# All in One: Multi-Task Prompting for Graph Neural Networks

Best Paper 2023 KDD

Yuting Hu 11/02/2023

University at Buffalo
Graduate School of Education

# Background

Graph tasks (node level, edge level, and graph level) are highly diverse.

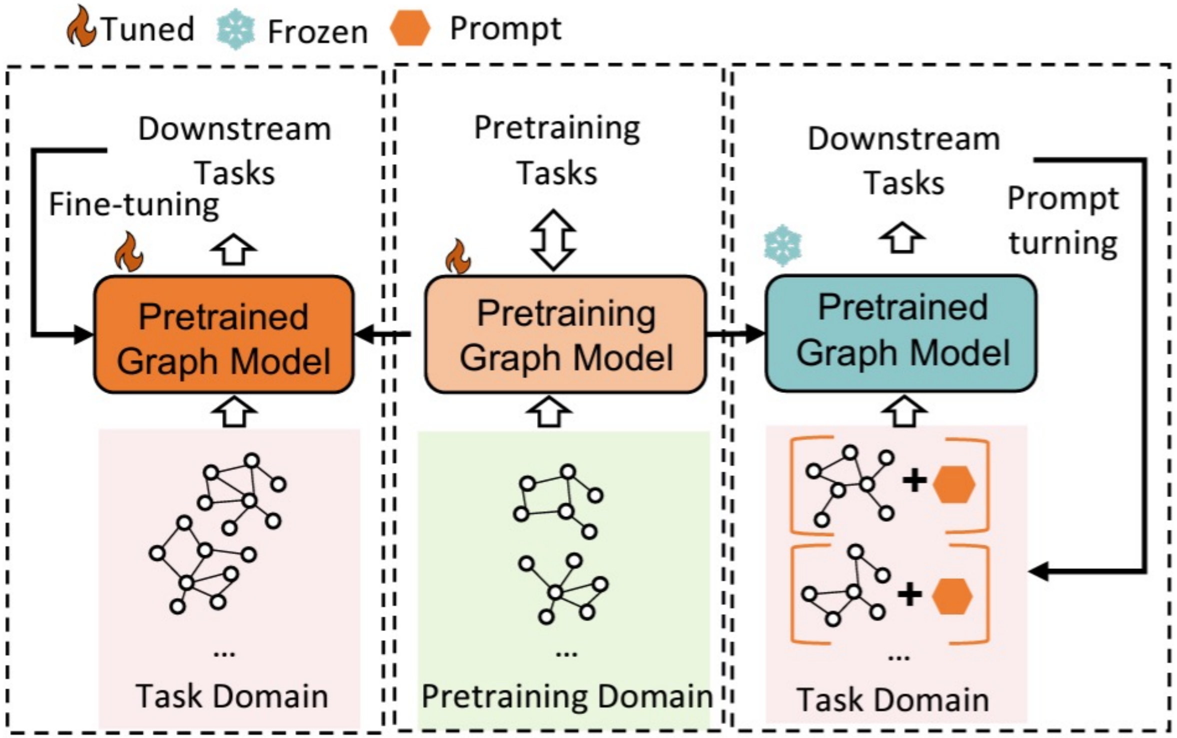
Pre-training and fine-tuning refers to pretrain a graph model with easily accessible data and transfer the graph knowledge to a new domain or task via tuning the last layer of the pre-trained model.

How to fill the gap between pre-trained models and different downstream graph tasks?

# Motivation

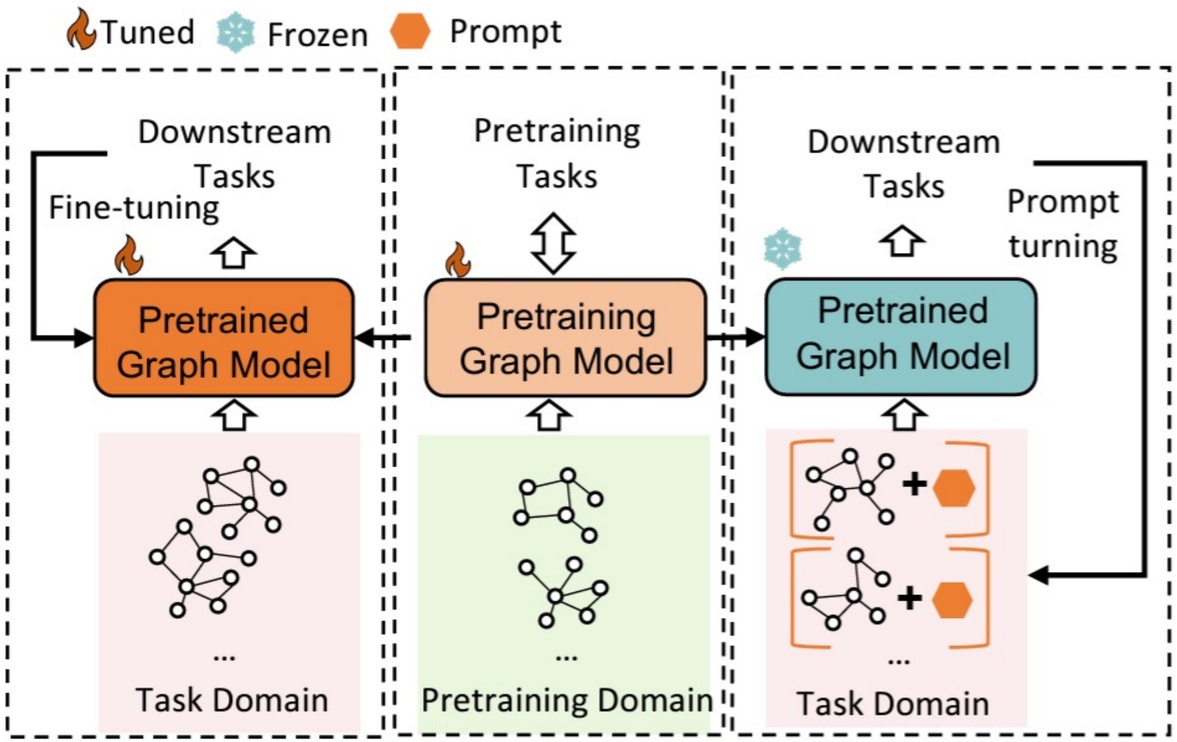
(Review) Prompt Learning Idea from NLP: generalizing pre-trained language models to a wide range of language applications.

How about introducing prompt idea to graph tasks?



# Main Idea

## Multi-Task Prompting for GNNs



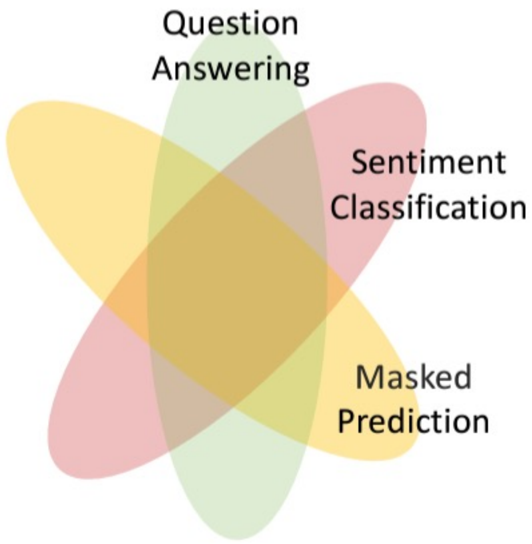

**Prompting:**
Use prompt to reformulate downstream tasks in line with the pre-training task!

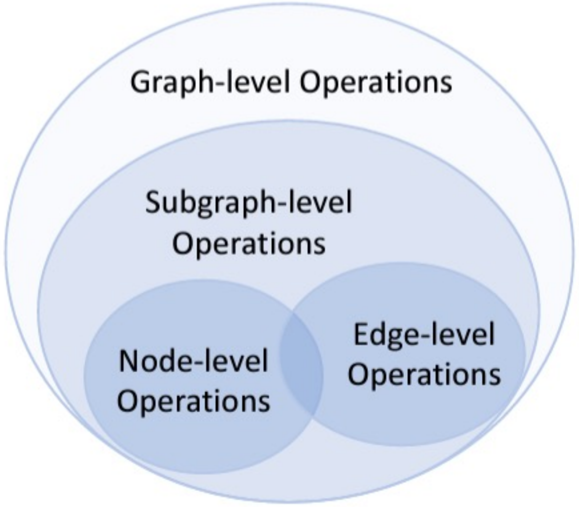
# Difficulties

## Thinking about prompting in NLP ...

**Language prompts:** preset phrases or learnable vectors, we only need to figure out prompt content.
**Graph prompts:** prompt content, how to organize prompt tokens, how to insert prompt into graphs.

**Pretrained LLMs:** based on masked prediction which share a large overlapping task sub-space, making it easily transferred.
**Graph learning:** hard to decide on an appropriate pre-training task to improve the capability of model generalization.



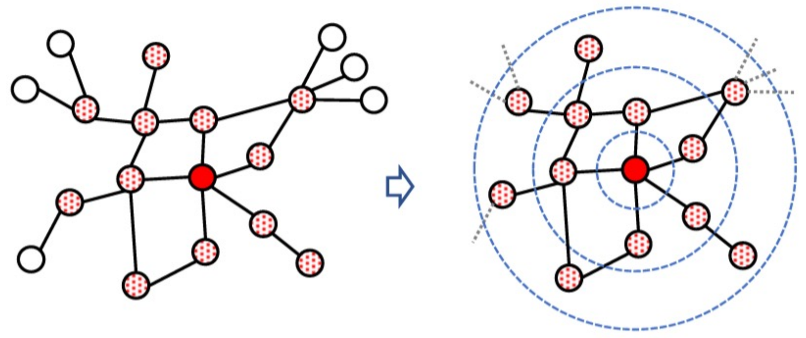



# Solution (1) : Reformulating Downstream Tasks

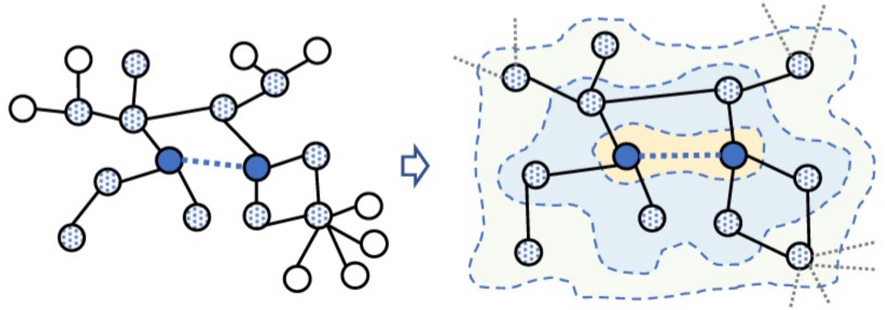
## Narrow down the gap by reformulating various graph tasks as graph-level tasks.

Node Level
Edge Level
→ Graph Level

Via building induced graphs ($\tau$ distance neighbors) for nodes and edges.

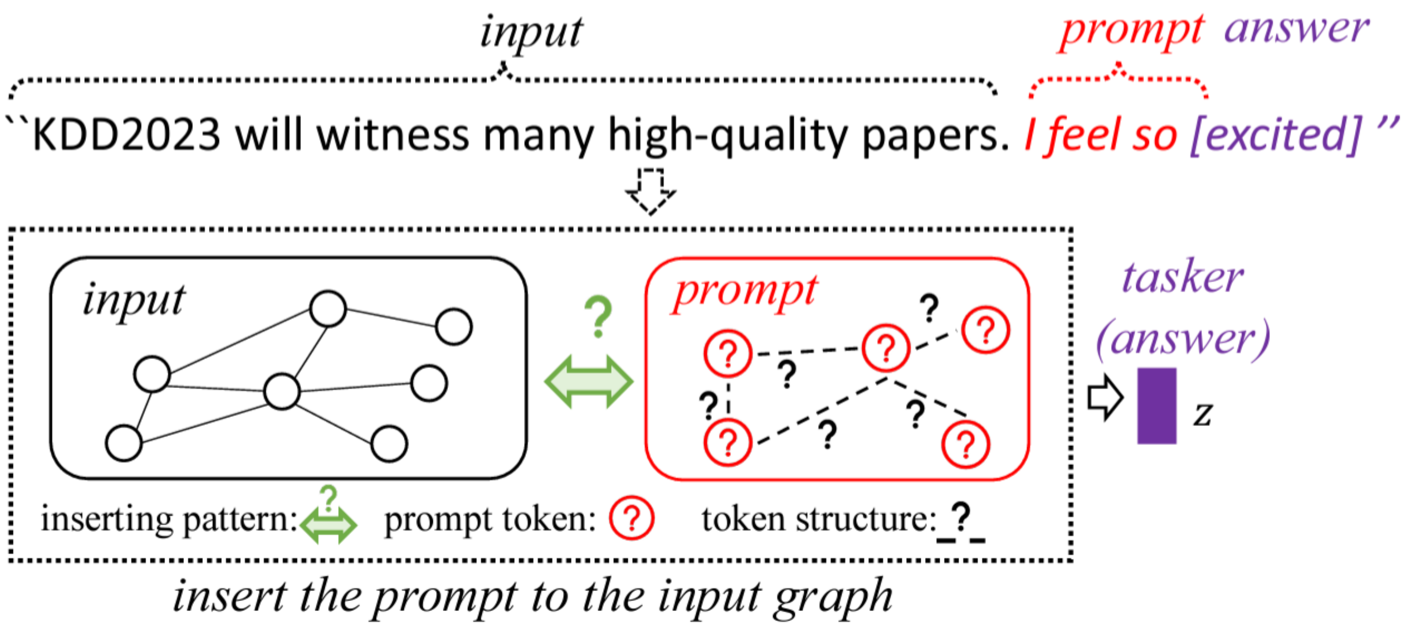


Node Label → Induced Graph Label



Edge Label → Induced Graph Label

# Solution (2) : Graph Prompts




$$\mathcal{G}_p = (\mathcal{P}, \mathcal{S})$$

## Element 1. prompt tokens $\mathcal{P} = \{p_1, p_2, \cdots, p_{|\mathcal{P}|}\}$

# Solution (2) : Graph Prompts

Element 2. token structure $\mathcal{S} = \{(p_i, p_j) | p_i, p_j \in \mathcal{P}\}$

Methods:

- learn tunable parameters. $\mathcal{A} = \bigcup_{\substack{i=1 \\ j=i+1}}^{|\mathcal{P}|-1} \{a_{ij}\}$
- use the dot product of each prompt token pair and prune them according to the dot value. $(p_i, p_j) \in \mathcal{S}$ iff $\sigma(\mathbf{p}_i \cdot \mathbf{p}_j) < \delta$
- treat the tokens as independent. $\mathcal{S} = \emptyset$.

# Solution (2) : Graph Prompts

## Element 3. inserting pattern $\mathcal{G}_m = \psi(\mathcal{G}, \mathcal{G}_p)$

the manipulated graph can be denoted as $\mathcal{G}_m = \psi(\mathcal{G}, \mathcal{G}_p)$. We can define the inserting pattern as the dot product between prompt tokens and input graph nodes, and then use a tailored connection like $\hat{\mathbf{x}}_i = \mathbf{x}_i + \sum_{k=1}^{|\mathcal{P}|} w_{ik}\mathbf{p}_k$ where $w_{ik}$ is a weighted value to prune unnecessary connections:

$$w_{ik} = \begin{cases} \sigma(\mathbf{p}_k \cdot \mathbf{x}_i^T), & \text{if } \sigma(\mathbf{p}_k \cdot \mathbf{x}_i^T) > \delta \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

As an alternative and special case, we can also use a more simplified way to get $\hat{\mathbf{x}}_i = \mathbf{x}_i + \sum_{k=1}^{|\mathcal{P}|} \mathbf{p}_k$.

# Solution (3) : Applying Meta learning to Graph Prompting.

## Task Level

Prompt Params $$\theta_i^k = \theta_i^{k-1} - \alpha \nabla_{\theta_i^{k-1}} \mathcal{L}_{\mathcal{D}^s_{\tau_i}} \left( f_{\theta_i^{k-1}, \phi_i^{k-1} | \pi^*} \right)$$

Task Params $$\phi_i^k = \phi_i^{k-1} - \alpha \nabla_{\phi_i^{k-1}} \mathcal{L}_{\mathcal{D}^s_{\tau_i}} \left( f_{\theta_i^{k-1}, \phi_i^{k-1} | \pi^*} \right)$$

## Meta Level

$$\theta^*, \phi^* = \arg\min_{\theta, \phi} \sum_{\tau_i \in \mathcal{T}} \mathcal{L}_{\mathcal{D}^q_{\tau_i}} \left( f_{\theta_i, \phi_i | \pi^*} \right)$$

**Algorithm 1:** Overall Learning Process

**Input:** Overall pipeline $f_{\theta,\phi|\pi^*}$ with prompt parameter $\theta$, pre-trained model with frozen parameter $\pi^*$, and task head parameterized by $\phi$; Multi-task episodes $\mathcal{E} = \{\mathcal{E}_1, \cdots, \mathcal{E}_n\}$;

**Output:** Optimal pipeline $f_{\theta^*,\phi^*|\pi^*}$

```
Initialize θ and ϕ
while not done do
    // inner adaptation
    Sample E_i ∈ E where E_i = (T_{E_i}, L_{E_i}, S_{E_i}, Q_{E_i})
    for τ_◁t ∈ T_{E_i}, ◁ = g, n, ℓ do
        θ_{τ_◁t}, ϕ_{τ_◁t} ← θ, ϕ
        θ_{τ_◁t} ← θ_{τ_◁t} − α∇_{θ_{τ_◁t}} L^{(◁)}_{D^s_{τ_◁t}} (f_{θ_{τ_◁t}, ϕ_{τ_◁t} | π*})
        ϕ_{τ_◁t} ← ϕ_{τ_◁t} − α∇_{ϕ_{τ_◁t}} L^{(◁)}_{D^s_{τ_◁t}} (f_{θ_{τ_◁t}, ϕ_{τ_◁t} | π*})
    end
    // outer meta update
    Update θ, ϕ by Equation (4) on
     Q_{E_i} = {D^q_{τ_◁t} | τ_◁t ∈ T_{E_i}, ◁ = g, n, ℓ}
end
return f_{θ*, ϕ* | π*}
```

# Results (1): Multi-Task Performance with Few-shot Learning Settings

**Table 2: Node-level performance (%) with 100-shot setting. IMP (%): the average improvement of prompt over the rest.**

| Training schemes | Methods | Cora | | | CiteSeer | | | Reddit | | | Amazon | | | Pubmed | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Acc | F1 | AUC | Acc | F1 | AUC | Acc | F1 | AUC | Acc | F1 | AUC | Acc | F1 | AUC |
| supervised | GAT | 74.45 | 73.21 | 82.97 | 83.00 | 83.20 | 89.33 | 55.64 | 62.03 | 65.38 | 79.00 | 73.42 | 97.81 | 75.00 | 77.56 | 79.72 |
| | GCN | 77.55 | 77.45 | 83.71 | 88.00 | 81.79 | 94.79 | 54.38 | 52.47 | 56.82 | 95.36 | 93.99 | 96.23 | 53.64 | 66.67 | 69.89 |
| | GT | 74.25 | 75.21 | 82.04 | 86.33 | 85.62 | 90.13 | 61.50 | 61.38 | 65.56 | 85.50 | 86.01 | 93.01 | 51.50 | 67.34 | 71.91 |
| pre-train + fine-tune | GraphCL+GAT | 76.05 | 76.78 | 81.96 | 87.64 | 88.40 | 89.93 | 57.37 | 66.42 | 67.43 | 78.67 | 72.26 | 95.65 | 76.03 | 77.05 | 80.02 |
| | GraphCL+GCN | 78.75 | 79.13 | 84.90 | 87.49 | 89.36 | 90.25 | 55.00 | 65.52 | 74.65 | 96.00 | 95.92 | 98.33 | 69.37 | 70.00 | 74.74 |
| | GraphCL+GT | 73.80 | 74.12 | 82.77 | 88.50 | 88.92 | 91.25 | 63.50 | 66.06 | 68.04 | 94.39 | 93.62 | 96.97 | 75.00 | 78.45 | 75.05 |
| | SimGRACE+GAT | 76.85 | 77.48 | 83.37 | 90.50 | 91.00 | 91.56 | 56.59 | 65.47 | 67.77 | 84.50 | 84.73 | 89.69 | 72.50 | 68.21 | 81.97 |
| | SimGRACE+GCN | 77.20 | 76.39 | 83.13 | 83.50 | 84.21 | 93.22 | 58.00 | 55.81 | 56.93 | 95.00 | 94.50 | 98.03 | 77.50 | 75.71 | 87.53 |
| | SimGRACE+GT | 77.40 | 78.11 | 82.95 | 87.50 | 87.05 | 91.85 | 66.00 | 69.95 | 70.03 | 79.00 | 73.42 | 97.58 | 70.50 | 73.30 | 74.22 |
| prompt | GraphCL+GAT | 76.50 | 77.26 | 82.99 | 88.00 | 90.52 | 91.82 | 57.84 | 67.02 | 75.33 | 80.01 | 75.62 | 97.96 | 77.50 | 78.26 | 83.02 |
| | GraphCL+GCN | 79.20 | 79.62 | 85.29 | 88.50 | 91.59 | 91.43 | 56.00 | 68.57 | 78.82 | 96.50 | 96.37 | 98.70 | 72.50 | 72.64 | 79.57 |
| | GraphCL+GT | 75.00 | 76.00 | 83.36 | 91.00 | 91.00 | 93.29 | 65.50 | 66.08 | 68.86 | 95.50 | 95.43 | 97.56 | 76.50 | 79.11 | 76.00 |
| | SimGRACE+GAT | 76.95 | 78.51 | 83.55 | 93.00 | 93.14 | 92.44 | 57.63 | 66.64 | 69.43 | 95.50 | 95.43 | 97.56 | 73.00 | 74.04 | 81.89 |
| | SimGRACE+GCN | 77.85 | 76.57 | 83.79 | 90.00 | 89.47 | 94.87 | 59.50 | 55.97 | 59.46 | 95.00 | 95.24 | 98.42 | 78.00 | 78.22 | 87.66 |
| | SimGRACE+GT | 78.75 | 79.53 | 85.03 | 91.00 | 91.26 | 95.62 | 69.50 | 71.43 | 70.75 | 86.00 | 83.72 | 98.24 | 73.00 | 73.79 | 76.64 |
| IMP (%) | | 1.47 | 1.94 | 1.10 | 3.81 | 5.25 | 2.05 | 3.97 | 5.04 | 6.98 | 4.49 | 5.84 | 2.24 | 8.81 | 4.55 | 4.62 |
| Reported Acc of GPPT (Label Ratio 50%) | | 77.16 | – | – | 65.81 | – | – | 92.13 | – | – | 86.80 | – | – | 72.23 | – | – |
| appr. Label Ratio of our 100-shot setting | | ~ 25% | | | ~ 18% | | | ~ 1.7% | | | ~ 7.3% | | | ~ 1.5% | | |

# Results (1): Multi-Task Performance with Few-shot Learning Settings

Table 12: Edge-level performance (%) with 100-shot setting. IMP (%): the average improvement of prompt over the rest.

| Training schemes | Methods | Cora | | | CiteSeer | | | Reddit | | | Amazon | | | Pubmed | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Acc | F1 | AUC | Acc | F1 | AUC | Acc | F1 | AUC | Acc | F1 | AUC | Acc | F1 | AUC |
| supervised | GAT | 84.30 | 83.35 | 85.43 | 68.63 | 82.79 | 89.98 | 93.50 | 93.03 | 94.48 | 85.00 | 82.67 | 88.78 | 80.05 | 77.07 | 79.26 |
| | GCN | 83.85 | 84.90 | 85.90 | 66.67 | 81.01 | 89.62 | 83.50 | 84.51 | 91.43 | 89.00 | 89.81 | 98.85 | 79.00 | 77.73 | 80.19 |
| | GT | 85.95 | 86.01 | 87.25 | 69.70 | 83.03 | 82.46 | 95.50 | 94.52 | 96.89 | 94.00 | 93.62 | 99.34 | 74.50 | 65.77 | 85.19 |
| pre-train + fine-tune | GraphCL+GAT | 85.64 | 85.97 | 87.22 | 72.67 | 82.85 | 92.98 | 94.00 | 93.75 | 98.43 | 86.50 | 86.96 | 84.47 | 85.54 | 83.92 | 91.78 |
| | GraphCL+GCN | 86.36 | 85.82 | 86.39 | 70.67 | 81.82 | 90.00 | 94.00 | 93.94 | 97.04 | 86.50 | 84.92 | 98.41 | 80.00 | 78.05 | 85.21 |
| | GraphCL+GT | 85.79 | 86.27 | 87.51 | 86.01 | 85.38 | 88.58 | 96.67 | 95.38 | 97.65 | 96.50 | 97.42 | 98.12 | 85.50 | 87.11 | 81.68 |
| | SimGRACE+GAT | 86.85 | 86.80 | 88.12 | 85.33 | 85.26 | 90.04 | 95.50 | 95.54 | 97.11 | 87.50 | 86.34 | 88.65 | 80.01 | 81.03 | 86.89 |
| | SimGRACE+GCN | 85.62 | 85.38 | 87.83 | 89.33 | 86.34 | 95.10 | 88.00 | 87.88 | 94.49 | 98.45 | 97.57 | 98.29 | 80.50 | 82.58 | 91.22 |
| | SimGRACE+GT | 86.35 | 87.03 | 88.47 | 86.00 | 89.52 | 90.42 | 97.50 | 95.54 | 96.92 | 96.50 | 96.45 | 99.09 | 81.00 | 79.57 | 85.69 |
| prompt | GraphCL+GAT | 86.85 | 86.88 | 87.92 | 76.67 | 83.00 | 96.22 | 95.36 | 94.50 | 98.65 | 88.50 | 86.00 | 87.15 | 86.50 | 84.75 | 92.61 |
| | GraphCL+GCN | 86.87 | 86.80 | 87.79 | 76.67 | 82.37 | 93.54 | 95.50 | 95.52 | 97.75 | 86.96 | 85.63 | 98.66 | 81.50 | 78.61 | 86.11 |
| | GraphCL+GT | 87.02 | 86.90 | 87.97 | 86.67 | 88.00 | 91.10 | 97.03 | 95.94 | 98.62 | 98.50 | 98.48 | 98.53 | 86.50 | 87.78 | 82.21 |
| | SimGRACE+GAT | 87.37 | 87.33 | 88.37 | 91.33 | 92.30 | 95.18 | 95.72 | 96.69 | 97.64 | 95.50 | 95.38 | 98.89 | 80.50 | 82.03 | 87.86 |
| | SimGRACE+GCN | 86.85 | 86.80 | 88.67 | 93.47 | 97.69 | 97.08 | 88.00 | 88.12 | 95.10 | 98.50 | 98.52 | 98.55 | 81.00 | 83.76 | 91.41 |
| | SimGRACE+GT | 87.30 | 87.24 | 88.74 | 95.33 | 96.52 | 94.46 | 98.00 | 98.02 | 99.38 | 98.50 | 98.52 | 99.10 | 82.50 | 80.45 | 87.61 |
| | IMP(%) | 1.65 | 1.48 | 1.28 | 12.26 | 6.84 | 5.21 | 1.94 | 2.29 | 1.88 | 3.63 | 3.44 | 2.03 | 2.98 | 4.66 | 3.21 |

# Results (1): Multi-Task Performance with Few-shot Learning Settings

**Table 13: Graph-level performance (%) with 100-shot setting. IMP (%): the average improvement of prompt over the rest.**

| Training schemes | Methods | Cora Acc | Cora F1 | Cora AUC | CiteSeer Acc | CiteSeer F1 | CiteSeer AUC | Reddit Acc | Reddit F1 | Reddit AUC | Amazon Acc | Amazon F1 | Amazon AUC | Pubmed Acc | Pubmed F1 | Pubmed AUC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| supervised | GAT | 84.40 | 86.44 | 87.60 | 86.50 | 84.75 | 91.75 | 79.50 | 79.76 | 82.11 | 93.05 | 94.04 | 93.95 | 69.86 | 72.30 | 66.92 |
| | GCN | 83.95 | 86.01 | 88.64 | 85.00 | 82.56 | 93.33 | 64.00 | 70.00 | 78.60 | 91.20 | 91.27 | 94.33 | 61.30 | 59.97 | 66.29 |
| | GT | 85.85 | 85.90 | 89.59 | 77.50 | 75.85 | 89.72 | 69.62 | 68.01 | 66.32 | 90.33 | 91.39 | 94.39 | 60.30 | 60.88 | 67.62 |
| pre-train + fine-tune | GraphCL+GAT | 85.50 | 85.54 | 89.31 | 83.00 | 85.47 | 92.13 | 72.03 | 72.82 | 83.23 | 92.15 | 92.18 | 94.78 | 85.50 | 85.50 | 86.33 |
| | GraphCL+GCN | 85.50 | 85.59 | 87.94 | 86.50 | 84.57 | 94.56 | 71.00 | 71.90 | 80.33 | 93.58 | 93.55 | 94.93 | 78.75 | 77.29 | 89.40 |
| | GraphCL+GT | 85.95 | 85.05 | 87.92 | 84.50 | 81.87 | 88.36 | 69.63 | 70.06 | 81.35 | 91.68 | 91.55 | 94.78 | 86.85 | 86.93 | 88.91 |
| | SimGRACE+GAT | 86.04 | 86.33 | 88.55 | 83.50 | 85.84 | 90.09 | 81.32 | 81.64 | 88.61 | 93.58 | 93.57 | 93.91 | 87.33 | 86.70 | 88.02 |
| | SimGRACE+GCN | 85.95 | 86.05 | 89.33 | 84.50 | 86.46 | 91.60 | 80.50 | 81.52 | 89.11 | 90.73 | 90.52 | 94.85 | 85.26 | 84.64 | 86.99 |
| | SimGRACE+GT | 86.40 | 86.47 | 89.64 | 81.00 | 81.54 | 89.81 | 69.50 | 70.97 | 77.11 | 92.63 | 92.56 | 94.04 | 85.95 | 86.05 | 89.37 |
| prompt | GraphCL+GAT | 86.40 | 86.47 | 89.46 | 86.50 | 89.93 | 92.24 | 73.36 | 73.32 | 84.77 | 94.08 | 94.02 | 94.20 | 85.95 | 85.97 | 87.17 |
| | GraphCL+GCN | 85.95 | 86.01 | 88.95 | 87.00 | 85.87 | 95.35 | 72.50 | 72.91 | 81.37 | 94.05 | 94.05 | 94.98 | 84.60 | 84.43 | 88.96 |
| | GraphCL+GT | 86.05 | 85.17 | 88.93 | 85.50 | 85.28 | 88.60 | 72.63 | 70.97 | 82.39 | 92.63 | 92.64 | 94.82 | 87.03 | 86.96 | 89.10 |
| | SimGRACE+GAT | 86.67 | 86.36 | 89.51 | 87.50 | 88.37 | 91.47 | 82.62 | 83.33 | 89.41 | 93.35 | 94.66 | 94.61 | 87.75 | 87.69 | 88.88 |
| | SimGRACE+GCN | 86.85 | 86.90 | 89.95 | 85.00 | 85.85 | 91.95 | 81.00 | 82.24 | 89.43 | 93.95 | 92.06 | 93.89 | 85.50 | 85.54 | 87.30 |
| | SimGRACE+GT | 86.85 | 86.87 | 89.75 | 87.50 | 86.63 | 90.85 | 76.50 | 80.82 | 86.84 | 94.05 | 94.06 | 94.96 | 86.40 | 86.50 | 89.74 |
| | IMP(%) | 1.12 | 0.43 | 0.79 | 3.52 | 4.54 | 0.53 | 4.69 | 4.31 | 6.13 | 1.72 | 1.39 | 0.14 | 10.66 | 10.77 | 9.16 |

# Results (2): Transferability Evaluation

**Table 3: Transferability (%) on Amazon from different level tasks spaces. Source tasks: graph-level tasks and node-level tasks. Target task: edge-level tasks.**

| Source task | Methods | Accuracy | F1-score | AUC score |
|---|---|---|---|---|
| graph level | hard | 51.50 | 65.96 | 40.34 |
| | fine-tune | 62.50 | 70.59 | 53.91 |
| | prompt | 70.50 | 71.22 | 74.02 |
| node level | hard | 40.50 | 11.85 | 29.48 |
| | fine-tune | 46.00 | 54.24 | 37.26 |
| | prompt | 59.50 | 68.73 | 55.90 |

**Table 3: Transferability (%) on Amazon from different level tasks spaces. Source tasks: graph-level tasks and node-level tasks. Target task: edge-level tasks.**

| Source task | Methods | Accuracy | F1-score | AUC score |
|---|---|---|---|---|
| graph level | hard | 51.50 | 65.96 | 40.34 |
| | fine-tune | 62.50 | 70.59 | 53.91 |
| | prompt | 70.50 | 71.22 | 74.02 |
| node level | hard | 40.50 | 11.85 | 29.48 |
| | fine-tune | 46.00 | 54.24 | 37.26 |
| | prompt | 59.50 | 68.73 | 55.90 |

# Results (3): Ablation Study

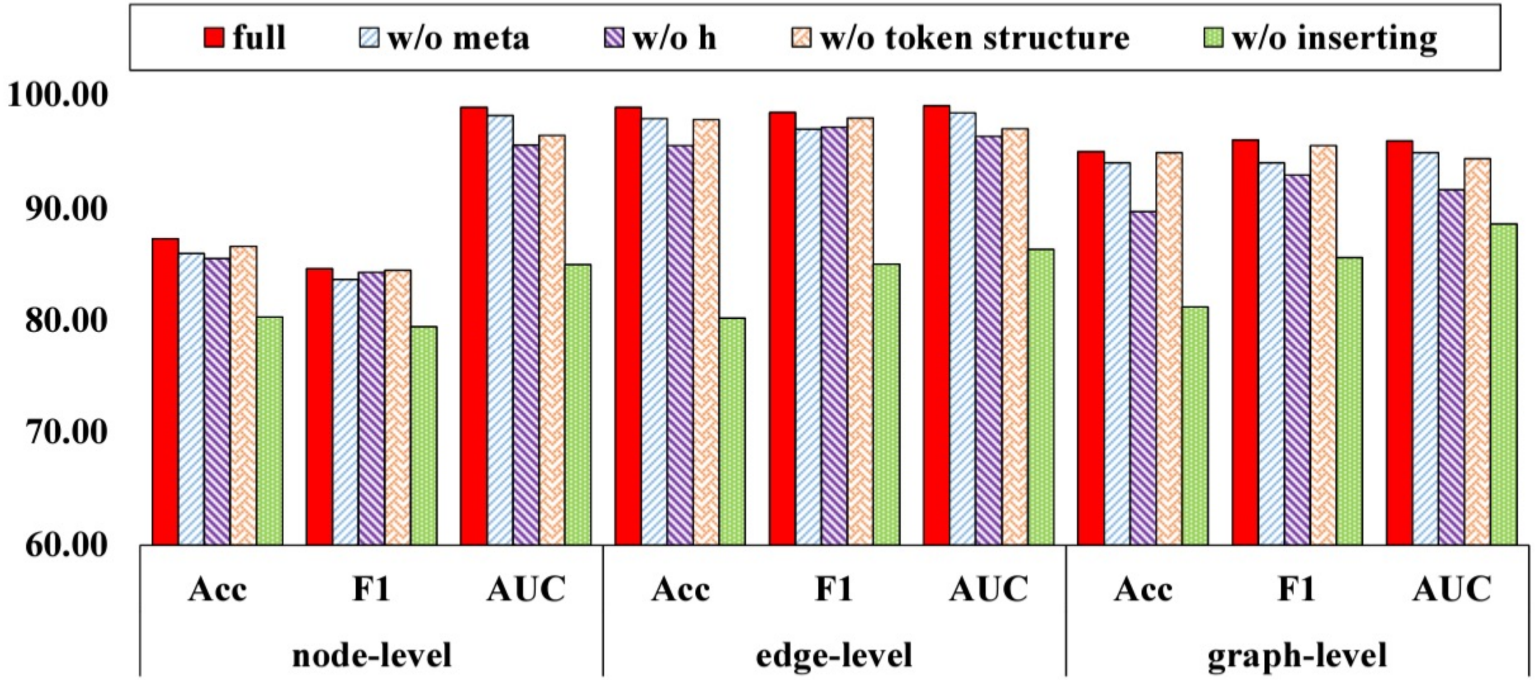



**Figure 5: Effectiveness of main components**

# Novelties

- Proposes graph prompt learning concept and format.
- Unifies various graph tasks as graph-level tasks.
- Applies meta-learning to multi-task prompt learning.
- Valid and effective results.

# Thanks!
# Any Questions?

University at Buffalo
Graduate School of Education