GraphPrompt: Unifying Pre-Training and Downstream Tasks for Graph Neural Networks

Zemin Liu\textsuperscript{1*}, Xingtong Yu\textsuperscript{2*}, Yuan Fang\textsuperscript{3†}, Xinming Zhang\textsuperscript{2†}

\textsuperscript{1} National University of Singapore, Singapore
\textsuperscript{2} University of Science and Technology of China, China
\textsuperscript{3} Singapore Management University, Singapore

\textit{In Proceeding of THE WEB CONFERENCE, APRIL 30 - MAY 4, 2023}

* Co-first author
† Corresponding author
Outline

1. Motivation
2. Challenges
3. Proposed Model: GraphPrompt
4. Experiment
5. Conclusions
GNNs’ performance heavily depends on labeled data[1,2]

Scarce of labeled data

Pre-Training+Finetuning [3,4]

Gap between pre-train and downstream tasks[5]

Pre-Training+ Prompt

Motivation

Problem 1:

- task-specific labeled data is often difficult or costly to obtain

Problem 2:

- pre-training step aims to preserve various intrinsic graph properties
- fine-tuning step aims to reduce the downstream task loss

Outline

1. Motivation
2. Challenges
3. Proposed Model: GraphPrompt
4. Experiment
5. Conclusions
Challenges

• Different downstream tasks often have different objectives [6]
• Distinction between various downstream tasks

**C1:** How to unify pre-training with various downstream tasks on graph?

**C2:** How to design prompts on graphs? [7]


[7] Mingchen Sun, Kaixiong Zhou, Xin He, Ying Wang, and Xin Wang. 2022. GPPT: Graph Pre-training and Prompt Tuning to Generalize Graph Neural Networks. SIGKDD

Figure 1: Illustration of the motivation. (a) Pre-training on graphs. (b/c) Downstream node/graph classification.
Outline

1. Motivation
2. Challenges
3. Proposed Model: GraphPrompt
4. Experiment
5. Conclusions
Proposed Method: GraphPrompt

Unified task template

Link Prediction

\[ \text{sim}(s_v, s_a) > \text{sim}(s_v, s_b) \]

Node Classification (NC)

\[ \hat{s}_c = \frac{1}{k} \sum_{(v_i, l_i) \in D, l_i = c} s_{v_i} \]

\[ f_j = \arg \max_{c \in C} \text{sim}(s_{v_j}, \hat{s}_c) \]

Graph Classification (GC)

\[ \hat{s}_c = \frac{1}{k} \sum_{(G_i, l_i) \in D, l_i = c} s_{G_i} \]

\[ L_j = \arg \max_{c \in C} \text{sim}(s_{G_j}, \hat{s}_c) \]

A Notation for NC and GC

\[ y = \arg \max_{c \in Y} \text{sim}(s_X, \hat{s}_c) \]

\[ s_X = \text{READOUT}(\{h_v : v \in V(S_X)\}) \]

Pre-Training Objective

\[ L_{\text{pre}}(\Theta) = - \sum_{(v, a, b) \in \mathcal{T}_{\text{pre}}} \ln \frac{\exp(\text{sim}(s_v, s_a)/\tau)}{\sum_{u \in \{a, b\}} \exp(\text{sim}(s_v, s_u)/\tau)} \]

Prompt Design

\[ s_{t, X} = \text{READOUT}(\{p_t \odot h_u : u \in V(S_X)\}) \]

Figure 2: Overall framework of GraphPrompt.
Outline

1. Motivation
2. Challenges
3. Proposed Model: GraphPrompt
4. Experiment
5. Conclusions
GraphPrompt outperforms all baselines for both node classification task and graph classification task, which implies
- GraphPrompt is able to narrow the gap between pre-training task and downstream tasks.
- GraphPrompt could effectively derive the downstream tasks to exploit the pre-trained model in task-specific manner.
Experiment

- GraphPrompt consistently outperforms the baselines especially with lower shots
- For node classification task, 10 shot is sufficient for semi-supervised learning since graph is small
- For graph classification task, GraphPrompt can be surpassed by some baselines when given more shots
Outline

1. Motivation
2. Challenges
3. Proposed Model: GraphPrompt
4. Experiment
5. Conclusions
• **Problem: Pretraining-Prompting**
  • Unify pre-training task and downstream tasks
  • Attain task-specific optima

• **Proposed-Model: GraphPrompt**
  • Unify upstream and downstream tasks via subgraph similarity
  • Using prompt vector to change the feature weights of each dimension of the node embedding to guide subgraph readout

• **Experiment**
  • GraphPrompt outperforms all baselines for both node classification task and graph classification task
Thanks!

Paper, data & code available at https://xingtongyu.netlify.app/

Zemin Liu*, Xingtong Yu*, Yuan Fang†, Xinming Zhang†

GraphPrompt: Unifying Pre-Training and Downstream Tasks for Graph Neural Networks

*Proceeding of THE WEB CONFERENCE, APRIL 30 - MAY 4, 2023