中国科学院自动化研究所
模式识别实验室
New Laboratory of Pattern Recognition

中国科学院自动化研究所
Institute of Automation
Chinese Academy of Sciences

Molecular Representation Learning and Property Prediction

Liang Wang

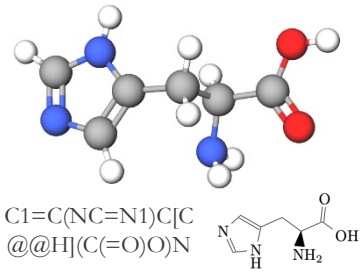
Advisor: Prof. Liang Wang
Institute of Automation, Chinese Academy of Sciences
1 November 2024

Outline

Research Background

Molecular Generation, Molecular Optimization,
Molecular Spectroscopy Analysis

Molecular Structure



$f: u \rightarrow \mathbb{R}^d$

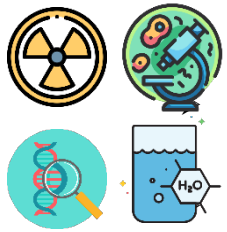
Molecular Representation

vector

$\mathbb{R}^d$

Feature representation, embedding

Molecular Properties



Solubility
Toxicity
Blood-Brain Barrier Permeability
Biological Activity
.....

Application Fields

- Drug discovery
- Materials science
- Biotechnology

Chemical Reaction, Retrosynthesis Planning,
Intermolecular Interactions, Target-Ligand Interaction

Molecular Representation Learning

Translate the molecular structures into vectorized molecular representations to understand and predict various molecular properties, interactions, chemical reactions.

$$h = f(molecule)$$

Molecular Property Prediction

Based on molecular representations, predict various physical, chemical, and biological molecular properties, to guide scientific research and industrial applications.

$$< properties >= g(h)$$

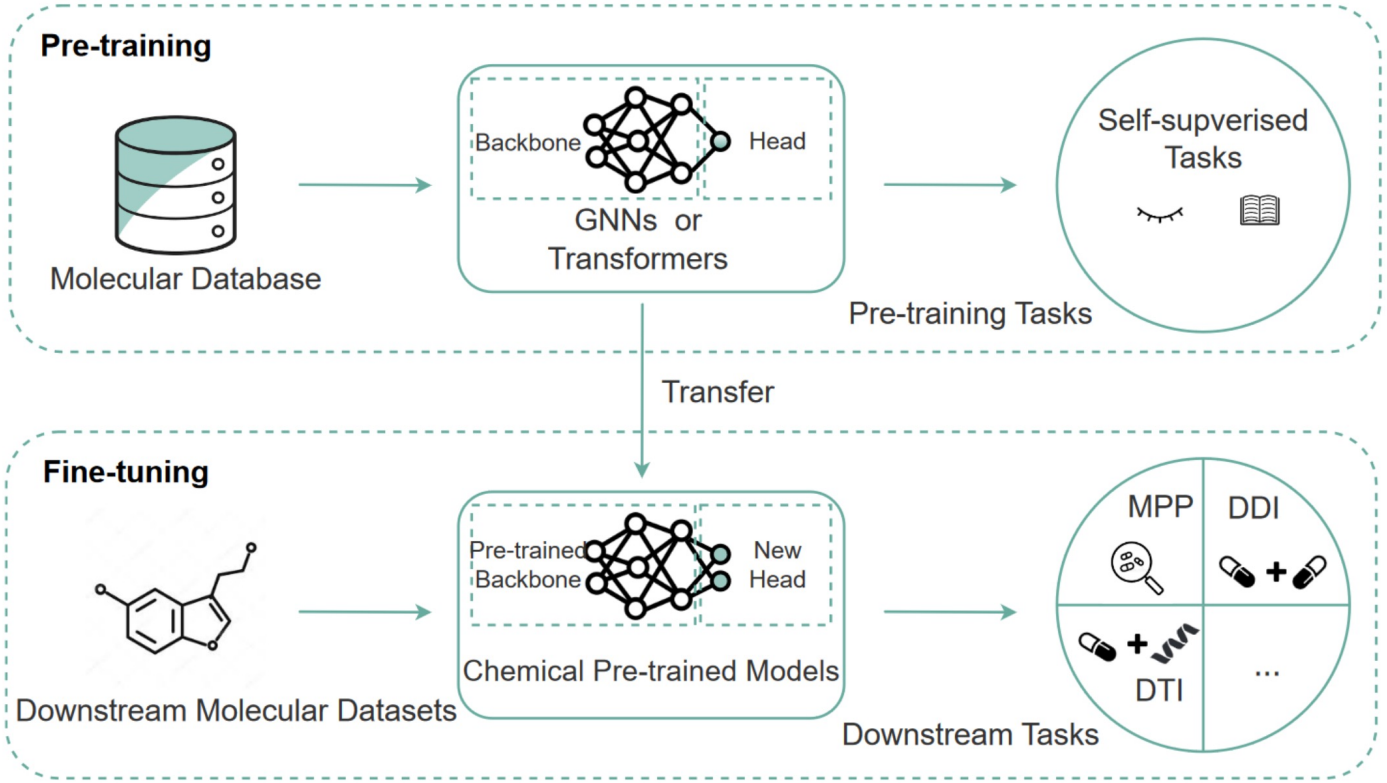
$$p_1, \ldots, p_{|P|} = g_1(h), \ldots, g_{|P|}(h)$$

Challenges of supervised molecular representation learning

(1) Scarcity of labeled data.

(2) Poor out-of-distribution generalization capability.

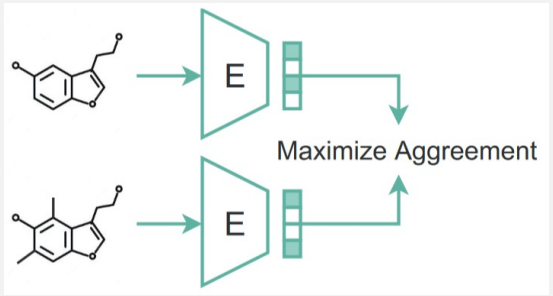
Pipeline of Molecular Representation Pre-training



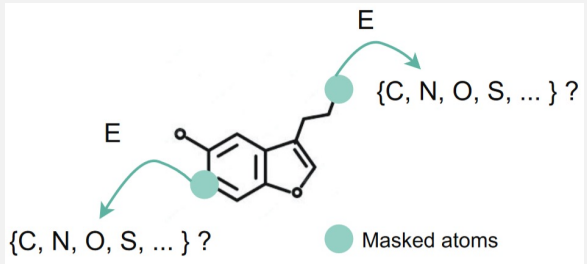
- ✓ Pre-trained on large-scale unlabeled molecules.
- ✓ Fine-tuned on various downstream tasks.

Self-supervised Strategies

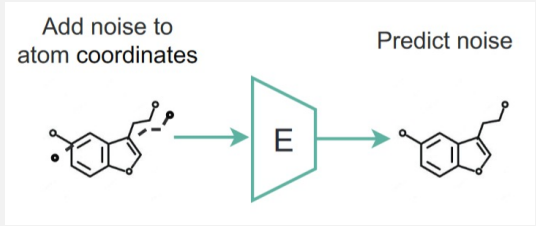
Contrastive Learning



Masked Components Modeling



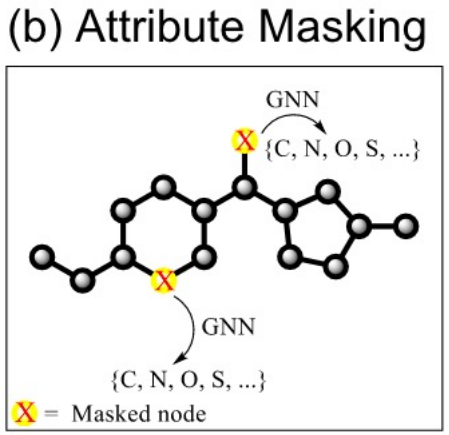
Denoising



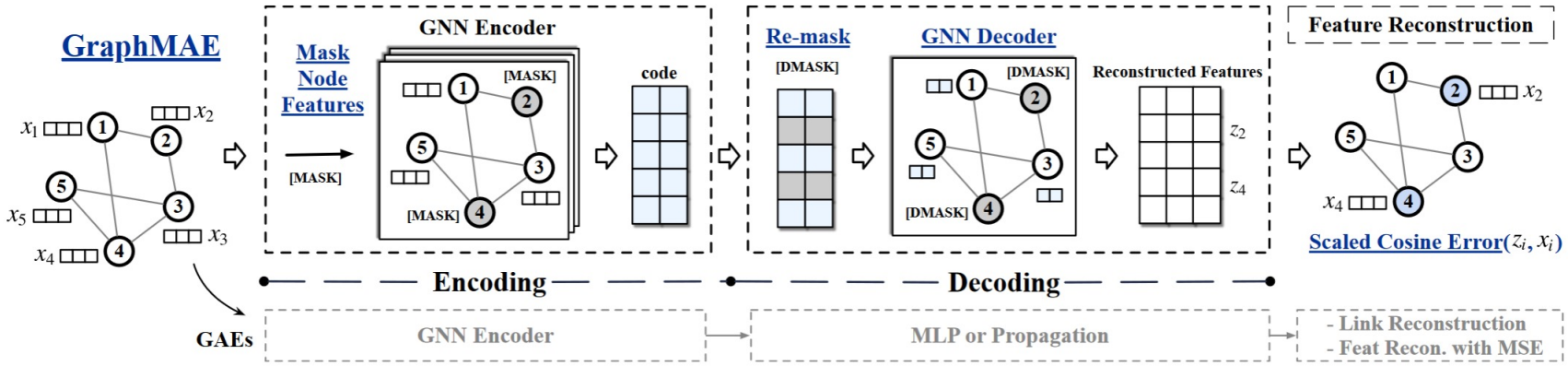
Representative Work:

- GraphCL (NeurIPS 2020)
- MoCL (KDD 2021)
- MolCLR (NMI 2022)
- AttrMasking (ICLR 2020)
- GraphMAE (KDD 2022)
- Mole-BERT (ICLR 2023)
- Uni-Mol (ICLR 2023)
- Coord (ICLR 2023)
- SliDe (ICLR 2024)

Masked Components Modeling



AttrMasking (ICLR 2020)



GraphMAE (KDD 2022)

1. **Linear decoder -> GNN decoder**
2. **Remask**
3. **Cross entropy loss -> scaled cosine error (SCE) loss**

$$\mathcal{L}_{\mathrm{SCE}} = \mathbb{E}_{v_i \in \tilde{\mathcal{V}}} \left(1 - \boldsymbol{x}_i{}^{\top} h(c_i)\right)^{\gamma}$$

Background

- **Denoising as learning a force field.**
  - It is not feasible to learn the molecular force field directly, because it is either unknown or expensive to evaluate.
  - Alternative: approximate the data-generating force field with one that can be cheaply evaluated.
  - Prove that the denoising objective is equivalent to learning the molecular force field:
    - Molecular structure: $\mathbf{x} \in \mathbb{R}^{3N}$
    - The structure follows the Boltzmann distribution: $p_{\text{physical}}(\mathbf{x}) \propto \exp(-E(\mathbf{x}))$
    - Force field: $\nabla_{\mathbf{x}} \log p_{\text{physical}}(\mathbf{x}) = -\nabla_{\mathbf{x}} E(\mathbf{x})$
    - Approximate p_{physical} with a mixture of Gaussians centered at the known equilibrium structures

$$p_{\text{physical}}(\tilde{\mathbf{x}}) \approx q_\sigma(\tilde{\mathbf{x}}) := \frac{1}{n} \sum_{i=1}^{n} q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}_i)$$

where $q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}_i) = \mathcal{N}(\tilde{\mathbf{x}}; \mathbf{x}_i, \sigma^2 I_{3N})$

Background

- **Denoising as learning a force field. (Cont.)**
  - Learning the force field now yields a score-matching objective:

$$\mathbb{E}_{q_\sigma(\tilde{\mathbf{x}})}\left[\left\|\mathrm{GNN}_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}})\right\|^2\right] \tag{1}$$

  - According to reference [1], minimizing the following two objectives is equivalent:

$$J_1(\theta) = \mathbb{E}_{q_\sigma(\tilde{\mathbf{x}})}\left[\left\|\mathrm{GNN}_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}})\right\|^2\right]$$

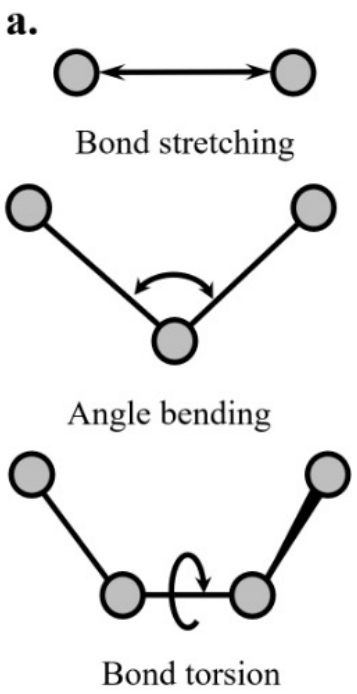
$$J_2(\theta) = \mathbb{E}_{q_\sigma(\tilde{\mathbf{x}},\mathbf{x})}\left[\left\|\mathrm{GNN}_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})\right\|^2\right]$$

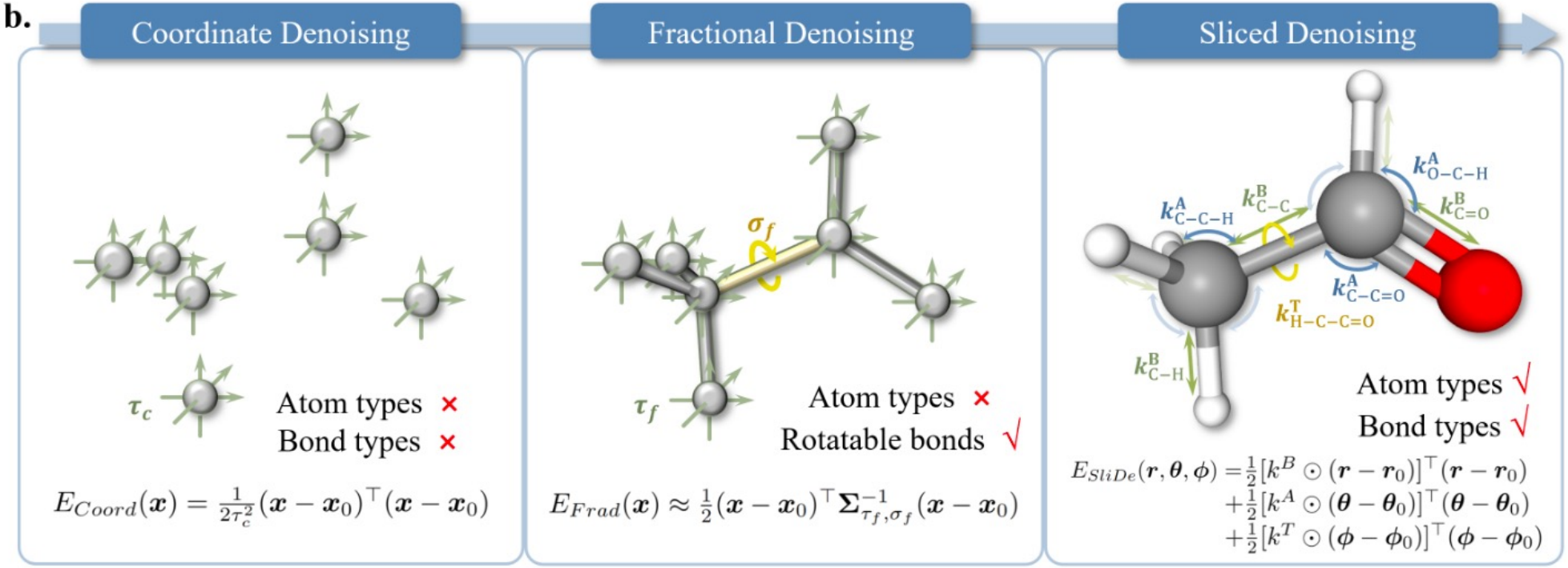
  - Thus, the objective in Eq. (1) is equivalent to:

$$\mathbb{E}_{q_\sigma(\tilde{\mathbf{x}},\mathbf{x})}\left[\left\|\mathrm{GNN}_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})\right\|^2\right] = \mathbb{E}_{q_\sigma(\tilde{\mathbf{x}},\mathbf{x})}\left[\left\|\mathrm{GNN}_\theta(\tilde{\mathbf{x}}) - \frac{\mathbf{x}-\tilde{\mathbf{x}}}{\sigma^2}\right\|^2\right]$$

[1] Pascal Vincent. **“A Connection Between Score Matching and Denoising Autoencoders.”** *Neural Computation* July 2011

Background


a.
Bond stretching
Angle bending
Bond torsion


b.
Coordinate Denoising
Fractional Denoising
Sliced Denoising
τ_c
Atom types ×
Bond types ×
$E_{Coord}(\boldsymbol{x}) = \frac{1}{2\tau_c^2}(\boldsymbol{x}-\boldsymbol{x}_0)^\top(\boldsymbol{x}-\boldsymbol{x}_0)$
σ_f
τ_f
Atom types ×
Rotatable bonds √
$E_{Frad}(\boldsymbol{x}) \approx \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_0)^\top \boldsymbol{\Sigma}^{-1}_{\tau_f,\sigma_f}(\boldsymbol{x}-\boldsymbol{x}_0)$
k^{A}_{C-C-H}
k^{B}_{C-C}
k^{A}_{O-C-H}
$k^{B}_{C=O}$
$k^{A}_{C-C=O}$
$k^{T}_{H-C-C=O}$
k^{B}_{C-H}
Atom types √
Bond types √
$E_{SliDe}(\boldsymbol{r},\boldsymbol{\theta},\boldsymbol{\phi}) = \frac{1}{2}[k^B \odot (\boldsymbol{r}-\boldsymbol{r}_0)]^\top(\boldsymbol{r}-\boldsymbol{r}_0)$
$+\frac{1}{2}[k^A \odot (\boldsymbol{\theta}-\boldsymbol{\theta}_0)]^\top(\boldsymbol{\theta}-\boldsymbol{\theta}_0)$
$+\frac{1}{2}[k^T \odot (\boldsymbol{\phi}-\boldsymbol{\phi}_0)]^\top(\boldsymbol{\phi}-\boldsymbol{\phi}_0)$

Outline

中国科学院自动化研究所
模式识别实验室
New Laboratory of Pattern Recognition

中国科学院自动化研究所
Institute of Automation
Chinese Academy of Sciences

[AAAI 2024]

Rethinking Graph Masked Autoencoders through Alignment and Uniformity

Liang Wang[1,2], Xiang Tao[1,2], Qiang Liu[1,2], Shu Wu[1,2], Liang Wang[1,2]

[1]Institute of Automation, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences

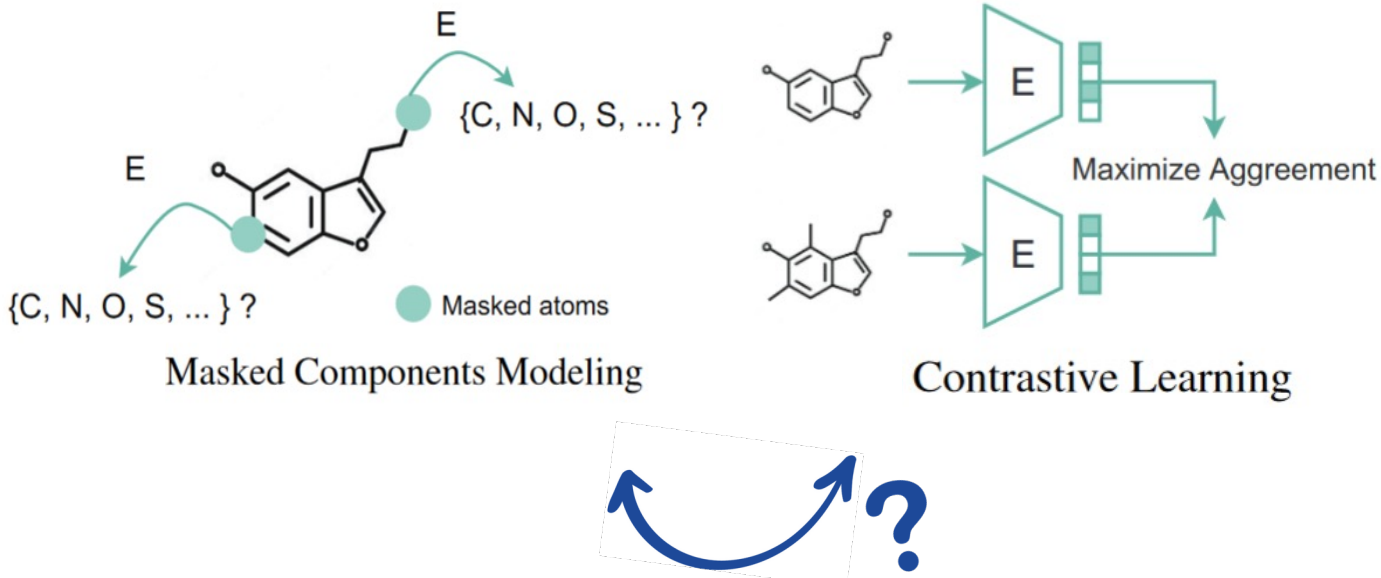
AAAI
Association for the Advancement of Artificial Intelligence

25 October 2024

Rethinking Graph Masked Autoencoders through Alignment and Uniformity

Liang Wang[1,2*], **Xiang Tao**[1,2*], **Qiang Liu**[1,2], **Shu Wu**[1,2†], **Liang Wang**[1,2]

[1]Center for Research on Intelligent Perception and Computing
State Key Laboratory of Multimodal Artificial Intelligence Systems
Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
{liang.wang, xiang.tao}@cripac.ia.ac.cn, {qiang.liu, shu.wu, wangliang}@nlpr.ia.ac.cn



Are GraphMAE and GCL completely different methods, or ***do they share any commonality****?*

Theoretical Understanding of GraphMAE

Assumption 4.1. For any graph decoder g, we assume the existence of a pseudo-inverse graph encoder f_g such that the resulting pseudo graph autoencoder $h_g = g \circ f_g$ satisfies $\mathbb{E}_x \left\| h_g(\boldsymbol{x}) - \boldsymbol{x} \right\|^2 \leq \varepsilon$, where $\boldsymbol{x}$ represents the feature of masked node $v \in \widetilde{\mathcal{V}}$.

Theorem 4.2. Under Assumption 4.1, the SCE loss in Eq. (2) can be lower bounded by a pretext loss:

$$\mathcal{L}_{\text{SCE}}(h) \geq \frac{\gamma}{2}\mathcal{L}_{\text{Pretext}}(h) - \frac{\gamma}{2}\varepsilon + \text{const}, \quad \text{where } \mathcal{L}_{\text{Pretext}}(h) = -\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} h_g(\boldsymbol{x}_i)^\top h(c_i). \tag{6}$$

Definition 4.3. ***(Context-Level Alignment Loss)*** The alignment loss for positive context pairs (c, c^+) is defined as:

$$\mathcal{L}^{\text{c}}_{\text{Align}}(h) = -\mathop{\mathbb{E}}_{(c,c^+)\sim p^{\text{c}}_{\text{pos}}} h(c)^\top h(c^+). \tag{7}$$

Theorem 4.4. The pretext loss in Eq. (6) can be lower bounded by the context-level alignment loss in Eq. (7):

$$\mathcal{L}_{\text{Pretext}}(h) \geq \frac{1}{2}\mathcal{L}^{\text{c}}_{\text{Align}}(h) + \text{const}. \tag{8}$$

Theorem 4.5. Under Assumption 4.1, GraphMAE's nodel-level reconstruction loss in Eq. (2) can be lower bounded by the context-level alignment loss in Eq. (7):

$$\begin{aligned}\mathcal{L}_{\text{SCE}}(h) &\geq \frac{\gamma}{4}\mathcal{L}^{\text{c}}_{\text{Align}}(h) - \frac{\gamma}{2}\varepsilon + \text{const} \\ &= -\frac{\gamma}{4}\mathop{\mathbb{E}}_{c,c^+} h(c)^\top h(c^+) - \frac{\gamma}{2}\varepsilon + \text{const}.\end{aligned} \tag{10}$$

Proof Sketch

$$\begin{aligned}
\mathcal{L}_{\text{SCE}} &= \mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} (1 - \boldsymbol{x}_i^\top h(c_i))^\gamma \\
&\geq \mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} (1 - \gamma \boldsymbol{x}_i^\top h(c_i)) && \text{(Bernoulli's inequality)} \\
&= \mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} (1 - \gamma(1 - \frac{1}{2}\|\boldsymbol{x}_i - h(c_i)\|^2)) && \text{(features are normalized)} \\
&= 1 - \gamma + \frac{\gamma}{2}\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} \|\boldsymbol{x}_i - h(c_i)\|^2 \\
&= 1 - \gamma + \frac{\gamma}{2}\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} (\|\boldsymbol{x}_i - h(c_i)\|^2 + \varepsilon) - \frac{\gamma}{2}\varepsilon && \text{(Assumption 4.1)} \\
&\geq 1 - \gamma + \frac{\gamma}{2}\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} (\|\boldsymbol{x}_i - h(c_i)\|^2 + \|h_g(\boldsymbol{x}_i) - \boldsymbol{x}_i\|^2) - \frac{\gamma}{2}\varepsilon. \\
&\geq 1 - \gamma + \frac{\gamma}{4}\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} \|h_g(\boldsymbol{x}_i) - h(c_i)\|^2 - \frac{\gamma}{2}\varepsilon \\
&= 1 - \gamma + \frac{\gamma}{4}\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} (2 - 2h_g(\boldsymbol{x}_i)^\top h(c_i))) - \frac{\gamma}{2}\varepsilon \\
&= -\frac{\gamma}{2}\mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} h_g(\boldsymbol{x}_i)^\top h(c_i) - \frac{\gamma}{2}\varepsilon + 1 - \frac{\gamma}{2} \\
&= \frac{\gamma}{2}\mathcal{L}_{\text{Pretext}}(h) - \frac{\gamma}{2}\varepsilon + \text{const}.
\end{aligned}$$

Proof Sketch

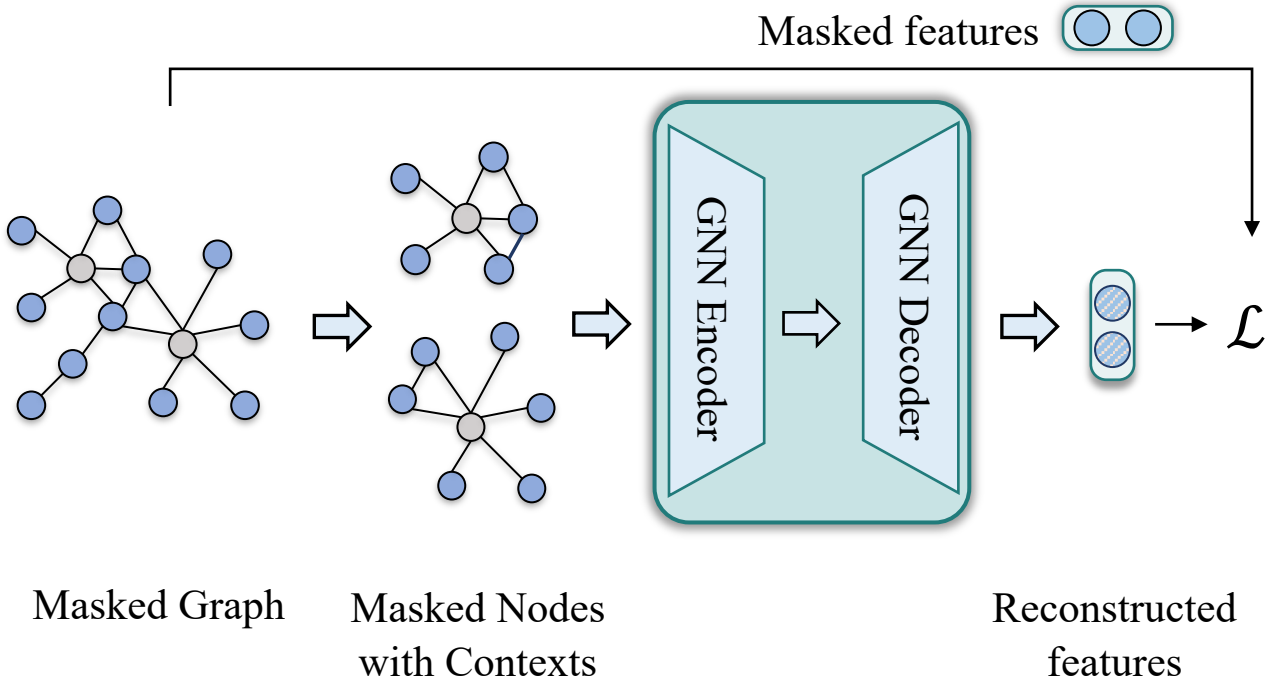
$$\begin{aligned}
\mathcal{L}_{\text{Pretext}}(h) &= -\operatorname{tr}(\mathbf{H}_g^\top \tilde{\mathbf{A}}_{\text{CF}} \mathbf{H}) \\
&\geq -\frac{1}{2}\left(\|\mathbf{H}_g\|_{\text{F}}^2 + \|\tilde{\mathbf{A}}_{\text{CF}}\mathbf{H}\|_{\text{F}}^2\right) && (\operatorname{tr}(\mathbf{AB}) \leq \tfrac{1}{2}\left(\|\mathbf{A}\|_{\text{F}}^2 + \|\mathbf{B}\|_{\text{F}}^2\right)) \\
&= -\frac{1}{2}\operatorname{tr}\left(\tilde{\mathbf{A}}_{\text{CF}}^\top \tilde{\mathbf{A}}_{\text{CF}} \mathbf{H}\mathbf{H}^\top\right) - \frac{1}{2} && (\|\mathbf{H}_g\|_{\text{F}}^2 = \textstyle\sum_{f_j} d_{f_j} \|h_g(f_j)\|^2 = 1) \\
&= -\frac{1}{2}\sum_{c,c^+}\sum_{f_j} \frac{w_{c,f_j} w_{c^+,f_j}}{d_{f_j}} h(c)^\top h(c^+) - \frac{1}{2} \\
&= -\frac{1}{2}\sum_{c,c^+} (\mathbf{A}_{\text{C}})_{c,c^+} h(c)^\top h(c^+) - \frac{1}{2} \\
&= \frac{1}{2}\mathcal{L}^{\text{c}}_{\text{align}}(h) - \frac{1}{2},
\end{aligned}$$

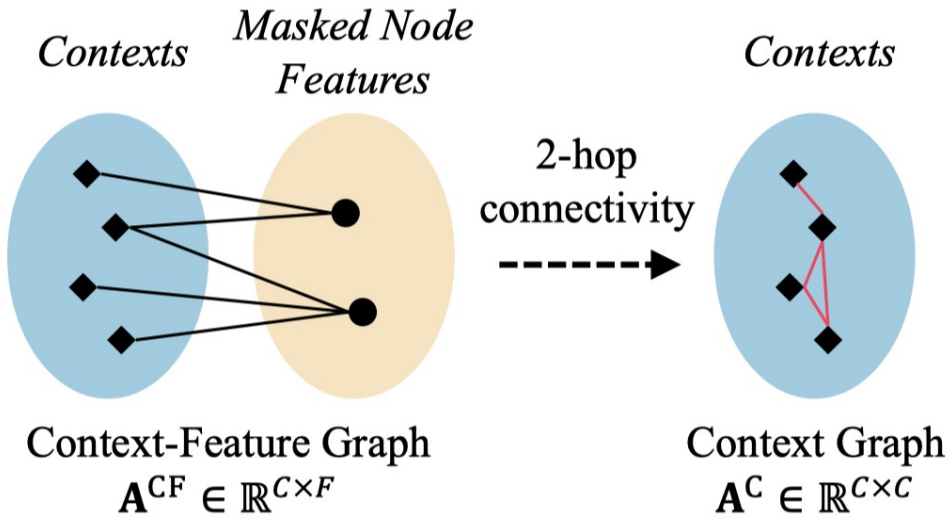
Theoretical result:
GraphMAE performs implicit context-level graph contrastive learning.

Theoretical Understanding of GraphMAE

Theoretical result: GraphMAE performs implicit context-level graph contrastive learning.

Intuitive Explanation:

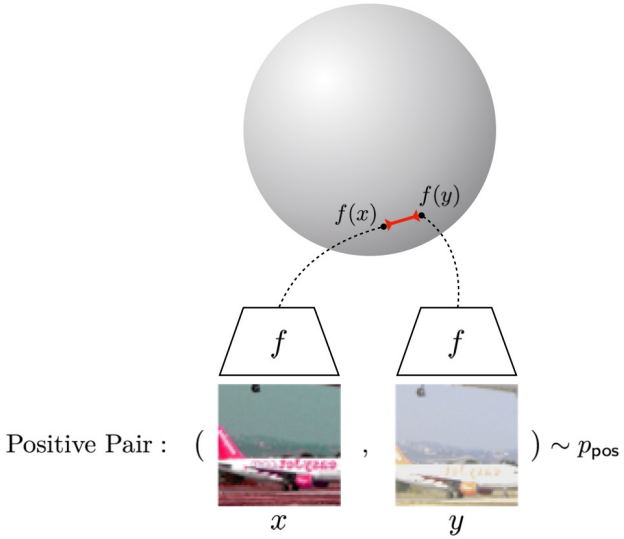




Theoretical Understanding of GraphMAE

Measure representation quality of GraphMAE:

- **Representation Alignment?**
- **Representation Uniformity?**



Alignment: Similar samples have similar features.
(Figure inspired by Tian et al. (2019).)

Uniformity: Preserve maximal information.

Note:

- **Alignment (一致性)** refers to the concentration of samples from the same class within the same region of the hypersphere.
- **Uniformity (均匀性)** refers to the uniform distribution of all samples on the hypersphere.

[1] Tongzhou Wang, Phillip Isola. "Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere." In *ICML*. 2020

Theoretical Understanding of GraphMAE

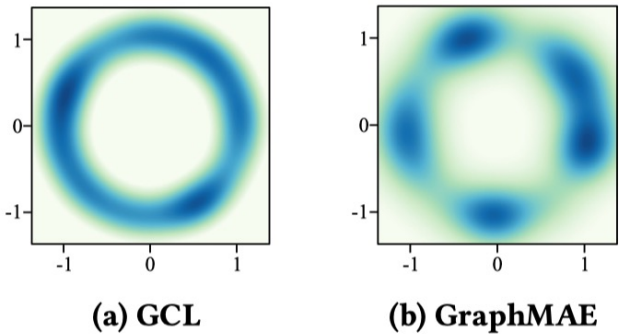
Limitations of GraphMAE:

- **Alignment** performance is still restricted by the mask distribution, which is **decided by the masking strategy**.

$$\mathcal{L}_{\mathrm{SCE}} = \mathop{\mathbb{E}}_{v_i \in \widetilde{\mathcal{V}}} \left(1 - x_i{}^T \cdot g(f(c_i))\right)^{\gamma}, \gamma \geq 1,$$

$$\mathcal{L}^{\mathrm{c}}_{\mathrm{Align}}(h) = -\mathop{\mathbb{E}}_{c,c^+} h(c)^{\top} h\left(c^+\right).$$

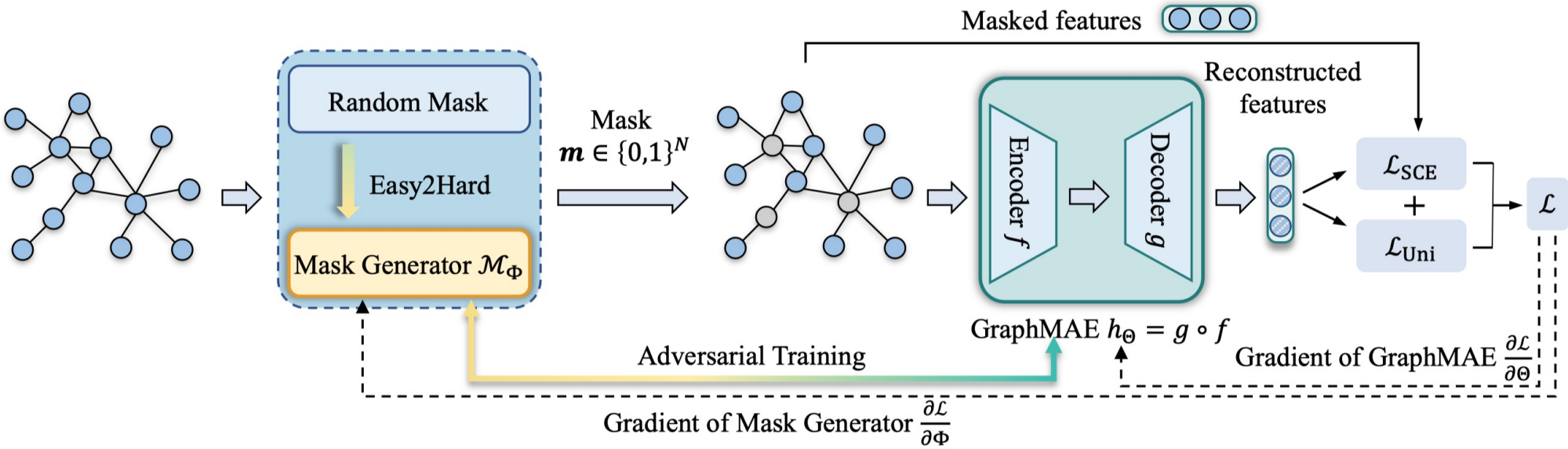
- **Uniformity** performance is **not strictly guaranteed**.



(a) GCL (b) GraphMAE

Figure 1: Distribution of nodes representations on the unit hypersphere learned by GCL (taking GRACE [52] as an example) and GraphMAE [7]. The representations learned by GCL is more uniformly distributed than GraphMAE.

Alignment-Uniformity Enhanced Graph Masked Autoencoders



Alignment Enhancement

- Adversarial Masking

$$\Phi^\star = \arg\max_{\Phi}(\mathcal{L}_{\text{SCE}}(\mathcal{G};\Theta,\Phi) - \lambda_1 \sin(\frac{\pi}{N}\sum_{i=1}^{N} m_i)^{-1}),$$

$$\Theta^\star = \arg\min_{\Theta}(\mathcal{L}_{\text{SCE}}(\mathcal{G};\Theta,\Phi) + (1-\alpha_{\text{adv}})\lambda_2 \mathcal{L}_{\text{Uni}}(\mathcal{G};\Theta)),$$

- Easy-to-Hard Masking

$$prob(t) = (1-\alpha_{\text{adv}}(t)) \cdot prob_{\text{rand}} + \alpha_{\text{adv}}(t) \cdot prob_{\text{adv}}(t),$$

$$\alpha_{\text{adv}}(t) = \alpha_0 + \Delta\alpha(t) = \alpha_0 + (\frac{t}{T})^\eta \cdot (\alpha_T - \alpha_0),$$

Uniformity Enhancement

- Explicit Uniformity Regularizer

$$\mathcal{L}_{\text{Uni}} = \log \mathop{\mathbb{E}}_{(z_i,z_j)\sim p_{\text{data}}} e^{-t\|z_i - z_j\|^2},$$

Experimental Results

Performance on node classification and graph classification.

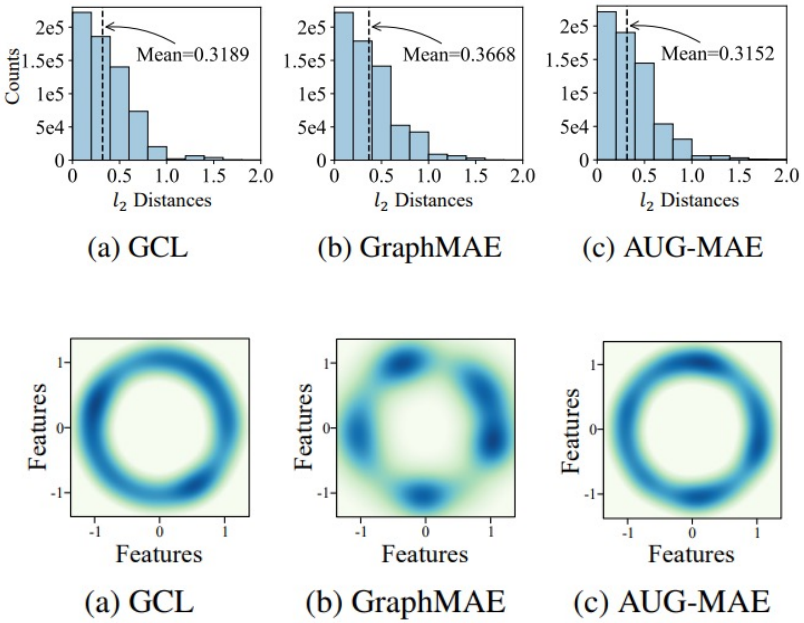
	Method	Cora	CiteSeer	PubMed	Ogbn-arxiv	PPI	Reddit	Corafull	Flickr	WikiCS	A.R.
Contrastive	DGI	82.3 ± 0.6	71.8 ± 0.7	76.8 ± 0.6	70.3 ± 0.2	63.8 ± 0.2	94.0 ± 0.1	48.2 ± 0.5	45.0 ± 0.2	64.8 ± 0.6	7.89
	MVGRL	83.5 ± 0.4	73.3 ± 0.5	80.1 ± 0.7	-	-	-	52.6 ± 0.5	-	64.8 ± 0.7	5.20
	GRACE	81.9 ± 0.4	71.2 ± 0.5	80.6 ± 0.4	71.5 ± 0.1	69.7 ± 0.2	94.7 ± 0.1	45.2 ± 0.1	-	68.0 ± 0.7	6.50
	BGRL	82.7 ± 0.6	71.1 ± 0.8	79.6 ± 0.5	71.6 ± 0.1	73.6 ± 0.2	94.2 ± 0.1	47.4 ± 0.5	39.4 ± 0.1	65.5 ± 1.5	6.56
	InfoGCL	83.5 ± 0.3	**73.5 ± 0.4**	79.1 ± 0.2	-	-	-	-	-	-	4.67
	CCA-SSG	84.0 ± 0.4	73.1 ± 0.3	81.0 ± 0.4	71.2 ± 0.2	73.3 ±0.2	95.1 ± 0.1	53.5 ± 0.4	49.1 ± 0.1	67.4 ±0.9	3.89
Generative	SeeGera	82.8 ± 0.3	71.6 ± 0.2	79.2 ± 0.3	71.2 ± 0.3	73.4 ± 0.3	95.2 ± 0.2	52.0 ± 0.4	49.4 ± 0.5	65.8 ± 0.2	5.78
	MaskGAE	82.6 ± 0.3	73.1 ± 0.6	81.0 ± 0.3	71.2 ± 0.3	73.9 ± 0.3	95.4 ± 0.1	52.2 ± 0.1	49.1 ± 0.4	66.0 ± 0.2	4.78
	GraphMAE	84.0 ± 0.6	73.1 ± 0.4	80.9 ± 0.4	71.3 ± 0.6	74.1 ± 0.4	95.8 ± 0.4	53.3 ± 0.4	49.5 ± 0.5	70.6 ± 0.9	3.00
	AUG-MAE	**84.3 ± 0.4**	73.2 ± 0.4	**81.4 ± 0.4**	**71.9 ± 0.2**	**74.3 ± 0.1**	**96.1 ± 0.1**	**57.6 ± 0.3**	**50.3 ± 0.2**	**71.7 ± 0.6**	**1.22**

Table 1: Node classification results on benchmarks. We report Micro-F1(%) score for PPI and accuracy(%) for the other datasets. The best results are highlighted in **bold** and the runner ups are highlighted with underlines. A.R. means the average rank.

	Method	IMDB-B	IMDB-M	PROTEINS	COLLAB	MUTAG	REDDIT-B	A.R.
Contrastive	Graph2vec	71.10 ± 0.54	50.44 ± 0.87	73.30 ± 2.05	-	83.15 ± 9.25	75.78 ± 1.03	7.00
	InfoGraph	73.03 ± 0.87	49.69 ± 0.53	74.44 ± 0.31	70.65 ± 1.13	89.01 ± 1.13	82.50 ± 1.42	5.17
	GraphCL	71.14 ± 0.44	48.58 ± 0.67	74.39 ± 0.45	71.36 ± 1.15	86.80 ± 1.34	89.53 ± 0.84	5.83
	JOAO	70.21 ± 3.08	49.20 ± 0.77	74.55 ± 0.41	69.50 ± 0.36	87.35 ± 1.02	85.29 ± 1.35	6.33
	GCC	72.0	49.4	-	78.9	-	**89.8**	4.50
	MVGRL	74.20 ± 0.70	51.20 ± 0.50	-	-	89.70 ± 1.10	84.50 ± 0.60	4.00
	InfoGCL	75.10 ± 0.90	51.40 ± 0.80	-	80.00 ± 1.30	88.28 ± 0.98	-	2.25
Generative	GraphMAE	75.30 ± 0.59	51.35 ± 0.78	75.30 ± 0.52	80.32 ± 0.42	88.19 ± 1.26	87.83 ± 0.25	3.00
	AUG-MAE	**75.56 ± 0.61**	**51.80 ± 0.86**	**75.83 ± 0.24**	**80.48 ± 0.50**	**91.20 ± 1.30**	87.98 ± 0.43	**1.83**

Table 2: Graph classification results on benchmarks. We report accuracy(%) for all datasets. The best results are highlighted in **bold** and the runner ups are highlighted with underlines. A.R. means the average rank.

Performance on representation alignment and uniformity.



中国科学院自动化研究所
模式识别实验室
New Laboratory of Pattern Recognition

中国科学院自动化研究所
Institute of Automation
Chinese Academy of Sciences

[NeurIPS 2024]

Pin-Tuning: Parameter-Efficient In-Context Tuning for Few-Shot Molecular Property Prediction

Liang Wang[1,2], Qiang Liu[1,2], Shaozhen Liu[3] , Xin Sun[4], Shu Wu[1,2], Liang Wang[1,2,4]

[1]Institute of Automation, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
[3]Beijing Institute of Technology
[4]University of Science and Technology of China

NEURAL INFORMATION PROCESSING SYSTEMS

25 October 2024

Few-Shot Molecular Property Prediction

Key Elements underlying Molecular Property Prediction

Representation

Generalization

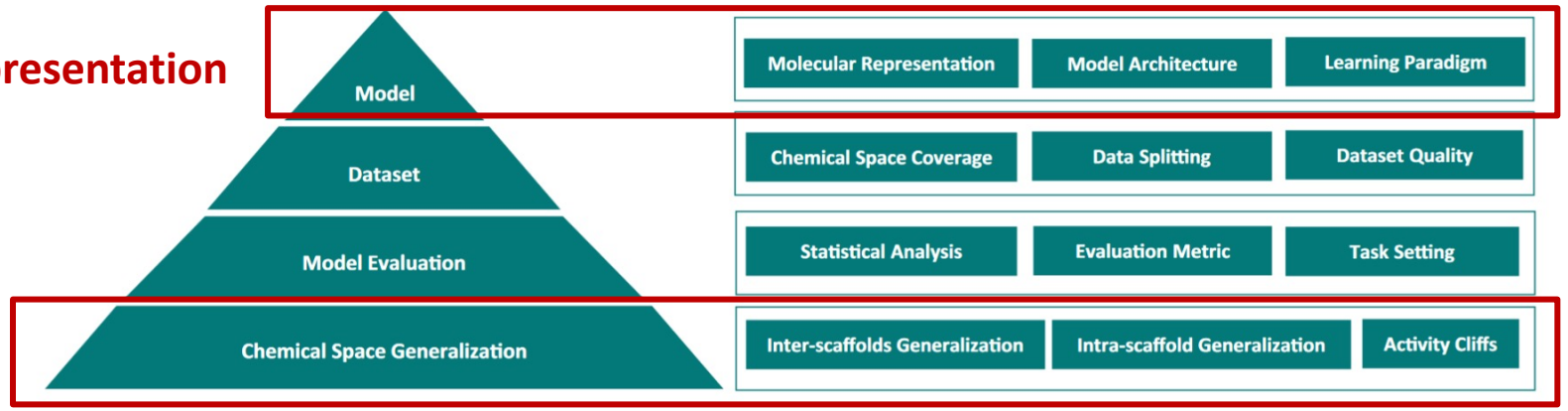
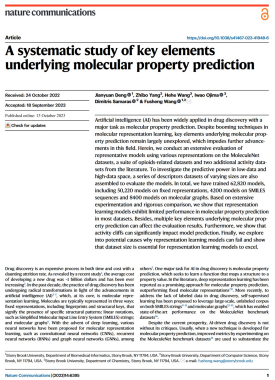


Fig. 1 | Key elements underlying molecular property prediction. There are four aspects involved: model, dataset, model evaluatcgqrrently in the literature, the focus is more on the model, which aims at developing novel learning paradigms or model architectures on certain molecular representations. However, it is also necessary to consider other crucial elements, pertaining to (1) what the model is built upon, (2) how the model is evaluated, and (3) eventually what the model is capable of. For the dataset, its chemical space coverage (w.r.t. both structures and labels), and scrutiny of its quality, including dataset size and label accuracy (e.g., duplicates, contradictories, and noise), as well as data splitting, is essential before developing a model for a specific property prediction task. For the model evaluation, thoughtful consideration of statistical analysis, evaluation metrics, and task settings is critical as they impact the observed prediction performance. For the chemical space generalization, it is important to clarify the model's applicability and if the activity-cliffs issue is addressed.


nature communications

Article

A systematic study of key elements underlying molecular property prediction

[1] **"A Systematic Study of Key Elements Underlying Molecular Property Prediction."** *Nature Communications*, 2023

Few-Shot Molecular Property Prediction

Representative Work

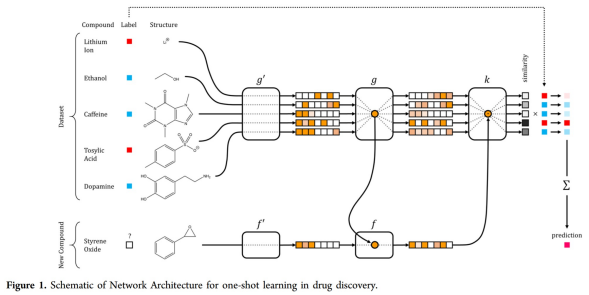


Figure 1. Schematic of Network Architecture for one-shot learning in drug discovery.

IterRefLSTM, ACS Central Science, 2017

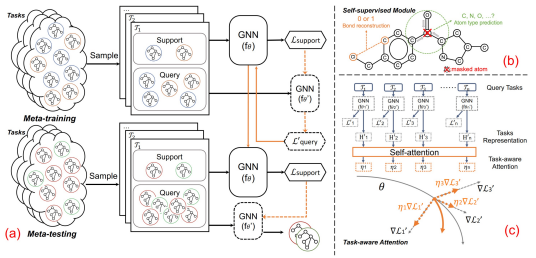
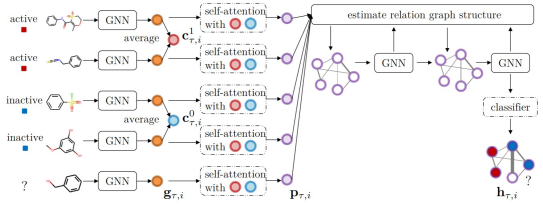
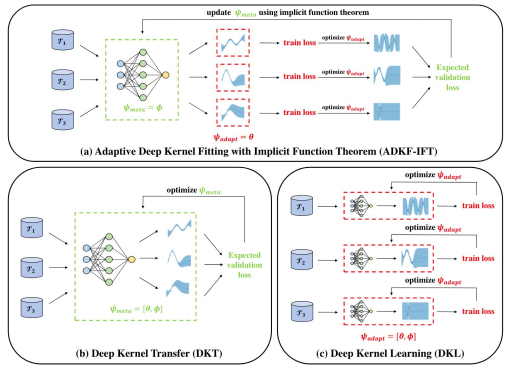


Figure 1: (a) The overall framework of Meta-MGNN: It first samples a batch of training tasks. For each task, there are a few

Meta-MGNN, WWW, 2021

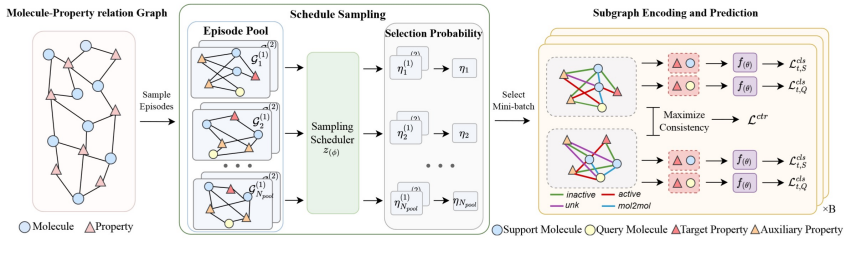


PAR, NeurIPS, 2021

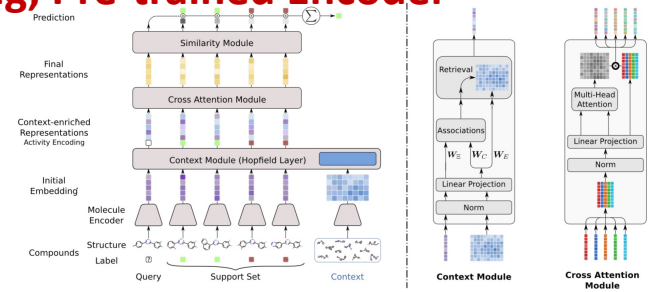


ADKF-IFT, ICLR, 2023

Matching Network, Meta-Learning, Pre-trained Encoder

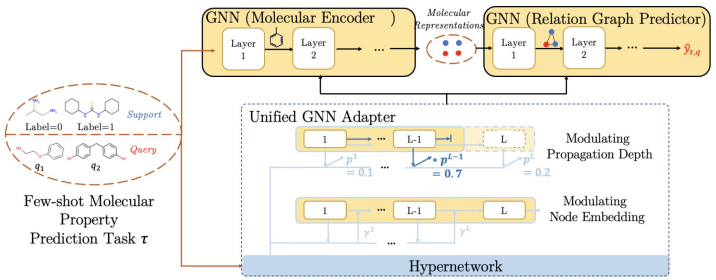


GS-Meta, IJCAI, 2023

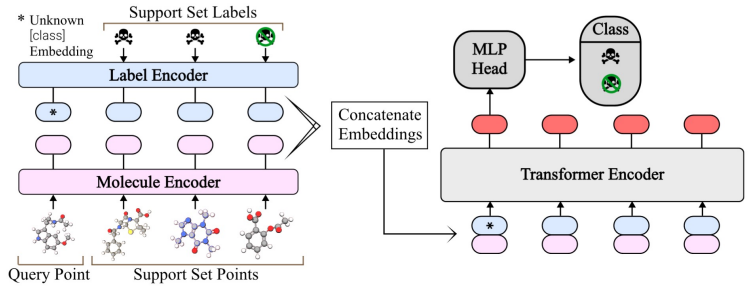


MHNfs, ICLR, 2023

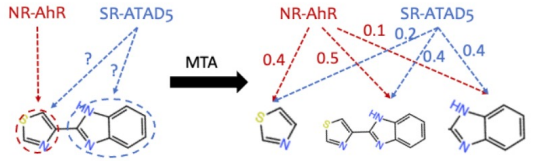
Molecular Context



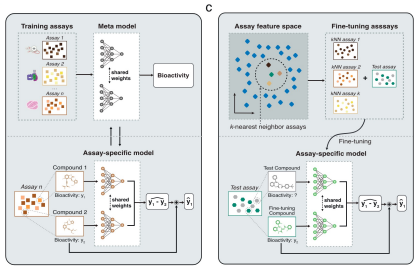
PACIA, IJCAI 2024



CAMP, submitted to ICLR 2024



MTA, SDM 2023



ActFound, bioRxiv 2024

Adaptation (hypernetwork, in-context learning)

Cross-property (task augmentation, pairwise learning)

Few-Shot Molecular Property Prediction

Evaluation Benchmark

MoleculeNet

A Benchmark for Molecular Machine Learning

Dataset Details

Category	Dataset	Data Type	Task Type	# Tasks	# Compounds	Rec - Split[a]	Rec - Metric[b]
Quantum Mechanics	QM7	SMILES, 3D coordinates	Regression	1	7160	Stratified	MAE
	QM7b	3D coordinates	Regression	14	7210	Random	MAE
	QM8	SMILES, 3D coordinates	Regression	12	21786	Random	MAE
	QM9	SMILES, 3D coordinates	Regression	12	133885	Random	MAE
Physical Chemistry	ESOL	SMILES	Regression	1	1128	Random	RMSE
	FreeSolv	SMILES	Regression	1	642	Random	RMSE
	Lipophilicity	SMILES	Regression	1	4200	Random	RMSE
Biophysics	PCBA	SMILES	Classification	128	437929	Random	PRC-AUC
	MUV	SMILES	Classification	17	93087	Random	PRC-AUC
	HIV	SMILES	Classification	1	41127	Scaffold	ROC-AUC
	PDBbind	SMILES, 3D coordinates	Regression	1	11908	Time	RMSE
	BACE	SMILES	Classification	1	1513	Scaffold	ROC-AUC
Physiology	BBBP	SMILES	Classification	1	2039	Scaffold	ROC-AUC
	Tox21	SMILES	Classification	12	7831	Random	ROC-AUC
	ToxCast	SMILES	Classification	617	8575	Random	ROC-AUC
	SIDER	SMILES	Classification	27	1427	Random	ROC-AUC
	ClinTox	SMILES	Classification	2	1478	Random	ROC-AUC

Dataset	Tox21	SIDER	MUV	ToxCast	PCBA
#Compound	7831	1427	93127	8575	437929
#Property	12	27	17	617	128
#Train Property	9	21	12	451	118
#Test Property	3	6	5	158	10
%Positive Label	6.24	56.76	0.31	12.60	0.84
%Negative Label	76.71	43.24	15.76	72.43	59.84
%Unknown Label	17.05	0	84.21	14.97	39.32

Molecular Property Prediction, N-way K-shot

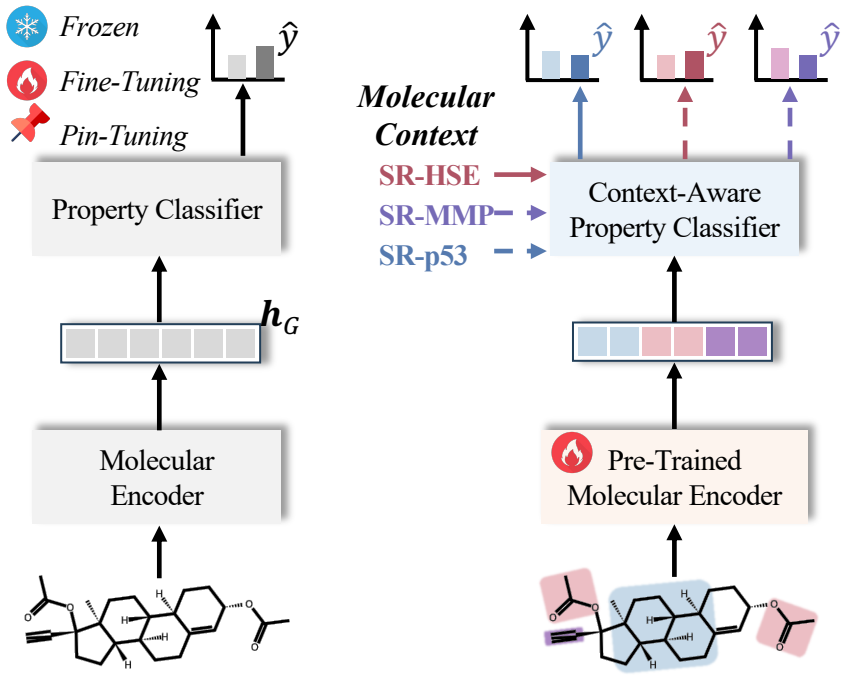
FS-Mol: A Few-Shot Learning Dataset of Molecules

	Datasets			
	ExCAPE-ML	PCBA	LSC	FS-Mol
# measurements	49,316,517	34,017,170	5,100,411	489,133
# compounds	955,386	437,929	449,391	233,786
# tasks	526	128	1310	5120
Mean # compounds / task	93,758	265,759	3872	94
Median # compounds / task	1820	309,562	320	46
Mean inactive:active / task	268:1	46:1	7:1	1:1
Raw values available?	Yes	No	No	Yes
Source	PubChem/ChEMBL	PubChem	ChEMBL18	ChEMBL27

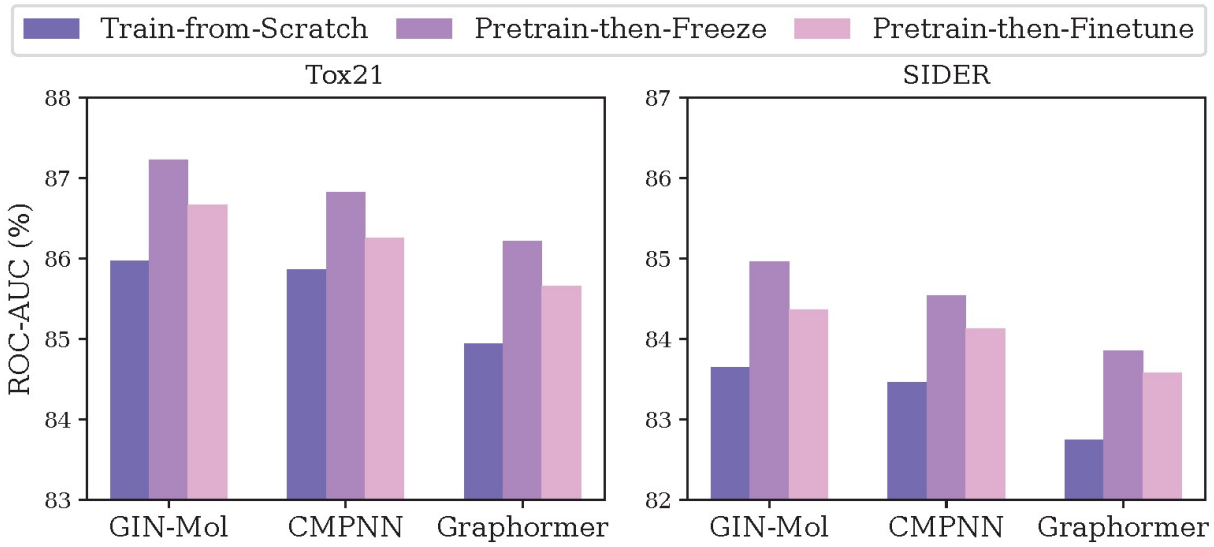
Bioactivity Prediction, N-shot (support set size), Stratified Random Split

Recent Work Ⅱ —— Pin-Tuning

Few-Shot Molecular Property Prediction



(a) Vanilla MPP framework. (b) Existing FSMPP framework.



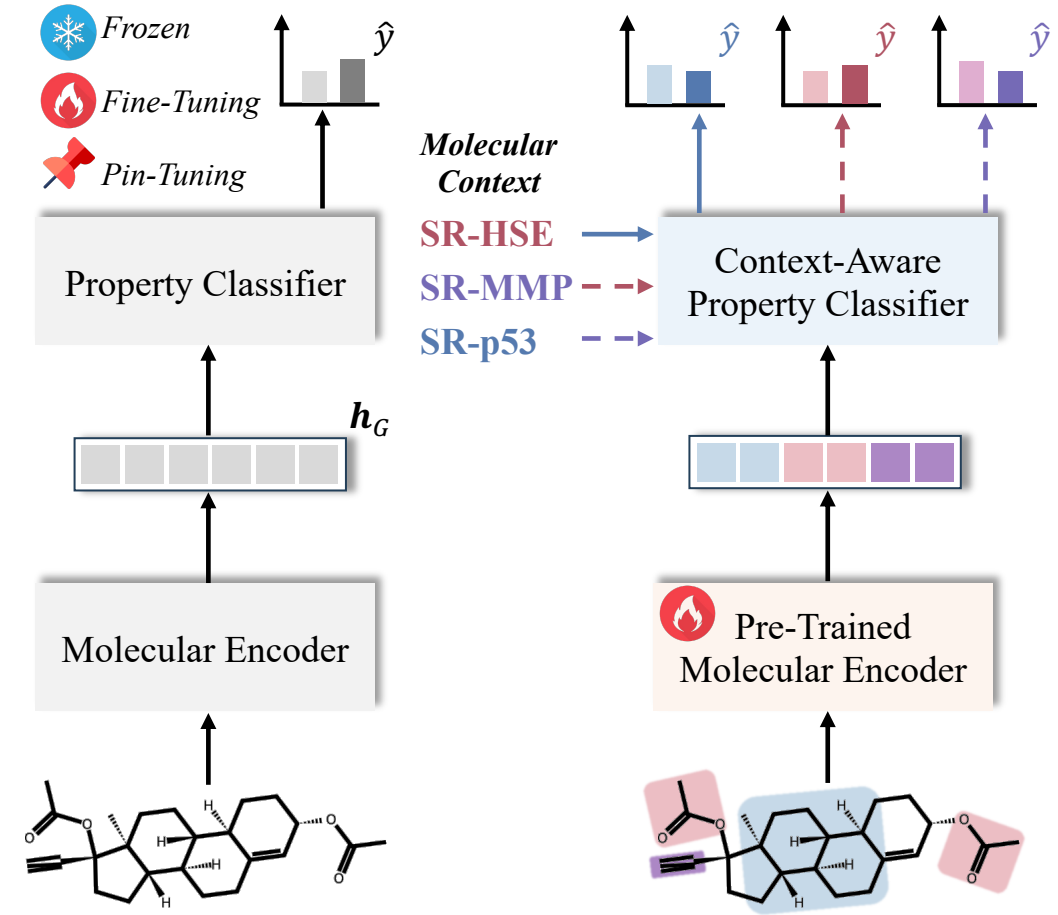
Observation

Train-from-Scratch < Pretrain-then-Finetune ≤ Pretrain-then-Freeze

Pre-training is effective, but fine-tuning is ineffective.

How to adapt molecular pre-trained models *to downstream tasks, especially in* ***few-shot*** *scenarios?*

Few-Shot Molecular Property Prediction

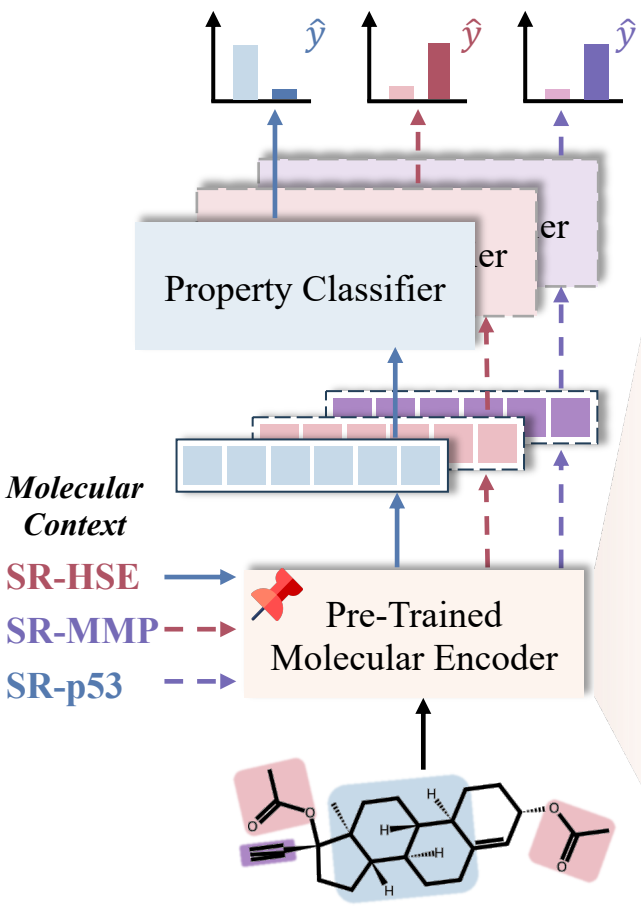


(a) Vanilla MPP framework. (b) Existing FSMPP framework.

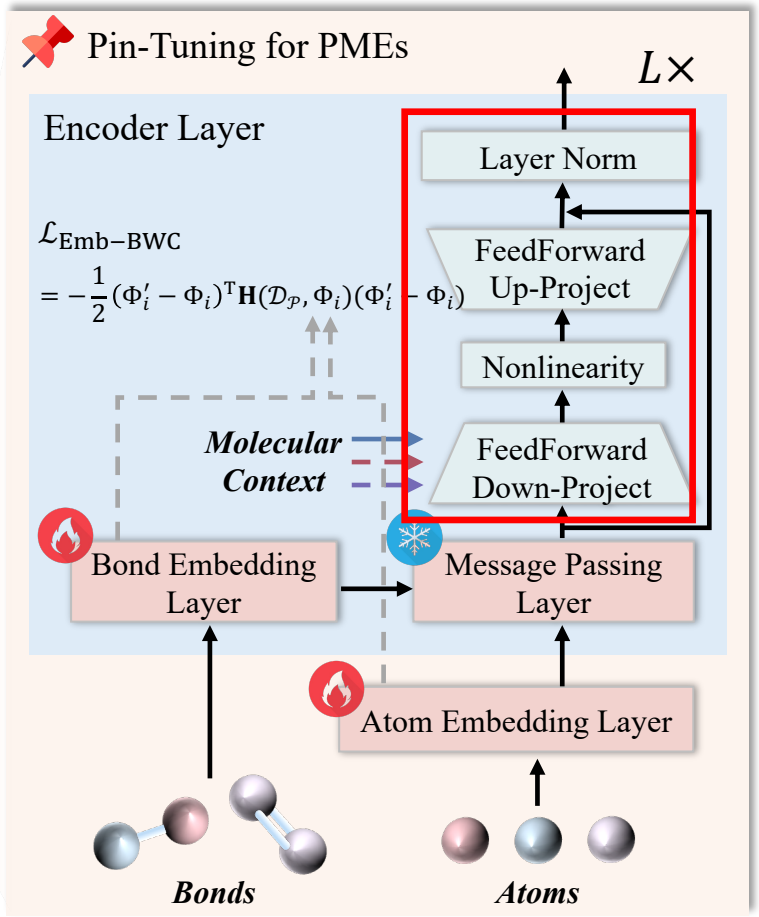
Reasons:

1. **Imbalance between the abundance of tunable parameters and the scarcity of labeled molecules.**
2. **Limited contextual perceptiveness in the encoder.**

Pin-Tuning: Parameter-Efficient In-Context Tuning for Few-Shot Molecular Property Prediction



Our FSMPP framework.



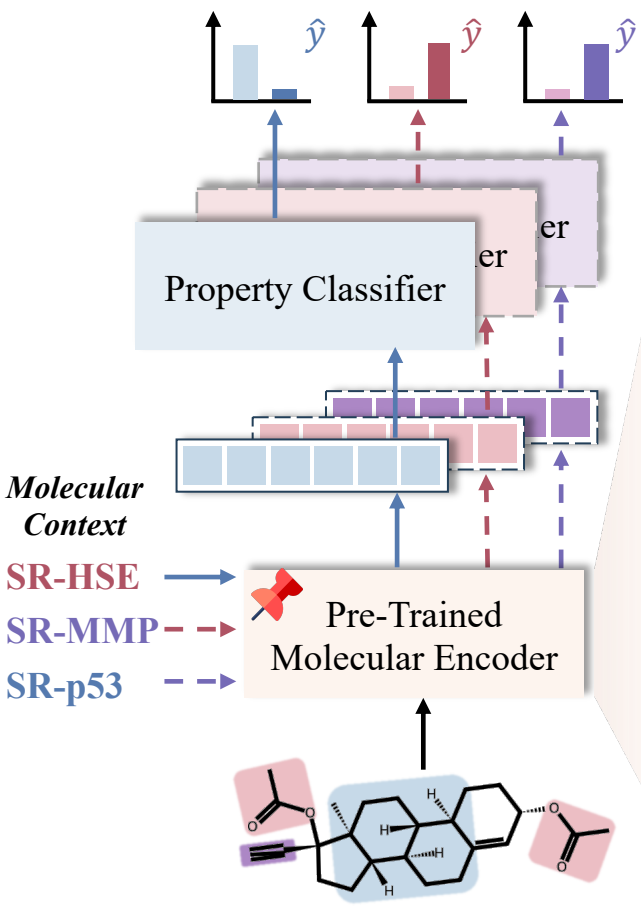
Our Pin-Tuning method for PMEs.

MP-Adapter: message passing layer-oriented adapter

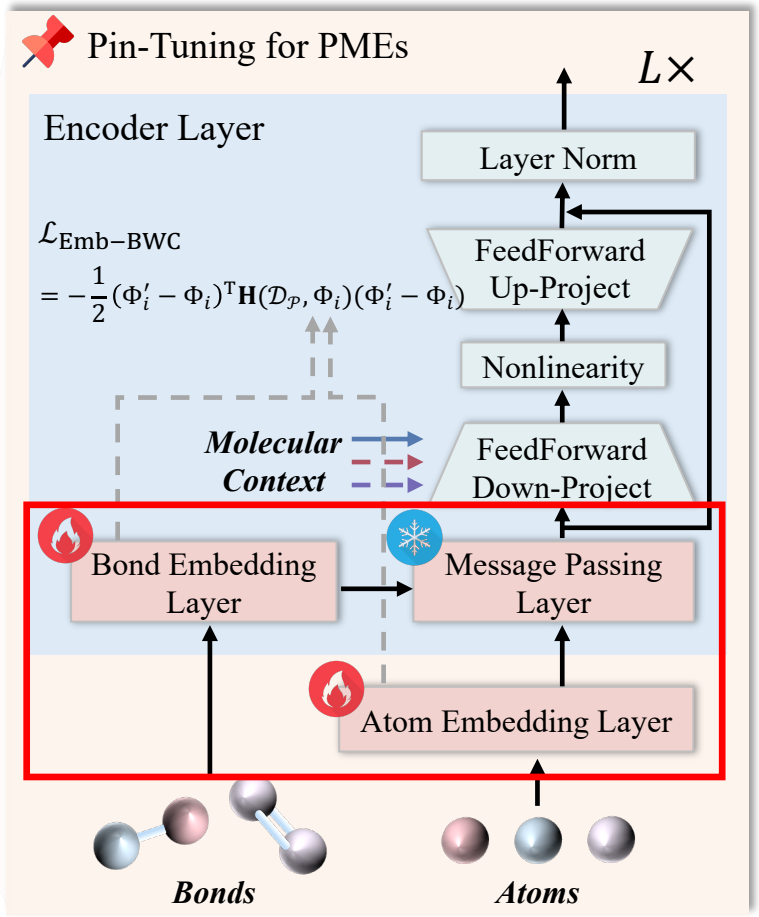
$$\boldsymbol{z}_v^{(l)} = \texttt{FeedForward}_{\texttt{down}}(\boldsymbol{h}_v^{(l)}) \in \mathbb{R}^{d_2},$$
$$\Delta\boldsymbol{h}_v^{(l)} = \texttt{FeedForward}_{\texttt{up}}(\phi(\boldsymbol{z}_v^{(l)})) \in \mathbb{R}^{d},$$
$$\tilde{\boldsymbol{h}}_v^{(l)} = \texttt{LayerNorm}(\boldsymbol{h}_v^{(l)} + \Delta\boldsymbol{h}_v^{(l)}) \in \mathbb{R}^{d},$$

- *Bottleneck*
- *Near-zero initialization*
- *Skip-connection*

Pin-Tuning: Parameter-Efficient In-Context Tuning for Few-shot Molecular Property Prediction



Our FSMPP framework.



Our Pin-Tuning method for PMEs.

Emb-BWC: embedding layer-oriented Bayesian weight consolidation

$$\mathcal{L}_{\text{Emb-BWC}} = -\frac{1}{2}\sum_{i=1}^{E}(\Phi_i' - \Phi_i)^\top \mathbf{H}(\mathcal{D}_\mathcal{P}, \Phi_i)(\Phi_i' - \Phi_i),$$

- *Maximum a posterior (MAP) estimation*
- *Bayesian learning theory*
- *Second-order Taylor expansion*

Three choices of diagonal approximation of Hessian

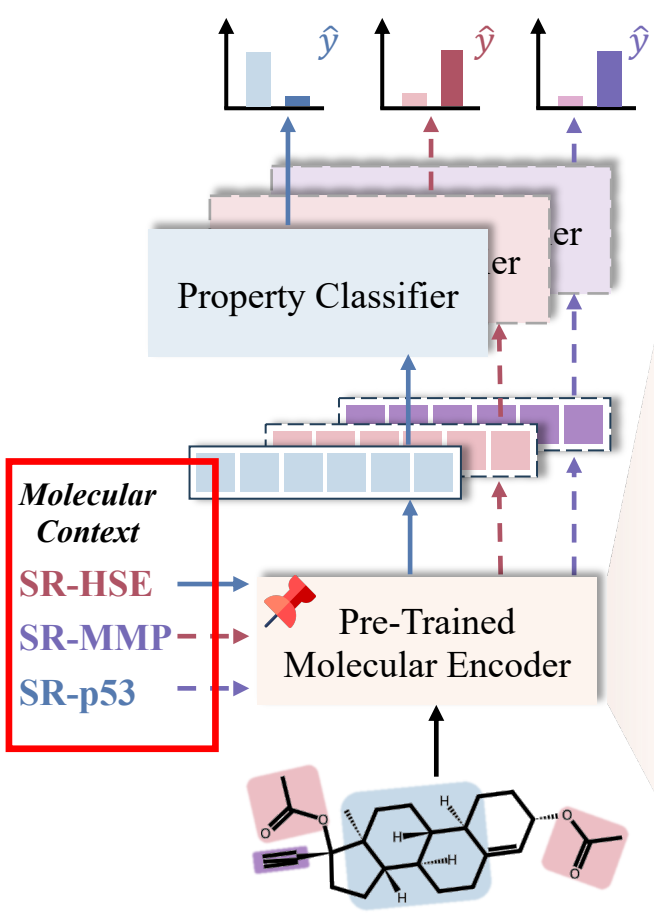
$$\mathcal{L}^{\text{IM}}_{\text{Emb-BWC}} = \tfrac{1}{2}\textstyle\sum_{i=1}^{E}\sum_{j=1}^{d}(\Phi'_{i,j} - \Phi_{i,j})^2$$

$$\mathcal{L}^{\text{FIM}}_{\text{Emb-BWC}} = \tfrac{1}{2}\textstyle\sum_{i=1}^{E}\hat{\mathbf{F}}_i(\Phi_i' - \Phi_i)^2$$

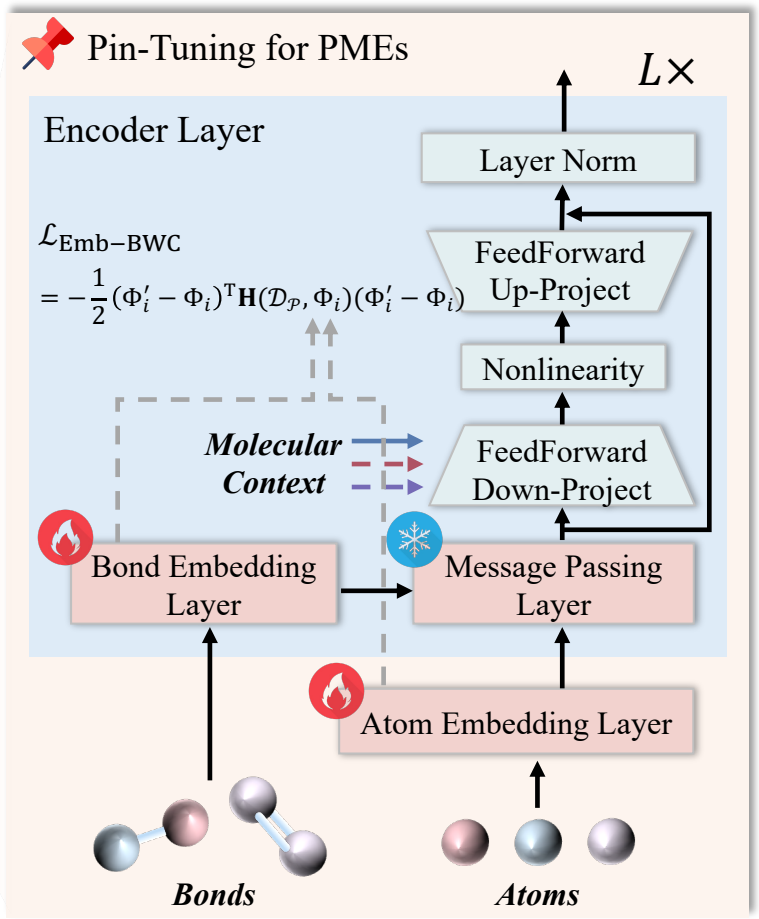
$$\mathcal{L}^{\text{EFIM}}_{\text{Emb-EWC}} = \tfrac{1}{2}\textstyle\sum_{i=1}^{E}\tilde{\mathbf{F}}_i(\tilde{\Phi}_i' - \tilde{\Phi}_i)^2$$

- *Identity matrix.*
- *Diagonal of Fisher information matrix.*
- *Diagonal of embedding-wise Fisher information matrix.*

Pin-Tuning: Parameter-efficient In-Context Tuning for Few-shot Molecular Property Prediction

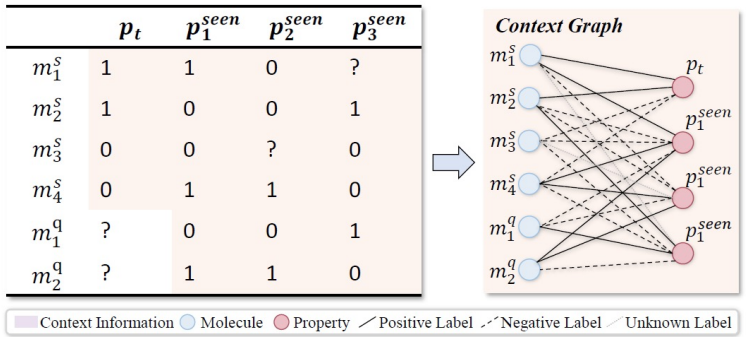


Our FSMPP framework.



Our Pin-Tuning method for PMEs.

Enabling contextual perceptiveness in MP-Adapter



	p_t	p_1^{seen}	p_2^{seen}	p_3^{seen}
m_1^s	1	1	0	?
m_2^s	1	0	0	1
m_3^s	0	0	?	0
m_4^s	0	1	1	0
m_1^q	?	0	0	1
m_2^q	?	1	1	0

Figure 3: Convert the context information of a 2-shot episode into a context graph.

$$\mathbf{C} = \texttt{ContextEncoder}(\mathcal{V}_t, \mathbf{A}_t, \mathbf{X}_t)$$

$$\boldsymbol{z}^{(l)} = \texttt{FeedForward}_{\texttt{down}}(\boldsymbol{h}_v^{(l)} \| \boldsymbol{c}_m \| \boldsymbol{c}_p),$$

Experiment Results

Table 1: ROC-AUC scores (%) on benchmark datasets, compared with methods trained from scratch (first group) and methods that leverage pre-trained molecular encoder (second group). The best is marked with **boldface** and the second best is with underline. Δ*Improve.* indicates the relative improvements over the baseline models in percentage.

Model	Tox21		SIDER		MUV		ToxCast		PCBA	
	10-shot	5-shot	10-shot	5-shot	10-shot	5-shot	10-shot	5-shot	10-shot	5-shot
Siamese	80.40(0.35)	-	71.10(4.32)	-	59.96(5.13)	-	-	-	-	-
ProtoNet	74.98(0.32)	72.78(3.93)	64.54(0.89)	64.09(2.37)	65.88(4.11)	64.86(2.31)	68.87(0.43)	66.26(1.49)	64.93(1.94)	62.29(2.12)
MAML	80.21(0.24)	69.17(1.34)	70.43(0.76)	60.92(0.65)	63.90(2.28)	63.00(0.61)	68.30(0.59)	67.56(1.53)	66.22(1.31)	65.25(0.75)
TPN	76.05(0.24)	75.45(0.95)	67.84(0.95)	66.52(1.28)	65.22(5.82)	65.13(0.23)	69.47(0.71)	66.04(1.14)	67.61(0.33)	63.66(1.64)
EGNN	81.21(0.16)	76.80(2.62)	72.87(0.73)	60.61(1.06)	65.20(2.08)	63.46(2.58)	74.02(1.11)	67.13(0.50)	69.92(1.85)	67.71(3.67)
IterRefLSTM	81.10(0.17)	-	69.63(0.31)	-	49.56(5.12)	-	-	-	-	-
Pre-GNN	82.14(0.08)	82.04(0.30)	73.96(0.08)	76.76(0.53)	67.14(1.58)	70.23(1.40)	75.31(0.95)	74.43(0.47)	76.79(0.45)	75.27(0.49)
Meta-MGNN	82.97(0.10)	76.12(0.23)	75.43(0.21)	66.60(0.38)	68.99(1.84)	64.07(0.56)	76.27(0.56)	75.26(0.43)	72.58(0.34)	72.51(0.52)
PAR	84.93(0.11)	83.95(0.15)	78.08(0.16)	77.70(0.34)	69.96(1.37)	68.08(2.42)	79.41(0.08)	76.89(0.32)	73.71(0.61)	72.79(0.98)
GS-Meta	86.67(0.41)	86.43(0.02)	84.36(0.54)	84.57(0.01)	66.08(1.25)	64.50(0.20)	83.81(0.16)	82.65(0.35)	79.40(0.43)	77.47(0.29)
Pin-Tuning	**91.56**(2.57)	**90.95**(2.33)	**93.41**(3.52)	**92.02**(3.01)	**73.33**(2.00)	**70.71**(1.42)	**84.94**(1.09)	**83.71**(0.93)	**81.26**(0.46)	**79.23**(0.52)
Δ*Improve.*	5.64%	5.23%	10.73%	8.81%	4.82%	3.86%	1.35%	1.28%	2.34%	2.27%

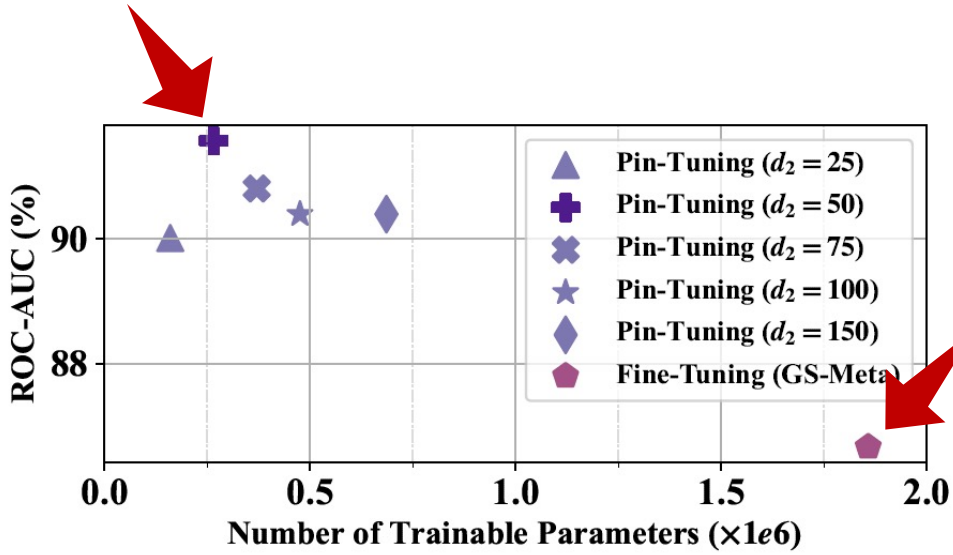
Tunable Parameter Size Analysis

$$N_{Fine-Tuning} = |E_n|d + L(|E_e|d + 2dd_1 + 3d + d_1).$$

$$N_{Pin-Tuning} = |E_n|d + L(|E_e|d + 2dd_2 + 3d + d_2).$$

$$\Delta N = (d_1 - d_2)L(2d + 1).$$

Ours (14.2% parameters, higher performance)



Baseline

Figure 5: ROC-AUC (%) and number of trainable parameters of Pin-Tuning with varied value of d_2 and full Fine-Tuning method (e.g., GS-Meta) on the Tox21 dataset.

Visualization

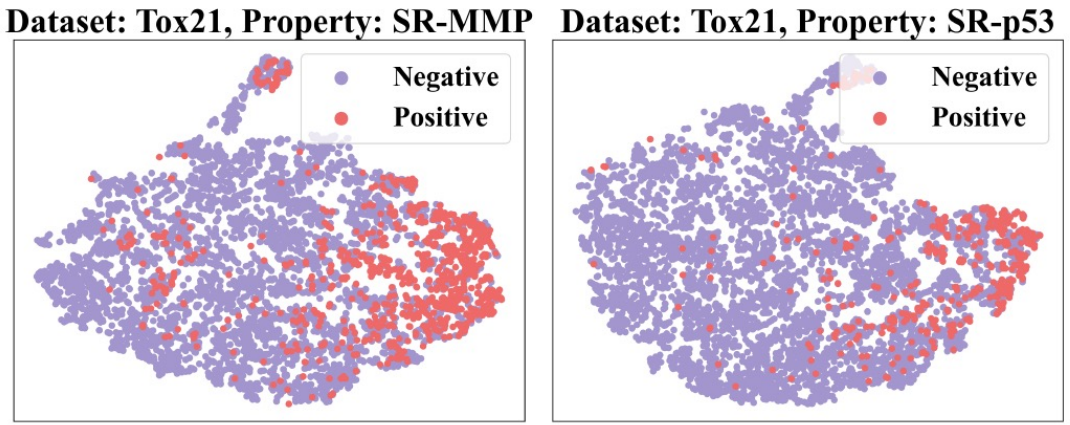


Figure 6: Molecular representations encoded by GS-Meta [58].

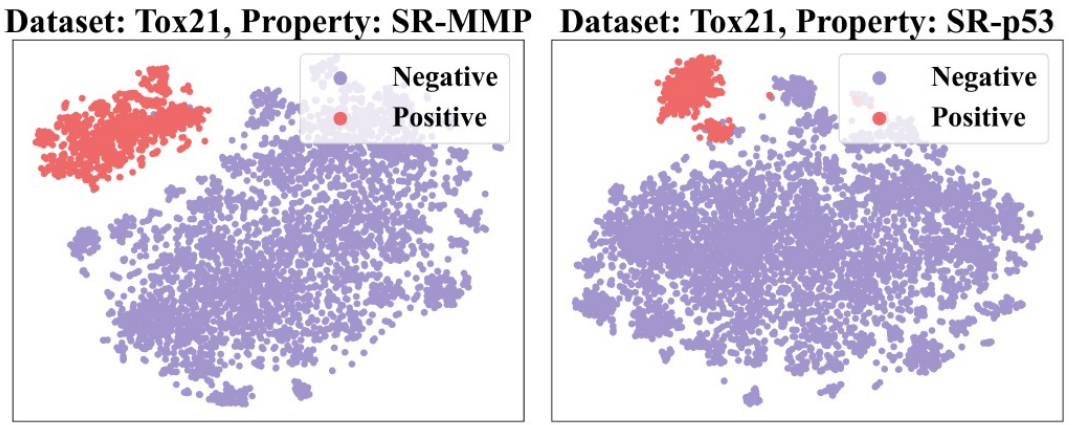


Figure 7: Molecular representations encoded by `Pin-Tuning`.

中国科学院自动化研究所
Institute of Automation
Chinese Academy of Sciences

Thank you for your attention!

Contact： liang.wang@cripac.ia.ac.cn