

# 基于上下文提示的知识图谱推理基座模型

## A Prompt-Based Knowledge Graph Foundation Model for Universal In-Context Reasoning

Yuanning Cui<sup>1</sup>, Zequn Sun<sup>1</sup>, and Wei Hu<sup>1,2</sup>

<sup>1</sup>State Key Laboratory for Novel Software Technology, Nanjing University, China

<sup>2</sup>National Institute of Healthcare Data Science, Nanjing University, China

<https://github.com/nju-websoft/KG-ICL>

Contact: [yncui.nju@gmail.com](mailto:yncui.nju@gmail.com)

报告人：崔员宁  
2024.11.21





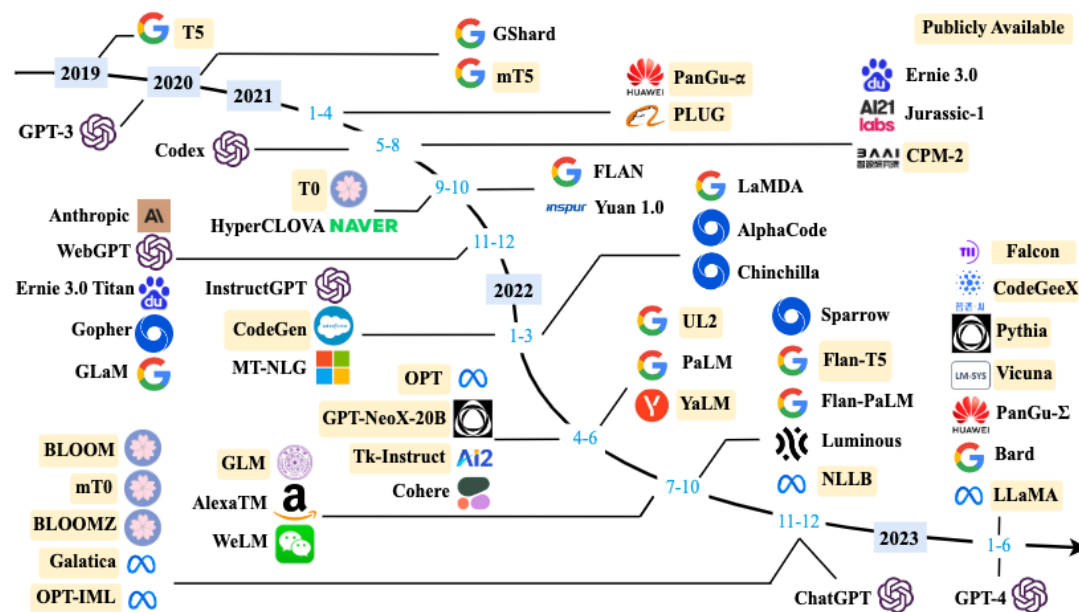
# 研究背景：基座模型

## ■ 基座模型 (foundation model) 在各领域受到广泛关注

- 计算机视觉 (CV) 和自然语言处理 (NLP)
- 深切改变了相关研究领域
- 作为backbone模型成为驱动AI创新的重要引擎

## ■ 基座模型的主要特点

- 是支持多领域应用的预训练模型
- 共享统一的模型架构和知识表征
- 强大的迁移能力和泛化能力





## 研究背景：知识图谱需要基座模型

### ■ 已有很多知识图谱被构建

- 通用知识图谱
- 领域知识图谱
- 企业知识图谱

### ■ 知识图谱持续更新

- 真实世界变化
- 持续的知识采集
- 应用需求的演化

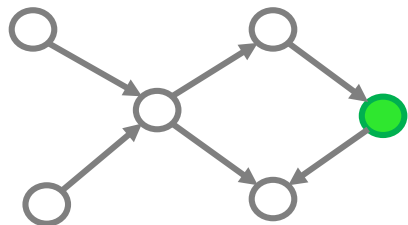




## 研究背景：构建基座模型面临挑战

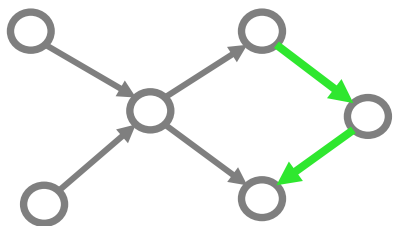
- **问题根源：**现有工作将每个实体、关系视作独立token来学习embeddings，无法编码新tokens

新实体



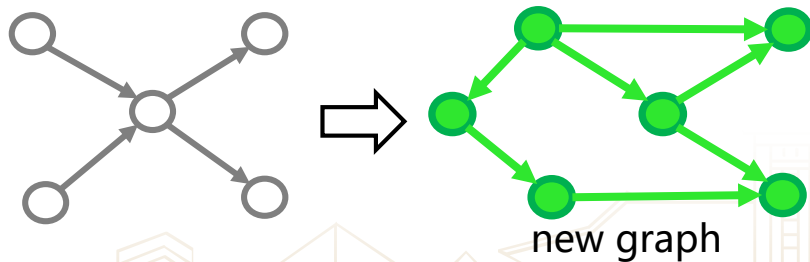
new entity

新关系



new relation

新图谱



new graph

表征学习难泛化的问题在于  
对泛化目标表征的依赖！



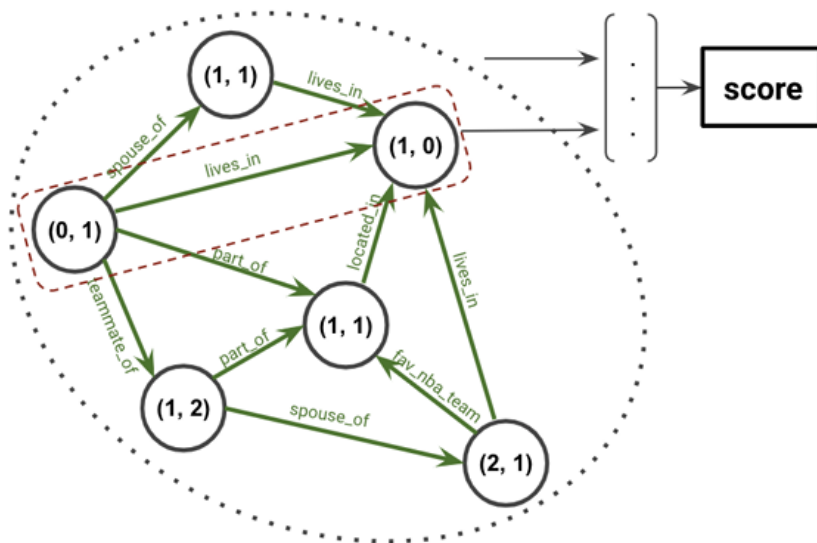


# 相关工作: Inductive Reasoning

- 关系不变，实体增加，如何表征新增实体？

基于封闭子图：

GraIL



基于条件表征：

NBFNet, RED-GNN, A\*Net, MStar

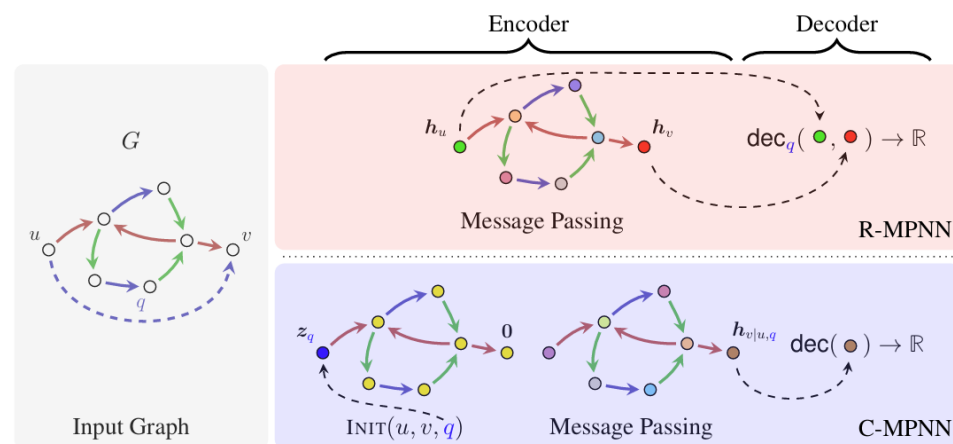


Figure 1: Visualization of R-MPNN and C-MPNN. The dashed arrow is the target query  $q(u, v)$ . Arrow colors indicate distinct relation types, while node colors indicate varying hidden states. R-MPNN considers a unary encoder and relies on a binary decoder, while C-MPNN first initializes binary representation based on the target query  $q(u, v)$ , and then uses a unary decoder.

GraIL: Inductive Relation Prediction by Subgraph Reasoning. ICML 2020

Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction. NeurIPS 2021

Knowledge Graph Reasoning with Relational Digraph. WWW 2022

A\*Net: A Scalable Path-based Reasoning Approach for Knowledge Graphs. NeurIPS 2023

A Theory of Link Prediction via Relational Weisfeiler-Leman on Knowledge Graphs. NeurIPS 2023

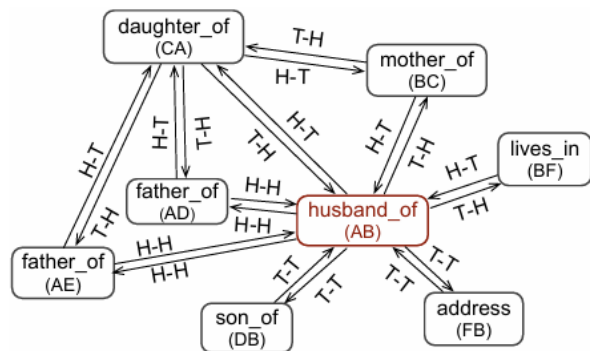
Expanding the Scope: Inductive Knowledge Graph Reasoning with Multi-Starting Progressive Propagation. ISWC 2024



# 相关工作: Fully-Inductive Reasoning

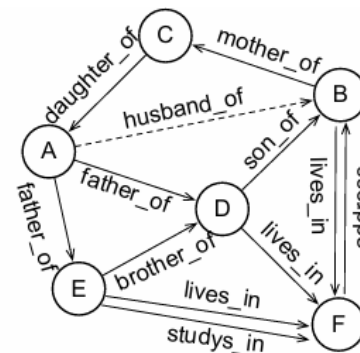
- 关系和实体都在增加, 如何表征新增关系?

关系图→关系表征

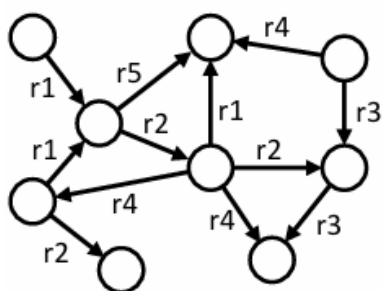


(b) Graph in Relation View

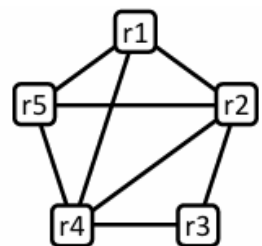
关系表征→实体表征



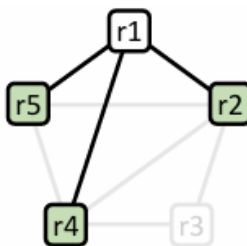
(a) Graph in Entity View



Training Graph



Relation Graph of Training Graph



Relation-level Aggregation



Entity-level Aggregation

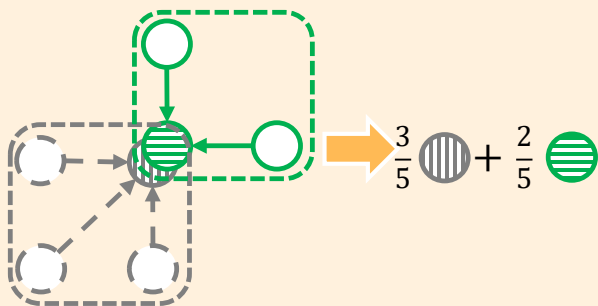
RMPI: Relational Message Passing for Fully Inductive Knowledge Graph Completion. ICDE 2023

INGRAM: Inductive Knowledge Graph Embedding via Relation Graphs. ICML 2023

## ■ 持续表征学习: 持续微调表征模型, 保持已学习知识记忆

### 平衡新旧知识的学习

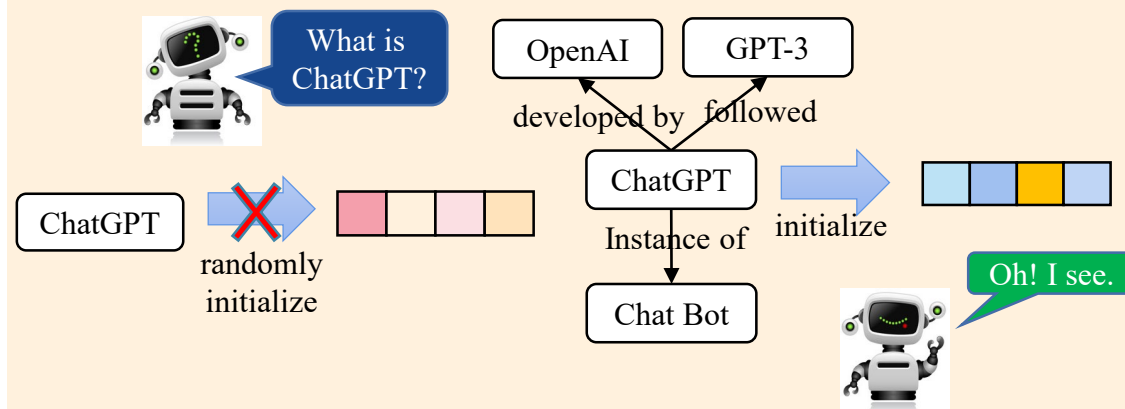
#### 1. 掩码自编码模块



自适应地平衡新旧知识学习

### 基于已有知识加速新知识学习

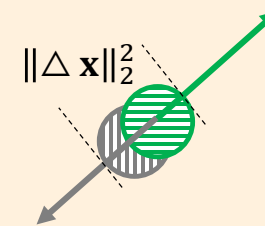
#### 2. 嵌入迁移机制



加速学习, 保持局部语义空间的稳定性

### 避免遗忘

#### 3. 正则化约束更新



避免旧知识灾难性遗忘



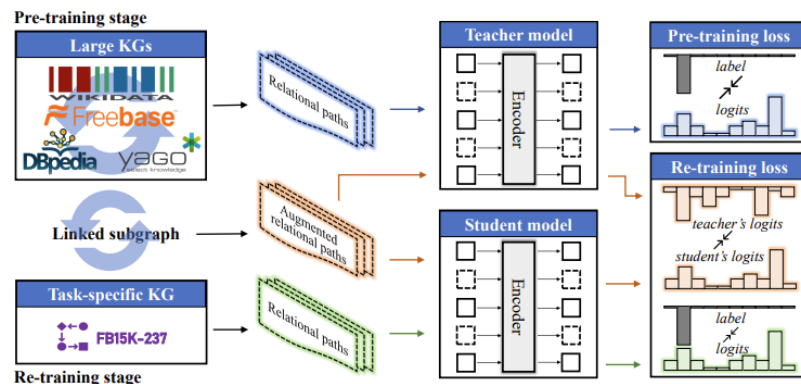




# 相关工作：KG Foundation Model

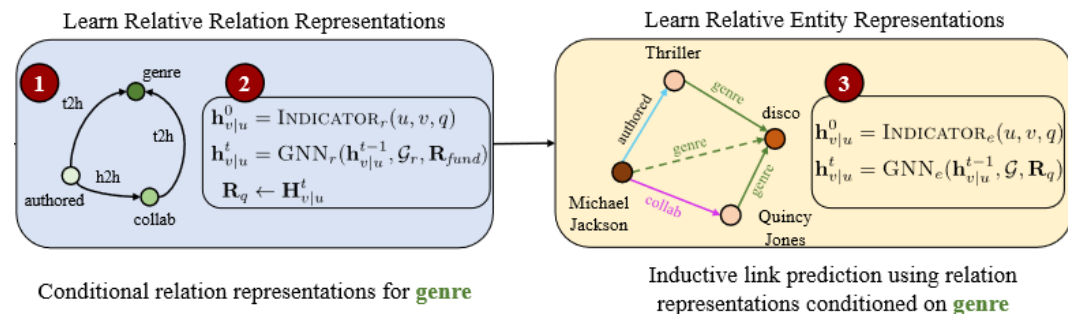
## 预训练→重训练

- 大规模图预训练
- 小规模图重训练



## 预训练→迁移泛化

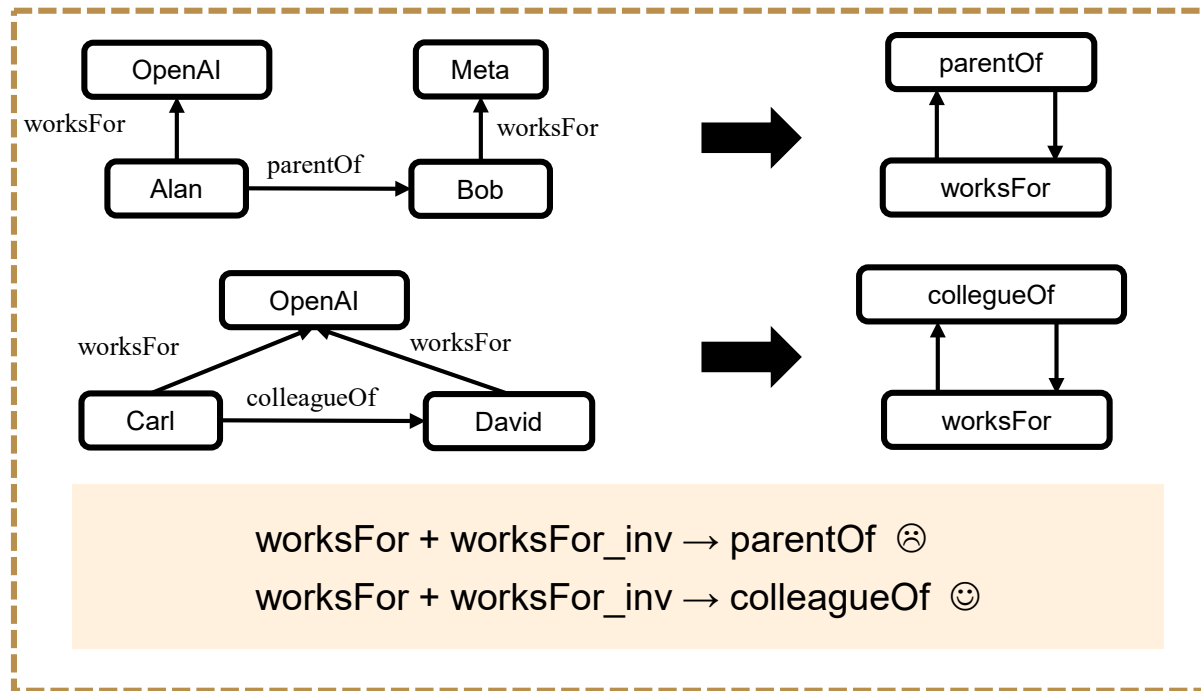
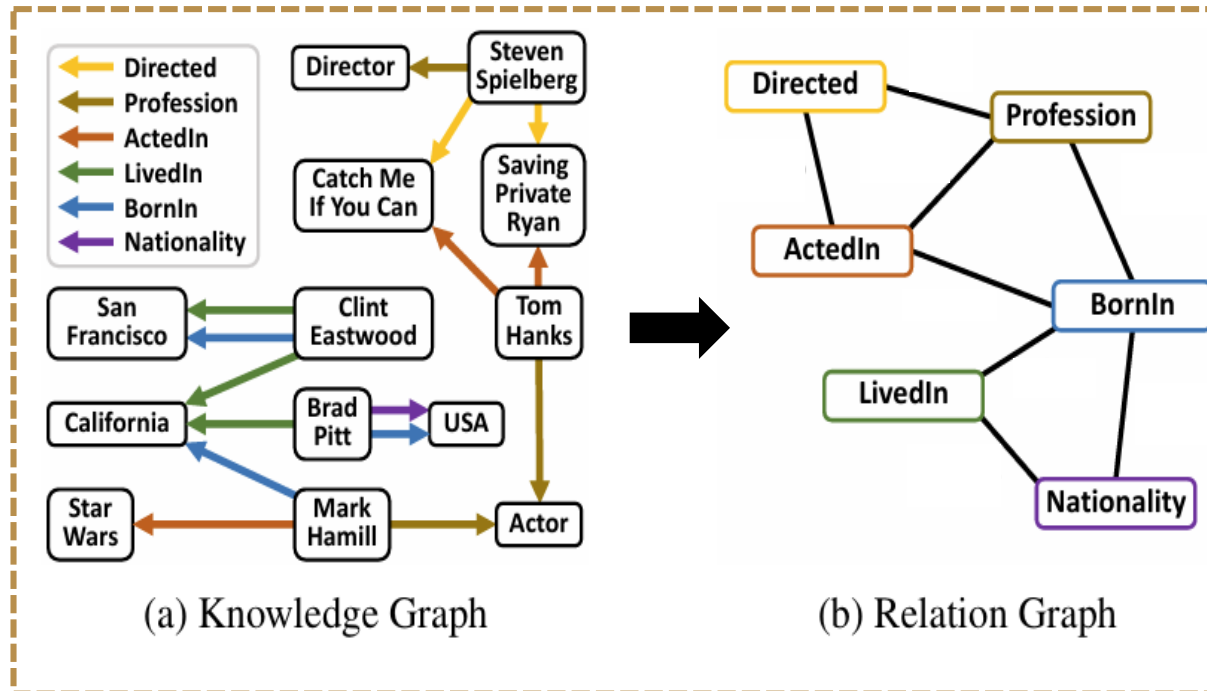
- 关系图→关系表征
- 关系表征→实体表征





## 现有关系图表征存在问题

- 现有关系图存在问题：基于相邻性构建，存在干扰关系路径，难以区分重要关系与干扰关系



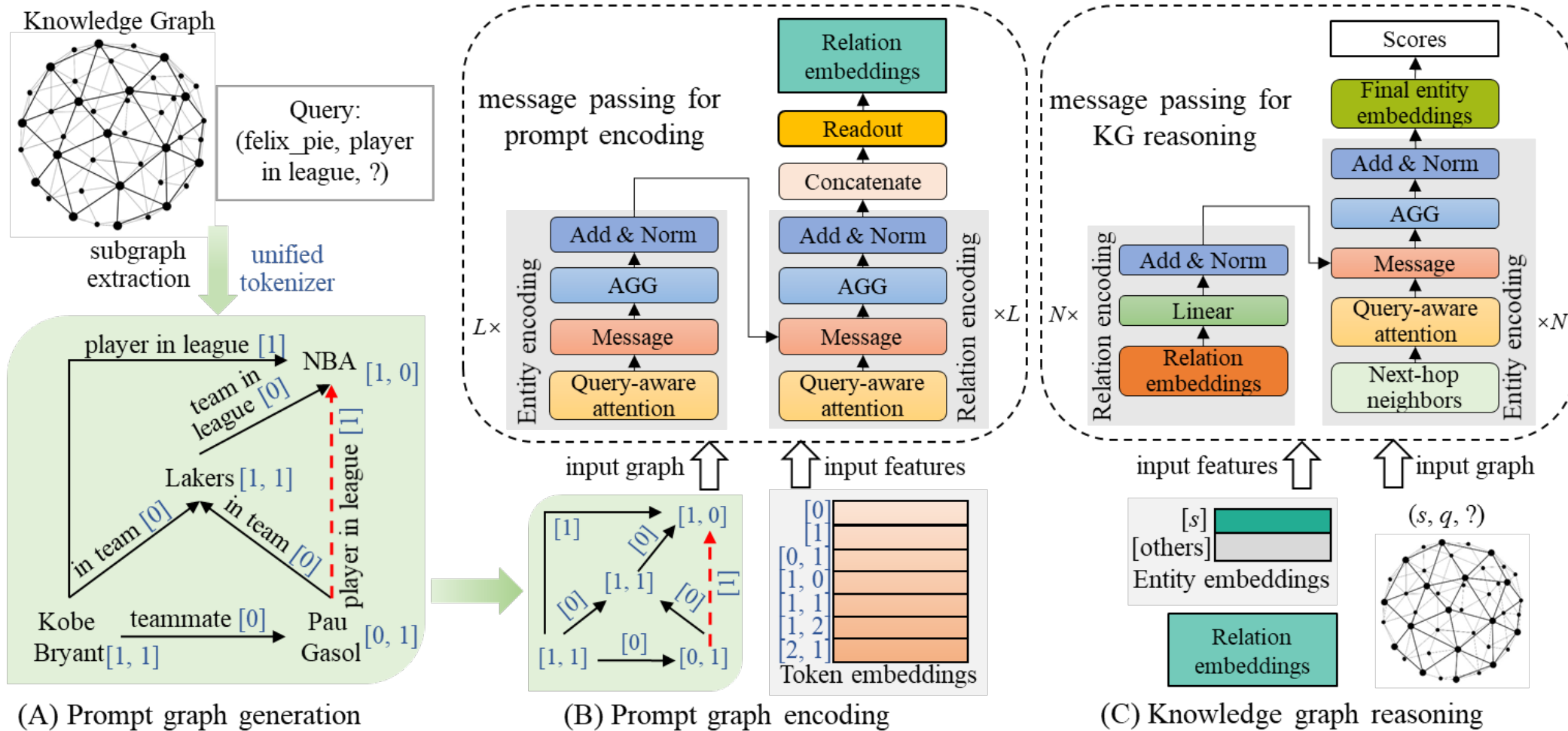
### 如何解决上述问题？

- 保留实例级的实体连接信息
- 以关系为单位建模推理模式





## KG-ICL: 基于上下文提示的通用知识图谱基座模型



## 提示图构建

## KG→样例→提示图

## 提示图编码

## 提示图→关系编码

## 知识图谱推理

## 关系编码→实体分数



## 挑战&工具

- 挑战1：无法在预训练获取未知关系的训练样本，如何实现能迁移与泛化的关系建模？
- 挑战2：不同KGs的提示图中包含不同的实体与关系，如何统一表征这些实体与关系？

### 工具1：提示图

- 示例样本
  - 以包含查询关系的训练样本作为示例
- 提示图作为上下文
  - KG中包含示例样本的子图
- 提示图需要包含对该关系推理的重要信息

### 工具2：统一分词器

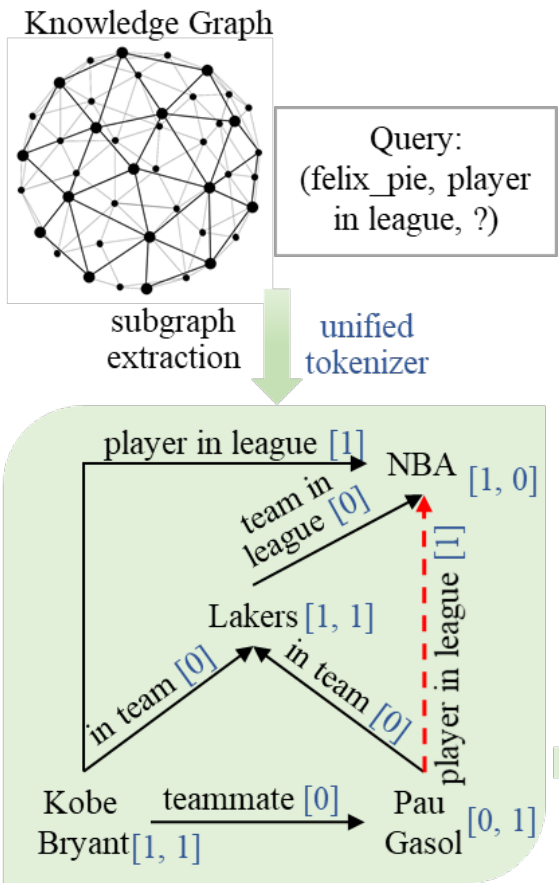
- 实体分词：在提示图中与头尾实体的距离
  - 如 $[1, 1]$ 表示与头尾实体最短路径长度均为1的实体
- 关系分词：是否为查询关系
  - 若为查询关系则为 $[1]$ ，否则为 $[0]$
- 每个预定义分词映射到一个可学习分词向量





## Step 1: 提示图构建

- 对于给定关系，采样包含该关系的示例，并为每个示例构建一个提示图



### (A) Prompt graph generation

- ## ■ 示例采样

- 从KG中采样 $M$ 个包含关系 $r$ 的示例样本

$$\mathcal{S}_q = \{c_i\}_{i=1}^M, \quad c_i \sim \text{Uniform}(\mathcal{N}_q),$$

- ## ■ 子图抽取

- 每个示例样本抽取一个子图作为提示图

- 头尾实体间的 $k$ 跳路径
- 头尾实体的邻居实体

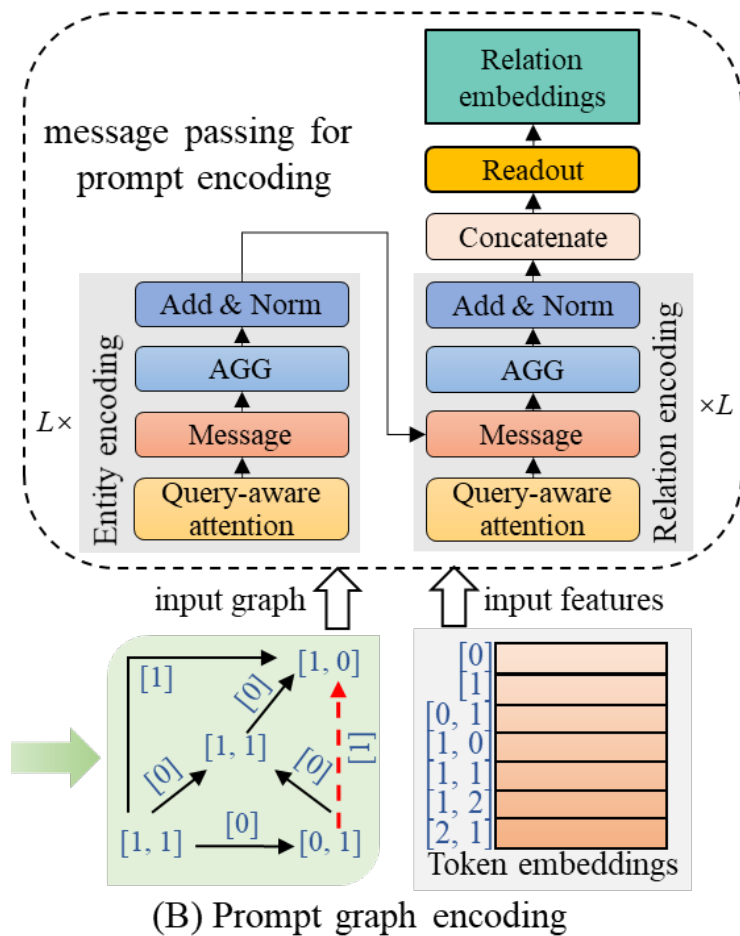
$$\mathcal{E}_{\text{pmt}} = \{x \mid \exists(x, r, u) \in \mathcal{T}\} \cup \{x \mid \exists(x, r, v) \in \mathcal{T}\} \\ \cup \{x \mid \text{dist}(x, u) + \text{dist}(x, v) \leq k\},$$

- ## ■ 分词

- ## 用统一分词器来对提示图中的实体和关系分词

## Step 2: 提示图编码

### ■ 编码提示图，读取关系向量作为提示向量



### ■ 特征初始化

- 每个分词映射到一个预定义的分词向量

### ■ 编码模块

- L层消息传递网络

$$\mathbf{H}_E^{(l+1)} \leftarrow \text{Aggregation}_E \left( \left\{ \text{Message}(\mathbf{H}_E^{(l)}, \mathbf{H}_R^{(l)}, n, q) \right\} \right),$$

$$\forall e \in \mathcal{E}_{\text{pmt}}, \forall n \in \mathcal{N}_e$$

$$\mathbf{H}_R^{(l+1)} \leftarrow \text{Aggregation}_R \left( \left\{ \text{Message}(\mathbf{H}_E^{(l+1)}, \mathbf{H}_R^{(l)}, n, q) \right\} \right),$$

$$\forall r \in \mathcal{R}_{\text{pmt}}, \forall n \in \mathcal{N}_r$$

### ■ 读取

- 提示图最后一层关系编码

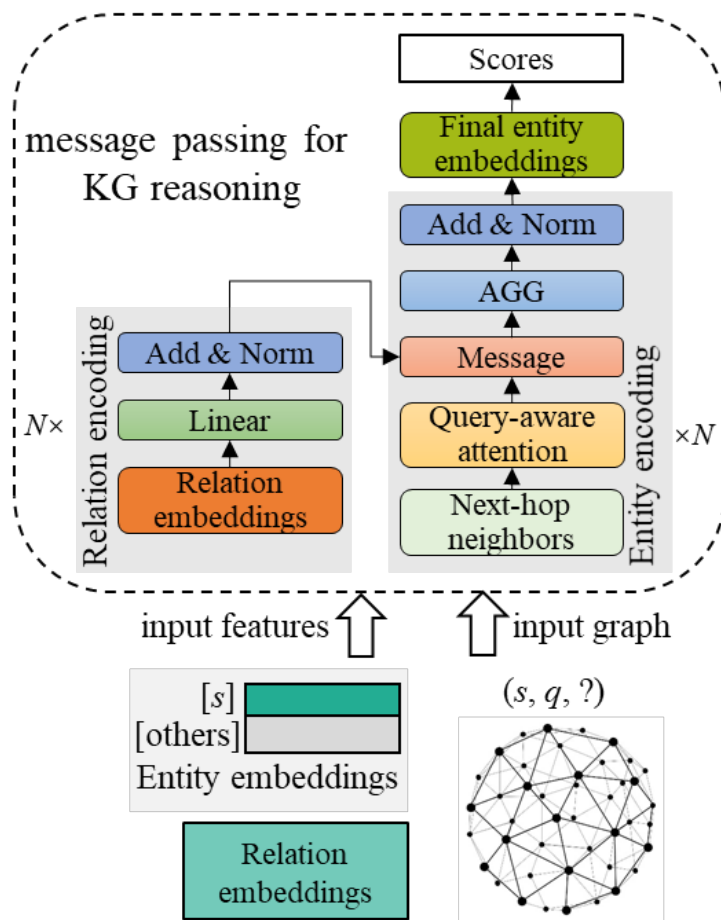
$$\mathbf{H}_{\mathcal{P}} = \mathbf{W}_{\text{Readout}} \left( \mathbf{H}_R^{(1)} \parallel \mathbf{H}_R^{(2)} \parallel \cdots \parallel \mathbf{H}_R^{(L)} \right),$$

- M个提示图编码的平均池化

$$\bar{\mathbf{H}}_{\text{pmt}} = \frac{1}{|\mathcal{S}_q|} \sum_{c \in \mathcal{S}_q} \hat{\mathbf{H}}_{\mathcal{P}_c},$$

## Step 3: 知识图谱编码&推理

### ■ 基于提示关系向量，编码知识图谱中的实体并打分



(C) Knowledge graph reasoning

### ■ 特征初始化

- 实体：头实体  $\leftarrow$  查询关系向量，其他实体  $\leftarrow$  0向量
- 关系：知识图谱关系向量  $\leftarrow$  提示图关系向量

### ■ 知识图谱编码

- 从头实体出发，每层扩展一跳邻居

$$\mathbf{V}_R^{(l+1)} = \text{LN}\left(\mathbf{V}_R^{(l)} + \text{ReLU}(\mathbf{W}_R^{(l)} \mathbf{V}_R^{(l)})\right),$$

$$\mathbf{V}_E^{(l+1)} \leftarrow \text{Aggregation}_E \left( \left\{ \text{Message}(\mathbf{V}_E^{(l)}, \mathbf{V}_R^{(l+1)}, n, q) \right\} \right),$$

$$\forall e \in \mathcal{L}^{(l+1)}, \forall n \in \mathcal{N}_e$$

$$\mathcal{L}^{(0)} = \{s\}, \mathcal{L}^{(l+1)} = \mathcal{L}^{(l)} \cup \{e \mid \exists (x, y, e) \in \mathcal{T} \wedge x \in \mathcal{L}^{(l)}\}$$

### ■ 实体打分

- 读取最后一层的实体编码并打分

$$f(s, q, e) = \mathbf{W}_{\text{score}} \mathbf{e}_{s,q}^{(N)}$$



# 实验设置

## 评测数据集

- 14个归纳式数据集 (inductive datasets)
- 13个全归纳式数据集 (fully-inductive datasets)
- 16个归纳式数据集 (transductive datasets)

## 预训练数据集

- FB V1 (180个关系)
- NELL V1 (14个关系)
- CoDEx-Small (42个关系)

## 测评指标

- 链接预测任务
- 指标: MRR, Hits@10
- 按照数据集类型分组并计算平均指标

## 参数设置

- 示例数: 5
- 路径跳数: 3
- 模型结构: 3层提示图编码层, 6层KG编码层
- 模型参数量: 89k参数
- 预训练时长: 少于6小时







## 主实验结果

### ■ KG-ICL在各类数据集上均具有良好的图级泛化能力

#### ■ 对比模型

- 有监督SOTA模型
- ULTRA pre-train ULTRA finetune
- KG-ICL pre-train KG-ICL finetune

#### ■ 实验结果

- KG-ICL在所有设定上均优于baselines
- KG-ICL的finetune优于pre-train
- 在Inductive数据集上提升更多

Table 1: KG reasoning results in various settings.

Models	Inductive (14 KGs)		Fully-inductive (13 KGs)		Transductive (16 KGs)		Average (43 KGs)	
	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
Supervised SOTA	0.466	0.607	0.210	0.347	0.365	0.511	0.351	0.493
ULTRA pre-train	0.513	0.664	0.352	0.536	0.329	0.479	0.396	0.557
ULTRA finetune	0.528	0.684	0.350	0.542	<u>0.384</u>	<u>0.548</u>	0.421	0.590
KG-ICL pre-train	<u>0.554</u>	<u>0.707</u>	<u>0.439</u>	<u>0.635</u>	0.346	0.493	<u>0.442</u>	<u>0.606</u>
KG-ICL finetune	<b>0.582</b>	<b>0.727</b>	<b>0.449</b>	<b>0.647</b>	<b>0.397</b>	<b>0.554</b>	<b>0.473</b>	<b>0.638</b>





## 详细实验结果

### 43个数据集的详细实验结果

- 大多数数据集上优于baselines
- 在预训练和非预训练数据集上都有提升，且提升比例没有明显差异

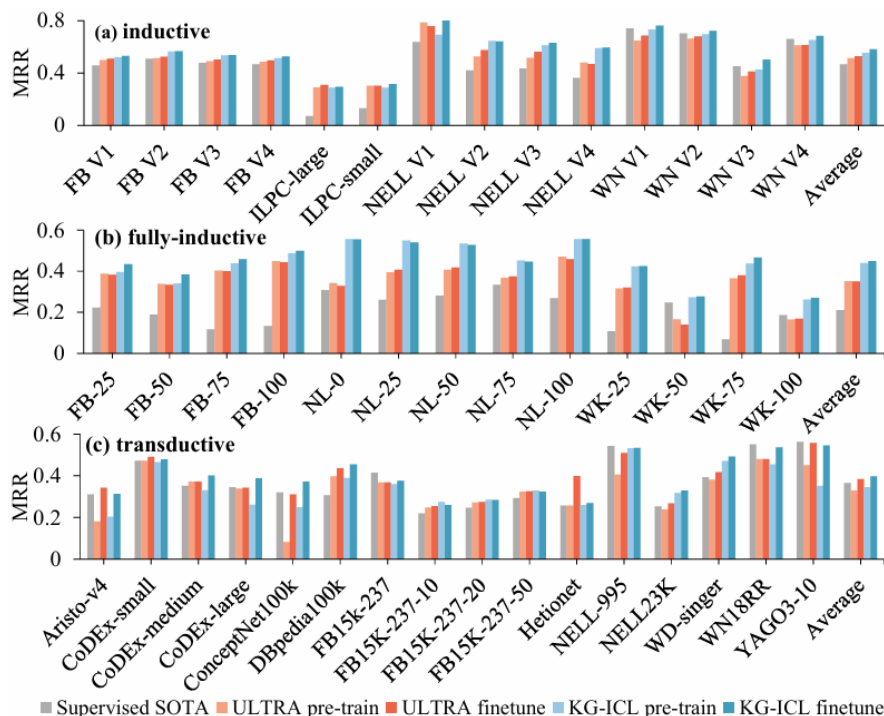


Figure 2: MRR results on various KGs.

Table 11: Detailed results on 43 datasets.

Datasets	Supervised SOTA		ULTRA pre-train		KG-ICL pre-train		ULTRA finetune		KG-ICL finetune	
	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
FB V1	0.457	0.589	0.498	0.656	0.520	0.678	0.509	0.670	<b>0.531</b>	<b>0.700</b>
FB V2	0.510	0.672	0.512	0.700	0.565	0.749	0.524	0.710	<b>0.568</b>	<b>0.768</b>
FB V3	0.476	0.637	0.491	0.654	0.535	0.695	0.504	0.663	<b>0.537</b>	<b>0.704</b>
FB V4	0.466	0.645	0.486	0.677	0.513	0.699	0.496	0.684	<b>0.525</b>	<b>0.706</b>
ILPC-large	0.070	0.146	0.290	0.424	0.288	0.412	<b>0.308</b>	<b>0.431</b>	0.295	0.411
ILPC-small	0.130	0.251	0.302	0.443	0.288	0.446	0.303	0.453	<b>0.316</b>	<b>0.473</b>
NELL V1	0.637	0.866	0.785	0.913	0.693	0.915	0.757	0.878	<b>0.841</b>	<b>0.995</b>
NELL V2	0.419	0.601	0.526	0.707	<b>0.644</b>	<b>0.835</b>	0.575	0.761	0.641	0.835
NELL V3	0.436	0.594	0.515	0.702	0.613	0.792	0.563	0.755	<b>0.631</b>	<b>0.799</b>
NELL V4	0.363	0.556	0.479	0.712	0.590	0.791	0.469	0.733	<b>0.594</b>	<b>0.802</b>
WN V1	0.741	0.826	0.648	0.768	0.733	<b>0.838</b>	0.685	0.793	<b>0.762</b>	0.827
WN V2	0.704	<b>0.798</b>	0.663	0.765	0.696	0.783	0.679	0.779	<b>0.721</b>	0.787
WN V3	0.452	0.568	0.376	0.476	0.425	0.548	0.411	0.546	<b>0.503</b>	<b>0.626</b>
WN V4	0.661	0.743	0.611	0.705	0.652	0.722	0.614	0.720	<b>0.683</b>	<b>0.749</b>
FB-25	0.223	0.371	0.388	0.640	0.396	0.656	0.383	0.635	<b>0.434</b>	<b>0.694</b>
FB-50	0.189	0.325	0.338	0.543	0.341	0.559	0.334	0.538	<b>0.384</b>	<b>0.598</b>
FB-75	0.117	0.218	0.403	0.604	0.438	0.633	0.400	0.598	<b>0.458</b>	<b>0.664</b>
FB-100	0.133	0.271	0.449	0.642	0.487	0.694	0.444	0.643	<b>0.499</b>	<b>0.703</b>
NL-0	0.309	0.506	0.342	0.523	<b>0.557</b>	<b>0.777</b>	0.329	0.551	0.555	0.765
NL-25	0.261	0.464	0.395	0.569	<b>0.550</b>	<b>0.736</b>	0.407	0.596	0.540	0.730
NL-50	0.281	0.453	0.407	0.570	<b>0.534</b>	0.704	0.418	0.595	0.528	<b>0.708</b>
NL-75	0.334	0.501	0.368	0.547	<b>0.452</b>	0.673	0.374	0.570	0.446	<b>0.681</b>
NL-100	0.269	0.431	0.471	0.651	0.556	0.762	0.458	0.684	<b>0.557</b>	<b>0.766</b>
WK-25	0.107	0.169	0.316	0.532	0.423	0.621	0.321	0.535	<b>0.425</b>	<b>0.628</b>
WK-50	0.247	0.362	0.166	0.324	0.273	0.430	0.140	0.280	<b>0.277</b>	<b>0.432</b>
WK-75	0.068	0.135	0.365	0.537	0.437	0.602	0.380	0.530	<b>0.466</b>	<b>0.626</b>
WK-100	0.186	0.309	0.164	0.286	0.262	0.409	0.168	0.286	<b>0.270</b>	<b>0.415</b>
AristoV4	0.311	0.447	0.182	0.282	0.203	0.306	<b>0.343</b>	<b>0.496</b>	0.313	0.480
CoDEx-small	0.473	0.663	0.472	0.667	0.465	0.654	<b>0.490</b>	<b>0.686</b>	0.479	0.662
CoDEx-medium	0.352	0.490	0.372	0.525	0.330	0.474	0.372	0.525	<b>0.402</b>	<b>0.565</b>
CoDEx-large	0.345	0.473	0.338	0.469	0.261	0.376	0.343	0.478	<b>0.388</b>	<b>0.508</b>
ConceptNet100K	0.320	0.553	0.082	0.162	0.249	0.416	0.310	0.529	<b>0.371</b>	<b>0.584</b>
DBpedia100K	0.306	0.418	0.398	0.576	0.390	0.541	0.436	0.603	<b>0.455</b>	<b>0.604</b>
FB15k-237	<b>0.415</b>	<b>0.599</b>	0.368	0.564	0.359	0.541	0.368	0.564	0.376	0.538
FB15k-237-10	0.219	0.337	0.248	0.398	<b>0.274</b>	<b>0.433</b>	0.254	0.411	0.260	0.416
FB15k-237-20	0.247	0.391	0.272	0.436	<b>0.285</b>	0.454	0.274	0.445	0.284	<b>0.456</b>
FB15k-237-50	0.293	0.458	0.324	0.526	<b>0.329</b>	0.520	0.325	<b>0.528</b>	0.324	0.499
Hetionet	0.257	0.403	0.257	0.379	0.260	0.371	<b>0.399</b>	<b>0.538</b>	0.269	0.402
NELL-995	<b>0.543</b>	0.651	0.406	0.543	0.532	0.653	0.509	0.660	0.534	<b>0.672</b>
NELL23K	0.253	0.419	0.239	0.408	0.317	0.532	0.268	0.450	<b>0.329</b>	<b>0.552</b>
WD-singer	0.393	0.500	0.382	0.498	0.470	0.582	0.417	0.526	<b>0.493</b>	<b>0.599</b>
WN18RR	<b>0.551</b>	<b>0.666</b>	0.480	0.614	0.455	0.527	0.480	0.614	0.536	0.637
YAGO3-10	<b>0.563</b>	0.708	0.451	0.615	0.352	0.503	0.557	<b>0.710</b>	0.545	0.688
Average	0.351	0.493	0.396	0.557	0.442	0.606	0.421	0.590	<b>0.473</b>	<b>0.638</b>

## 消融实验

### ■ 模块消融变体

- w/o prompt graph
- w/o unified tokenizer
- w/ GraIL's labeling

### ■ 实验结果

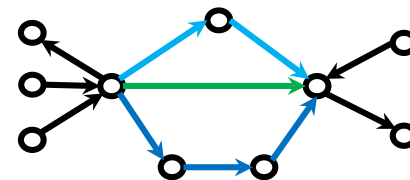
- 提示图对上下文学习至关重要
- 统一分词器对结果有较大影响，但去掉仍可运行
- 可以结合其他编码方式，如GraIL的one-hot编码

Table 2: Ablation study results in various settings.

Models	Inductive (14 KGs)		Fully-inductive (13 KGs)		Transductive (16 KGs)		Average (43 KGs)	
	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
Intact model	0.554	0.707	0.439	0.635	0.346	0.493	0.442	0.606
w/o prompt graph	0.219	0.420	0.105	0.228	0.076	0.143	0.132	0.259
w/o unified tokenizer	0.511	0.660	0.419	0.617	0.296	0.453	0.403	0.570
w/ GraIL's labeling	0.531	0.704	0.434	0.634	0.343	0.492	0.431	0.604

### ■ 提示图消融变体

- neighbor & 3-hop path (intact model)
- neighbor
- 1-hop path
- 2-hop path
- 3-hop path



### ■ 实验结果

- 仅有一跳或两跳内路径不足以支持有效推理

Table 3: MRR results on diverse prompt graphs.

Models	Inductive (14 KGs)		Fully-inductive (13 KGs)		Transductive (16 KGs)		Average (43 KGs)	
	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
Neighbor & 3-hop path	0.554	0.707	0.439	0.635	0.346	0.493	0.442	0.606
Neighbor	0.552	0.702	0.429	0.628	0.311	0.459	0.425	0.590
1-hop path	0.208	0.449	0.145	0.314	0.112	0.216	0.153	0.322
2-hop path	0.256	0.419	0.137	0.285	0.125	0.235	0.171	0.310
3-hop path	0.544	0.697	0.409	0.601	0.294	0.464	0.410	0.582



## 分析实验：模型鲁棒性&示例利用效率

### 模型鲁棒性

#### ■ 低资源关系鲁棒性

- 常见transductive和inductive数据集
- 统计各关系的训练样本数量
- 按训练样本数量分组
- 分析各组关系的平均MRR值

#### ■ KG-ICL在低资源关系上具有良好鲁棒性

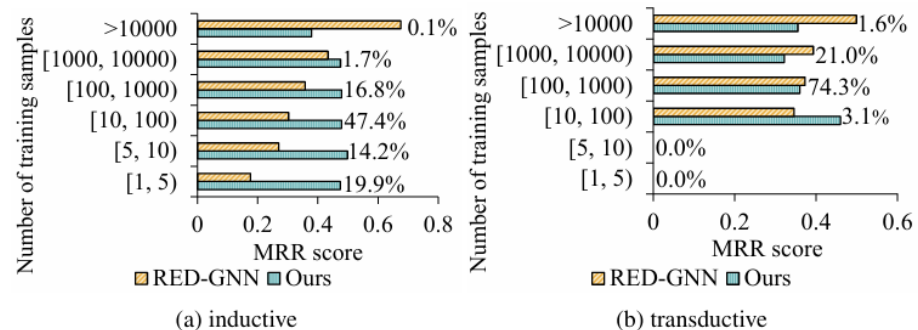


Figure 4: Average MRR results of relation subgroups. Relations in the inductive and transductive dataset groups are divided into 6 subgroups based on the number of training samples, and the results represent the average scores for the relations within their respective subgroups. The percentage on the right side of each data bar indicates the proportion of relations in that subgroup to the total number of relations in their respective groups.

### 示例利用效率

#### ■ 示例数量鲁棒性

- {1,3,5,10,20}个示例样本
- 统计分析各类实验结果

#### ■ KG-ICL在少量示例条件下有良好的鲁棒性

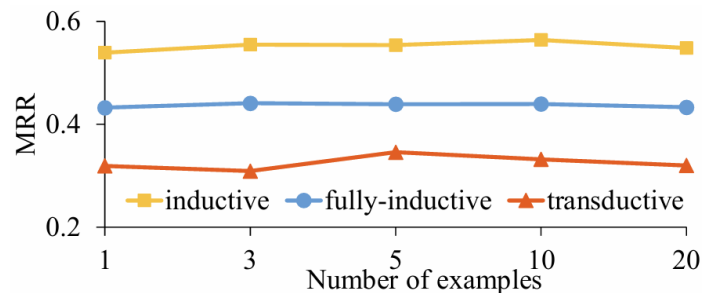


Figure 3: MRR with different numbers of examples.



## 总结与展望

### ■ 总结

- 提出一种基于上下文提示的**知识图谱推理基座模型**
- 提出**两种泛化工具**：提示图和统一分词器
- 提出**两个图神经网络**，分别用于提示编码和实体打分
- 3种设定43个数据集的实验结果表明，所提基座模型具有**强大通用推理能力**

### ■ 展望

- 更好的关系建模，提出更强的知识图谱推理基座模型
- 实验表明KG-ICL的低资源鲁棒性，未来可用于小型的个人图谱
- 开发更通用的图基座模型，用于推荐系统等图谱驱动的下游应用





# 谢谢!

The source code is available at <https://github.com/nju-websoft/KG-ICL>.

This work was funded by National Natural Science Foundation of China (Nos. 62272219 and 62406136), Postdoctoral Fellowship Program of CPSF (No. GZC20240685), and CCF-Tencent Rhino-Bird Open Research Fund.



Contact: [yncui.nju@gmail.com](mailto:yncui.nju@gmail.com)  
Homepage: <https://yncui-nju.github.io>

