Strategies for Pre-training Graph Neural Networks

W. Hu, B. Liu, J. Gomes, M. Zitnik

P. Liang, V. Pande and J. Leskovec

Stanford University,
The University of Iowa and Harvard University

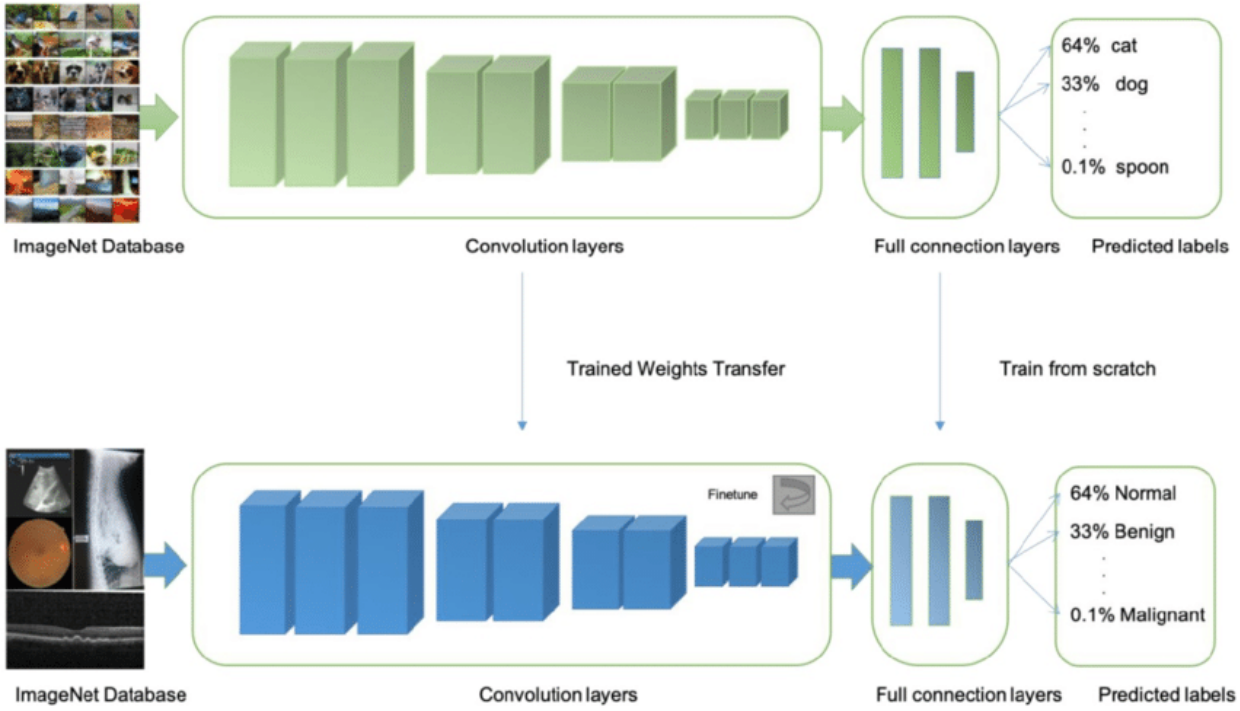
ICLR 2020

cheon.research@gmail.com

서울시립대학교
UNIVERSITY OF SEOUL

2020. 10. 20.

Transfer Learning (ImageNET, VGG, Resnet, BERT, GPT...)



https://www.researchgate.net/publication/336874848_Current_status_and_future_trends_of_clinical_diagnoses_via_image-based_deep_learning

서울시립대학교 UNIVERSITY OF SEOUL

Challenges with Learning on Graph Data

1. Scarcity of labeled data

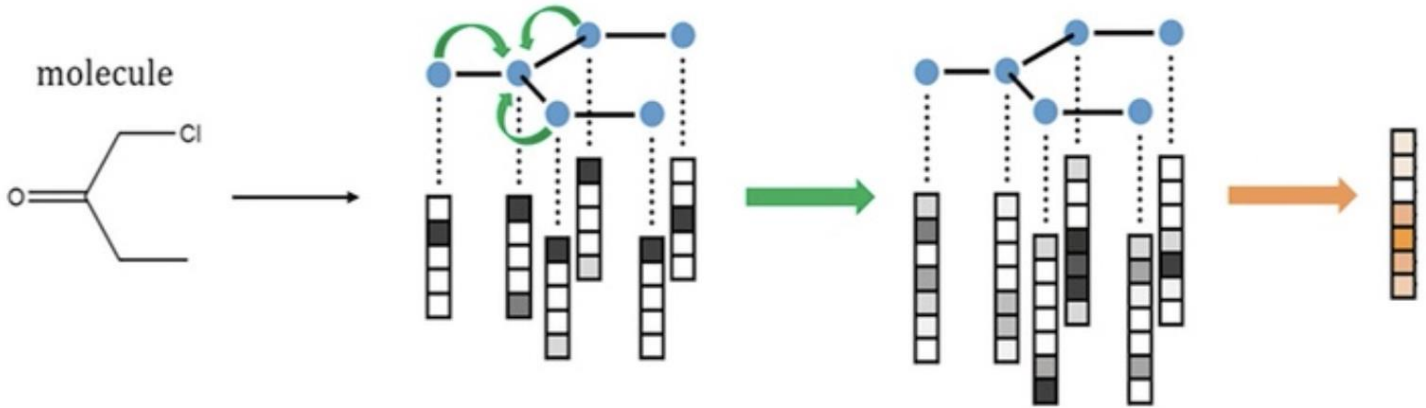
- ✓ Test examples tend to be very different from training examples.
- ✓ GNNs overfit to small training datasets.

2. Out-of-distribution Prediction

- ✓ Obtaining labels requires expensive lab experiments.
- ✓ GNNs extrapolate poorly.

Supervised Learning of Graphs

✓ To learn a representation vector h_G that predicts the label of a graph G.



Graph Neural Networks (GNNs)

- ✓ GNNs use the graph connectivity as well as node and edge features to learn a representation vector h_v for every node.

k-hop neighborhood

feature vector of edge between u and v

$$h_v^{(k)} = \text{COMBINE}^{(k)}\left(h_v^{(k-1)}, \text{AGGREGATE}^{(k)}\left(\left\{\left(h_v^{(k-1)}, h_u^{(k-1)}, e_{uv}\right): u \in \mathcal{N}(v)\right\}\right)\right)$$

representation of node v at the k-th iteration

set of neighbors of v

Graph Representation Learning

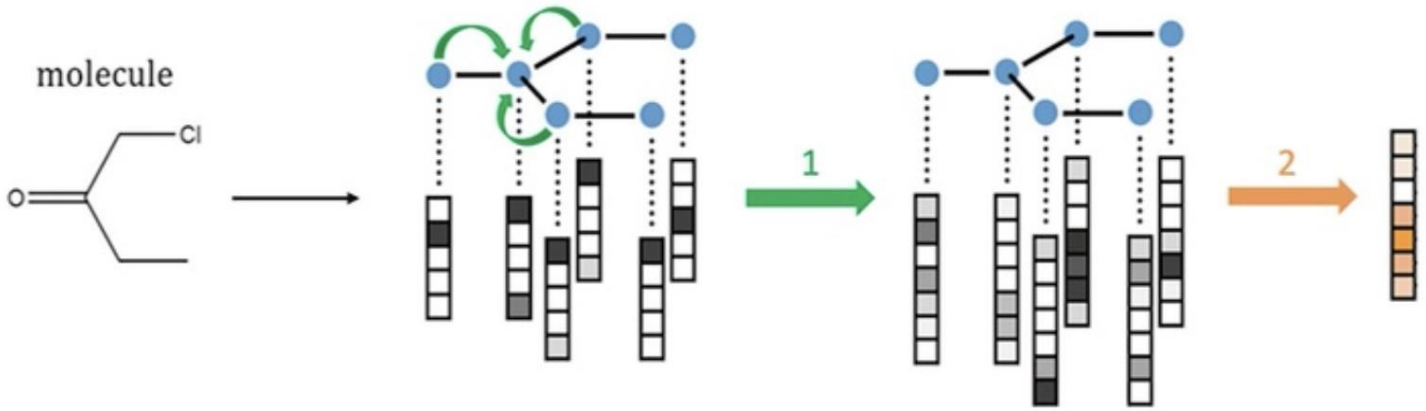
✓ The READOUT function pools node features from the final iteration *K* to obtain the entire graph's representation h_G.

permutation-invariant function

$$h_G = \mathrm{READOUT}\left(\left\{h_v^{(K)} \mid v \in G\right\}\right)$$

graph representation

Graph Representation Learning

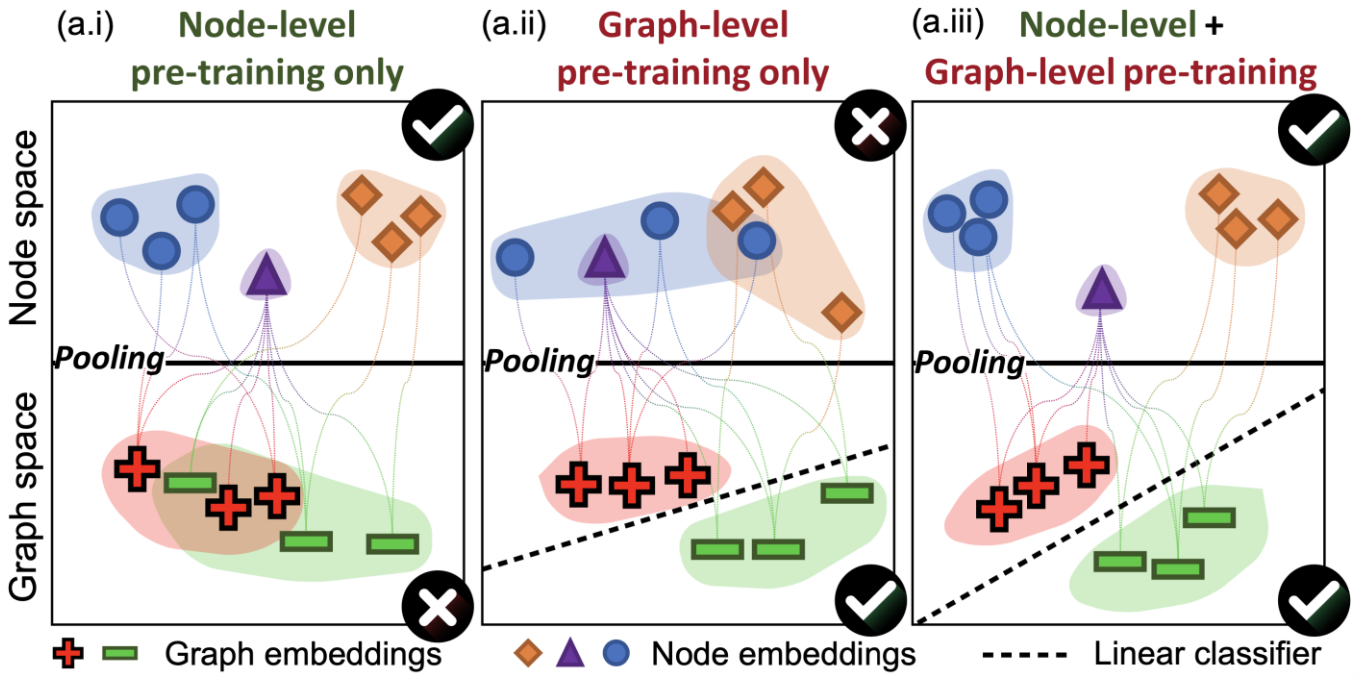


1. Iteratively aggregating neighboring information to obtain node embeddings.

2. Pooling node embeddings to obtain graph embedding.

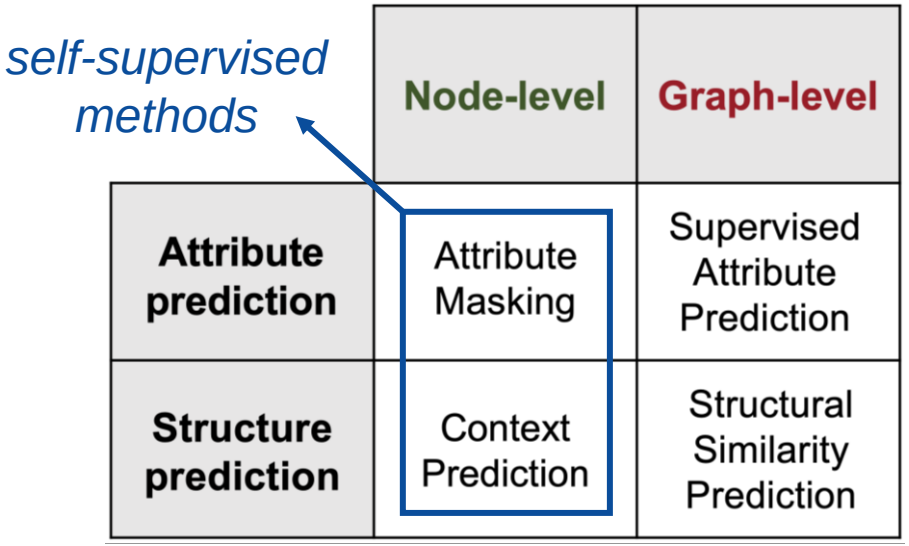
Strategies for Pre-training GNNs

✓ **Key idea:** Pre-train both **node** and **graph** embeddings.



Strategies for Pre-training GNNs

✓ **Key idea** Pre-train both **node** and **graph** embeddings.



Node-level Pre-training

✓ To use easily-accessible unlabeled data to capture domain-specific knowledge/regularities in the graph.

❑ **Context Prediction**: Exploiting Distribution of Graph Structure

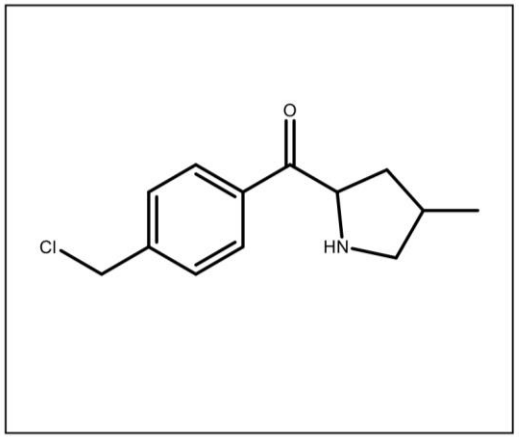
- ✓ To use subgraphs to predict their surrounding graph structures.
- ✓ Mapping nodes appearing in similar structural contexts to nearby embeddings.

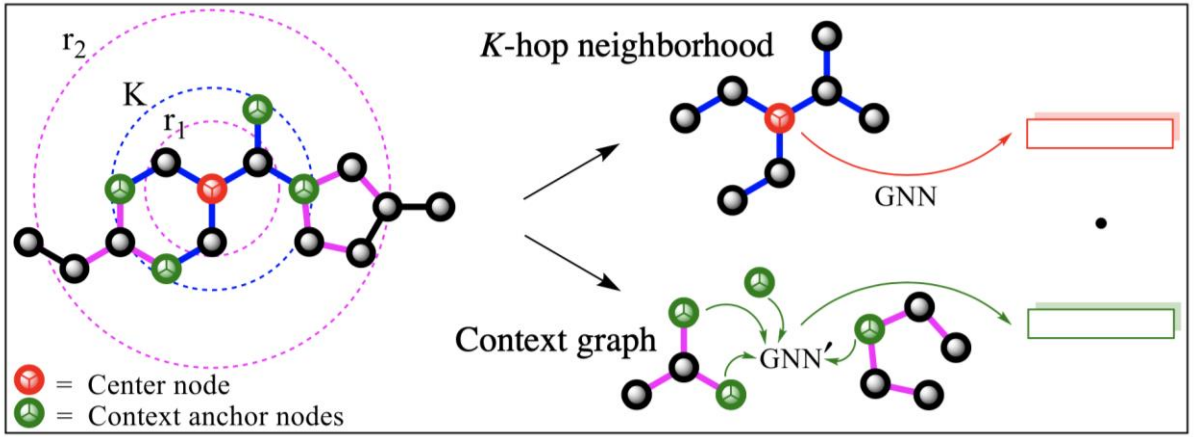
❑ **Attribute Masking**: Exploiting Distribution of Graph Attributes

- ✓ To capture domain knowledge by learning the regularities of the node/edge attributes distributed over graph structure.

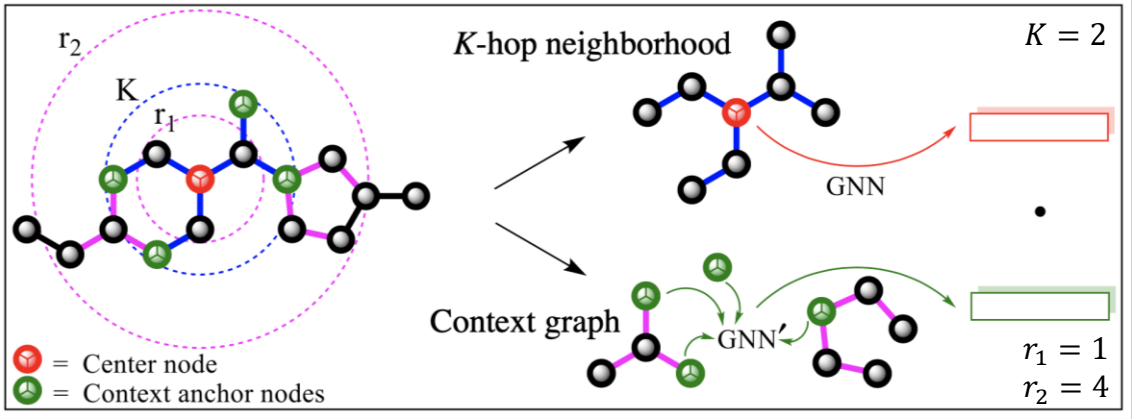
Neighborhood and Context Graphs

- ✓ To pre-train a GNN so that it maps nodes appearing in similar structural contexts to nearby embeddings.





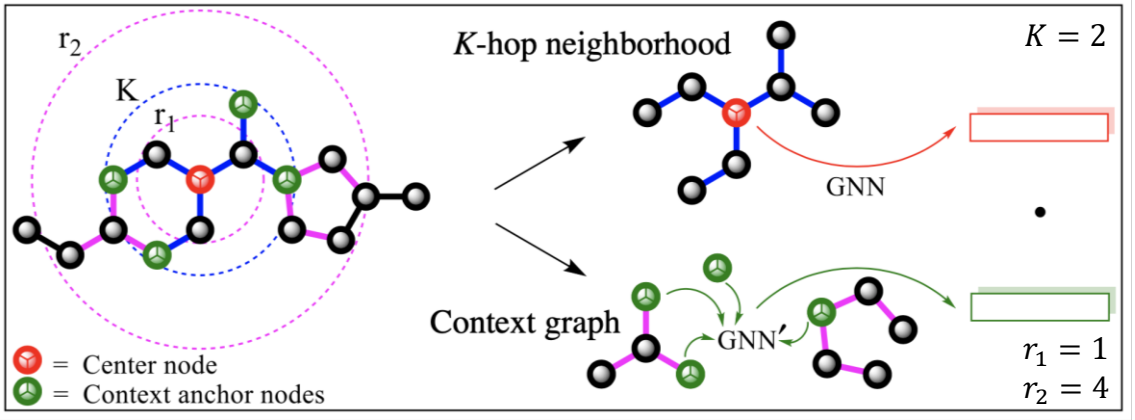
Neighborhood Graphs



✓ Motivated by *K*-layer GNN aggregation.

✓ *K*-hop neighborhood contains all nodes and edges that are at most *K*-hops.

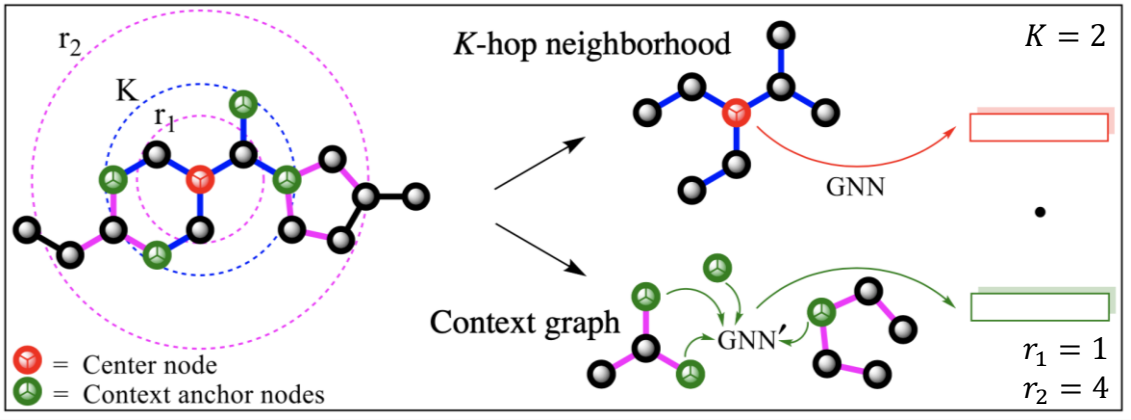
Context Graphs



✓ Context graph of v as graph structure that surrounds v's neighborhood.

✓ Context graph is a subgraph that is between r_1-hops and r_2-hops.

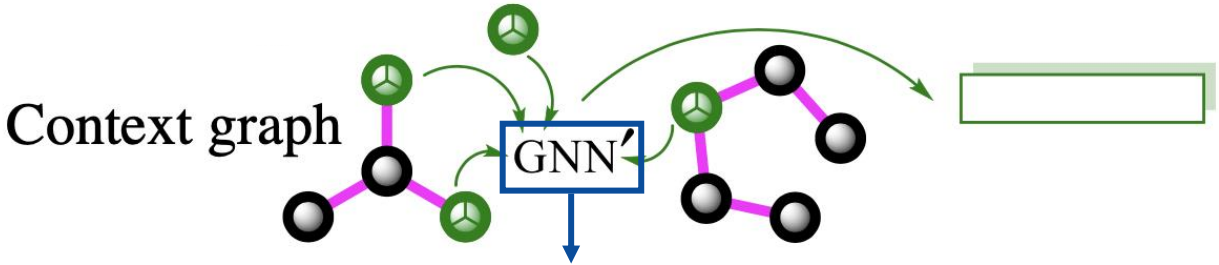
Context Anchor Nodes



- ✓ Context anchor nodes are shared between the neighborhood and the context graph.
- ✓ Context anchor nodes provide information about how the neighborhood and context graphs are connected with each other.

Encoding context into a fixed vector using an auxiliary GNN

- ✓ Directly predicting the context graph is intractable.
- ✓ To encode context graph as *fixed-length vectors* for context prediction.
- ✓ To average embeddings of *context anchor nodes* to obtain a *fixed-length context embedding* c_v^G.



Learning via Negative Sampling

- ✓ The learning objective of Context Prediction is a binary classification of whether a particular neighborhood and a particular context graph belong to the same node.

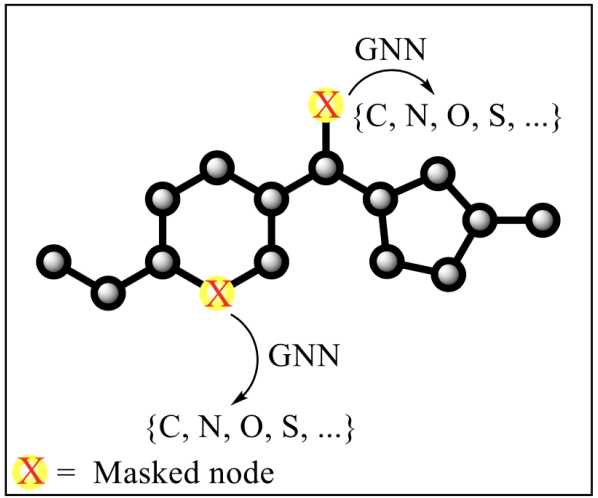
node embedding *indicator function*

$$\sigma\left(h_v^{(K)\top} c_{v'}^{G'}\right) \approx \mathbf{1}\{v \text{ and } v' \text{ are the same nodes}\}$$

sigmoid function *context embedding*

Masking node and Edges Attributes

- ✓ Masking node/edge attributes and let GNNs predict attributes based on neighboring structure.
- ✓ Beneficial for richly-annotated graphs from scientific domains.

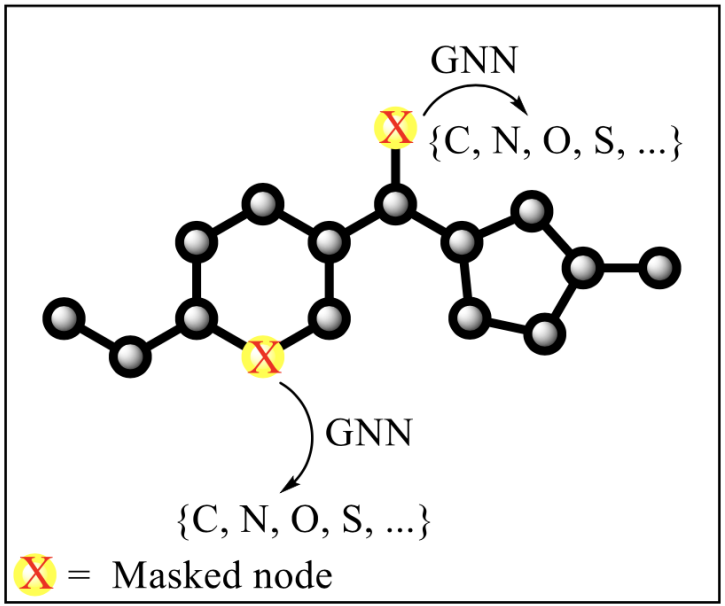


Masking node and Edges Attributes

1. Mask node/edge attributes.

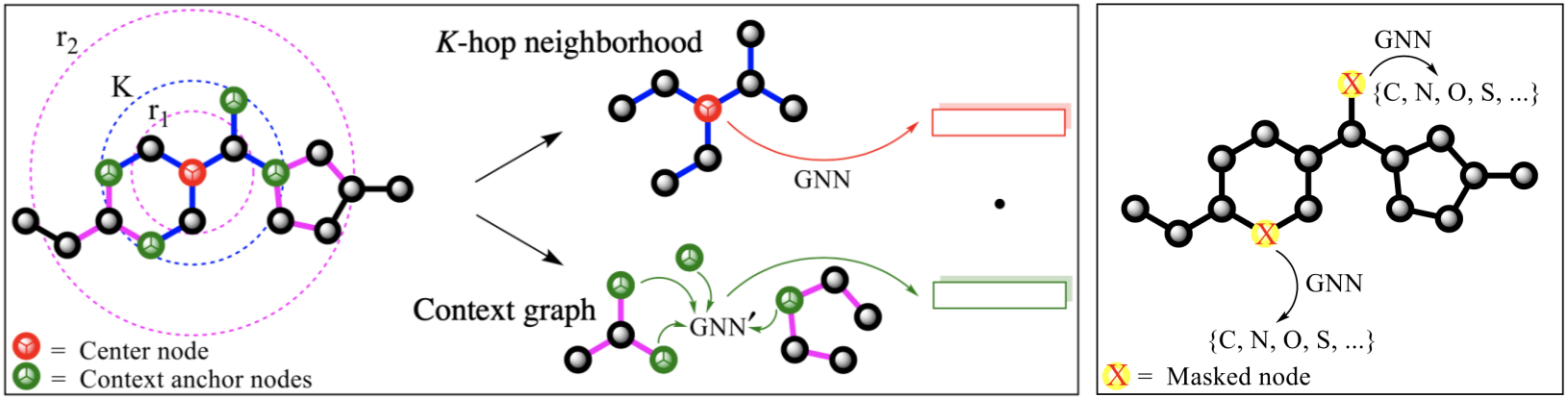
2. Obtain the corresponding node/edge embedding applying GNN.

3. Predict a masked node/edge attribute from a linear model that is applied on top of embeddings.



Node-level Pre-training

- ✓ To use easily-accessible unlabeled data to capture domain-specific knowledge/regularities in the graph.



Graph-level Pre-training

✓ To generate useful graph embeddings composed of the meaningful node embeddings obtained by node-level pre-training.

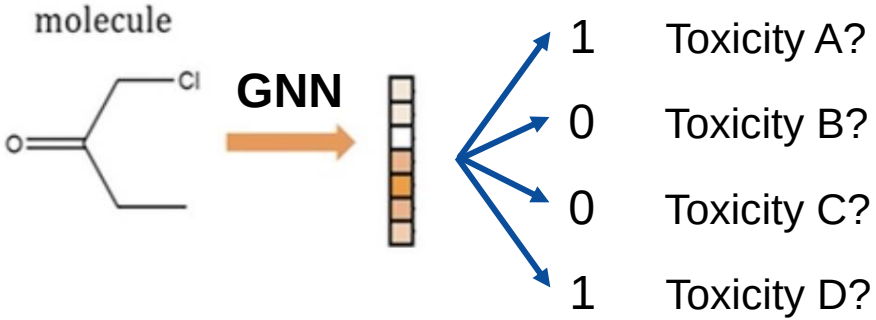
	Node-level	Graph-level
Attribute prediction	Attribute Masking	Supervised Attribute Prediction
Structure prediction	Context Prediction	Structural Similarity Prediction

Making predictions about domain-specific attributes of entire graphs.

Making predictions about graph structure.

Supervised Graph-level Property Prediction

- ✓ Inject graph-level domain-specific knowledge into our pretrained embeddings by defining supervised graph-level prediction tasks.
- ✓ Graph-level multi-task supervised pre-training to jointly predict a diverse set of supervised labels of individual graphs



Negative Transfer

- ✓ Naively performing the extensive multi-task graph-level pre-training alone can fail to give transferable graph-level representations.

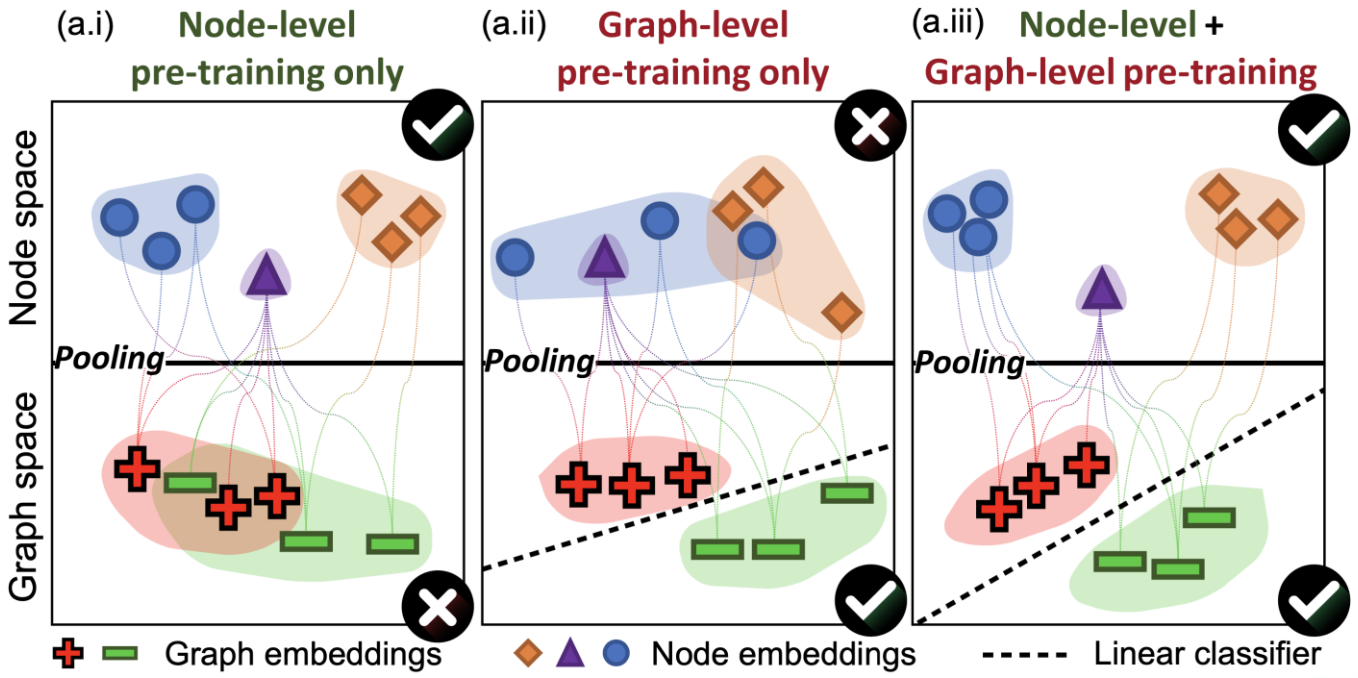
- ✓ To select "truly-relevant" supervised pre-training tasks.

  → selecting the relevant task requires significant domain expertise.

- ✓ To first **regularize** GNNs at the level of individual nodes via node-level pre-training methods before performing graph-level pre-training.

Strategies for Pre-training GNNs

✓ **Key idea:** Pre-train both **node** and **graph** embeddings.



Structural Similarity Prediction

- ✓ To define a graph-level predictive task where the goal would be to model the structural similarity of two graphs.
- ✓ However, finding the ground truth graph distance values is a difficult problem, and in large datasets there is a quadratic number of graph pairs to consider.

future work...

Pre-training GNNs and Fine-tunning for Downstream Tasks

1. **Node-level** self-supervised pre-training on unlabeled data

2. **Graph-level** multi-task supervised pre-training on labeled data

3. **Fine-tune** on downstream data (add linear classifiers)

Experiments

Pre-training Datasets

❑ For the Chemistry Domain

- ✓ 2 million unlabeled molecules sampled from the ZINC15 database.
- ✓ 456K molecules with 1310 kinds of diverse and extensive biochemical assays.

❑ For the Biology Domain

- ✓ 395K unlabeled protein ego-networks derived from PPI networks of 50 species.
- ✓ 88K labeled protein ego-network to jointly predict 5000 coarse-grained biological.

Downstream Classification Datasets

❑ For the Chemistry Domain

- ✓ 8 binary classification tasks
- ✓ scaffold split

❑ For the Biology Domain

- ✓ 40 binary classification tasks
- ✓ species split

Experimental Setup

❑ GNN architectures

- ✓ mainly study Graph Isomorphism Networks(GINs, 2019)
- ✓ GCN(2016), GraphSAGE(2017) and GAT(2019)
- ✓ 5 GNN layers($K = 5$)
- ✓ Average pooling for the READOUT function

❑ Pre-training

- ✓ Molecular networks ($r_1 = 4, r_2 = 7$)
- ✓ PPI networks ($r_1 = 1, r_2 = 4$)
- ✓ 3 GNN layers to encode the context structure
- ✓ Randomly mask 15% of node(molecular) or edge attribute(PPI)

Test ROC-AUC Performance on Molecular Prediction Using Different Pre-training Strategies with GIN

negative transfer

Dataset		BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	HIV	BACE	**Average**
# Molecules		2039	7831	8575	1427	1478	93087	41127	1513	/
# Binary prediction tasks		1	12	617	27	2	17	1	1	/
Pre-training strategy		Out-of-distribution prediction (scaffold split)								
Graph-level	Node-level									
–	–	65.8 ±4.5	74.0 ±0.8	63.4 ±0.6	57.3 ±1.6	58.0 ±4.4	71.8 ±2.5	75.3 ±1.9	70.1 ±5.4	67.0
–	Infomax	**68.8 ±0.8**	75.3 ±0.5	62.7 ±0.4	58.4 ±0.8	69.9 ±3.0	75.3 ±2.5	76.0 ±0.7	75.9 ±1.6	70.3
–	EdgePred	67.3 ±2.4	76.0 ±0.6	64.1 ±0.6	60.4 ±0.7	64.1 ±3.7	74.1 ±2.1	76.3 ±1.0	79.9 ±0.9	70.3
–	AttrMasking	64.3 ±2.8	76.7 ±0.4	64.2 ±0.5	61.0 ±0.7	71.8 ±4.1	74.7 ±1.4	77.2 ±1.1	79.3 ±1.6	71.1
–	ContextPred	68.0 ±2.0	75.7 ±0.7	63.9 ±0.6	60.9 ±0.6	65.9 ±3.8	75.8 ±1.7	77.3 ±1.0	79.6 ±1.2	70.9
Supervised	–	68.3 ±0.7	77.0 ±0.3	64.4 ±0.4	62.1 ±0.5	57.2 ±2.5	79.4 ±1.3	74.4 ±1.2	76.9 ±1.0	70.0
Supervised	Infomax	68.0 ±1.8	77.8 ±0.3	64.9 ±0.7	60.9 ±0.6	**71.2 ±2.8**	**81.3 ±1.4**	77.8 ±0.9	80.1 ±0.9	72.8
Supervised	EdgePred	66.6 ±2.2	**78.3 ±0.3**	**66.5 ±0.3**	**63.3 ±0.9**	70.9 ±4.6	78.5 ±2.4	77.5 ±0.8	79.1 ±3.7	72.6
Supervised	AttrMasking	66.5 ±2.5	77.9 ±0.4	65.1 ±0.3	**63.9 ±0.9**	**73.7 ±2.8**	**81.2 ±1.9**	77.1 ±1.2	80.3 ±0.9	73.2
Supervised	ContextPred	**68.7 ±1.3**	**78.1 ±0.6**	65.7 ±0.6	62.7 ±0.8	**72.6 ±1.5**	**81.3 ±2.1**	**79.9 ±0.7**	**84.5 ±0.7**	**74.2**

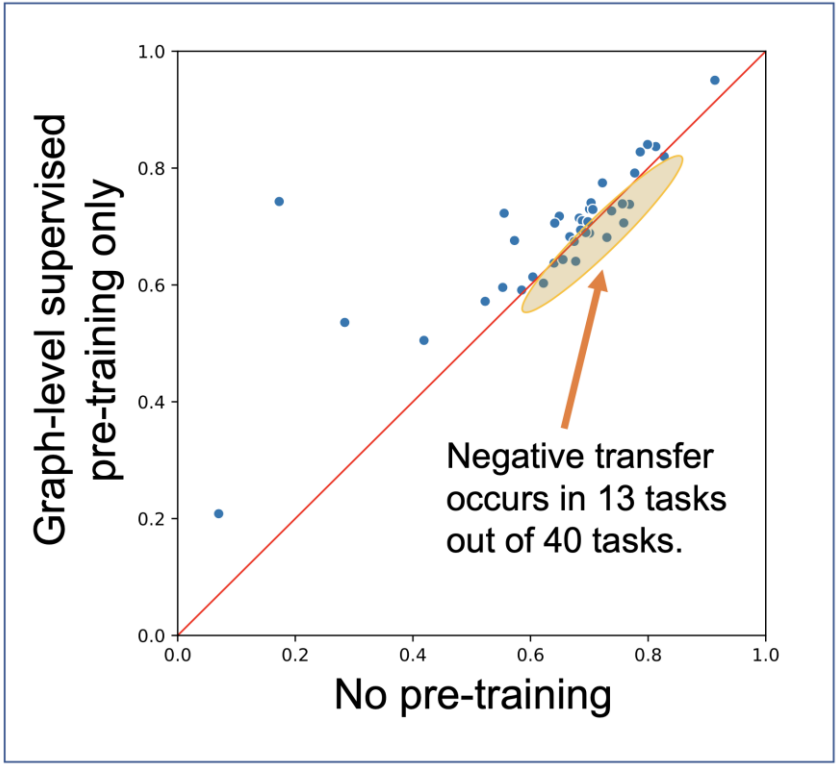
Test ROC-AUC Performance of Different GNN Architectures with and without Pre-training

	Chemistry			Biology		
	Non-pre-trained	Pre-trained	Gain	Non-pre-trained	Pre-trained	Gain
GIN	67.0	**74.2**	**+7.2**	64.8 ± 1.0	**74.2 ± 1.5**	**+9.4**
GCN	**68.9**	72.2	+3.4	63.2 ± 1.0	70.9 ± 1.7	+7.7
GraphSAGE	68.3	70.3	+2.0	65.7 ± 1.2	68.5 ± 1.5	+2.8
GAT	66.8	60.3	-6.5	**68.2 ± 1.1**	67.8 ± 3.6	-0.4

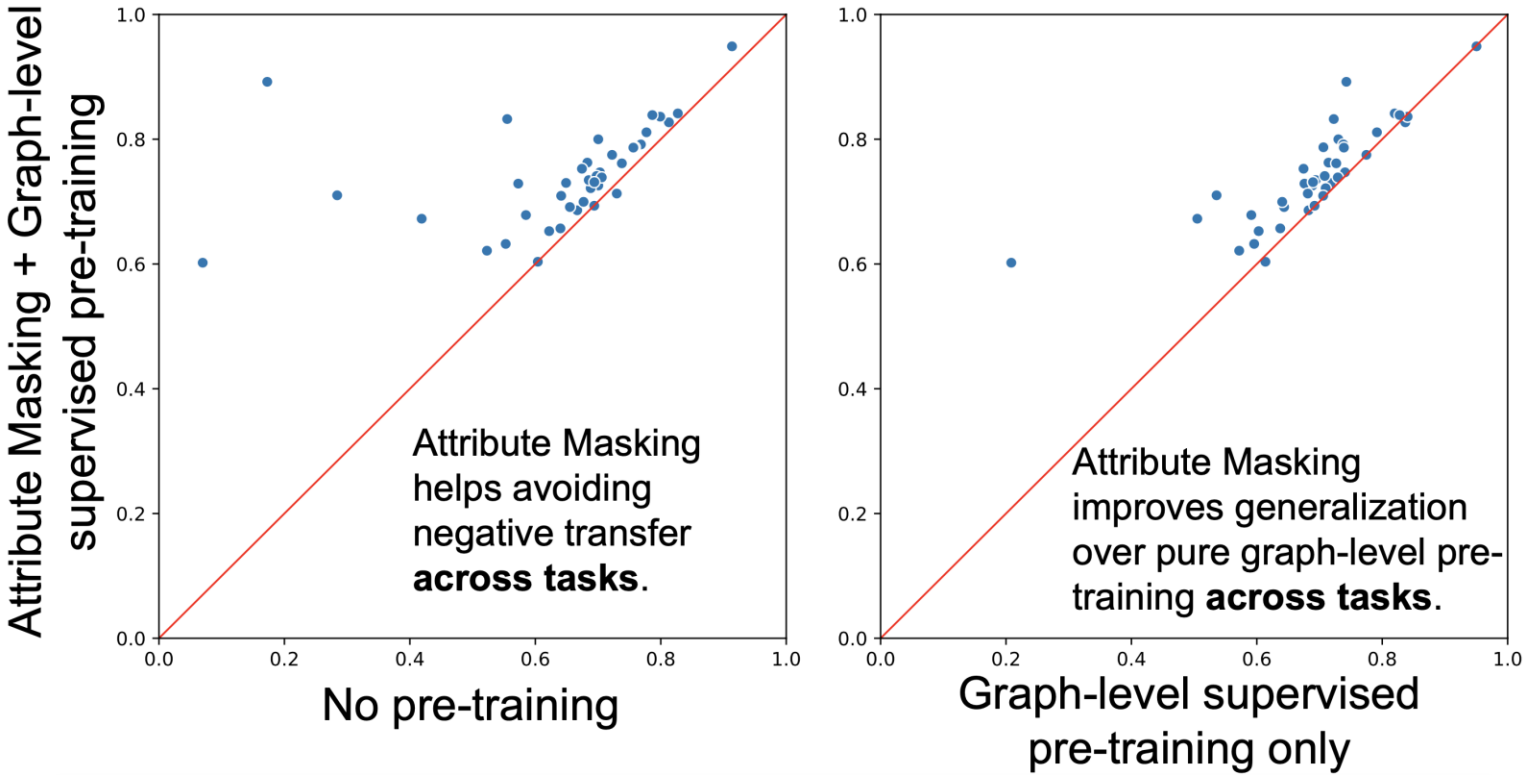
Test ROC-AUC of Protein Function Prediction Using Different Pre-training Strategies with GIN

Pre-training strategy		Out-of-dist. (species split)
Graph-level	Node-level	
–	–	64.8 ±1.0
–	Infomax	64.1 ±1.5
–	EdgePred	65.7 ±1.3
–	ContextPred	65.2 ±1.6
–	AttrMasking	64.4 ±1.3
Supervised	–	69.0 ±2.4
Supervised	Infomax	72.8 ±1.5
Supervised	EdgePred	72.3 ±1.4
Supervised	ContextPred	73.8 ± 1.0
Supervised	AttrMasking	**74.2 ±1.5**

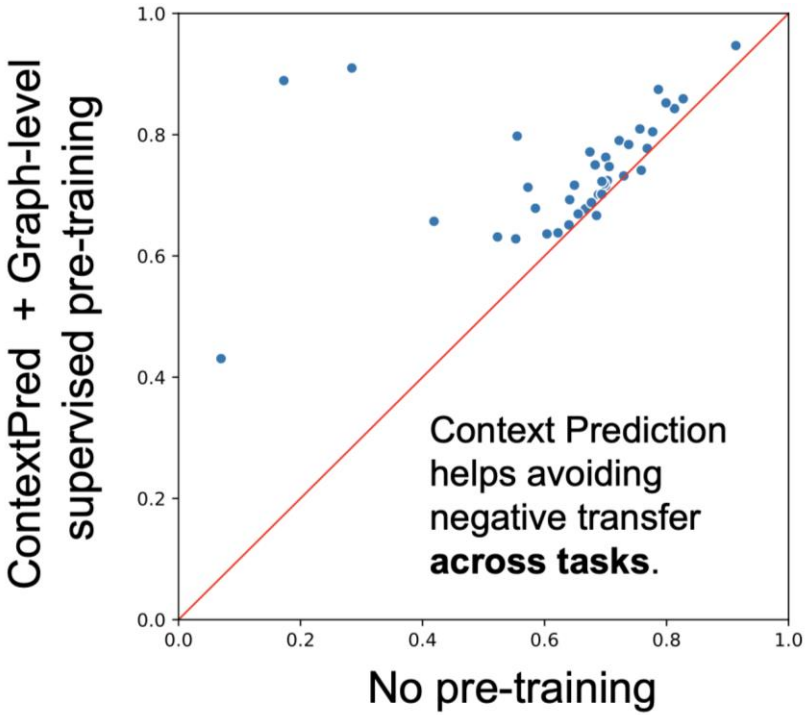
Negative Transfer

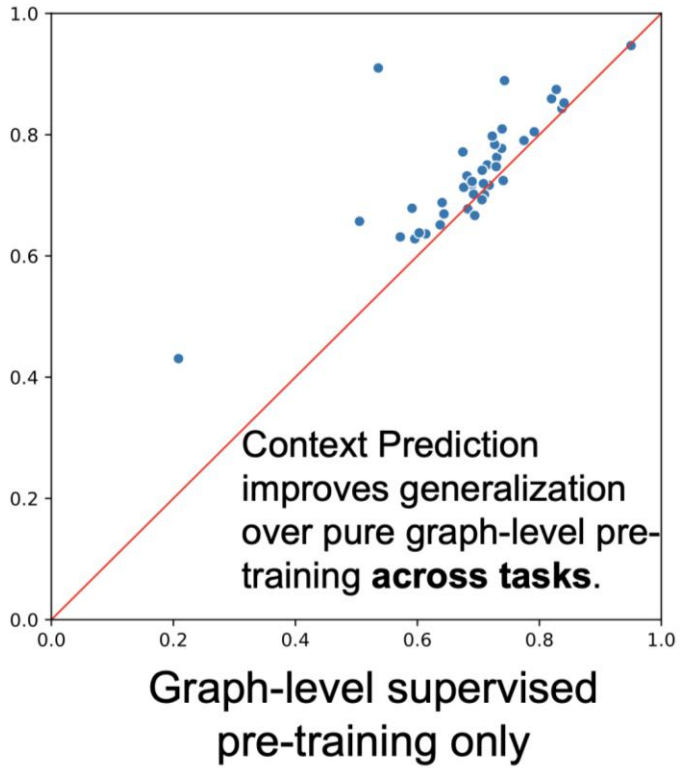


Avoiding Negative Transfer (1)

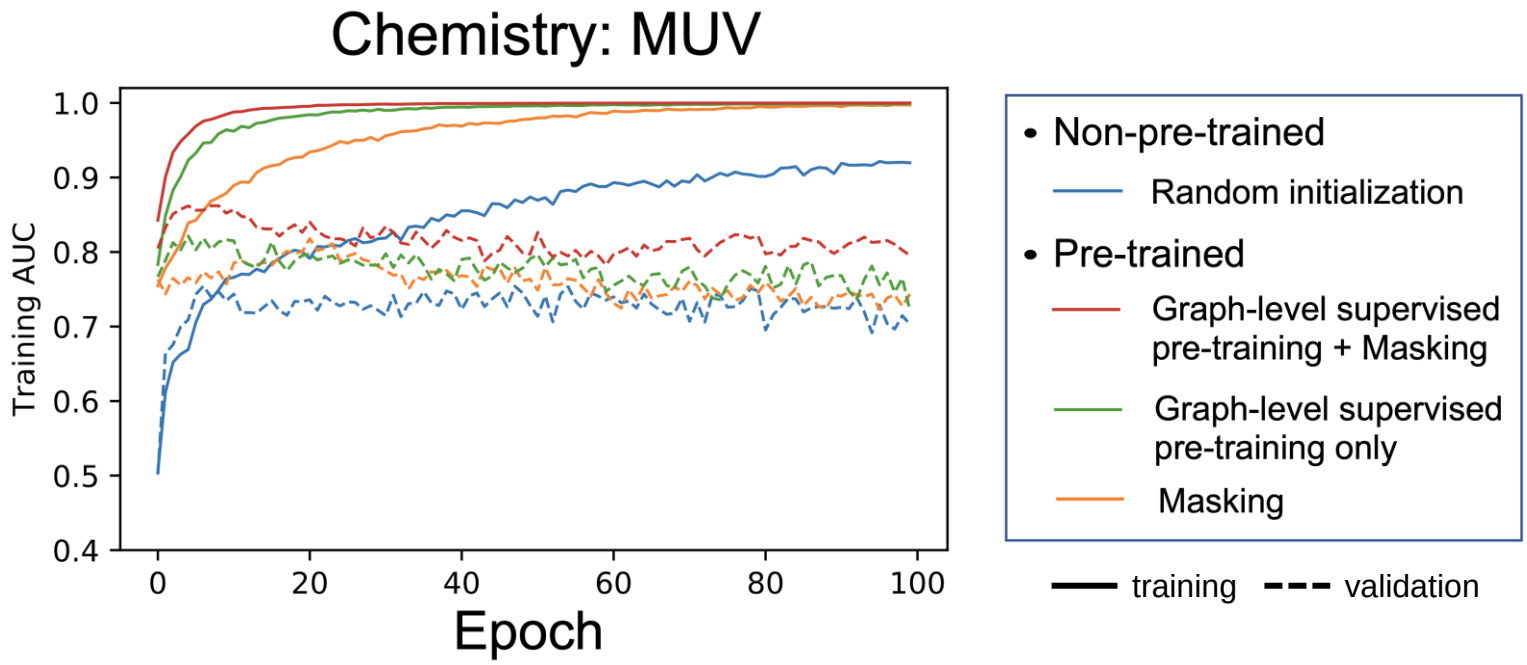


Avoiding Negative Transfer (2)

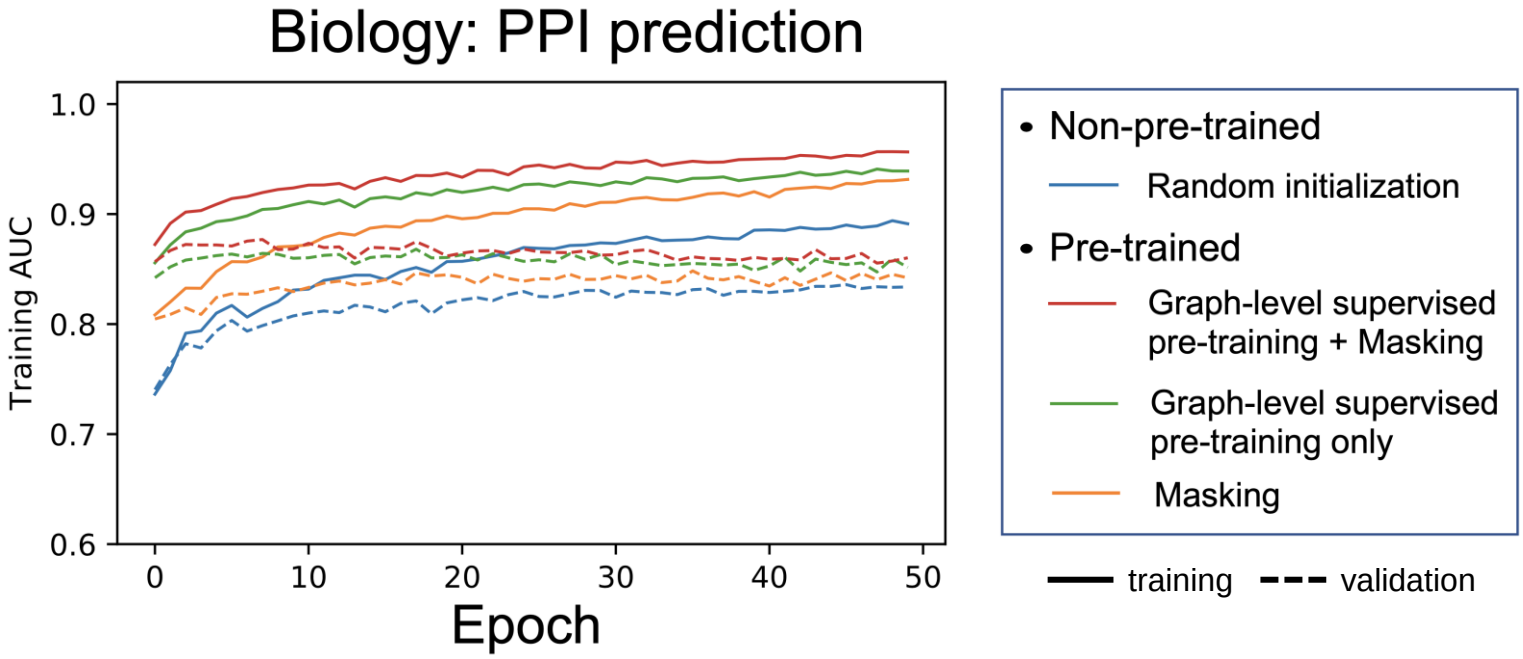




Training and Validation Curves of Different Pre-training strategies on GINs (1)



Training and Validation Curves of Different Pre-training strategies on GINs (2)



Thank you ☺