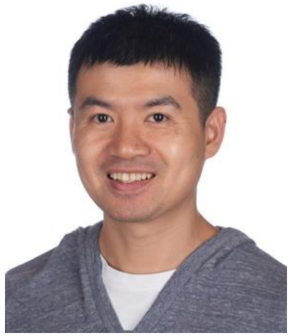
Few-Shot Learning on Graphs: From Meta-Learning to LLM-empowered Pre-Training and Beyond

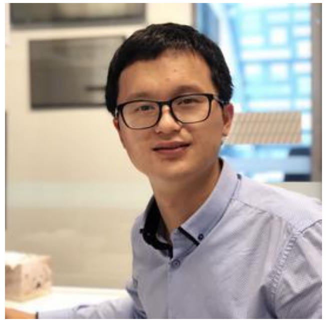
Lecture-style Tutorial

The Web Conference 2025


Yuan Fang
SMU SINGAPORE MANAGEMENT UNIVERSITY


Yuxia Wu
SMU SINGAPORE MANAGEMENT UNIVERSITY

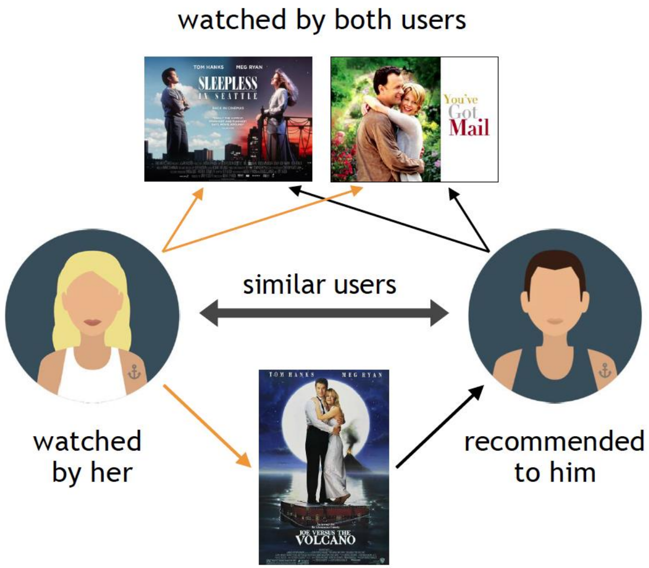

Xingtong Yu
SMU SINGAPORE MANAGEMENT UNIVERSITY


Shirui Pan
Griffith UNIVERSITY

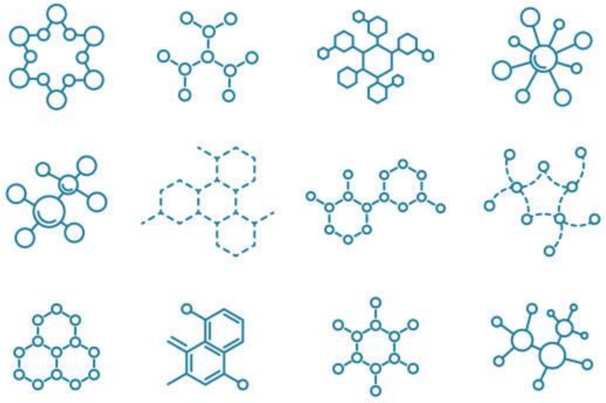
Graphs

- Graphs model the interactions among various objects

Social network



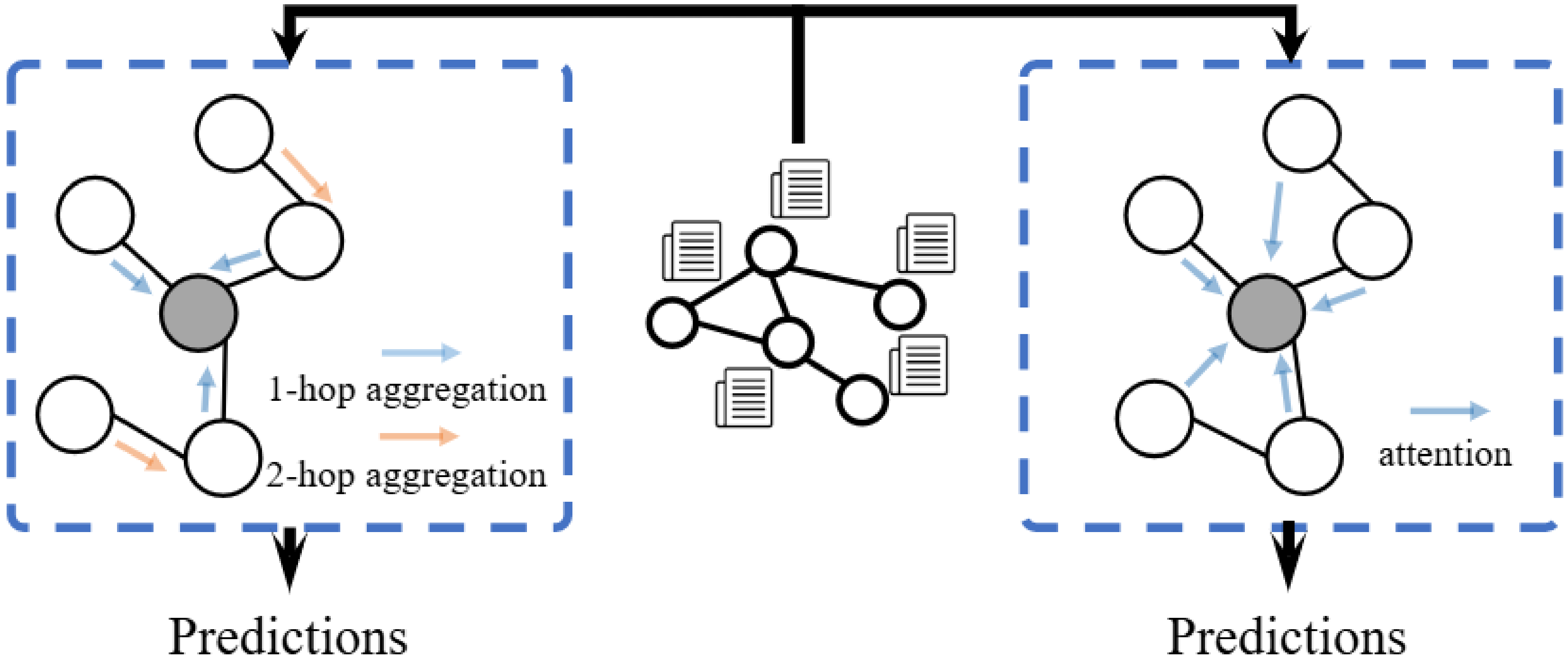
Recommendation System



Molecular graphs

End to End Graph Learning

- (Semi-)Supervised graph representation learning methods



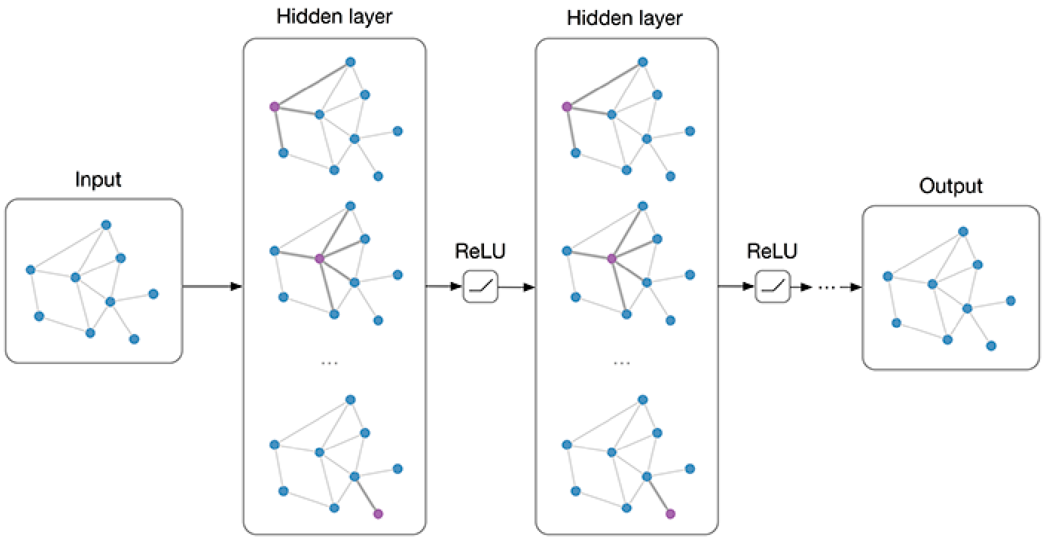
Graph Neural Networks (GNNs)

Graph Transformer

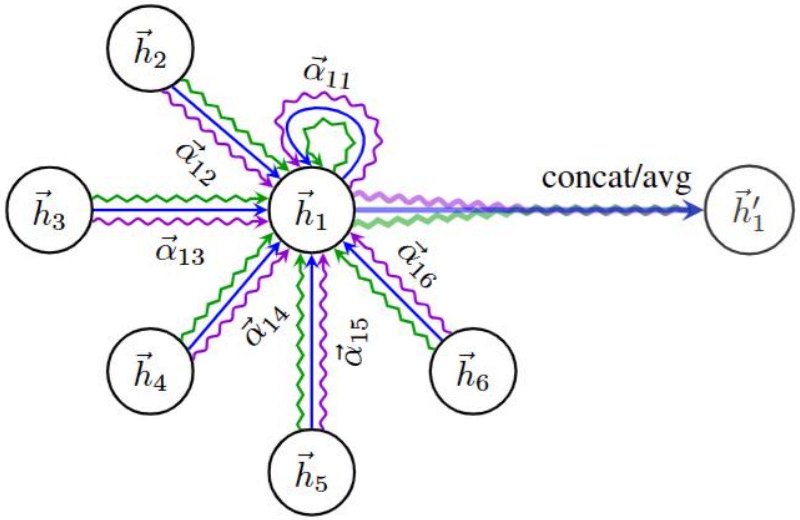
Liu, et al. Towards graph foundation models: A survey and beyond. TPAMI 2025.

Graph Neural Networks

- GNNs typically leverage message-passing framework



GCN



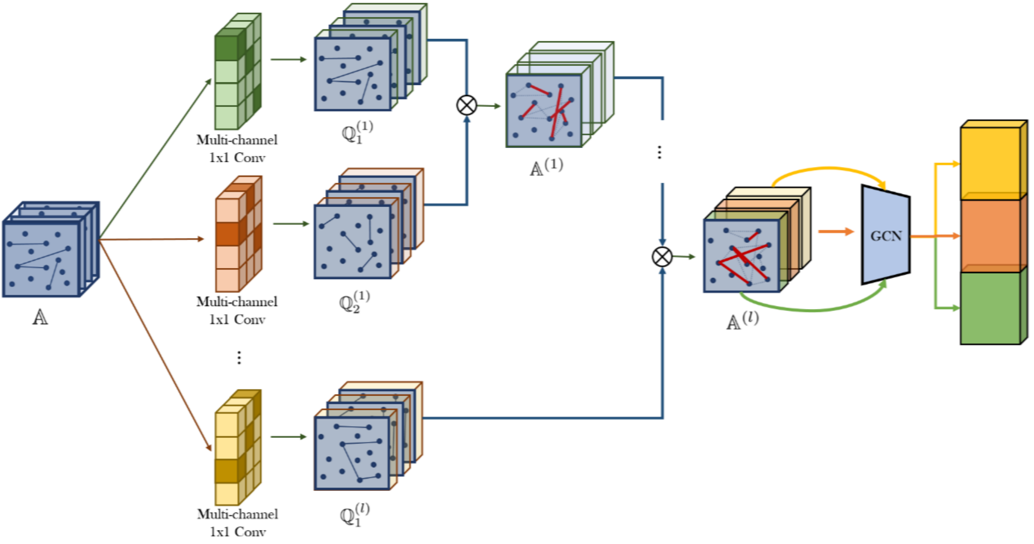
GAT

Kipf, et al. “Semi-supervised classification with graph convolutional networks.” ICLR’17.

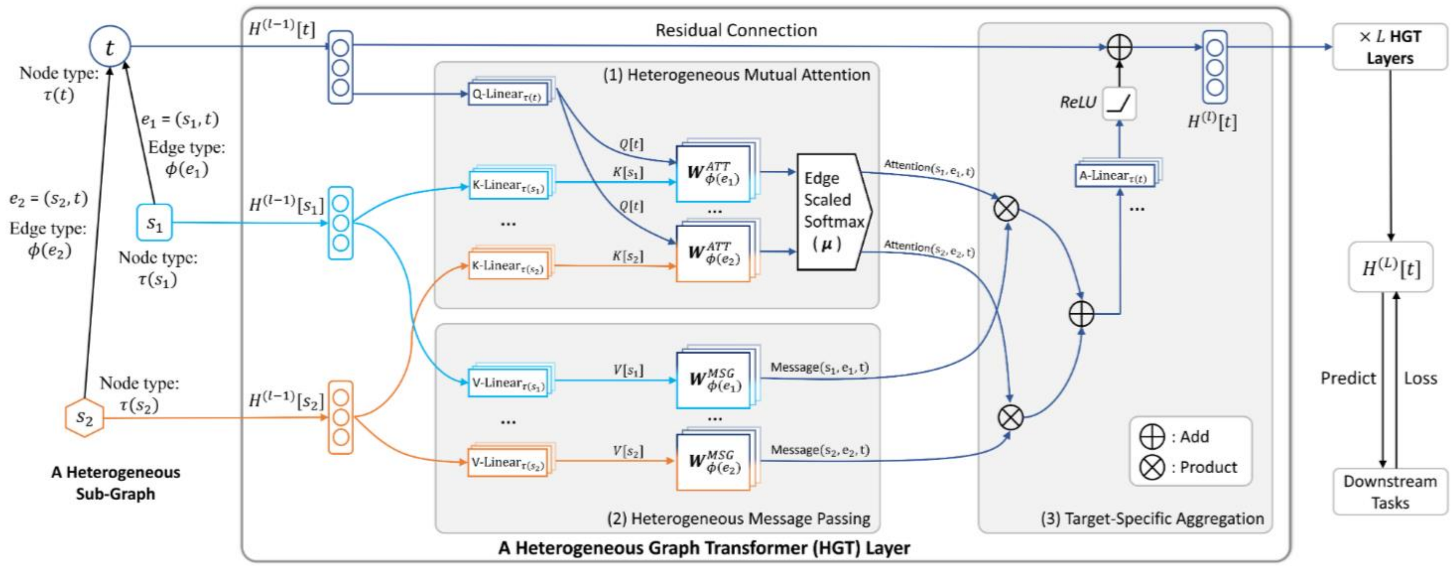
Velickovic, et al. “Graph attention networks”. ICLR’18.

Graph Transformers

- Transformers are widely used in graph learning



GTN



HGT

Yun, et al. “Graph transformer networks.” NeurIPS’19.

Hu, et al. “Heterogeneous graph transformer.” WWW’20.

Few-shot Learning Problems

- Performance highly depends on

➢ Abundant labeled data

➢ Challenging or expensive to obtain labels, leads to

Label scarcity

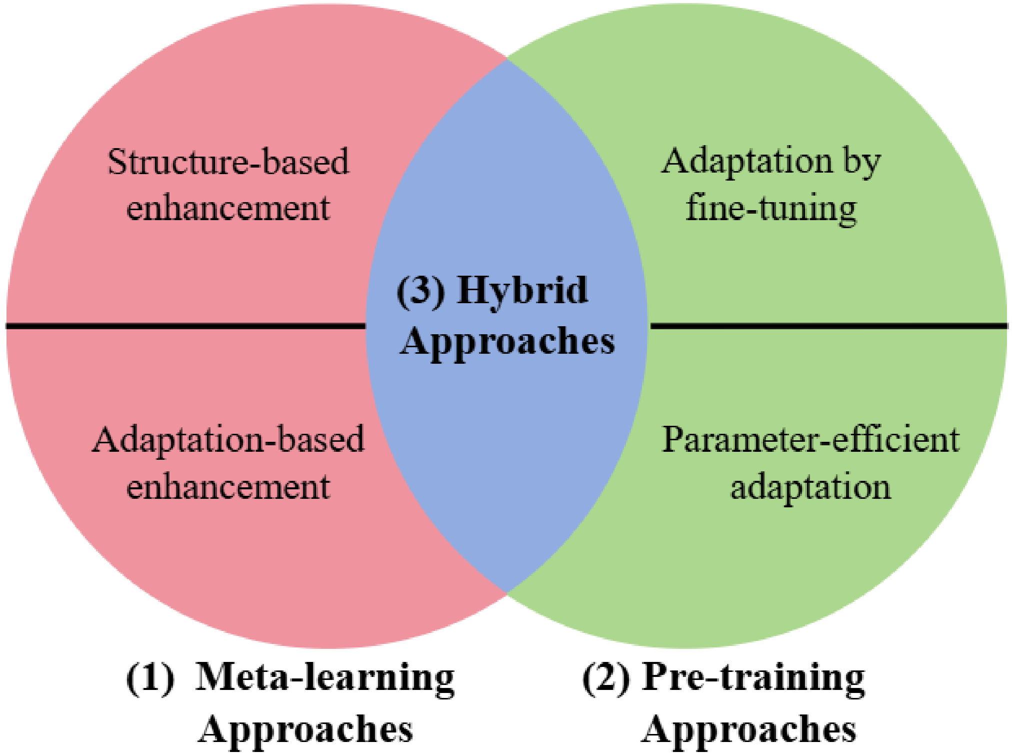
➢ Rich Structure

➢ Graph structure may be sparse, leads to

Structure scarcity

Few-shot Learning Methods

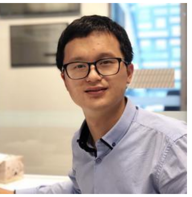
- Learn prior knowledge and adapt to downstream applications



Few-shot Learning Methods

- **Meta learning methods**
  - Derive prior knowledge from a series of “meta-training” task
- **Pre-training methods**
  - Utilize unlabeled data to optimize self-supervised pretext tasks
  - Employ fine-tuning or parameter-efficient adaption
- **Hybrid methods**
  - Integrate both paradigms

Outline



Shirui Pan Griffith University
09:00-09:10 Introduction (10mins)

Yuan Fang Singapore Management University
09:10-09:40 Problem and Applications (30mins)

Yuxia Wu Singapore Management University
09:40-10:05 Meta-Learning Approaches (25mins)

Xingtong Yu Singapore Management University
10:05-10:30 Pre-LLM (25mins)

10:30-11:00 Break (30mins)

Yuxia Wu Singapore Management University
10:30-11:00 LLM Era (30mins)

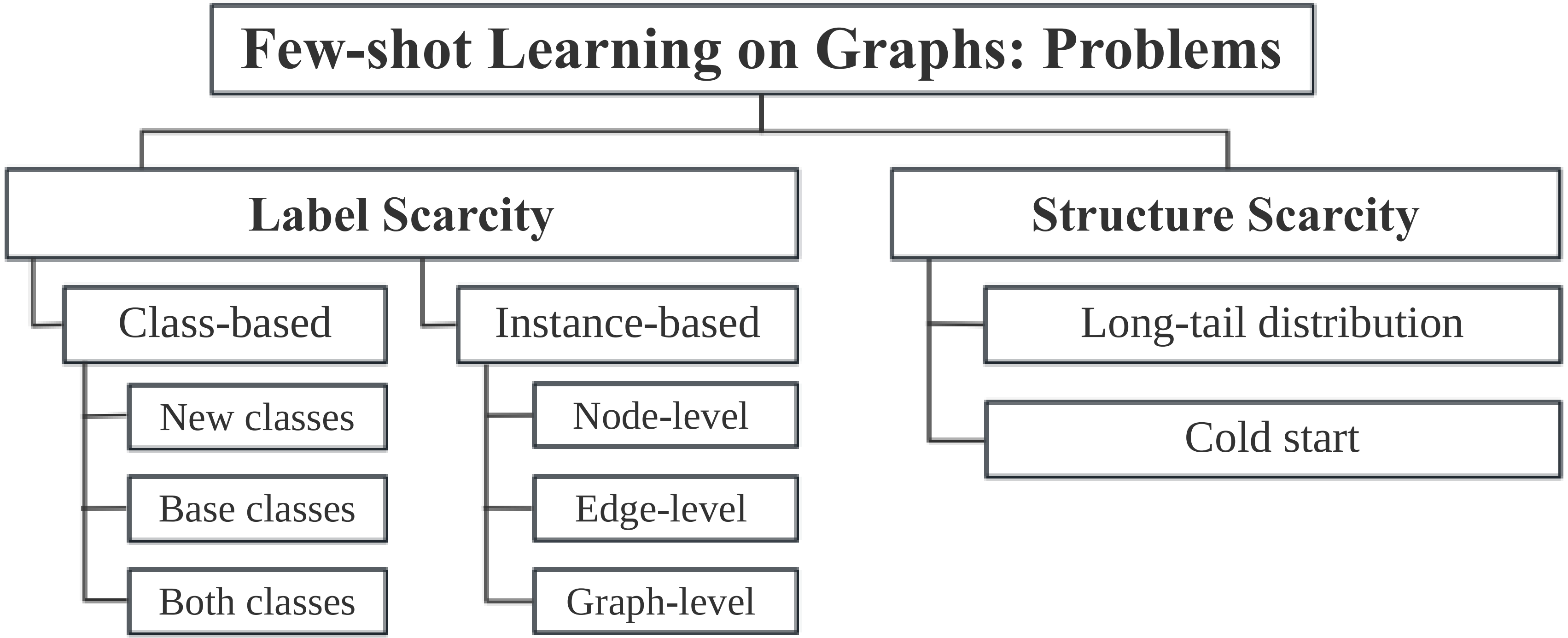
Xingtong Yu Singapore Management University
11:00-11:20 Hybrid Approaches (20mins)

Yuan Fang Singapore Management University
11:20-11:40 Problem and Applications (20mins)

Host: Yuan Fang
11:40-12:30 Panel (50mins)

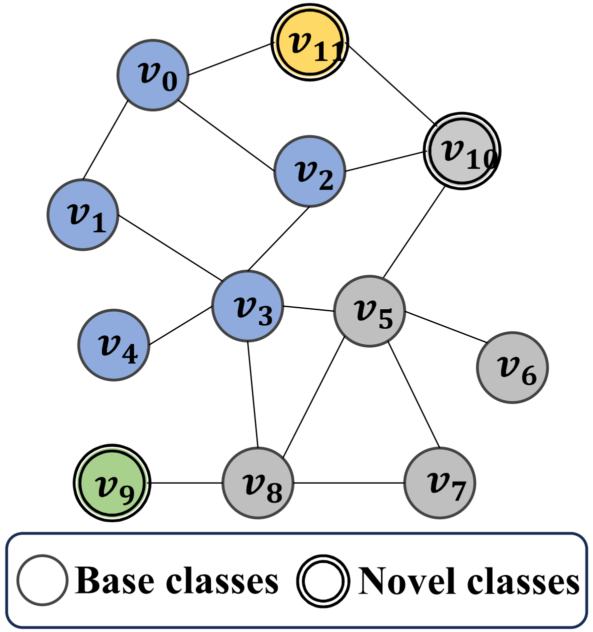
Few-Shot Learning Problems on Graphs

- Label scarcity: the lack of labeled data
- Structure scarcity:



Label Scarcity Problems on Graphs

- Class-based Label Scarcity



The entire set of classes (C) on a graph

- Base class set C_{base} for model training
- New class set C_{new} for testing
- $C = C_{base} \cup C_{new}$
- $C_{base} \cap C_{new} = \emptyset$

Label scarcity could happen in either subsets or both

Label Scarcity Problems on Graphs

- Class-based Label Scarcity

  - Label scarcity in new classes C_{new}

Transfer prior knowledge

Labeled base classes C_{base}

Sufficient!

New classes C_{new}

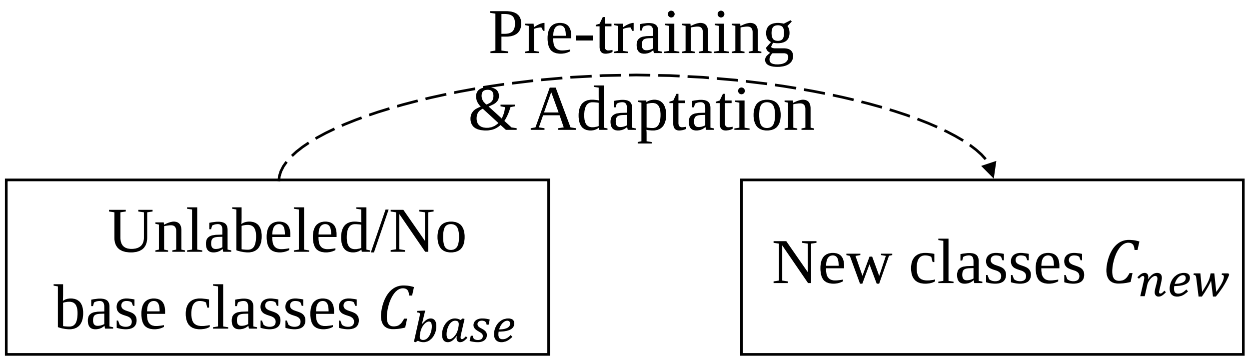
Meta-learning based approaches

- *Support set*: labeled data from C_{new}
- *Query set*: unlabeled data from C_{new}

Heavily rely on abundant labeled data in a large number of base classes!

Label Scarcity Problems on Graphs

- Class-based Label Scarcity
  - ➢ Label scarcity in base classes C_{base}



✗ **Meta-learning based methods**

✓ **Self-supervised methods**

- ❑ **Pre-train** graph encoders on C_{base}
- ❑ **Fine-tuning** on novel tasks

Label Scarcity Problems on Graphs

- Class-based Label Scarcity

  ➢ Label scarcity in both classes: labeled data are limited in both C_{base} and C_{new}

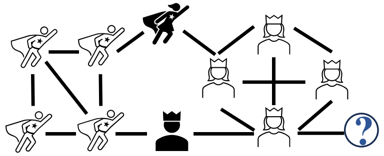
Self-supervised methods

- ❑ **Pre-train** graph encoders
- ❑ **Fine-tuning** on novel downstream tasks: parameter-efficient adaptation

Label Scarcity Problems on Graphs

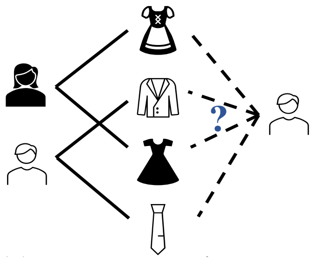
- Instance-based Label Scarcity

➢Node-level label scarcity



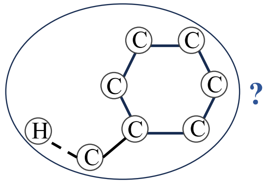
Social network

➢Edge-level label scarcity



Recommender system

➢Graph-level label scarcity



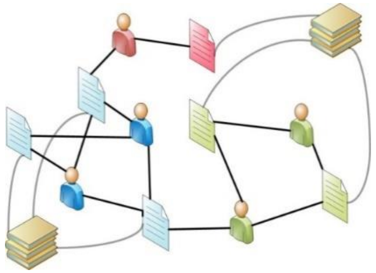
Molecular graph

Instance	**Application domain**
Node	Academic network Social network E-commerce network Protein-protein interaction Traffic flow
Edge	Drug-drug interaction Protein multimer structure E-comm./academic network Knowledge graphs
Graph	Molecular graph Protein graph Social network

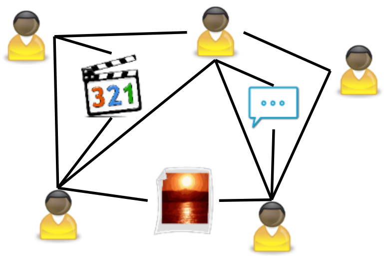
Label Scarcity Problems on Graphs

- Instance-based Label Scarcity
  - ➢ Node-level label scarcity

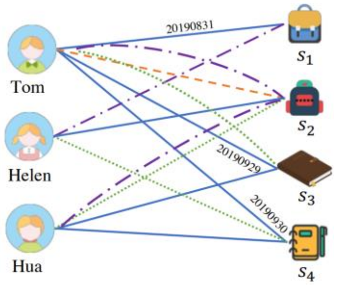
Academic network



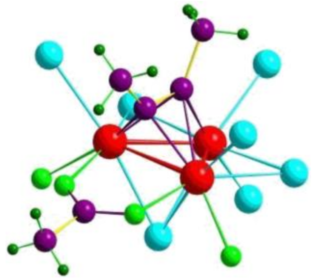
Social network



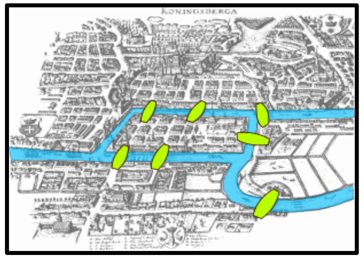
Recommender system



Molecular graph



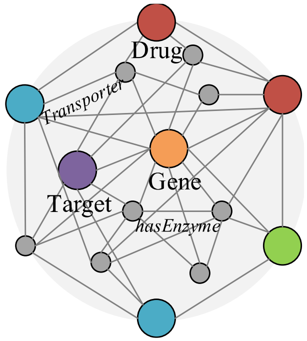
Traffic Flow



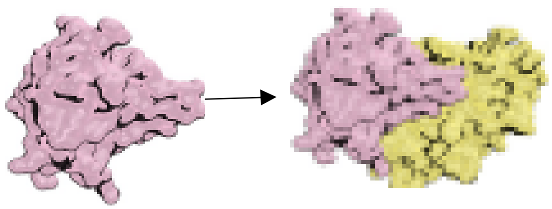
Label Scarcity Problems on Graphs

- Instance-based Label Scarcity
  - ➢Edge-level label scarcity

Drug-drug interaction



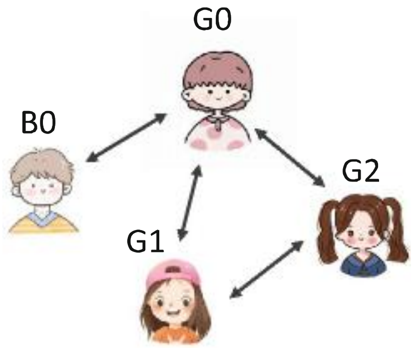
Multimer structure prediction



Knowledge graph

[Image from Microsoft]

E-commerce



Product sharing

Baek, et al. "Learning to extrapolate knowledge: Transductive few-shot out-of-graph link prediction ." NeurIPS'20

Gao, et al. "Protein multimer structure prediction via prompt learning." ICLR'24

Zhu, et al. "Few-shot link prediction for event-based social networks via meta- learning. " DASFAA'23

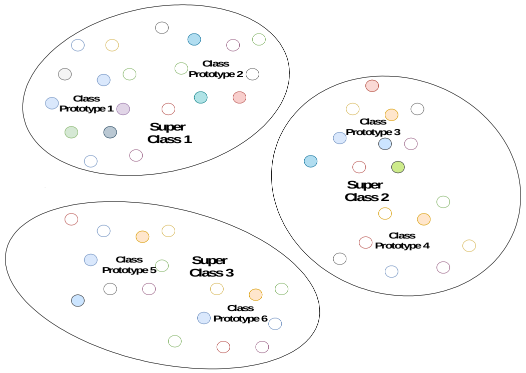
Label Scarcity Problems on Graphs

- Instance-based Label Scarcity

  - ➢Graph-level label scarcity:

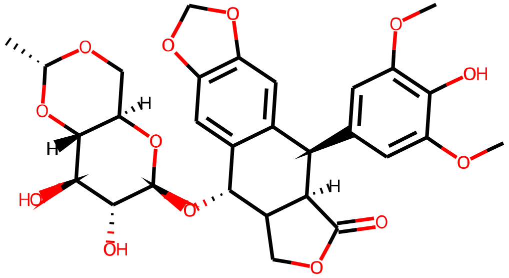
    Predicting properties/categories for subgraphs/whole graphs with limited labeled data

Social Network



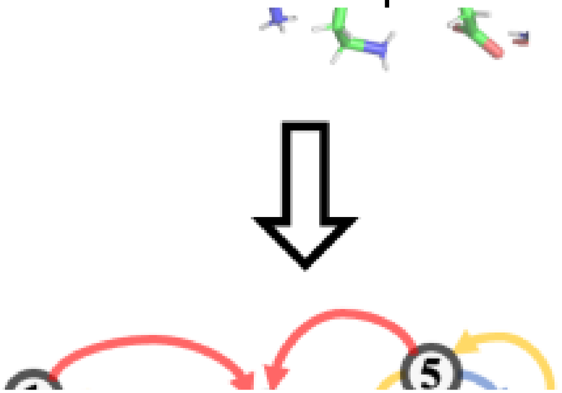
User communities classification

Molecular Graph



Property prediction

Protein Graph



Property prediction

Chauhan, et al. "Few-shot learning on graphs via super-classes based on graph spectral measures." ICLR'20

Zhu, et al. "Dual-view Molecular Pre-training." KDD'23

Zhang, et al. "Protein Representation Learning by Geometric Structure Pretraining. " ICLR'23

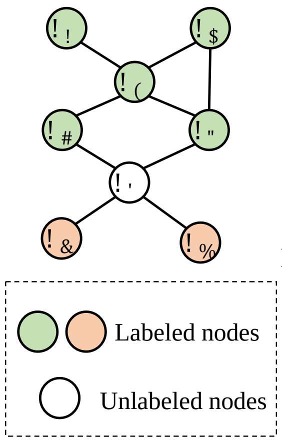
Structure Scarcity Problems on Graphs

- Long-tailed distribution
  - learn from an imbalanced distribution : a large number of nodes have few connections
- Cold-start
  - Learn representations for new nodes with no or very few connections

Goal	**Application domain**
Long-tailed distribution	Academic network Social network E-commerce network Protein-protein interaction Air traffic control
Cold start	Social network E-commerce network

Structure Scarcity Problems on Graphs

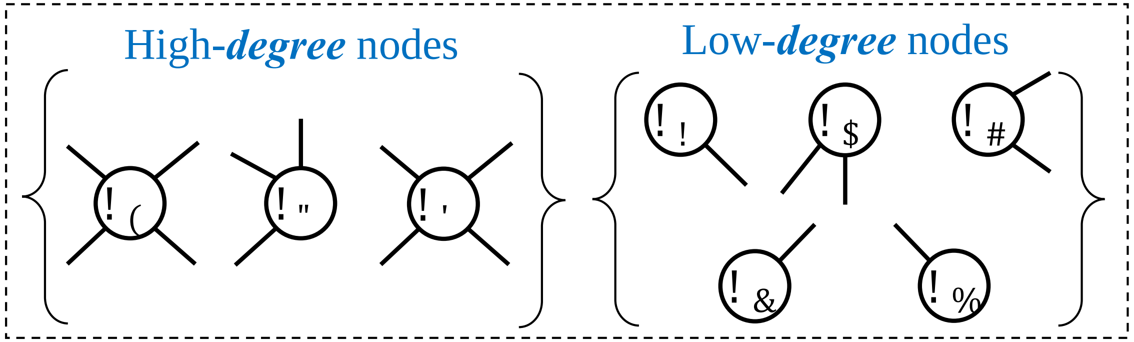
- Long-tailed distribution

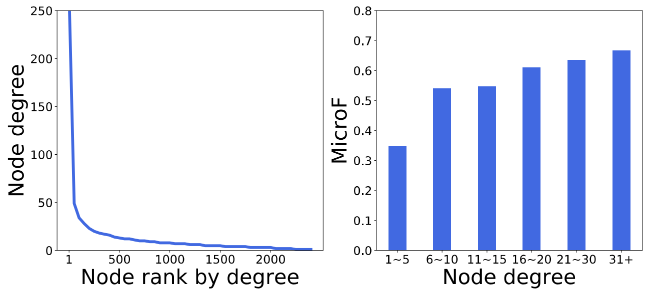


Data split

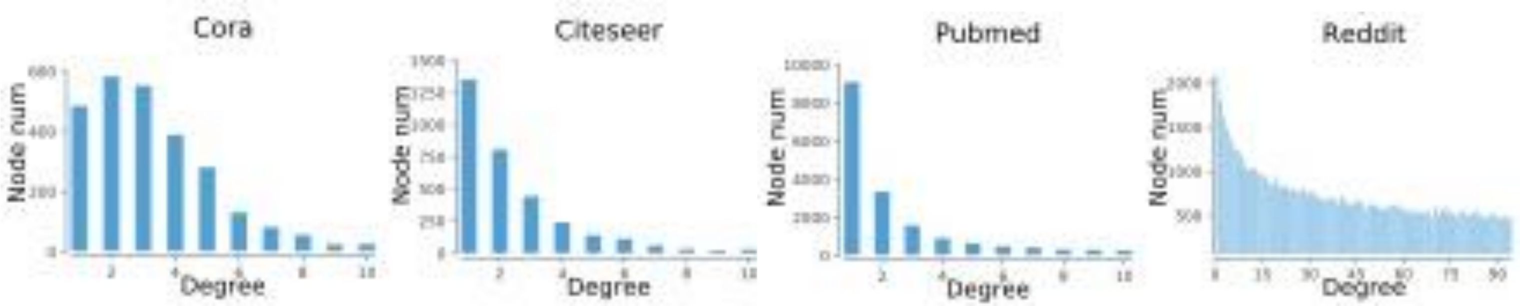
Example:
Node representation learning with different degrees
(High: degree ≥ 3)

Head nodes *Tail* nodes





(a) Degree distribution (b) Classification performance



Liu, et al. "A Survey of Imbalanced Learning on Graphs: Problems, Techniques, and Future Directions. " arXiv'23

Tang, et al. "Investigating and Mitigating Degree-Related Biases in Graph Convolutional Networks." CIKM'20

Structure Scarcity Problems on Graphs

- Cold-start learning: new nodes with few connections

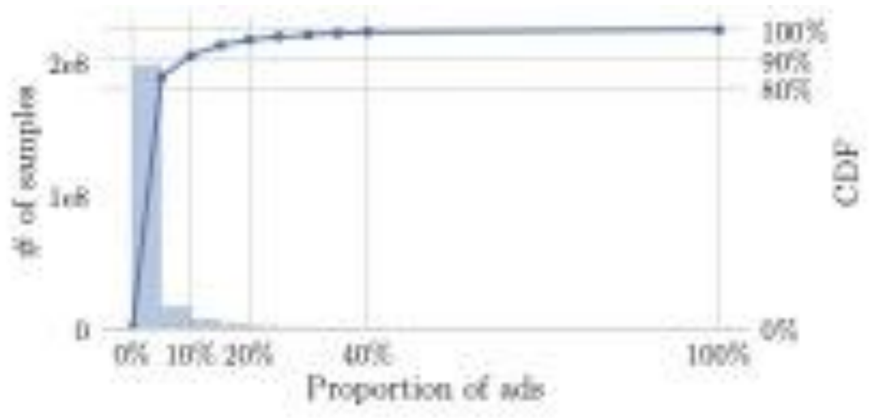


Figure 1: Histogram of the number of samples over different proportions of ads of the KDD Cup 2012 search ads dataset.

5% of ads accounted for over 80% of samples;
95% ads had a very small amount of data.

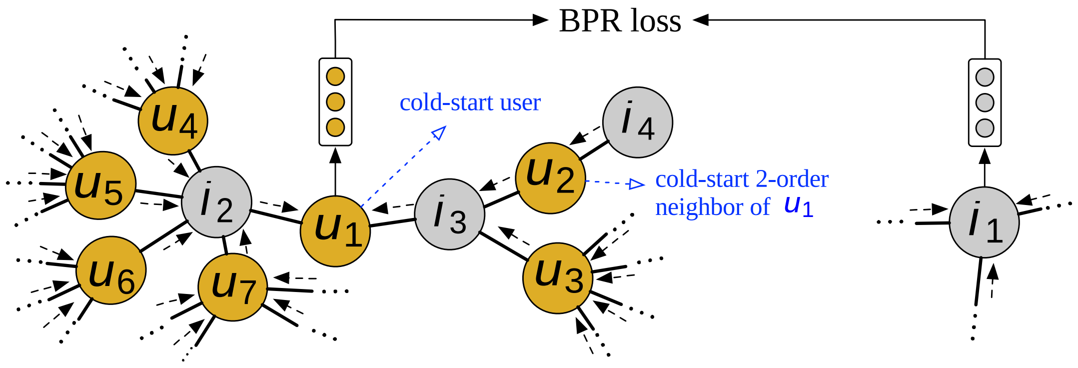


Figure 1: A GNN model for recommendation.

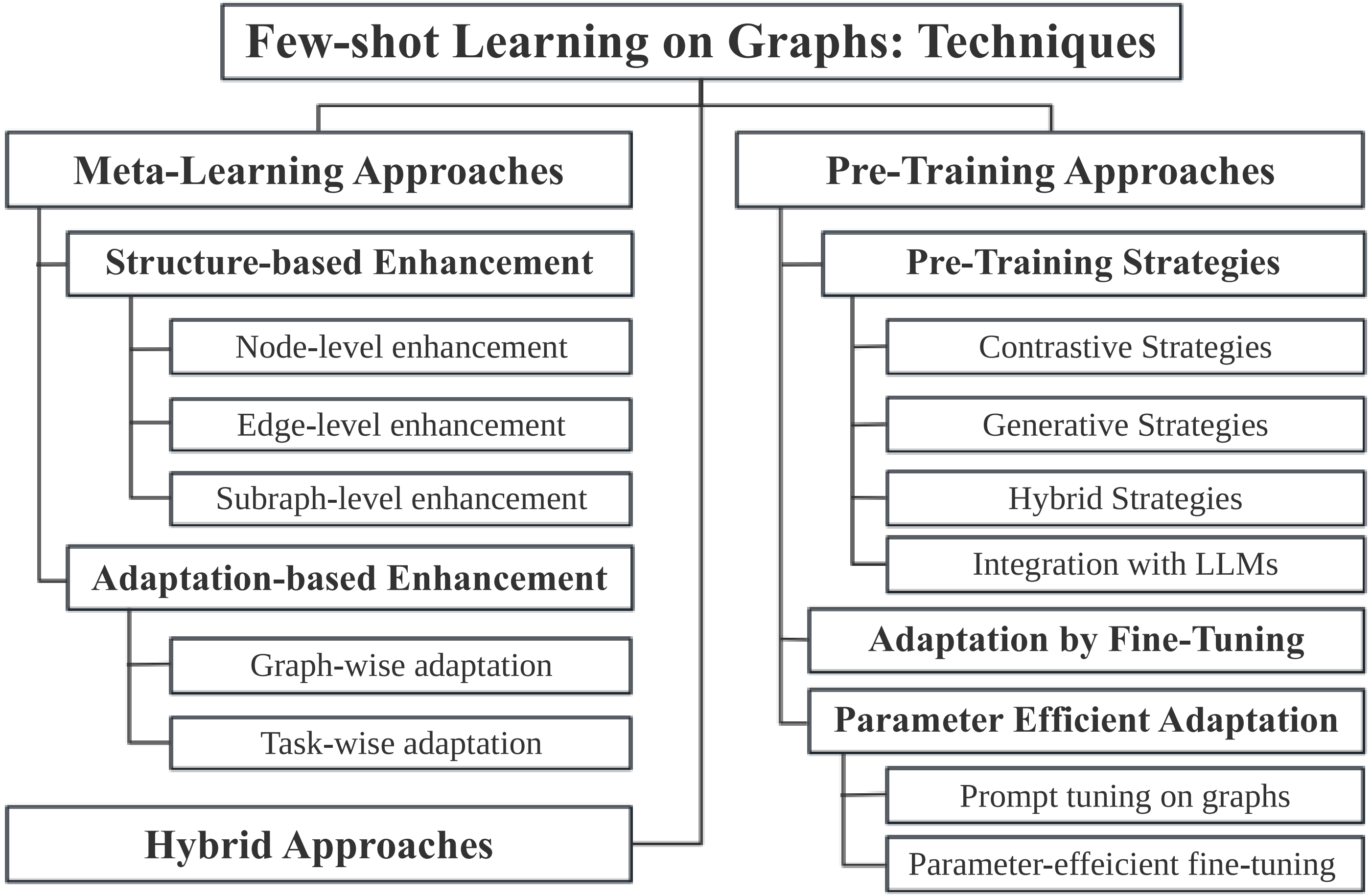
Classic GNNs may have limited effectiveness in addressing cold-start problems

Pan, et al. “Warm Up Cold-start Advertisements- Improving CTR Predictions via Learning to Learn ID Embeddings.” SIGIR’19

Hao, et al. “Pre-Training Graph Neural Networks for Cold-Start Users and Items Representation.” WSDM’21

Overall Taxonomy

- Taxonomy of few-shot learning techniques on graphs

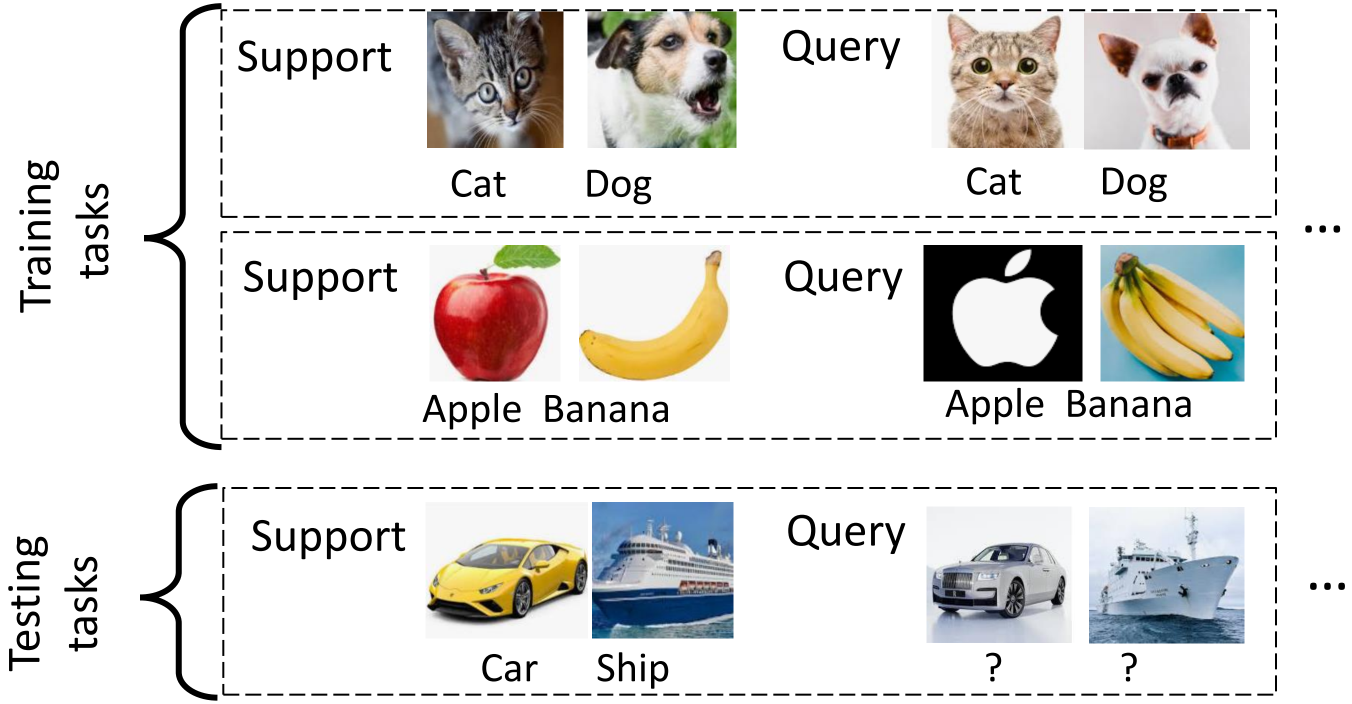


Meta-learning techniques on graphs

- Standard meta-learning techniques

Meta-learning

(MAML)



Learn a prior ϕ from the training tasks

Adapt

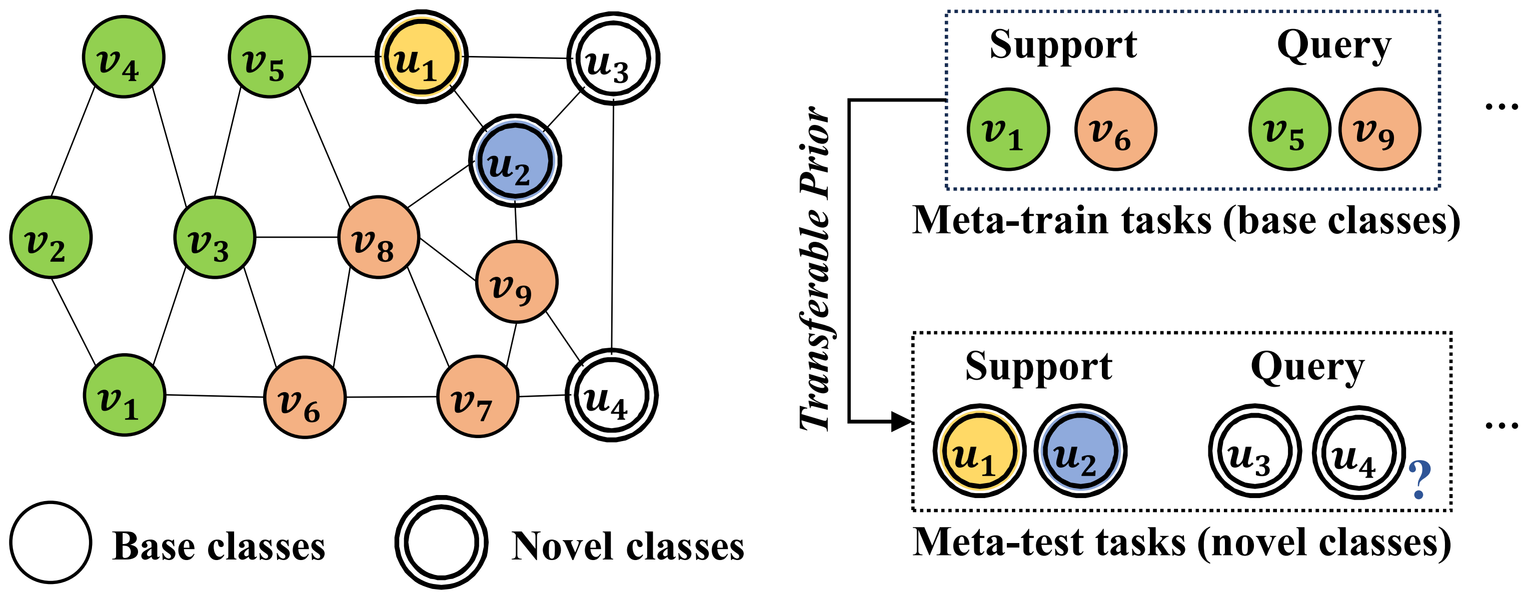
$f_{\phi'}(\;) \rightarrow \text{car}$

"Learn to learn"

C. Finn et al. "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks." ICML 2017.

Meta-learning techniques on graphs

- Standard meta-learning on graph



(a) Toy graph with base and novel classes (b) Few-shot node classification

Structure-based Enhancement on Graphs

- Node-level enhancement: GPN

Differentiating node weights in a task to reflect their varying structural importance

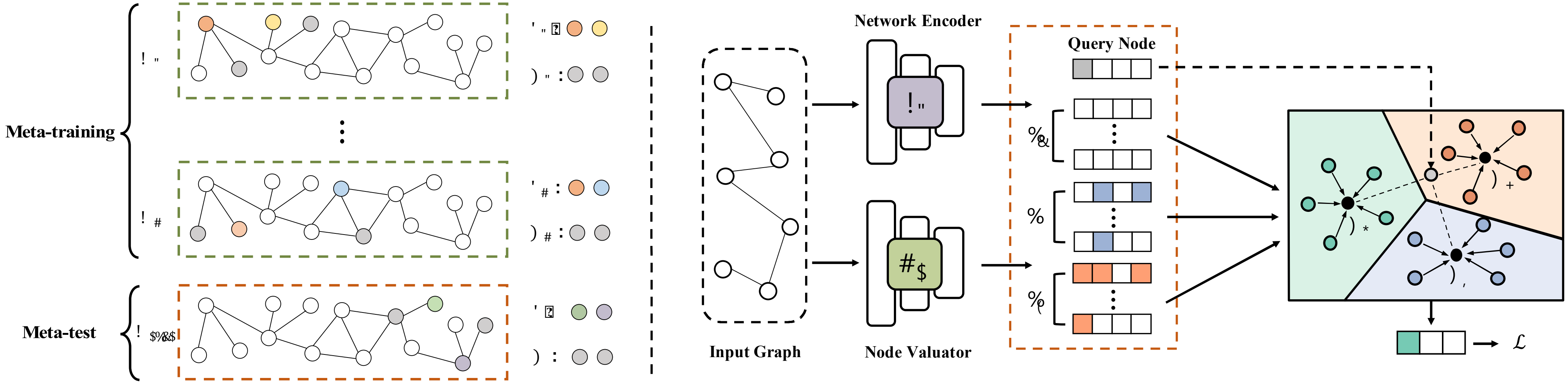


Figure 2: (Left) Episodic training on attributed networks. In each episode, we create a semi-supervised few-shot node classification task by random sampling; (Right) The architecture of the proposed framework Graph Prototypical Networks (GPN).

K. Ding, *et al* "Graph prototypical networks for few-shot learning on attributed networks." CIKM'20

Structure-based Enhancement on Graphs

- Node-level enhancement: FAAN

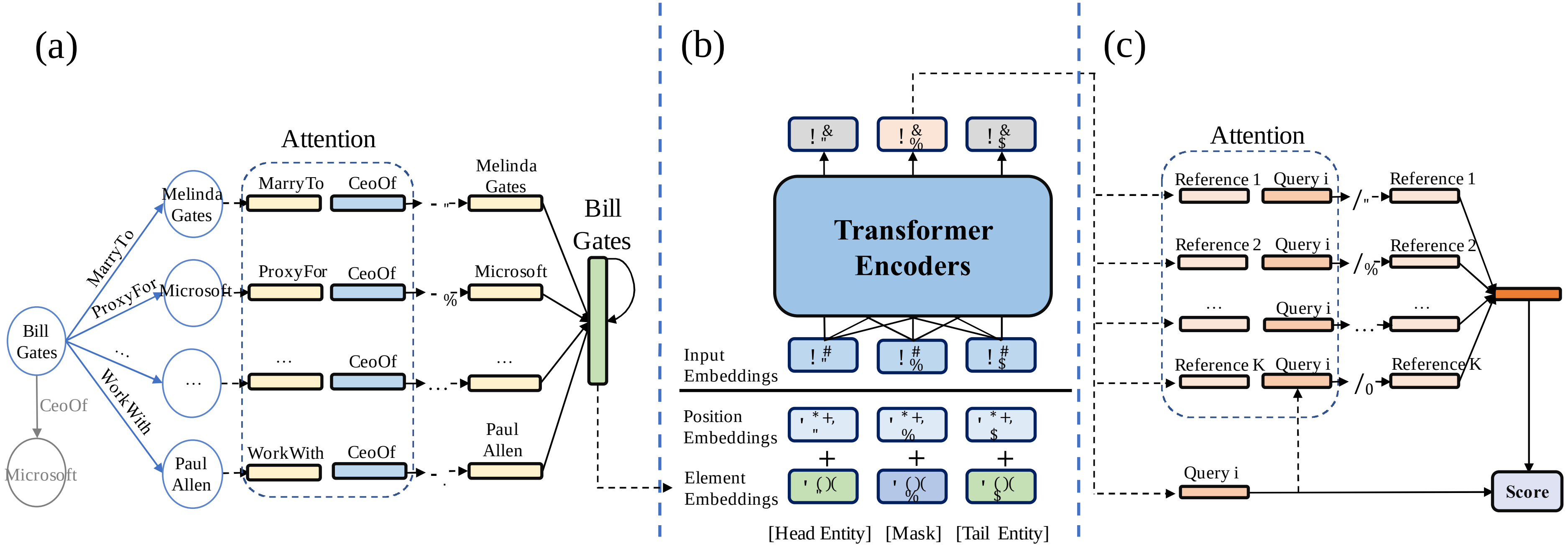


Figure 2: The framework of FAAN: (a) Adaptive neighbor encoder for entities; (b) Transformer encoder for entity pairs; (c) Adaptive matching processor to match K -shot references and the query.

J. Sheng, *et al* "Adaptive attentional network for few-shot knowledge graph completion." EMNLP'20

Structure-based Enhancement on Graphs

- Edge-level enhancement: HMNet, TCVAE
  - ➢Leverage auxiliary information associated with edges

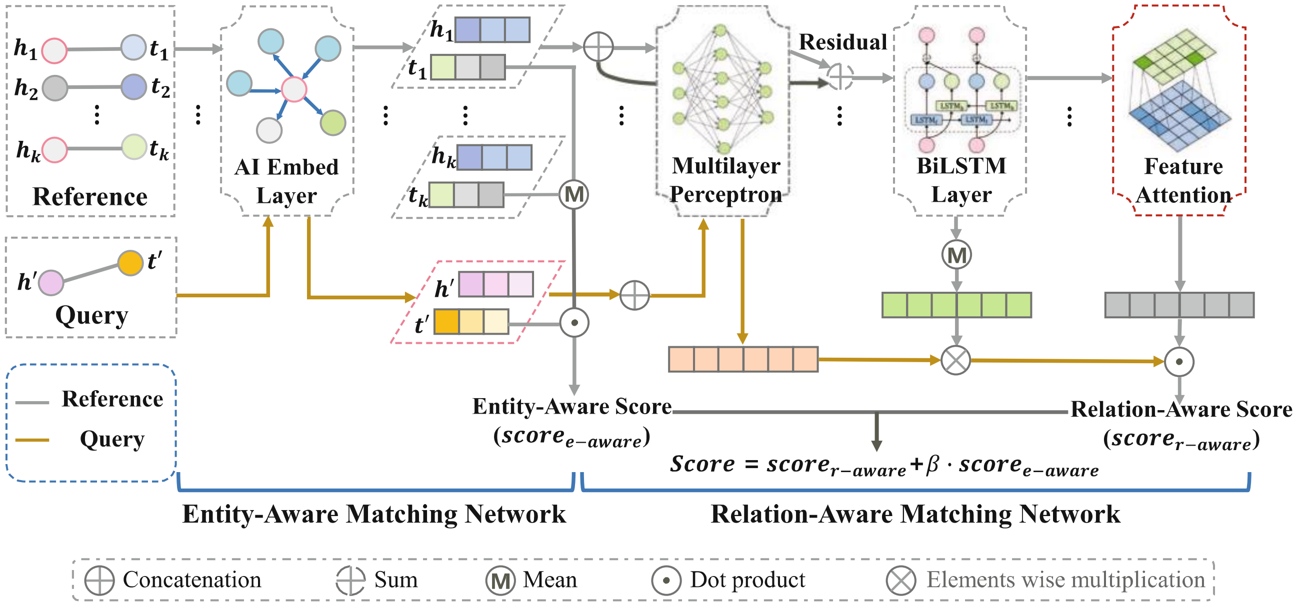


Fig. 2. Illustration of the proposed HMNet model.

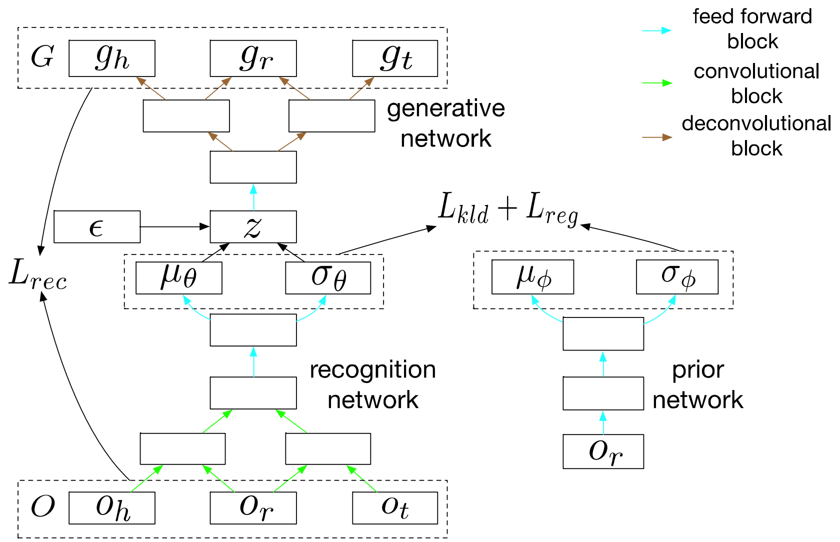


Figure 3: Structure of TCVAE. This figure should be viewed in color.

S. Xiao, et al. “HMNet- Hybrid Matching Network for Few-Shot Link Prediction.” DASFAA’21

Z. Wang, *et al* “Tackling Long-Tailed Relations and Uncommon Entities in Knowledge Graph Completion.” EMNLP’19

Structure-based Enhancement on Graphs

- Edge-level enhancement: RALE
  - ➢Leverage paths to capture long-range dependencies between distant node

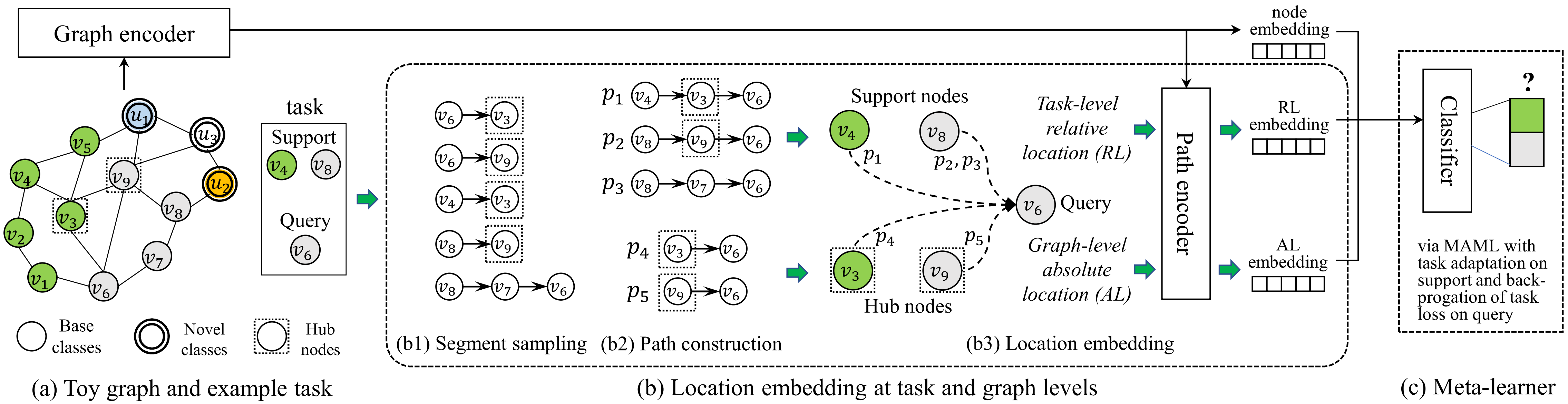


Figure 3: Overview of the proposed model RALE.

- Paths between each query node and the support nodes: Task-level dependencies
- Paths between each target node and the hub nodes: Graph-level dependencies

Z. Liu, *et al* “Relative and Absolute Location Embedding for Few-Shot Node Classification on Graph.” AAAI’21

Structure-based Enhancement on Graphs

- Edge-level enhancement: MetaHIN

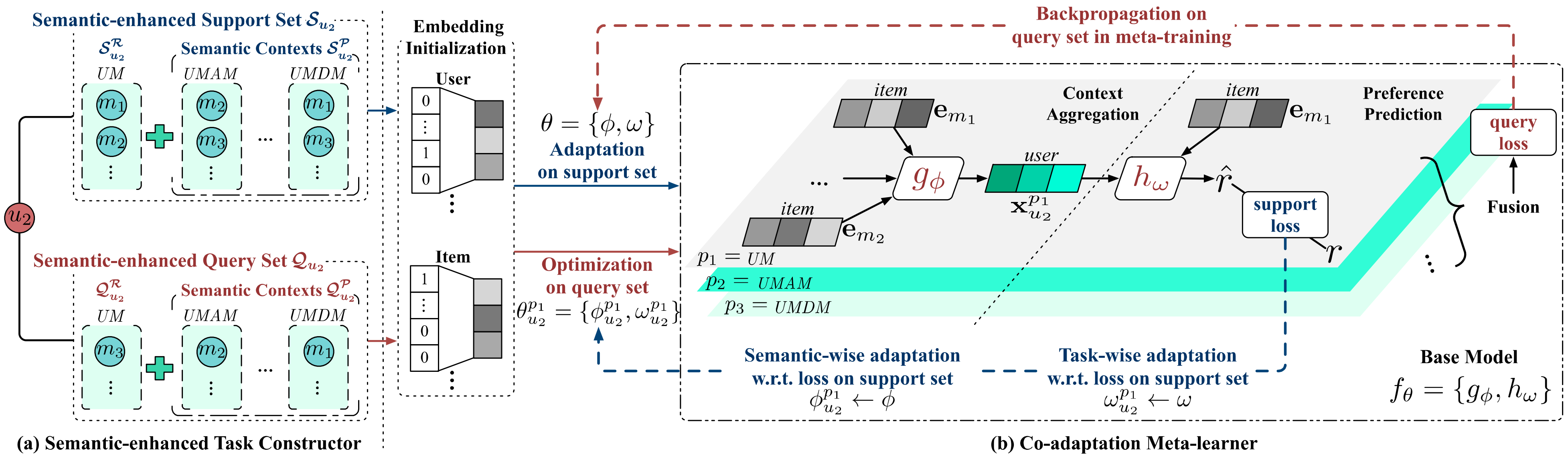


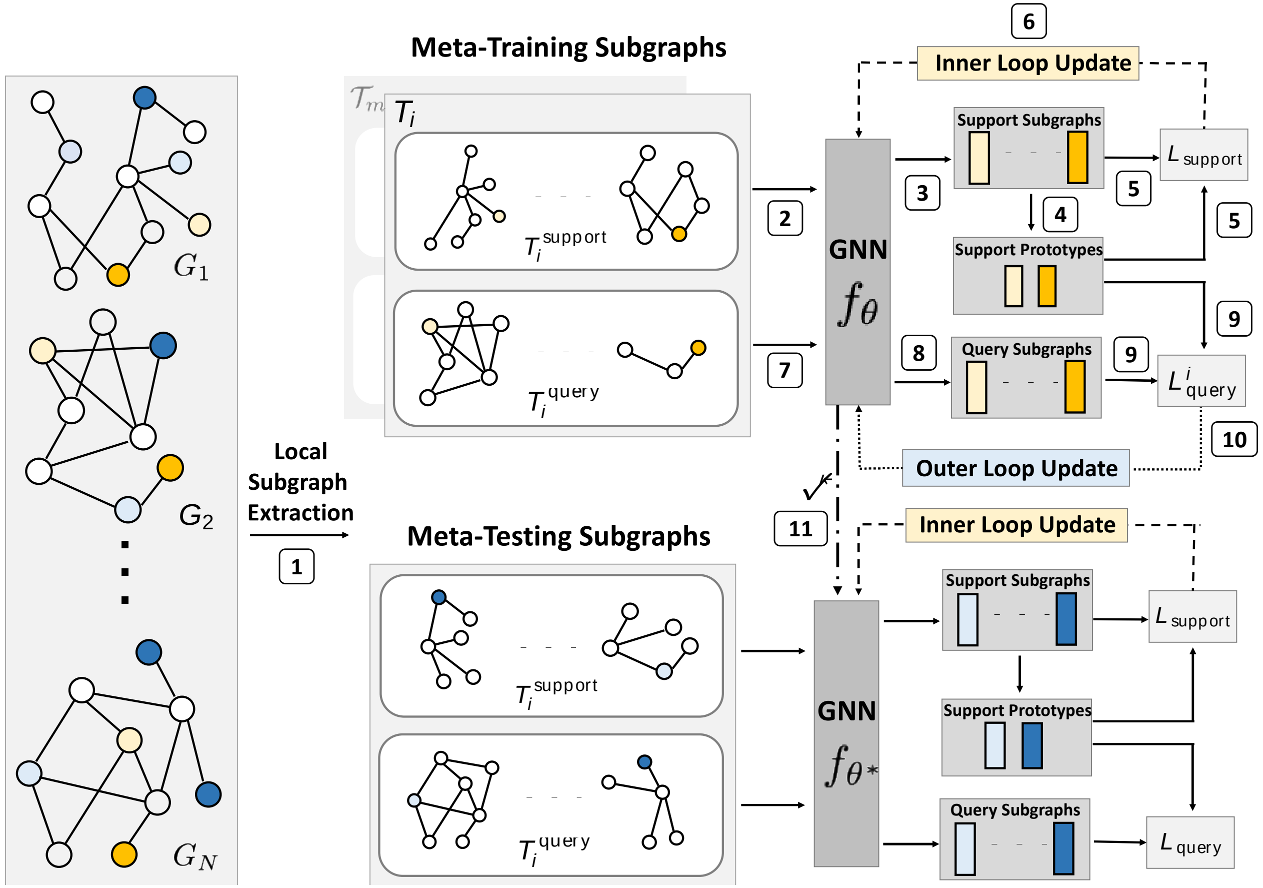
Figure 2: Illustration of the meta-training procedure of a task in MetaHIN. (a) Semantic-enhanced task constructor, where the support and query sets are augmented with meta-path based heterogeneous semantic contexts. (b) Co-adaptation meta-learner, with semantic- and task-wise adaptations on the support set, while the global prior θ is optimized on the query set. During meta-testing, each task follows the same procedure except updating the global prior.

- Meta-paths: heterogeneous semantic relationships

Y. Lu, *et al* "Meta-learning on Heterogeneous Information Networks for Cold-start Recommendation ." KDD'20

Structure-based Enhancement on Graphs

- Subgraph-level enhancement: G-Meta

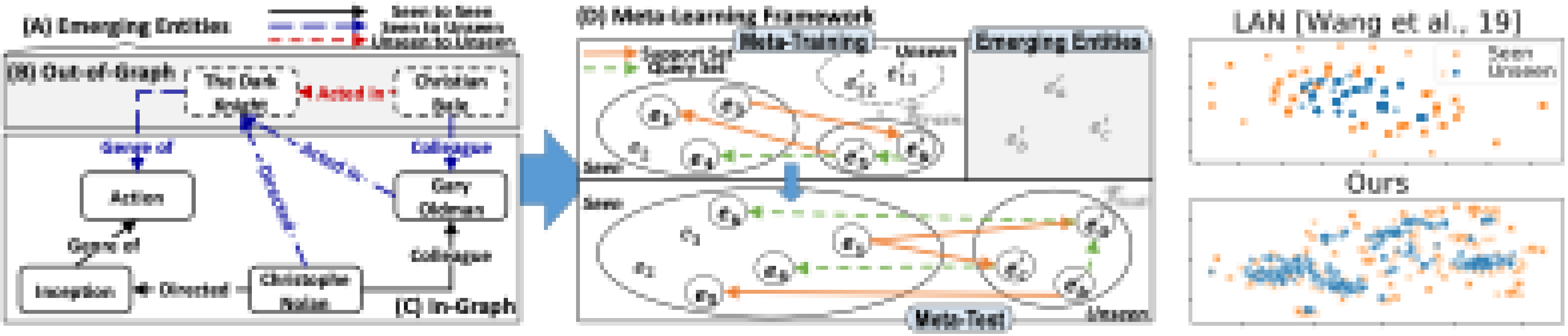


- Generate class prototypes from subgraph
- Expand query node to its subgraph

K. Huang, *et al* "Graph Meta Learning via Local Subgraphs." NeurIPS'20

Structure-based Enhancement on Graphs

- Subgraph-level enhancement: GEN



  - Extrapolate knowledge through the neighbors (one-hop subgraph) of the support set
  - Embeds the unseen entities on the manifold of seen entities

J. Baek, *et al* "Learning to Extrapolate Knowledge: Transductive Few-shot Out-of-Graph Link Prediction." NeurIPS'20

Structure-based Enhancement on Graphs

- Subgraph-level enhancement: Meta-tail2vec

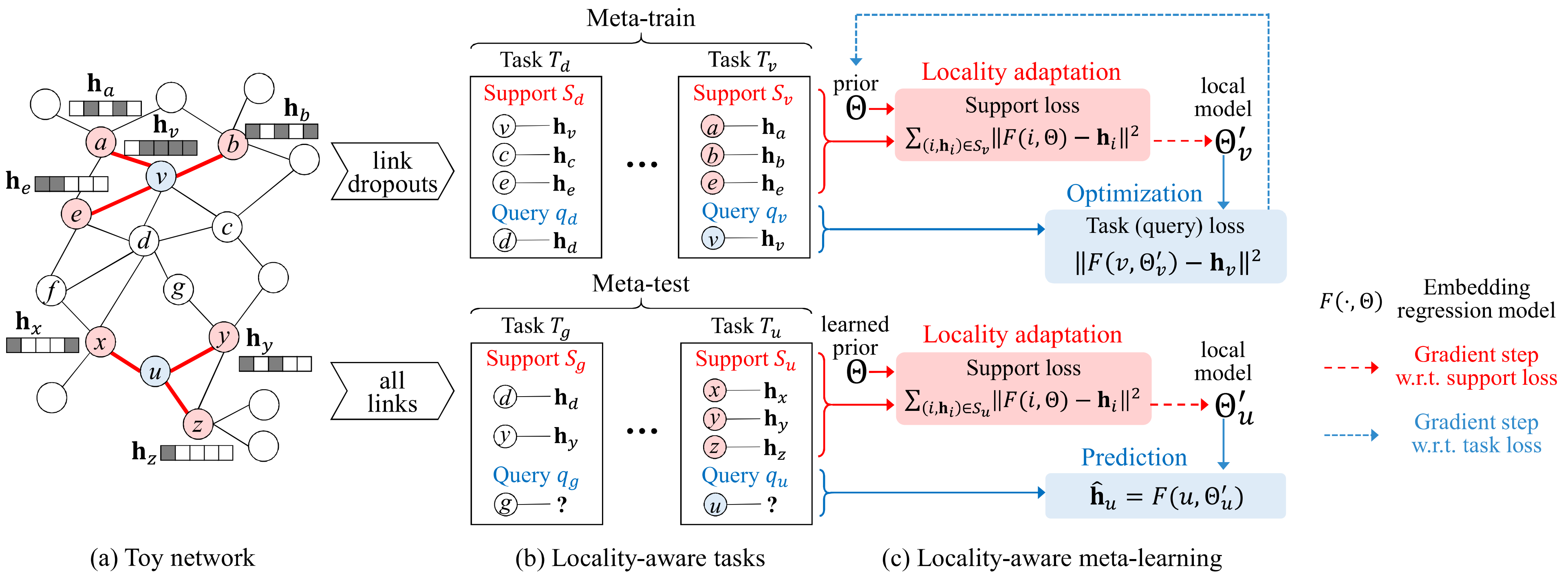


Figure 3: Overall framework of our locality-aware tail node embedding model meta-tail2vec. (Best viewed in color.)

- Locality-aware tasks: support set sampled from the neighborhood subgraph of the query node

Z. Liu, *et al* “Towards Locality-Aware Meta-Learning of Tail Node Embeddings on Networks.” CIKM’20

Adaptation-based Enhancement on Graphs

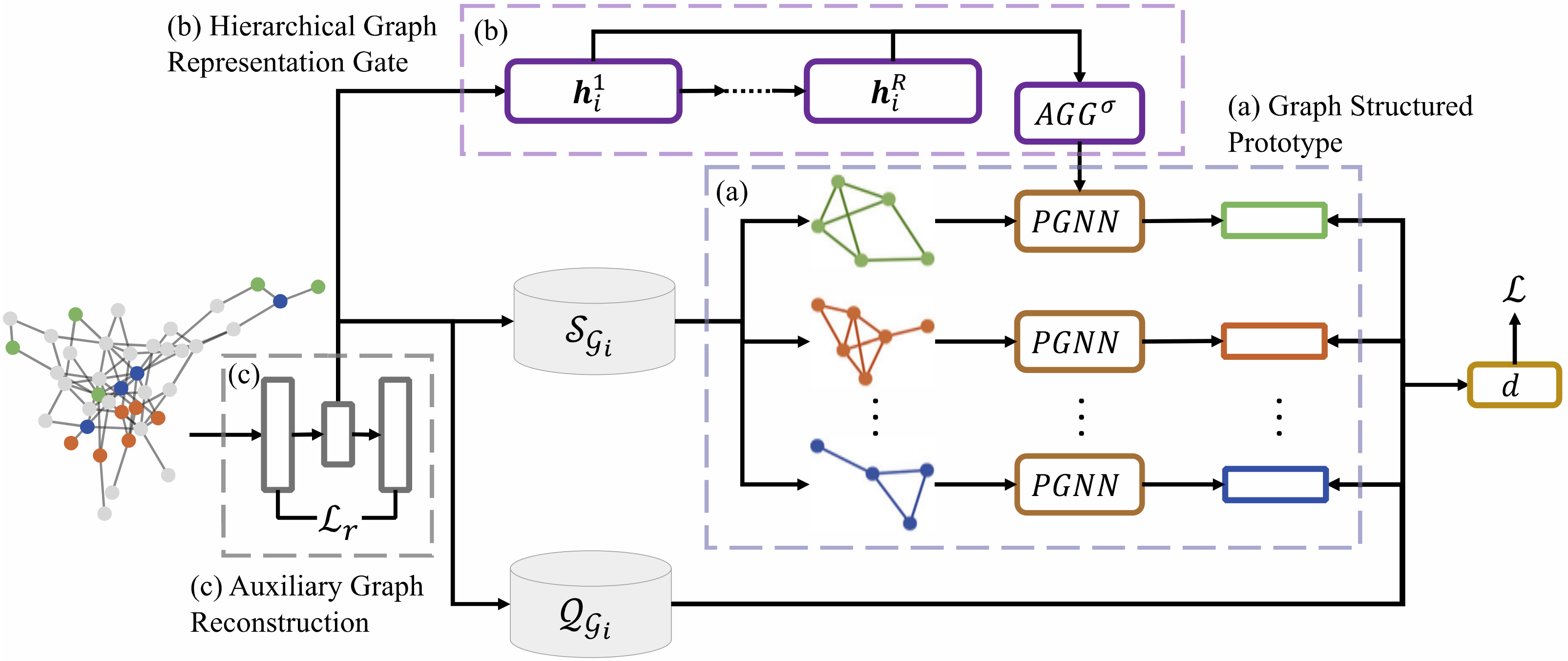
- Customization of a globally shared prior into a localized or specialized model for each task

TABLE V: Adaptation-based meta-learning enhancement for few-shot learning on graphs.

Method	**Adaptation enhancement**	**Meta learner**	**Task** Node	Edge	Graph
GFL [36]	graph	MAML	✓	×	×
MI-GNN [145]	graph	hybrid	✓	×	×
MetaTNE [32]	task	Protonets	✓	×	×
AMM-GNN [65]	task	MAML	✓	×	×
AS-MAML [148]	step size	MAML	×	×	✓
MetaDyGNN [137]	hybrid	MAML	×	✓	×

Adaptation-based Enhancement on Graphs

- Graph-wise adaptation: GFL



  - Recognize the topological variances across different graphs
  - Customize a global prior for each individual graph (class prototypes tailored to each graph)
  - Apply gate function to the global prior

H. Yao, *et al*. "Graph Few-Shot Learning via Knowledge Transfer." AAAI'20

Adaptation-based Enhancement on Graphs

- Graph-wise adaptation: MI-GNN

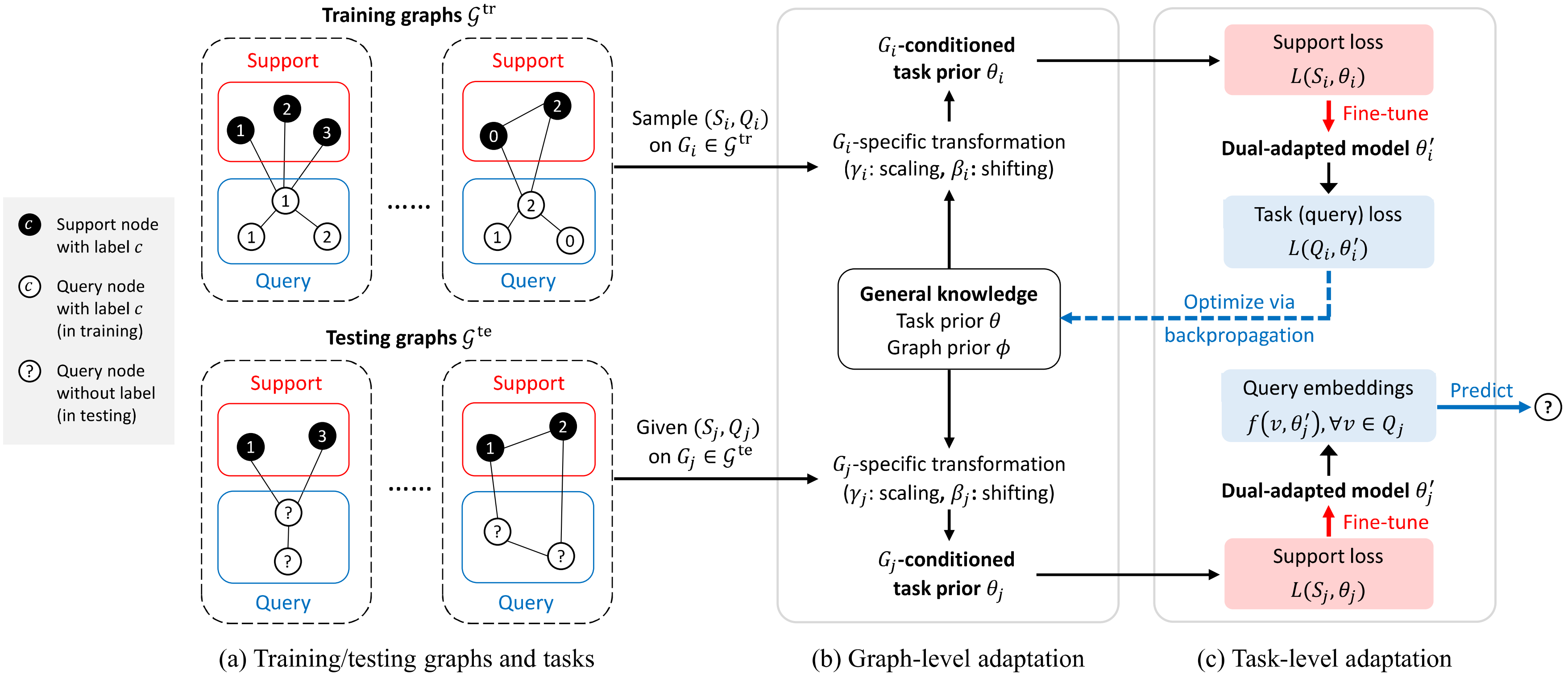


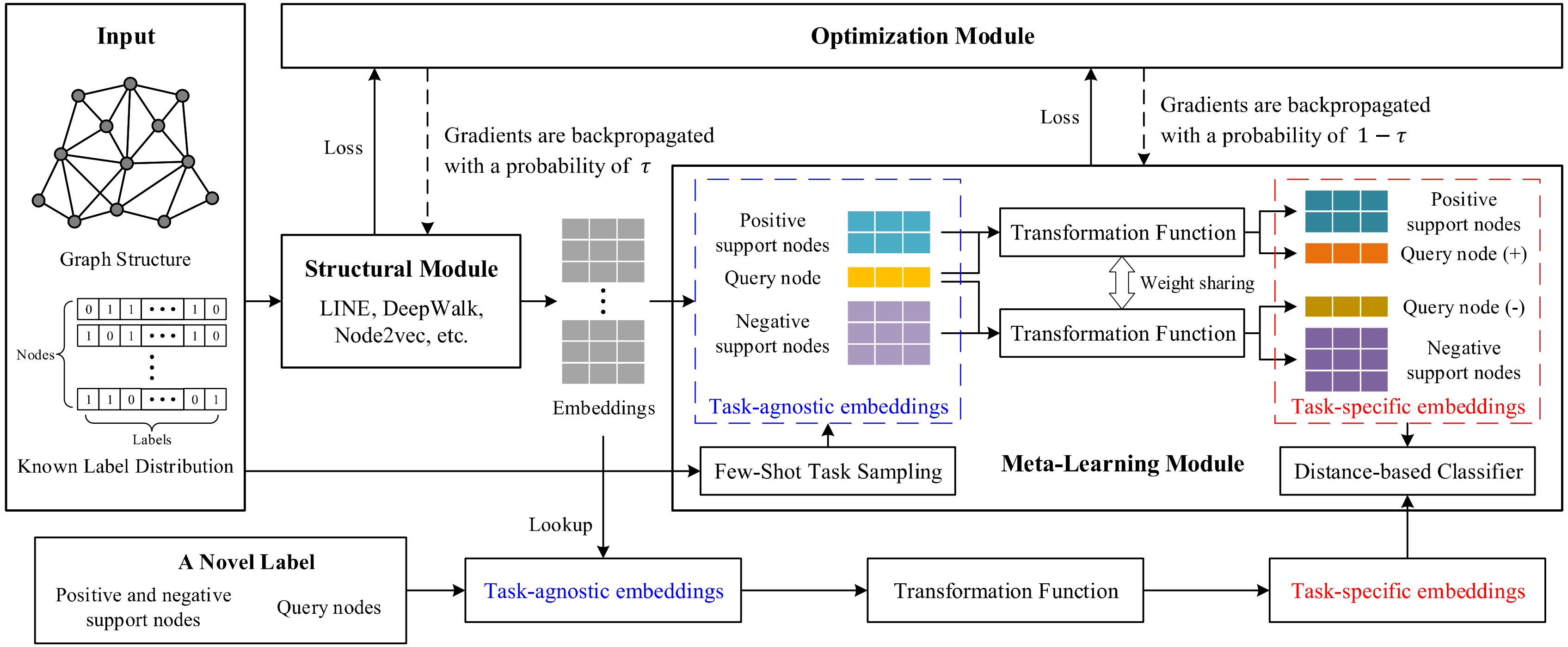
Figure 2: Overall framework of MI-GNN, illustrating the pipeline on a training graph G_i and a testing graph G_j.

- Employ a Feature-wise Linear Modulation (FiLM) to modulate the global prior for each graph

Z. Wen, *et al.* "Meta-inductive node classification across graphs." SIGIR'21

Adaptation-based Enhancement on Graphs

- Task-wise adaptation: MetaTNE



- Multi-label few-shot classification: same node could be associated with different labels in different tasks
- Adaptation for the node embeddings (the query set in each task)

L. Lan, *et al*. "Node classification on graphs with few-shot novel labels via meta transformed network embedding." NeurIPS'20

Adaptation-based Enhancement on Graphs

- Task-wise adaptation: AMM-GNN

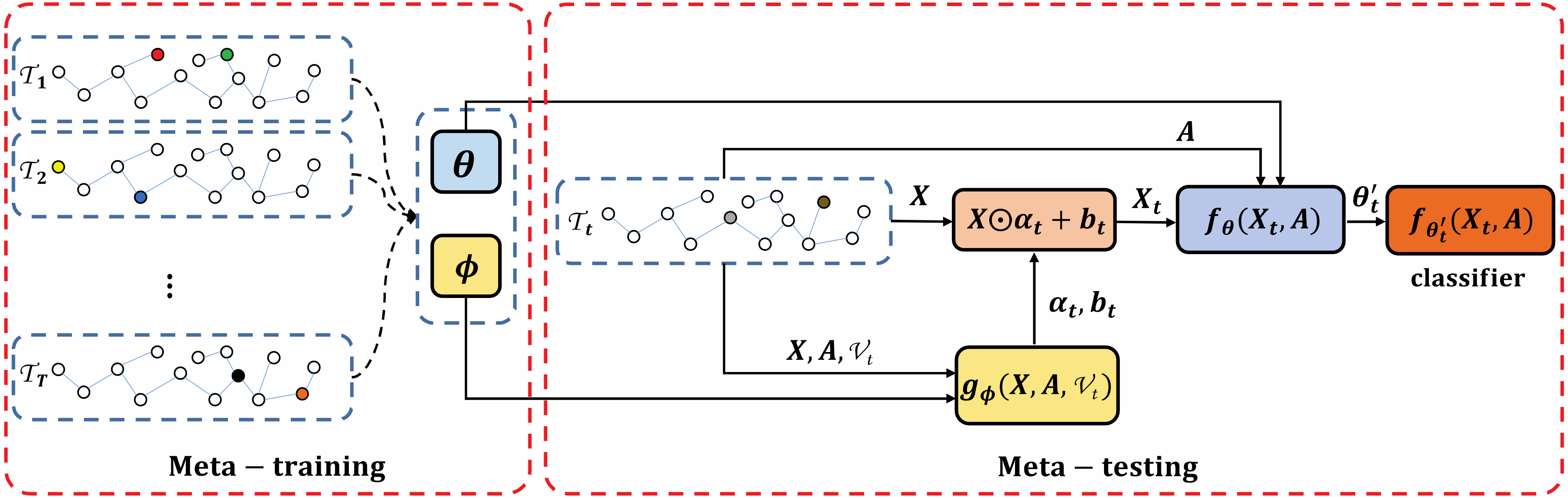


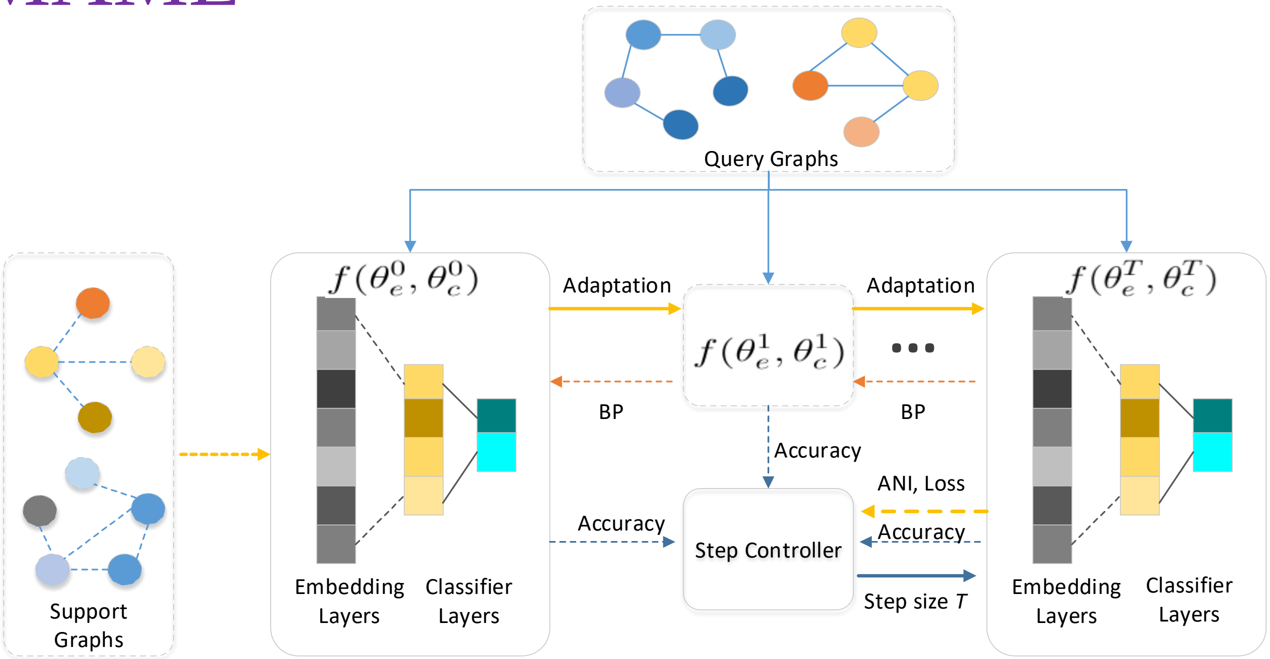
Figure 1: The overview of the proposed AMM-GNN framework. *Left*: In the meta-training phase, multiple tasks are sampled to train the meta-learning model, and we obtain two parameter sets θ and ϕ. *Right*: In the meta-testing phase, we use parameter sets ϕ and θ for attribute matching and gradient descent respectively, and obtain the classifier $f_{\theta'_t}(\cdot)$ for a new sampled task $\mathcal{T}_t$.

  - Customize a task-specific feature matrix for adaptation

N. Wang, *et al*.. "Graph Few-shot Learning with Attribute Matching." CIKM'20

Adaptation-based Enhancement on Graphs

- Others: AS-MAML

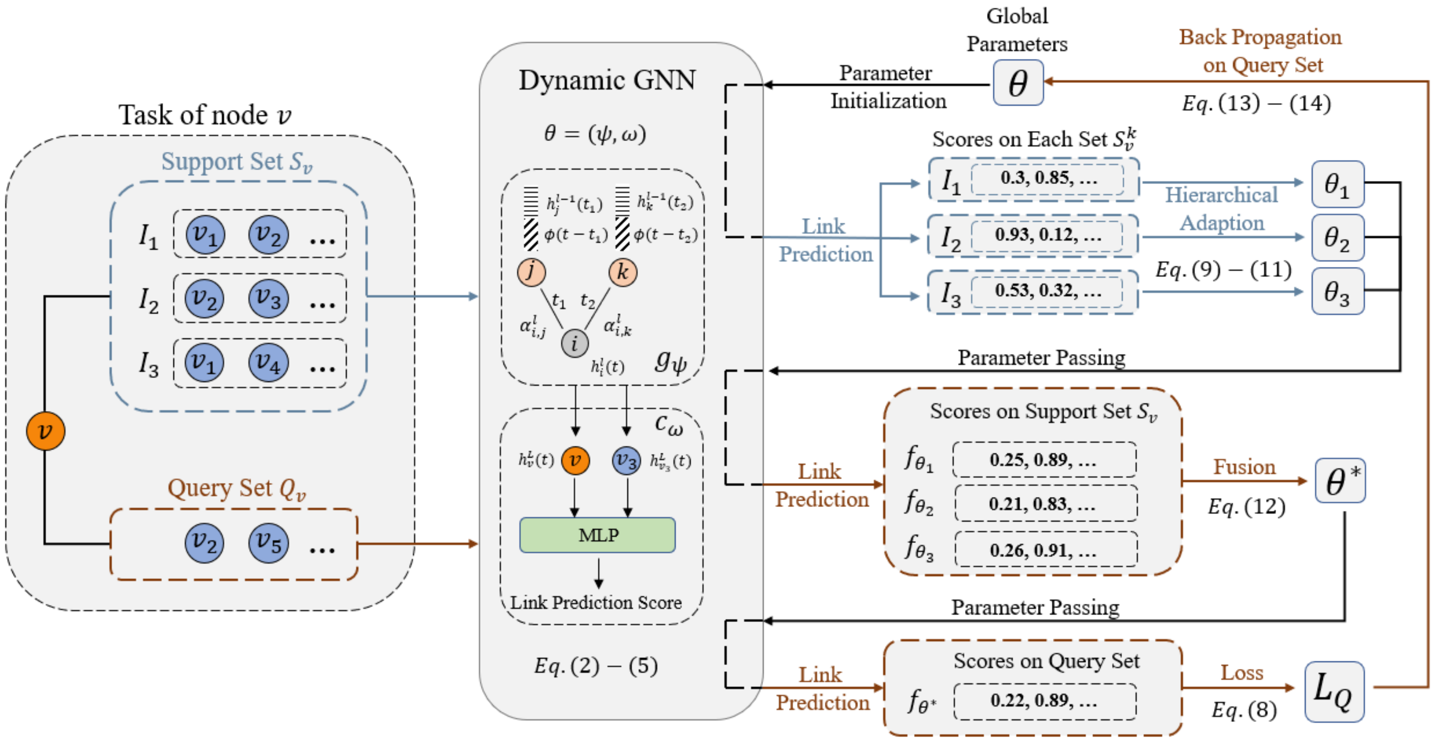


- Improve adaptation from an optimization standpoint
- Reinforcement learning-based controller to determine the optimal step size for the adaptation process

N. Ma, *et al*. "Adaptive-Step Graph Meta-Learner for Few-Shot Graph Classification." CIKM'20

Adaptation-based Enhancement on Graphs

- Others: MetaDyGNN



- Adaptation for dynamic graphs: time- and node-weise

C. Yang, et al. "Few-shot Link Prediction in Dynamic Networks." WSDM'22

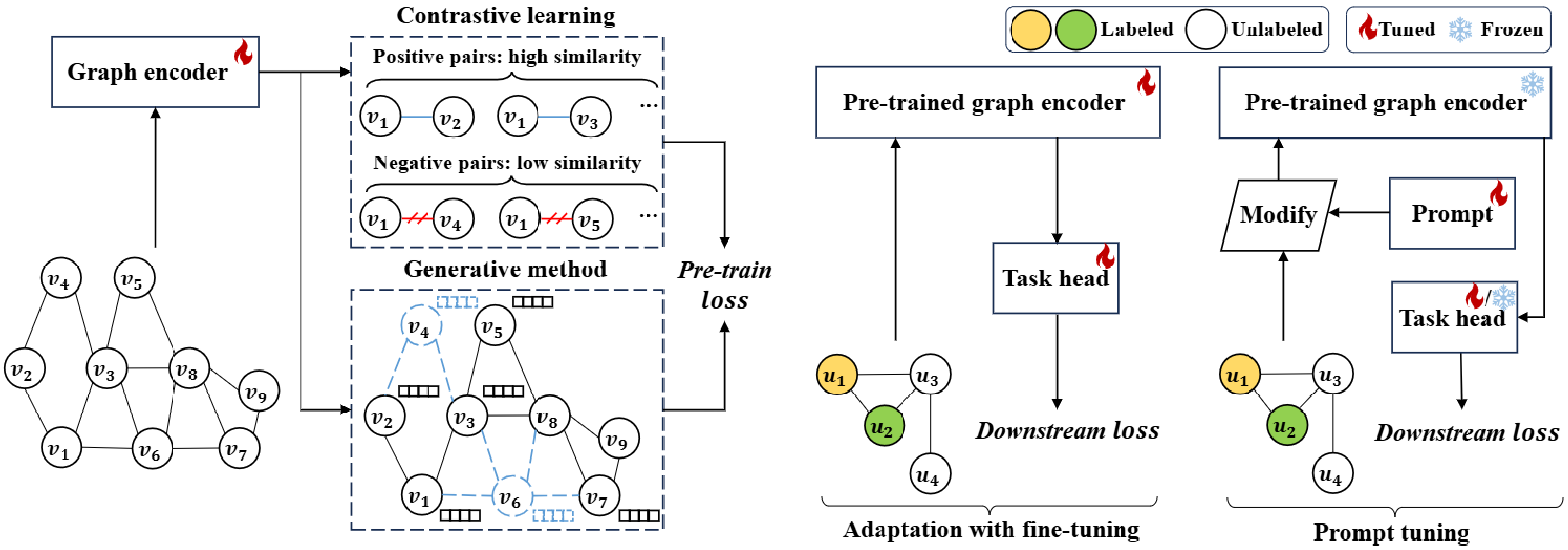
Summary

- Existing research often enhances a standard meta-learner: structural augmentation or refining the adaptation process
- Drawbacks:
  - Require abundant labels for a base set during the meta-training phase
  - Fail to leverage the vast amount of unlabeled data to learn a more comprehensive prior
  - Limited by the i.i.d. assumption in task distribution, and cannot handle different types of downstream tasks

Can we address a *diverse* range of few-shot tasks on graphs *without an extensively annotated base set*, while *utilizing abundant unlabeled graphs*?

Pre-training on Graphs

- Pre-training stage utilizes self-supervised method
- Prior knowledge are then adapted to downstream tasks



(a) Unlabeled graph data (b) Pre-training stage (c) Downstream adaptation

Pre-training Strategies

- Graph pre-training strategies mainly fall into:
  - Contrastive strategies
  - Generative strategies

Contrastive Strategies

- Contrasting instances at various scales within a graph
  - ➢ Sample positive and negative instances
  - ➢ Positive instances closer to the target
  - ➢ Negative instances further to the target

Pre-training data $\mathcal{T}_{\text{pre}}$ Target instance o

Positive samples $\mathcal{P}_o$ Negative samples $\mathcal{N}_o$

Contrastive loss:

$$-\sum_{o\in\mathcal{T}_{\text{pre}}} \ln \frac{\sum_{a\in\mathcal{P}_o} \exp\left(\frac{\mathtt{sim}(\mathbf{h}_a,\mathbf{h}_o)}{\tau}\right)}{\sum_{a\in\mathcal{P}_o} \exp\left(\frac{\mathtt{sim}(\mathbf{h}_a,\mathbf{h}_o)}{\tau}\right)+\sum_{b\in\mathcal{N}_o} \exp\left(\frac{\mathtt{sim}(\mathbf{h}_b,\mathbf{h}_o)}{\tau}\right)}$$

Method	Instance	Augmentation	Graph types
GRACE [72]	node	uniform	general
GCC [30]	graph	uniform	general
GraphCL [40]	graph	uniform	general
SimGRACE [74]	graph	perturbing encoder	general
GraphLoG [73]	dataset	uniform	general
DGI [29]	cross-scale	uniform	general
InfoGraph [42]	cross-scale	uniform	general
Subg-Con [71]	cross-scale	uniform	general
MVGRL [149]	cross-scale	diffusion	general
JOAO [41]	graph	adaptive to loss	general
GCGM [150]	node	adaptive to loss	general
You *et al.* [151]	graph	view generator	general
GCA [152]	node	adaptive to instance	general
HeCo [153]	node	uniform	hetero.
CPT-HG [154]	cross-scale	uniform	hetero.
PT-HGNN [155]	cross-scale	uniform	hetero.
SelfRGNN [76]	node	curvature over time	dynamic
DDGCL [156]	graph	uniform	dynamic
CPDG [75]	cross-scale	temporal-aware sampling	dynamic
GearNet [157]	graph	uniform	3D

Generative Strategies

- Reconstruct parts of the graph

➢Structure reconstruction
- Entire graph structure
- Part of graph structure

➢Feature reconstruction
- Origin feature
- Latent embedding

Method	**Reconstruction objective**						**Graph type**
	node feat.	node deg.	edge	adj. matrix	graph feat.	other info.	
VGAE [43]	×	×	×	✓	×	×	general
GPT-GNN [39]	✓	×	✓	×	×	×	general
MaskGAE [77]	×	✓	✓	×	×	×	general
NWR-GAE [161]	✓	✓	×	×	×	×	general
LaGraph [162]	✓	×	×	×	✓	×	general
GraphMAE [163]	✓	×	×	×	×	×	general
GraphMAE2 [78]	✓	×	×	×	×	×	general
Liu *et al.* [164]	✓	×	×	×	×	×	KG
Wen *et al.* [79]	✓	×	✓	×	×	×	KG
MPKG [165]	✓	×	✓	×	×	✓	KG
PT-DGNN [166]	×	×	✓	×	×	×	dynamic
STEP [167]	✓	×	×	×	×	×	dynamic
PMGT [168]	✓	×	✓	×	×	✓	MMG
ColdGPT [169]	✓	×	×	×	×	✓	MMG

Fine-tuning

- Prior knowledge are transferred to downstream tasks by initializing a downstream model with the pre-trained weights

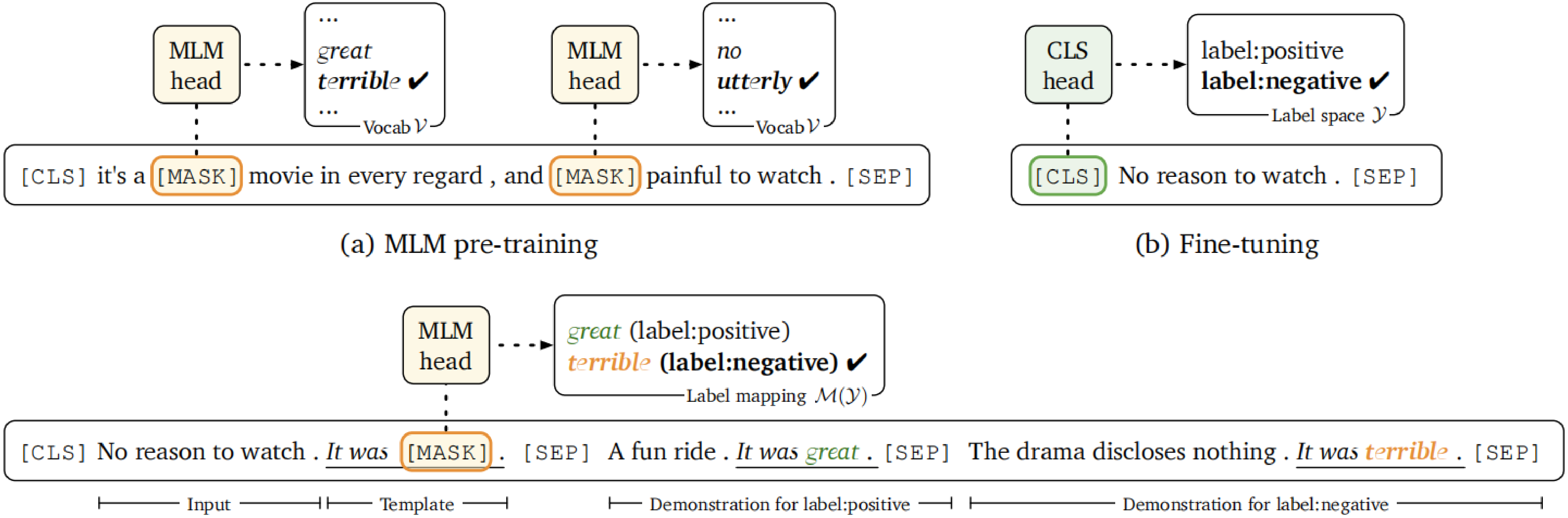
➢ Task-specific projection head

➢ Objective gap between pretext and downstream tasks

➢ Update the parameters in
  - Pretrained model
  - Task head

➢ Updating all parameters is inefficient

Prompt tuning



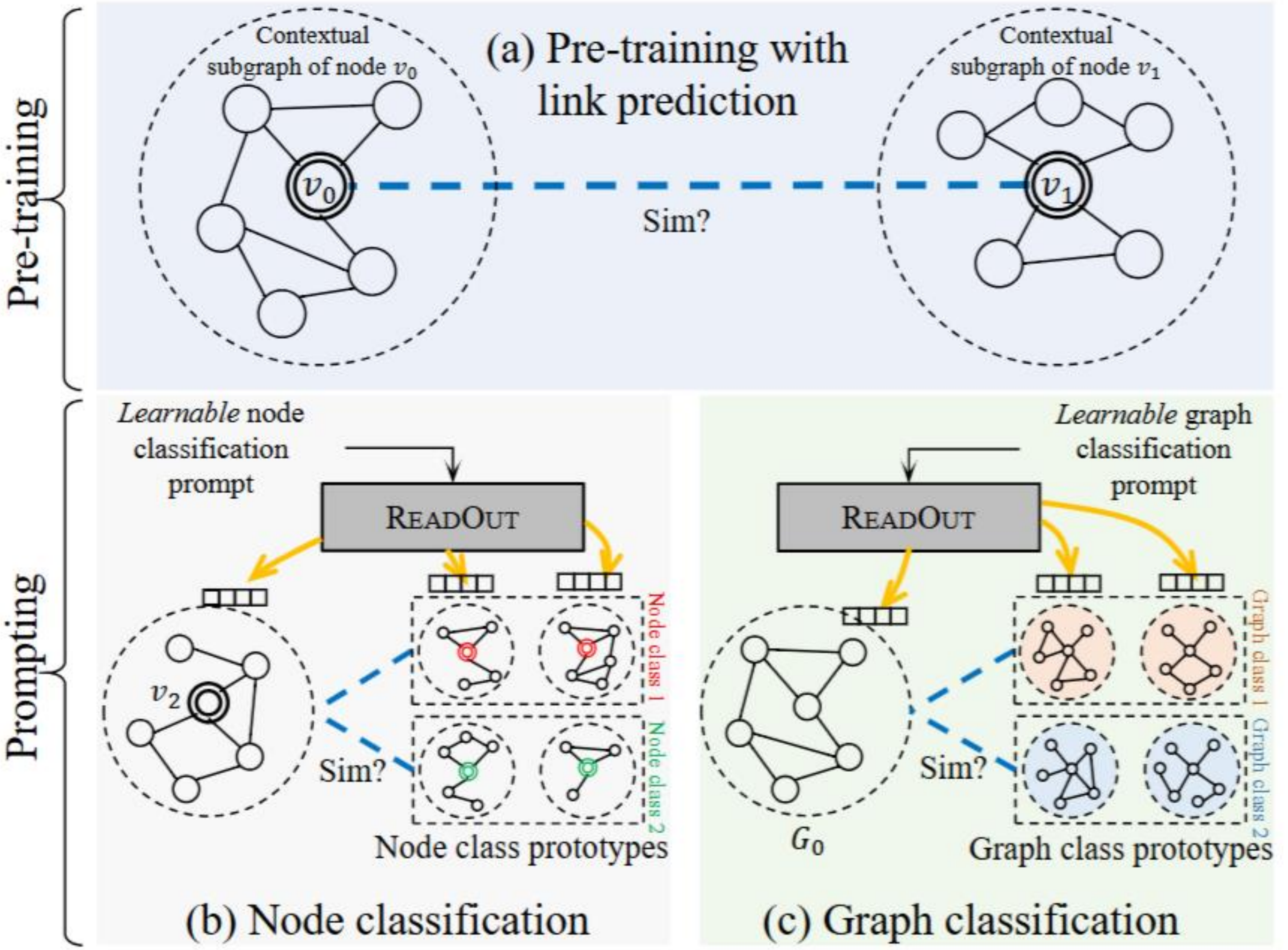
Prompt tuning

- Unified template
  - Aligns the pretext and downstream losses
- Prompt
  - Modify the original input/embedding for the pre-trained model

Paper	Template	Feature prompt	Structure prompt	Multiple pretext tasks	Prompt Initialization	Downstream Task Node	Downstream Task Edge	Downstream Task Graph
GPPT [48]	subgraph-token similarity: $\mathrm{sim}(\mathbf{s}_v, \mathbf{t}_y)$	input	×	×	random	✓	×	×
VPGNN [178]	node-token matching: $\mathrm{match}(\mathbf{h}_v, \mathbf{t}_y)$	×	✓	×	random	✓	×	×
GraphPrompt [19]	subgraph similarity: $\mathrm{sim}(\mathbf{s}_u, \mathbf{s}_v)$	readout	×	×	random	✓	✓	✓
MOP [179]		readout	×	×	random	×	✓	×
GraphPrompt+ [80]		all layers	×	×	random	✓	✓	✓
ProNoG [180]		readout	×	×	conditional	✓	✓	✓
MDGPT [181]	node similarity: $\mathrm{sim}(\mathbf{h}_u, \mathbf{h}_v)$	readout	×	×	pretext tokens	✓	✓	✓
MultiGPrompt [84]		all layers	×	✓	pretext tokens	✓	✓	✓
HetGPT [116]		input	×	×	random	✓	×	×
GPF [182]	universal feature/spectral space	input	×	×	random	✓	✓	✓
IGAP [115]		signal	×	×	random	✓	×	✓
SGL [90]	dual-template: $\mathrm{CL}(\mathbf{h}_u, \mathbf{h}_v), \mathrm{GL}(\mathbf{x}_v, \tilde{\mathbf{x}}_v)$	×	✓	✓	random	×	×	✓
HGPrompt [83]	dual-template: $\mathrm{sim}(\mathbf{s}_u, \mathbf{s}_v)$, graph template	readout	×	×	random	✓	✓	✓
SAP [85]	view similarity: $\mathrm{sim}(\mathrm{MLP}(X), \mathrm{GNN}(X, A))$	×	✓	✓	random	✓	×	✓
ULTRA-DP [86]	node-node/group similarity: $\mathrm{sim}(\mathbf{h}_u, \mathbf{h}_v)$	input	✓	✓	random	✓	✓	×
VNT [87]	node attribute reconstruction: $\mathrm{MSE}(\mathbf{x}_v, \tilde{\mathbf{x}}_v)$ structure recovery: $\mathrm{MSE}(\{\mathbf{h}_u, \mathbf{h}_v\})$	input	×	×	meta-trained	✓	×	×
ProG [49]	subgraph classification: $\mathrm{CLS}(\mathbf{s})$	×	✓	×	meta-trained	✓	✓	✓
DyGPrompt [183]	temporal node similarity: $\mathrm{sim}(\mathbf{h}_{t,u}, \mathbf{h}_{t,v})$	input	×	×	conditional	✓	✓	×
TIGPrompt [184]		input	×	×	time-based	✓	✓	×

GraphPrompt

- Motivation
  - ➢Gap between graph pre-training and downstream tasks
- Challenges
  - ➢What is the unified task template?
  - ➢How to design task-specific prompts?



Liu, et al. "Graphprompt: Unifying pre-training and downstream tasks for graph neural networks." WWW'23.

GraphPrompt

Unified task template

Link Prediction

$$\mathrm{sim}(\mathbf{s}_v, \mathbf{s}_a) > \mathrm{sim}(\mathbf{s}_v, \mathbf{s}_b)$$

Node Classification(NC)

$$\tilde{\mathbf{s}}_c = \frac{1}{k} \sum_{(v_i, \ell_i) \in D, \ell_i = c} \mathbf{s}_{v_i}$$

$$\ell_j = \arg\max_{c \in C} \mathrm{sim}(\mathbf{s}_{v_j}, \tilde{\mathbf{s}}_c)$$

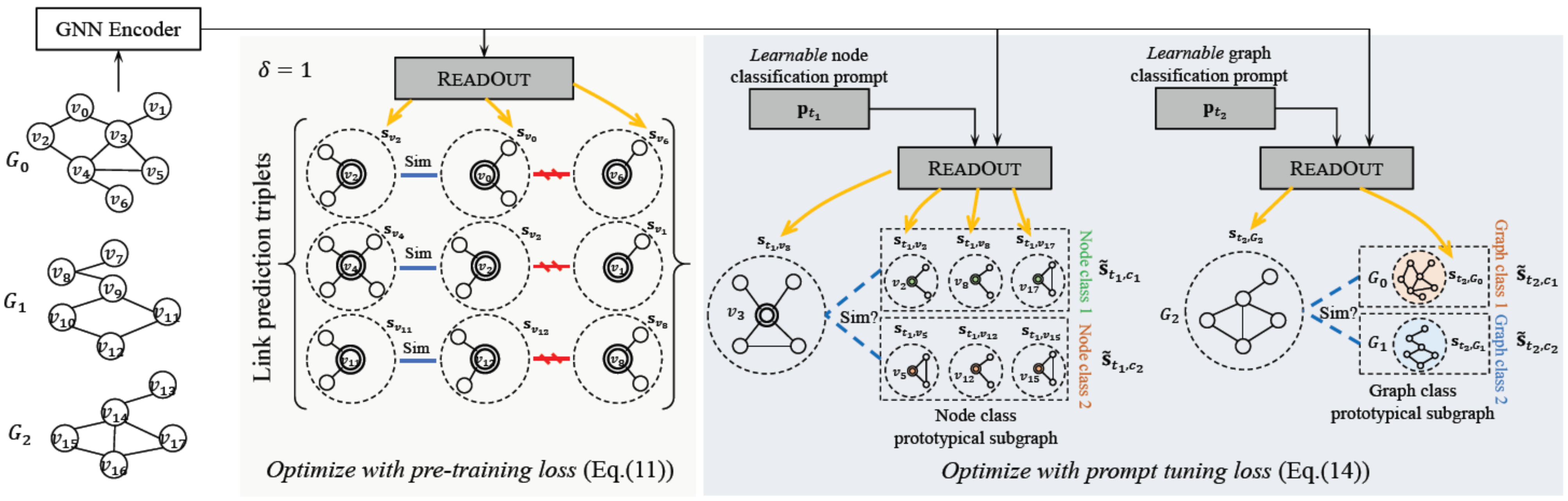
Graph Classification(GC)

$$\tilde{\mathbf{s}}_c = \frac{1}{k} \sum_{(G_i, L_i) \in \mathcal{D}, L_i = c} \mathbf{s}_{G_i}$$

$$L_j = \arg\max_{c \in C} \mathrm{sim}(\mathbf{s}_{G_j}, \tilde{\mathbf{s}}_c)$$

mean embedding of (sub)graphs

class label



(a) Toy graphs (b) Pre-training (c) Prompting for node classification (left) or graph classification (right)

Figure 2: Overall framework of GraphPrompt.

A Notation for NC and GC

$$y = \arg\max_{c \in Y} \mathrm{sim}(\mathbf{s}_x, \tilde{\mathbf{s}}_c)$$

$$\mathbf{s}_x = \mathrm{ReadOut}(\{\mathbf{h}_v : v \in V(S_x)\})$$

Pre-Training Objective

$$\mathcal{L}_{\mathrm{pre}}(\Theta) = -\sum_{(v,a,b) \in \mathcal{T}_{\mathrm{pre}}} \ln \frac{\exp(\mathrm{sim}(\mathbf{s}_v, \mathbf{s}_a)/\tau)}{\sum_{u \in \{a,b\}} \exp(\mathrm{sim}(\mathbf{s}_v, \mathbf{s}_u)/\tau)}$$

Prompt Design

$$\mathbf{s}_{t,x} = \mathrm{ReadOut}(\{\mathbf{p}_t \odot \mathbf{h}_v : v \in V(S_x)\})$$

Liu, et al. "Graphprompt: Unifying pre-training and downstream tasks for graph neural networks." WWW'23.

Generalized Graph Prompt

- Motivation

  - Can more advanced pretext tasks be unified under the subgraph similarity calculation template?

  - How to utilize hierarchical knowledge across multiple layers of the pre-trained graph encoders

Yu, et al. "Generalized graph prompt: Toward a unification of pre-training and downstream tasks on graphs". TKDE 2024.

Generalized Graph Prompt

- Any standard contrastive pretext task on graphs can be unified under the loss:

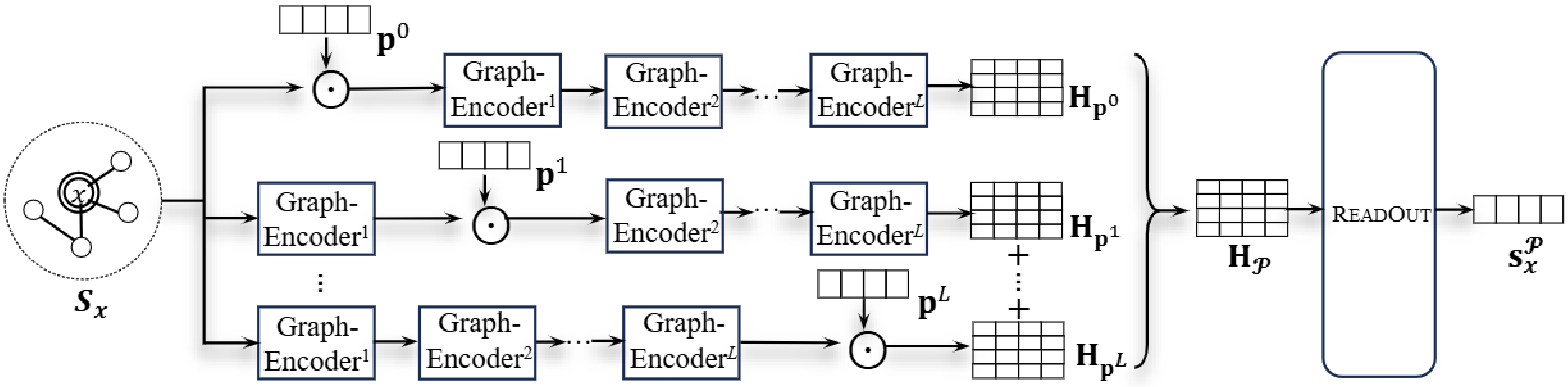
$$\mathcal{L}(\Theta) = -\sum_{o \in \mathcal{T}_{\text{pre}}} \ln \frac{\sum_{a \in Pos_o} \exp(\text{sim}(\mathbf{s}_a, \mathbf{s}_o)/\tau)}{\sum_{b \in Neg_o} \exp(\text{sim}(\mathbf{s}_b, \mathbf{s}_o)/\tau)}$$

	Target instance o	Positive instance a	Negative instance b
LP [39]	a node v	a node linked to v	a node not linked to v
DGI [34]	a graph G	a node in G	a node in G', a corrupted graph of G
InfoGraph [36]	a graph G	a node in G	a node in $G' \neq G$
GraphCL [35]	an augmented graph G_i from a graph G by strategy i	an augmented graph G_j from a graph G by strategy j	an augmented graph G'_j from a graph $G' \neq G$ by strategy j
GCC [22]	a random walk induced subgraph G_v^r from a node v's r-egonet	a random walk induced subgraph $\tilde{G}_v^r \neq G_v^r$ from v's r-egonet	a random walk induced subgraph $G_v^{r'}$ from v's r'-egonet, $r' \neq r$

Yu, et al. "Generalized graph prompt: Toward a unification of pre-training and downstream tasks on graphs". TKDE 2024.

Generalized Graph Prompt

- Layer wise prompt design



Prompts

$$\mathcal{P} = \{\mathbf{p}^0, \mathbf{p}^1, \dots, \mathbf{p}^L\}$$

Layer-wise modification

$$\mathbf{H}_{\mathbf{p}} = \text{GRAPHENCODER}_{\mathbf{p}}(\mathbf{X}, \mathbf{A}; \Theta)$$

$$\mathbf{H}^{l+1} = \text{AGGR}(\mathbf{p}^l \odot \mathbf{H}^l, \mathbf{A}; \theta^{l+1})$$

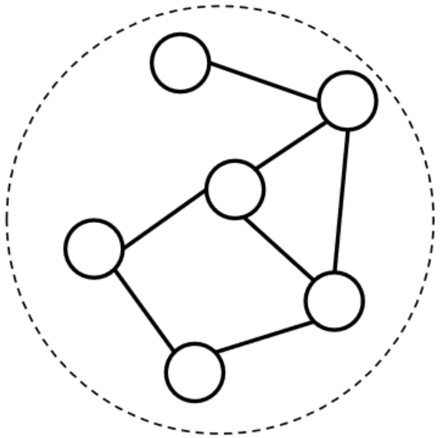
Fusion

$$\mathbf{H}_{\mathcal{P}} = \textstyle\sum_{l=0}^{L} w^l \mathbf{H}_{\mathbf{p}^l}$$

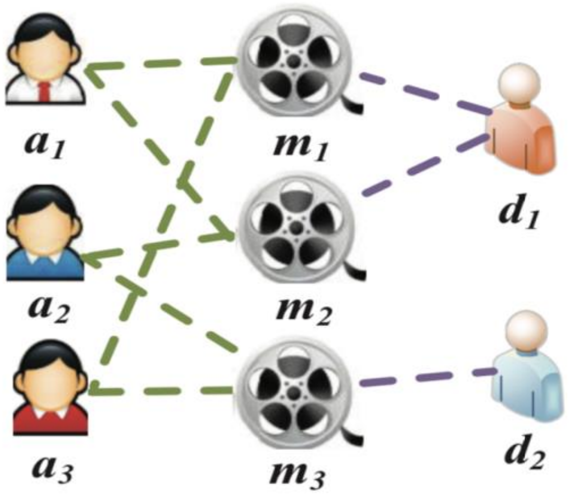
Yu, et al. “Generalized graph prompt: Toward a unification of pre-training and downstream tasks on graphs”. TKDE 2024.

HGPrompt

- Motivation
  - ➢How to unify homogeneous graphs and heterogeneous graphs?
  - ➢How to transfer task-specific heterogeneous knowledge?



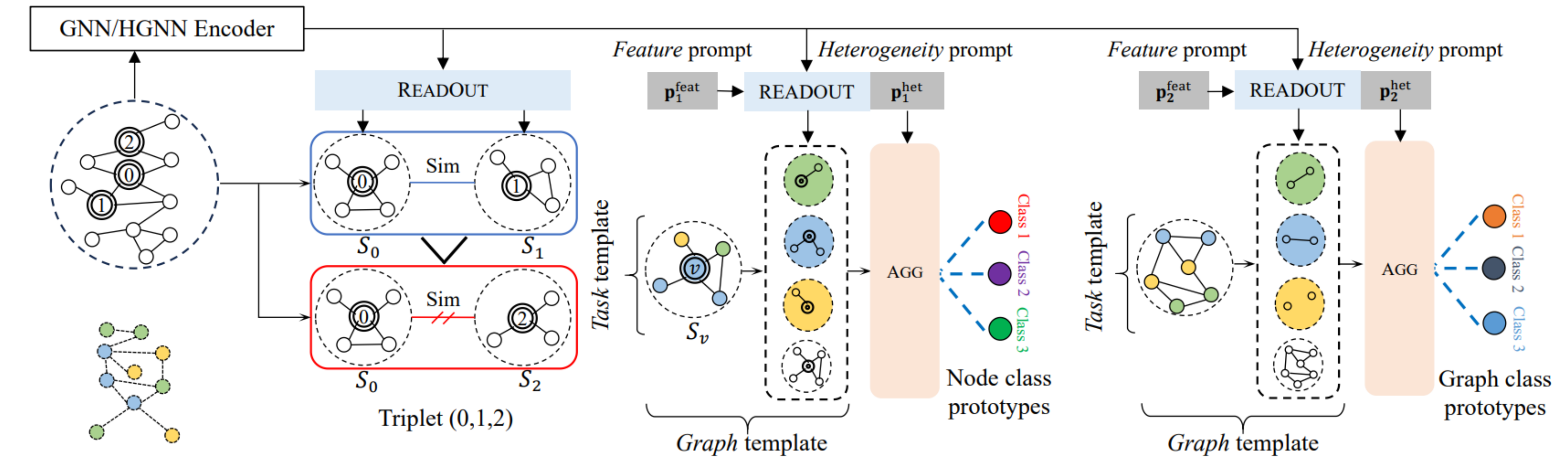
Homogeneous graph



Heterogeneous graph

Yu, et al. “HGPrompt: Bridging Homogeneous and Heterogeneous Graphs for Few-shot Prompt Learning.” AAAI’24.

HGPrompt



(a) Pre-training graphs (b) Pre-training (link prediction) (c) Downstream node classification (d) Downstream graph classification

Dual templates

Task template

$$\mathcal{GT}(G) = \{G^0\} \cup \{G^i : i \in A\}$$

Graph template

$$\mathrm{sim}(\mathbf{s}_v, \mathbf{s}_a) > \mathrm{sim}(\mathbf{s}_v, \mathbf{s}_b)$$

$$\ell_j = \arg\max_{c \in C} \mathrm{sim}(\mathbf{s}_{v_j}, \tilde{\mathbf{s}}_c)$$

$$L_j = \arg\max_{c \in \mathcal{C}} \mathrm{sim}(\mathbf{s}_{G_j}, \tilde{\mathbf{s}}_c)$$

Dual prompts

Feature prompt

$$\mathrm{READOUT}(\{\mathbf{p}^{\text{feat}} \odot \mathbf{h}_v \mid v \in V(S)\})$$

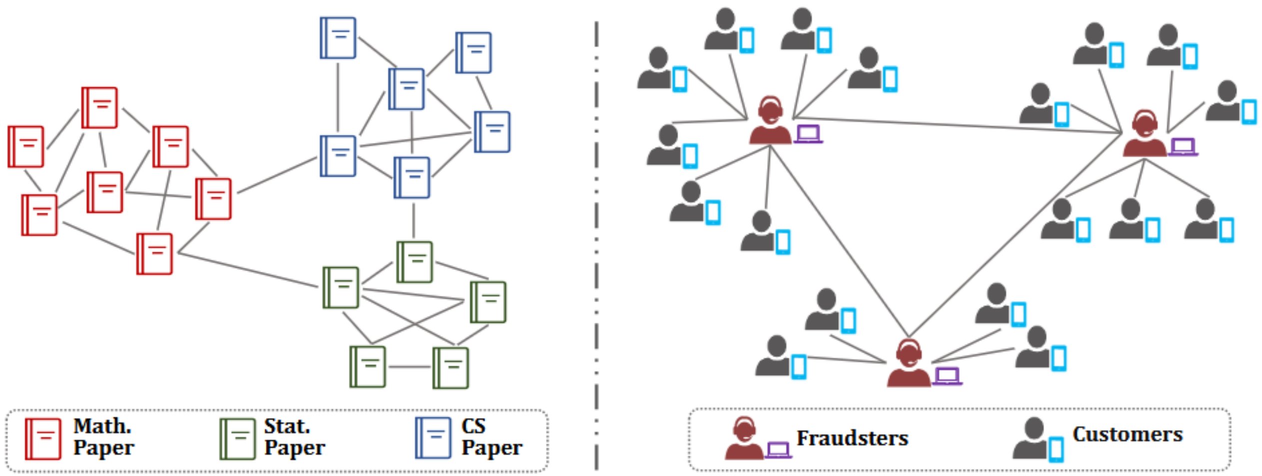
Heterogeneity prompt

$$\mathrm{AGG}(\{(1 + p_i^{\text{het}}) \odot \mathrm{READOUT}(S^i) \mid S^i \in \mathcal{GT}(S)\})$$

Yu, et al. “HGPrompt: Bridging Homogeneous and Heterogeneous Graphs for Few-shot Prompt Learning.” AAAI’24.

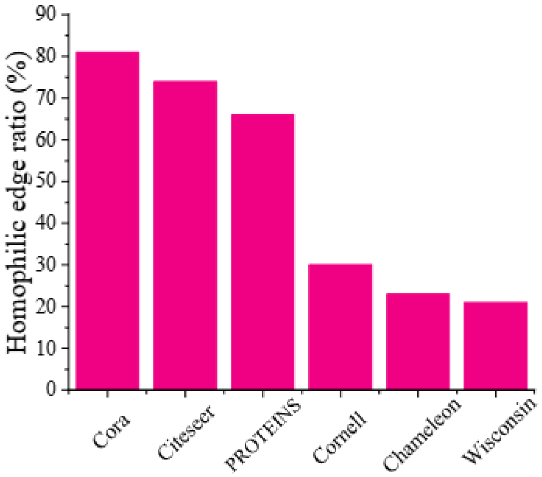
ProNoG

- Motivation
  - ➢ Graphs exhibit different homophily ratio depending on nodes label
  - ➢ How to capture node specific homophily pattern?

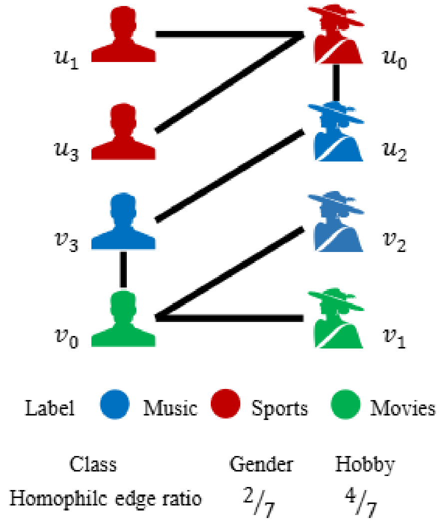


Homophily graph

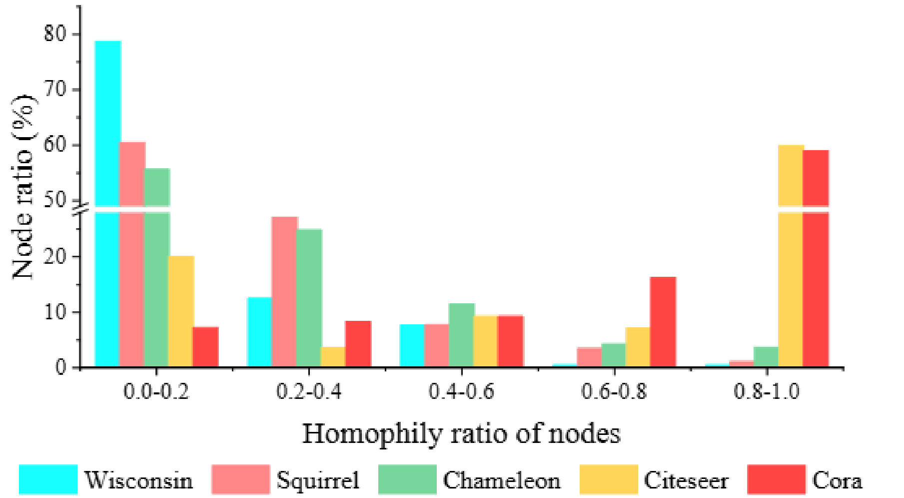
Heterophily graph



(a) Varying non-homophilic patterns across different graphs



(b) Dependence of homophiliy ratio on the target label



(c) Diverse non-homophilic patterns across nodes in the same graph

Yu, et al. “Non-homophilic graph pre-training and prompt learning.” SIGKDD’25.

ProNoG

Contrastive pre-training method loss function

$$\mathcal{L}_T = -\sum_{u \in V} \ln P(u, \mathcal{A}_u, \mathcal{B}_u), \tag{4}$$

$$P(u, \mathcal{A}_u, \mathcal{B}_u) \triangleq \frac{\sum_{a \in \mathcal{A}_u} \mathtt{sim}(\mathbf{h}_u, \mathbf{h}_a)}{\sum_{a \in \mathcal{A}_u} \mathtt{sim}(\mathbf{h}_u, \mathbf{h}_a) + \sum_{b \in \mathcal{B}_u} \mathtt{sim}(\mathbf{h}_u, \mathbf{h}_b)}, \tag{5}$$

Theorems

THEOREM 1. *For a homophily task T, adding a homophily sample always results in a smaller loss than adding a non-homophily sample.*

THEOREM 2. *Consider a graph $G = (V, E)$ with a label mapping function $V \to Y$, and let $y_v \in Y$ denote the label mapped to $v \in V$. Suppose the label mapping satisfies that*

$$\forall u, a, b \in V, y_u = y_a \wedge y_u \neq y_b \Rightarrow \mathtt{sim}(u, a) > \mathtt{sim}(u, b).$$

Let $\mathbb{E}_T$ denote the expected number of homophily samples for a homophily task T on the graph G. Then, $\mathbb{E}_T$ increases monotonically as the homophily ratio $\mathcal{H}(G)$ defined w.r.t. Y increases.

Definition of homophily task

DEFINITION 1 (HOMOPHILY TASK). *On a graph $G = (V, E)$, a pre-training task $T = (\{\mathcal{A}_u : u \in V\}, \{\mathcal{B}_u : u \in V\})$ is a homophily task if and only if, $\forall u \in V, \forall a \in \mathcal{A}_u, \forall b \in \mathcal{B}_u, (u, a) \in E \wedge (u, b) \notin E$. A task that is not a homophily task is called a non-homophily task.* □

Insights

For non-homophilic graphs, especially those with low homophily ratio, non-homophily tasks are a better choice compared to homophily tasks when optimizing the training loss.

Table 6: Positive and negative samples for homophily and non-homophily methods.

Pre-training task	Positive instances $\mathcal{A}_u$	Negative instances $\mathcal{B}_u$	Homophily task
Link prediction [26, 62, 64]	a node connected to node u	nodes disconnected to node u	Yes
DGI [48]	nodes in graph G	nodes in corrupted graph G'	No
GraphCL [60]	an augmented graph from graph G	augmented graphs from $G' \neq G$	No
GraphACL [55]	nodes with similar ego-subgraph to node u	nodes with dissimilar ego-subgraph to node u	No

Yu, et al. "Non-homophilic graph pre-training and prompt learning." SIGKDD'25.

ProNoG

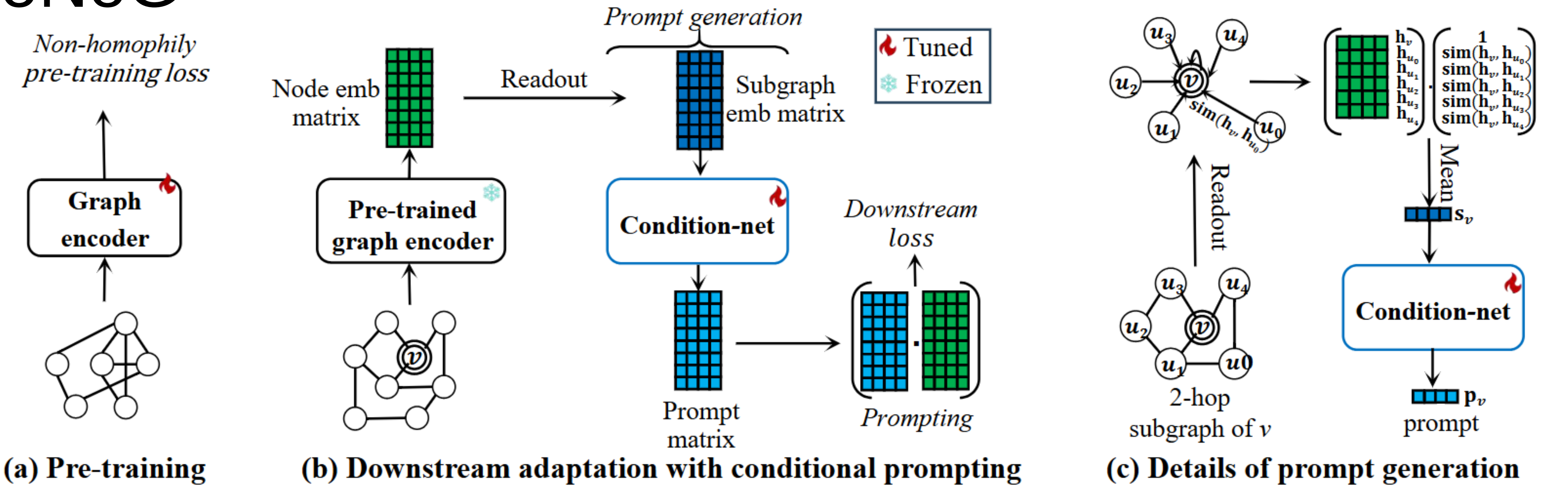


Figure 2: Overall framework of ProNoG.

Prompt generation

$$\mathbf{s}_v = \frac{1}{|S_v|} \sum_{u \in S_v} \mathbf{h}_u \cdot \mathtt{sim}(\mathbf{h}_u, \mathbf{h}_v),$$

$$\mathbf{p}_{t,v} = \mathtt{CondNet}(\mathbf{s}_v; \phi_t),$$

Prompt tuning

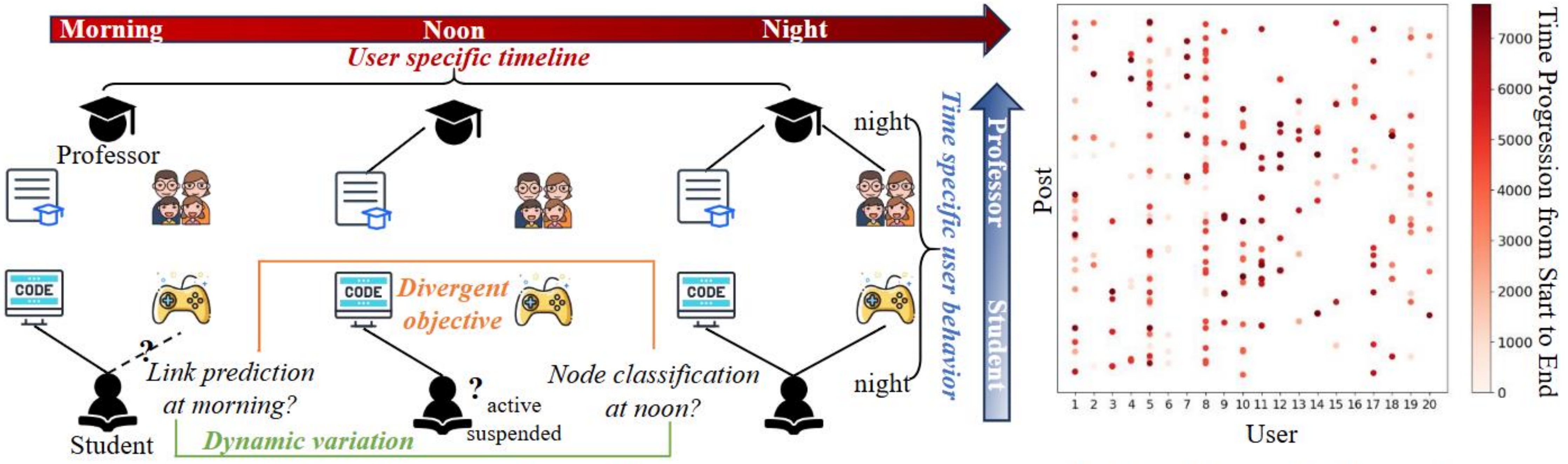
$$\tilde{\mathbf{h}}_{t,v} = \mathbf{p}_{t,v} \odot \mathbf{h}_v,$$

$$\mathcal{L}_{\text{down}}(\phi_t) = -\sum_{(x_i, y_i) \in \mathcal{D}_t} \ln \frac{\exp\left(\frac{1}{\tau}\mathrm{sim}(\tilde{\mathbf{h}}_{t,x_i}, \bar{\mathbf{h}}_{t,y_i})\right)}{\sum_{c \in Y} \exp\left(\frac{1}{\tau}\mathrm{sim}(\tilde{\mathbf{h}}_{t,x_i}, \bar{\mathbf{h}}_{t,c})\right)},$$

Yu, et al. "Non-homophilic graph pre-training and prompt learning." SIGKDD'25.

DyGPrompt

- Motivation
  - ➢ How to design bridge temporal variations across time and different task objectives
  - ➢ How to capture evolving patterns across different nodes and time points

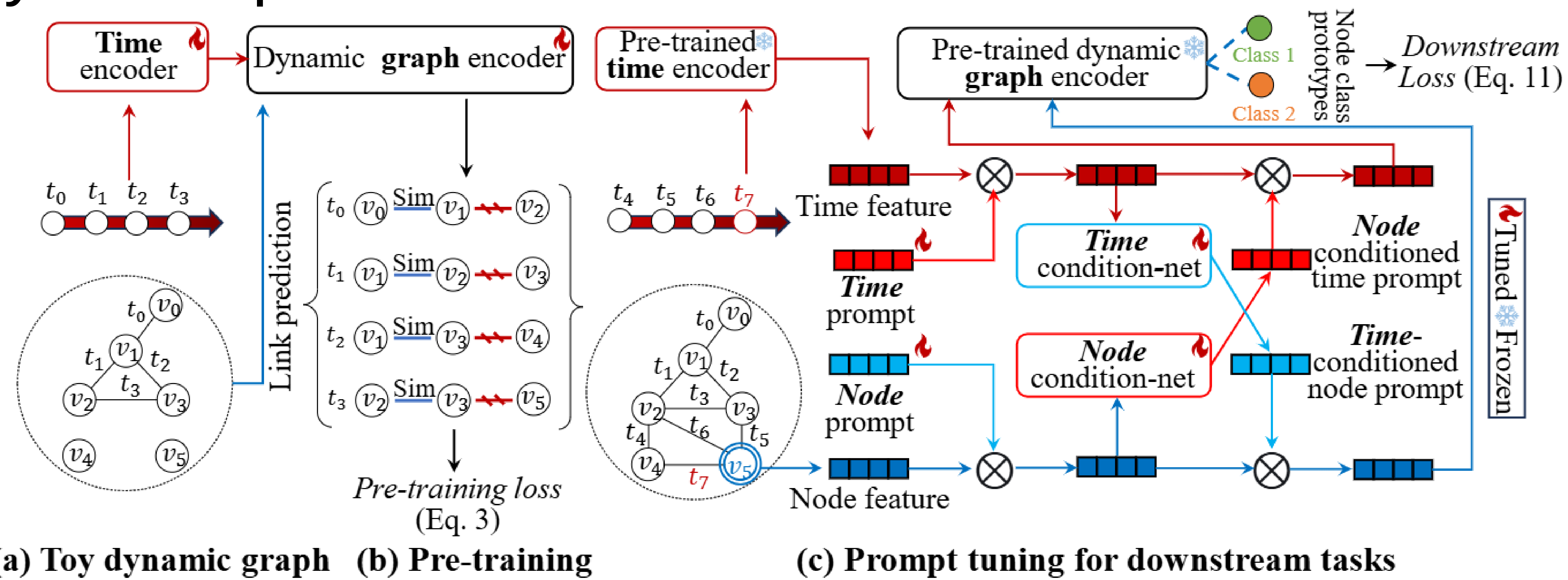


(a) Users' comments over time

(b) Envolving node-time patterns

Yu, et al. "Node-Time Conditional Prompt Learning In Dynamic Graphs." ICLR'25.

DyGPrompt



Dual prompts

Node prompt

$$\mathbf{x}_{t,v}^{\text{node}} = \mathbf{p}^{\text{node}} \odot \mathbf{x}_{t,v}$$

Time prompt

$$\mathbf{f}_t^{\text{time}} = \mathbf{p}^{\text{time}} \odot \mathbf{f}_t$$

Dual condition prompts

Time conditioned node prompts

$$\tilde{\mathbf{p}}_t^{\text{node}} = \mathtt{TCN}(\mathbf{f}_t^{\text{time}}; \kappa)$$

$$\tilde{\mathbf{x}}_{t,v}^{\text{node}} = \tilde{\mathbf{p}}_t^{\text{node}} \odot \mathbf{x}_{t,v}^{\text{node}}$$

Node conditioned time prompts

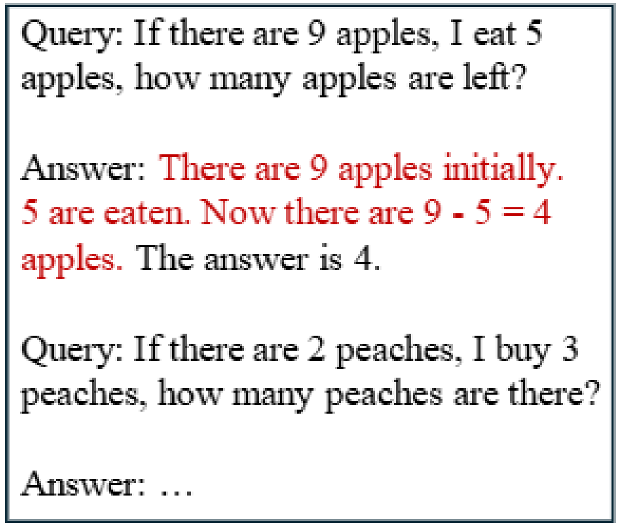
$$\tilde{\mathbf{p}}_{t,v}^{\text{time}} = \mathtt{NCN}(\mathbf{x}_{t,v}^{\text{node}}; \phi)$$

$$\tilde{\mathbf{f}}_{t,v}^{\text{time}} = \tilde{\mathbf{p}}_{t,v}^{\text{time}} \odot \mathbf{f}_t^{\text{time}}$$

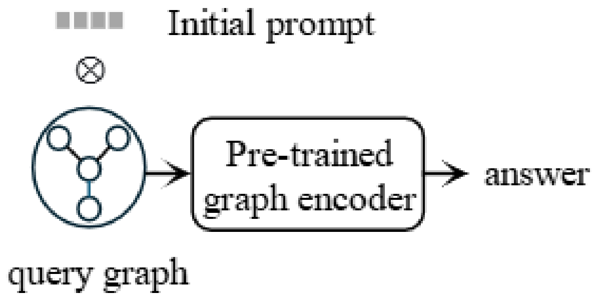
Yu, et al. “Node-Time Conditional Prompt Learning In Dynamic Graphs.” ICLR’25.

GCoT

- Motivation
  - Chain-of-Thought (CoT) mimics the logical process a person may employ to solve a task
  - Current graph models inference in single step
  - *Would introducing additional inference steps in a CoT style enhance the ability of pre-trained graph models to refine their predictions*



(a) Chain-of-Thought prompting in NLP



(b) Standard graph prompting

Yu, et al. "GCoT: Chain-of-Thought Prompt Learning for Graphs." arXiv preprint.

GCoT

- CoT in natural language processing
  - ➢ Handcrafted before the learning phase and typically consists of a structured text in the form ⟨input, chain of thought, output⟩.
  - ➢ Prompt serves as an example to guide the model in generating intermediate thoughts that lead to the final answer.
- CoT for graphs
  - ➢ What should be the inference steps and thoughts for a graph task?
  - ➢ How can we leverage a “thought” to learn prompts and guide the next-step inference

Yu, et al. “GCoT: Chain-of-Thought Prompt Learning for Graphs.” arXiv preprint.

GCoT

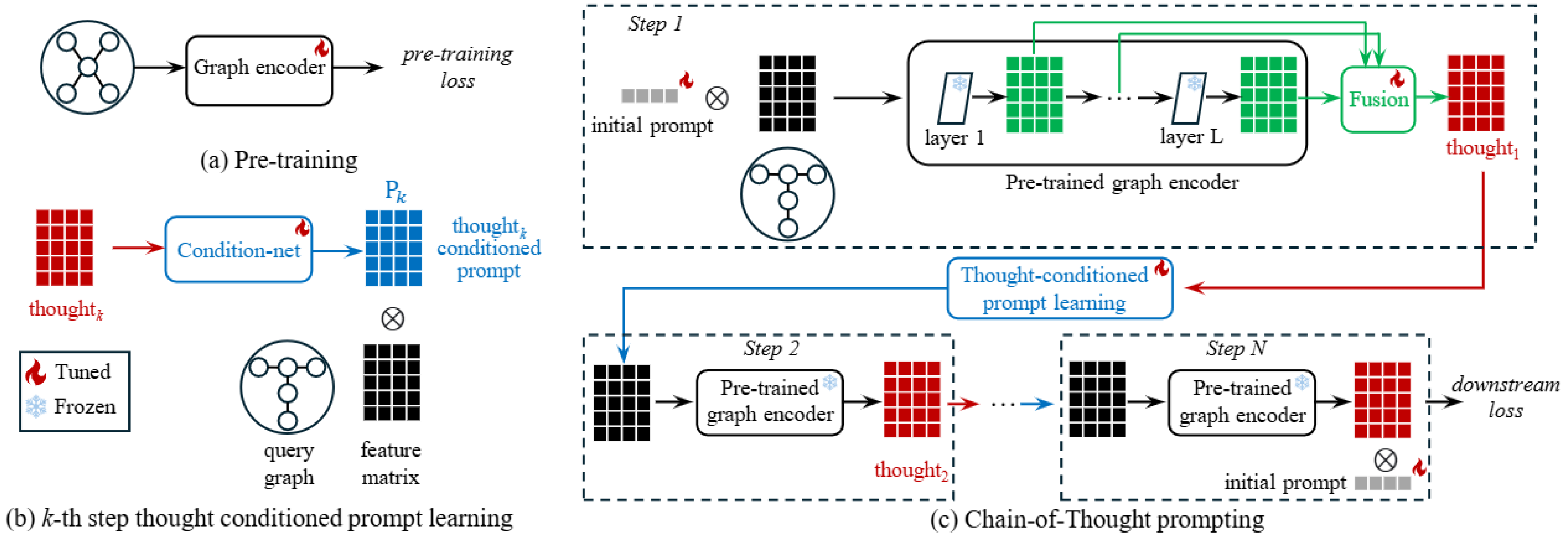


Figure 2: Overall framework of GCoT.

Prompt based inference

$$\{H_k^1, H_k^2, \cdots, H_k^L\} = \textsc{GraphEncoder}(X_k, G; \Theta_0)$$

Thought construction

$$T_k = \mathrm{Fuse}(H_k^1, H_k^2, \cdots, H_k^L),$$

Thought conditioned prompting

$$P_k = \mathrm{CondNet}(T_k; \phi),$$
$$X_{k+1} = P_k \odot X,$$

Yu, et al. "GCoT: Chain-of-Thought Prompt Learning for Graphs." arXiv preprint.

MultiGPrompt

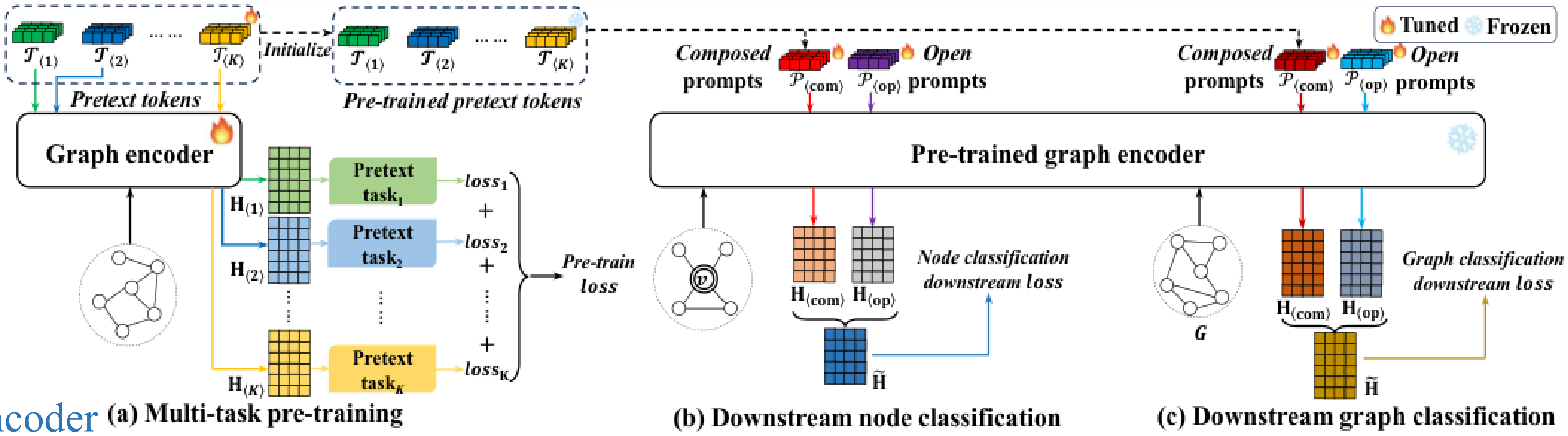
- Motivation

  - ➢ How to leverage diverse pretext tasks for graph models in a synergistic manner?

  - ➢ How to transfer both task specific and global pre-trained knowledge to downstream tasks?

Yu, et al. "MultiGPrompt for Multi-Task Pre-Training and Prompting on Graphs." WWW'24.

MultiGPrompt



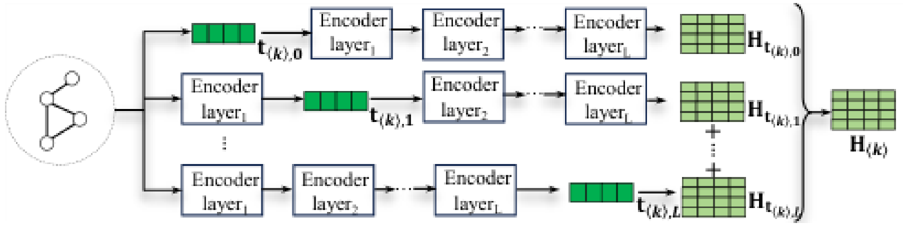
Multi-task pre-training

Pretext tokens

$$\mathcal{T}_{\langle k \rangle} = \{\mathbf{t}_{\langle k \rangle,0}, \mathbf{t}_{\langle k \rangle,1}, \ldots, \mathbf{t}_{\langle k \rangle,L}\}$$

Add token to each layer of graph encoder

$$\mathbf{H}^{l+1} = \mathrm{MP}(\mathbf{t}_{\langle k \rangle,l} \odot \mathbf{H}^l, \mathbf{A}; \theta^l)$$



Graph encoder output embedding

$$\mathbf{H}_{\mathbf{t}} = \textsc{GraphEncoder}_{\mathbf{t}}(\mathbf{X}, \mathbf{A}; \Theta)$$

Overall embedding

$$\mathbf{H}_{\langle k \rangle} = \sum_{l=0}^{L} \alpha_l \mathbf{H}_{\mathbf{t}_{\langle k \rangle,l}}$$

Pre-Training Objective

$$\mathcal{L}_{\text{pre}}(\mathcal{H}; \mathcal{T}, \Theta) = \sum_{k=1}^{K} \beta_k \mathcal{L}_{\text{pre}_{\langle k \rangle}}(\mathbf{H}_{\langle k \rangle}; \mathcal{T}_{\langle k \rangle}, \Theta),$$

Prompt tuning

Composed prompt

$$\mathcal{P}_{\langle \text{com} \rangle} = \{\mathbf{p}_{\langle \text{com} \rangle,0}, \mathbf{p}_{\langle \text{com} \rangle,1}, \ldots, \mathbf{p}_{\langle \text{com} \rangle,L}\}$$

$$\mathbf{p}_{\langle \text{com} \rangle,l} = \textsc{Compose}(\mathbf{t}_{\langle 1 \rangle,l}, \mathbf{t}_{\langle 2 \rangle,l}, \ldots, \mathbf{t}_{\langle K \rangle,l}; \Gamma)$$

Open prompt

$$\mathcal{P}_{\langle \text{op} \rangle} = \{\mathbf{p}_{\langle \text{op} \rangle,0}, \mathbf{p}_{\langle \text{op} \rangle,1}, \ldots, \mathbf{p}_{\langle \text{op} \rangle,L}\}$$

Add prompt to each layer of graph encoder

$$\mathbf{H}_{\mathbf{p}} = \textsc{GraphEncoder}_{\mathbf{p}}(\mathbf{X}, \mathbf{A}; \Theta_{\text{pre}})$$

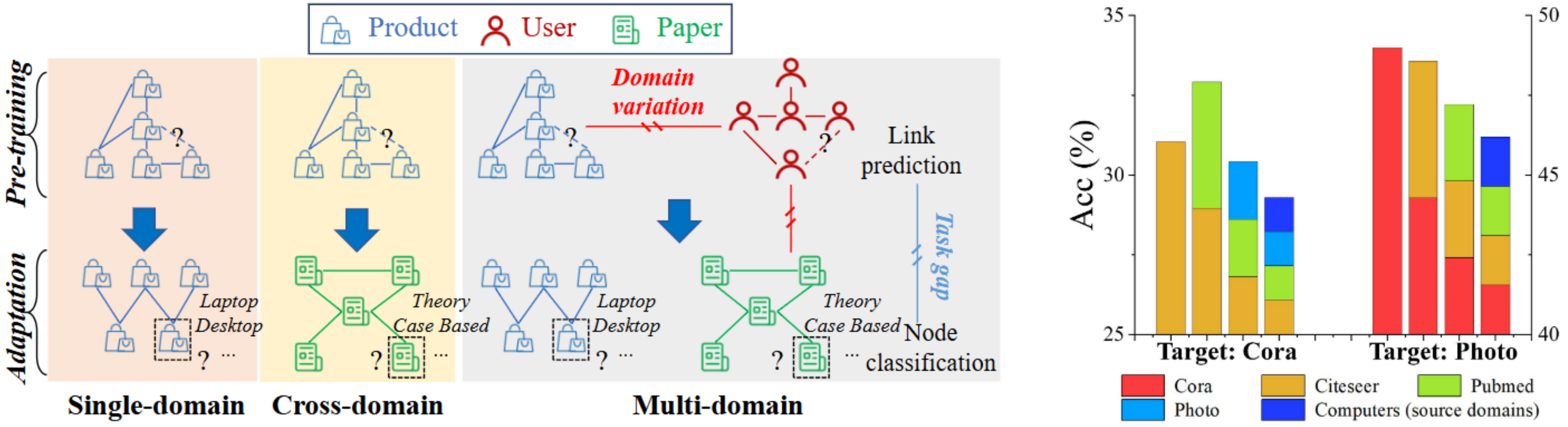
Aggregate dual prompt

$$\tilde{\mathbf{H}} = \textsc{Aggr}(\mathbf{H}_{\langle \text{com} \rangle}, \mathbf{H}_{\langle \text{op} \rangle}; \Delta)$$

Yu, et al. "MultiGPrompt for Multi-Task Pre-Training and Prompting on Graphs." WWW'24.

MDGPT & SAMGPT

- Motivation



(a) Various transfer scenarios **(b) Observation of domain conflicts**

➢ How to align multi-domain graphs in the pre-training phase in both feature and structure level

➢ How to adapt multi-domain prior knowledge to downstream tasks in different domains?

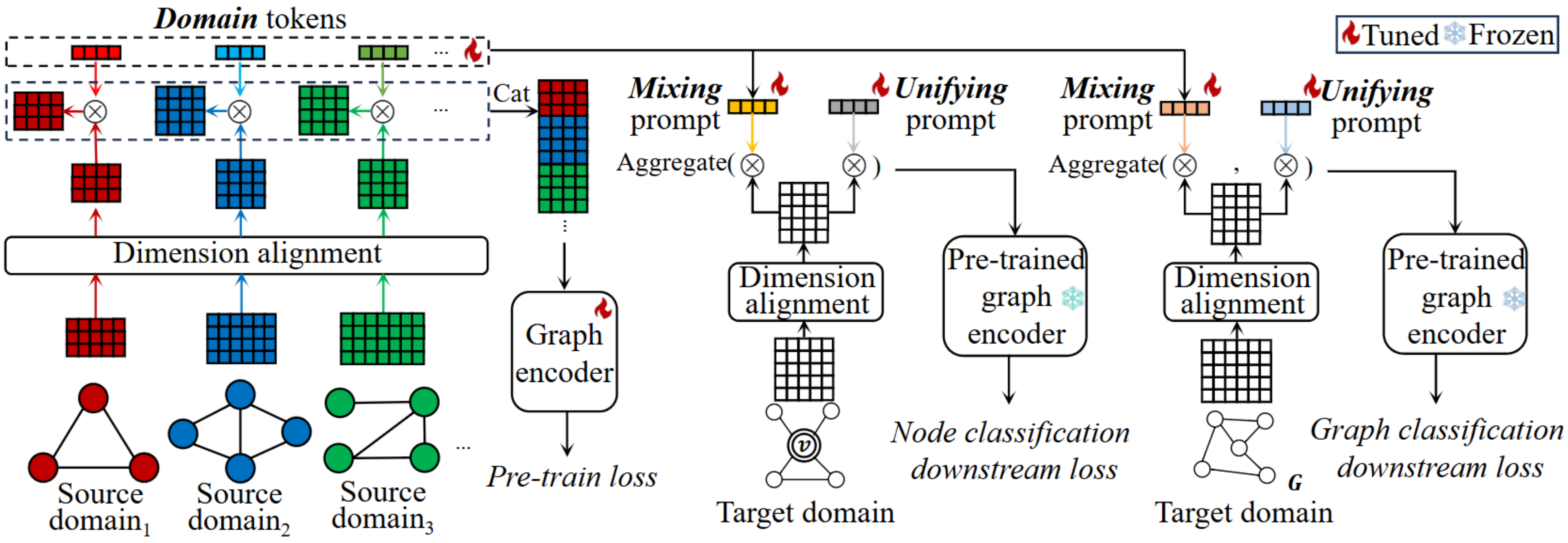
	Nodes	Edges	Feature dimension	Node classes	Avg. nd	Avg. spl	Avg. cc
Cora	2,708	10,556	1,433	7	3.89	6.30	0.24
Citeseer	3,327	9,104	3,703	6	2.73	9.31	0.14
Pubmed	19,717	88,648	500	3	4.49	6.33	0.06
Photo	7,650	238,162	745	8	31.13	4.05	0.40
Computers	13,752	491,722	767	10	35.75	3.38	0.34
Facebook	22,470	342,004	128	4	15.22	4.97	0.35
LastFM	7,624	55,612	128	18	7.29	5.23	0.21

nd: node degree, spl: shortest path length [3], cc: clustering coefficient [8].

Yu, et al. “Text-free multi-domain graph pre-training: Toward graph foundation models.” arXiv preprint.

Yu, et al. “SAMGPT: Text-free Graph Foundation Model for Multi-domain Pre-training and Cross-domain Adaptation.” WWW’25.

MDGPT



Multi-domain pre-training

Downstream adaptation

Dimension alignment

$\tilde{\mathbf{X}}_i = \mathsf{DA}_{S_i}(\mathbf{X}_i)$

$\mathsf{DA}_{S_i} : \mathbb{R}^{|V| \times d_{S_i}} \to \mathbb{R}^{|V| \times \tilde{d}}$

Semantic alignment

$\hat{\mathbf{X}}_i = \mathbf{t}_{S_i} \odot \tilde{\mathbf{X}}_i,$

$\mathbf{H}_S = \mathsf{GE}(\mathcal{G}_S, \mathcal{X}_S; \Theta),$

Dimension alignment

$\tilde{\mathbf{X}} = \mathsf{DA}_T(\mathbf{X})$

Unifying prompt

$\mathbf{p}_{\text{uni}}$

Mixing prompt

$\mathbf{p}_{\text{mix}} = \sum_{i=1}^{K} \gamma_i \mathbf{t}_{S_i}$

$$\mathbf{H} = \mathsf{GE}(G, \mathbf{p}_{\text{uni}} \odot \tilde{\mathbf{X}}; \Theta_{\text{pre}}) + \mathsf{GE}(G, \mathbf{p}_{\text{mix}} \odot \tilde{\mathbf{X}}; \Theta_{\text{pre}}),$$

Yu, et al. “Text-free multi-domain graph pre-training: Toward graph foundation models.” arXiv preprint.

SAMGPT

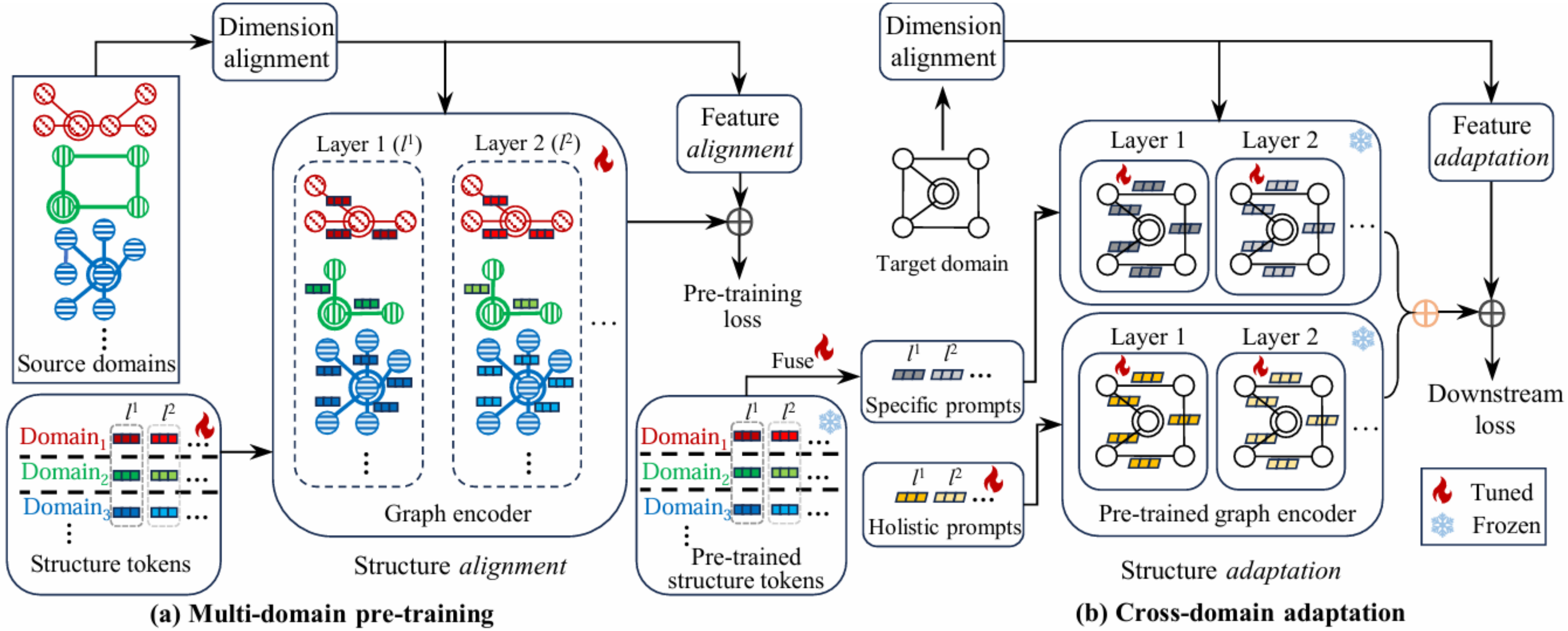


Figure 2: Overall framework of SAMGPT.

Multi-domain pre-training

Downstream adaptation

Structural alignment

$$\mathbf{h}_v^l = \mathtt{Aggr}(\mathbf{h}_v^{l-1}, \{\mathbf{t}_{S_i}^l \odot \mathbf{h}_u^{l-1} : u \in \mathcal{N}_v\}; \theta^l), \ \forall v \in V_i,$$

Holistic prompt

$$\mathbf{h}_v^l = \mathtt{Aggr}(\mathbf{h}_v^{l-1}, \{\mathbf{p}_{\text{hol}}^l \odot \mathbf{h}_u^{l-1} : u \in \mathcal{N}_v\}; \theta_{\text{pre}}^l),$$

Specific prompt

$$\mathbf{p}_{\text{spe}}^l = \sum_{i=1}^{K} \lambda_i^l \mathbf{t}_{S_i}^l,$$

Yu, et al. “SAMGPT: Text-free Graph Foundation Model for Multi-domain Pre-training and Cross-domain Adaptation.” WWW’25.

Summary

- Existing research often focus on text-free graphs, fail to leverage the vast amount of textual data to learn a more comprehensive knowledge
- LLMs have achieved significant performance

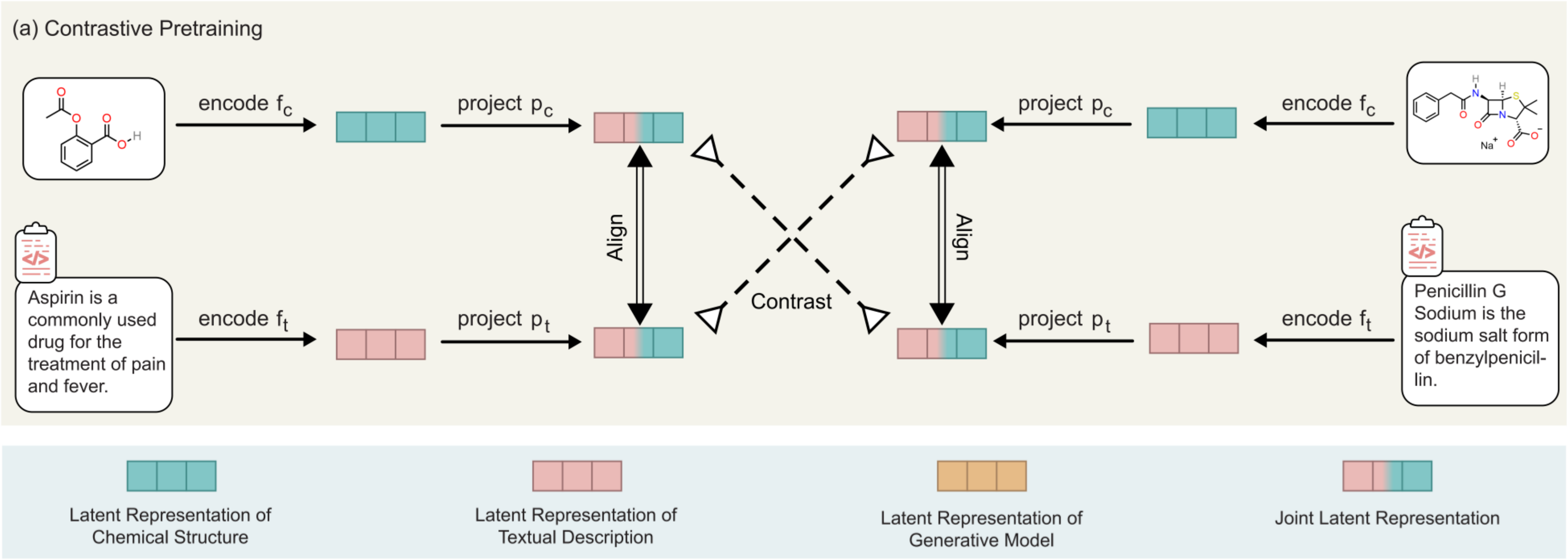
Can we leverage LLMs to integrate textual data and thereby improve the performance of graph few-shot learning?

Graph + PLM

- Pre-training:
  - ➢Contrastive pre-training
  - ➢Language modeling
- Adaptation:
  - ➢Prompt-tuning
  - ➢Parameter-effeicient fine-tuning (PEFT)

Graph + PLM: Pre-training

- MoleculeSTM: Contrastive pre-training

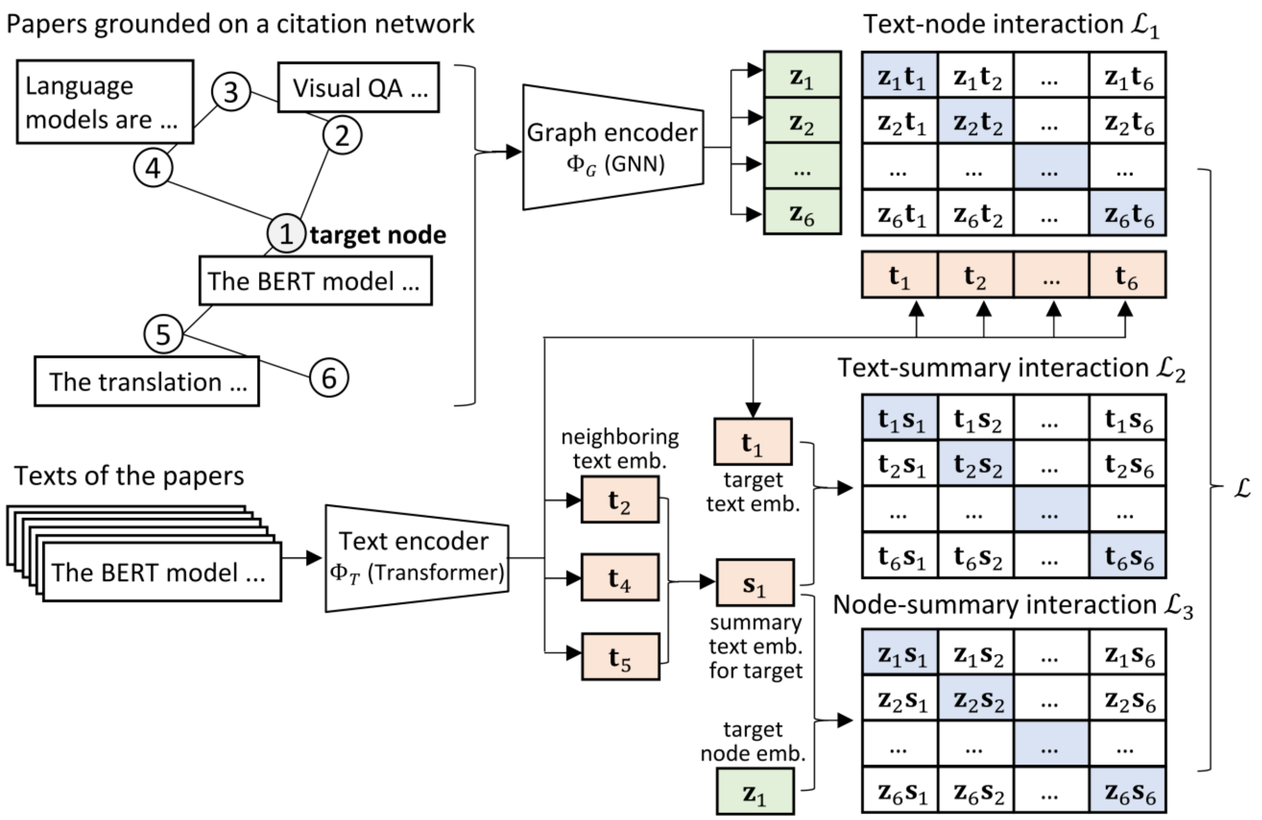


Graph-Text contrastive learning: Graph encoder + Text encoder → projector layers

S. Liu, *et al*. "Multi-modal molecule structure–text model for text-based retrieval and editing." Nature Machine Intelligence 2023

Graph + PLM: Pre-training

- G2P2: Contrastive pre-training



Graph encoder + Text encoder → Three contrastive loss:

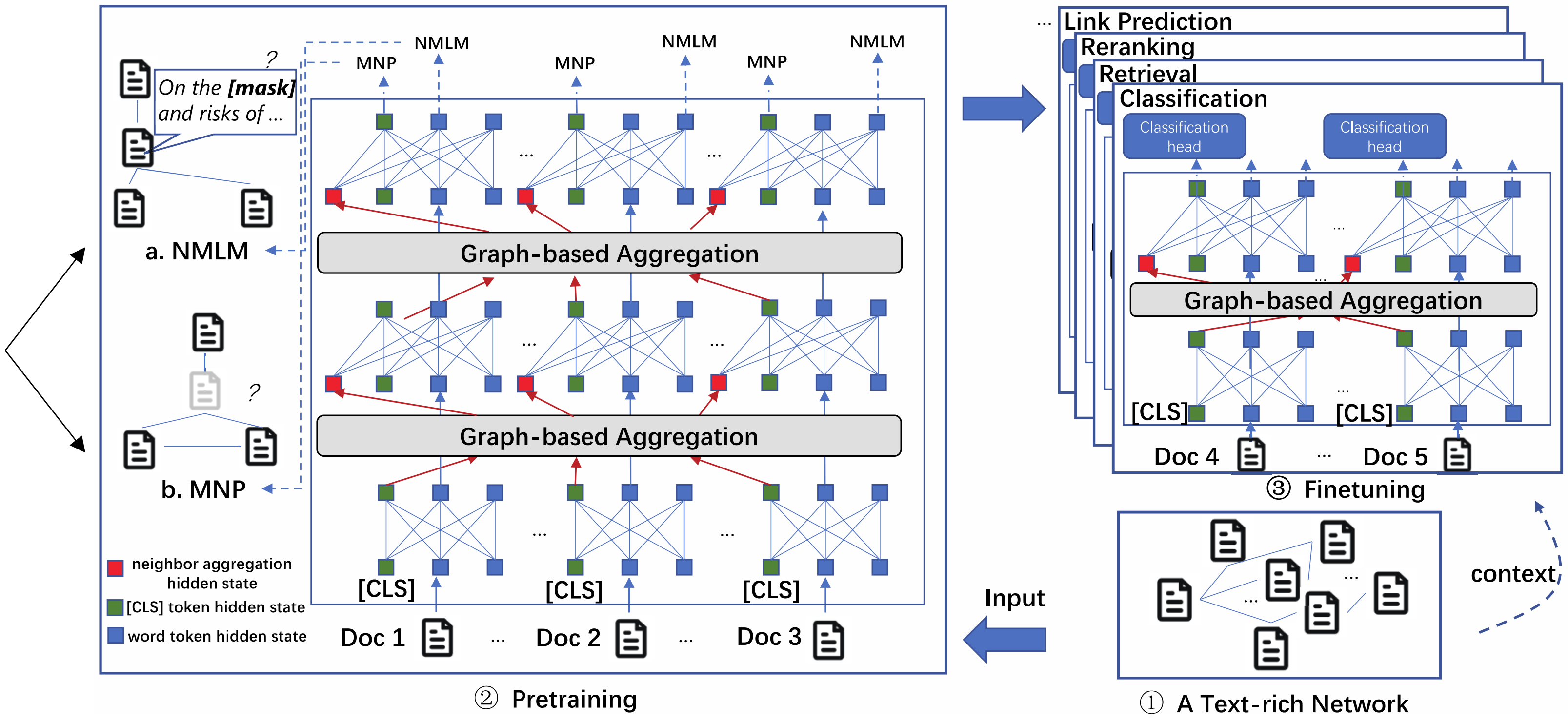
- Text-Node
- Text summary-Text
- Text summary-Node

Z. Wen, *et al*. "Augmenting low-resource text classification with graph-grounded pre-training and prompting." SIGIR'*23*

Graph + PLM: Pre-training

- PATTON: Masked language modeling + Masked node prediction

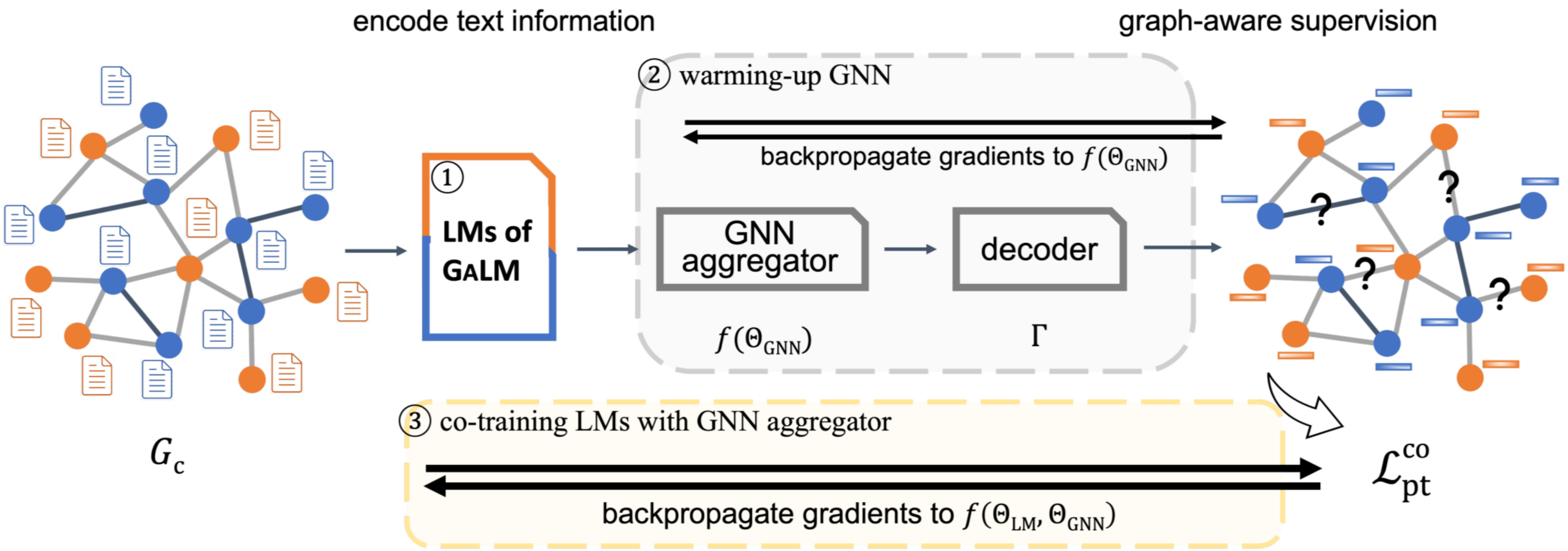
Two pretraining strategies



B. Jin, *et al*. "PATTON : Language Model Pretraining on Text-Rich Networks." ACL'*23*

Graph + PLM: Pre-training

- GaLM: Graph-aware language model pre-training



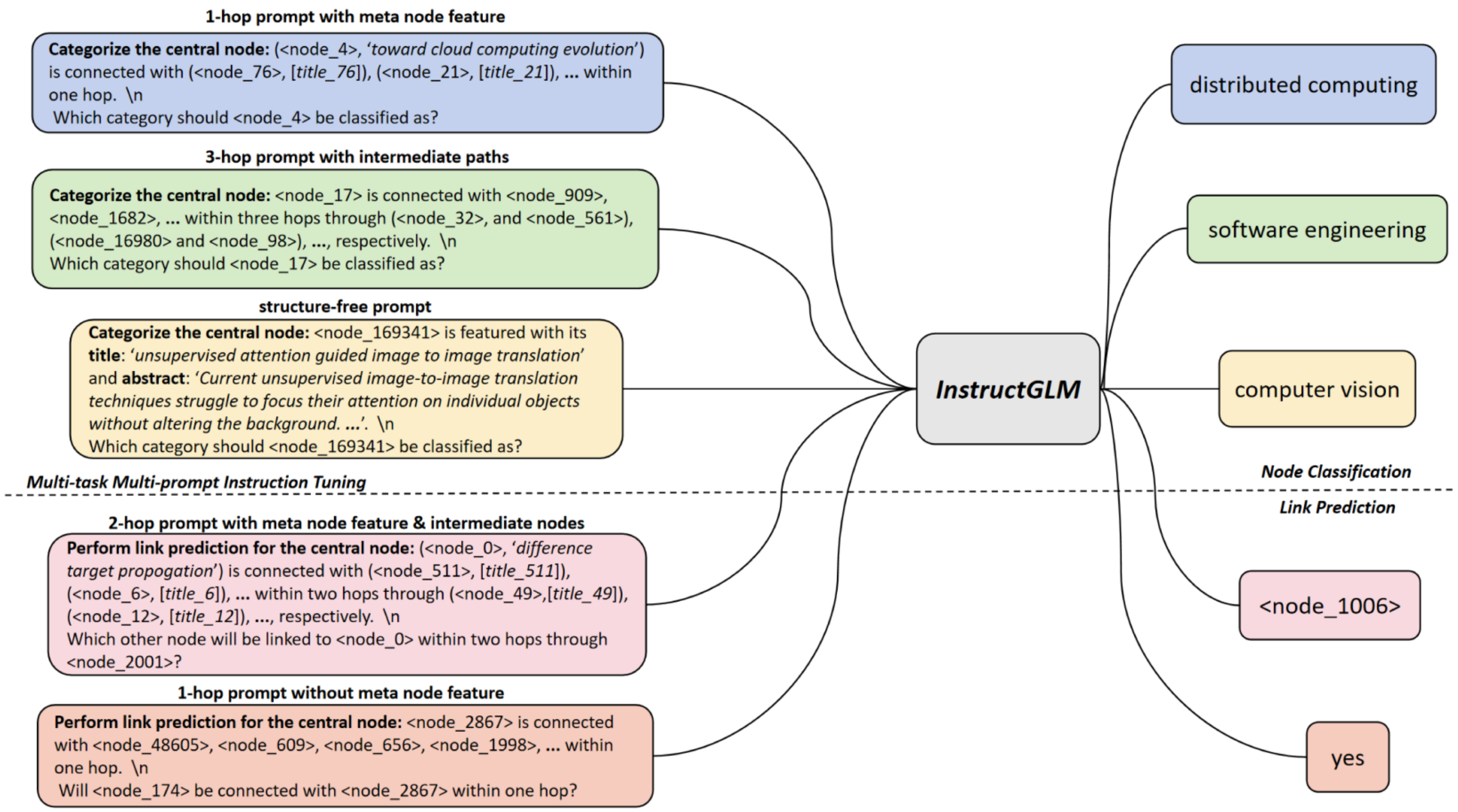
LLM-encoded node embeddings → Graph encoder
Link prediction task→ Pre-train both the LLM and the graph encoder

H. Xie, *et al*. "Graph-aware language model pre-training on a large graph corpus can help multiple graph applications." *KDD'23*.

Graph + PLM: Pre-training

- InstructGLM: Language model pre-training

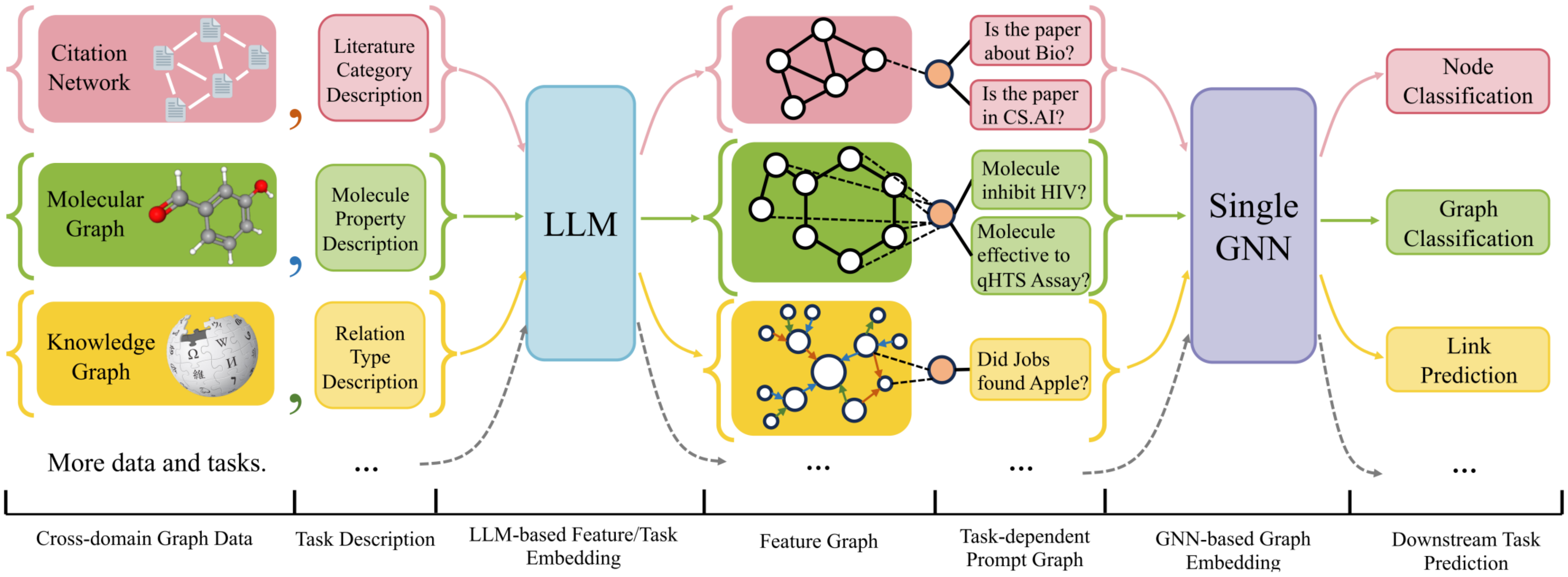
Natural language→
Describe graph structures



R. Ye, *et al*. " Language is All a Graph Needs." *EACL'24*.

Graph + PLM: Pre-training

- One for all: LLM + GNN pre-training



LLM: text/task embedding GNN: prompted graph

H. Liu, *et al*. "One for all: Towards training one graph model for all classification tasks." ICLR'24.

Graph + PLM: Adaptation

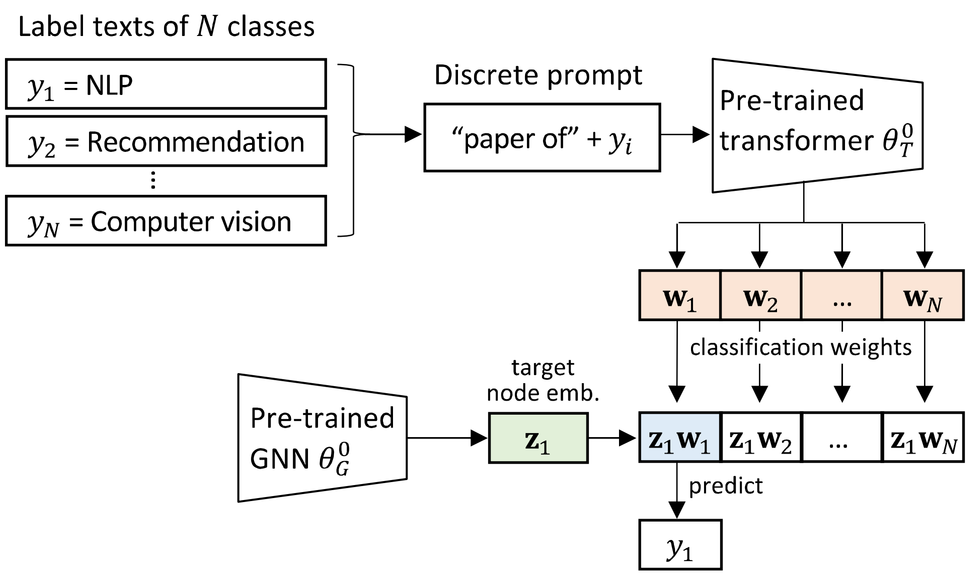
- Prompt-tuning
- Parameter-effeicient fine-tuning (PEFT)

TABLE IX: Summary of prompt tuning on text-attributed graphs.

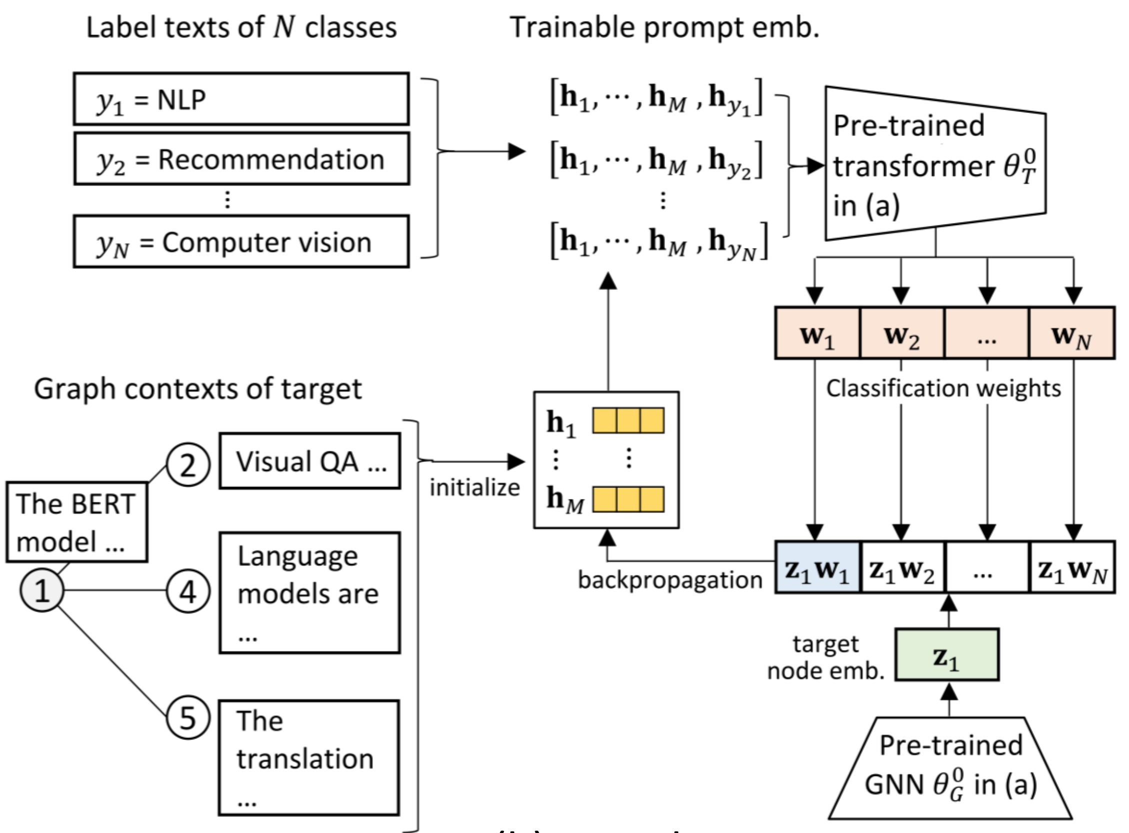
Paper	Instruction Text	Instruction Graph	Learnable prompt	Downstream Task Node	Downstream Task Edge	Downstream Task Graph
G2P2 [44]	✓	×	vector	✓	×	×
G2P2* [82]	✓	×	condition-net	✓	×	×
GraphGPT [170]	✓	✓	×	✓	×	×
InstructGLM [175]	✓	✓	×	✓	×	×
GIMLET [185]	✓	✓	×	×	×	✓
OFA [45]	✓	✓	×	✓	✓	✓
HiGPT [186]	✓	✓	×	✓	×	×

Graph + PLM: Prompt-Tuning

- G2P2
  - ➢ Discrete prompt-tuning: zero-shot
  - ➢ Trainable prompt-tuning: few-shot



(a) Zero-shot

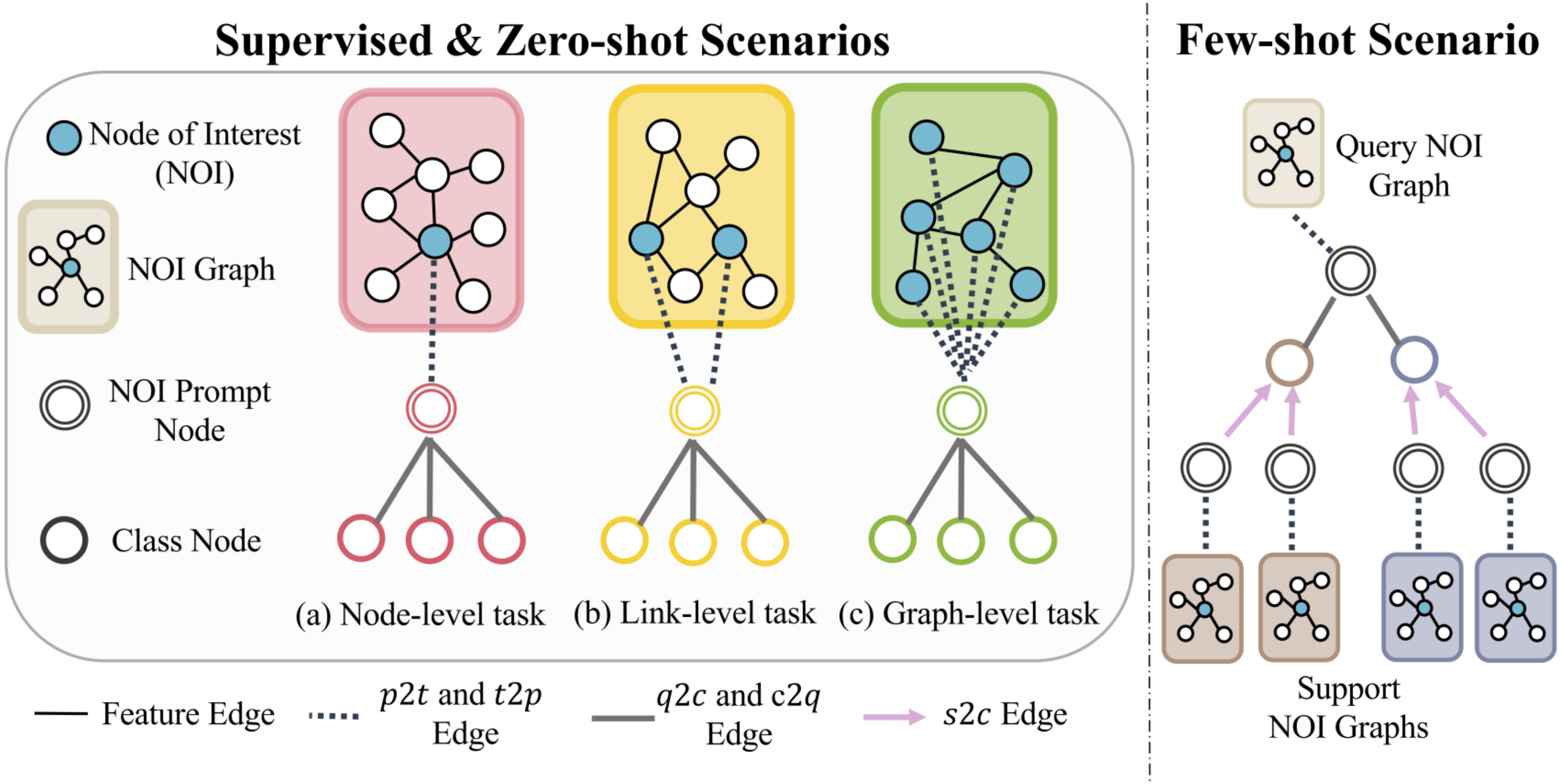


(b) Few-shot

Z. Wen, *et al*. "Augmenting low-resource text classification with graph-grounded pre-training and prompting." SIGIR'23.

Graph + PLM: Prompt-Tuning

- One for all



H. Liu, *et al*. "One for all: Towards training one graph model for all classification tasks." ICLR'24.

Graph + PLM: PEFT

- MolCA

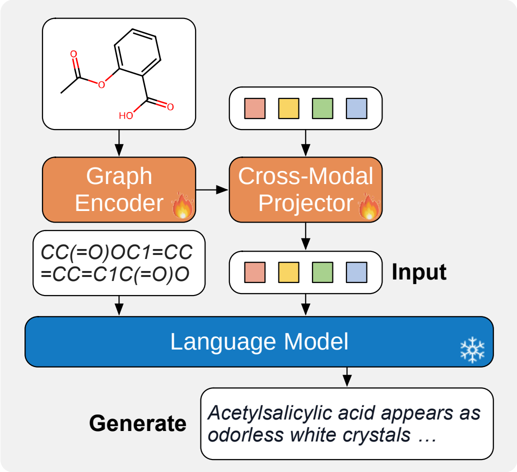


Figure 4: MolCA's pretrain stage 2 by molecule captioning.

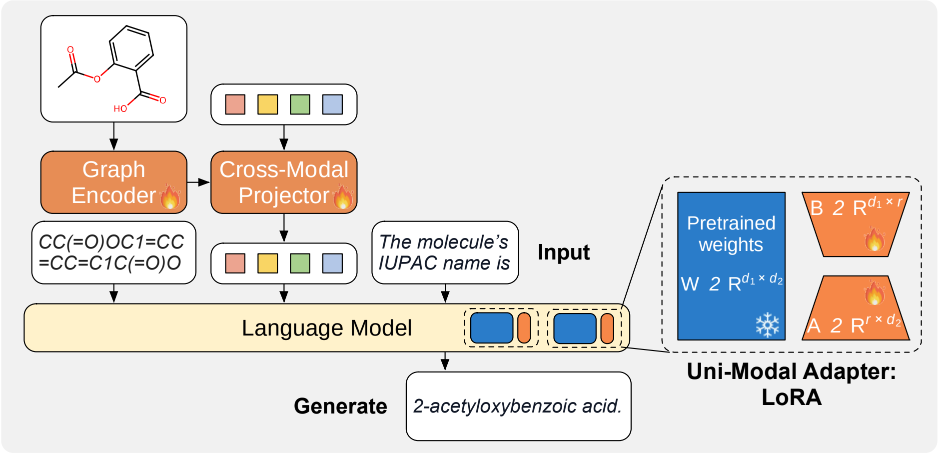


Figure 5: MolCA's fine-tune stage for molecule-to-text generation. The example shows the prediction of a molecule's IUPAC name.

Cross-Modal Projector: bridge the gap between graph structural and textual representations

Uni-Modal LoRA Adapter: efficient downstream adaptation

Z. Liu, *et al*. "MolCA: Molecular Graph-Language Modeling with Cross-Modal Projector and Uni-Modal Adapter." EMNLP'23

Graph + PLM: PEFT

- GraphGPT: only fintune projector
  - ➢Aligh graph to LLM

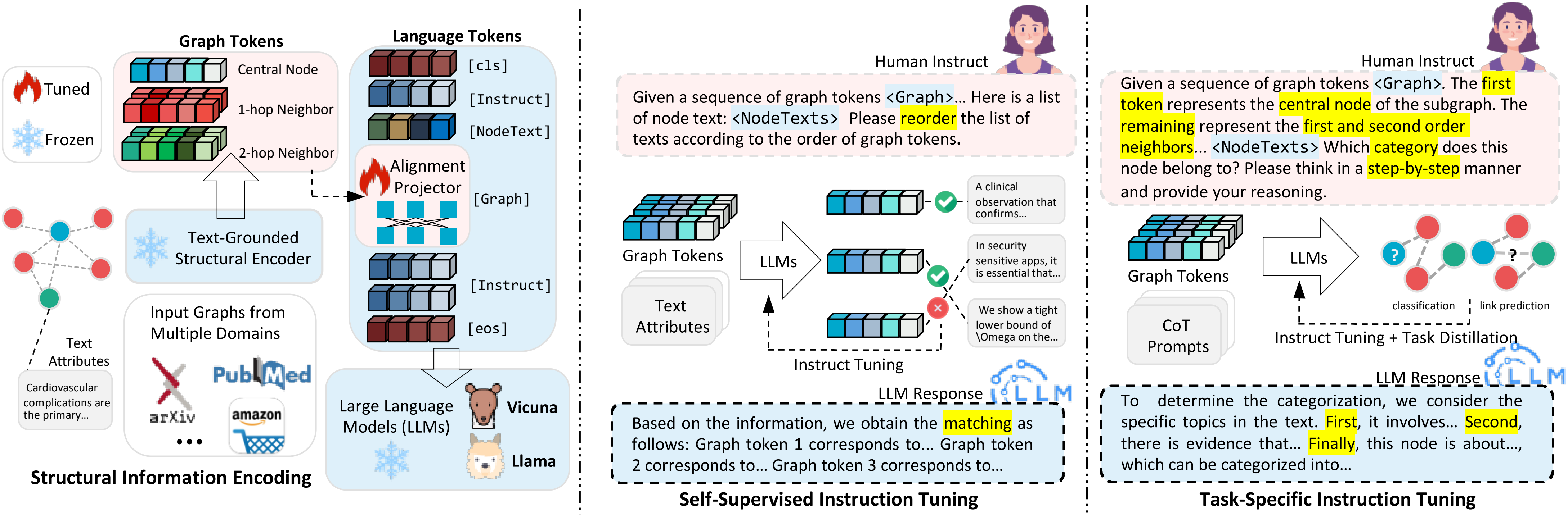


Figure 2: The overall architecture of our proposed GraphGPT with graph instruction tuning paradigm.

J. Tang, *et al*. "GraphGPT: Graph instruction tuning for large language models." SIGIR'24.

Graph + PLM: PEFT

- GraphTranslator

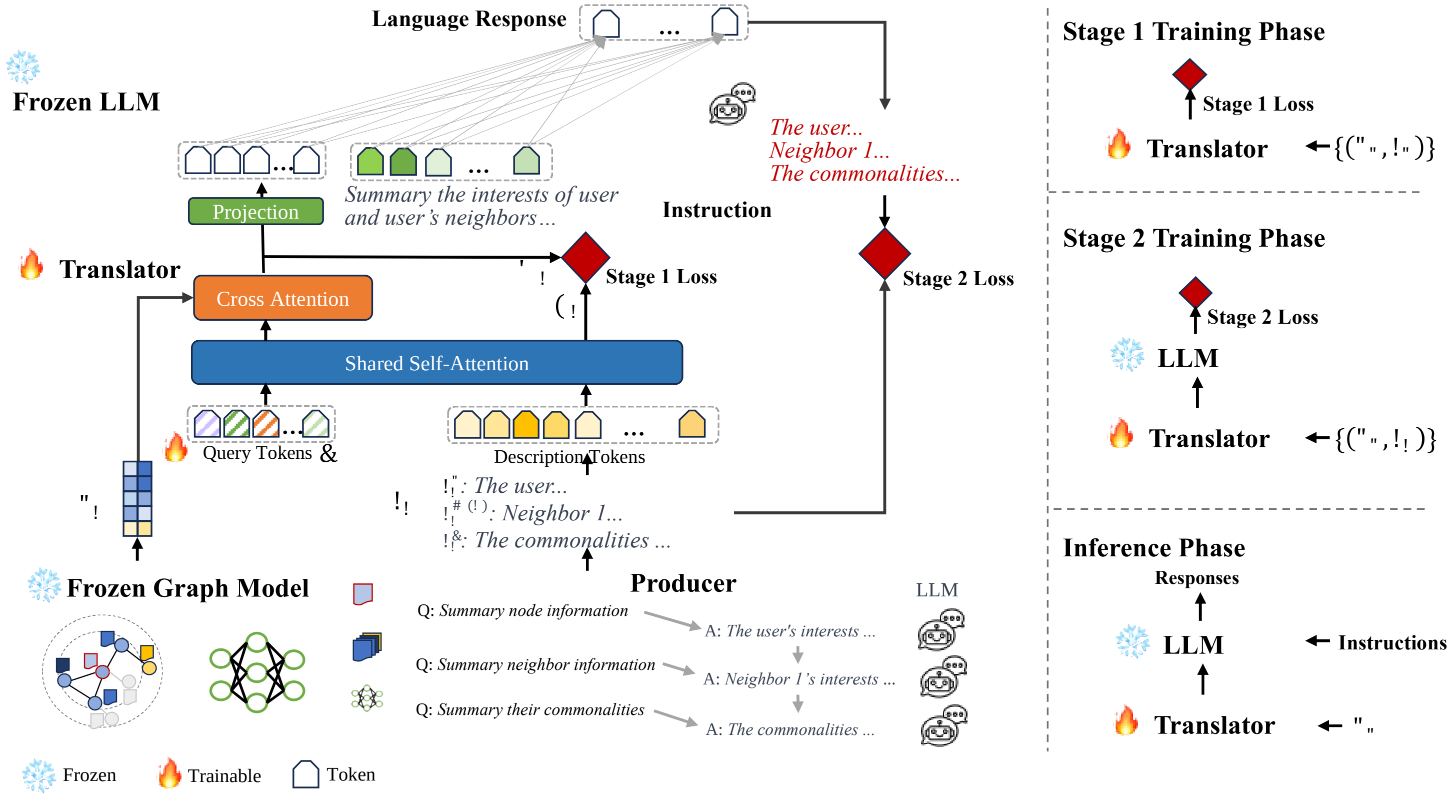


Figure 2: The overall framework of our *GraphTranslator*, which aligns GM to LLM by Translator for open-ended tasks. We train the lightweight Translator module following a two-stage paradigm, with the alignment data generated by our Producer.

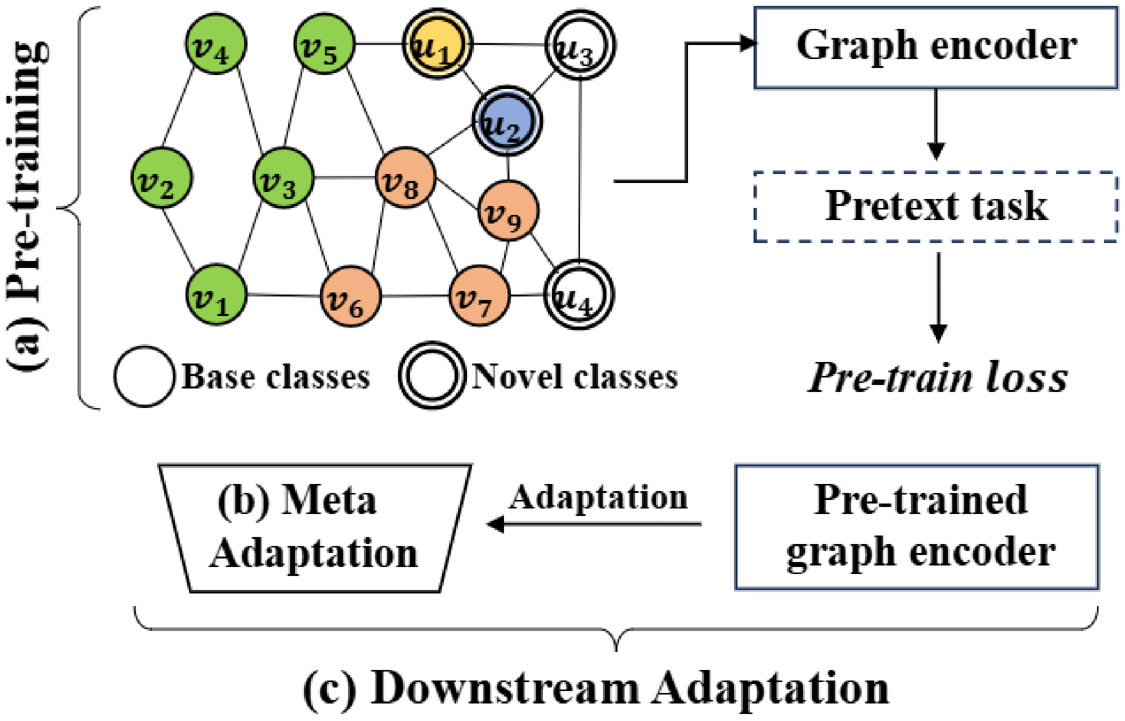
M. Zhang, *et al*. "GraphTranslator: Aligning Graph Model to Large Language Model for Open-ended Tasks." WWW'24

Summary

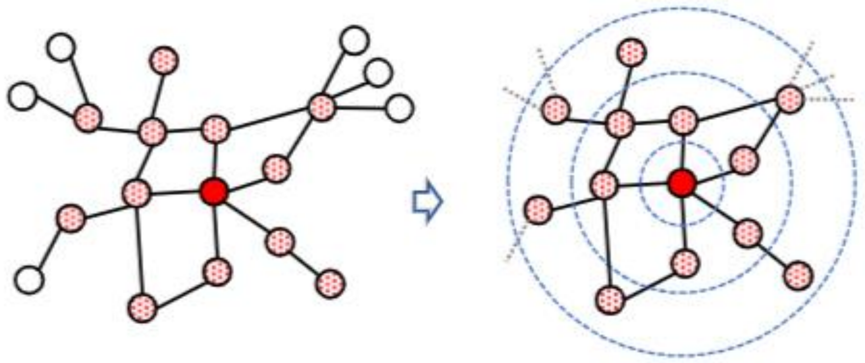
- Pre-training approaches employ self-supervised pretext tasks on unlabeled data
- Pre-training approaches are more effective in scenarios where labeled data are limited to novel tasks without a pre-existing set of annotated tasks.
- When a large annotated base set is available, meta-learning tends to perform better as it can leverage related meta-training tasks derived from the base set.
- Parameter-efficient adaptation strategies, including prompt tuning, adapter tuning and LoRA, present a more promising direction for few-shot learning on graphs.

Hybrid Approaches

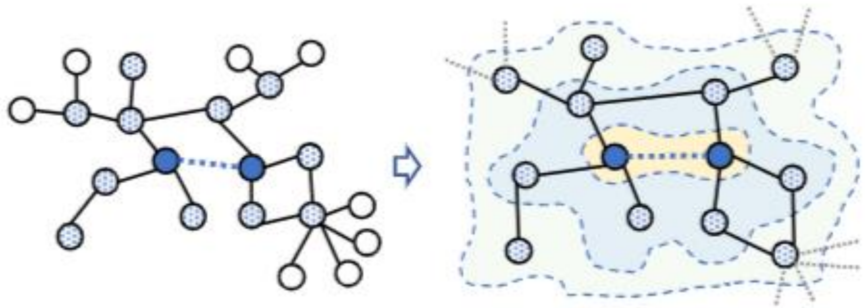
- Adopt pretext tasks to pre-train a graph encoder
  - ➢Unlabeled data for pre-training
- The pre-trained model is adapted in conjunction with meta-learning
  - ➢Annotated base set for meta-learning



ProG



(a) Induced graphs for nodes



(b) Induced graphs for edges

Unify node level and edge level tasks as graph level tasks

Prompt graph

$$\mathcal{G}_p = (\mathcal{P}, \mathcal{S}) \quad \mathcal{P} = \{p_1, p_2, \cdots, p_{|\mathcal{P}|}\}$$

Prompt modification

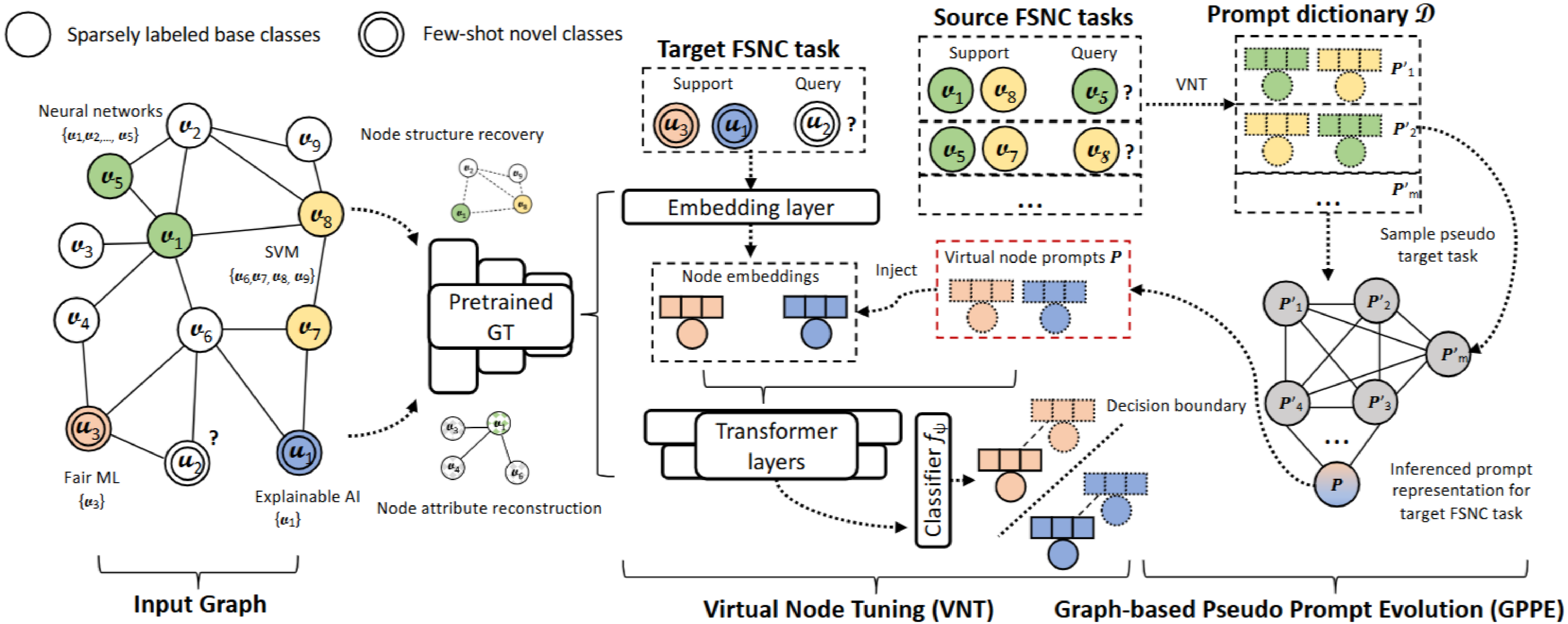
$$\hat{\mathbf{x}}_i = \mathbf{x}_i + \sum_{k=1}^{|\mathcal{P}|} w_{ik} \mathbf{p}_k$$

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

First pre-train a graph encoder, then apply meta-learning to the prompting phase

Sun, et al. "All in one: Multi-task prompting for graph neural networks." SIGKDD'23.

VNT



First pre-train a graph transformer

Prompt modification

Prompts

$$P = [p_1; \ldots; p_p; \ldots; p_P]$$

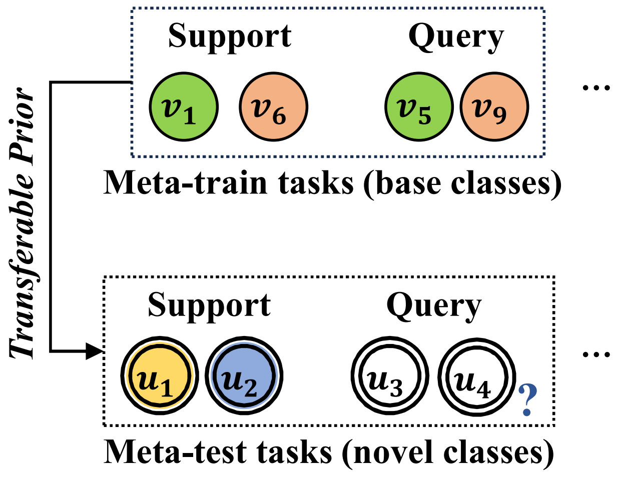
$$[E^1 || Z^1] = L^1([E^0 || P]) \in \mathbb{R}^{(V+P)\times F}$$

Tan, et al. “Virtual node tuning for few-shot node classification.” SIGKDD’23.

Future Avenues in Problem Settings

- Structure scarcity learning on graphs

Existing solution:

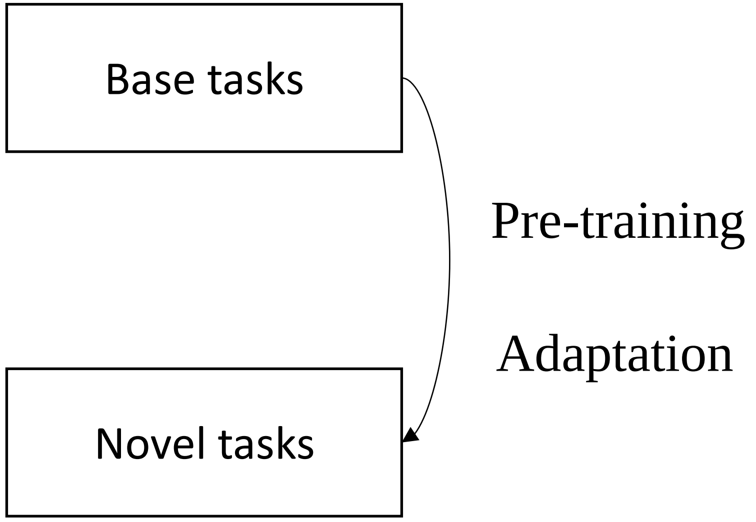


Structure-rich

Structure-scarce

Constrain:
Independent and Identically Distributed (i.i.d.)

Future direction:



Bridge the gap between base and novel tasks

Future Avenues in Problem Settings

- Few-shot learning on large-scale graphs

Category	Name	Scale	#Graphs	Average #Nodes	Average #Edges	Average Node Deg.	Average Clust. Coeff.	MaxSCC Ratio	Graph Diameter
Node ogbn-	products	medium	1	2,449,029	61,859,140	50.5	0.411	0.974	27
	proteins	medium	1	132,534	39,561,252	597.0	0.280	1.000	9
	arxiv	small	1	169,343	1,166,243	13.7	0.226	1.000	23
	papers100M	large	1	111,059,956	1,615,685,872	29.1	0.085	1.000	25
	mag	medium	1	1,939,743	25,582,108	21.7	0.098	1.000	6
Link ogbl-	ppa	medium	1	576,289	30,326,273	73.7	0.223	0.999	14
	collab	small	1	235,868	1,285,465	8.2	0.729	0.987	22
	ddi	small	1	4,267	1,334,889	500.5	0.514	1.000	5
	citation	medium	1	2,927,963	30,561,187	20.7	0.178	0.996	21
	wikikg	medium	1	2,500,604	17,137,181	12.2	0.168	1.000	26
	biokg	small	1	93,773	5,088,434	47.5	0.409	0.999	8
Graph ogbg-	molhiv	small	41,127	25.5	27.5	2.2	0.002	0.993	12.0
	molpcba	medium	437,929	26.0	28.1	2.2	0.002	0.999	13.6
	ppa	medium	158,100	243.4	2,266.1	18.3	0.513	1.000	4.8
	code	medium	452,741	125.2	124.2	2.0	0.0	1.000	13.5

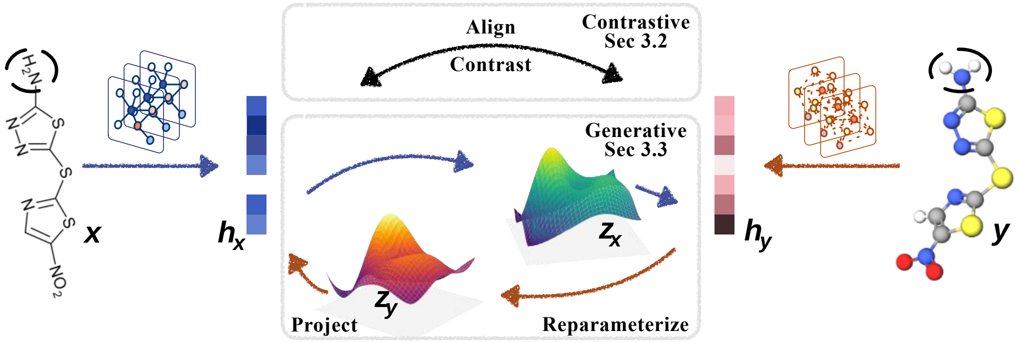
Dataset	Nodes	Edges	Classes	splitting (Train/Validation/Test)	Task
Flickr	89,250	899,756	7	0.50 / 0.25 / 0.25	Multi-Class Classification
Reddit	232,965	11,606,919	41	0.66 / 0.10 / 0.24	Multi-Class Classification
ogbn-products	2,449,029	61,859,140	47	0.10 / 0.02 / 0.88	Multi-Class Classification

Challenges: Finer-grained adaptation strategies to deal with potential variations among distant localities on a large graph.

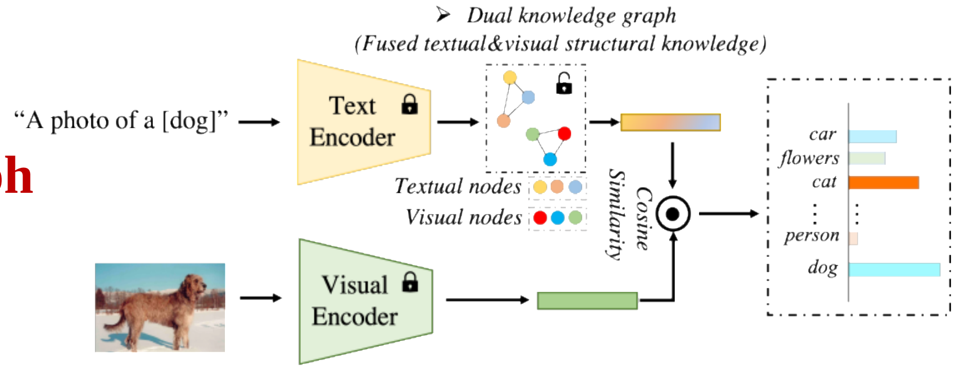
Future Avenues in Problem Settings

- Few-shot learning on complex graphs

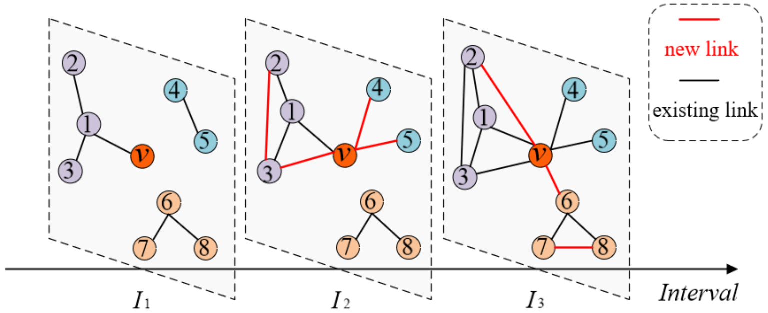
3D graph



Multi-modal graph



Dynamic graph



S. Liu, *et al.* "Pre-training Molecular Graph Representation with 3D Geometry." ICLR'22

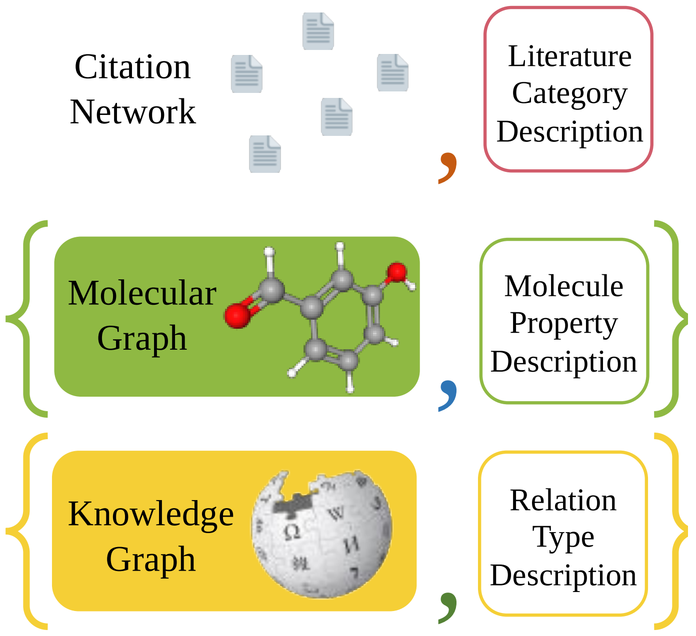
X. Li, *et al.*"GraphAdapter: Tuning Vision-Language Models With Dual Knowledge Graph."NeurIPS'23

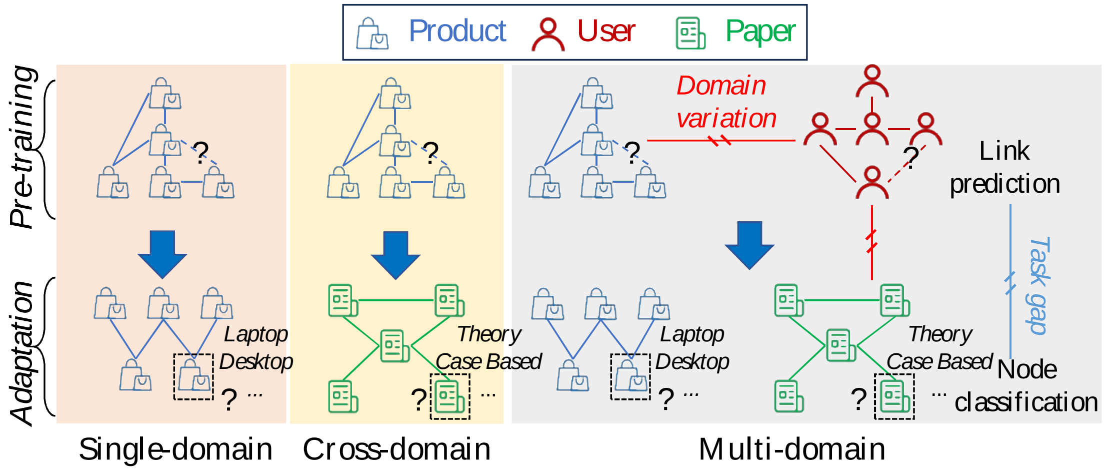
C. Yang, *et al.* "Few-shot Link Prediction in Dynamic Networks." WSDM'22

Future Avenues in Problem Settings

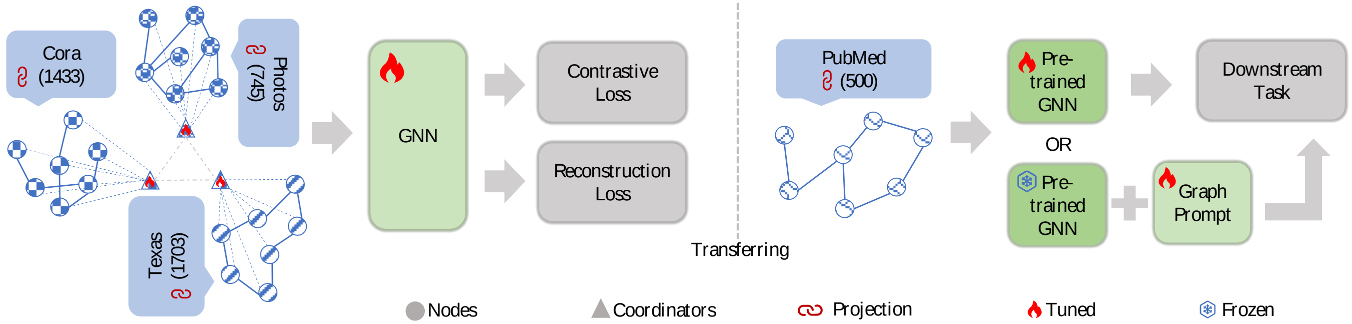
- Few-shot learning on cross-domain graphs

Text-Attribute Multi-domain Graph





Text-Free Multi-domain Graph



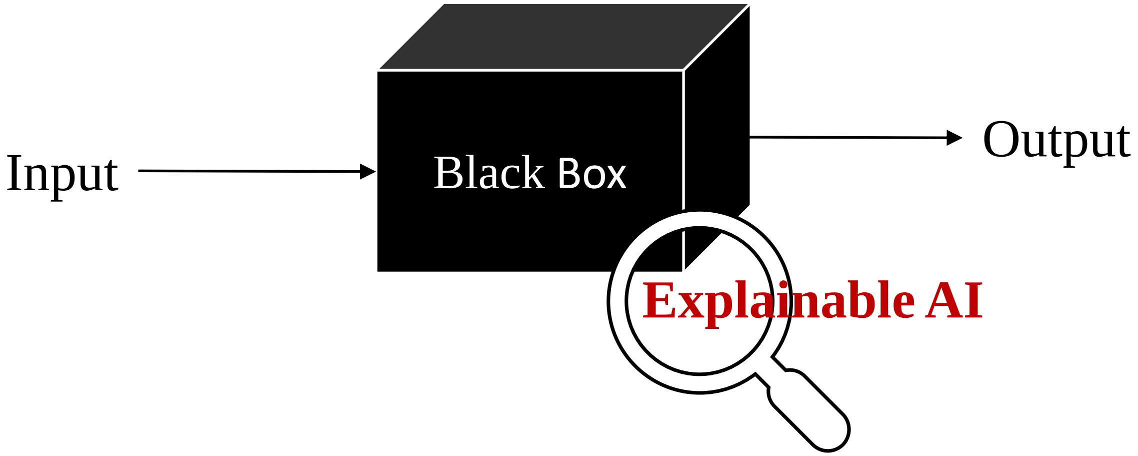
H. Liu, et al. "One for All: Towards Training One Graph Model for All Classification Tasks." ICLR'24

X. Yu, et al. "Text-Free Multi-domain Graph Pre-training: Toward Graph Foundation Models." ArXiv 2024

H. Zhao, et al, "All in One and One for All: A Simple yet Effective Method towards Cross-domain Graph Pretraining."KDD'24

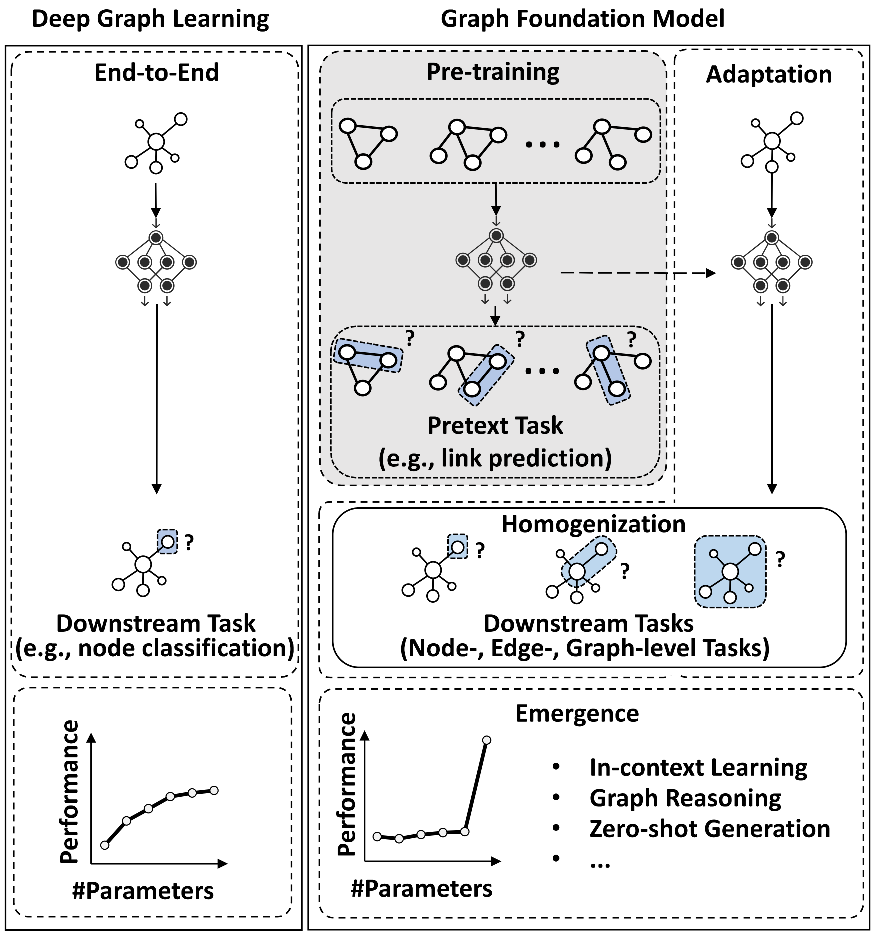
Future Avenues in Techniques

- Improving interpretability



Future Avenues in Techniques

- Foundation models on graphs



J. Liu, et al. “Graph Foundation Models: Concepts, Opportunities and Challenges.” TPAMI’25