Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift

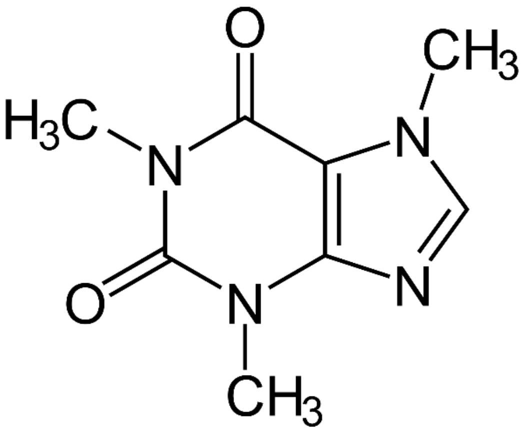
Yongduo Sui, Qitian Wu, Jiancan Wu, Qing Cui, Longfei Li,
Jun Zhou, Xiang Wang, Xiangnan He

1.1 Background

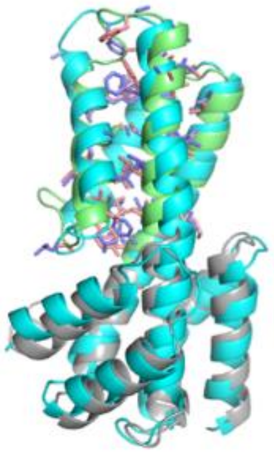
- Graph data are everywhere
  - Social network
  - Chemical molecule
  - Biological protein

- Graph learning tasks
  - Node classification
  - Link prediction
  - Graph classification

Social network



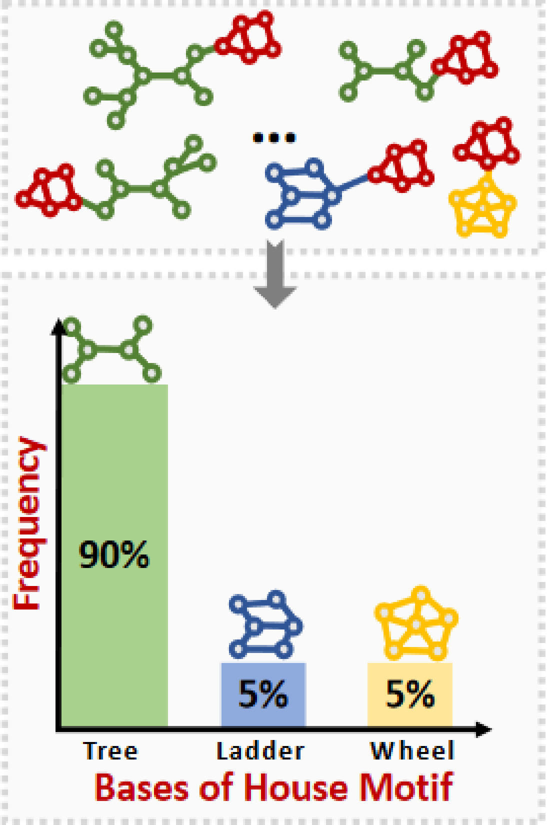
Molecule

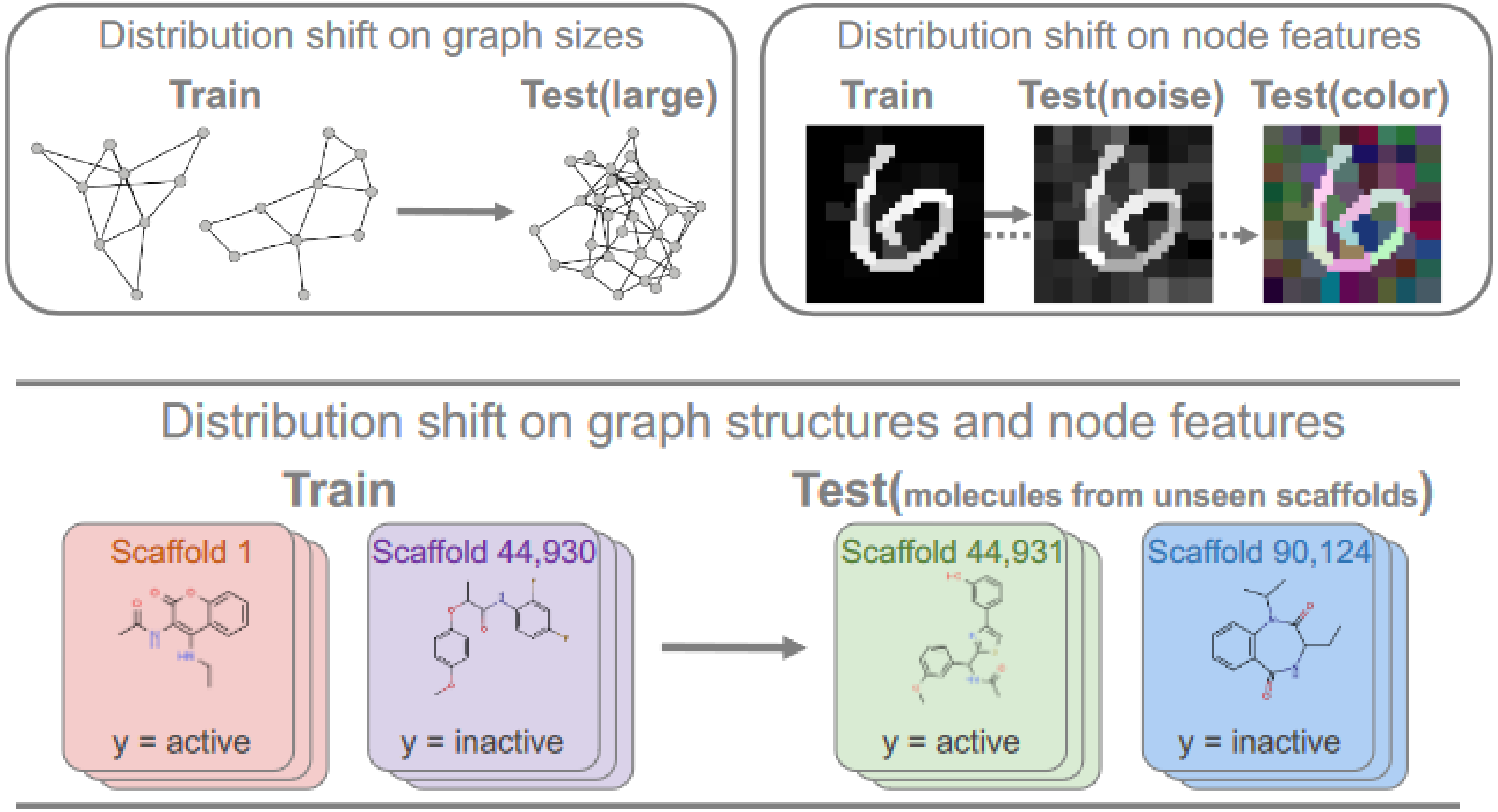


Protein structure

1.2 Graph Out-of-distribution Issue

☐ OOD Issue in Graph Classification





[1] Discovering Invariant Rationales for Graph Neural Networks, ICLR 2022

[2] OOD-GNN: Out-of-Distribution Generalized Graph Neural Network, TKDE 2022

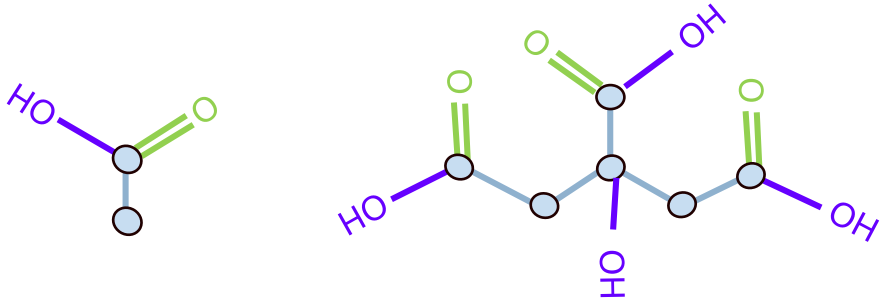
2.1 Assumption of Graph Generation

Stable (*aka. Causal, Invariant, Rationale*) Feature & Environmental Feature

Stable feature: functional group, e.g. –OH, -COOH
Environmental feature: scaffold, e.g. carbon ring, carbon chain

Molecule: *Cyclopropanol*
Scaffold: *3-carbon ring*

Molecule: *1, 4-cyclohexanediol*
Scaffold: *6-carbon ring*



Molecule: *acetic acid*
Scaffold: *small size*

Molecule: *citric acid*
Scaffold: *large size*

Sufficiency & Invariance Assumption

Assumption 3.1. *Given* G, *there exists an optimal invariant subgraph generator* $\Phi^*(\mathrm{G})$ *satisfying:*
a. Invariance property*:* $\forall e, e' \in \mathrm{supp}(\mathcal{E}), P^e(\mathrm{Y}|\Phi^*(\mathrm{G})) = P^{e'}(\mathrm{Y}|\Phi^*(\mathrm{G}))$.
b. Sufficiency property*:* $\mathrm{Y} = w^*(g^*(\Phi^*(\mathrm{G}))) + \epsilon,\ \epsilon \perp \mathrm{G}$, *where* $g^*(\cdot)$ *denotes a representation learning function,* w^* *is the classifier,* $\perp$ *indicates statistical independence, and* ϵ *is random noise.*

[4] Learning Invariant Graph Representations for Out-of-Distribution Generalization, NeurIPS 2022
[5] Learning Substructure Invariance for Out-of-Distribution Molecular Representations, NeurIPS 2022

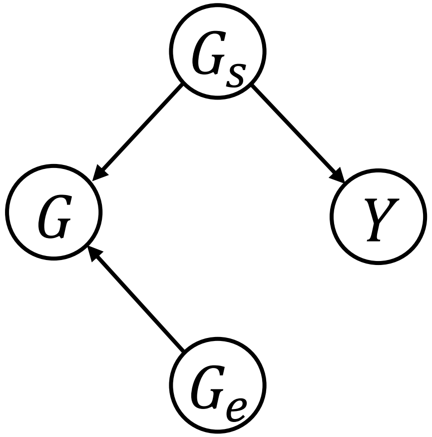
3.1 Two Types of Distribution Shifts

- Correlation Shift v.s. Covariate Shift

➢ The joint distribution of training and test data as $P_{\text{tr}}(G, Y)$ and $P_{te}(G, Y)$

Distribution shift means that $P_{\text{tr}}(G, Y) \neq P_{te}(G, Y)$

$$P_{\text{tr}}(G, Y) = P_{\text{tr}}(Y|G)\ P_{\text{tr}}(G)$$
$$P_{te}(G, Y) = P_{te}(Y|G)\ P_{te}(G)$$



Graph Data Generation

➢ Correlation shift: $P_{\text{tr}}(G) = P_{te}(G)$ but $P_{\text{tr}}(Y|G) \neq P_{te}(Y|G)$

➢ Covariate shift: $P_{\text{tr}}(G) \neq P_{te}(G)$ but $P_{\text{tr}}(Y|G) = P_{te}(Y|G)$

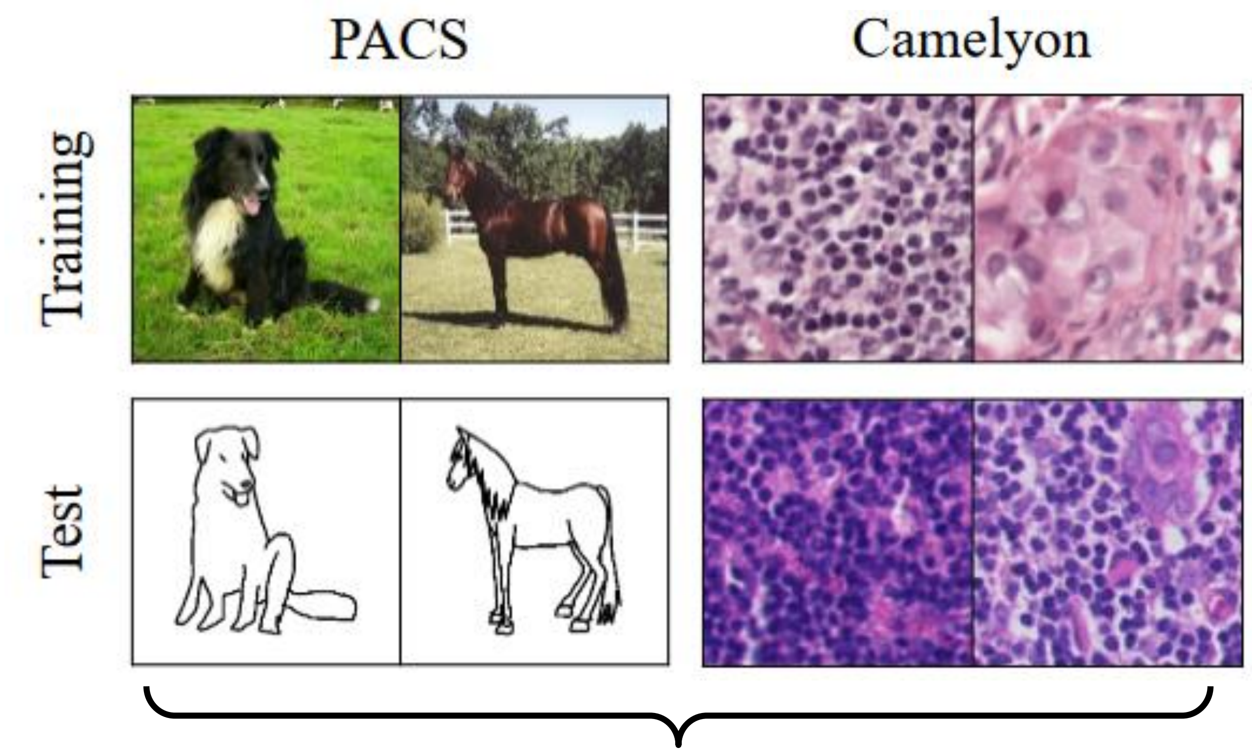
Our scope: these distribution shifts mainly caused by the environmental features.

[1] OoD-Bench: Quantifying and Understanding Two Dimensions of Out-of-Distribution Generalization, CVPR 2022

[6] GOOD: A Graph Out-of-Distribution Benchmark, NeurIPS 2022

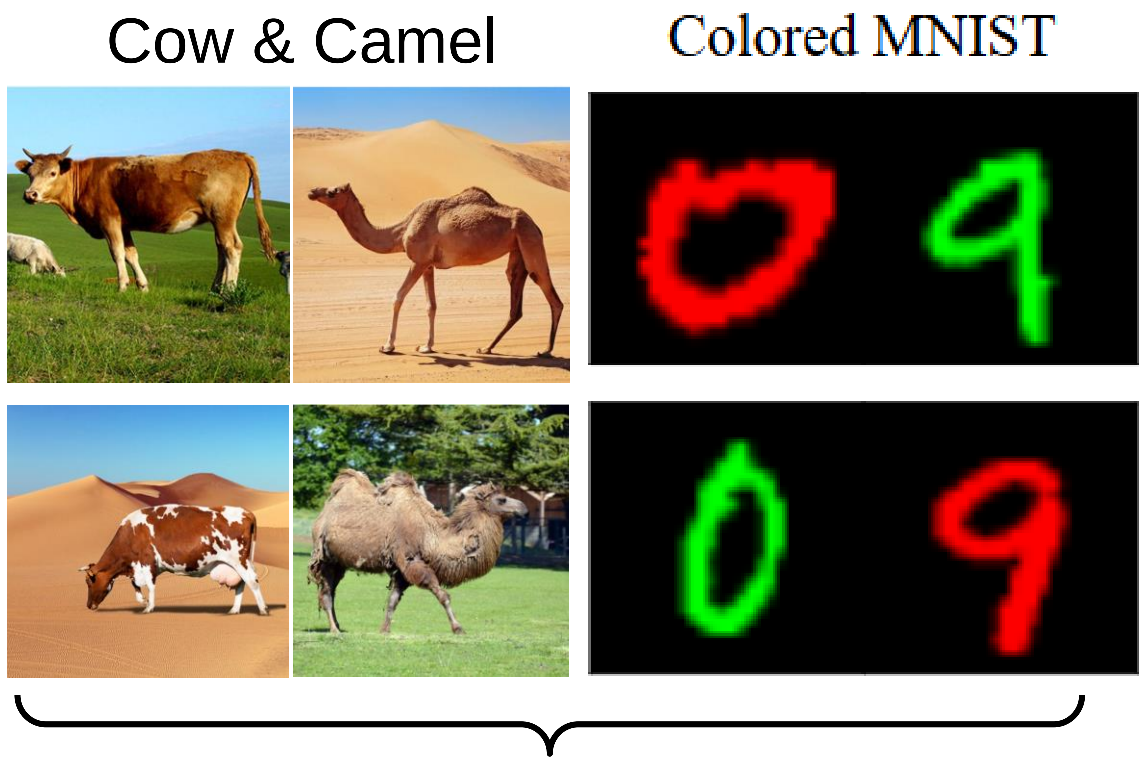
3.1 Two Types of Distribution Shifts

- Correlation Shift v.s. Covariate Shift



Covariate (diversity) shift

e.g. Domain Generalization

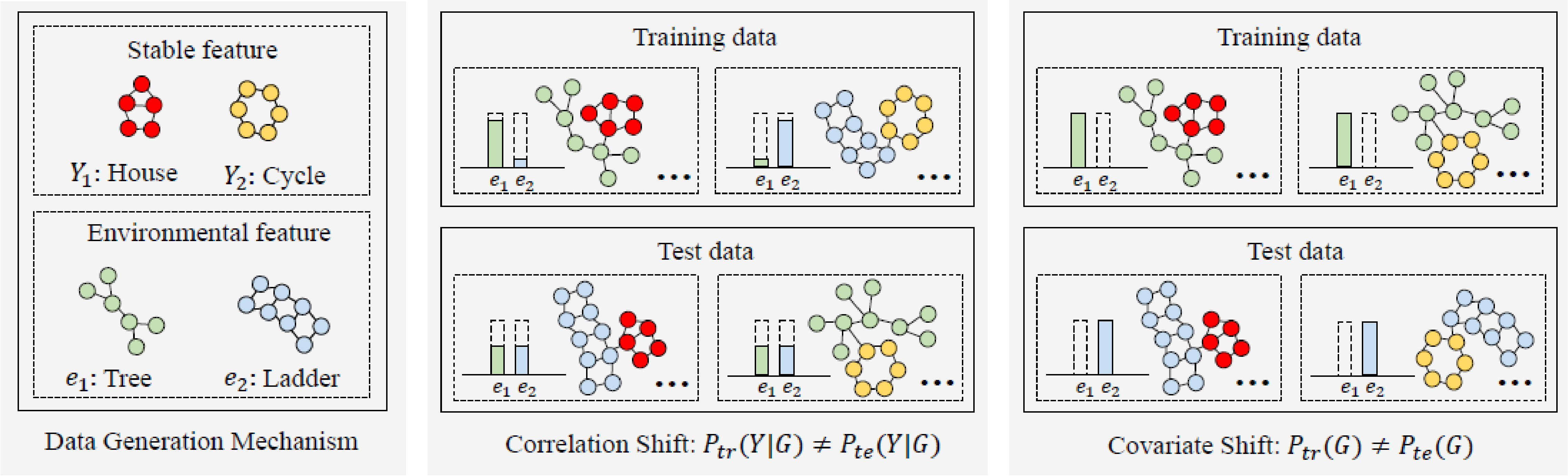


Correlation (concept) shift

[1] OoD-Bench: Quantifying and Understanding Two Dimensions of Out-of-Distribution Generalization, CVPR 2022

3.1 Two Types of Distribution Shifts

- Correlation Shift v.s. Covariate Shift



Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.2 Related Studies

Methods	Limitations
General Generalization Algorithms *Zhang, et al, ICLR' 18,* *Sagawa, et al, ICLR' 20,* *Arjovsky, et al, Arxiv' 19.*	Due to the irregularity of graph data, they are difficult to achieve significant performance improvements.
Graph Data Augmentation *Rong, et al, ICLR' 2020,* *You, et al, NeurIPS' 2020,* *Han, et al, ICML' 2022.*	They are prone to destroy the stable features in the data, resulting in the insensitivity to stable features.
Graph Invariant Learning *Wu, et al, ICLR' 2022(a),* *Wu, et al, ICLR' 2022(b),* *Li, et al, NeurIPS' 2022,* *Sui, et al, KDD' 2022.*	They are difficult to improve the environmental discrepancy.

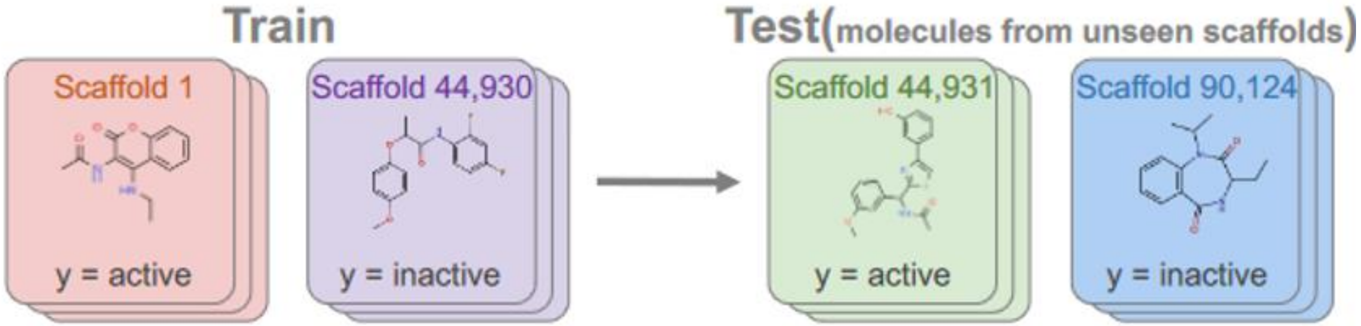
3.3 How to Address Covariate Shift on Graphs?

Existing Issue: Insufficient discrepancy of environmental features

- Graph Covariate Shift

$$\mathrm{GCS}(P_{\mathrm{tr}}, P_{\mathrm{te}}) = \frac{1}{2}\int_{\mathcal{S}} |P_{\mathrm{tr}}(g) - P_{\mathrm{te}}(g)| dg$$

$$\mathcal{S} = \{g \in \mathbb{G} | P_{\mathrm{tr}}(g) \cdot P_{\mathrm{te}}(g) = 0\}$$



- Our Idea
  - Using data augmentation to increase the environmental discrepancy

Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.3 How to Address Covariate Shift on Graphs?

Two Principles for Graph Augmentation

- **Principle 1** (Environmental Feature Discrepancy): Environmental features should remain discrepant during augmentation

> **Principle 3.1 (Environmental Feature Discrepancy)** *Given a graph set $\{g\}$ with distribution function P, let $T(\cdot)$ denote an augmentation function that augments graphs $\{T(g)\}$ to distribution $\widetilde{P}$. Then $T(\cdot)$ should meet* $\mathrm{GCS}(P, \widetilde{P}) \to 1$.

- **Principle 2** (Stable Feature Consistency): Stable features should remain consistent during augmentation

> **Principle 3.2 (Stable Feature Consistency)** *Given a set of graphs $\{g\}$ with a corresponding stable feature set $\{g_{\mathrm{sta}} = (\mathbf{A}_{\mathrm{sta}}, \mathbf{X}_{\mathrm{sta}})\}$. Let $T(\cdot)$ denote an augmentation function that augments graphs $\{T(g)\}$ with a corresponding stable feature set $\{\widetilde{g}_{\mathrm{sta}} = (\widetilde{\mathbf{A}}_{\mathrm{sta}}, \widetilde{\mathbf{X}}_{\mathrm{sta}})\}$. Then $T(\cdot)$ should meet* $\mathbb{E}[\|\mathbf{A}_{\mathrm{sta}} - \widetilde{\mathbf{A}}_{\mathrm{sta}}\|_F^2] \to 0$ *and* $\mathbb{E}[\|\mathbf{X}_{\mathrm{sta}} - \widetilde{\mathbf{X}}_{\mathrm{sta}}\|_F^2] \to 0$, *where $\|\cdot\|_F$ is the Frobenius norm.*

Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.4 Adversarial Invariant Augmentation

- Distributionally Robust Optimization

$$\min_{\theta}\left\{\sup_{\widetilde{P}}\{\mathbb{E}_{\widetilde{P}}[\ell(f(g),y)] : D(\widetilde{P},P)\le\rho\}\right\},$$

- Wasserstein Distance

$$D(\widetilde{P},P) := \inf_{\mu\in\Gamma(\widetilde{P},P)}\mathbb{E}_{\mu}[c(\widetilde{g},g)],$$

- Transportation Cost

$$c(\widetilde{g},g) = \|h(\widetilde{g})-h(g)\|_2^2.$$

- Lagrangian relaxation

$$\min_{\theta}\left\{\sup_{\widetilde{P}}\{\mathbb{E}_{\widetilde{P}}[\ell(f(g),y)]-\gamma D(\widetilde{P},P)\} = \mathbb{E}_{P}[\phi(f(g),y)]\right\},$$

Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.4 Adversarial Invariant Augmentation

- Robust surrogate loss： $\phi(f(g), y)$

$$\phi(f(g), y) := \sup_{\widetilde{g}\in\mathbb{G}}\{\ell(f(\widetilde{g}), y) - \gamma c(\widetilde{g}, g)\}$$

- Adversarial Augmentation

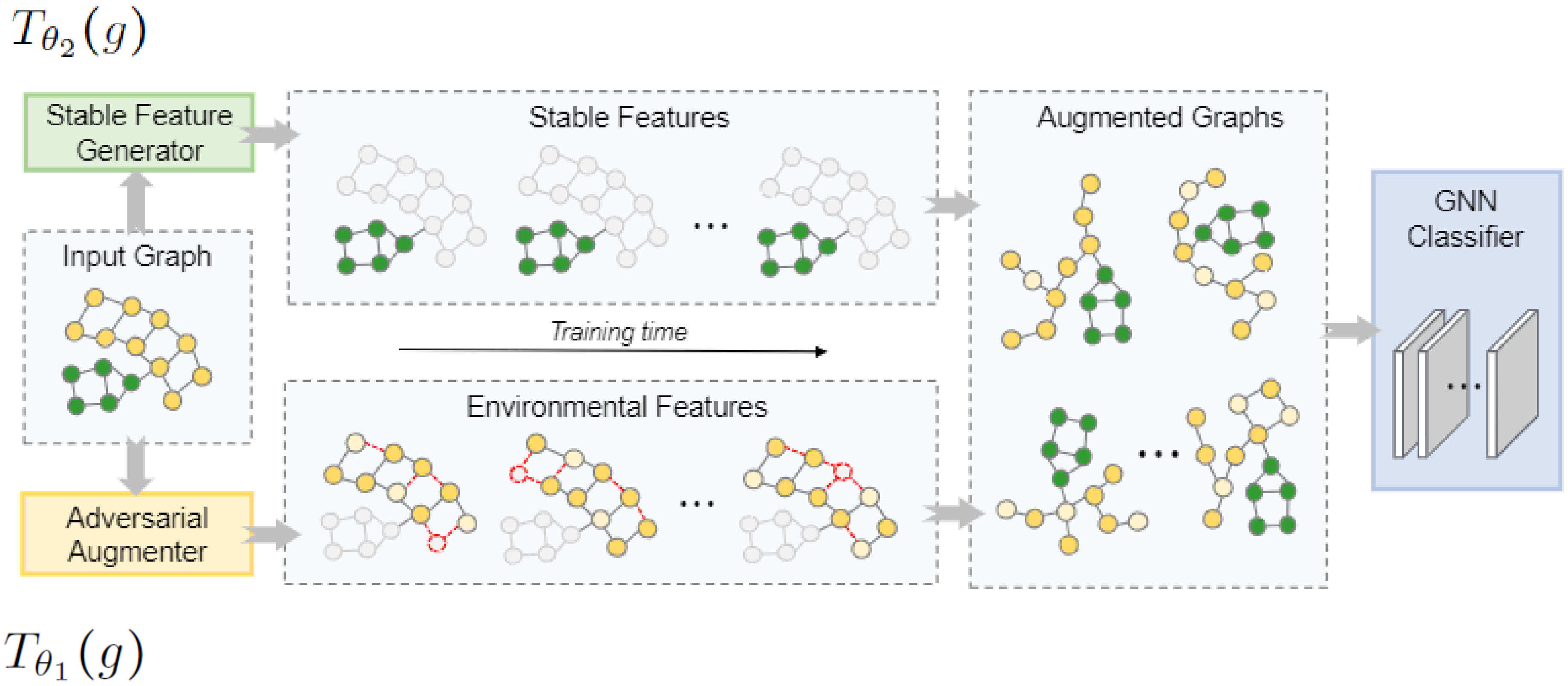
$$\nabla_\theta \phi(f(g), y) = \nabla_\theta \ell(f(\widetilde{g}^*), y),$$
$$\text{where } \widetilde{g}^* = \arg\max_{\widetilde{g}\in\mathbb{G}}\{\ell(f(\widetilde{g}), y) - \gamma c(\widetilde{g}, g)\}.$$

Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.4 Adversarial Invariant Augmentation

- Adversarial Augmenter & Stable Feature Generator

$$T_{\theta_1}(g) = (\mathbf{A} \odot \mathbf{M}^{a}_{\text{adv}}, \mathbf{X} \odot \mathbf{M}^{x}_{\text{adv}})$$



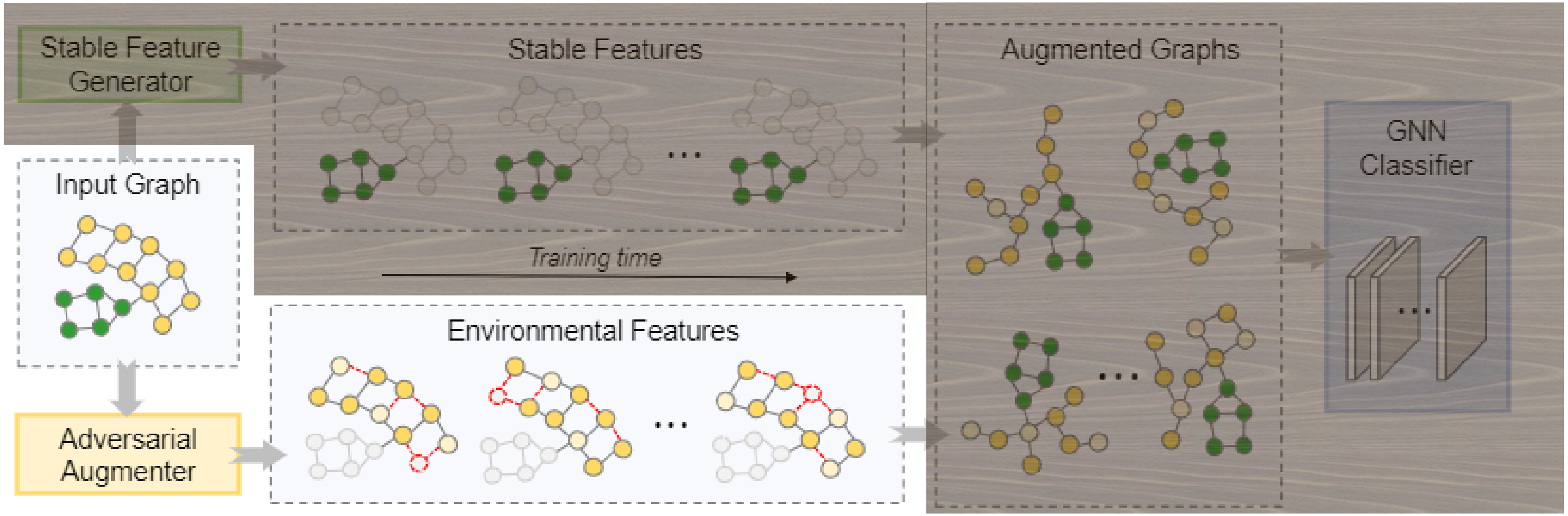
Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.4 Adversarial Invariant Augmentation

- Maximization

$$\max_{\theta_1} \{\mathcal{L}_{\text{adv}} = \mathbb{E}_{P_{\text{tr}}}[\ell(f(T_{\theta_1}(g)), y) - \gamma c(T_{\theta_1}(g), g)]\}$$

$T_{\theta_2}(g)$



$T_{\theta_1}(g)$

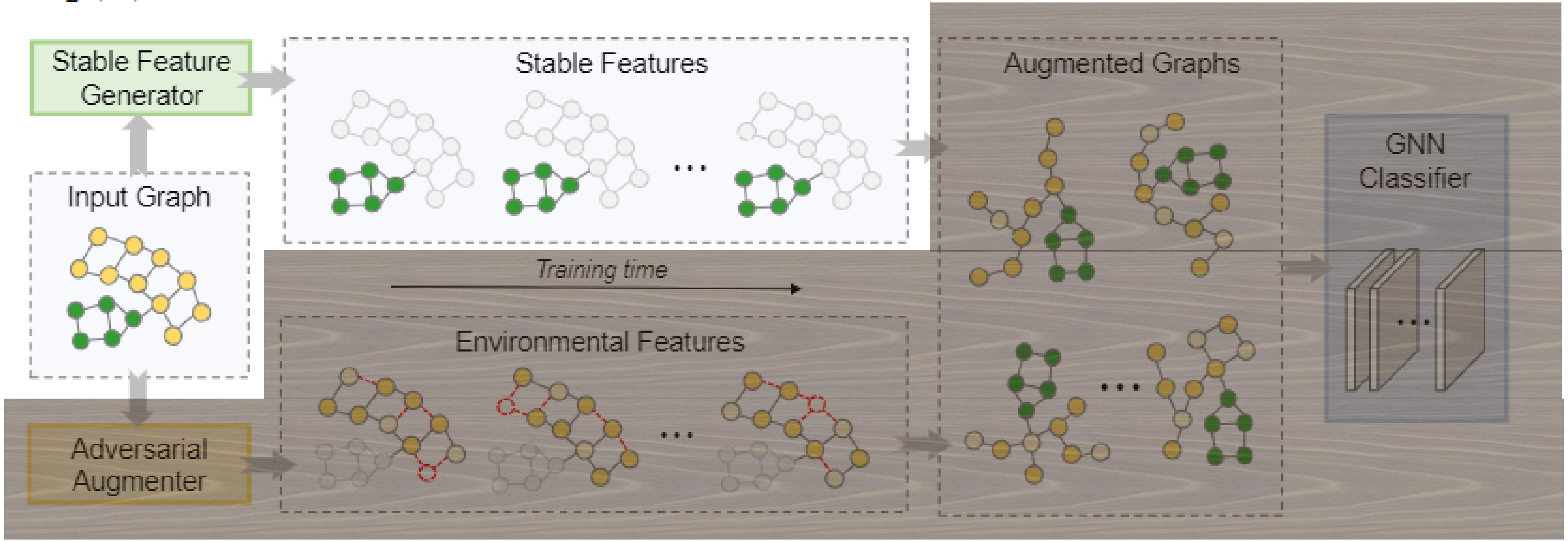
Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.4 Adversarial Invariant Augmentation

□ Minimization

$$\min_{\theta,\theta_2}\{\mathcal{L}_{\text{cau}} = \mathbb{E}_{P_{\text{tr}}}[\ell(f(T_{\theta_2}(g)),y) + \ell(f(\widetilde{g}),y)]\} \begin{cases} \widetilde{g} = (\mathbf{A}\odot\widetilde{\mathbf{M}}^a, \mathbf{X}\odot\widetilde{\mathbf{M}}^x) \\ \widetilde{\mathbf{M}}^a = (\mathbf{1}^a - \mathbf{M}^a_{\text{cau}})\odot\mathbf{M}^a_{\text{adv}} + \mathbf{M}^a_{\text{cau}} \\ \widetilde{\mathbf{M}}^x = (\mathbf{1}^x - \mathbf{M}^x_{\text{cau}})\odot\mathbf{M}^x_{\text{adv}} + \mathbf{M}^x_{\text{cau}} \end{cases}$$

$T_{\theta_2}(g)$



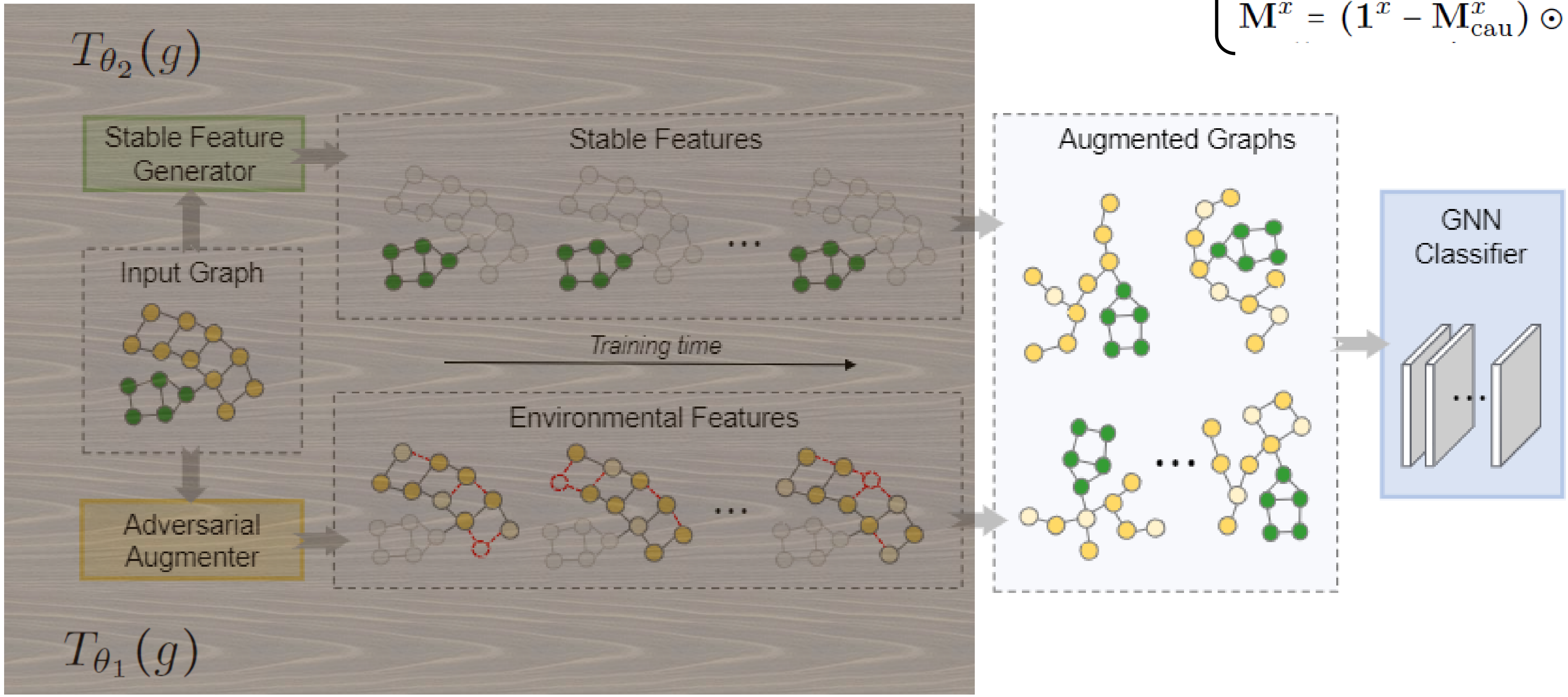
$T_{\theta_1}(g)$

Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

3.4 Adversarial Invariant Augmentation

- Mask Combination

$$\min_{\theta,\theta_2} \{\mathcal{L}_{\text{cau}} = \mathbb{E}_{P_{\text{tr}}}[\ell(f(T_{\theta_2}(g)), y) + \ell(f(\widetilde{g}), y)]\} \begin{cases} \widetilde{g} = (\mathbf{A} \odot \widetilde{\mathbf{M}}^a, \mathbf{X} \odot \widetilde{\mathbf{M}}^x) \\ \widetilde{\mathbf{M}}^a = (\mathbf{1}^a - \mathbf{M}^a_{\text{cau}}) \odot \mathbf{M}^a_{\text{adv}} + \mathbf{M}^a_{\text{cau}} \\ \widetilde{\mathbf{M}}^x = (\mathbf{1}^x - \mathbf{M}^x_{\text{cau}}) \odot \mathbf{M}^x_{\text{adv}} + \mathbf{M}^x_{\text{cau}} \end{cases}$$



Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

4.1 Experiments: Main Results

Table 1: Performance on synthetic and real-world datasets. Numbers in **bold** indicate the best performance, while the underlined numbers indicate the second best performance.

Type	Method	Motif		CMNIST	Molbbbp		Molhiv	
		base	size	color	scaffold	size	scaffold	size
General Generalization	ERM	68.66±4.25	51.74±2.88	28.60±1.87	68.10±1.68	78.29±3.76	69.58±2.51	59.94±2.37
	IRM	70.65±4.17	51.41±3.78	27.83±2.13	67.22±1.15	77.56±2.48	67.97±1.84	59.00±2.92
	GroupDRO	68.24±8.92	51.95±5.86	29.07±3.14	66.47±2.39	79.27±2.43	70.64±2.57	58.98±2.16
	VREx	71.47±6.69	52.67±5.54	28.48±2.87	68.74±1.03	78.76±2.37	70.77±2.84	58.53±2.88
Graph Generalization	DIR	62.07±8.75	52.27±4.56	33.20±6.17	66.86±2.25	76.40±4.43	68.07±2.29	58.08±2.31
	CAL	65.63±4.29	51.18±5.60	27.99±3.24	68.06±2.60	79.50±4.81	67.37±3.61	57.95±2.24
	GSAT	62.80±11.41	53.20±8.35	28.17±1.26	66.78±1.45	75.63±3.83	68.66±1.35	58.06±1.98
	OOD-GNN	61.10±7.87	52.61±4.67	26.49±2.94	66.72±1.23	79.48±4.19	70.46±1.97	60.60±3.77
	StableGNN	57.07±14.10	46.93±8.85	28.38±3.49	66.74±1.30	77.47±4.69	68.44±1.33	56.71±2.79
	CIGA	66.43±11.31	49.14±8.34	32.22±2.67	64.92±2.09	65.98±3.31	69.40±2.39	59.55±2.56
	DisC	51.08±3.08	50.39±1.15	24.99±1.78	67.12±2.11	56.59±10.09	68.07±1.75	58.76±0.91
Graph Augmentation	DropEdge	45.08±4.46	45.63±4.61	22.65±2.90	66.49±1.55	78.32±3.44	70.78±1.38	58.53±1.26
	GREA	56.74±9.23	54.13±10.02	29.02±3.26	69.72±1.66	77.34±3.52	67.79±2.56	60.71±2.20
	FLAG	61.12±5.39	51.66±4.14	32.30±2.69	67.69±2.36	79.26±2.26	68.45±2.30	60.59±2.95
	M-Mixup	70.08±3.82	51.48±4.91	26.47±3.45	68.75±0.34	78.92±2.43	68.88±2.63	59.03±3.11
	$\mathcal{G}$-Mixup	59.66±7.03	52.81±6.73	31.85±5.82	67.44±1.62	78.55±4.16	70.01±2.52	59.34±2.43
	AIA (ours)	**73.64±5.15**	**55.85±7.98**	**36.37±4.44**	**70.79±1.53**	**81.03±5.15**	**71.15±1.81**	**61.64±3.37**

Unleashing the Power of Graph Data Augmentation on Covariate Distribution Shift, NeurIPS 2023

5 Conclusion

- We aim to address the covariate shift issue in graph learning, which is important yet largely unexplored.

- We introduce a novel graph augmentation method, AIA, grounded in two principles: environmental feature discrepancy and stable feature consistency.

- We conduct extensive experiments and the results demonstrate the effectiveness of our method.