Implicit Subgraph Neural Network

Yongjian Zhong, **Liao Zhu, Hieu Vu, Bijaya Adhikari**

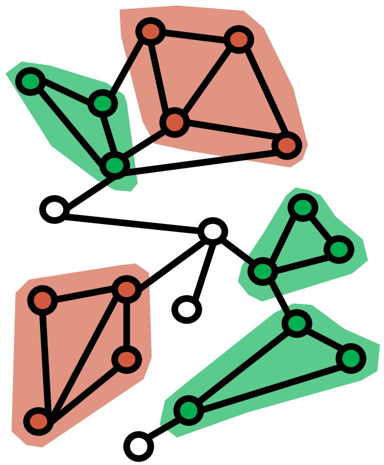
University of Iowa

ICML 2025

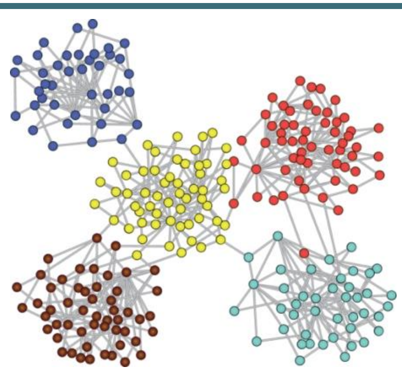
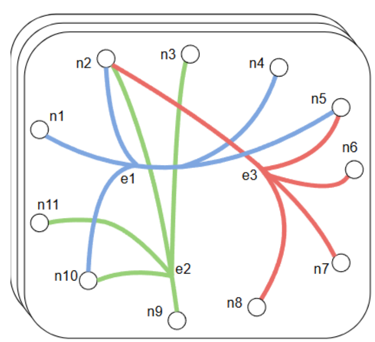
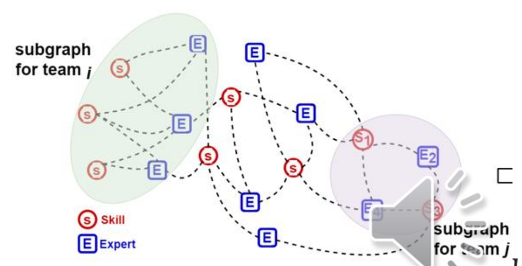
Content

- Background & Challenges
- Our Method
- Experiments

Subgraph Representation Learning

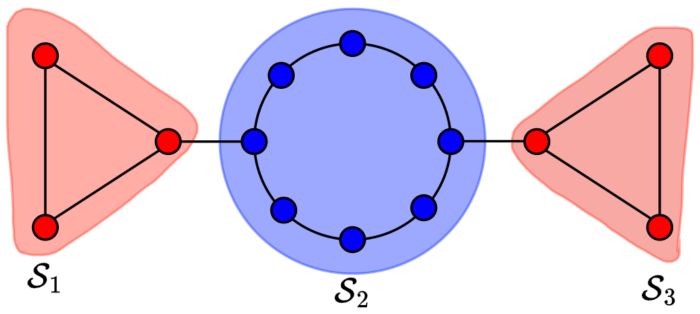


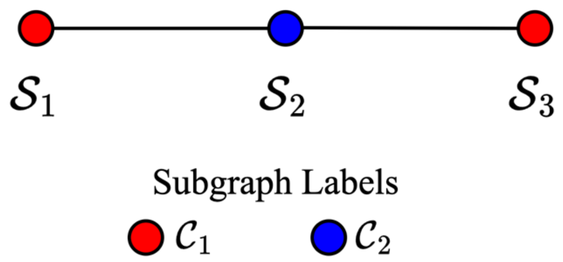
- **Given**
  - A Base Graph G
  - Subgraphs $\{S_i\}_{i=1}^{N}$ of G
- **Output**
  - Embeddings of subgraphs $\{z_i\}_{i=1}^{N}$

Community Detection[1]	Gene Networks[2]	Collaboration Networks[3]
		

1. Zhang, Xingyi, et al. "Constrained social community recommendation." Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining. 2023.
2. Luo, Yuan. "Shine: Subhypergraph inductive neural network." Advances in Neural Information Processing Systems 35 (2022): 18779-18792.
3. Hamidi Rad, Radin, et al. "Subgraph representation learning for team mining." Proceedings of the 14th ACM Web Science Conference 2022. 2022.

Challenges: Incorporating Subgraph Info



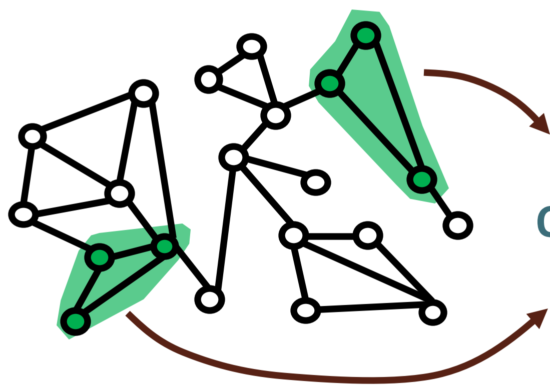


S_1 and S_3 are $\frac{n}{2}$ away in base graph but close in subgraph-level graph.

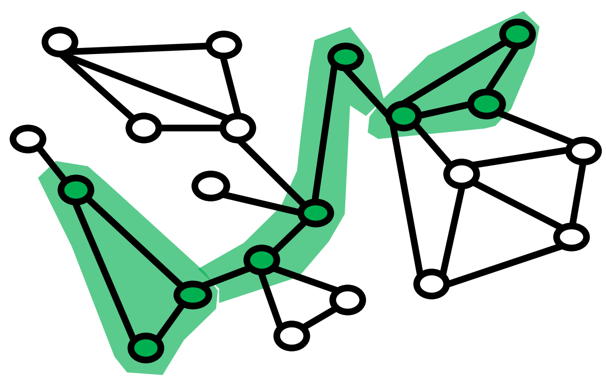
Subgraph relation can help!

Challenges: Long-range Dependencies

- Subgraph can be disconnected.



- Subgraph can have large diameter.



Need long-range dependencies!

Existing Works

- **SubGNN**[1]
  - ✓ Hand-crafted subgraph channels (Neighbor, Structure, Position)
  - ✗ Poor performance
- **GLASS**[2]
  - ✓ Node labeling
  - ✗ Ignore subgraph-level structure
- **SSNP**[3]
  - ✓ Random walk sampling
  - ✗ Ignore subgraph-level structure

How to incorporate subgraph information to improve on existing approaches?

1. Alsentzer, Emily, et al. "Subgraph neural networks." Advances in Neural Information Processing Systems 33 (2020): 8017-8029..
2. Wang, Xiyuan, and Muhan Zhang. "GLASS: GNN with labeling tricks for subgraph representation learning." International conference on learning representations. 2021.
3. Jacob, Shweta Ann, Paul Louis, and Amirali Salehi-Abari. "Stochastic subgraph neighborhood pooling for subgraph classification." Proceedings of the 32nd ACM international conference on information and knowledge management. 2023.

To This End …

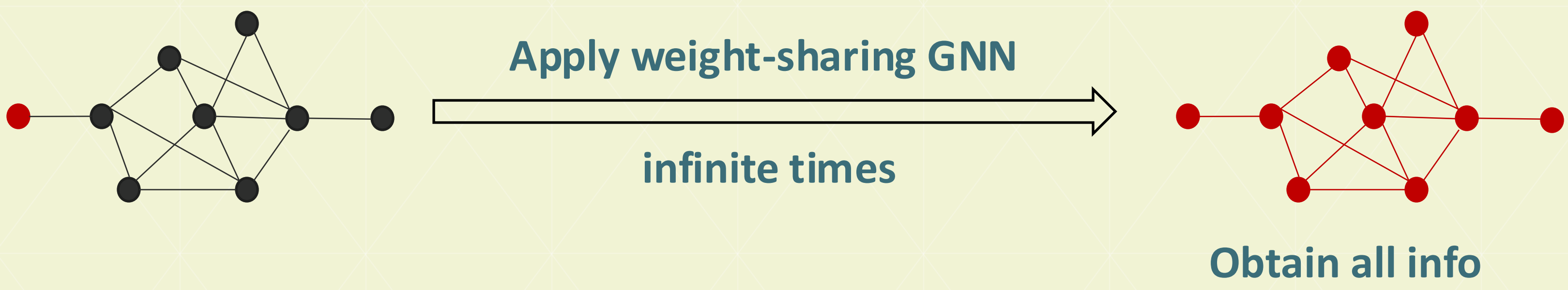
▪ Goal

- Incorporate subgraph information
- Capture long-range dependencies

▪ Our Contributions

- Label-aware hybrid graph
- Implicit subgraph model
- Efficient bilevel optimization for training

Background: Graph Implicit Models



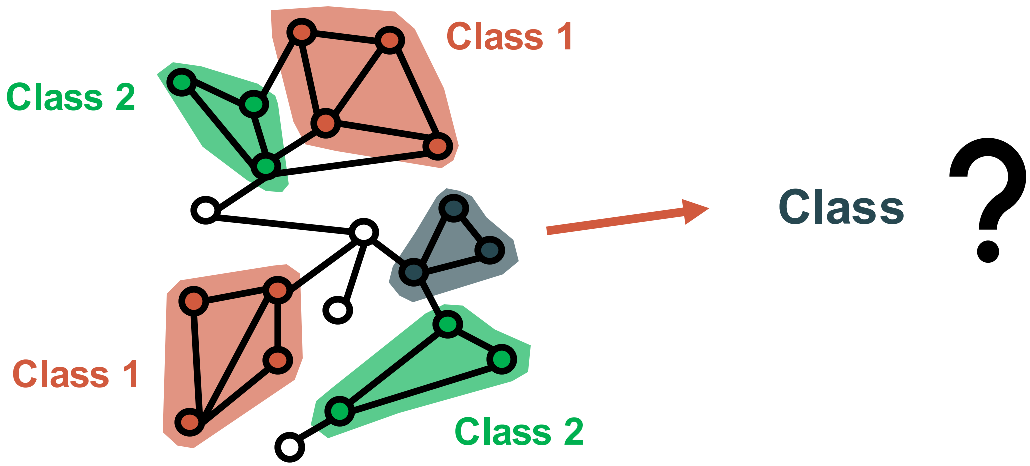
$$Z^{t+1} = \sigma(WZ^{t}A + VX) \xrightarrow{t \to \infty} Z^{*} = \sigma(WZ^{*}A + VX)$$

Content

- Background & Challenges
- **Our Method**
- Experiments

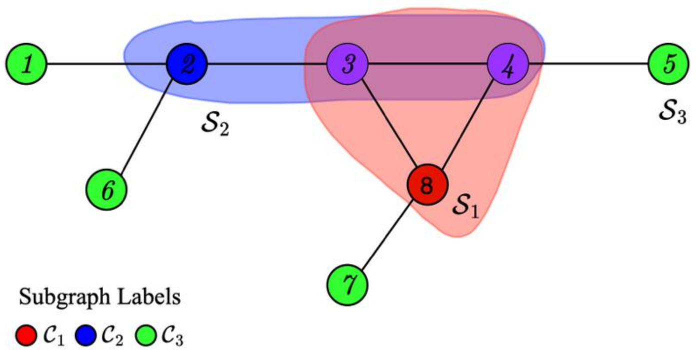
Problem Setup

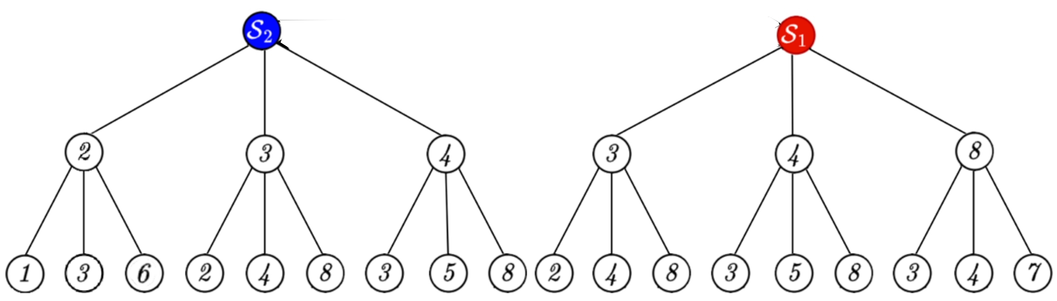
- **Given**
  - A Base Graph G
  - Indices of subgraphs $\{S_i\}_{i=1}^{N}$
- **Do**
  - Subgraph Classification



Subgraph-level Graph

- We construct a subgraph channel that can help the model to distinguish subgraphs.





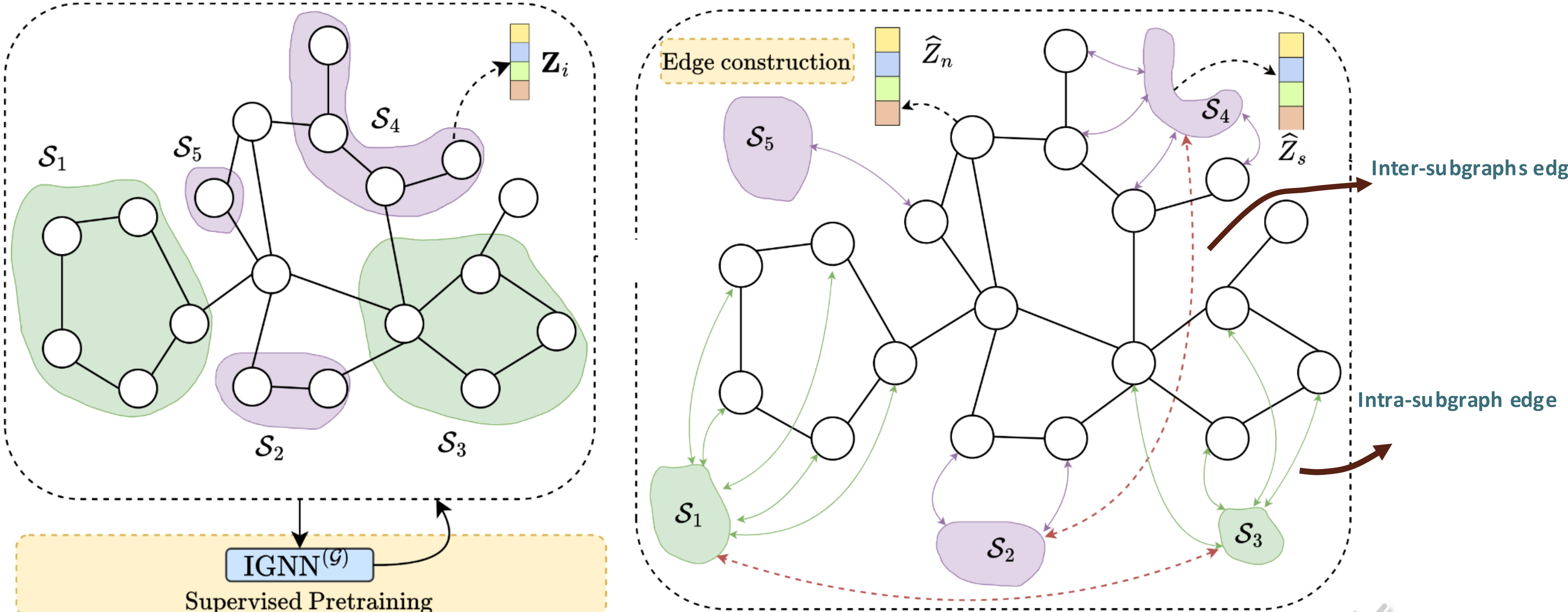
S_1 and S_3 should have the same embeddings considering unit feature.

Even with labeling trick, we cannot distinguish them

Idea: Add an asymmetric edge at the subgraph level

Hybrid Graph Construction

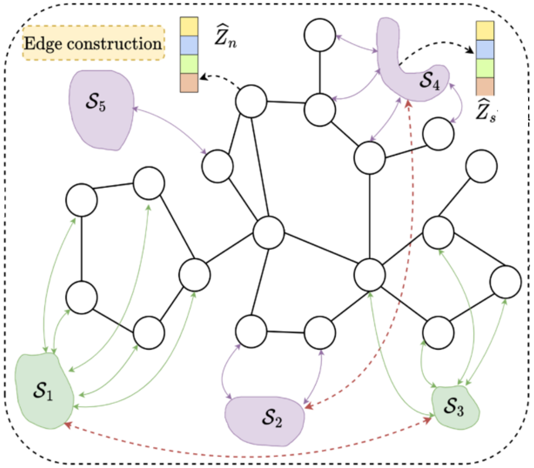
- Get subgraph embeddings through pretraining
- Connect subgraph nodes using embeddings and labels

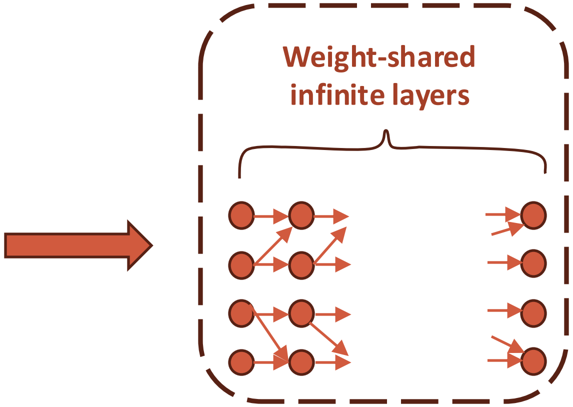


Idea: For each class, we connect k pairs of the most distant subgraph nodes.

Implicit Subgraph Neural Network

- **Implicit Models aim to find the fixed-point embeddings**
  - **A straightforward way: directly using implicit models on the hybrid graph.**





Implicit Models

However, this approach is unstable.

Details in the paper.

Bilevel Formulation

- Objective under bilevel optimization perspective

Classification loss function

Classifier

Embedding of i-th subgraph

$$\min_{\boldsymbol{x}\in\mathcal{X}} F(\boldsymbol{x};\mathbf{Z}^*) \triangleq \sum_{i=1}^{m} \ell(y_i, \phi_\theta([\mathbf{Z}^*]_\mathbf{i}))$$

Fixed point minimizes this problem. Denoted as $g(.)$

$$\text{s.t. } \mathbf{Z}^* \triangleq \arg\min_{\mathbf{Z}} \frac{1}{2}\|\mathbf{Z} - f(\mathbf{Z}, \hat{G}; \xi, \mathbf{W})\|_F^2$$

where f is the implicit model from EIGNN[1], which has form

$$f(\mathbf{Z}, \hat{G}; \xi, \mathbf{W}) = \alpha \mathbf{A}\mathbf{Z}h(\mathbf{W}) + \psi_\xi(\hat{\mathbf{X}})$$

$$h(\mathbf{W}) = \frac{\mathbf{W}^T\mathbf{W}}{\|\mathbf{W}\|\|\mathbf{W}\| + e_h}$$

We propose a bilevel optimization algorithm that solve this objective *efficiently*.

THE UNIVERSITY OF IOWA

1. Liu J, Kawaguchi K, Hooi B, et al. Eignn: Efficient infinite-depth graph neural networks[J]. Advances in Neural Information Processing Systems, 2021, 34: 18762-18773.

Details in the paper.

Bilevel Optimization Algorithm

- The first-order bilevel algorithm for implicit models

Fixed-point iteration

Proxy gradient for penalty term

Algorithm 1 ISNN Training Algorithm

Input: Graph $\hat{G} = (\mathcal{V} \cup \mathcal{V}_s, \mathcal{E} \cup \mathcal{E}_s \cup \mathcal{E}_{ns}, \hat{X})$, Learning rate η, hyperparameter γ;
$\mathbf{Z}^1 \leftarrow 0$;
for i=1,...,T **do**
  $\hat{\mathbf{Z}}_0^i \leftarrow \mathbf{Z}^i$
  for j=1,...,K **do**
    $\hat{\mathbf{Z}}_j^i \leftarrow f(\hat{\mathbf{Z}}_{j-1}^i, \hat{G};);$
  end for
  $\nabla g := \nabla g(\boldsymbol{x}^i, \mathbf{Z}^i) - \bar{\nabla} g_k(\boldsymbol{x}^i, \hat{\mathbf{Z}}_K^i)$
  $\nabla F_\gamma := \nabla F(\boldsymbol{x}^i; \mathbf{Z}^i) + \gamma \nabla g$
  $(\boldsymbol{x}^{i+1}, \mathbf{Z}^{i+1}) \leftarrow \mathrm{Proj}\left((\boldsymbol{x}^i, \mathbf{Z}^i) - \eta \nabla F_\gamma\right)$
end for
Return: $(\boldsymbol{x}^T, \mathbf{Z}^T)$.

The algorithm has smaller gradient oracle calls and provable convergence guarantee.

The University of Iowa

Outline

- Background & Challenges
- Problem Formulation
- **Experiments**

Setup

- Data

Dataset	#Nodes	#Edges	#Subgraphs	#Labels/Classes
PPI-BP	17,080	316,591	1,591	6
HPO-METAB	14,587	3,238,174	2,400	6
HPO-NEURO	14,587	3,238,174	4,000	10
EM-USER	57,333	4,573,417	324	2

- Tasks
  - Subgraph classification

- Evaluation
  - *AUROC*
  - *Micro-F1*

Result in Micro-F1

Method	PPI-BP	HPO-METAB	HPO-NEURO	EM-USER
MLP	0.297±0.027	0.443±0.063	0.490±0.059	0.808±0.138
GCN-plain	0.398±0.058	0.452±0.025	0.535±0.032	0.561±0.021
Sub2Vec	0.309±0.023	0.114±0.021	0.206±0.073	0.522±0.043
GLASS	0.618±0.006	0.598±0.014	0.675±0.007	0.884±0.008
SubGNN	0.598±0.032	0.531±0.015	0.644±0.009	0.815±0.054
SSNP	0.636±0.007	0.587±0.010	0.682±0.004	0.888±0.005
IGNN-plain	0.389±0.025	0.284±0.021	0.215±0.002	0.579±0.008
EIGNN-plain	0.425±0.050	0.252±0.009	0.312±0.017	0.591±0.006
SoftIGNN	0.594±0.006	0.520±0.002	0.653±0.005	0.820±0.008
SoftEIGNN	0.592±0.006	0.522±0.002	0.658±0.004	0.829±0.010
ISNN	**0.731**±**0.026**	**0.646**±**0.014**	**0.688**±**0.004**	**0.914**±**0.009**

10% higher than the second best

Our method outperforms other baselines

Result in AUROC

Method	PPI-BP	HPO-METAB	HPO-NEURO	EM-USER
MLP	0.498±0.009	0.814±0.032	0.764±0.104	0.896±0.143
GCN-plain	0.663±0.044	0.772±0.018	0.773±0.027	0.525±0.065
Sub2Vec	0.544±0.011	0.496±0.010	0.504±0.015	0.518±0.048
GLASS	0.835±0.002	0.891±0.002	0.852±0.001	**0.960±0.004**
SubGNN	0.816±0.012	0.862±0.005	0.843±0.014	0.911±0.042
SSNP	0.831±0.008	0.883±0.007	0.867±0.004	0.952±0.011
IGNN-plain	0.514±0.046	0.496±0.063	0.709±0.065	0.541±0.089
EIGNN-plain	0.630±0.189	0.579±0.092	0.601±0.121	0.553±0.072
SoftIGNN	0.797±0.005	0.818±0.001	0.868±0.004	0.932±0.005
SoftEIGNN	0.798±0.008	0.821±0.001	0.868±0.002	0.927±0.006
ISNN	**0.924±0.012**	**0.919±0.002**	**0.896±0.002**	0.959±0.005

Our method outperforms other baselines expect on EM-USER.

Ablation Study: Hybrid Graph

HPO-METAB	ISNN-P	ISNN-S	IGNN-N	ISNN-rand
F1	$0.586_{\pm 0.021}$	$0.598_{\pm 0.021}$	$0.595_{\pm 0.021}$	$0.589_{\pm 0.028}$
AUROC	$0.874_{\pm 0.007}$	$0.874_{\pm 0.010}$	$0.876_{\pm 0.007}$	$0.876_{\pm 0.008}$

Construct the hybrid graph using hand-crafted channels

Construct the hybrid graph by adding subgraph-level edges randomly

Hand-crafted subgraph channels can be as bad as random.

More results in the paper

Efficiency

SOTA

Using implicit models on base graph

Method	PPI-BP	HPO-NEURO	HPO-METAB	EM-USER
SSNP	130.47±4.120	204.15±25.78	162.34±19.45	158.29±28.33
IGNN-plain	439.29±58.74	1629.86±89.14	1142.88±97.42	1386.28±85.90
EIGNN-plain	114.35±0.237	275.48±1.489	185.82±0.775	176.99±5.405
ISNN	**104.66**±28.14	**128.26**±4.571	**160.83**±19.37	**135.29**±35.70

Our method is efficient.

The University of Iowa

THANKS