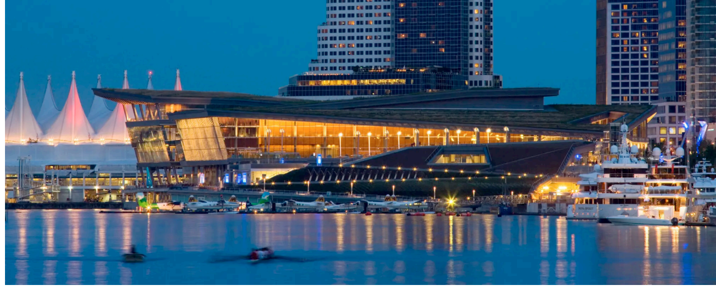
NEURAL INFORMATION
PROCESSING SYSTEMS



NeurIPS’24

Deep Graph Mating

Yongcheng Jing[1], Seok-Hee Hong[1], Dacheng Tao[2]

[1]The University of Sydney, [2]Nanyang Technological University

THE UNIVERSITY OF
SYDNEY

NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Outline

- □ Background: Graph-Centric Model Reuse
- □ Motivations and Problem Definition
- □ Vanilla Methodologies and Challenge Pre-analysis
- □ Proposed Approach: Dual-Message Coordinator and Calibrator
- □ Experiments
- □ Conclusion

Background: GNNs

□ Graph Neural Networks (GNNs)

- GNNs specialise in capturing the inherent **topological structure** of **irregular graph data** (e.g., molecules and social networks).

□ Challenges in GNNs

- The **scale of graph data** has dramatically increased, often necessitating costly human efforts for **data annotation.**

For example, Twitter user graph, comprising over 288 million nodes and an estimated 208 edges per user[1].

- The **model size** of GNNs has grown substantially, leading to **computationally intensive training**.

Managing such large-scale graphs often demands complex GNN architectures, resulting in significant training efforts and heightened memory requirements[2].

[1] A. Ching, et al. One trillion edges: Graph processing at facebook-scale. Proceedings of the VLDB Endowment (2015)
[2] X. Liu, et al. Survey on graph neural network acceleration: An algorithmic perspective. IJCAI (2022)

Background: GNN Reuse

□ Challenges in GNNs

- The **scale of graph data** has dramatically increased, often necessitating costly human efforts for **data annotation.**
- The **model size** of GNNs has grown substantially, leading to **computationally intensive training**.

□ *Graph-Centric Model Reuse (GNN Reuse)*

- **Goal:** Reuse **Pre-trained** GNNs to:
  - □ Enhance Performance;
  - □ Minimise Training Effort;
  - □ Reduce Human Annotation Effort;
  - □ Improve Inference Speed;
  - □ …

Background: Existing GNN Reuse Works

□ Single-GNN Reuse

- **Graph-Centric Knowledge Distillation[3]:**

  A favourable student GNN is learned from **a single pre-trained teacher.**

□ Multi-GNN Reuse

- **Graph-Centric Knowledge Amalgamation[4]:**

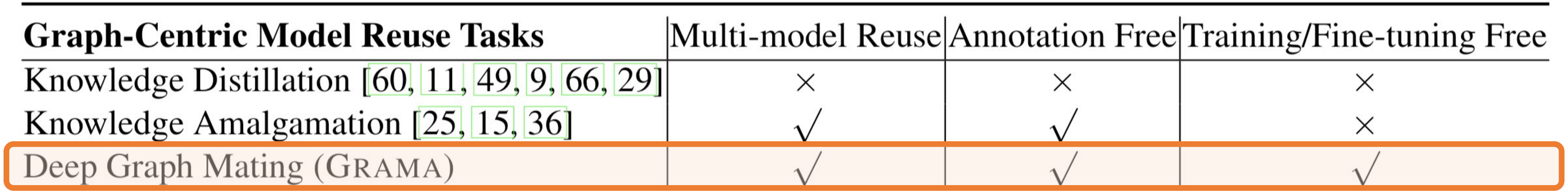
  Learn a single, compact student GNN that integrates the diverse expertise of **multiple pre-trained teacher GNNs, without accessing human annotations.**

[3] Y. Yang, et al. Distilling knowledge from graph convolutional networks. In CVPR, 2020.
[4] Y. Jing, et al. Amalgamating knowledge from heterogeneous graph neural networks. In CVPR, 2021

Motivations and Problem Definition

□ Motivations

- Existing graph-centric model reuse methods inherently limited by the **resource-intensive nature**;
- Requiring the **re-training/fine-tuning** of a student GNN to transfer knowledge from existing GNNs.



Graph-Centric Model Reuse Tasks	Multi-model Reuse	Annotation Free	Training/Fine-tuning Free
Knowledge Distillation [60, 11, 49, 9, 66, 29]	×	×	×
Knowledge Amalgamation [25, 15, 36]	√	√	×
Deep Graph Mating (GRAMA)	√	√	√

Ours:

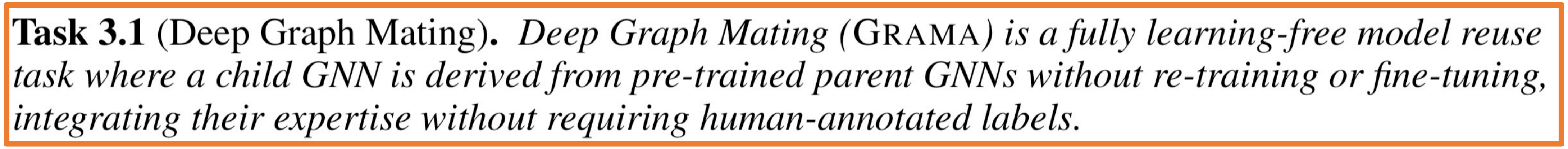
Table 1: Comparison of various model reuse tasks in the non-Euclidean domain, tailored for GNNs.

□ Problem Definition: Our Deep Graph Mating (GRAMA) scheme

Motivations and Problem Definition

□ Problem Definition: Our Deep Graph Mating (GRAMA) scheme

- GRAMA advances beyond existing methods by eliminating the need for **any training or label dependency**, paving the way for more widespread and versatile model reuse applications.



Task 3.1 (Deep Graph Mating). *Deep Graph Mating (GRAMA) is a fully learning-free model reuse task where a child GNN is derived from pre-trained parent GNNs without re-training or fine-tuning, integrating their expertise without requiring human-annotated labels.*

□ Problem Domain

- Given the novelty and complexity of GRAMA, our initial investigation in this paper is confined to scenarios where pre-trained GNNs possess **identical architectures** yet are **trained on separate datasets**, termed as **homogenous GRAMA**.
- We leave heterogeneous GRAMA as a future work.

Vanilla Methodologies

□ Vanilla Methodologies for GRAMA

- We develop **two vanilla approaches** for GRAMA.

❑ Vanilla Parameter Interpolation (VPI):

A straightforward linear interpolation of weights from two pre-trained GNNs: $W^{(\ell)} = \alpha W_a^{(\ell)} + (1 - \alpha) W_b^{(\ell)}$.

Drawback: VPI requires the pre-trained models to share a portion of their training trajectory and remain sufficiently close in the parameter space, typically achieved by fine-tuning from **the same initial model**.

Vanilla Methodologies

□ Vanilla Methodologies for GRAMA

- We develop **two vanilla approaches** for GRAMA.

❑ Vanilla Parameter Interpolation (VPI):

A straightforward linear interpolation of weights from two pre-trained GNNs: $W^{(\ell)} = \alpha W_a^{(\ell)} + (1-\alpha) W_b^{(\ell)}$.

Drawback: VPI requires the pre-trained models to share a portion of their training trajectory and remain sufficiently close in the parameter space, typically achieved by fine-tuning from **the same initial model**.

⇩

❑ Vanilla Alignment Prior to Interpolation (VAPI):

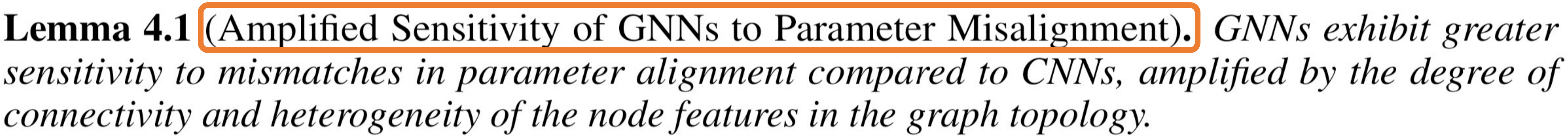
Aligning the neurons between pre-trained models by permuting parameter matrices with permutation matrices $P^{(\ell)}$ before performing linear interpolation: $W^{(\ell)} = \alpha W_a^{(\ell)} + (1-\alpha) P^{(\ell)} W_b^{(\ell)} (P^{(\ell-1)})^T$.

Challenge Analysis

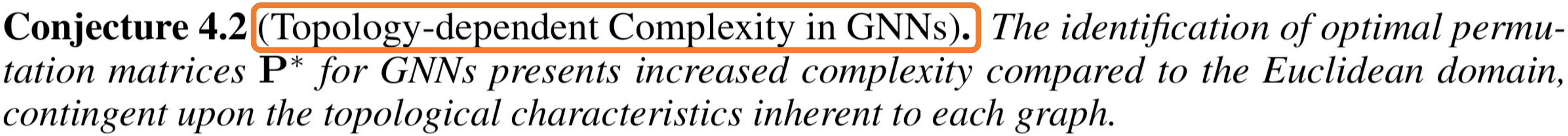
□ Initial Empirical Observation:

- Both vanilla methodologies **(even VAPI)** yielded **unfavourable results** for our GRAMA task.
- Why?

❑ Our theoretical analysis shows:

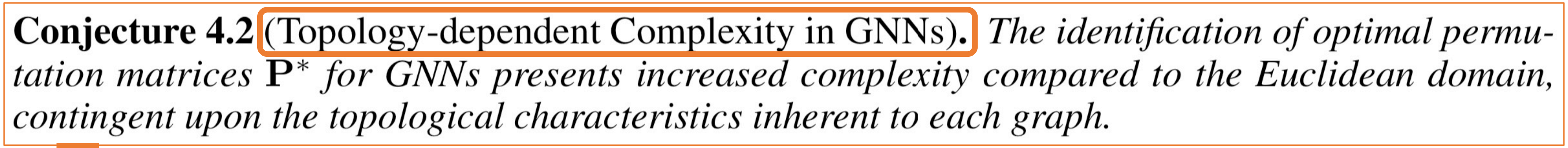


Lemma 4.1 (Amplified Sensitivity of GNNs to Parameter Misalignment)**.** *GNNs exhibit greater sensitivity to mismatches in parameter alignment compared to CNNs, amplified by the degree of connectivity and heterogeneity of the node features in the graph topology.*



Conjecture 4.2 (Topology-dependent Complexity in GNNs)**.** *The identification of optimal permutation matrices* $\mathbf{P}^*$ *for GNNs presents increased complexity compared to the Euclidean domain, contingent upon the topological characteristics inherent to each graph.*

Proposed Approach



Conjecture 4.2 (Topology-dependent Complexity in GNNs). *The identification of optimal permutation matrices* $\mathbf{P}^*$ *for GNNs presents increased complexity compared to the Euclidean domain, contingent upon the topological characteristics inherent to each graph.*

□ Aim of the Proposed Approach

- We propose a **Dual-Message Coordination and Calibration (DuMCC)** methodology.
- DuMCC is specifically designed to harness the unique **topological features of input graphs** for achieving GRAMA without relying on human annotations.

Proposed Approach

□ 1st Part of the Proposed Approach: PMC

- The proposed DuMCC is composed of **two strategic schemes**.
- In particular, the first **Parent Message Coordination (PMC)** scheme effectively integrates **topological information** by deriving optimal permutation matrices from **layer-specific aggregation results**.
- PMC is based on the rationale that **aggregated messages** inherently **encapsulate essential graph topologies**.

⇩

□ Drawback of PMC

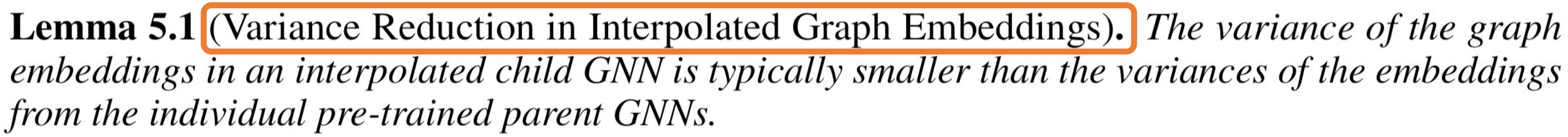
- Empirically, the child GNN, derived from the proposed PMC, exhibits a **reduction in the variance** of node embeddings.

Proposed Approach

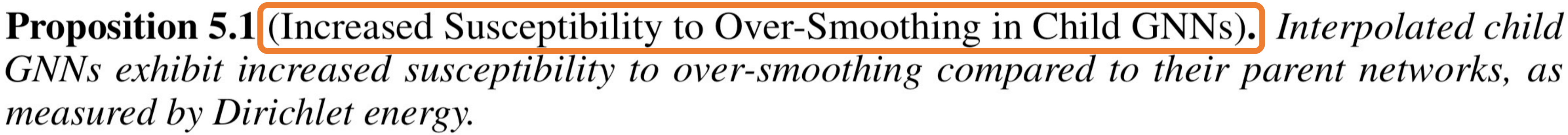
□ Drawback of PMC

- Empirically, the child GNN, derived from the proposed PMC, exhibits a **reduction in the variance** of node embeddings.
- Why?

❑ Theoretical analysis proves that:

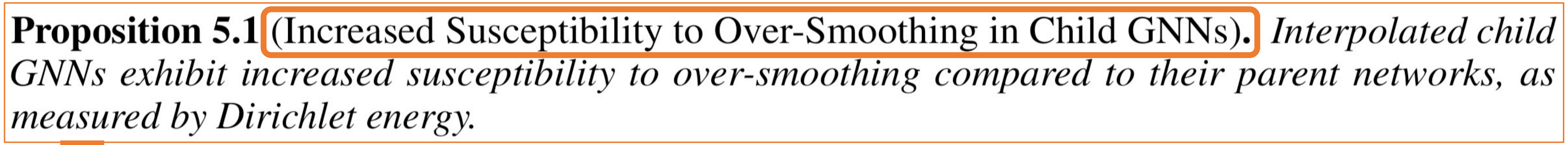


Lemma 5.1 (Variance Reduction in Interpolated Graph Embeddings). *The variance of the graph embeddings in an interpolated child GNN is typically smaller than the variances of the embeddings from the individual pre-trained parent GNNs.*



Proposition 5.1 (Increased Susceptibility to Over-Smoothing in Child GNNs). *Interpolated child GNNs exhibit increased susceptibility to over-smoothing compared to their parent networks, as measured by Dirichlet energy.*

Proposed Approach



Proposition 5.1 (Increased Susceptibility to Over-Smoothing in Child GNNs). *Interpolated child GNNs exhibit increased susceptibility to over-smoothing compared to their parent networks, as measured by Dirichlet energy.*

□ 2nd Part of the Proposed Approach: CMC

- To mitigate the over-smoothing issue identified in the 1st part of our method PMC, we propose a **Child Message Calibration (CMC)** scheme.
- CMC is designed to refine the **message statistics** of the obtained child GNN without the need for re-training or ground-truth labels.
- Central to CMC is our **Learning-Free Message Normalisation (LFNorm)** layer that reduces the risk of over-smoothing in child GNNs by preserving essential topological statistics from the parent models.

Experiments: Implementation

□ Implementation Details

- We adopt the **dataset partition** strategy widely used in model merging within the Euclidean domain;
- Each dataset is randomly split into **two disjoint subsets**: the first subset comprises 20% of the data with odd labels and 80% with even labels, while the second subset is arranged vice versa;
- We set the **interpolation factor** to 0.5 for all experiments to maintain a balanced representation of central tendencies from both pre-trained parent models.

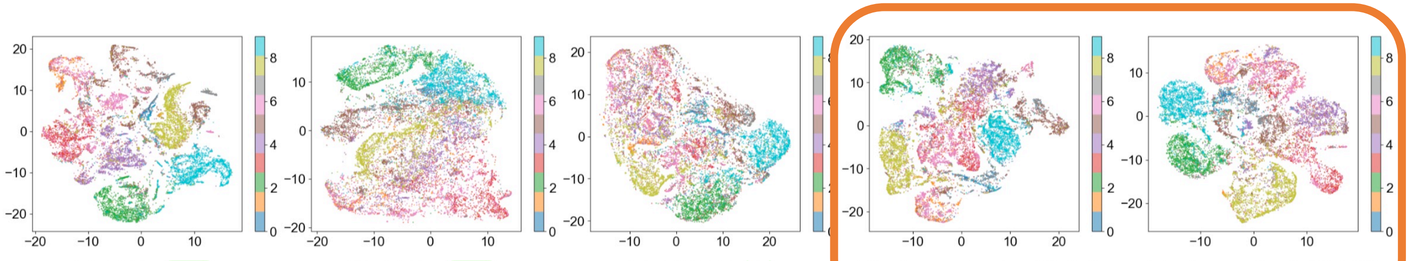
Experiments: Results

Table 2: Multi-class molecule property prediction results for parent GNNs, each pre-trained on disjoint partitions of the `ogbn-arxiv` and `ogbn-products` datasets [18].

Methods	Re-train?	`ogbn-arxiv`		`ogbn-products`	
		Dataset A	Dataset B	Dataset C	Dataset D
Parent GCN A [30]	-	0.7193	0.5516	N/A	N/A
Parent GCN B [30]	-	0.6564	0.7464	N/A	N/A
Parent GraphSAGE C [13]	-	N/A	N/A	0.7982	0.7308
Parent GraphSAGE D [13]	-	N/A	N/A	0.7626	0.7904
KA [25] (Section 3)	√	0.7150	0.6687	0.7973	0.7775
VPI [54] (Section 4)	×	0.3486	0.4361	0.6568	0.6546
VAPI [1] (Section 4)	×	0.6140	0.5752	0.5425	0.5779
Ours (w/o CMC)	×	0.6531	0.5957	0.7374	0.7414
Ours (w/ CMC)	×	**0.6645**	**0.6382**	**0.7647**	**0.7515**

Methods	Re-train?	Dataset I	Dataset J
Parent DGCNN I [53]	-	0.9159	0.8151
Parent DGCNN J [53]	-	0.8862	0.9275
KA [25] (Section 3)	√	0.9250	0.9283
VPI [54] (Section 4)	×	0.4518	0.4096
VAPI [1] (Section 4)	×	0.6538	0.5482
Ours (w/o CMC)	×	0.8326	0.8088
Ours (w/ CMC)	×	**0.8920**	**0.8574**

Table 4: Results of the point cloud classification task on ModelNet40 [55] using DGCNN, with two parent models trained on disjoint partitions.



(a) KA [25] (b) VPI [54] (c) VAPI [1] (d) Ours (w/o CMC) (e) Ours (w/ CMC)

Figure 2: The t-SNE visualisations of various methods on a subset comprising the first 10 classes of `ogbn-arxiv`. Additional visualisations for the remaining classes are available in Appendix E.

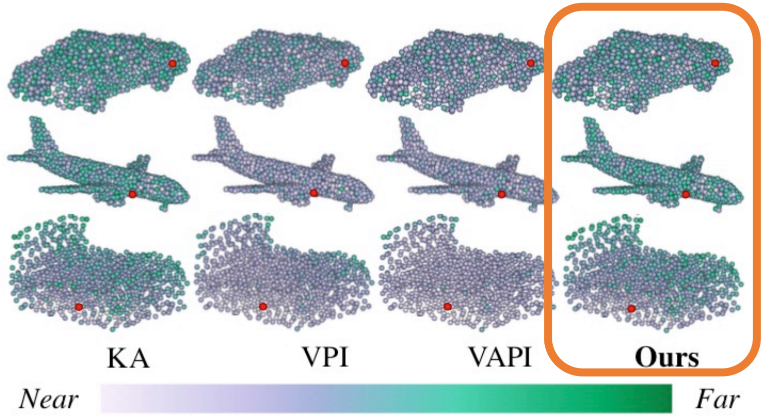


Figure 3: Visualisations of feature space structures, depicted by the distances between the red point and all other points.

Conclusions and Limitations

- □ In this work, we explore a novel GRAMA task for **learning-free GNN reuse.**
- □ Uniquely, GRAMA establishes the **first paradigm** in GNN reuse that operates entirely without re-training or fine-tuning, while also eliminating the need for ground-truth labels**.**
- □ Despite its strengths, the proposed method is primarily designed for homogeneous GRAMA.
- □ Our approach does not support cross-architecture heterogeneous GRAMA, where parent models have different architectures, such as a combination of GCN and GraphSAGE, an issue we plan to explore in our future work.

Thanks！