

Natural Language is All a Graph Needs

报告人：宋传承

报告时间：2023.11.10



中国科学院 信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING, CAS



Natural Language is All a Graph Needs

Attention Is **All You Need**

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Lukasz Kaiser*
Google Brain
lukaszkaizer@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Natural Language is **All a Graph Needs**

Ruosong Ye
Rutgers University
ruosong.ye@rutgers.edu

Caiqi Zhang
University of Cambridge
cz391@cam.ac.uk

Runhui Wang
Rutgers University
runhui.wang@rutgers.edu

Shuyuan Xu
Rutgers University
shuyuan.xu@rutgers.edu

Yongfeng Zhang
Rutgers University
yongfeng.zhang@rutgers.edu

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

Xiangguo Sun
Department of Systems Engineering
and Engineering Management, and
Shun Hing Institute of Advanced
Engineering, The Chinese University
of Hong Kong
xiangguosun@cuhk.edu.hk

Hong Cheng
Department of Systems Engineering
and Engineering Management, and
Shun Hing Institute of Advanced
Engineering, The Chinese University
of Hong Kong
hcheng@se.cuhk.edu.hk

Jia Li
Data Science and Analytics Thrust,
The Hong Kong University of Science
and Technology (Guangzhou)
jialeel@ust.hk

Bo Liu
School of Computer Science and
Engineering, Southeast University
Purple Mountain Laboratories
bliu@seu.edu.cn

Jihong Guan
Department of Computer Science and
Technology, Tongji University
jhguan@tongji.edu.cn

**ONE FOR ALL: TOWARDS TRAINING ONE GRAPH
MODEL FOR ALL CLASSIFICATION TASKS**

**Hao Liu^{1*} Jiarui Feng^{1*} Lecheng Kong^{1*} Ningyue Liang¹ Dacheng Tao²
Yixin Chen¹ Muhan Zhang³**
{liuhao, feng.jiarui, jerry.kong, fliang, ychen25}@wustl.edu,
dacheng.tao@gmail.com, muhan@pku.edu.cn

¹Washington University in St. Louis ²JD Explore Academy ³Peking University

Natural Language is All a Graph Needs

动机

- LLM已逐渐取代CNN和RNN统一了CV和NLP领域，然而将图学习问题纳入生成语言建模框架的工作仍然非常有限。随着LLM的重要性不断增长，**探索LLM是否也可以取代GNN作为图的基础模型**变得至关重要。
- **研究趋势一：模型架构的统一**
- Before Transformer：具有不同inductive biases的不同人工智能领域具有不同的基础模型架构。
例如：
 - CNN的设计考虑了图像的空间不变性，从而在计算机视觉任务中具有卓越的性能。
 - RNN和LSTM等记忆增强模型被广泛用于处理自然语言和音频等序列数据。
 - GNN擅长通过消息传递和聚合机制捕获拓扑信息，使其长期以来成为图学习领域的首选。

Natural Language is All a Graph Needs

动机

- After Transformer: 语言、视觉、声音等各领域出现了“大一统”的希望^[1], 例如:

语言:

1. BERT (Bidirectional Encoder Representations from Transformers) : 这是一种基于 Transformer 的预训练语言模型^Q, 在自然语言理解^Q任务中取得了非常优秀的结果。
2. GPT-3^Q (Generative Pre-training Transformer 3) : 这是一种基于 Transformer 的自然语言生成^Q模型, 具有非常强大的生成能力。
3. T5 (Text-To-Text Transfer Transformer) : 这是一种基于 Transformer 的多任务学习模型, 在许多自然语言处理^Q任务中取得了非常优秀的结果。
4. XLNet (eXtreme Transformer) : 这是一种基于 Transformer 的预训练语言模型, 在自然语言理解任务中取得了很好的结果
5. ERNIE: Sun, Y., Chen, Z., Liu, Z., Liu, Q., Chen, E., & Sun, M. (2019). ERNIE: Enhanced Representation through Knowledge Integration. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 4891-4898). 这篇论文描述了 ERNIE 模型, 它是一种基于 Transformer 的预训练语言模型, 可以从知识库中获取额外的信息来增强表示能力。
6. NeZha: Wang, W., Chen, Z., & Sun, Y. (2019). NEZHA: Fast and Accurate Pre-trained Model for Chinese. arXiv preprint arXiv:1909.00204. 这篇论文描述了 NeZha 模型, 它是一种基于 Transformer 的预训练语言模型, 专门用于中文自然语言处理任务。

视觉:

1. ViT (Vision Transformer) : 这是一种基于 Transformer 的视觉模型, 它可以在不使用卷积神经网络^Q的情况下进行图像分类^Q。
2. DETR (End-to-End Object Detection with Transformers) : 这是一种基于 Transformer 的目标检测模型^Q, 它可以同时进行目标检测和分类。
3. DALL-E (Deep Aligned Low-Rank Learning for Image Generation) : 这是一种基于 Transformer 的图像生成模型, 它可以根据文本描述生成相应的图像。
4. DeiT (Distilled Transformer for Image Classification) : 这是一种基于 Transformer 的视觉模型, 它通过蒸馏技术^Q减小了模型的尺寸, 同时保持了较高的准确率。

声音:

1. Speech Transformer: 这是一种基于 Transformer 的语音识别^Q模型, 它可以直接从音频信号中进行识别, 无需人工标注的文本数据。
2. FastSpeech: 这是一种基于 Transformer 的语音合成^Q模型, 它可以将文本转化为人声语音^Q。
3. Transformer-TTS (Transformer-based Text-to-Speech) : 这是一种基于 Transformer 的语音合成模型, 它可以将文本转化为人声语音。
4. Transformer ASR (Automatic Speech Recognition) : 这是一种基于 Transformer 的语音识别模型, 它可以从音频信号中进行识别。

- 此外, 在图机器学习中的非结构化数据、决策序列在强化学习中等领域均有应用。但是不如在传统 CV和NLP领域中应用的广泛。

Attention Is **All You Need**

[1] <https://www.zhihu.com/question/510273885>

Natural Language is All a Graph Needs

动机

- **研究趋势二：多模态数据处理方法的统一**
- T5^[1]建立了文本到文本框架，统一了所有NLP任务作为序列生成问题。
- CLIP^[2]利用图像-文本对来完成多模态任务，并使用自然语言描述的图像。
- Unified Agent^[3]采用自然语言来描述Agent的环境状态，成功解决了许多强化学习问题。
- P5^[4]通过prompt将所有个性化推荐任务重新表述为语言建模任务，进一步推动了这一趋势。
- 上述工作共同表明，利用**自然语言进行多模态信息表示**已成为一个突出且有前途的趋势。
- 然而在图机器学习领域，利用自然语言处理图相关任务的探索仍然有限。



[1] Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research, 21(1):5485–5551, 2020.

[2] Learning transferable visual models from natural language supervision. In International conference on machine learning, pages 8748–8763. PMLR, 2021.

[3] Towards a unified agent with foundation models. In Workshop on Reincarnating Reinforcement Learning at ICLR 2023, 2023.

[4] Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In Proceedings of the 16th ACM Conference on Recommender Systems, pages 299–315, 2022.

Natural Language is All a Graph Needs

动机

- **研究趋势三：图任务处理方法的统一/固化**
- ①**基于GNN的方法**；②**基于Transformer的方法**；③GNN+Transformer；④**基于LLM**
- GNN长期以来在图机器学习领域占据主导地位。利用消息传递和聚合机制，GNN擅长同时学习节点特征、边缘特征和拓扑结构信息。但是GNN容易出现过平滑问题^[1]、无法直接处理来自各种模态（文本或图像等非数字信息）的原始数据。弥补措施是让数据适应GNN的需求，这对于与视觉和语言等其他人工智能领域集成的统一智能系统**缺乏兼容性**。
- Transformer建模于注意力机制，可以通过将图中每个节点和边表示为不同的token来用于图处理，但是处理大规模图时计算量过大、具有基本注意力机制的全局加权平均计算无法有效捕获和学习图的拓扑结构。补救措施通常涉及复杂的注意力机制或数据转换，进一步显著**增加了模型训练的难度**。

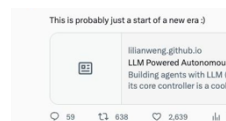
Natural Language is All a Graph Needs

动机

- **研究趋势三：图任务处理方法的统一/固化**
- ①基于GNN的方法；②基于Transformer的方法；③**GNN+Transformer**；④基于LLM
- GNN擅长学习结构信息，而Transformer擅长捕获多模态特征，结合起来可以有效地解决与图相关的任务。但是双方该有的缺陷也跑不了一点，Transformer和GNN之间缺乏解耦，需要训练多个模型，很容易产生巨量的计算开销；模型性能仍然容易受到GNN固有问题的影响（过平滑等）。

- 寻找新的增长点：

- **研究趋势二：多模态数据处理方法的统一**
- T5^[1]建立了文本到文本框架，统一了所有NLP任务作为序列生成问题。
- CLIP^[2]利用图像-文本对来完成多模态任务，并使用自然语言描述的图像。
- Unified Agent^[3]采用自然语言来描述Agent的环境状态，成功解决了许多强化学习问题。
- P5^[4]通过prompt将所有个性化推荐任务重新表述为语言建模任务，进一步推动了这一趋势。
- 上述工作共同表明，利用自然语言进行多模态信息表示已成为一个突出且有前途的趋势。
- 然而在图机器学习领域，利用自然语言处理图相关任务的探索仍然有限。



Prompt Tuning也许会是深度学习时代的 Feature Engineering问题，如何给各大任务设计合理的Prompt将会是很有意思的科学问题

——刘知远

Natural Language is All a Graph Needs

动机

- **研究趋势三：图任务处理方法的统一/固化**
- ①基于GNN的方法；②基于Transformer的方法；③GNN+Transformer；④**基于LLM**
- 利用LLM处理图任务的现有方法可大致分为以下四类：
- **初级LLM应用**：查询LLM来选择最合适的图处理器、嵌入方法等
- **LLM+GNN**：硬结合，LLM充当特征提取器或数据增强模块，以提高下游GNN的性能。GNN仍负责学习图的结构信息，往往会继承GNN的缺点（过平滑等）。
- **LLM+Transformer**：为图transformer设计prompt，复杂Pipeline通过特殊的注意力机制或token表示来传递结构信息，这使得模型无法像GNN那样直接观察和学习结构信息，限制了性能的进一步提高。E.g.上周的DeepGPT，复杂度高且prompt作为可学习参数扩展性低。
- **进阶LLM应用**：在LLM中通过fine-tuning、prompt等方式模仿GNN的机制，利用LLM泛化性强的特点适应图的下游任务。

Natural Language is All a Graph Needs

动机

- 鉴于LLM在许多AI领域占据主导地位，且直观上，自然语言作为最具表现力的媒介之一，擅长描述复杂的结构，因此本文的目标是回答这样一个问题：**LLM是否也可以取代GNN作为图机器学习的基础模型？**

Given that LLMs have been dominant in many AI domains, we aim to answer the question: **Can LLMs also replace GNNs as the foundation model in graph machine learning?** Intuitively, as one of the most expressive medium, natural language is adept at describing complex structures such

- 为了解决基于LLM的图学习器中存在的问题并弥合基于自然语言的图学习的差距，本文提出了 **InstructGLM (Instruction-finetuned Graph Language Model)** 。

Natural Language is All a Graph Needs

Proposed Model

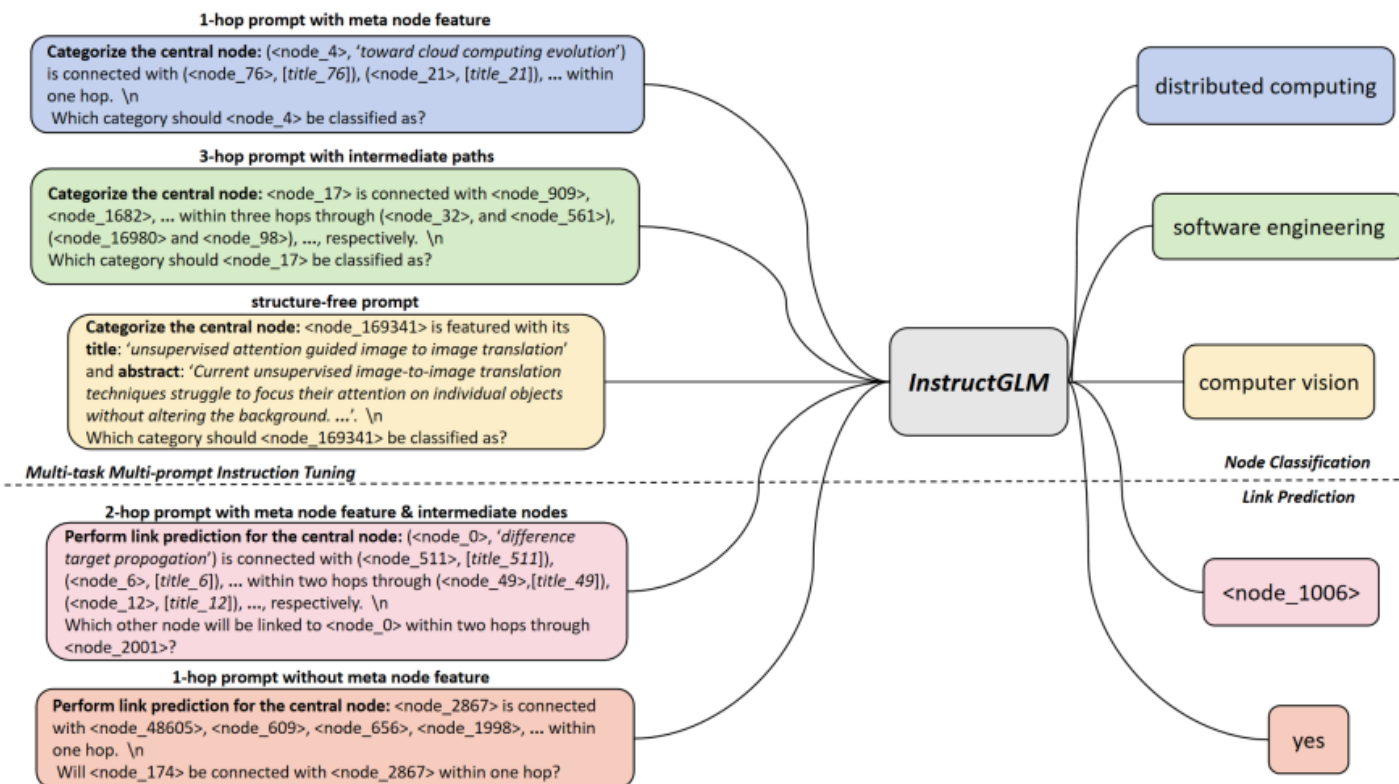


Figure 1: Illustration of the InstructGLM Framework. We fine-tune InstructGLM under a Multi-task Multi-prompt instruction tuning framework, enabling it to solve various graph machine learning tasks with the structure information purely described by natural language.

- 基本任务是**节点分类**任务，同时通过自监督**链路预测**对其进行增强以提高性能。
- 受到GNN中各种消息传递管道的启发，**设计了一系列可扩展的图prompt**，用于在LLM上进行指令微调。
- Specifically，在确定中心节点并进行邻居采样后，我们根据prompt系统地**使用自然语言来描述图的拓扑**，从而清晰直观地向LLM提供图结构，而无需针对图定制复杂的pipeline。

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
2. 指令Prompt的设计
3. Pipeline的设计 3.1面向节点分类的指令微调 3.2辅助的自监督链路预测

- 形式上，图可以表示为 $G = (\mathcal{V}, \mathcal{A}, E, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E})$ ，其中 \mathcal{V} 是节点集， E 是边集， \mathcal{A} 是邻接矩阵， \mathcal{N}_v 是节点特征， \mathcal{E}_e 是边特征。
- 节点特征和边特征可以是多种形式、多种模态的。例如，节点特征可以是引文网络或社交网络中的文本信息、摄影图中的视觉图像、客户系统中的用户配置文件，甚至电影网络中的视频或音频信号，而边特征可以是推荐系统中用户-物品交互图的评论。

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
 2. 指令Prompt的设计
 3. Pipeline的设计
- 3.1 面向节点分类的指令微调 3.2 辅助的自监督链路预测

- 为了全面传达图的结构信息，保证创建的指令prompt对各类图的适应性，我们系统地设计了一套以中心节点为中心的图描述prompt。这些提示可以根据以下三个问题来区分：

- 1. prompt中中心节点的邻居信息的最大跳数是多少？
 - 一跳
 - 多跳：最多到三跳^[1]
- 2. prompt是否包含节点特征或边特征？
 - 包含
 - 不包含
- 3. 对于中心节点具有大 (≥ 2) 跳级邻居的prompt，prompt是否包含有关中间节点或沿相应连接路由的路径的信息？
 - 包含
 - 不包含

[1] 先前的研究表明，利用最多 3 跳连接足以获得出色的性能，而超过 3 跳的信息通常对改进影响较小，甚至可能导致负面影响。

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
2. 指令Prompt的设计
3. Pipeline的设计 3.1面向节点分类的指令微调 3.2辅助的自监督链路预测

- 我们将instruct prompt表示为 $\mathcal{T}(\cdot)$, 使得 $\mathcal{I} = \mathcal{T}(v, \mathcal{A}, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E})$ 是LLM的输入句子, v 是该prompt的中心节点及其相应的用自然语言描述的图结构。例如, 包含最多2跳邻居详细信息的图描述的最简单形式是:

$$\mathcal{T}(v, \mathcal{A}) = \{v\} \text{ is connected with } \{[v_2]_{v_2 \in \mathcal{A}_2^v}\} \text{ within two hops.}$$

- 而其中最详细的形式应包括节点特征、边特征和相应的不同路径:

$$\begin{aligned} \mathcal{T}(v, \mathcal{A}, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E}) = & \{(v, \mathcal{N}_v)\} \text{ is connected with } \{[(v_2, \mathcal{N}_{v_2})]_{v_2 \in \mathcal{A}_2^v}\} \\ & \text{within two hops through } \{[(v_1, \mathcal{N}_{v_1})]_{v_1 \in \mathcal{A}_1^v}\} \text{ and featured} \\ & \text{paths } \{[(\mathcal{E}_{(v, v_1)}, \mathcal{E}_{(v_1, v_2)})]_{v_1 \in \mathcal{A}_1^v, v_2 \in \mathcal{A}_1^{v_1}}\}, \text{ respectively.} \end{aligned}$$

- 其中 \mathcal{A}_k^v 表示节点 v 的 k 跳邻居节点列表。本质上, 上面的prompt包含具有节点和边特征的所有2跳路径, 即以节点 v 为中心的 $(v, \mathcal{N}_v) \xrightarrow{\mathcal{E}_{(v, v_1)}} (v_1, \mathcal{N}_{v_1}) \xrightarrow{\mathcal{E}_{(v_1, v_2)}} (v_2, \mathcal{N}_{v_2})$

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
 2. 指令Prompt的设计
 3. Pipeline的设计
- 3.1 面向节点分类的指令微调 3.2 辅助的自监督链路预测

Prompt ID: 1-2-3-2

Input template:

{{central node}}, {{text feature}} is connected with {{3-hops neighbor list attached text feature}} within three hops through {{corresponding 2-hops intermediate path list attached text feature}}, respectively. Which category should {{central node}}, {{text feature}} be classified as?

Target template: {{category}}

Prompt ID: 1-0-0-0

Input template:

{{central node}} is featured with its {{text feature}}. Which category should {{central node}} be classified as?

Target template: {{category}}

Prompt ID: 2-1-2-4

Input template:

{{central node}} is connected with {{2-hops neighbor list}} within two hops through {{corresponding 1-hop intermediate node list}}, respectively. Which other node will be connected to {{central node}} within two hops through {{specified 1-hop intermediate node}}?

Target template: {{node_id}}

编码规则

- prompt ID用4位数字来编码
- 第1位数字表示任务索引，1表示节点分类，2表示链路预测
- 第2位数字表示是否使用节点特征或边特征的嵌入表示。1表示未使用，2表示已使用。
- 第3位数字表示考虑的结构信息相对应的最大跳次。1/2/3表示包含1/2/3跳邻居的结构信息。
- 第4位数字表示是否考虑高阶连接中的中间节点信息(即路径)。如果数字是偶数，则表示考虑中间节点，如果数字是奇数，则表示不考虑中间节点。
- 特别地，在节点分类任务中设计了一个无图结构的prompt，并将其编号为1-0-0-0。

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
2. 指令Prompt的设计
3. Pipeline的设计 3.1面向节点分类的指令微调 3.2辅助的自监督链路预测

- 形式上, 给定图 $\mathcal{G} = (\mathcal{V}, \mathcal{A}, E, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E})$ 和特定的instruction prompt $I = \mathcal{T}(\cdot)$, \mathbf{x} 和 \mathbf{y} 是LLM的输入和目标句, 那么我们的pipeline可以表示为:

$$P_{\theta}(\mathbf{y}_j | \mathbf{x}, \mathbf{y}_{<j}) = \text{LLM}_{\theta}(\mathbf{x}, \mathbf{y}_{<j}), \mathbf{x} = \text{Concatenate}(\mathcal{P}; \mathcal{I}; \mathcal{Q})$$

$$\mathcal{L}_{\theta} = - \sum_{j=1}^{|\mathbf{y}|} \log P_{\theta}(\mathbf{y}_j | \mathbf{x}, \mathbf{y}_{<j})$$

- 其中 \mathcal{L}_{θ} 表示NLL损失, $\mathcal{I} = \mathcal{T}(v, \mathcal{A}, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E})$ 是以节点 v 为中心的图结构描述, \mathcal{P} 和 \mathcal{Q} 是task-specific指令的prefix和query.
- Specifically, 对于节点分类, 我们设计节点分类的 \mathcal{P} 和 \mathcal{Q} 如下

\mathcal{P} = '将中心节点分为以下类别: [[< 所有类别 >](#)]. 注意节点之间的多跳链接关系'

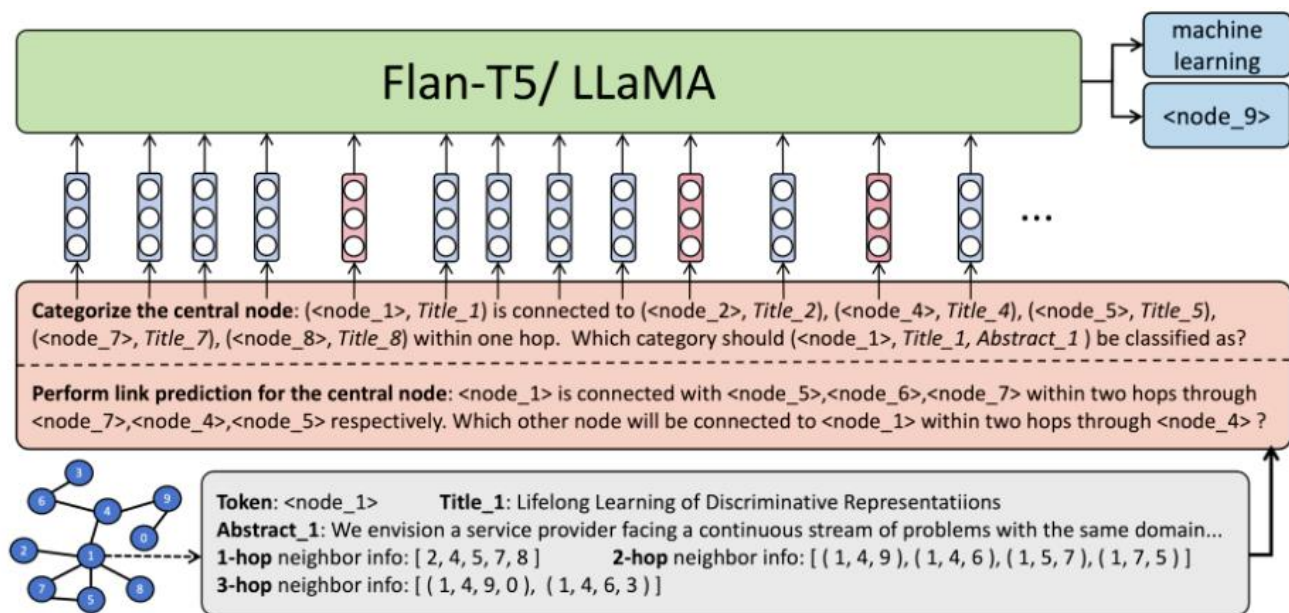
\mathcal{Q} = '{ v }应该属于哪一类?'

node classification, we design \mathcal{P} and \mathcal{Q} for node classification as follows: \mathcal{P} = 'Classify the central node into following categories: [[<All category>](#)]. Pay attention to the multi-hop link relationships between the nodes.' and \mathcal{Q} = 'Which category should { v } be classified as?'. More details of the

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
 2. 指令Prompt的设计
 3. Pipeline的设计
- 3.1 面向节点分类的指令微调 3.2 辅助的自监督链路预测



- 首先，使用图prompt描述每个节点的多跳连接和特征，通过纯自然语言简洁直观地传达图结构以供学习。随后，指令LLM在统一语言建模的Pipeline中生成各种图学习任务响应。
- 为每个节点创建一个新token来扩展LLM的词汇量。



采用LLM的预训练嵌入来表示自然语言token



将图的固有节点特征向量设置为这些新token的嵌入

Figure 2: Illustration of InstructGLM. We use graph prompts to describe each node's multi-hop connectivity and meta features in a scalable mini-batch manner, conveying graph structure concisely and intuitively by pure natural language for learning. Subsequently, we instruct LLM to generate responses for various graph learning tasks in a unified language modeling pipeline. We also expand LLM's vocabulary by creating a new and unique token for every node. More specifically, we set the graph's inherent node feature vectors (e.g. BoW, OGB) as the embedding for these new tokens (depicted as red vectors in the figure) and employ LLM's pre-trained embedding (depicted as blue vectors in the figure) for natural language tokens.

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
2. 指令Prompt的设计
3. Pipeline的设计
- 3.1 面向节点分类的指令微调
- 3.2 辅助的自监督链路预测

- **InstructGLM V.S. GNNs: 机制有相似，表达能力更强，“一力降十会”**
- **混合图卷积**：InstructGLM在训练过程中以中间信息和高跳级邻居为特征，将具有不同跳级信息的prompt信息混合在一起，与Jumping Knowledge将来自不同卷积层的结果结合在一起，和MixHop在不同跳级提取的子图上进行图卷积的做法类似。
- **邻居采样和图正则**：由于LLM的输入长度限制，InstructGLM与GraphSAGE类似，在填写prompt时对中心节点进行邻居采样以形成mini-batch训练，也类似于防止过度平滑的图正则化技术，如DropEdge。
- **图注意力**：由于InstructGLM的输入是直接展示图描述，因此可以被视为GAT的高级加权平均聚合机制，促进了InstructGLM有效掌握不同邻居对中心节点的不同重要性。
- **表达能力和泛化**：InstructGLM中，即使是单个图描述在表达能力上也相当于k层GNN。可以轻松适应图任务的inductive bias，而无需对LLM的架构和流程进行任何更改。

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
 2. 指令Prompt的设计
 3. Pipeline的设计
- 3.1 面向节点分类的指令微调 3.2 辅助的自监督链路预测

- 引入了辅助链路预测任务，可以显著增强模型对图结构的理解，获得了更好的节点表示和节点或图分类的性能^[1]。受它们的启发，也为了消除InstructGLM的指令prompt只能将标记训练节点视为单任务半监督学习中的中心节点的限制，引入自监督链接预测作为辅助任务。
- 给定任意hop level，对于图中的每个节点，我们可以随机选择该hop level的邻居或非邻居作为其候选者。然后对模型进行两种方式的prompt：
 - **Discriminative prompt**: 判别中心节点和候选节点之间的该跳级别是否存在连接
 - **Generative prompt**: 以生成方式直接生成正确的邻居

[1] How to find your friendly neighborhood: Graph attention design with self-supervision. arXiv preprint arXiv:2204.04879, 2022.

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
 2. 指令Prompt的设计
 3. Pipeline的设计
- 3.1 面向节点分类的指令微调 3.2 辅助的自监督链路预测

- 形式上, 给定图 $\mathcal{G} = (\mathcal{V}, \mathcal{A}, E, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E})$ 和特定的instruction prompt $I = \mathcal{T}(\cdot)$, \mathbf{x} 和 \mathbf{y} 是LLM的输入和目标句, 那么我们的pipeline可以表示为:

$$P_{\theta}(\mathbf{y}_j | \mathbf{x}, \mathbf{y}_{<j}) = \text{LLM}_{\theta}(\mathbf{x}, \mathbf{y}_{<j}), \quad \mathbf{x} = \text{Concatenate}(\mathcal{P}; \mathcal{I}; \mathcal{Q})$$

$$\mathcal{L}_{\theta} = - \sum_{j=1}^{|\mathbf{y}|} \log P_{\theta}(\mathbf{y}_j | \mathbf{x}, \mathbf{y}_{<j})$$

- 其中 \mathcal{L}_{θ} 表示NLL损失, $\mathcal{I} = \mathcal{T}(v, \mathcal{A}, \{\mathcal{N}_v\}_{v \in \mathcal{V}}, \{\mathcal{E}_e\}_{e \in E})$ 是以节点 v 为中心的图结构描述, \mathcal{P} 和 \mathcal{Q} 是task-specific指令的prefix和query.
- Specifically, 对于链路预测, 我们设计链路预测的 \mathcal{P} 和 \mathcal{Q} 如下

Natural Language is All a Graph Needs

Proposed Model

1. Preliminary
 2. 指令Prompt的设计
 3. Pipeline的设计
- 3.1 面向节点分类的指令微调 3.2 辅助的自监督链路预测

- Specifically, 对于链路预测, 我们设计链路预测的P和Q如下

$P =$ '对于中心节点进行链路预测。注意节点之间的多跳链接关系。'

$Q_{generative} =$ '{ h } - hop内哪个节点将链接到{ v }?'

$Q_{discriminative} =$ '{ \tilde{v} }会在{ h } - hop内链接到{ v }吗?'

其中 v 是中心节点, \tilde{v} 是候选节点, h 是指定的跳级。

task-specific prefix and two different query templates for it. Specifically, we design \mathcal{P} and \mathcal{Q} for link prediction as follows: $\mathcal{P} =$ 'Perform link prediction for the central node. Pay attention to the multi-hop link relationships between the nodes.', $Q_{generative} =$ 'Which other node will be connected to { v } within { h } hop?' and $Q_{discriminative} =$ 'Will { \tilde{v} } be connected to { v } within { h } hop?', where v is the central node, \tilde{v} is the candidate node and h is the specified hop level.

- 至此, 我们将InstructGLM扩展为一个multi-task、multi-prompt的指令微调框架。

Natural Language is All a Graph Needs

实验

- 实验任务：节点分类、链路预测
- 数据集：ogbn-arxiv (OGB open benchmark: 54%/18%/28%)、Cora (60%/20%/20%)、PubMed (60%/20%/20%)，其中每个节点代表特定主题的学术论文，其标题和摘要包含在原始文本格式中，并且如果两篇论文之间存在引用，那么对应的两个节点之间就会有一条边。
- 采用数据集提供的默认节点特征嵌入，通过添加节点方式新构建的token来扩展LLM的词汇量。

Table 1: Dataset Statistics

Dataset	#Node	#Edge	#Class	Default Feature	#Features
ogbn-arxiv	169,343	1,166,243	40	Skip-gram / GIANT	128 / 768
Cora	2,708	5,429	7	Bag of Words	1433
PubMed	19,717	44,338	3	TF-IDF	500

- 为所有实验采用多prompt指令微调框架，并将test accuracy作为我们的指标。

Natural Language is All a Graph Needs

实验

- 实验结果达到了新的SOTA，在三个数据集上都超过了所有单一模型图学习器（GNNs & Transformers），这表明大语言模型作为图学习的基础模型是大有可为的趋势。

Table 2: Results on ogbn-arxiv

Method		OGB	GIANT
MLP		55.50 ± 0.23	73.06 ± 0.11
GAMLP	(Zhang et al. [78])	56.53 ± 0.16	73.35 ± 0.08
GraphSAGE	(Hamilton et al. [13])	71.19 ± 0.21	74.35 ± 0.14
GCN	(Kipf and Welling [11])	71.74 ± 0.29	73.29 ± 0.01
DeeperGCN	(Li et al. [79])	71.92 ± 0.16	–
ALT-OPT	(Han et al. [80])	72.76 ± 0.00	–
GTAN	(Wu and Wang [81])	72.97 ± 0.17	–
UniMP	(Shi et al. [82])	73.11 ± 0.20	–
LEGNN	(Yu et al. [83])	73.37 ± 0.07	–
GAT	(Veličković et al. [12])	73.66 ± 0.11	74.15 ± 0.05
AGDN	(Sun et al. [84])	73.75 ± 0.21	76.02 ± 0.16
RvGAT	(Li et al. [74])	74.02 ± 0.18	75.90 ± 0.19
DRGAT	(Zhang et al. [85])	74.16 ± 0.07	76.11 ± 0.09
NodeFormer	(Wu et al. [86])	59.90 ± 0.42	–
CoarFormer	(Kuang et al. [87])	71.66 ± 0.24	–
SGFormer	(Wu et al. [88])	72.63 ± 0.13	–
Graphormer	(Ying et al. [23])	72.81 ± 0.23	–
E2EG	(Dinh et al. [89])	73.62 ± 0.14	–
InstructGLM-Flan-T5-base (ours)		73.51 ± 0.16	74.45 ± 0.11
InstructGLM-Flan-T5-large (ours)		74.67 ± 0.08	74.80 ± 0.18
InstructGLM-Llama-v1-7b (ours)		75.70 ± 0.12	76.42 ± 0.09

GNNs

Transformers

- 表2中的实验结果表明，InstructGLM的性能优于所有的基于GNN和基于Transformer的代表性方法。
- 使用OGB特征时，我们的InstructGLM比最佳GNN方法提高了1.54%，比最佳Transformer方法提高了2.08%。
- 使用GIANT特征时也获得了新的SOTA性能。

Natural Language is All a Graph Needs

实验

- 实验结果达到了新的SOTA，在三个数据集上都超过了所有单一模型图学习器（GNNs & Transformers），这表明大语言模型作为图学习的基础模型是大有可为的趋势。

Table 3: Results on Cora and PubMed

	Cora	Acc	PubMed	Acc	
GNNs	MixHop (Abu-El-Haija et al. [35])	75.65 ± 1.31	GAT (Veličković et al. [12])	83.28 ± 0.12	
	GAT (Veličković et al. [12])	76.70 ± 0.42	SGC-v2 (Wu et al. [91])	85.36 ± 0.52	
	Geom-GCN (Pei et al. [92])	85.27 ± 1.48	GraphSAGE (Hamilton et al. [13])	86.85 ± 0.11	
	SGC-v2 (Wu et al. [91])	85.48 ± 1.48	BernNet (He et al. [93])	88.48 ± 0.41	
	GraphSAGE (Hamilton et al. [13])	86.58 ± 0.26	RevGAT (Li et al. [74])	88.50 ± 0.05	
	GCN (Kipf and Welling [11])	87.78 ± 0.96	GCN (Kipf and Welling [11])	88.90 ± 0.32	
	BernNet (He et al. [93])	88.52 ± 0.95	GCNII (Chen et al. [94])	89.80 ± 0.30	
	FAGCN (Bo et al. [95])	88.85 ± 1.36	FAGCN (Bo et al. [95])	89.98 ± 0.54	
	GCNII (Chen et al. [94])	88.93 ± 1.37	MixHop (Abu-El-Haija et al. [35])	90.04 ± 1.41	
	RevGAT (Li et al. [74])	89.11 ± 0.00	Geom-GCN (Pei et al. [92])	90.05 ± 0.14	
	ACM-Snowball-3 (Luan et al. [96])	89.59 ± 1.58	ACM-GCN+ (Luan et al. [96])	90.96 ± 0.62	
	ACM-GCN+ (Luan et al. [96])	89.75 ± 1.16	ACM-Snowball-3 (Luan et al. [96])	91.44 ± 0.59	
	Transformers	Graphormer (Ying et al. [23])	80.41 ± 0.30	GT (Dwivedi and Bresson [33])	88.75 ± 0.16
		GT (Dwivedi and Bresson [33])	86.42 ± 0.82	Graphormer (Ying et al. [23])	88.24 ± 1.50
CoarFormer (Kuang et al. [87])		88.69 ± 0.82	CoarFormer (Kuang et al. [87])	89.75 ± 0.31	
	InstructGLM-Llama-v1-7b (ours)	87.08 ± 0.32	InstructGLM-Llama-v1-7b (ours)	93.84 ± 0.25	
	InstructGLM-Flan-T5-base (ours)	90.77 ± 0.52	InstructGLM-Flan-T5-base (ours)	94.45 ± 0.12	
	InstructGLM-Flan-T5-large (ours)	88.93 ± 1.06	InstructGLM-Flan-T5-large (ours)	94.62 ± 0.13	

- 表3中的实验结果表明，InstructGLM的性能优于所有的基于GNN和基于Transformer的代表性方法。
- 在Cora数据集上比最佳GNN方法提高了1.02%，比最佳Transformer方法提高了2.08%；
- 在PubMed数据集上比最佳GNN方法提高了3.18%，比最佳Transformer方法提高了4.87%。

Natural Language is All a Graph Needs

消融实验

- 实验中有助于InstructGLM在节点分类中取得卓越性能的两个关键操作是：
 - multi-prompt instruction-tuning为LLM提供多跳图结构信息
 - 利用自监督链接预测作为一项辅助任务。
- 下面验证两个关键组件对模型性能的影响：

Table 4: Ablation Study Results

Hop Info	Link Prediction	ogbn-arxiv	Cora	PubMed
		Llama-v1-7b	Flan-T5-base	Flan-T5-base
Multi-hop	w/	75.70%	90.77%	94.45%
Multi-hop	w/o	75.37%	87.27%	94.35%
1-hop	w/o	75.25%	86.90%	94.30%
Structure-Free-Tuning	w/o	74.97%	75.65%	94.22%

- Hop Info: Structure-Free-Tuning表示不考虑图的结构，即直接根据节点的标题和摘要对模型进行微调；1-hop和Multi-hop分别表示我们使用的prompt信息包括来自单跳邻居和多跳邻居的信息。
- 实验结果表明，包含多跳信息和链接预测任务都能提高模型在节点分类任务中的性能。

Natural Language is All a Graph Needs

扩展实验

- 之前实验中的数据分割都保证了相对较高的标记训练节点比例。为了进一步研究InstructGLM的可扩展性和鲁棒性，在PubMed数据集上使用一个标签比率极低（60个、0.3%）的分割进行实验。

Table 5: Results on PubMed with 60 training nodes

Method	Acc
GraphSAGE (Hamilton et al. [13])	76.8 ± 0.9
GAT (Veličković et al. [12])	79.0 ± 1.4
Snowball (Luan et al. [97])	79.2 ± 0.3
GCN (Kipf and Welling [11])	80.4 ± 0.4
SuperGAT (Kim and Oh [75])	81.7 ± 0.5
ALT-OPT (Han et al. [80])	82.5 ± 1.7
GRAND (Feng et al. [98])	82.7 ± 0.6
SAIL (Yu et al. [99])	83.8 ± 0.1
ANS-GT (Zhang et al. [100])	79.6 ± 1.0
NodeFormer (Wu et al. [86])	79.9 ± 1.0
SGFormer (Wu et al. [88])	80.3 ± 0.6
InstructGLM-Llama-v1-7b (ours)	85.1 ± 0.6
InstructGLM-Flan-T5-base (ours)	88.2 ± 0.3
InstructGLM-Flan-T5-large (ours)	89.6 ± 0.4

GNNs

Transformers

- 表5中的实验结果表明，InstructGLM不仅优于所有GNN方法，相对于最佳GNN基线提高了5.8%，同时也超过了最好的基于Transformer的模型9.3%，成功实现了新的SOTA。

Natural Language is All a Graph Needs

总结

- 作者表示本文是**第一个提出纯粹使用自然语言进行图结构表示**，并对**LLM进行指令微调来解决图相关问题的**。消除了设计特定的复杂注意力机制或为图量身定制tokenizer的要求。相反，我们为图机器学习提供了一个**简洁高效的自然语言处理接口**，它对统一的多模态和多任务框架表现出高可扩展性，符合AGI领域的当前趋势。
- 受GNN中各种消息传递机制的启发，为通用图结构表示和图机器学习**设计了一系列基于规则、高度可扩展的指令prompt**，延续了GNN的优点，规避了GNN的缺点。
- 证明了链路预测对于大语言模型更好地理解图结构的重要性，为未来基于LLM的多任务图学习提供了valuable insights。

Given that LLMs have been dominant in many AI domains, we aim to answer the question: **Can LLMs also replace GNNs as the foundation model in graph machine learning?** Intuitively, as one of the most expressive medium, natural language is adept at describing complex structures such
- 对三个广泛使用的数据集进行了广泛的实验，结果实现了top-ranked性能，强调了**利用LLM代替GNN作为图机器学习的基础模型**的趋势。

Thanks!

