

# HiGPT: Heterogeneous Graph Language Model

Jiabin Tang<sup>1</sup>, Yuhao Yang<sup>2</sup>, Wei Wei<sup>2</sup>, Lei Shi<sup>3</sup>,  
Long Xia<sup>3</sup>, Dawei Yin<sup>3</sup> and Chao Huang<sup>1,2\*</sup>

<sup>1</sup>Musketeers Foundation Institute of Data Science,

<sup>2</sup>Department of Computer Science, University of Hong Kong, <sup>3</sup>Baidu Inc.

**Project Page:** <https://HiGPT-HKU.github.io>, **Github:** <https://github.com/HKUDS/HiGPT>

## ABSTRACT

Heterogeneous graph learning aims to capture complex relationships and diverse relational semantics among entities in a heterogeneous graph to obtain meaningful representations for nodes and edges. Recent advancements in heterogeneous graph neural networks (HGNNs) have achieved state-of-the-art performance by considering relation heterogeneity and using specialized message functions and aggregation rules. However, existing frameworks for heterogeneous graph learning have limitations in generalizing across diverse heterogeneous graph datasets. Most of these frameworks follow the "pre-train" and "fine-tune" paradigm on the same dataset, which restricts their capacity to adapt to new and unseen data. This raises the question: "Can we generalize heterogeneous graph models to be well-adapted to diverse downstream learning tasks with distribution shifts in both node token sets and relation type heterogeneity?" To tackle those challenges, we propose HiGPT, a general large graph model with Heterogeneous graph instruction-tuning paradigm. Our framework enables learning from arbitrary heterogeneous graphs without the need for any fine-tuning process from downstream datasets. To handle distribution shifts in heterogeneity, we introduce an in-context heterogeneous graph tokenizer that captures semantic relationships in different heterogeneous graphs, facilitating model adaptation. We incorporate a large corpus of heterogeneity-aware graph instructions into our HiGPT, enabling the model to effectively comprehend complex relation heterogeneity and distinguish between various types of graph tokens. Furthermore, we introduce the Mixture-of-Thought (MoT) instruction augmentation paradigm to mitigate data scarcity by generating diverse and informative instructions. Through comprehensive evaluations conducted in various settings, our proposed framework demonstrates exceptional performance in terms of generalization performance, surpassing current leading benchmarks.

### ACM Reference Format:

Jiabin Tang<sup>1</sup>, Yuhao Yang<sup>2</sup>, Wei Wei<sup>2</sup>, Lei Shi<sup>3</sup>, Long Xia<sup>3</sup>, Dawei Yin<sup>3</sup> and Chao Huang<sup>1,2\*</sup>. 2024. HiGPT: Heterogeneous Graph Language Model. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 19 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

## 1 INTRODUCTION

Heterogeneous graphs have garnered extensive popularity and adoption in various domains, including recommendation systems [6], knowledge graphs [29], social network analysis [5], and biological networks [17]. These graphs encompass entities of diverse types

that engage in a multitude of interactions, enabling a comprehensive representation of complex systems [4]. The focus of heterogeneous graph learning is to derive meaningful representations for the nodes and edges within such graphs [12, 16, 45]. These representations aim to capture the intricate relationships and diverse relational semantics that exist within the graph, facilitating a deeper understanding of the underlying structural heterogeneity.

In recent years, there has been a growing recognition of the significant potential of heterogeneous graph neural networks (HGNNs) in capturing the intricate and diverse information that resides within heterogeneous graph structures [31, 40]. HGNNs leverage the expressive capabilities of high-order message passing techniques, enabling them to effectively model the complex relationships, diverse entity types, and heterogeneous semantics present in these graphs. By aggregating and propagating information across various node and edge types, HGNNs facilitate a deeper understanding and analysis of the intricate inter-dependencies that exist within heterogeneous graph structures. Some notable examples of HGNNs include: i) Metapath-based GNNs such as HAN [32] and MAGNN [6]; ii) Transformer-enhanced GNNs like HGT [10]. Furthermore, heterogeneous graph self-supervised learning, including contrastive methods (e.g., DMGI [19] and HeCo [34]), and generative (e.g., HGMAE [27]) methods, showcase effectiveness in alleviating data scarcity in real-world heterogeneous graph data.

Despite the demonstrated effectiveness of current frameworks for heterogeneous graph learning, they possess limitations when it comes to generalizing across diverse heterogeneous graph datasets. These frameworks commonly adopt the "pre-train" and "fine-tune" paradigm, where they are initially trained on a specific dataset and subsequently fine-tuned on the same dataset [10, 27, 34]. However, this approach presents challenges in adapting and achieving optimal performance on new and unseen data. The heavy reliance on the characteristics and patterns of the original training dataset hinders their ability to effectively handle the intricacies and complexities inherent in different heterogeneous graph datasets. As a result, these frameworks may encounter difficulties in effectively addressing the diverse nuances and variations present in various heterogeneous graph datasets, particularly when applied to downstream tasks.

This research aims to push the boundaries of heterogeneous graph models by addressing a fundamental question: "Can we develop highly adaptable and versatile heterogeneous graph models capable of effectively addressing diverse downstream learning tasks, even when faced with distribution shifts in node token sets and relation type heterogeneity?" To tackle this challenge, we introduce HiGPT as a novel and general solution. Our model is specifically designed to overcome key challenges associated with generalization across various downstream heterogeneous graph learning tasks.

\* Chao Huang is the Corresponding Author.

**C1. Relation Type Heterogeneity Shift.** One of the primary challenges we focus on in this research is the shift in relation type heterogeneity within various heterogeneous graph structures. In such graphs, entities are connected by various types of relations, and these relation types can differ significantly across diverse datasets. To illustrate this, let’s consider two examples. In a recommendation system, a heterogeneous graph may involve node-wise relationships between users and items. The relation types in this scenario could include "click," "favorite," "review," and "purchase." On the other hand, in an academic graph, the relations could involve "paper-paper," "author-paper," and "paper-venue." These examples demonstrate how different heterogeneous graphs can exhibit diverse relation heterogeneity with distinct semantics across domains.

**Solution: In-Context Heterogeneous Graph Tokenizer.** To achieve adaptability in a wide range of heterogeneous graph scenarios with varying node and edge types, we introduce the in-context heterogeneous graph tokenizer. This tokenizer captures the diverse semantic relationships found in different heterogeneous graphs, providing a unified approach. It comprises two essential components: the in-context parameterized heterogeneity projector, which utilizes language to encode distinct node and edge types, facilitating effective model adaptation, and the parameter allocator, which dynamically assigns tailored parameters to the tokenizer. To optimize performance and integrate the tokenizer seamlessly into the HiGPT framework, we employ pre-training with a lightweight text-graph contrastive alignment paradigm. This pre-training process directly incorporates the tokenizer into HiGPT, enhancing its capabilities and ensuring smooth functionality within the overall model architecture, including integration with the language model.

**C2. Complex Heterogeneous Graph Structures.** The primary focus of this study is to tackle the challenge of integrating large language models (LLMs) into heterogeneous graph learning, with the goal of enhancing model generalization. Our specific objective is to develop a graph-oriented language model that excels in comprehending the intricate structural information inherent in complex heterogeneous graph structures. In doing so, we strive to equip the graph model with the ability to not only recognize the heterogeneity of relations among different types of nodes, but also capture the distinct characteristics of entities belonging to the same type.

**Solution: Heterogeneous Graph Instruction-Tuning.** We introduce a novel heterogeneous graph instruction-tuning framework that integrates inter-type and intra-type token matching tasks to fine-tune large language models (LLMs). Our framework specifically targets the enhancement of LLMs’ understanding of both heterogeneous relation awareness and homogeneous relation awareness. By utilizing these tasks, our aim is to bolster the LLMs’ capabilities in the following areas: (i) distinguishing between different types of graph tokens, (ii) comprehending intricate relationships within heterogeneous graphs, (iii) preserving the distinctive attributes of entities within homogeneous graphs, and (iv) effectively harnessing diverse graph instructions during the training process.

**C3. Data Scarcity for Model Fine-Tuning.** In practical scenarios involving heterogeneous graph learning, one of the key challenges is the limited availability of data [11, 41]. This scarcity of data poses a significant obstacle when fine-tuning models for real-world applications. For instance, when utilizing heterogeneous graphs to

model cold-start users or items in recommendation systems, the sparse nature of user interaction data restricts the availability of supervised signals [36, 39]. This data scarcity hampers the effectiveness of task-specific model fine-tuning and necessitates the development of novel techniques to overcome this challenge.

**Solution: Mixture-of-Thought Augmentation.** Our approach introduces a novel mechanism for augmenting graph instructions, emphasizing the use of Mixture-of-Thought (MoT) combined with various prompting techniques. This integration enables us to generate a diverse and comprehensive set of informative task-specific instructions. By seamlessly incorporating these augmented graph instructions into our framework, we anticipate that our model enhancement will effectively address the challenge of data sparsity.

**Experiments.** To assess the efficacy of our proposed approach, we perform a comprehensive set of experiments to showcase the impressive generalization capabilities of our heterogeneous graph language model across diverse settings. We thoroughly investigate the design rationales, effectiveness, and efficiency of our model.

## 2 PRELIMINARIES

**Heterogeneous Graph.** A heterogeneous graph is a graph denoted as  $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{A}, \mathcal{T}, \mathcal{R}, \mathbf{X})$ . It consists of nodes represented by  $\mathcal{V}$ , edges represented by  $\mathcal{E}$ , and an adjacency matrix  $\mathbf{A}$  that captures the relationships between nodes. The sets  $\mathcal{T}$  and  $\mathcal{R}$  signify the types of nodes and edges, respectively. Additionally, the feature matrix  $\mathbf{X} = \{X_{T_i} \in \mathbb{R}^{|\mathcal{V}_{T_i}| \times d_{T_i}}\}$  contains attributes associated with each node. Here,  $T_i$  refers to a specific node type, while  $d_{T_i}$  represents the dimensionality of the corresponding node attributes.

**Meta Relation.** In a heterogeneous graph, a meta relation is a representation of the relationship between different types of nodes connected by an edge. Specifically, for an edge  $e$  that links a source node  $u$  of type  $T_i$  to a target node  $v$  of type  $T_j$ , the meta relation of  $e = (u, v)$  is denoted as  $\langle \tau(u), \rho(e), \tau(v) \rangle$ . Here,  $\tau(u)$  and  $\tau(v)$  represent the node types of  $u$  and  $v$  respectively, drawn from the set of node types  $\mathcal{T}$ , while  $\rho(e)$  denotes the relation type of the edge  $e$ , which is selected from the set of relation types  $\mathcal{R}$ . This meta relation provides a concise representation of the heterogeneous connections in the graph, capturing the types of nodes involved.

**Heterogeneous Graph Neural Networks (HGNNs).** In the context of a heterogeneous graph  $\mathcal{G}$ , Heterogeneous Graph Neural Networks (HGNNs) employ message passing and aggregation techniques to incorporate neighbor information based on different node and edge categories. This enables the modeling of heterogeneous structural semantic relationships, as expressed below:

$$h_v^{(l)} = \underset{\forall u \in \mathcal{N}(v), \forall e \in \mathcal{E}(u,v)}{\text{Aggregate}} \left( \underset{\text{Propagate}}{h_u^{(l-1)}; h_v^{(l-1)}, e} \right) \quad (1)$$

Here,  $\mathcal{N}(v)$  represents all the source nodes connected to node  $v$ , and  $\mathcal{E}(u, v)$  denotes the set of edges connecting node  $u$  and node  $v$ . In most HGNNs, the parameters of the **Propagate** ( $\cdot$ ) and **Aggregate** ( $\cdot$ ) functions depend on the types of nodes  $u$  and  $v$ , as well as the edge  $e$ . However, this implies that HGNNs are limited to modeling the specific heterogeneous graph they were trained on and cannot be effectively applied to new heterogeneous graphs

with different node and edge types. This limitation greatly hampers the generalization ability of HGNNs in capturing the diverse semantic relationships across various heterogeneous graphs.

### 3 METHODOLOGY

#### 3.1 In-Context Heterogeneous Graph Tokenizer

To make our HiGPT adaptable to a wide range of heterogeneous graph scenarios with varying node and edge types, we propose an in-context heterogeneous graph tokenizer. This method captures the diverse semantic relationships present in different heterogeneous graphs, ensuring a unified approach. It comprises two essential components: the in-context parameterized heterogeneity projector and the parameter allocator. The adaptive in-context projector utilizes language to encode the distinct node and edge types within the heterogeneous graphs, facilitating model adaptation.

Meanwhile, the parameter allocator dynamically assigns parameters tailored specifically for the tokenizer. To optimize the tokenizer’s performance and seamlessly integrate it within the HiGPT framework, we conduct pre-training using a simplified text-graph contrastive learning framework. This pre-training process directly incorporates the tokenizer into the HiGPT framework and effectively integrates it with the Large Language Model (LLM). This approach enhances the tokenizer’s capabilities and ensures its smooth functioning within the overall model architecture.

**3.1.1 Graph Tokenization with Meta Projector.** Given a heterogeneous graph  $\mathcal{G}$  with a feature matrix  $\mathbf{X} = \{X_{T_i} \in \mathbb{R}^{|\mathcal{V}_{T_i}| \times d_{T_i}}, T_i \in \mathcal{T}\}$  and an adjacency matrix  $\mathbf{A}$ , the goal of the heterogeneous graph tokenizer is to encode the hidden representations of the heterogeneous graph, denoted as  $\mathbf{H} = \{H_{T_i} \in \mathbb{R}^{|\mathcal{V}_{T_i}| \times f_{T_i}}, T_i \in \mathcal{T}\}$ . This is achieved through the function  $\mathbf{H} = \text{HG-Tokenizer}(\mathbf{X}, \mathbf{A})$ , where  $f_{T_i}$  represents the hidden dimension of node type  $T_i$ . The  $\text{HG-Tokenizer}(\cdot)$  can be implemented using various backbone HGNN architectures, such as HetGNN [44], HAN [33], or HGT [10].

However, the generalization capability of these heterogeneous GNNs is constrained by their inherent design, which includes pre-defined parameter learning tailored to specific heterogeneous graphs. As a result, the trained HGNNs cannot be readily applied to other unseen heterogeneous graphs, which goes against the objective of achieving unified encoding with the HG-Tokenizer. To illustrate, let’s consider HGT as an example. In HGT, the calculation of  $h_v^{(l)}$  involves utilizing functions such as  $\text{Attention}(\cdot)$  and  $\text{Message}(\cdot)$  to process information from the source nodes:

$$\begin{aligned} \tilde{h}_v^{(l)} &= \bigoplus_{u \in \mathcal{N}(v)} (\text{Attention}(u, e, v) \cdot \text{Message}(u, e, v)) \\ h_v^{(l)} &= \mathcal{F}_{\Theta_1}^{\tau(v)} \left( \sigma \left( \tilde{h}_v^{(l)} \right) \right) + h_v^{(l-1)} \\ &= \mathbf{W}_1^{\tau(v)} \cdot \left( \sigma \left( \tilde{h}_v^{(l)} \right) \right) + \mathbf{b}_1^{\tau(v)} + h_v^{(l-1)} \end{aligned} \quad (2)$$

The notation  $\mathcal{F}_{\Theta_1}^{\tau(v)}(\cdot)$  represents a fully-connected layer with parameters  $\Theta_1 = \{\mathbf{W}^{\tau(v)}, \mathbf{b}^{\tau(v)}\}$ . Here,  $\tau(v)$  denotes the node type of

$v$ , and  $\sigma(\cdot)$  represents the activation function. The specific formulation of the  $\text{Attention}(\cdot)$  and  $\text{Message}(\cdot)$  functions, with  $h$  heads:

$$\begin{aligned} \text{Attention}(u, e, v) &= \text{Softmax}_{\forall u \in \mathcal{N}(v)} \left( \prod_{i \in [1, h]} \mathcal{F}_{\Theta_2}^{\tau(u)} \left( h_u^{(l-1)} \right) \mathbf{W}_1^{\rho(e)} \mathcal{F}_{\Theta_3}^{\tau(v)} \left( h_v^{(l-1)} \right) \right) \\ \text{Message}(u, e, v) &= \prod_{i \in [1, h]} \mathcal{F}_{\Theta_4}^{\tau(u)} \left( h_u^{(l-1)} \right) \mathbf{W}_2^{\rho(e)} \end{aligned} \quad (3)$$

**Adaptive Parameterized Heterogeneity Projector.** To make our HiGPT adaptable to a wide range of heterogeneous graphs with varying graph heterogeneity settings, and to eliminate the requirement of pre-defining the number of type-specific projections in advance, we propose the design of a type-aware parameterized projector. This projector dynamically and automatically encodes the relation heterogeneity into latent representations. More specifically, the type-aware projectors with the parameters  $\mathcal{F}_{\Theta_i}^{\tau(v)}$  and  $\mathbf{W}_i^{\rho(e)}$  are generated automatically according to the following procedure:

$$\Theta_i = \{\mathbf{W}_i^{\tau(v)}; \mathbf{b}_i^{\tau(v)}\} = \mathcal{F}_{\Omega} \left( \mathbf{T}^{\tau(v)} \right); \quad \mathbf{W}_i^{\rho(e)} = \mathcal{F}_{\Omega} \left( \mathbf{T}^{\rho(e)} \right) \quad (4)$$

$\mathcal{F}_{\Omega}$  is a fully-connected layer with parameters  $\Omega$ , while  $\mathbf{T}^{\tau(v)}$  and  $\mathbf{T}^{\rho(e)}$  are the features associated with node type  $\tau(v)$  and edge type  $\rho(e)$ , respectively. It is important to note that the example provided showcases the usage of the in-context parameterized heterogeneity projector within the heterogeneous graph transformer framework. However, our HiGPT is designed to be versatile and adaptable, allowing for the integration of diverse heterogeneous GNNs.

**Language-Enriched Heterogeneity Representation.** We leverage natural language as a means to generate universal heterogeneity representations for nodes and edges based on their respective types. For instance, in the heterogeneous IMDB dataset, we can describe a "movie" node as "This node represents a movie" using natural language. Similarly, the edge ("movie", "to", "director") can be expressed as "The movie is directed by the director". To encode these natural language descriptions of nodes and edges, we employ a pre-trained language model such as Sentence-BERT [22] to obtain type representations. To ensure distinguishability and diversity among different types, we utilize multiple languages to describe the same type. The encoded representations from the pre-trained language models are averaged to derive the final representation. This process can be defined as follows:

$$\begin{aligned} \mathbf{T}^{\tau(v)} &= \text{Mean-Pooling} \left( \text{Sentence-BERT} \left( \mathbf{S}^{\tau(v)} \right) \right) \\ \mathbf{T}^{\rho(e)} &= \text{Mean-Pooling} \left( \text{Sentence-BERT} \left( \mathbf{S}^{\rho(e)} \right) \right) \end{aligned} \quad (5)$$

$\mathbf{S}^{\tau(v)}$  and  $\mathbf{S}^{\rho(e)}$  represent sets of descriptions for node type  $\tau(v)$  and edge type  $\rho(e)$ , respectively. For instance, consider the example of the edge ("movie", "to", "director"). One possible description is:

$$\begin{aligned} \mathbf{S}^{\text{"movie", "to", "director"}} &= \{ \\ &\text{"The movie is directed by the director",} \\ &\text{"The film features direction by the director", \dots \} \end{aligned} \quad (6)$$

For comprehensive descriptions featuring text-enriched heterogeneity representations of various datasets, please consult the appendix.

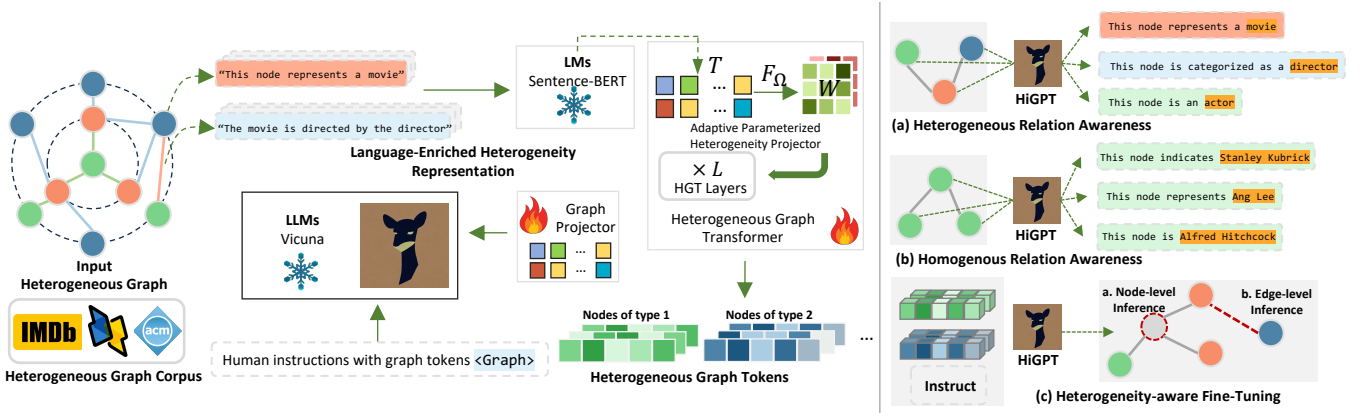


Figure 1: The overall architecture of our HiGPT.

**3.1.2 Lightweight Text-Graph Contrastive Alignment.** Building upon recent advancements in aligning cross-modality semantics [20, 38], we draw inspiration to employ a text-graph contrastive alignment paradigm for pre-training the proposed heterogeneous graph tokenizer. This approach aims to align the modeling capabilities of language and heterogeneous structures, enabling better collaboration between the tokenizer and the language models. To begin, we consider raw textual contents represented as  $C = c_i \in \mathbb{R}^{l_i \times d}$ ,  $1 \leq i \leq N$ , where  $N$  denotes the total number of heterogeneous graph nodes  $X = X_{T_i} \in \mathbb{R}^{l_i \times d_{T_i}}$ . Here,  $l_i$  represents the length of the textual content associated with the  $i$ -th node. In our approach, we adopt a lightweight text-graph contrastive alignment paradigm formally presented as follows:

$$\begin{aligned} \hat{H} &= \text{norm}(\text{HG-Tokenizer}(X)), \hat{T} = \text{norm}(\text{LM-Tokenizer}(C)) \\ \mathcal{L} &= \frac{1}{2} (\text{CE}(\Lambda, \mathbf{y}) + \text{CE}(\Lambda^\top, \mathbf{y})), \Lambda = (\hat{H}\hat{T}^\top) \cdot \exp(\tau) \end{aligned} \quad (7)$$

We use the contrastive label  $\mathbf{y} = (0, 1, \dots, n-1)^\top$  and the Cross-Entropy function  $\text{CE}(\cdot)$ . Our implementation employs the multi-layer vanilla transformer for LM-Tokenizer( $\cdot$ ).

## 3.2 Heterogeneous Graph Instruction Tuning

The objective of HiGPT is to empower language models to directly generate predictions for downstream tasks with the unseen heterogeneous graph and corresponding instructions. The natural language instruction is first encoded by a tokenizer into text embeddings, denoted as  $X_I = \text{LM-tokenizer}(\text{instruction})$ . To align the dimensions, we employ a projector that maps graph tokens to the same dimension as the text embeddings, given by  $X_G = f_p(H)$ , which can be as simple as a linear layer. For a sequence of length  $L$ , we determine the probability of generating the target output  $X_O$ :

$$p(X_O | X_G, X_I) = \prod_{i=1}^L p_\Phi(x_i | X_G, X_I, <i>, X_O, <i>) \quad (8)$$

where  $\Phi$  represents the learnable parameters within HiGPT.

### 3.2.1 Instruction Tuning with Heterogeneous Graph Corpus.

To enable the Language Model (LLM) to effectively differentiate between different types of input heterogeneous graph tokens and the specific nodes within each type, based on natural language instructions, we propose instruction pre-training using a large "corpus"

consisting of heterogeneous graph-instruction pairs. This approach equips the fine-tuned HiGPT with a comprehensive understanding of both homogeneous and heterogeneous graph structures.

- **Heterogeneous Relation Awareness.** Our objective is to enhance the language model's proficiency in distinguishing between specific types of nodes within a heterogeneous context, taking into account the intricate relationships. This is achieved by leveraging the information encoded in the graph tokens.
- **Homogeneous Relation Awareness.** Our aim is to equip the model with the ability to establish a significant correspondence between sequences of graph tokens that belong to the same category and their corresponding natural language descriptions.

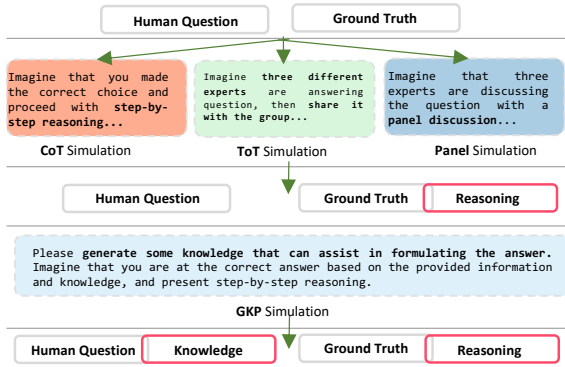
**Heterogeneous Graph Instruction.** In our graph instruction, we incorporate a heterogeneous subgraph generated through random neighbor sampling, accompanied by a question generated by a human. To enhance the diversity of the heterogeneous graph corpus, we conduct multiple samplings focusing on nodes from different types. Additionally, we introduce the  $\langle \text{graph} \rangle$  token as a graph indicator within the human question. i) *To achieve heterogeneous relation awareness*, we introduce the inter-type token matching task. This task involves providing the Language Model (LLM) with encoded sequences of graph tokens from different types, enabling it to differentiate between the various types. ii) *For homogeneous relation awareness*, we design the intra-type matching task, where the LLM receives encoded sequences of graph tokens from a specific type, allowing it to establish correspondence with the relevant descriptions. Further details regarding the instruction template at this stage are illustrated in Table 1 and Appendix Section A.4.

**3.2.2 Heterogeneity-aware Fine-Tuning.** To customize the reasoning abilities of the language model for specific downstream tasks on heterogeneous graphs, we propose Heterogeneity-aware Fine-Tuning. This approach entails conducting supervised learning with task-specific instructions following the initial instruction pre-training phase with heterogeneous graph corpus. It allows us to refine the LLM's performance and adapt it to the specific requirements of the targeted tasks on heterogeneous graphs.

In this stage, we incorporate a randomly sampled heterogeneous subgraph centered around the target node, along with a human-generated question. Given that the previous phase of instruction pre-training has already equipped the model with heterogeneous

**Table 1: Prompts for the three tasks of heterogeneous graph instruction-tuning.**

HeteroGraph	Human Question	HiGPT Response
(a) Heterogeneous Instruction Pre-training		
central_nodes: ("movie": [1, ..., n]), num_neighbors:[10, 10]	Given a heterogeneous graph about <b>movies</b> , there are <b>3 types of nodes</b> : <DESC>. By performing random sampling, a <b>heterogeneous subgraph</b> is obtained. Separately <b>nodes of different types</b> are: 1. <graph>, 2. <graph>... Please sequentially <b>provide the types</b> for the node sequences.	Based on graph tokens, types of the graph tokens should be <b>1. movie, 2. ....</b>
(b) Homogenous Instruction Pre-training		
central_nodes: ("paper": [1, ..., n]), num_neighbors: [10, 10]	Given a heterogeneous graph about <b>papers</b> , there are <b>4 types of nodes</b> : <DESC>. ..., a <b>heterogeneous subgraph</b> is obtained. The <b>nodes for "paper"</b> are: <graph>. Also, a list of textual descriptions for the papers are: <DESC>. Please <b>reorder the text list based on the order of graph tokens</b> .	The matching of graph tokens and papers should be: <b>&lt;ANSWER&gt;</b> .
(c) Heterogeneous Supervised Fine-Tuning		
central_nodes: ("movie": [1]), num_neighbors: [10, 10]	Given a heterogeneous graph about <b>movies</b> , there are <b>3 types of nodes</b> : <DESC>. ..., a <b>heterogeneous subgraph</b> is obtained. There are <b>nodes of different types</b> : "movie" nodes: <graph>, <DESC> where the 0-th node is the central node. "actor" nodes: <graph>; "director" nodes: <graph>. <b>Which of the following classes does this movie belong to</b> : action, comedy, drama?	Based on the given information, the likely category for <b>movie</b> is <b>Action</b> .



**Figure 2: Mixture-of-Thought (MoT) Augmentation**

and homogeneous relation awareness, we design human questions that are rich in heterogeneity. These questions contain sequences of graph tokens from different types, indicated by multiple occurrences of the <graph> token. Additionally, the human question includes pertinent auxiliary information pertaining to the target node. The designs of these instructions are presented in Figure 1.

### 3.3 Mixture-of-Thought (MoT) for Graph Instruction Augmentation

In practical scenarios of heterogeneous graph learning, data scarcity often poses a challenge. This is especially true when using heterogeneous graphs to model cold-start users/items in recommendation systems, where sparse user interaction data limits the availability of supervised signals. To address the issue of data sparsity, we propose enhancing our HiGPT by incorporating augmented graph instructions. Drawing inspiration from previous works [24], we introduce a novel method for instruction augmentation in the context of heterogeneous graph instruction tuning. This method utilizes prompt engineering techniques, particularly Mixture-of-Thought (MoT), to generate diverse and informative instructions. The goal is to effectively overcome the challenges posed by data scarcity. By incorporating augmented graph instructions, we expect our model enhancement to effectively handle data sparsity.

**3.3.1 Mixture-of-Thought (MoT) Prompting.** Our focus is on designing and optimizing prompts to effectively utilize language models [15, 25, 35, 42]. We employ several techniques to enhance

language models: i) **Chain-of-Thought (CoT)** [35]: CoT prompts introduce intermediate steps, enabling complex reasoning and sophisticated capabilities. ii) **Tree-of-Thought (ToT)** [42]: ToT maintains a tree structure of coherent language sequences called thoughts. These thoughts serve as systematic intermediate steps for problem-solving. iii) **PanelGPT** [25]: PanelGPT incorporates panel discussions among language models, enhancing the prompt engineering process through collaboration. iv) **Generated Knowledge Prompting (GKP)** [15]: GKP involves incorporating additional knowledge into prompts for enhancement. By leveraging these techniques, our objective is to augment heterogeneous graph instructions, especially in scenarios with limited data availability.

**3.3.2 Instruction Augmentation with Prior Knowledge.** We utilize seven instruction augmentation strategies, each generating seven augmented instructions for every question-answer pair, incorporating the characteristics of Mixture-of-Thought (MoT). However, closed-source language models such as ChatGPT may produce incorrect answers, resulting in flawed guidance. To overcome this issue, we propose incorporating prior knowledge, specifically the correct answer, into the prompt. It enables LLM to simulate generating the correct answer and produce intermediate reasoning steps using different MoT methods, as shown in Figure 1 and Appendix.

## 4 EVALUATION

To assess the effectiveness of our HiGPT model, our experiments are designed to address the following research questions:

- **RQ1:** How does the performance of our HiGPT compare to that of baseline methods in both few-shot and zero-shot scenarios?
- **RQ2:** To what extent do the key components of HiGPT contribute to its overall performance across various settings?
- **RQ3:** Can the HiGPT’s universal heterogeneity knowledge be leveraged to achieve graph in-context learning solely through graph instruction examples, without any model optimization?

### 4.1 Experimental Settings

**4.1.1 Experimental Datasets.** The experiments were conducted on three benchmark datasets, *i.e.*, IMDB [8], DBLP [8], and ACM [33]. **IMDB** is an extensive dataset that focuses on online movies and television programs. It encompasses 4278 movies, 2081 directors,

and 5257 actors. Each movie is categorized into one of three classes: Action, Comedy, or Drama. **DBLP**, on the other hand, consists of 4057 authors, 14328 papers, 7723 terms, and 20 publication venues. This dataset was gathered from a computer science bibliography website and the authors are distributed among four research areas: Database, Data Mining, Artificial Intelligence, and Information Retrieval. Lastly, the **ACM** dataset comprises 3025 papers, 5835 authors, and 56 subjects. The papers are classified into three classes: Database, Wireless Communication, and Data Mining.

**4.1.2 Evaluation Protocols.** To ensure consistency in the feature dimension of nodes across all datasets, we utilize a pre-trained Sentence-BERT to encode nodes of all types from each dataset into a standardized dimension. For the supervised few-shot node classification, we randomly select 1, 3, 5, 10, 20, 40, or 60 labeled nodes per class as our training set. Additionally, we reserve 1,000 nodes for validation and another 1,000 nodes for testing purposes. Our evaluation metrics encompass Micro-F1, Macro-F1, and AUC.

**4.1.3 Compared Baseline Methods.** For our comprehensive performance comparison, we evaluate various state-of-the-art methods from three different categories: i) The first category consists of representative homogeneous graph neural networks, including **SAGE** [9] and **GAT** [28]. ii) The second category includes approaches utilizing message-passing mechanisms in heterogeneous graph neural networks. This category features models such as **HAN** [33], **HGT** [10], and **HetGNN** [44]. iii) The third category focuses on self-supervised techniques for heterogeneous graph learning. This category incorporates generative strategies like **HGMAE** [27], as well as contrastive schemes such as **DMGI** [19] and **HeCo** [34].

**4.1.4 Implementation Details.** In Appendix Sec A.5.1, we offer comprehensive descriptions of the implementation details, including the datasets used, training hyperparameters, configurations of the base LLM, and more. These details provide a deeper understanding of our implementation approach.

## 4.2 Overall Performance Comparison (RQ1)

We performed node classification tasks on three datasets, exploring both few-shot and zero-shot settings. In the few-shot settings, our model was trained on the IMDB dataset with shot numbers ranging from 1 to 60, and evaluated on the IMDB test set of 1,000 samples [27, 34]. For the zero-shot settings, the model was trained on the IMDB dataset with the same shot numbers, and tested on separate test sets from the DBLP and ACM datasets, each containing 1,000 samples. To enable cross-dataset transferability in supervised heterogeneous Graph Neural Networks (GNNs), we unified node and edge categories, and utilized a classifier trained with transfer data to accommodate variations in class quantities across datasets.

For self-supervised methods focused on learning embeddings for downstream heterogeneous graph nodes, we excluded the zero-shot settings. The overall performance is partially shown in Table 2, with detailed results in Table 13 in the appendix. "-std" and "-cot" notations represent the standard test prompt with direct answers and the prompt with a Chain-of-Thought (CoT) feature, respectively. These details provide insights into our node classification experiments in both supervised and zero-shot settings.

**Superiority of HiGPT in Few-Shot Settings.** HiGPT outperforms state-of-the-art baselines consistently in supervised settings,

even with only one sample per class. The success can be attributed to our effective instruction-tuning on a large-scale heterogeneous graph corpus. This enables the LLM to extract valuable and transferable heterogeneous structural information from graph tokens, resulting in a significant performance boost in downstream tasks. Additionally, our proposed MoT graph instruction augmentation method enhances the LLM with diverse mixed reasoning capabilities without the need for additional supervision signals. As a result, it effectively mitigates the data scarcity in few-shot scenarios.

**Zero-shot Superiority of HiGPT.** In zero-shot settings, our HiGPT surpasses baselines with significant improvements. Unlike traditional models constrained by training graph types, our approach leverages an in-context heterogeneous graph tokenizer. This tokenizer adapts tokenization based on the input graph, allowing the LLM to seamlessly combine graph tokens that capture higher-order structural features with its semantic understanding. As a result, our model effectively overcomes the limitation of graph heterogeneity shift, performing exceptionally well even in cross-domain scenarios. This showcases the remarkable adaptability of our HiGPT.

**Effectiveness of Mixture-of-Thought Augmentation.** Through the implementation of the MoT approach, our model harnesses the varied reasoning capabilities of the formidable LLM (specifically, GPT-3.5) and seamlessly integrates them into our more compact language model. This integration serves to bolster our model’s ability to effectively navigate data scarcity and elevate its performance in situations characterized by limited supervised signals. The MoT technique assumes a pivotal role in generating dynamic and diverse instructions, thereby offsetting the dearth of data and empowering our model to make notably precise predictions across both supervised and zero-shot settings.

## 4.3 Model Ablation Test (RQ2)

To evaluate the proposed modules’ effectiveness, we individually remove the key techniques in HiGPT. The results are summarized in Table 3. Here are the ablated variants and the key conclusions:

- **Effect of Heterogeneous Graph Instruction-Tuning.** To validate the effectiveness of instruction tuning in the tuning stage on the large heterogeneous graph corpus, we generated the "w/o S1" variant by directly tuning the instructions solely on the downstream task data. Through experiments with different epoch settings (15, 50, and 100), we observed that models tuned solely on the downstream task data failed to provide complete and accurate answers in all cases. However, our HiGPT achieved state-of-the-art performance within just 15 epochs across all settings. This success can be attributed to the fact that our HiGPT learns from a vast heterogeneous graph context corpus, enabling it to understand and extract crucial structural information. As a result, in the second stage, our HiGPT requires only a minimal amount of supervised data (even in a 1-shot scenario) to quickly align with the downstream task. Conversely, directly aligning LLMs with sparse supervised data proves to be challenging.
- **Effect of In-Context Heterogeneous Graph Tokenizer.** We tested the necessity of incorporating heterogeneous graph structural information from our in-context tokenizer. By excluding the introduction of heterogeneous graph tokens and solely training the LLM’s embeddings weights on the downstream data,



**Table 2: Performance comparison on node classification tasks in both few-shot and zero-shot settings. However, since SSL methods focus on learning embeddings from downstream graphs, we excluded the zero-shot settings for them ("-").**

Datasets	Metric	train-on	test-on	SAGE	GAT	HAN	HGT	HeGNN	DMGI	HGMAE	HeCo	HiGPT-std	HiGPT-cot
Supervised	Mi-F1	IMDB-1	IMDB-1000	0.4663±0.0025	0.4567±0.0122	0.4890±0.0271	0.4977±0.0186	0.4790±0.0134	0.4570±0.0126	0.3609±0.0145	0.3874±0.0159	0.5099±0.0073	<b>0.5360±0.0065</b>
		IMDB-5	IMDB-1000	0.5010±0.0051	0.5170±0.0029	0.4840±0.0094	0.5003±0.0093	0.5020±0.0045	0.4413±0.0173	0.3652±0.0062	0.3385±0.0169	0.6180±0.0027	<b>0.6320±0.0085</b>
		IMDB-20	IMDB-1000	0.5930±0.0093	0.6117±0.0012	0.5763±0.0046	0.5750±0.0065	0.5957±0.0054	0.5497±0.0256	0.4107±0.0106	0.3781±0.0148	0.6090±0.0255	<b>0.6440±0.0075</b>
		IMDB-40	IMDB-1000	0.6170±0.0112	0.6261±0.0015	0.6198±0.0025	0.5923±0.0040	0.6177±0.0046	0.5813±0.0033	0.3946±0.0067	0.3927±0.0134	0.6260±0.0057	<b>0.6280±0.0071</b>
	Ma-F1	IMDB-1	IMDB-1000	0.4425±0.0068	0.3974±0.0183	0.4229±0.0104	0.4020±0.0112	0.4456±0.0036	0.4083±0.0288	0.3573±0.0117	0.4023±0.0137	0.4986±0.0141	<b>0.5247±0.0061</b>
		IMDB-5	IMDB-1000	0.4613±0.0086	0.4767±0.0098	0.4695±0.0037	0.4676±0.0153	0.4677±0.0145	0.4254±0.0124	0.3500±0.0080	0.3468±0.0213	0.6111±0.0091	<b>0.6243±0.0060</b>
		IMDB-20	IMDB-1000	0.5953±0.0095	0.6121±0.0024	0.5756±0.0051	0.5723±0.0056	0.5969±0.0055	0.5495±0.0270	0.4065±0.0089	0.3904±0.0172	0.6068±0.0146	<b>0.6398±0.0083</b>
		IMDB-40	IMDB-1000	0.6182±0.0107	0.6254±0.0009	0.6224±0.0057	0.5909±0.0068	0.6234±0.0038	0.5786±0.0064	0.3866±0.0072	0.3988±0.0147	<b>0.6265±0.0090</b>	0.6237±0.0059
	AUC	IMDB-1	IMDB-1000	0.6079±0.0061	0.6151±0.0065	0.6234±0.0252	0.6249±0.0170	0.6107±0.0075	0.5780±0.0130	0.5274±0.0058	0.5712±0.0099	0.6565±0.0146	<b>0.6685±0.0037</b>
		IMDB-5	IMDB-1000	0.6309±0.0049	0.6372±0.0012	0.6102±0.0059	0.6197±0.0152	0.6290±0.0022	0.5832±0.0132	0.5262±0.0041	0.5067±0.0228	0.7308±0.0125	<b>0.7310±0.0086</b>
		IMDB-20	IMDB-1000	0.6976±0.0064	0.7122±0.0020	0.6815±0.0052	0.6801±0.0048	0.7005±0.0030	0.6657±0.0179	0.5766±0.0064	0.5541±0.0145	0.7227±0.0034	<b>0.7424±0.0113</b>
		IMDB-40	IMDB-1000	0.7171±0.0069	0.7210±0.0014	0.7204±0.0015	0.6970±0.0060	0.7145±0.0035	0.6860±0.0027	0.5488±0.0049	0.5653±0.0105	0.7323±0.0036	<b>0.7331±0.0074</b>
Zero-shot	Mi-F1	IMDB-1	DBLP-1000	0.2353±0.0372	0.1893±0.0373	0.2653±0.0203	0.2573±0.0519	0.2900±0.0638	-	-	-	0.3180±0.0072	<b>0.3500±0.0073</b>
		IMDB-5	DBLP-1000	0.2607±0.0082	0.2737±0.0176	0.2577±0.0094	0.2453±0.0458	0.2427±0.0452	-	-	-	0.3180±0.0044	<b>0.3620±0.0047</b>
		IMDB-20	DBLP-1000	0.2810±0.0289	0.2780±0.0033	0.2710±0.0000	0.2803±0.0208	0.2333±0.0353	-	-	-	0.3840±0.0088	<b>0.4180±0.0083</b>
		IMDB-40	DBLP-1000	0.2400±0.0324	0.2847±0.0053	0.2710±0.0000	0.2937±0.0005	0.2027±0.0345	-	-	-	0.3320±0.0087	<b>0.3630±0.0045</b>
	Ma-F1	IMDB-1	DBLP-1000	0.0963±0.0132	0.1169±0.0089	0.1047±0.0063	0.1016±0.0169	0.1778±0.0629	-	-	-	0.2048±0.0068	<b>0.2472±0.0070</b>
		IMDB-5	DBLP-1000	0.1042±0.0028	0.1291±0.0145	0.1024±0.0030	0.1138±0.0296	0.0971±0.0148	-	-	-	0.1917±0.0046	<b>0.2773±0.0085</b>
		IMDB-20	DBLP-1000	0.1448±0.0573	0.1274±0.0060	0.1066±0.0000	0.1143±0.0116	0.1008±0.0191	-	-	-	0.3142±0.0074	<b>0.3733±0.0051</b>
		IMDB-40	DBLP-1000	0.1068±0.0060	0.1588±0.0078	0.1066±0.0000	0.1268±0.0105	0.0984±0.0161	-	-	-	0.2331±0.0069	<b>0.2912±0.0056</b>
	AUC	IMDB-1	DBLP-1000	0.4999±0.0001	0.4513±0.0295	0.5000±0.0000	0.5000±0.0000	0.5206±0.0306	-	-	-	0.5222±0.0069	<b>0.5406±0.0040</b>
		IMDB-5	DBLP-1000	0.4978±0.0030	0.4908±0.0078	0.5000±0.0000	0.5031±0.0043	0.4998±0.0003	-	-	-	0.5184±0.0081	<b>0.5493±0.0091</b>
		IMDB-20	DBLP-1000	0.5154±0.0213	0.4918±0.0020	0.5000±0.0000	0.5011±0.0016	0.4957±0.0060	-	-	-	0.5669±0.0041	<b>0.5907±0.0089</b>
		IMDB-40	DBLP-1000	0.5027±0.0031	0.4976±0.0021	0.5000±0.0000	0.5008±0.0006	0.4884±0.0164	-	-	-	0.5296±0.0070	<b>0.5508±0.0086</b>
Mi-F1	IMDB-1	ACM-1000	0.3293±0.0418	0.3567±0.0053	0.3567±0.0111	0.3240±0.0014	0.3743±0.0434	-	-	-	0.4160±0.0106	<b>0.4540±0.0089</b>	
	IMDB-5	ACM-1000	0.3820±0.0113	0.3787±0.0057	0.3630±0.0086	0.3160±0.0169	0.3583±0.0198	-	-	-	0.4580±0.0173	<b>0.4880±0.0131</b>	
	IMDB-20	ACM-1000	0.2807±0.0074	0.3013±0.0188	0.3133±0.0031	0.3530±0.0000	0.2840±0.0226	-	-	-	<b>0.5080±0.0129</b>	0.5030±0.0064	
	IMDB-40	ACM-1000	0.3173±0.0005	0.2393±0.0144	0.2697±0.0194	0.3560±0.0099	0.3180±0.0116	-	-	-	0.4750±0.0149	<b>0.5050±0.0127</b>	
Ma-F1	IMDB-1	ACM-1000	0.2647±0.0269	0.2908±0.0131	0.2250±0.0416	0.1631±0.0005	0.3139±0.0468	-	-	-	0.3949±0.0078	<b>0.4177±0.0124</b>	
	IMDB-5	ACM-1000	0.3208±0.0130	0.3009±0.0137	0.2782±0.0026	0.1969±0.0301	0.3087±0.0225	-	-	-	0.4336±0.0085	<b>0.4510±0.0114</b>	
	IMDB-20	ACM-1000	0.2694±0.0091	0.2422±0.0098	0.2412±0.0050	0.2094±0.0501	0.2715±0.0181	-	-	-	<b>0.4964±0.0075</b>	0.4877±0.0070	
	IMDB-40	ACM-1000	0.3117±0.0017	0.2141±0.0071	0.2313±0.0132	0.2749±0.0122	0.3144±0.0017	-	-	-	0.4176±0.0116	<b>0.4585±0.0089</b>	
AUC	IMDB-1	ACM-1000	0.4934±0.0247	0.5248±0.0038	0.5128±0.0086	0.5000±0.0000	0.5318±0.0295	-	-	-	0.5672±0.0040	<b>0.5969±0.0082</b>	
	IMDB-5	ACM-1000	0.5433±0.0082	0.5415±0.0047	0.5282±0.0073	0.4950±0.0134	0.5256±0.0145	-	-	-	0.5991±0.0103	<b>0.6224±0.0054</b>	
	IMDB-20	ACM-1000	0.4601±0.0048	0.4772±0.0137	0.4877±0.0029	0.5038±0.0053	0.4625±0.0163q	-	-	-	<b>0.6352±0.0094</b>	0.6318±0.0068	
	IMDB-40	ACM-1000	0.4867±0.0013	0.4320±0.0108	0.4545±0.0146	0.5148±0.0043	0.4872±0.0026	-	-	-	0.6138±0.0047	<b>0.6360±0.0051</b>	

**Table 3: Ablation study of our HiGPT.**

Datasets	Metric	train-on	test-on	w/o S1	w/o HG	w/o IA	HiGPT
Supervised	Mi-F1	IMDB-1	IMDB-1000	fail	0.3740	0.4260	<b>0.5360</b>
		IMDB-3	IMDB-1000	fail	0.5000	0.4540	<b>0.5730</b>
		IMDB-10	IMDB-1000	fail	0.5660	0.4380	<b>0.5810</b>
		IMDB-20	IMDB-1000	fail	0.5640	0.5620	<b>0.6440</b>
	Ma-F1	IMDB-1	IMDB-1000	fail	0.2433	0.3978	<b>0.5247</b>
		IMDB-3	IMDB-1000	fail	0.4969	0.4289	<b>0.5591</b>
		IMDB-10	IMDB-1000	fail	0.5619	0.3966	<b>0.5762</b>
		IMDB-20	IMDB-1000	fail	0.5636	0.5364	<b>0.6398</b>
	AUC	IMDB-1	IMDB-1000	fail	0.5195	0.6023	<b>0.6685</b>
		IMDB-3	IMDB-1000	fail	0.6340	0.6186	<b>0.6935</b>
		IMDB-10	IMDB-1000	fail	0.6790	0.5903	<b>0.6875</b>
		IMDB-20	IMDB-1000	fail	0.6891	0.6840	<b>0.7424</b>
Zero-shot	Mi-F1	IMDB-1	DBLP-1000	fail	0.2980	0.2800	<b>0.3500</b>
		IMDB-3	DBLP-1000	fail	0.3430	0.3180	<b>0.3660</b>
		IMDB-10	DBLP-1000	fail	0.3640	0.3140	<b>0.4020</b>
		IMDB-20	DBLP-1000	fail	0.3920	0.2800	<b>0.4180</b>
	Ma-F1	IMDB-1	DBLP-1000	fail	0.2444	0.2145	<b>0.2472</b>
		IMDB-3	DBLP-1000	fail	0.2768	0.2503	<b>0.2814</b>
		IMDB-10	DBLP-1000	fail	0.3211	0.2581	<b>0.3386</b>
		IMDB-20	DBLP-1000	fail	0.3689	0.1836	<b>0.3733</b>
	AUC	IMDB-1	DBLP-1000	fail	0.5275	0.5035	<b>0.5406</b>
		IMDB-3	DBLP-1000	fail	0.5422	0.5286	<b>0.5524</b>
		IMDB-10	DBLP-1000	fail	0.5636	0.5269	<b>0.5777</b>
		IMDB-20	DBLP-1000	fail	0.5834	0.4995	<b>0.5907</b>

we obtained a variant called "w/o HG". Our HiGPT consistently outperformed this variant across different shot settings, especially in scenarios with limited samples (e.g., 1 or 3 shots). This improvement is attributed to the introduction of graph tokens, which enable the LLM to extract high-dimensional heterogeneous structural information from the in-context graph tokenizer. This enhanced understanding significantly improves the LLM's accuracy, particularly with sparse supervised signals.

- **Effect of MoT Instruction Augmentation.** To verify the effectiveness of the MoT graph instruction augmentation strategy, we trained the variant "-IA" using only direct-answer instructions. Results showed a significant drop in model performance without

instruction augmentation, highlighting its importance in tackling the scarcity of labels in downstream tasks. Additionally, HiGPT's superior performance in zero-shot settings can be attributed to its enhanced reasoning ability, acquired through training with diverse reasoning instructions. This improved capacity enables effective cross-dataset and cross-domain transfer.

#### 4.4 Graph In-Context Learning (RQ3)

In-context learning (ICL)[18] is a method for adapting large language models (LLMs) to new tasks without gradient updates, using a prompt with task examples. In this subsection, we explore the impact of Graph In-Context Learning on improving HiGPT's performance. We conduct comprehensive tests by adding prefatory examples from the training set to models trained with different shots of IMDB data. We randomly sampled training examples corresponding to the test data. "-ICL-1" and "-ICL-2" denote one and two prefatory examples, respectively. "-ICL-DBLP" signifies the inclusion of DBLP examples before the ACM test prompt. The results, depicted in Figure 3, reveal the following observations:

**1-shot Beat 60-shot with Graph ICL in HiGPT.** Results show that, even with just a single example, most 1-shot models using Graph ICL consistently outperform 60-shot models without further training in both supervised and zero-shot settings. Increasing the number of examples enhances the effect of in-context learning. This improvement can be attributed to HiGPT's two-stage instruction tuning process, which enables it to understand and analyze heterogeneous graph tokens, benefiting downstream tasks. By providing question-and-answer examples with graph tokens, the model gains a deeper understanding of the graph-text relationship. Analyzing and emulating these examples leads to more accurate responses.

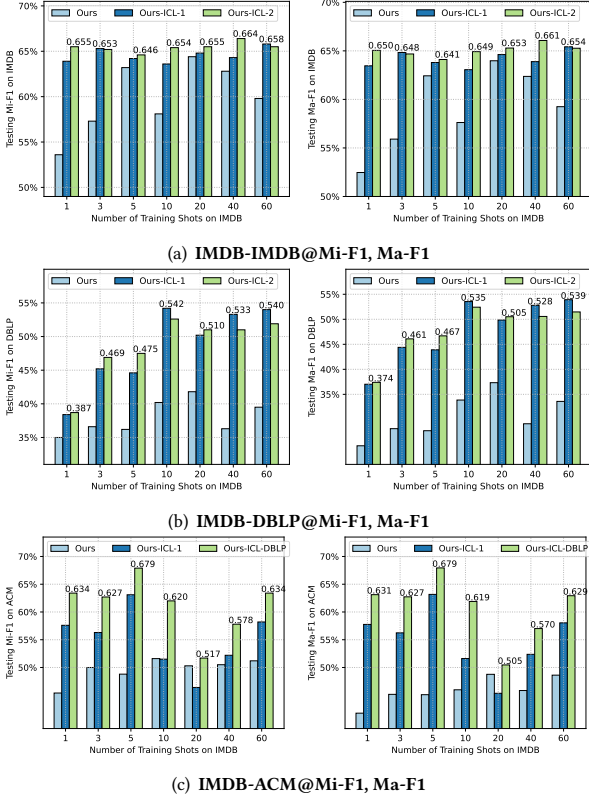


Figure 3: Graph In-Context Learning of our HiGPT.

**Enhanced Transferability with our Graph ICL.** The advantages of our Graph ICL in HiGPT are particularly evident in zero-shot transfer scenarios. This indicates that the Graph ICL approach significantly improves HiGPT’s transferability without the need to optimize model parameters. Our HiGPT does not simply overfit to a single dataset but develops the ability to analyze text alongside heterogeneous graph tokens. By incorporating graph examples from other datasets, the model effortlessly transfers this analytical capability, demonstrating strong transfer learning capacity.

**Benefit of Irrelevant Graph Examples.** We experimented with HiGPT with irrelevant graph examples, like using DBLP Q&A examples for testing on the ACM dataset. Surprisingly, using DBLP graph examples yielded the best results. Despite different target tasks, our HiGPT effectively leverages in-context information from heterogeneous graph tokens, enhancing downstream tasks. This confirms that our HiGPT learns valuable information from heterogeneous graph structures, rather than relying solely on text. Using ACM’s own examples did not perform as well due to a deficiency in encoding the ACM graph in the alignment and stage 1 process. However, the DBLP examples mitigated this issue to some extent.

## 4.5 Case Study

We perform a case study to showcase our HiGPT’s robust generalization in understanding complex graph structures with diverse nodes and connections. Our model generates graph-aware predictions and responses, demonstrating its profound comprehension and awareness of graph-related aspects. Furthermore, we validate the positive impact of our MoT instruction augmentation. For more comprehensive details, please refer to Appendix Section A.5.2.

## 5 RELATED WORK

**Heterogeneous Graph Neural Networks.** Heterogeneous Graph Neural Networks (HGNNs) capture complex relationships and diverse semantics among entities in a heterogeneous graph [2, 6, 44]. They use specialized message functions and aggregation rules to model relation heterogeneity. Existing models, such as MAGNN [6] and HetGNN [44], leverage metapaths to capture composite relations and guide neighbor selection. Heterogeneous graph convolution frameworks like HeteGCN [21] draws inspiration from graph convolutional networks. Heterogeneous graph attention networks, including HAN [32], HGT [10], and HGAT [14], use attention mechanisms to effectively capture and aggregate information from different node types. However, most HGNNs require sufficient labels to learn accurate graph representations.

**Heterogeneous Graph Self-Supervised Learning.** Recent research has addressed the limited availability of labeled data by incorporating self-supervised learning techniques into heterogeneous graph modeling [13, 30]. Contrastive and generative approaches have proven effective in augmenting data. Contrastive learning methods like DMGI [19] and HeCo [34] bring similar instances closer and push dissimilar instances apart in a latent space, capturing relevant patterns and structure. Generative learning approaches such as HGMAE [27] use masked autoencoders to reconstruct heterogeneous graphs. However, these approaches still have limitations in handling relation heterogeneity shift across downstream tasks, potentially leading to poor generalization ability.

**Large Language Models for Graph Data.** Recent research has combined large language models (LLMs) and graph models to understand complex relationships in graph data [1, 23, 37, 43]. Two primary approaches integrate graph structural information: utilizing textual prompts and incorporating graph embeddings as input tokens. Chen et al. [3] craft tailored prompts for graph learning tasks, while InstructGLM [43] and GraphGPT [26] propose to integrate prompt instructions with graph embeddings for fine-tuning LLM. Moreover, advancements have introduced LLMs to improve the reasoning capabilities of models when working with graph-structured data. Prominent examples include the works of Fatemi et al. [7] and Chai et al. [1]. However, existing LLM-enhanced graph models have primarily focused on homogeneous graphs, overlooking the inherent heterogeneity in real-world graphs. This calls for further exploration of heterogeneous graph language models with strong generalization abilities across diverse downstream tasks.

## 6 CONCLUSION

This work introduce HiGPT, a general and versatile graph model that offers the ability to learn from diverse heterogeneous graphs without the need for downstream fine-tuning processes. To address distribution shifts in heterogeneity, we propose an in-context heterogeneous graph tokenizer that captures semantic relationships across different heterogeneous graphs, facilitating seamless model adaptation. By incorporating the heterogeneity-aware graph instructions into our HiGPT, the model becomes proficient in comprehending intricate relation heterogeneity and accurately discerning between various types of graph tokens. Our proposed framework has undergone extensive evaluations across diverse scenarios, demonstrating outstanding generalization performance.



## REFERENCES

- [1] Z. Chai, T. Zhang, L. Wu, K. Han, X. Hu, X. Huang, and Y. Yang. Graphllm: Boosting graph reasoning ability of large language model. *arXiv preprint arXiv:2310.05845*, 2023.
- [2] M. Chen, C. Huang, L. Xia, W. Wei, Y. Xu, and R. Luo. Heterogeneous graph contrastive learning for recommendation. In *WSDM*, pages 544–552, 2023.
- [3] Z. Chen, H. Mao, H. Li, et al. Exploring the potential of large language models (llms) in learning on graphs. *CoRR*, abs/2307.03393, 2023.
- [4] Y. Dong, N. V. Chawla, and A. Swami. metapath2vec: Scalable representation learning for heterogeneous networks. In *KDD*, pages 135–144, 2017.
- [5] A. El-Kishky, T. Markovich, S. Park, C. Verma, B. Kim, R. Eskander, Y. Malkov, F. Portman, S. Samaniego, Y. Xiao, et al. Twhin: Embedding the twitter heterogeneous information network for personalized recommendation. In *KDD*, pages 2842–2850, 2022.
- [6] S. Fan, J. Zhu, X. Han, C. Shi, L. Hu, B. Ma, and Y. Li. Metapath-guided heterogeneous graph neural network for intent recommendation. In *KDD*, pages 2478–2486, 2019.
- [7] B. Fatemi, J. Halcrow, and B. Perozzi. Talk like a graph: Encoding graphs for large language models. *arXiv preprint arXiv:2310.04560*, 2023.
- [8] X. Fu, J. Zhang, Z. Meng, and I. King. MAGNN: metapath aggregated graph neural network for heterogeneous graph embedding. In *WWW*, pages 2331–2341. ACM / IW3C2, 2020.
- [9] W. L. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *NeurIPS*, pages 1024–1034, 2017.
- [10] Z. Hu, Y. Dong, K. Wang, and Y. Sun. Heterogeneous graph transformer. In *WWW*, pages 2704–2710. ACM / IW3C2, 2020.
- [11] D. Hwang, J. Park, S. Kwon, K. Kim, J.-W. Ha, and H. J. Kim. Self-supervised auxiliary learning with meta-paths for heterogeneous graphs. *NeurIPS*, 33:10294–10305, 2020.
- [12] D. Jin, C. Huo, C. Liang, and L. Yang. Heterogeneous graph neural network via attribute completion. In *WWW*, pages 391–400, 2021.
- [13] B. Jing, S. Feng, Y. Xiang, X. Chen, Y. Chen, and H. Tong. X-goal: multiplex heterogeneous graph prototypical contrastive learning. In *CIKM*, pages 894–904, 2022.
- [14] H. Linmei, T. Yang, C. Shi, H. Ji, and X. Li. Heterogeneous graph attention networks for semi-supervised short text classification. In *EMNLP*, pages 4821–4830, 2019.
- [15] J. Liu, A. Liu, X. Lu, S. Welleck, P. West, R. L. Bras, Y. Choi, and H. Hajishirzi. Generated knowledge prompting for commonsense reasoning. In *ACL (1)*, pages 3154–3169. Association for Computational Linguistics, 2022.
- [16] Q. Lv, M. Ding, Q. Liu, Y. Chen, W. Feng, S. He, C. Zhou, J. Jiang, Y. Dong, and J. Tang. Are we really making much progress? revisiting, benchmarking and refining heterogeneous graph neural networks. In *KDD*, pages 1150–1160, 2021.
- [17] A. Ma, X. Wang, J. Li, C. Wang, T. Xiao, Y. Liu, H. Cheng, J. Wang, Y. Li, Y. Chang, et al. Single-cell biological network inference using a heterogeneous graph transformer. *Nature Communications*, 14(1):964, 2023.
- [18] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, and L. Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *EMNLP*, pages 11048–11064, 2022.
- [19] C. Park, D. Kim, J. Han, and H. Yu. Unsupervised attributed multiplex network embedding. In *AAAI*, volume 34, pages 5371–5378, 2020.
- [20] A. Radford, J. W. Kim, C. Hallacy, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pages 8748–8763. PMLR, 2021.
- [21] R. Ragesh, S. Sellamanickam, A. Iyer, R. Bairi, and V. Lingam. Hetegcn: heterogeneous graph convolutional networks for text classification. In *WSDM*, pages 860–868, 2021.
- [22] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP*. Association for Computational Linguistics, 11 2019.
- [23] X. Ren, W. Wei, L. Xia, L. Su, S. Cheng, J. Wang, D. Yin, and C. Huang. Representation learning with large language models for recommendation. *arXiv preprint arXiv:2310.15950*, 2023.
- [24] K. Shridhar, A. Stolfo, and M. Sachan. Distilling reasoning capabilities into smaller language models. In *ACL*, pages 7059–7073, 2023.
- [25] H. Sun, A. Hüyük, and M. van der Schaar. Query-dependent prompt evaluation and optimization with offline inverse rl. *arXiv e-prints*, pages arXiv-2309, 2023.
- [26] J. Tang, Y. Yang, W. Wei, L. Shi, L. Su, S. Cheng, D. Yin, and C. Huang. Graphgpt: Graph instruction tuning for large language models, 2023.
- [27] Y. Tian, K. Dong, C. Zhang, C. Zhang, and N. V. Chawla. Heterogeneous graph masked autoencoders. In *AAAI*, volume 37, pages 9997–10005, 2023.
- [28] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, et al. Graph attention networks. In *ICLR (Poster)*. OpenReview.net, 2018.
- [29] H. Wang, H. Ren, and J. Leskovec. Relational message passing for knowledge graph completion. In *KDD*, pages 1697–1707, 2021.
- [30] P. Wang, K. Agarwal, C. Ham, S. Choudhury, and C. K. Reddy. Self-supervised learning of contextual embeddings for link prediction in heterogeneous networks. In *WWW*, pages 2946–2957, 2021.
- [31] X. Wang, D. Bo, C. Shi, S. Fan, Y. Ye, and S. Y. Philip. A survey on heterogeneous graph embedding: methods, techniques, applications and sources. *Transactions on Big Data (TBD)*, 9(2):415–436, 2022.
- [32] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu. Heterogeneous graph attention network. In *WWW*, pages 2022–2032, 2019.
- [33] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, et al. Heterogeneous graph attention network. In *WWW*, pages 2022–2032. ACM, 2019.
- [34] X. Wang, N. Liu, H. Han, and C. Shi. Self-supervised heterogeneous graph neural network with co-contrastive learning. In *KDD*, pages 1726–1736, 2021.
- [35] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.
- [36] W. Wei, C. Huang, L. Xia, Y. Xu, J. Zhao, and D. Yin. Contrastive meta learning with behavior multiplicity for recommendation. In *WSDM*, pages 1120–1128, 2022.
- [37] W. Wei, X. Ren, J. Tang, Q. Wang, L. Su, S. Cheng, J. Wang, D. Yin, and C. Huang. Llmrec: Large language models with graph augmentation for recommendation. *arXiv preprint arXiv:2311.00423*, 2023.
- [38] Z. Wen and Y. Fang. Augmenting low-resource text classification with graph-grounded pre-training and prompting. In *SIGIR*, 2023.
- [39] H. Xuan, Y. Liu, B. Li, and H. Yin. Knowledge enhancement for contrastive multi-behavior recommendation. In *WSDM*, pages 195–203, 2023.
- [40] C. Yang, Y. Xiao, Y. Zhang, Y. Sun, and J. Han. Heterogeneous network representation learning: A unified framework with survey and benchmark. *Transactions on Knowledge and Data Engineering (TKDE)*, 34(10):4854–4873, 2020.
- [41] Y. Yang, Z. Guan, Z. Wang, W. Zhao, C. Xu, W. Lu, and J. Huang. Self-supervised heterogeneous graph pre-training based on structural clustering. *NeurIPS*, 35:16962–16974, 2022.
- [42] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *CoRR*, abs/2305.10601, 2023.
- [43] R. Ye, C. Zhang, R. Wang, S. Xu, and Y. Zhang. Natural language is all a graph needs. *arXiv preprint arXiv:2308.07134*, 2023.
- [44] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla. Heterogeneous graph neural network. In *KDD*, pages 793–803. ACM, 2019.
- [45] J. Zhao, X. Wang, C. Shi, B. Hu, G. Song, and Y. Ye. Heterogeneous graph structure learning for graph neural networks. In *AAAI*, volume 35, pages 4697–4705, 2021.

## A APPENDIX

In the supplementary materials, we provide detailed information pertaining to our experiments. First, in Section A.1, we present the statistical information of the datasets used. Next, in Section A.2, we provide a comprehensive description of the baselines employed. In Section A.3, we outline the specific descriptions of nodes and edges in our text-enriched heterogeneity representations. Section A.4 elaborates on the templates for instructions and prompts used in our HiGPT, including a detailed explanation of MoT graph instruction augmentation, an instruction construction template for two-stage instruction tuning, and an instruction construction template for graph in-context learning. Additionally, Section A.5 presents additional experimental results, covering implementation details of our model, model case studies, overall performance, and comprehensive results of graph in-context learning.

### A.1 Detailed Statistics of Datasets

In Table 4, we present the statistical information of the datasets used in our experiments, where the types of the target nodes in each heterogeneous graph are highlighted with an underline.

**Table 4: Detailed statistics of utilized datasets.**

Dataset	# Nodes	# Edges	Metapaths	# Classes
ACM	<u>Paper</u> : 3025	P-A: 9949	PAP	3
	<u>Author</u> : 5959	P-S: 3025	PSP	
	<u>Subject</u> : 56	P-T: 255619	PTP	
	<u>Term</u> : 1902	P-P: 5343		
DBLP	<u>Author</u> : 4057	P-A: 19645	PAP	4
	<u>Paper</u> : 14328	P-C: 14328	APCPA	
	<u>Term</u> : 7723	P-T: 85810	APTPA	
	<u>Conference</u> : 20			
IMDB	<u>Movie</u> : 4278	M-D: 4278	MAM	3
	<u>Director</u> : 2081	M-A: 12828	MDM	
	<u>Actor</u> : 5257			

### A.2 Detailed Descriptions of Baselines

To conduct a thorough evaluation, our baseline set includes the following methods, which are presented below:

#### 1) Homogeneous Graph Neural Networks

- **SAGE** [9]: It was developed to facilitate the learning of inductive representations on large-scale homogeneous graphs, allowing for the generation of highly effective node embeddings for unseen data. However, we made adaptations and modifications to tailor it specifically to the unique demands of heterogeneous graphs.
- **GAT** [28]: It utilizes graph attention networks, which overcome the limitations of graph convolutional networks (GCN) by incorporating masked self-attention layers. What sets this method apart is its ability to selectively weigh the aggregated information from different nodes, thereby enhancing the message passing mechanism and refining the overall process.

#### 2) Heterogeneous Graph Neural Encoders.

- **HAN** [33]: It is a heterogeneous graph neural encoder that incorporates hierarchical attention mechanisms, including node-level and semantic-level attentions, to address the complexity of heterogeneous graphs with various types of nodes and links, thereby

improving the representation and interpretability of node embeddings through feature aggregation based on meta-path neighbors.

- **HGT** [10]: HGT is an advanced graph neural network framework designed to model the complexities of large-scale heterogeneous graphs, featuring type-dependent parameters for nodes and edges to enable heterogeneous attention, a relative temporal encoding to capture dynamic relationships, and an efficient graph sampling algorithm for scalable training.
- **HetGNN** [44]: HetGNN is a powerful heterogeneous graph neural network model that seamlessly integrates both the structural information and diverse content attributes of nodes. It achieves this by employing a two-module architecture for feature aggregation and incorporating a well-designed random walk sampling process. This comprehensive approach enables HetGNN to generate meaningful and informative node embeddings.

#### 3) SSL-enhanced Heterogeneous Graph Learning Approaches.

- **HGMAE** [27]: HGMAE is a generative SSL approach that addresses the challenges of capturing complex structures, incorporating diverse node attributes, and encoding node positions in heterogeneous graphs. It achieves this through innovative masking techniques and tailored training strategies, utilizing a heterogeneous graph masked autoencoder. HGMAE efficiently learns to generate meaningful representations while effectively preserving the rich information present in heterogeneous graphs.
- **DMGI** [19]: It is an effective unsupervised network embedding method for attributed multiplex networks that maximizes mutual information between graph patches and a global representation, integrating multiple relation-type embeddings with a consensus regularization framework and a universal discriminator, further enhanced by an attention mechanism to weigh relation types.
- **HeCo** [34]: This is a self-supervised heterogeneous graph neural network framework that employs a co-contrastive learning mechanism across two views (network schema and meta-path) to capture both local and high-order structures, with a view mask mechanism for effective cross-view supervision, enhanced by extensions for generating higher-quality negative samples.

### A.3 Detailed Descriptions of Text-Enriched Heterogeneity Representations

In Table 5 and 6, we present diverse descriptions for different node and edge types across three datasets, *i.e.*, IMDB, DBLP and ACM.

### A.4 Instruction and Prompting Templates

**A.4.1 Detailed prompt of MoT graph instruction augmentation.** In Table 7, we display all MoT graph instruction augmentation strategies that incorporate various prompt engineering techniques, including the prompting template to prompt GPT-3.5, and the template for constructing Instructions after obtaining the results.

**A.4.2 Instruction construction template for two stage instruction tuning.** In Table 8, we showcase the instruction template for the second stage of node classification for IMDB, where the prompting suffix allows our instruction to be combined with a variety of different prompt techniques for instruction tuning, thereby distilling multiple reasoning abilities for the powerful GPT-3.5. Table 9 presents two types of instruction templates for the first

**Table 5: Detailed Descriptions of Text-Enriched Heterogeneity Representations for IMDB and DBLP.**

Node (Edge) Type	Sets of Descriptions
(a) IMDB	
"Movie"	{"This node represents a movie"; "This is an action movie"; "This is a comedy movie"; "This is a drama movie"}
"Director"	{"This node represents a director"; "This is an action film director"; "The director specializes in action"; "This is a comedy film director"; "The director specializes in comedy"; "This is a drama film director"; "The director specializes in dram"}
"Actor"	{"This node represents an actor"; "This is an action film actor"; "The actor specializes in action"; "This is a comedy film actor"; "The actor specializes in comedy"; "This is a drama film actor"; "The actor specializes in drama"}
("movie", "to", "director")	{"The movie is directed by the director"; "The film features direction by the director"; "The movie's direction was in the hands of the director"; "The movie was helmed by the director"; "The film is a directorial effort by the director"; "The movie bears the directorial signature of the director"}
("movie", "to", "actor")	{"The movie has the actor"; "The movie features the actor"; "The film includes the actor in its lineup"; "The movie showcases the talent of the actor"; "The film's cast includes the actor"; "The movie presents the actor on its roster"}
("director", "to", "movie")	{"This director is responsible for the film's direction"; "The director take the helm for the movie"; "The director steers the production of the movie"; "The director in question crafts the narrative of the film"; "The director provides the creative direction for the film"; "The director orchestrates the making of the movie"}
("actor", "to", "movie")	{"The actor appears in the movie"; "The actor is part of the movie's cast"; "The actor stars in the movie"; "The actor is featured in the film"; "The actor has a role in the movie"}
(b) DBLP	
"paper"	{"This node represents a paper"; "A paper in the area of Database"; "A paper in the area of Data Mining"; "A paper in the area of AI"; "A paper in the area of Information Retrieval"; "A paper published in a conference"; "A paper in the area of computer science"}
"conference"	{"This node represents a conference"; "A conference in the area of Database"; "A conference in the area of Data Mining"; "A conference in the area of AI"; "A conference in the area of Information Retrieval"; "A conference about computer science"}
"author"	{"This node represents an author"; "An author in the area of Database"; "An author in the area of Data Mining"; "An author in the area of AI"; "An author in the area of Information Retrieval"; "An author in the area of computer science"}
"term"	{"This node represents a key term related to a paper"; "The term is included in a paper"; "The term is related to Database"; "The term is related to Data Mining"; "The term is related to AI"; "The term is related to Information Retrieval"}
("author", "to", "paper")	{"The author has the paper"; "The author publishes the paper"; "The author writes the paper"; "The author is the author of the paper"; "The author releases the paper"; "The author issues the paper"; "The author disseminates the paper"}
("paper", "to", "author")	{"The paper has the author"; "The paper is published by the author"; "The paper is written by the author"; "The paper lists the author"; "The paper is put forth by the author"; "The paper is made public by the author"}
("paper", "to", "term")	{"The paper has the term"; "The paper includes the term"; "The paper contains the term"; "The paper encompasses the term"; "The paper lists the term"}
("paper", "to", "conference")	{"The paper is published in the conference"; "The paper is accepted by the conference"; "The paper is included in the proceedings of the conference"; "The paper is presented at the conference"; "The paper appears in the conference proceedings"; "The paper is part of the conference's official record"; "The paper makes its debut at the conference"; "The paper is documented in the conference's scholarly collection"}
("term", "to", "paper")	{"The term is related to the paper"; "The term is included in the paper"; "The term is in the paper"; "The term is featured in the paper"; "The term is incorporated into the paper"; "The term is part of the paper's content"; "The term is found within the paper"; "The term appears in the paper"}
("conference", "to", "paper")	{"The conference has the paper"; "The conference includes the paper"; "The conference accepts this paper"; "The proceedings of the conference includes the paper"; "The conference features the publication of the paper"; "The conference includes the paper in its publications"; "The conference serves as the platform for the paper's publication"}

```

IMDB-ICL-1:
Q: Given a heterogeneous graph about internet movie... {Human Question}
A: {Ground Truth Answer&Reasoning}
Q: Given a heterogeneous graph about internet movie... {Human Question}
IMDB-ICL-2:
Q: Given a heterogeneous graph about internet movie... {Human Question}
A: {Ground Truth Answer&Reasoning}
Q: Given a heterogeneous graph about internet movie... {Human Question}
A: {Ground Truth Answer&Reasoning}
Q: Given a heterogeneous graph about internet movie... {Human Question}
ACM-ICL-DBLP:
Q: Given a heterogeneous academic network graph about computer science
from DBLP website ... {Human Question}
A: {Ground Truth Answer&Reasoning}
Q: Given a heterogeneous academic network graph about computer science
collected from ACM website... {Human Question}

```

stage, targeting heterogeneous relation awareness and homogeneous relation awareness, respectively. The construction method for other datasets is the same as that for IMDB.

**A.4.3 Instruction construction template for graph in-context learning.** In Figure 4, we illustrate the construction of instructions for our graph in-context learning tests, which includes "-ICL-1"

**Figure 4: Instruction construction template for graph in-context learning, including "ICL-1", "ICL-2" and "ICL-DBLP".**

**Table 6: Detailed Descriptions of Text-Enriched Heterogeneity Representations for ACM (Continued).**

Node (Edge) Type	Sets of Descriptions
(c) ACM	
"paper"	{"This node represents a paper", "A paper in the area of Database", "A paper in the area of Wireless Communication", "A paper in the area of Data Mining", "A paper published in a conference (one of KDD, SIGMOD, SIGCOMM, MobiCOMM, and VLDB)", "A paper in the area of computer science"}
"subject"	{"This node represents a subject", "The subject is related to Database", "The subject is related to Wireless Communication", "The subject is related to Data Mining", "The subject is related to computer science"}
"author"	{"This node represents an author", "An author in the area of Database", "An author in the area of Data Mining", "An author in the area of AI", "An author in the area of Information Retrieval", "An author in the area of computer science"}
"term"	{"This node represents a key term related to a paper", "The term is included in a paper", "The term is related to Database", "The term is related to Wireless Communication", "The term is related to Data Mining", "The term is related to computer science"}
("author", "to", "paper")	{"The author has the paper", "The author publishes the paper", "The author writes the paper", "The author is the author of the paper", "The author releases the paper", "The author issues the paper", "The author disseminates the paper"}
("paper", "to", "author")	{"The paper has the author", "The paper is published by the author", "The paper is written by the author", "The paper lists the author", "The paper is put forth by the author", "The paper is made public by the author"}
("paper", "to", "term")	{"The paper has the term", "The paper includes the term", "The paper contains the term", "The paper encompasses the term", "The paper lists the term"}
("term", "to", "paper")	{"The term is related to the paper", "The term is included in the paper", "The term is in the paper", "The term is featured in the paper", "The term is incorporated into the paper", "The term is part of the paper's content", "The term is found within the paper", "The term appears in the paper"}
("paper", "to", "subject")	{"The paper pertains to the subject", "The paper is concerned with the subject", "The paper addresses this subject's matter", "The paper contributes to the discourse on the subject", "The paper explores the subject in depth", "The paper examines the subject", "The paper is dedicated to the analysis of the subject", "The paper's content is relevant to the subject", "The paper provides insights into the subject", "The paper discusses the subject comprehensively"}
("subject", "to", "paper")	{"The subject serves as the focus for the paper", "The subject is the central theme of the paper", "The subject forms the basis of the paper's inquiry", "The subject underpins the scholarly work presented in the paper", "The subject informs the paper's research focus", "The subject delineates the scope of the paper's investigation", "The subject is the cornerstone of the paper's theoretical foundation"}
("paper", "cite", "paper")	{"The paper cites the paper", "The paper includes references to a previous paper", "The manuscript references earlier research in its bibliography", "The study attributes findings to an antecedent scholarly paper", "The paper assimilates insights from a previously published study."}
("paper", "ref", "paper")	{"The paper is cited by the paper", "The paper receives a citation from a subsequent publication", "The article is referenced within the bibliography of another scholarly work", "The manuscript is acknowledged by another study in its references", "The document is listed in the citations of another academic article"}

with one example, "-ICL-2" with two examples, and "-ICL-DBLP" where examples from ACM are concatenated with DBLP examples. We represent the examples and the final question using a Q: ...A: ...Q: ... sequence.

## A.5 Supplementary Experimental Results

**A.5.1 Implementation Details.** In the implementation of our HiGPT, we employ heterogeneous graph data with corresponding textual contents from IMDB and DBLP to conduct text-graph contrastive alignment and obtain a heterogeneous graph tokenizer. In the first phase, we utilize a heterogeneous graph corpus consisting of IMDB and DBLP for instruction tuning. We set the batch size to 1 per GPU and train for 1 epoch with the learning rate  $2e^{-5}$ , the warmup ratio  $3e^{-2}$  and the weight decay  $1e^{-4}$ . The projector obtained from the first phase training is used as the initial state for the second phase, where we set the epochs to 15 and further perform instruction tuning on downstream tasks. The base model used in both stages is vicuna-7B-v1.5, with the maximum context length set to 2048. And For the evaluation of most baselines, we utilize their

publicly available code. We employ a grid-search strategy based on default hyperparameter settings to ensure a fair evaluation. For further details, please refer to our released source code.

**A.5.2 Model Case Study.** In this subsection, we explore the behavior of our HiGPT under different prompting techniques. Specifically, we utilize various prompting techniques to prompt the 10-shot IMDB model, obtaining six different responses, and the prediction cases for different categories of HiGPT are shown in Tables 10, 11, and 12, respectively. The parts showing the final answers are highlighted in pink. We make the following observations: **Obs.1** Our HiGPT, after instruction tuning with the MoT graph instruction augmentation strategy, can dynamically respond accurately to different prompts. **Obs.2** The CoT prompt in Table 10, which is unformatted, also shows a certain format (highlighted in yellow), which is attributed to the fact that mixing a variety of instructions can also benefit different prompting techniques. **Obs.3** As highlighted in green in multiple cases, our HiGPT, after our designed two-stage graph instruction tuning, is consciously considering issues from a graph perspective, further proving that our model is not only

**Table 7: Detailed Prompt of Mixture-of-Thought (MoT) Graph Instruction Augmentation.**

Prompt Engineering	Prompting Template	Augmented Instruction Template
<b>CoT without Format Constraint</b>	I have a question as below: {Human Question} ; and the answer is {Ground Truth}, imagine that you have made the correct choice and proceed with step-by-step reasoning. Example: Data mining. Based on ...	{Human Question} → {GPT’s Answer&Reasoning}
<b>CoT with Format Constraint</b>	I have a question as below: {Human Question} ; and the answer is {Ground Truth}, imagine that you have made the correct choice and proceed with step-by-step reasoning. Using the following format: Answer: [Answer] Reason: ...	{Human Question} → {GPT’s Answer&Reasoning}
<b>ToT with Multiple Round</b>	I have a question as below: {Human Question} ; and the answer is {Ground Truth}, imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realises they’re wrong at any point then they leave. And finally they make the correct choice. Using the following format: Expert 1:... Expert 2:... Expert 3:... Expert 1:... Expert 2:... Expert 3:... Final Answer:...	{Human Question} → {GPT’s Answer&Reasoning}
<b>ToT with Single Round</b>	I have a question as below: {Human Question} ; and the answer is {Ground Truth}, imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realises they’re wrong at any point then they leave. And finally they make the correct choice. Using the following format: Expert 1:... Expert 2:... Expert 3:... Final Answer:...	{Human Question} → {GPT’s Answer&Reasoning}
<b>PanelGPT</b>	I have a question as below: {Human Question} ; and the answer is {Ground Truth}, imagine that 3 experts are discussing the question with a panel discussion, trying to solve it step by step to make sure the result is correct and avoid penalty. And finally they make the correct choice.	{Human Question} → {GPT’s Answer&Reasoning}
<b>GKP-1</b>	I have a question as below: {Human Question} ; and the answer is {Ground Truth}, please generate some knowledge that can assist in formulating an answer, including, but not limited to: distinctions between the four categories. Imagine that you have arrived at the correct answer based on the provided information and knowledge, and present a step-by-step reasoning. Using the following format: Knowledge: ... Answer: ... Reason: ...	{Human Question} +{GPT’s Knowledge} → {GPT’s Answer&Reasoning}
<b>GKP-2</b>	Please generate some knowledge that can assist in formulating an answer, including, but not limited to: explanations of some technical terms present in the given information. Imagine that you have arrived at the correct answer based on the provided information and knowledge, and present a step-by-step reasoning. Using the following format: Knowledge: ... Answer: ... Reason: ...	{Human Question} +{GPT’s Knowledge} → {GPT’s Answer&Reasoning}

solving downstream problems from a textual perspective but has also developed a certain level of graph-awareness.

**A.5.3 Comprehensive Results of Overall Performance Comparison.** Table 13 showcases the results of our HiGPT in both few-shot and zero-shot settings, covering scenarios with 1, 3, 5, 10, 20, 40, and 60 shots. The results clearly indicate that our model consistently outperforms state-of-the-art models in all cases.

**A.5.4 Comprehensive Results of Graph In-Context Learning.**

The performance of our model on all metrics, across different shot scenarios and datasets, under the graph in-context learning is illustrated in Figure 5. The results clearly demonstrate that our graph in-context learning approach significantly improves the model’s performance without any modifications to the model parameters.

**Table 8: Instruction template for IMDB.**

---

**Node Classification**

---

**Input:** Given a heterogeneous graph about internet movie, there are three types of nodes, namely: movie, actor, director. The relationships (meta paths) between different nodes include: [movie is directed by director], [movie has actor]. By performing random sampling of 2-hop 10 neighbors centered on the target movie node, a heterogeneous subgraph is obtained. In the subgraph, "movie" nodes: <graph>, where the 0-th node is the central node that represents a movie with the following information: Name: {movie name} Director's name: {director name} Actors' names: {actor name} Plot keywords: {plot keywords} "actor" nodes: <graph>; "director" nodes: <graph>. Question: Which of the following classes does this movie belong to: action, comedy, drama? {Prompting Suffix}

**Output:** {Answer (& Reasoning)}.

---

**Prompting Suffix of Different Prompting Techniques**

---

**Standard with the direct answer:** Give likely categories directly.

**CoT without Format Constraint:** Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning.

**CoT with Format Constraint:** Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning. Using the following format: Answer: [The answer] Reason: ...

**ToT:** Imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realises they're wrong at any point then they leave.

**PanelGPT:** 3 experts are discussing the question with a panel discussion, trying to solve it step by step, and make sure the result is correct and avoid penalty.

---

**Table 9: Instruction template for IMDB (Continued).**

---

**Instruction for Heterogeneous Relation Awareness**

---

**Input:** Given a heterogeneous graph about internet movie, there are three types of nodes, namely: movie, actor, director. The relationships (meta paths) between different nodes include: [movie is directed by director], [movie has actor]. By performing random sampling of 2-hop 10 neighbors centered on the target movie node, a heterogeneous subgraph is obtained. In the subgraph, there are several sequences of heterogeneous graph nodes of different types as follows: 1. <graph> 2. <graph> 3. <graph> Question: please sequentially provide the categories for the node sequences. Example: 1. actor 2. movie 3. director.

**Output:** {The correct sequence of node types}.

---

**Instruction for Homogeneous Relation Awareness**

---

Given a heterogeneous graph about internet movie, there are three types of nodes, namely: movie, actor, director. The relationships (meta paths) between different nodes include: [movie is directed by director], [movie has actor]. By performing random sampling of 2-hop 10 neighbors centered on the target movie node, a heterogeneous subgraph is obtained. In the subgraph, there are "movie" nodes: <graph>. Here is a list of movies' names, their directors' names, their actors' names and their plot keywords: {movie lists} Question: please reorder the list of movies according to the order of graph tokens (*i.e.*, complete the matching of graph tokens and movies)

---

**Output:** {The correct sequence of movies' names}.

---



**Table 10: Visualization of our HiGPT's response with different prompting engineering techniques on IMDB for action genre.**

---

**Question:** Given a heterogeneous graph about internet movie, there are three types of nodes, namely: movie, actor, director. The relationships (meta paths) between different nodes include: [movie is directed by director], [movie has actor]. By performing random sampling of 2-hop 10 neighbors centered on the target movie node, a heterogeneous subgraph is obtained. In the subgraph, "movie" nodes: <graph>, where the 0-th node is the central node that represents a movie with the following information: Name: Live Free or Die Hard; Director's name: Len Wiseman; Actors' names: Bruce Willis|Jonathan Sadowski|Cyril Raffaelli; Plot keywords: fbi|hacker|independence day|police|terrorist; "actor" nodes: <graph>; "director" nodes: <graph>. Question: Which of the following classes does this movie belong to: action, comedy, drama?

---

**Ground-Truth Category:** Action

---

+ Give likely categories directly. **HiGPT (Direct Response):** Action.

---

+ Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning.

**HiGPT (CoT Response without Format Constraint):** **Answer: Action Reasoning:** The movie "Live Free or Die Hard" has several plot keywords that suggest an action genre, such as "fbi", "hacker", "independence day", "police", and "terrorist". These keywords imply that the movie is likely to have a lot of action scenes, chases, and possibly some thrilling moments. Additionally, the director Len Wiseman has directed other action movies in the past, such as "Underworld" and "Total Recall". This further supports the idea that "Live Free or Die Hard" belongs to the action genre. Finally, the actors in the movie also contribute to the action genre. Bruce Willis, who plays the lead role, is known for his roles in action movies such as "Die Hard", "Pulp Fiction", and "The Fifth Element". Jonathan Sadowski and Cyril Raffaelli, who also appear in the movie, have acted in other action movies and TV shows. Overall, based on the plot keywords, the director's past work, and the actors' roles, it is reasonable to conclude that "Live Free or Die Hard" belongs to the action genre.

---

+ Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning. Using the following format: Answer: [The answer] Reason: ...

**HiGPT (CoT Response with Format Constraint):** **Answer: Action Reason:** Based on the plot keywords provided, the movie seems to involve elements of action, such as the FBI, hackers, police, and terrorists. These keywords suggest that the movie may have a high level of action and suspense, which is a common characteristic of action movies. Additionally, the presence of Bruce Willis, a well-known action star, further supports the classification of this movie as an action film.

---

+ Imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realizes they're wrong at any point then they leave.

**HiGPT (ToT Response):** Expert 1: First, I will analyze the plot keywords of the movie. The keywords "fbi", "hacker", "independence day", "police", and "terrorist" suggest that the movie might belong to the action genre. However, the presence of the word "independence day" could also indicate that the movie might be a combination of action and drama. Expert 2: Next, I will look at the director's name. Len Wiseman is known for directing the "Underworld" film series, which are primarily action movies. Therefore, it is likely that "Live Free or Die Hard" belongs to the **action genre**. Expert 3: Finally, I will examine the actors' names. Bruce Willis is an action star, and Jonathan Sadowski and Cyril Raffaelli are relatively unknown actors. Therefore, it is more likely that the movie belongs to the action genre. **Based on the analysis of the plot keywords, the director's name, and the actors' names, it is likely that "Live Free or Die Hard" belongs to the action genre.**

---

+ 3 experts are discussing the question with a panel discussion, trying to solve it step by step, and make sure the result is correct and avoid penalty

**HiGPT (Panel Discussion Response):** Expert 1: Let's start by analyzing the plot keywords. The keywords "fbi", "hacker", "independence day", "police", and "terrorist" suggest that this movie might belong to the action genre. Expert 2: That's a good point. However, we should also consider the director's previous works. Len Wiseman is known for directing the "Underworld" series, which are primarily action movies. Therefore, it's likely that this movie also belongs to the action genre. Expert 3: I agree with both of you. However, we should also consider the actors' names. Bruce Willis is known for his roles in action movies, and Jonathan Sadowski and Cyril Raffaelli are relatively unknown actors. **Therefore, it's possible that this movie might also belong to the drama genre.** Expert 1: That's a good point. We should consider all possible genres. **Let's perform a random sampling of 2-hop 10 neighbors centered on the target movie node to obtain a heterogeneous subgraph.** Expert 2: Great idea. Let's see what we can find. Expert 3: Agreed. **After performing the random sampling, the experts found that the movie "Live Free or Die Hard" belongs to the action genre.**

---

**Table 11: Visualization of our HiGPT's response with different prompting engineering techniques on IMDB for comedy genre.**

---

**Question:** Given a heterogeneous graph about internet movie, there are three types of nodes, namely: movie, actor, director. The relationships (meta paths) between different nodes include: [movie is directed by director], [movie has actor]. By performing random sampling of 2-hop 10 neighbors centered on the target movie node, a heterogeneous subgraph is obtained. In the subgraph, "movie" nodes: <graph>, where the 0-th node is the central node that represents a movie with the following information: Name: Caddyshack; Director's name: Harold Ramis; Actors' names: Bill Murray|Rodney Dangerfield|Brian Doyle-Murray; Plot keywords: caddy|country club|golf course|gopher|nudity; "actor" nodes: <graph>; "director" nodes: <graph>. Question: Which of the following classes does this movie belong to: action, comedy, drama?

---

**Ground-Truth Category:** Comedy

---

+ Give likely categories directly. **HiGPT (Direct Response):** Comedy.

---

+ Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning.

**HiGPT (CoT Response without Format Constraint):** Based on the information provided in the subgraph, the movie "Caddyshack" belongs to the comedy genre. The plot keywords "caddy|country club|golf course|gopher|nudity" suggest that the movie is a comedy that involves a golf course, caddy, and potentially some nudity. Additionally, the director Harold Ramis is known for his work in comedy films, such as "Groundhog Day" and "Ghostbusters." To further support the classification of "Caddyshack" as a comedy, we can compare it to other movies in the same subgraph that are also directed by Harold Ramis. For example, "Groundhog Day" is also a comedy, and it shares some of the same actors, such as Bill Murray, with "Caddyshack." Therefore, it is reasonable to assume that "Caddyshack" also belongs to the comedy genre. In summary, the movie "Caddyshack" belongs to the comedy genre based on its plot keywords, director Harold Ramis' previous work in comedy films, and its similarity to other comedies in the same subgraph.

---

+ Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning. Using the following format: Answer: [The answer] Reason: ...

**HiGPT (CoT Response with Format Constraint):** Answer: Comedy Reason: The movie "Caddyshack" is a comedy film, as indicated by its plot keywords such as "caddy", "country club", "golf course", and "gopher", which suggest a light-hearted and humorous tone. Additionally, the director Harold Ramis has a history of directing comedies, further supporting the classification of this movie as a comedy.

---

+ Imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realizes they're wrong at any point then they leave.

**HiGPT (ToT Response):** Based on the plot keywords, "caddy," "country club," "golf course," and "nudity," it seems like the movie "Caddyshack" belongs to the comedy genre. Expert 1: The movie has keywords like "caddy," "country club," and "golf course," which are commonly associated with comedy movies. Expert 2: The presence of Bill Murray, Rodney Dangerfield, and Brian Doyle-Murray, who are known for their comedic roles, also suggests that the movie is a comedy. Expert 3: The plot keywords "nudity" may be a red herring, as it is not a common characteristic of comedy movies. Therefore, the movie likely belongs to the comedy genre.

---

+ 3 experts are discussing the question with a panel discussion, trying to solve it step by step, and make sure the result is correct and avoid penalty

**HiGPT (Panel Discussion Response):** To solve this problem, we need to analyze the plot keywords and the director's name to determine the genre of the movie. The plot keywords suggest that the movie is likely to be a comedy, as it contains keywords such as "caddy," "country club," "golf course," and "gopher," which are commonly associated with comedic situations in movies. The director's name, Harold Ramis, also supports this conclusion. He is known for directing comedies such as "Groundhog Day" and "Ghostbusters," which were both critical and commercial successes. Therefore, we can conclude that the movie "Caddyshack" belongs to the comedy genre. As for the sorting of the "actor" nodes, we can use the plot keywords to determine the order. The actors' names suggest that the movie has a mix of well-known and lesser-known actors, but the plot keywords suggest that the movie is primarily focused on the actors' interactions with the golf course and gopher. Therefore, we can sort the actors' names based on their prominence in the movie, with Bill Murray and Rodney Dangerfield being more well-known and therefore higher up in the list, and Brian Doyle-Murray being lesser-known but still important to the plot. So the sorted list of actors' names would be: 1. Bill Murray 2. Rodney Dangerfield 3. Brian Doyle-Murray As for the sorting of the "movie" nodes, we can use the plot keywords to determine the order. The plot keywords suggest that the movie is primarily focused on the golf course and gopher, so we can sort the movie nodes based on their prominence in the movie, with "Caddyshack" being the central node and the other two movies being lesser-known and therefore lower down in the list. So the sorted list of movie nodes would be: 1. Caddyshack 2. Happy Gilmore 3. The Greatest Game Ever Played Therefore, the final sorted list of nodes would be: 1. Caddyshack 2. Harold Ramis 3. Bill Murray 4. Rodney Dangerfield 5. Brian Doyle-Murray 6. Happy Gilmore 7. The Greatest Game Ever Played Note that the order of the "actor" nodes and the "movie" nodes may vary depending on the specific criteria used to sort them. However, the overall conclusion that "Caddyshack" belongs to the comedy genre and is the central node in the subgraph remains the same.

---

**Table 12: Visualization of our HiGPT’s response with different prompting engineering techniques on IMDB for drama genre.**

---

**Question:** Given a heterogeneous graph about internet movie, there are three types of nodes, namely: movie, actor, director. The relationships (meta paths) between different nodes include: [movie is directed by director], [movie has actor]. By performing random sampling of 2-hop 10 neighbors centered on the target movie node, a heterogeneous subgraph is obtained. In the subgraph, "movie" nodes: <graph>, where the 0-th node is the central node that represents a movie with the following information: Name: The Life of David Gale; Director’s name: Alan Parker; Actors’ names: Kevin Spacey|Kate Winslet|Matt Craven; Plot keywords: activist|death|death row|murder|texas"actor" nodes: <graph>; "director" nodes: <graph>. Question: Which of the following classes does this movie belong to: action, comedy, drama?

---

**Ground-Truth Category:** Drama

---

+ Give likely categories directly. **HiGPT (Direct Response):** Drama.

---

+ Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning.

**HiGPT (CoT Response without Format Constraint):** Based on the plot keywords provided, the movie "The Life of David Gale" belongs to the drama genre. The plot keywords include "activist," "death," "death row," "murder," and "texas," which suggest a serious and emotional storyline. These keywords are commonly associated with drama films, rather than action or comedy films. **Therefore, the movie "The Life of David Gale" belongs to the drama genre.**

---

+ Please think about the categorization in a step by step manner and avoid making false associations. Then provide your reasoning. Using the following format: Answer: [The answer] Reason: ...

**HiGPT (CoT Response with Format Constraint):** **Answer: Drama Reason:** The movie "The Life of David Gale" is a drama, as it deals with serious and emotional themes such as activism, death row, and murder. The plot keywords also suggest a dramatic tone.

---

+ Imagine three different experts are answering this question. All experts will write down 1 step of their thinking, then share it with the group. Then all experts will go on to the next step, etc. If any expert realizes they’re wrong at any point then they leave.

**HiGPT (ToT Response):** Expert 1: Based on the plot keywords, this movie seems to be a drama. Expert 2: I agree with Expert 1. The plot keywords such as "activist," "death," "death row," and "murder" suggest a serious and emotional theme. Expert 3: I also agree with Expert 1 and 2. The plot keywords and the fact that the movie is directed by Alan Parker, who is known for his dramatic films, further support the classification of this movie as a drama. **Final answer: The movie belongs to the drama class.**

---

+ 3 experts are discussing the question with a panel discussion, trying to solve it step by step, and make sure the result is correct and avoid penalty

**HiGPT (Panel Discussion Response):** 1. Expert 1: Let’s start by analyzing the plot keywords. We have "activist," "death," "death row," "murder," and "texas." Based on these keywords, I would say this movie belongs to the drama genre. 2. Expert 2: I agree with Expert 1. The plot keywords suggest a serious and emotional storyline. However, we also have "actor" nodes in the graph, which could indicate that there are comedic moments or performances. 3. Expert 3: I would like to add that we should also consider the director’s name, Alan Parker. He has directed both dramatic and comedic movies, such as "The Commitments" and "Birdy." Therefore, it’s difficult to determine the genre solely based on the plot keywords and director’s name. 4. Panel Discussion: Based on the analysis so far, **we can conclude that the movie "The Life of David Gale" belongs to the drama genre, but it may also have comedic elements due to the presence of actor nodes in the graph.**

---



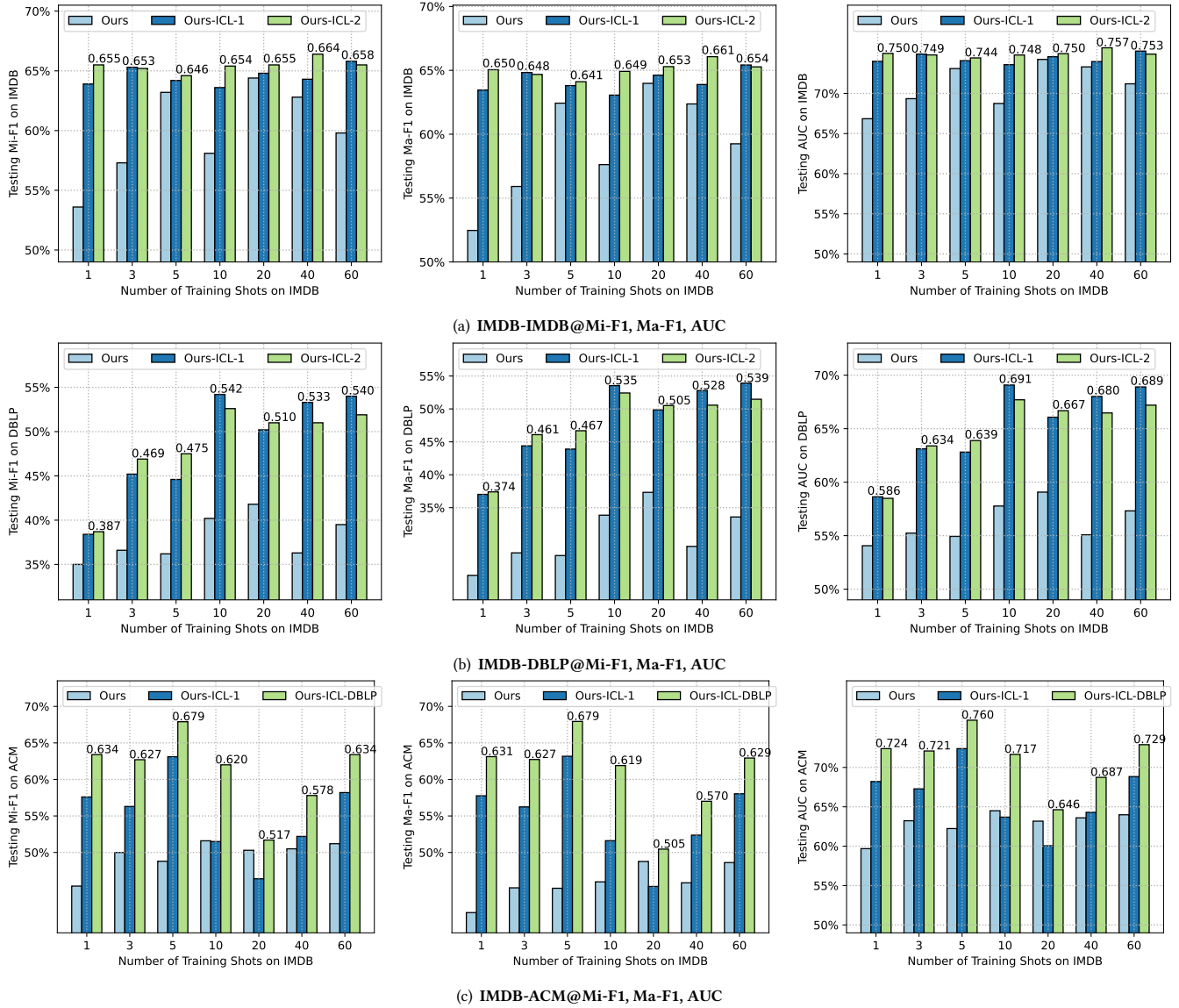


Figure 5: Comprehensive results of graph in-context learning of our HiGPT.