relationships and increasing views further extend the heterogeneity of data from different perspectives. But interestingly, graph learning uses topology as a unifying structure, providing a common ground to anchor, align, and integrate diverse information. Researchers have found that graph-based multi-view learning demonstrates promising performance [Li *et al.*, 2025; Wan *et al.*, 2024]. For example, Zhang *et al.*; Chen *et al.* facilitated view alignment by constructing graphs, which significantly enhanced clustering; while Duan *et al.*; Huang *et al.* focused on semi-supervised learning, fully exploiting the graph to amplify supervisory signals. However, these methods remain traditional, lacking a deeper understanding of graph theory, and mainly build on the concept of similarity preservation, struggling to cope with complex data. There is widespread interest in empowering multi-view learning with advanced graph techniques in the era of deep learning.

Recently, Graph Neural Networks (GNNs) have rapidly developed and become a powerful technology for graph learning. Spectral GNNs [Defferrard *et al.*, 2016] effectively extend convolutions to non-Euclidean graph data; Graph Convolutional Network (GCN) [Kipf and Welling, 2017], established upon this, was then proposed and quickly became a phenomenal model. Due to the powerful capabilities of GNNs [Xu *et al.*, 2019] and the synergistic effect between multi-view learning and graphs, Multi-view GNNs (MGNNs) have become an effective paradigm. Distinct from prior works, MGNNs typically explicitly formulate heterogeneous data as multi-view graphs and utilize modern GNN architectures. Although the graph structure is a powerful instrument for harmonizing heterogeneity, the classic dilemma between consistency and complementarity persists. The primary challenge shifts to *how to define and resolve this long-standing conflict within the new context of multi-view graphs*. Existing methods have made attempts, specifically, many MGNNs adopt a unified model that naturally allows for high collaboration between views through shared weights [Lin *et al.*, 2024; Chen *et al.*, 2023b; Cheng *et al.*, 2020; Xia *et al.*, 2022]. But this may lead to an overemphasis on consistency, thereby undermining complementarity. Also, in graph signal processing, a graph filter may fail to match the potentially heterogeneous multi-view graph structures. Another pipeline builds a multi-channel model [Chen *et al.*, 2024a; Lu *et al.*, 2024a; Wu *et al.*, 2023; Li *et al.*, 2020], where unique parameters are assigned to each view, ensuring complementarity. Because the interaction between views relies solely on post-fusion, this approach may sacrifice intrinsic inter-view interactions and lack architectural flexibility. The two existing pipelines are depicted in Figure 1. This raises a critical question: *How can we design a reasonable MGNN paradigm that achieves a balance between consistency and complementarity?*

In response to this challenge, we introduce a flexible and effective MGNN framework, termed **Multi-view Collaborative Graph Experts (MvCGE)** in this paper. Figure 2 briefly illustrates the architecture of MvCGE. First, by reviewing existing MGNNs through the lens of graph spectral theory, we identify a critical limitation: shared graph filters cannot effectively handle the heterogeneous structural properties of multi-view graphs. Drawing inspiration from Mixture of Experts (MoE), we propose layer-wise collaborative graph experts as the foundational building block of MvCGE. This block enables distinct graph filters, or regarded as GNN layers, to collaboratively process multi-view

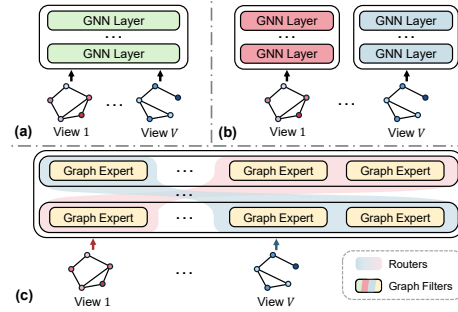


Figure 1: The illustration of MGNN paradigms based on (a) shared model, (b) multi-channel model, and (c) MvCGE.

graphs, thus preserving view-specific information. Notably, the experts are shared across views, establishing an architectural foundation for consistency. Moreover, the number of graph experts is decoupled from the number of views, offering better flexibility. The incorporated graph-aware routing allows for the dynamic selection of suitable graph experts based on the structural characteristics of each view. To safeguard against model collapse and the consequent loss of complementary information, we introduce two novel mechanisms. The load equilibrium loss ensures a balanced allocation of training effort among the experts, avoiding dominance or over-proliferation of certain experts. The graph discrepancy loss, grounded in graph isomorphism, treats multi-view graph latent representations as discrete distributions and encourages experts to learn distinct representations, preventing collapse into the same latent space. The main contributions of this paper are:

- We thoroughly review existing MGNN pipelines and reveal a critical challenge, i.e., shared or view-specific models fail to flexibly adapt to multi-view graphs.
- We propose MvCGE, a novel MoE-inspired MGNN framework, effectively balancing requirements for consistency and complementarity in handling heterogeneous data.
- Two innovative mechanisms are introduced to address model collapse and enhance the protection of view-specific information from different aspects.
- Experiments on both single- and multi-view graph datasets demonstrate that MvCGE outperforms competitors, showcasing impressive performance and generalization capabilities.

2 Related Work

Multi-view Learning Multi-view learning [Liu *et al.*, 2023, 2022; Chen *et al.*, 2022a] leverages information from multiple perspectives to enhance performance, with consistency capturing shared information and complementarity harnessing unique contributions. Graph-based approaches, known for modeling arbitrarily distributed data, have gained prominence in this field [Sun *et al.*, 2024; Wan *et al.*, 2023; Li *et al.*, 2019]. Nie *et al.* proposed to automatically weight view-wise graphs in multi-view clustering. Li *et al.* utilized graphs to enhance incomplete multi-view learning. Wen *et al.* leveraged graphs to refine cross-view correspondences. However, these methods fail to fully exploit graphs’ potential and lack integration with deep learning advancements.

Mixture of Experts The concept of MoE can be traced back to earlier works [Jacobs *et al.*, 1991; Jordan and Jacobs, 1994], designed for dynamically combining multiple specialized models to solve complex tasks. Modern advancements have adapted MoE to deep learning, making it powerful in vision and language models [Shazeer *et al.*, 2017; Lepikhin *et al.*, 2021; Fedus *et al.*, 2022; Fang *et al.*, 2022; Shen *et al.*, 2025; Miao *et al.*, 2025]. However, the potential of MoE in multi-view learning remains unexplored. Meanwhile, graph MoE is still in its infancy, with only a few researchers attempting to develop MoE for graph-based tasks [Ma *et al.*, 2024; Wang *et al.*, 2023; Hu *et al.*, 2022]. There remains a lack of investigation for multi-view graphs [Wang *et al.*, 2023].

Graph Neural Networks GNNs are a crucial tool for graph-based machine learning [Cai *et al.*, 2025; Wang *et al.*, 2025; Fang *et al.*, 2025b,a] and can be broadly divided into spectral and spatial approaches. Spectral GNNs, utilizing spectral graph theory to extend convolution operations to non-Euclidean space, once dominated the field. Kipf and Welling introduced the GCN, bridging the gap between spectral and spatial paradigms, and laying the foundation for modern GNNs. Building on these successes, MGNNs have emerged to tackle challenges in multi-view learning. Current MGNNs follow two main pipelines. One strategy constructs a unified model with shared parameters to process all views jointly, inherently ensuring consistency across views. To achieve this goal, Chen *et al.* proposed fusing multi-view graphs, while Lu *et al.* developed a graph generation approach. The other strategy employs multi-channel architectures, learning separate parameters for each view, as exemplified by [Lin *et al.*, 2024; Chen *et al.*, 2024b]. However, these two pipelines often lean toward two extremes: overemphasizing either consistency or complementarity. The former sacrifices view-specific information, while the latter may cause divergences and misalignments between views.

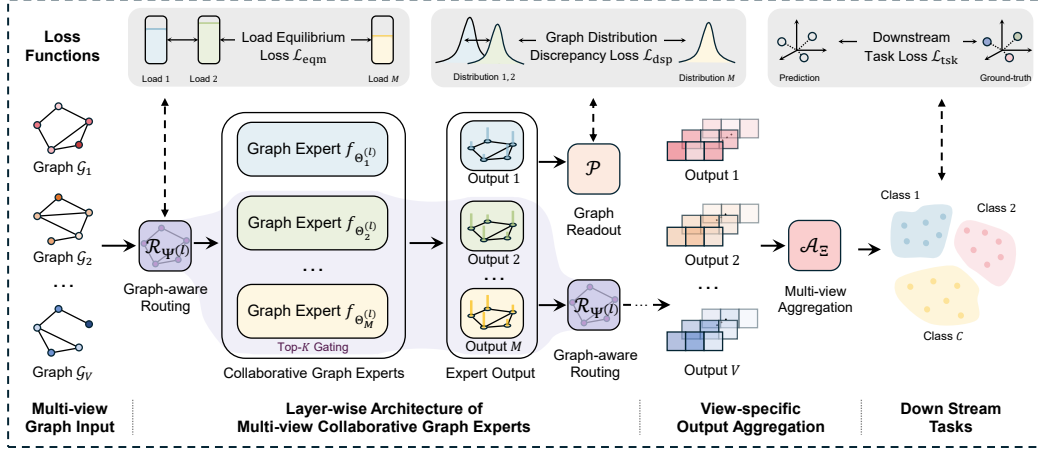


Figure 2: The illustration of a single layer of the proposed MvCGE.

3 Methodology

Notation We define an attributed graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, where \mathcal{V} is the node set with cardinality $|\mathcal{V}| = N$, and \mathcal{E} is the edge set. The graph’s topology is represented by its adjacency matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$. The node attributes are represented by a feature matrix $\mathbf{X} \in \mathbb{R}^{N \times D}$, where D is the dimension of the node features. The degree matrix $\mathbf{D} \in \mathbb{R}^{N \times N}$ is a diagonal matrix where $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$. The graph Laplacian is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$ or $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ for its symmetric version. A multi-view graph describes a set of graphs $\{\mathcal{G}_v : v \in [V]\}$ where the node set \mathcal{V} is associated with heterogeneous information. The v -th graph can be defined as $\mathcal{G}_v = (\mathcal{V}, \mathcal{E}_v, \mathbf{X})$ with corresponding adjacency matrix \mathbf{A}_v and shared attribute \mathbf{X} .

Motivation In this subsection, we will discuss existing paradigms from the lens of graph filters. To begin with, it is known that the Laplacian matrix can be decomposed as $\mathbf{L} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^\top$, which essentially induces a Fourier transform on the graph domain, where eigenvectors correspond to Fourier components and eigenvalues represent frequencies of the graph. Take \mathbf{X} as the input signal, the general form of a graph convolutional layer can be written as

$$\hat{\mathbf{X}} = f_\Theta(\mathcal{G}, \mathbf{X}), \quad (1)$$

where f_Θ is a function parameterized with Θ . For the early spectral GNNs [Bruna *et al.*, 2014], the function is specified as $h_\Theta(\mathbf{L})\mathbf{X}$, where $h_\Theta(\mathbf{L})$ is a parameterized graph filter. Due to the high computational burden, researchers then proposed to approximate the graph filter by a polynomial [Defferrard *et al.*, 2016], as

$$f_\Theta(\mathcal{G}, \mathbf{X}) = \sum_{t=0}^T \theta_t \mathbf{L}^t \mathbf{X}, \quad (2)$$

where $\Theta = \{\theta_t : t \in [T]\}$ are the parameters. [Defferrard *et al.*, 2016] achieved fast localized filters by using Chebyshev expansion, and the famous GCN [Kipf and Welling, 2017] further fixed $T = 2$ and simplified the convolution to bridge the spectral and spatial GNNs in a unified form. Therefore, without loss of generality, we describe the recursive formula of multi-layer modern GNNs:

$$\mathbf{H}^{(l+1)} = f_{\Theta^{(l)}}(\mathcal{G}, \mathbf{H}^{(l)}), \quad (3)$$

where $\mathbf{H}^{(l)} \in \mathbb{R}^{N \times D_l}$ is the input of the l -th layer, which is initialized by $\mathbf{H}^{(0)} = \mathbf{X}$, and $\Theta^{(l)}$ can be regarded as the extended multi-channel parameters of the graph filter. Training GNNs through optimizing the graph filter parameters $\Theta^{(l)}$ in each layer results in their powerful performance in graph-based machine learning tasks.

These GNNs have been rapidly applied to more complex and realistic multi-view graph scenarios. To ensure consistency across views, many approaches typically adopt a unified and shared model across all views. This strategy naturally enables highly collaborative learning among views by the weight

sharing, which leads to good consistency, but it often struggles to effectively model the multiplex graph structures inherent to diverse views. Considering any two views, we typically aim for the graph filter to produce results that exhibit consensus rather than significant disagreement. However, the effectiveness of a shared graph filter applied to two graph structures is significantly influenced by the spectral gap between these graphs. Discussions on the transferability or stability of graph filters are usually based on demanding assumptions [Ruiz *et al.*, 2020; Gama *et al.*, 2020; Levie *et al.*, 2021]. In other words, when the structures of the two graphs differ substantially, it becomes challenging to ensure that the graph filter achieves optimal performance across both views. An intuitive solution is to assign a separate graph filter to each view, allowing each view to learn parameters tailored to its own needs. This approach has been widely considered. However, while it takes into account more complementarities compared to a shared filter, it inevitably swings to the other extreme by sacrificing consistency in the architecture. Furthermore, the number of sub-modules is directly related to the number of views, which limits its flexibility. To address these limitations, we aim to engineer an MGNN framework that ensures scalable integration of inter-view dynamics, thereby better capturing and balancing the consistency and complementarity.

Overall Framework To begin with, let us consider the input from the v -th view and define the following layer-wise formulation:

$$\mathbf{H}_v^{(l+1)} = \mathcal{F}_{\Phi^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)}), \quad (4)$$

where $\mathbf{H}_v^{(l)}$ denotes the v -th graph embedding at the l -th layer with $\mathbf{H}_v^{(0)} = \mathbf{X}$, and the function $\mathcal{F}_{\Phi^{(l)}}$ with trainable parameter set $\Phi^{(l)}$ is assembled by multiple graph convolutions:

$$\mathcal{F}_{\Phi^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)}) = \mathcal{R}_{\Psi^{(l)}}(\{f_{\Theta_m^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)}) : m \in [M]\}), \quad (5)$$

where $\mathcal{R}_{\Psi^{(l)}}$ denotes the routing function parameterized by $\Psi^{(l)}$, and $\Phi^{(l)} = \{\Psi^{(l)}, \Theta_m^{(l)} : m \in [M]\}$. Building upon this, we introduce an ensemble of multiple graph filters, $\mathcal{F}_{\Phi^{(l)}}$, aiming to leverage graph filters equipped with distinct parameters to collaboratively process graph information from different views. Specifically, by designing the router $\mathcal{R}_{\Psi^{(l)}}$, appropriate combinations of graph filters are selected to handle potentially heterogeneous structures from various views. Subsequently, we consider processing a multi-view graph and constructing a multi-layer network. The proposed MvCGE framework calculates the final representation $\mathbf{H} \in \mathbb{R}^{N \times C}$ by

$$\mathbf{H} = \mathcal{A}_{\Xi}(\{\mathbf{H}_v = \mathcal{F}_{\Phi}(\mathcal{G}_v, \mathbf{X}) : v \in [V]\}), \quad (6)$$

where $\mathbf{H}_v \in \mathbb{R}^{N \times C}$ is the v -th learned final embedding, \mathcal{A}_{Ξ} is a multi-view aggregation function with parameter set Ξ , and \mathcal{F}_{Φ} is defined by concatenating L layer-wise functions:

$$\mathcal{F}_{\Phi} := \mathcal{F}_{\Phi^{(0)}} \circ \dots \circ \mathcal{F}_{\Phi^{(L-1)}}, \quad (7)$$

The entire function $\mathcal{F}_{\Phi^{(l)}}$ is expected to be shared across views, thereby facilitating inherent inter-view interactions. In our framework, \mathcal{F}_{Φ} is shared across views, and this characteristic ensures sufficient interaction between the graphs of different views at the architectural level. That is, all parameters Φ in the model are trained across the graphs of all views to capture consensus. Additionally, each layer $\mathcal{F}_{\Phi^{(l)}}$ of \mathcal{F}_{Φ} integrates multiple graph experts $f_{\Theta_m^{(l)}}$, which ensures that the model has sufficient capacity to adapt to the data of each view, thereby capturing their complementarities. Note that the number of graph experts M is independent of the number of views V , which provides the model with additional flexibility. Therefore, MvCGE strikes a balance between the inherent requirements for consistency and complementarity in multi-view learning. After outlining the framework of MvCGE, the following subsection will discuss the details and specific implementation.

Layer-wise Collaborative Graph Experts For the aforementioned design ideas, we implement our framework by drawing inspiration from the concept of MoE. Formally, given the input of the v -th view in the l -th layer $\mathbf{H}_v^{(l)}$, we specify that $\mathcal{F}_{\Phi^{(l)}}$ aggregate decisions from several graph experts by

$$[\mathbf{H}_v^{(l+1)}]_{i,:} = \sum_{m \in [M]} [\Gamma_v^{(l)}]_{i,m} f_{\Theta_m^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)})_{i,:}, \quad (8)$$

where $\Gamma_v^{(l)} \in \mathbb{R}^{N \times M}$ is the v -th routing matrix that contains the routing weights of all the graph filters in the l -th layer, which acts as graph experts in this framework. Through this layer-wise design,

each layer of MvCGE can be viewed as an independent MoE and is controlled by a specific router, allowing for more flexible selection and combination of graph experts.

Next, we specify the routing function $\mathcal{R}_{\Psi^{(l)}}$, whose role is to select the appropriate combinations of experts for each node in each view. We propose an improved graph-aware top- k gating mechanism. Specifically, we denote the gating weight of the i -th sample in the v -th view to the m -th expert as

$$[\Gamma_v^{(l)}]_{i,m} = \text{Softmax}(g_{\Psi^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)})_{i,:}) \quad \text{s.t. } \|[\Gamma_v^{(l)}]_{i,:}\|_0 = K, \forall i \in [N],$$

where the gating module $g_{\Psi^{(l)}}$ outputs an M -dimensional vector acting as the weights for integrating outputs from diverse experts. Note that this module has a constraint to ensure sparse gating. In multi-view graph learning scenarios, the design of $g_{\Psi^{(l)}}$ is critical, as it is key to preserving view specificity. It must both perceive the graph and view-level patterns.

$$g_{\Psi^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)})_{i,:} = \text{TopK}\left(\left[\mathcal{P}(\{\mathbf{H}_v^{(l)}\}_{j,:}, j \in \mathcal{S}_i)\right] \|\mathcal{P}(\mathbf{H}_v^{(l)})\| \mathbf{W}_{\text{gate}}^{(l)}\right). \quad (9)$$

where \mathcal{P} is a pooling function and \mathcal{S}_i is the node set associated with a sampled local subgraph of node i . Therefore, $\mathcal{P}(\{\mathbf{H}_v^{(l)}\}_{j,:}, j \in \mathcal{S}_i)$ and $\mathcal{P}(\mathbf{H}_v^{(l)})$ respectively encode the local graph structure and view-specific global information. $\mathbf{W}_{\text{gate}} \in \mathbb{R}^{2D \times M}$ is the trainable parameter matrix mapping the input to gating scores, respectively. We adopt Top- K sparse gating: for each node i , only the Top- K expert logits are kept while the rest are masked out (set to $-\infty$) before applying softmax. This yields $\|[\Gamma_v^{(l)}]_{i,:}\|_0 = K$ with the remaining weights being exactly zero, such that only the experts with top- K gating weight are selected.

Acceleration Trick When processing large-scale graphs, graph convolution operations can become a major bottleneck due to the massive number of edges. The employment of multiple graph experts imposes additional computational overhead. Therefore, we consider further specializing the graph experts and propose an acceleration trick. We formalize the graph expert as

$$f_{\Theta_m^{(l)}}(\mathcal{G}_v, \mathbf{H}_v^{(l)}) = \zeta(\mathbf{P}_v \mathbf{H}_v^{(l)} \Theta_m^{(l)}), \quad (10)$$

where ζ is a non-linear activation function, \mathbf{P}_v is a filtering operator for the view v , and $\Theta_m^{(l)} \in \mathbb{R}^{D_l \times D_{l+1}}$ is the trainable weight matrix. Here, we consider a class of simplified spectral GNNs where the filter parameters are treated as being encoded in the trainable weights $\Theta_m^{(l)}$, thereby decoupling a parameter-free operator \mathbf{P}_v . For example, it can be the Chebyshev polynomial $\sum_{t=0}^T \mathbf{T}_t(\mathbf{L}_v)$ for ChebNet [Defferrard *et al.*, 2016], and $\mathbf{I} - \mathbf{L}_v$ for GCN [Kipf and Welling, 2017]. For this class of graph experts, we only need to compute the parameter-free graph filtering once per MvCGE layer. This result is then shared among different experts, which apply their distinct trainable weights for different graph convolutions. This considerably accelerates the entire framework.

Regularization Although MvCGE ensures consistency through a flexible architecture, its training faces challenges that could lead to model collapse, thereby losing the complementary information between views.

Load Equilibrium: From the optimization perspective, due to the gating mechanism [Shazeer *et al.*, 2017], in some cases MvCGE may overly rely on a group of graph experts while preventing others from being trained. This imbalance is self-reinforcing, causing the selected experts to proliferate, while other experts become increasingly unlikely to be chosen. Ultimately, MvCGE will collapse into a model with only a few experts activated, losing the advantage of capturing view-specific information. To address this, we introduce a load equilibrium loss. For the v -th view, let

$$\rho_v^m = \frac{1}{N} \sum_i [\Gamma_v^{(l)}]_{i,m}, \quad \omega_v^m = \frac{1}{N} \sum_i \mathbb{I}\{[\Gamma_v^{(l)}]_{i,m} > 0\}$$

be the average gating weight and load of expert m , we have

$$\mathcal{L}_{\text{eqm}} = \frac{1}{V} \sum_{v=1}^V \frac{1}{M} \sum_{m=1}^M \omega_v^m \cdot \rho_v^m \log \rho_v^m, \quad (11)$$

where we consider the intra-view and then the inter-view equilibrium. By minimizing \mathcal{L}_{eqm} , each graph experts are encouraged to have similar selection scores and frequency.

Graph Discrepancy: From the perspective of the latent space, even if load equilibrium is ensured and different graph experts are given equal training opportunities, they may collapse into similar latent spaces. Different graph experts are supposed to perform distinct filtering on multi-view graphs, rather than collapsing into a single graph filter. To address this challenge, we take a single-layer MvCGE as an example to elucidate the computational mechanism of the graph discrepancy loss. Given the v -th view graph \mathcal{G}_v , let f_{Θ_m} and $f_{\Theta_{m'}}$ denote any two distinct graph experts, the expert-specific embeddings are defined as:

$$[\mathbf{Z}_v^m]_{i,:} = [\Gamma_v]_{i,m} f_{\Theta_m}(\mathcal{G}_v, \mathbf{X})_{i,:}, \quad (12)$$

where Γ_v denotes the routing matrix and \mathbf{Z}_v^m denotes the expert-specific embedding generated by expert f_{Θ_m} . Similarly, we obtain $\mathbf{Z}_v^{m'}$ for the other graph expert. For clarity, we omit the layer index in this single-layer illustration. Then the two embeddings are fed into the readout function, or called pooling function as

$$\mathbf{z}_v^m = \mathcal{P}(\mathbf{Z}_v^m). \quad (13)$$

Here, \mathbf{z}_v^m can be interpreted as a global graph embedding that encapsulates the semantic and structural information captured by expert f_{Θ_m} . Recent studies have demonstrated that readout functions paired with GNNs can effectively perform graph isomorphism computations [Xu *et al.*, 2019]. Therefore, we combine the readout function with a layer of GNN to form a model \mathcal{Q} . Then for all V views, these graph embeddings $\{\mathbf{z}_v^m : v \in [V]\}$ are treated as samples drawn from a distribution $\mathbb{P}_{\mathcal{Q}_m}$ induced by the function f_{Θ_m} where $\mathcal{Q}_m = \mathcal{P} \circ f_{\Theta_m}$.

After that, we leverage Maximum Mean Discrepancy (MMD) [Gretton *et al.*, 2012] to quantify the discrepancy between distributions from different graph experts:

$$\mathcal{M}_{\mathcal{H}}^2(\mathbb{P}_{\mathcal{Q}}, \mathbb{P}_{\mathcal{Q}'}) = \|\mu_{\mathcal{Q}_m} - \mu_{\mathcal{Q}_{m'}}\|_{\mathcal{H}}^2 = \mathbb{E}_{\mathbb{P}_{\mathcal{Q}}}[\kappa(\mathbf{z}_v^m, \mathbf{z}_{v'}^m)] + \mathbb{E}_{\mathbb{P}_{\mathcal{Q}'}}[\kappa(\mathbf{z}_{v'}^{m'}, \mathbf{z}_{v''}^{m'})] - 2\mathbb{E}_{\mathbb{P}_{\mathcal{Q}}, \mathbb{P}_{\mathcal{Q}'}}[\kappa(\mathbf{z}_v^m, \mathbf{z}_{v'}^{m'})], \quad (14)$$

where κ is a kernel function, e.g., Radial Basis Function kernel $\kappa(\mathbf{z}, \mathbf{z}') = \exp(-\frac{\|\mathbf{z} - \mathbf{z}'\|^2}{2\sigma^2})$. Graph embeddings \mathbf{z} and \mathbf{z}' are sampled from distribution $\mathbb{P}_{\mathcal{Q}}$ and $\mathbb{P}_{\mathcal{Q}'}$ respectively. $\mu_{\mathbb{P}_{\mathcal{Q}}}$ and $\mu_{\mathbb{P}_{\mathcal{Q}'}}$ are the means of distributions $\mathbb{P}_{\mathcal{Q}_m}$ and $\mathbb{P}_{\mathcal{Q}_{m'}}$ in the reproducing kernel Hilbert space (RKHS) \mathcal{H} , respectively. Then we introduce the generalization of MMD

$$\mathcal{M}_{\mathcal{H}}^2(\mathbb{P}_{\mathcal{Q}}, \mathbb{P}_{\mathcal{Q}'}) = \sup \{ \|\mu_{\mathcal{Q}} - \mu_{\mathcal{Q}'}\|_{\mathcal{H}}^2 : \kappa \in \mathcal{K} \}, \quad (15)$$

where \mathcal{K} is a family of kernels. According to [Gretton *et al.*, 2012], we specifically estimate MMD by

$$\mathcal{M}_{\mathcal{H}}^2(\mathbb{P}_{\mathcal{Q}_m^{(l)}}, \mathbb{P}_{\mathcal{Q}_{m'}^{(l)}}) = \sup_{\kappa \in \mathcal{K}} \left[\frac{1}{V^2} \sum_v \sum_{v'} \kappa(\mathbf{z}_v^m, \mathbf{z}_{v'}^m) + \frac{1}{V'^2} \sum_{v'} \sum_{v''} \kappa(\mathbf{z}_{v'}^{m'}, \mathbf{z}_{v''}^{m'}) - \frac{2}{V^2} \sum_v \sum_{v'} \kappa(\mathbf{z}_v^m, \mathbf{z}_{v'}^{m'}) \right]. \quad (16)$$

Consequently, considering the multi-layer case, we propose the graph discrepancy loss:

$$\mathcal{L}_{\text{dsp}} = -\frac{1}{L} \sum_l \frac{1}{M(M-1)} \sum_{m < m'} \mathcal{M}_{\mathcal{H}}^2(\mathbb{P}_{\mathcal{Q}_m^{(l)}}, \mathbb{P}_{\mathcal{Q}_{m'}^{(l)}}), \quad (17)$$

where $\mathcal{Q}_m^{(l)} = \mathcal{P} \circ f_{\Theta_m^{(l)}}$ is a model combining a readout function and a graph expert. Utilizing the two losses together ensures that the graph experts receive similar opportunities and learn meaningful latent representations, thereby promoting MvCGE to capture complementary information.

Training Strategy The proposed MvCGE is trained using the following objective function

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{tsk}} + \alpha \mathcal{L}_{\text{eqm}} + \beta \mathcal{L}_{\text{dsp}}, \quad (18)$$

where \mathcal{L}_{tsk} is the downstream task loss. The widely used Cross-Entropy loss is adopted for the semi-supervised classification task:

$$\mathcal{L}_{\text{tsk}} = -\sum_{i \in \Omega} \sum_j \mathbf{Y}_{i,j} \log \hat{\mathbf{Y}}_{i,j}, \quad (19)$$

where $\hat{\mathbf{Y}}, \mathbf{Y} \in \mathbb{R}^{|\Omega| \times C}$ are predicted and ground-truth labels, and Ω is the training set. Note that $\hat{\mathbf{Y}}_{i,:} = \text{Softmax}(\mathbf{H}_{i,:})$.

Table 1: Macro F1 and Micro F1 scores of all methods on multi-view datasets (mean value of ten runs), where the best and the second-best results are highlighted in **orange** and **blue**, respectively.

Datasets Metrics	ACM-M		DBLP		IMDB		YELP		AMINER	
	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1
GCN	0.7860	0.7881	0.9012	0.9163	0.2429	0.5544	0.5196	0.6743	0.6842	0.8156
DGI	0.2235	0.3690	0.2432	0.3763	0.2626	0.5519	0.5034	0.6831	0.3449	0.6460
HAN	0.9147	0.9139	0.8927	0.9039	0.2389	0.5586	0.4829	0.4893	0.7234	0.8476
DMGI	0.8666	0.8681	0.6567	0.7106	0.3533	0.5726	0.5161	0.6985	0.3029	0.6546
IGNN	0.8290	0.8270	0.8681	0.8750	0.4531	0.5481	0.6449	0.7123	0.7453	0.8521
MRGCN	0.8758	0.8745	0.8949	0.9047	0.4517	0.4769	0.5435	0.7370	0.7335	0.8294
SSDCM	0.8765	0.8763	0.8942	0.8990	0.4940	0.5910	0.5270	0.7020	0.2620	0.5548
MHGCN	0.8887	0.8907	0.9300	0.9360	0.5154	0.6423	0.6085	0.7328	0.7500	0.8517
AMOGCN	0.9241	0.9238	0.9227	0.9280	0.5016	0.6506	0.6753	0.7241	0.7570	0.8338
HMGE	0.9080	0.9066	0.9156	0.9235	0.3293	0.5735	0.5857	0.7484	0.7371	0.8478
MvCGE	0.9348	0.9245	0.9256	0.9317	0.5157	0.6883	0.7776	0.8008	0.7897	0.8640

Table 2: Macro F1 and Micro F1 scores of all methods on single-view datasets (mean value of ten runs), where the best and the second-best results are highlighted in **orange** and **blue**, respectively.

Datasets Metrics	ACM		Citeseer		CoraFull		Flickr		UAI	
	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1	MaF1	MiF1
GCN	0.8827	0.8842	0.6523	0.6909	0.5277	0.6279	0.5002	0.5104	0.4506	0.5851
SGC	0.8116	0.8087	0.6157	0.6642	0.5320	0.6292	0.4422	0.5101	0.4679	0.5652
APPNP	0.8837	0.8824	0.6510	0.6972	0.5236	0.6395	0.5124	0.5212	0.4655	0.6189
JKNet	0.8524	0.8545	0.6813	0.7292	0.5126	0.6268	0.5112	0.5431	0.4134	0.5623
DAGNN	0.8699	0.8917	0.6812	0.7290	0.5353	0.6481	0.6214	0.6065	0.4778	0.5927
GCNII	0.8978	0.8980	0.6712	0.7064	0.5828	0.6403	0.5904	0.5771	0.4869	0.6382
GNNHF	0.9052	0.9067	0.6760	0.7249	0.5846	0.6672	0.5867	0.6078	0.4495	0.5764
AMGNN	0.8996	0.8993	0.6873	0.7292	0.5664	0.6185	0.7580	0.7569	0.4787	0.6431
HiDNet	0.8958	0.8961	0.6661	0.7146	0.5304	0.6531	0.6200	0.6333	0.4568	0.6165
AGNN	0.8973	0.8974	0.6264	0.6721	0.5065	0.5930	0.5721	0.5844	0.4697	0.6206
MvCGE	0.9110	0.9114	0.6908	0.7356	0.5924	0.6693	0.7675	0.7790	0.4797	0.6591

4 Experiments

Experimental Setting To evaluate the effectiveness of MvCGE, we conducted experiments on ten datasets, including five multi-view graph datasets (ACM, DBLP, IMDB, YELP, AMINER) and five typical single-view graph datasets (ACM, Citeseer, CoraFull, Flickr, UAI) plus one large-scale multi-view graph Freebase and one large-scale single-view graph OGBN-arXiv. To adapt MvCGE to single-view graphs, we generate a supplementary view for each single-view dataset based on the k -Nearest Neighbor algorithm, where we set $k = 10$. Note that, to distinguish between the different ACM datasets, we denote the natively multi-view graph ACM as ACM-M. For different types of datasets, we compared MvCGE with various competitors, including nine methods designed for multi-view datasets (DGI [Velickovic *et al.*, 2019], HAN [Wang *et al.*, 2019], DMGI [Park *et al.*, 2020], IGNN [Gu *et al.*, 2020], MRGCN [Huang *et al.*, 2020], SSDCM [Mitra *et al.*, 2021], MHGCN [Yu *et al.*, 2022], AMOGCN [Chen *et al.*, 2024b], HMGE [Abdous *et al.*, 2024]) and ten methods designed for single-view datasets (GCN [Kipf and Welling, 2017], SGC [Wu *et al.*, 2019], APPNP [Gasteiger *et al.*, 2018], JKNet [Xu *et al.*, 2018], DAGNN [Liu *et al.*, 2020], GCNII [Chen *et al.*, 2020], GNNHF [Zhu *et al.*, 2021], AMGNN [Zhu *et al.*, 2023], HiDNet [Li *et al.*, 2024], AGNN [Chen *et al.*, 2023c]).

Performance In this subsection, we conduct comprehensive evaluations of MvCGE’s performance in semi-supervised classification tasks across diverse datasets, empirically demonstrating its superior performance compared to state-of-the-art methods. Experimental setups and analyses on these datasets are as follows: 1) *Multi-view Graphs*. Here, the training ratio is set to 20% and the Macro F1 and Micro F1 scores are recorded in Table 1. The experimental results reveal that the proposed

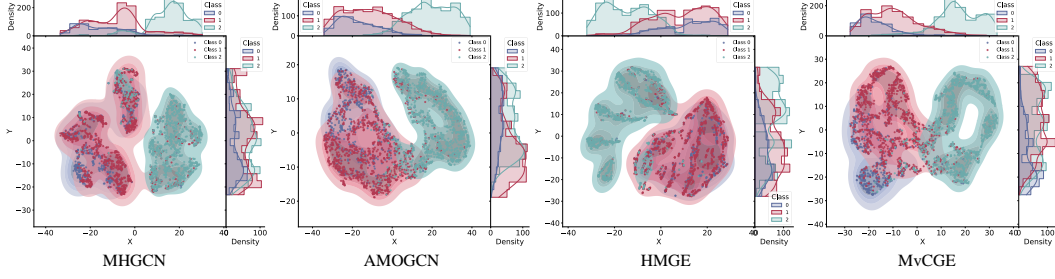


Figure 3: The visualization of representations learned by MHGCN, AMOGCN, HMGE, and MvCGE on the YELP dataset.

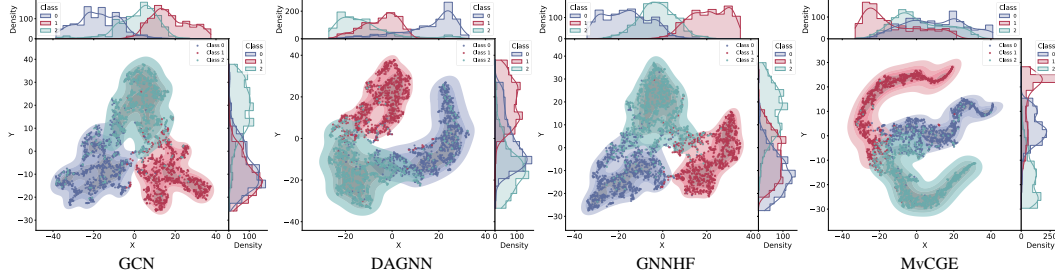


Figure 4: The visualization of representations learned by GCN, DAGNN, GNNHF, and MvCGE on the ACM dataset.

MvCGE outperforms baseline methods across most datasets. Notably, substantial performance gains are achieved on the Yelp dataset, primarily due to the inherent limitations of meta-path-based approaches in effectively modeling complex non-linear cross-view relationships. In contrast, guided by the principles of MoE, MvCGE effectively resolves the consistency-complementarity trade-off while preserving view-specific information in multi-view learning, as validated in our experiments. 2) *Single-view Graphs*. Following the commonly used semi-supervised node classification settings, we randomly select 20 samples per class for training, 500 samples for validation, and 1,000 samples for testing, with the detailed results presented in Table 2. The figure shows that MvCGE consistently outperforms all baseline methods and exhibits strong compatibility and adaptability to single-view datasets. Additionally, as depicted in Appendix, we also test MvCGE on the large-scale graph OGBN-arXiv and Freebase, showing its scalability.

Visualization We employ t-SNE [Van der Maaten L, 2008] to visualize the learned representations of top-performing baselines on the YELP (Figures 3) and ACM datasets (Figures 4). Each node is represented as a point, color-coded by its class label, with contour lines indicating category density distributions. Complementary histograms along the axes reveal the embedding distributions across dimensions. The visualizations demonstrate that MvCGE achieves superior cluster separation with well-defined class boundaries and compact intra-class distributions. Analysis of the histogram distributions reveals that MvCGE achieves better-separated representations in the feature space.

Sensitivity In this section, we assess the impact of varying the parameters α and β on the performance of MvCGE, where α and β balance the loss \mathcal{L}_{eqm} and \mathcal{L}_{dsp} , as depicted in Figure 5 (a) and (b). It can be seen that model performance exhibits similar stable fluctuation patterns for both α and β across ACM and YELP datasets, with the optimal performance achieved at $\alpha = 1e^{-1}$ or $1e^{-2}$ and $\beta = 1e^{-1}$ or $1e^{-3}$, demonstrating that the integrated two losses effectively capture complementary information and consistently enhance performance when appropriate parameter values are selected. Beyond this, we evaluated the model’s sensitivity to the number of experts. Specifically, we varied the total number of experts and selected half of them, with the corresponding results presented in Figure 6 (a). In Figure 6 (b), we fixed the total number of experts at 10 while varying the number of selected experts. The findings indicate that choosing an optimal subset of experts, rather than an excessive or minimal number, enhances the model’s performance.

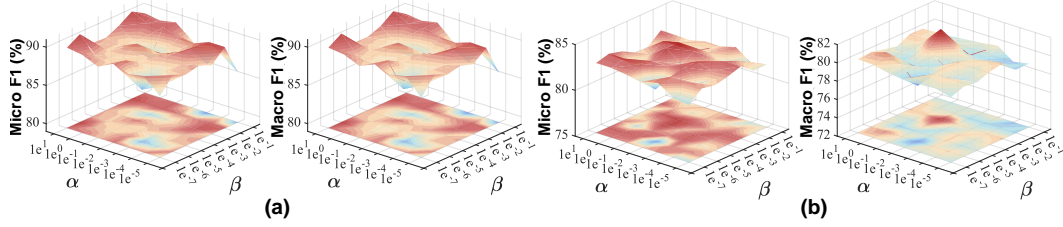


Figure 5: The classification performance of MvCGE w.r.t. hyperparameters α and β on the (a) ACM dataset and (b) YELP dataset.

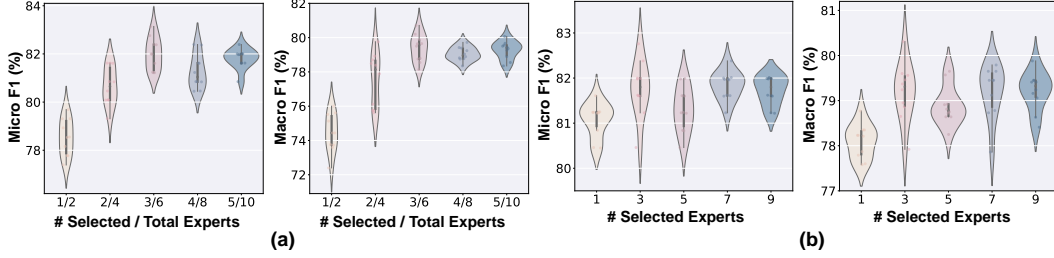


Figure 6: The classification performance of MvCGE w.r.t. (a) the number of selected and total graph experts and (b) the number of selected graph experts on the YELP dataset.

5 Conclusion

In this paper, we address the critical challenge of balancing consistency and complementarity in multi-view graph learning. Traditional multi-view GNNs often struggle to adapt to the heterogeneous structural properties of multi-view graphs due to rigid weight-sharing or isolated multi-channel designs. To overcome these limitations, we propose MvCGE, a novel framework inspired by Mixture of Experts, which introduces layer-wise collaborative graph experts to dynamically process multi-view graphs while preserving both shared and view-specific information. The integration of load equilibrium loss and graph discrepancy loss effectively mitigates model collapse and enhances the discriminative power of learned representations. Extensive experiments across diverse datasets demonstrate that MvCGE achieves state-of-the-art performance, showcasing its flexibility and robustness in handling complex multi-view graph data. One possible limitation is that we do not discuss the homophily and heterophily of graphs, although we adopt both types of graphs in our experiments.

6 Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant Nos. 62202422 and 62372408).

References

- Kamel Abdous, Nairouz Mrabah, and Mohamed Bouguessa. Hierarchical aggregations for high-dimensional multiplex graph embedding. *IEEE Transactions on Knowledge and Data Engineering*, 2024.
- Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations*, 2014.
- Jinyu Cai, Yunhe Zhang, Jicong Fan, and See-Kiong Ng. Lg-fgad: An effective federated graph anomaly detection framework. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 3760–3769, 2024.
- Jinyu Cai, Yunhe Zhang, and Jicong Fan. Self-discriminative modeling for anomalous graph detection. In *Proceedings of the International Conference on Machine Learning*, 2025.

- Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International conference on machine learning*, pages 1725–1735, 2020.
- Mansheng Chen, Tuo Liu, Chang-Dong Wang, Dong Huang, and Jian-Huang Lai. Adaptively-weighted integral space for fast multiview clustering. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 3774–3782. ACM, 2022.
- Yongyong Chen, Xiaolin Xiao, Chong Peng, Guangming Lu, and Yicong Zhou. Low-rank tensor graph learning for multi-view subspace clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):92–104, 2022.
- Man-Sheng Chen, Jia-Qi Lin, Chang-Dong Wang, Wudong Xi, and Dong Huang. On regularizing multiple clusterings for ensemble clustering by graph tensor learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3069–3077, 2023.
- Zhaoliang Chen, Lele Fu, Jie Yao, Wenzhong Guo, Claudia Plant, and Shiping Wang. Learnable graph convolutional network and feature fusion for multi-view learning. *Information Fusion*, 95:109–119, 2023.
- Zhaoliang Chen, Zhihao Wu, Zhenghong Lin, Shiping Wang, Claudia Plant, and Wenzhong Guo. Agnn: Alternating graph-regularized neural networks to alleviate over-smoothing. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.
- Man-Sheng Chen, Xi-Ran Zhu, Jia-Qi Lin, and Chang-Dong Wang. Contrastive multiview attribute graph clustering with adaptive encoders. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–12, 2024.
- Zhaoliang Chen, Zhihao Wu, Luying Zhong, Claudia Plant, Shiping Wang, and Wenzhong Guo. Attributed multi-order graph convolutional network for heterogeneous graphs. *Neural Networks*, 174:106225, 2024.
- Jiafeng Cheng, Qianqian Wang, Zhiqiang Tao, De-Yan Xie, and Quanxue Gao. Multi-view attribute graph convolution networks for clustering. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 2973–2979, 2020.
- Hanjun Dai, Bo Dai, and Le Song. Discriminative embeddings of latent variable models for structured data. In *International conference on machine learning*, pages 2702–2711. PMLR, 2016.
- Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*, 2016.
- Zhibin Dong, Siwei Wang, Jiaqi Jin, Xinwang Liu, and En Zhu. Cross-view topology based consistent and complementary information for deep multi-view clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19440–19451, 2023.
- Yu Duan, Zhoumin Lu, Rong Wang, Xuelong Li, and Feiping Nie. Toward balance deep semisupervised clustering. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2024.
- Xiang Fang, Daizong Liu, Pan Zhou, and Yuchong Hu. Multi-modal cross-domain alignment network for video moment retrieval. *IEEE Transactions on Multimedia*, 25:7517–7532, 2022.
- Hui Fang, Yang Gao, Peng Zhang, Sheng Zhou, Hongyang Chen, Jiajun Bu, and Haishuai Wang. Contrastive federated learning for graph anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.
- Hui Fang, Haishuai Wang, Yang Gao, Yonggang Zhang, Jiajun Bu, Bo Han, and Hui Lin. Insgnn: Interpretable spatio-temporal graph neural networks via information bottleneck. *Information Fusion*, 119:102997, 2025.
- William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

- Fernando Gama, Joan Bruna, and Alejandro Ribeiro. Stability properties of graph neural networks. *IEEE Transactions on Signal Processing*, 68:5680–5695, 2020.
- Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. In *Proceedings of the International Conference on Learning Representations*, 2018.
- Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.
- Fangda Gu, Heng Chang, Wenwu Zhu, Somayeh Sojoudi, and Laurent El Ghaoui. Implicit graph neural networks. In *Advances in Neural Information Processing Systems*, 2020.
- Fenyu Hu, Liping Wang, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Graphdive: Graph classification by mixture of diverse experts. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pages 2080–2086, 2022.
- Zhichao Huang, Xutao Li, Yunming Ye, and Michael K. Ng. MR-GCN: multi-relational graph convolutional networks based on generalized tensor product. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1258–1264, 2020.
- Aiping Huang, Zheng Wang, Yannan Zheng, Tiesong Zhao, and Chia-Wen Lin. Embedding regularizer learning for multi-view semi-supervised classification. *IEEE Transactions on Image Processing*, 30:6997–7011, 2021.
- Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.
- Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.
- Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.
- Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*, 2021.
- Ron Levie, Wei Huang, Lorenzo Bucci, Michael Bronstein, and Gitta Kutyniok. Transferability of spectral graph convolutional neural networks. *Journal of Machine Learning Research*, 22(272):1–59, 2021.
- Zhaoyang Li, Qianqian Wang, Zhiqiang Tao, Quanxue Gao, and Zhaohua Yang. Deep adversarial multi-view clustering network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 2952–2958, 2019.
- Shu Li, Wen-Tao Li, and Wei Wang. Co-gcn for multi-view semi-supervised learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 4691–4698, 2020.
- Xingfeng Li, Yinghui Sun, Quansen Sun, Zhenwen Ren, and Yuan Sun. Cross-view graph matching guided anchor alignment for incomplete multi-view clustering. *Information Fusion*, 100:101941, 2023.
- Yibo Li, Xiao Wang, Hongrui Liu, and Chuan Shi. A generalized neural diffusion framework on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8707–8715, 2024.
- Xingfeng Li, Yuangang Pan, Yuan Sun, Yinghui Sun, Quansen Sun, Zhenwen Ren, and Ivor W. Tsang. Scalable unpaired multi-view clustering with bipartite graph matching. *Information Fusion*, 116:102786, 2025.
- Ke Liang, Yue Liu, Sihang Zhou, Wenxuan Tu, Yi Wen, Xihong Yang, Xiangjun Dong, and Xinwang Liu. Knowledge graph contrastive learning based on relation-symmetrical structure. *IEEE Transactions on Knowledge and Data Engineering*, 36(1):226–238, 2024.

- Ke Liang, Lingyuan Meng, Hao Li, Jun Wang, Long Lan, Miaomiao Li, Xinwang Liu, and Huaimin Wang. From concrete to abstract: Multi-view clustering on relational knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(10):9043–9060, 2025.
- Jia-Qi Lin, Man-Sheng Chen, Xi-Ran Zhu, Chang-Dong Wang, and Haizhang Zhang. Dual information enhanced multiview attributed graph clustering. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–12, 2024.
- Meng Liu, Hongyang Gao, and Shuiwang Ji. Towards deeper graph neural networks. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 338–348, 2020.
- Jiyuan Liu, Xinwang Liu, Yuexiang Yang, Xifeng Guo, Marius Kloft, and Liangzhong He. Multiview subspace clustering via co-training robust data representation. *IEEE Transactions on Neural Networks and Learning Systems*, 33(10):5177–5189, 2022.
- Jiyuan Liu, Xinwang Liu, Yuexiang Yang, Qing Liao, and Yuanqing Xia. Contrastive multi-view kernel learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):9552–9566, 2023.
- Jielong Lu, Zhihao Wu, Zhaoliang Chen, Zhiling Cai, and Shiping Wang. Towards multi-view consistent graph diffusion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 186–195, 2024.
- Jielong Lu, Zhihao Wu, Luying Zhong, Zhaoliang Chen, Hong Zhao, and Shiping Wang. Generative essential graph convolutional network for multi-view semi-supervised classification. *IEEE Transactions on Multimedia*, 26:7987–7999, 2024.
- Li Ma, Haoyu Han, Juanhui Li, Harry Shomer, Hui Liu, Xiaofeng Gao, and Jiliang Tang. Mixture of link predictors on graphs. In *Advances in Neural Information Processing Systems*, 2024.
- Rui Miao, Yixin Liu, Yili Wang, Xu Shen, Yue Tan, Yiwei Dai, Shirui Pan, and Xin Wang. Blindguard: Safeguarding llm-based multi-agent systems under unknown attacks. *arXiv*, 2025.
- Anasua Mitra, Priyesh Vijayan, Sanasam Ranbir Singh, Diganta Goswami, Srinivasan Parthasarathy, and Balaraman Ravindran. Semi-supervised deep learning for multiplex networks. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1234–1244, 2021.
- Feiping Nie, Jing Li, and Xuelong Li. Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 1881–1887, 2016.
- Chanyoung Park, Donghyun Kim, Jiawei Han, and Hwanjo Yu. Unsupervised attributed multiplex network embedding. In *Proceedings of the AAAI conference on artificial intelligence*, pages 5371–5378, 2020.
- Luana Ruiz, Luiz Chamon, and Alejandro Ribeiro. Graphon neural networks and the transferability of graph neural networks. In *Advances in Neural Information Processing Systems*, pages 1702–1712, 2020.
- Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.
- Xu Shen, Yixin Liu, Yiwei Dai, Yili Wang, Rui Miao, Yue Tan, Shirui Pan, and Xin Wang. Understanding the information propagation effects of communication topologies in llm-based multi-agent systems. In *EMNLP*, 2025.
- Yuan Sun, Jian Dai, Zhenwen Ren, Yingke Chen, Dezhong Peng, and Peng Hu. Dual self-paced cross-modal hashing. In *Thirty-Eighth AAAI Conference on Artificial Intelligence*, pages 15184–15192, 2024.

- Zhiqiang Tao, Hongfu Liu, Sheng Li, Zhengming Ding, and Yun Fu. Marginalized multiview ensemble clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 31(2):600–611, 2019.
- Hinton G Van der Maaten L. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008.
- Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. In *Proceedings of the International Conference on Learning Representations*, 2019.
- Xinhang Wan, Xinwang Liu, Jiyuan Liu, Siwei Wang, Yi Wen, Weixuan Liang, En Zhu, Zhe Liu, and Lu Zhou. Auto-weighted multi-view clustering for large-scale data. In *Thirty-Seventh AAAI Conference on Artificial Intelligence*, pages 10078–10086, 2023.
- Xinhang Wan, Jiyuan Liu, Xinbiao Gan, Xinwang Liu, Siwei Wang, Yi Wen, Tianjiao Wan, and En Zhu. One-step multi-view clustering with diverse representation. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2024.
- Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *Proceedings of the International World Wide Web Conference*, pages 2022–2032, 2019.
- Haotao Wang, Ziyu Jiang, Yuning You, Yan Han, Gaowen Liu, Jayanth Srinivasa, Ramana Kompella, and Zhangyang "Atlas" Wang. Graph mixture of experts: Learning on large-scale graphs with explicit diversity modeling. In *Advances in Neural Information Processing Systems*, pages 50825–50837, 2023.
- Haishuai Wang, Yang Gao, Xin Zheng, Peng Zhang, Jiajun Bu, and Philip S. Yu. Graph neural architecture search with large language models. In *Science China Information Sciences*, 2025.
- Yi Wen, Siwei Wang, Qing Liao, Weixuan Liang, Ke Liang, Xinhang Wan, and Xinwang Liu. Unpaired multi-view graph clustering with cross-view structure matching. *IEEE Transactions on Neural Networks and Learning Systems*, 35(11):16049–16063, 2024.
- Felix Wu, Amauri H. Souza Jr., Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. In *Proceedings of the International Conference on Machine Learning*, volume 97, pages 6861–6871, 2019.
- Zhihao Wu, Xincan Lin, Zhenghong Lin, Zhaoliang Chen, Yang Bai, and Shiping Wang. Interpretable graph convolutional network for multi-view semi-supervised learning. *IEEE Transactions on Multimedia*, 25:8593–8606, 2023.
- Wei Xia, Qianqian Wang, Quanxue Gao, Xiangdong Zhang, and Xinbo Gao. Self-supervised graph convolutional network for multi-view clustering. *IEEE Transactions on Multimedia*, 24:3182–3192, 2022.
- Chang Xu, Dacheng Tao, and Chao Xu. Multi-view self-paced learning for clustering. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 3974–3980, 2015.
- Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *International conference on machine learning*, pages 5453–5462, 2018.
- Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.
- Pengyang Yu, Chaofan Fu, Yanwei Yu, Chao Huang, Zhongying Zhao, and Junyu Dong. Multiplex heterogeneous graph convolutional network. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2377–2387, 2022.
- Chao Zhang, Deng Xu, Xiuyi Jia, Chunlin Chen, and Huaxiong Li. Continual multi-view clustering with consistent anchor guidance. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 5434–5442, 2024.

- Yunhe Zhang, Jinyu Cai, Zhihao Wu, Pengyang Wang, and See-Kiong Ng. Mixture of experts as representation learner for deep multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 22704–22713, 2025.
- Meiqi Zhu, Xiao Wang, Chuan Shi, Houye Ji, and Peng Cui. Interpreting and unifying graph neural networks with an optimization framework. In *Proceedings of the International World Wide Web Conference*, pages 1215–1226, 2021.
- Meiqi Zhu, Xiao Wang, Chuan Shi, Yibo Li, and Junping Du. Towards adaptive information fusion in graph convolutional networks. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):13055–13069, 2023.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [NA].
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes]" is generally preferable to "[No]", it is perfectly acceptable to answer "[No]" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No]" or "[NA]" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”,**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [Yes]

Justification: The abstract and introduction accurately reflect the claims of the paper.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Limitations are discussed in Conclusion.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: The assumptions and proofs are included in Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: All information needed to reproduce the main experimental results of the paper is provided in the main body, Appendix, and source code.

Guidelines:

- The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is available in an anonymous link.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: The details are included in Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[NA\]](#)

Justification: [\[NA\]](#)

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: The computer resources are detailed in the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The research conducted in the paper conform with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: There is no societal impact of the work performed.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper has no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [\[Yes\]](#)

Justification: The creators or original owners of assets (e.g., code, data, models), used in the paper, are properly credited and the license and terms of use explicitly are mentioned and properly respected.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[NA\]](#)

Justification: This paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.