

# 图神经网络赋能知识图谱研究与应用

报告人：刘瑜

报告时间：2023.04.07



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

# 目录

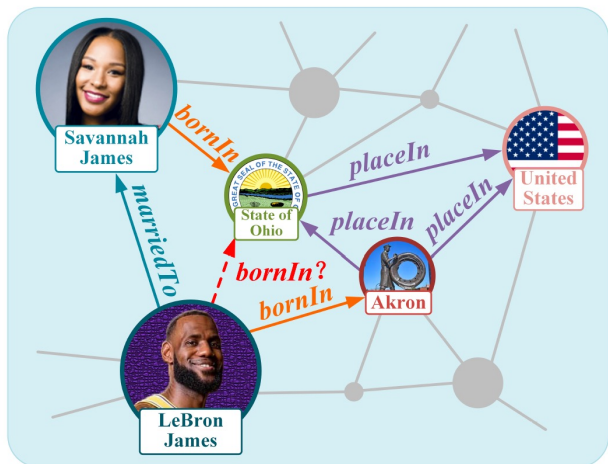
---

- 基于图神经网络的知识图谱推理
  - Transductive学习
  - Inductive学习
  - 复杂逻辑查询
- 大模型下的知识图谱审视
  - LLM和知识图谱的关系
  - LLM对知识图谱的影响
  - LLM和知识图谱的未来发展

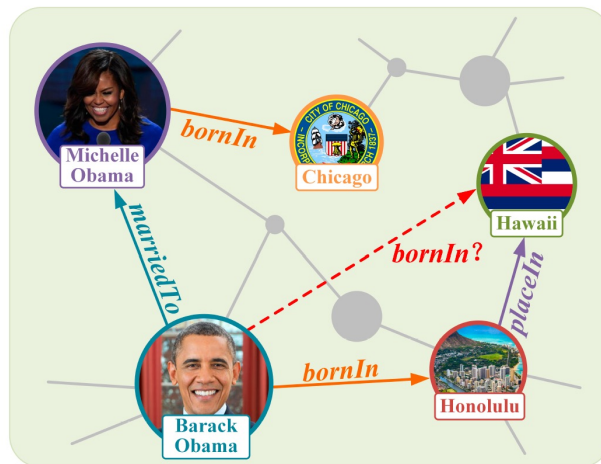
# 基于图神经网络的知识图谱推理

## 通用知识图谱推理研究方向

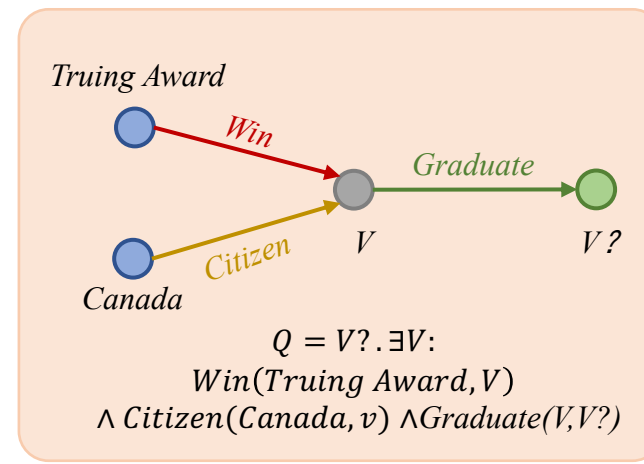
1. Transductive推理：传统的直推学习，测试集上的实体都存在于训练集中
2. Inductive推理：归纳学习，允许在测试集出现新兴实体，更接近真实场景
3. First-Order Logic查询：在知识图谱上执行一阶逻辑查询，包括关系映射和逻辑操作



(a) training graph and transductive reasoning.



(b) inductive reasoning.



(c) complex logic query

# Inductive推理- SNRI-IJCAI22

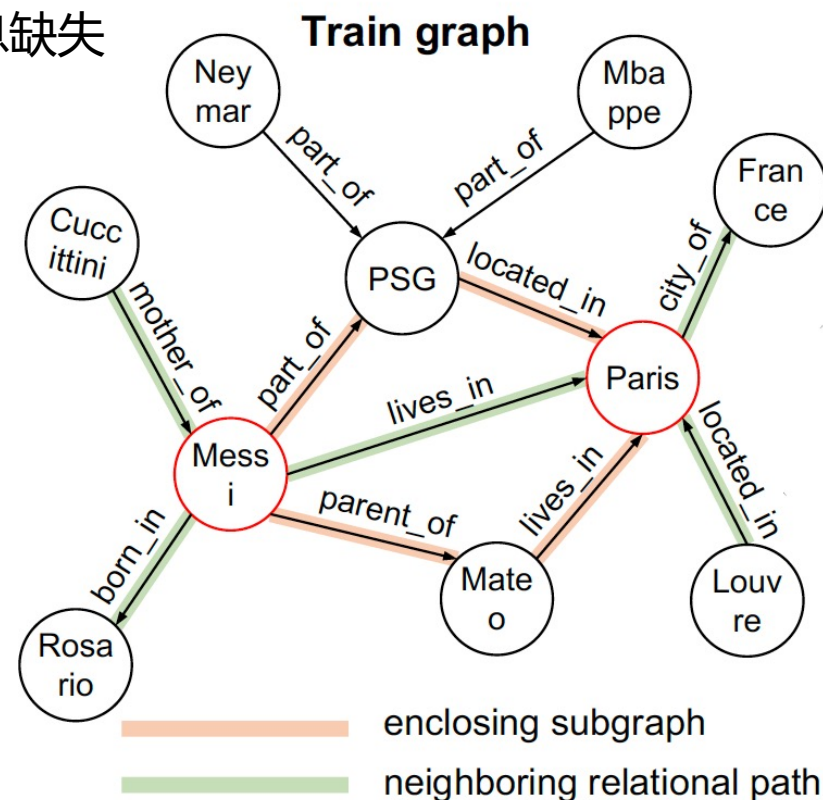
## 动机

1. 封闭子图没有考虑实体的全部邻居关系，导致价值信息缺失
2. 封闭子图存在稀疏性，难以获得有用信息

### 经典GraIL模型

给定 $(u, r, v)$ ，封闭子图的构造步骤：

1. 抽取头实体 $u$ 的 $k$ -hop邻居集合 $N_k(u)$
2. 抽取尾实体 $v$ 的 $k$ -hop邻居集合 $N_k(v)$
3. 将两个集合的并集 $N_k(u) \cup N_k(v)$ 作为封闭子图的节点
4. KG上实体的关系作为封闭子图的关系



# Inductive推理- SNRI-IJCAI22

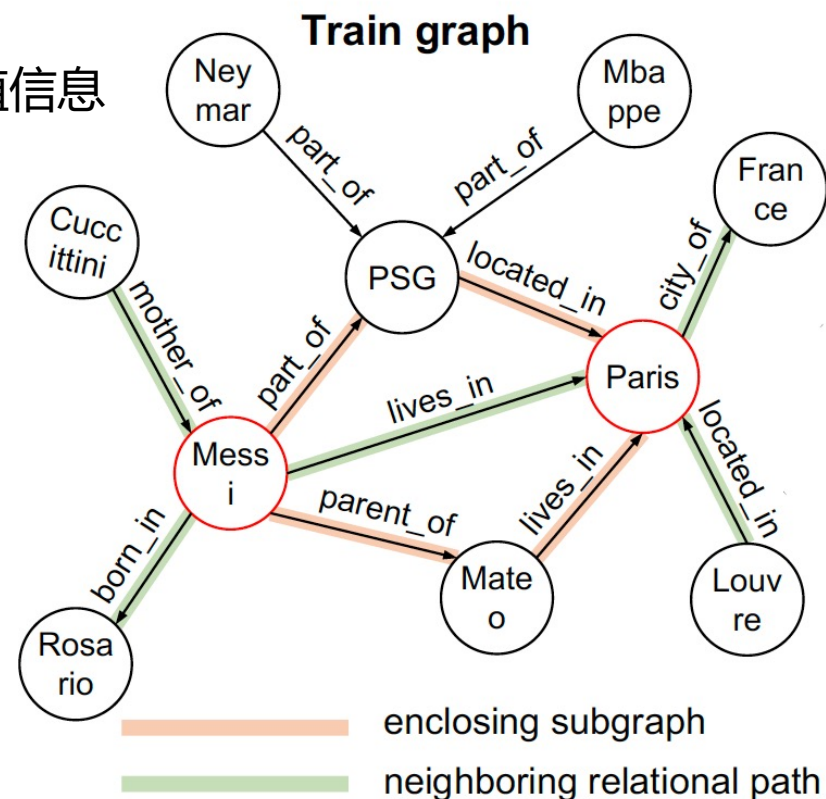
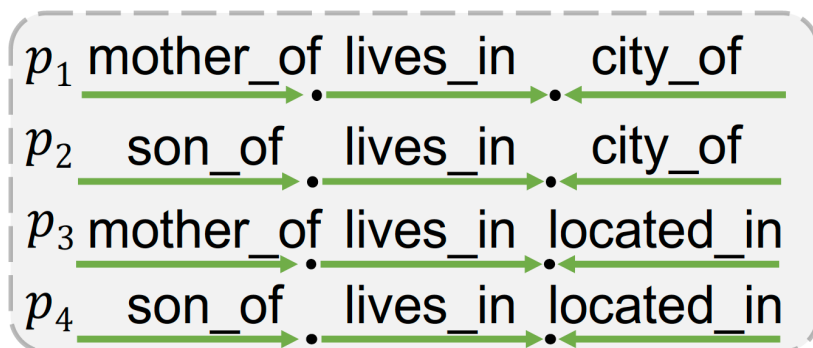
## 创新

问题1. 封闭子图没有考虑实体的全部邻居关系，导致价值信息缺失

方法1: 保留实体的完整邻居关系

问题2. 封闭子图存在稀疏性，难以获得有用信息

方法2: 利用多条邻居关系路径，来弥补稀疏问题



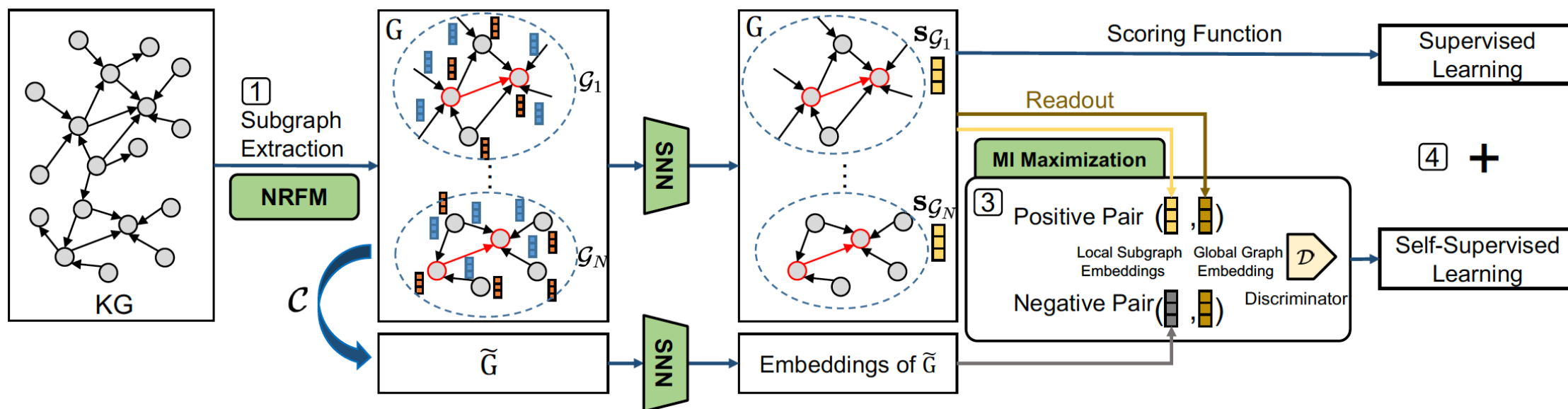
# Inductive推理- SNRI-IJCAI22

## SNRI 模型

**NRFM** : Neighboring Relational Feature Module , 邻居关系特征抽取 , 实现子图抽取 , 并初始化节点特征

**SNN** : Subgraph Neural Network , 子图神经网络 , 在封闭子图上利用GNN , 并加入关系路径模块

**MIM** : Mutual Information Maximization , 互信息最大化 , 用于全局子图和局部子图之间信息交互



# Inductive推理- SNRI-IJCAI22

## 模块一: 邻居关系特征模块 ( NRFM )

作用: 实现子图抽取, 并初始化节点特征

### 1. 构建局部封闭子图

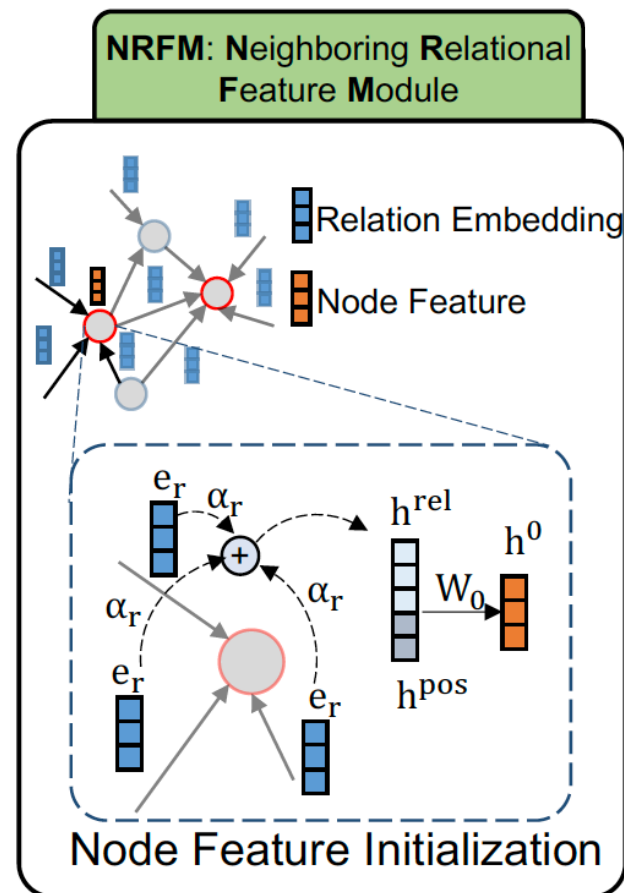
给定 $(u, r, v)$ , 封闭子图的构建方式和GraIL论文相同, 不同之处在于, 本文同时保留了每个节点的完整的邻居关系 $N^r(u)$ 。

### 2. 节点特征初始化: 关系特征+位置特征

$$\mathbf{h}_i^0 = \mathbf{W}_0[\mathbf{h}_i^{rel} \oplus \mathbf{h}_i^{pos}].$$

$$\mathbf{h}_i^{rel} = \sum_{r \in N^r(i)} \alpha_r \mathbf{e}_r$$

$$\mathbf{h}_i^{pos} = [\text{one-hot}(d(i, u)) \oplus \text{one-hot}(d(i, v))]$$

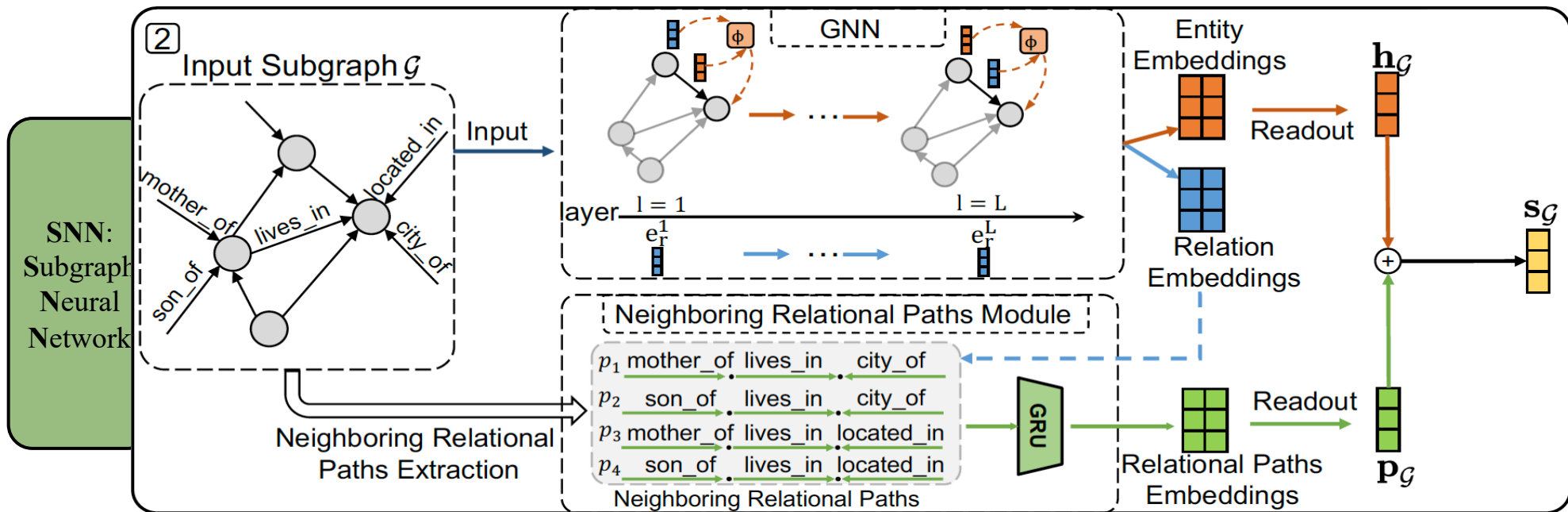


# Inductive推理- SNRI-IJCAI22

## 模块二: 子图神经网络模块 (SNN)

作用: 在封闭子图上利用GNN进行信息传递, 此外加入关系路径模块来弥补子图稀疏性

1. 子图表示更新模块: 利用GNN更新节点表示 $h$ 和关系表示 $e_r$ , 聚合所有的节点表示得到子图表示 $h_G$
2. 关系路径融合模块: 利用GRU学习多条关系路径的路径表示 $p$ , 聚合邻居路径表示得子图路径表示 $p_G$





# Inductive推理- SNRI-IJCAI22

## 模块三: 互信息最大化模块 (MI Maximization)

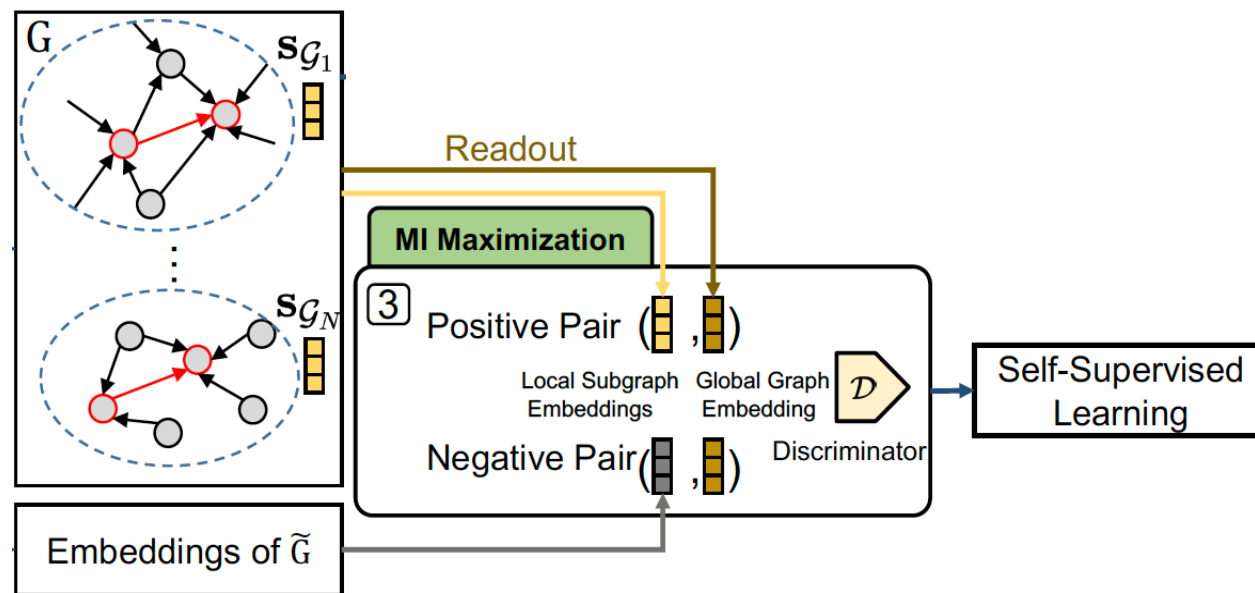
作用: 为了避免过于强调局部结构, 本文通过最大化局部-全局 (subgraph-graph) 互信息, 进一步以全局的方式对相邻关系进行建模, 即让局部子图向整个KG看齐, 寻求使相邻关系特征和路径能够捕获整个KG的全局信息。

1. KG上的全局表示 $s_G$

$$s_G = \frac{1}{N} \sum_{i=1}^N s_{G_i}$$

2. 计算全局表示 $s_G$ 和局部子图 $s_{G_i}$ 表示相关性

$$D(s_{G_i}, s_G) = \sigma(s_{G_i}^T \mathbf{W}_{MI} s_G)$$



# Inductive推理- SNRI-IJCAI22

## 模块四: 多任务训练

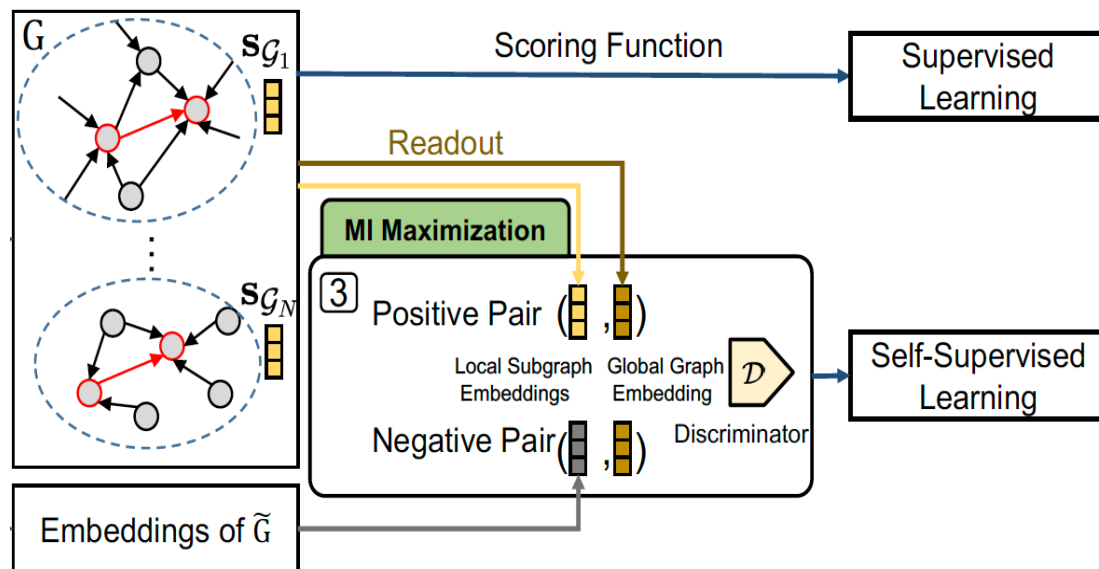
作用：采用多任务训练的方法，任务一在知识图谱上的链接预测任务；任务二最大化局部-全局 (subgraph-graph) 互信息。最终的目标函数包括两个部分： $\mathcal{L} = \mathcal{L}_{sup} + \lambda\mathcal{L}_{MI}$

### 1. 链接预测损失函数

$$\mathcal{L}_{sup} = \sum_{(u,r_t,v) \in \mathcal{G}} \max(0, f(u', r'_t, v') - f(u, r_t, v) + \gamma)$$

### 2. 互信息目标函数

$$\mathcal{L}_{MI} = \frac{1}{N + M} \left( \sum_{i=1}^N \mathbb{E}_{(\mathbf{X}, \mathbf{A})} [\log \mathcal{D}(\mathbf{s}_{G_i}, \mathbf{s}_G)] \right) + \sum_{j=1}^M \mathbb{E}_{(\tilde{\mathbf{X}}, \mathbf{A})} [\log (1 - \mathcal{D}(\tilde{\mathbf{s}}_{G_j}, \mathbf{s}_G))],$$



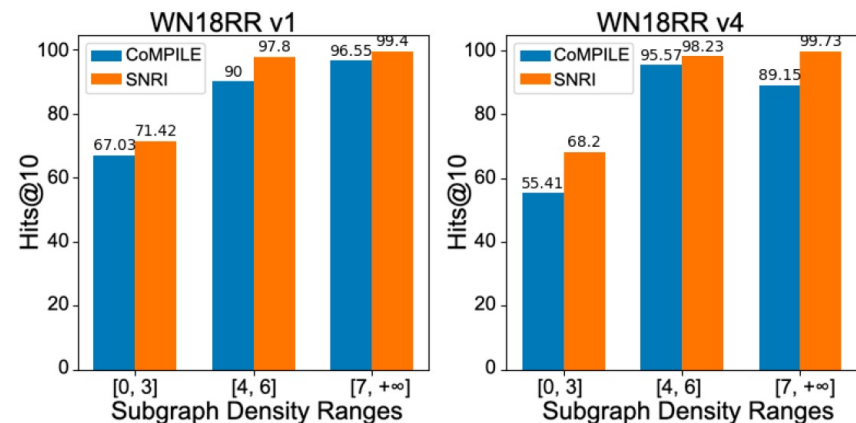
# Inductive推理- SNRI-IJCAI22

## 实验结果

Method	WN18RR							
	v1		v2		v3		v4	
	AP	H@10	AP	H@10	AP	H@10	AP	H@10
Neural-LP	86.02	74.37	83.78	68.93	62.90	46.18	82.06	67.13
DRUM	86.02	74.37	84.05	68.93	63.20	46.18	82.06	67.13
RuleN	90.26	80.85	89.01	78.23	76.46	53.39	85.75	71.59
GraIL	94.32	82.45	94.18	78.68	85.80	58.43	92.72	73.41
CoMPILE	98.23	83.60	99.56	79.82	93.60	60.69	<b>99.80</b>	75.49
SNRI	<b>99.10</b>	<b>87.23</b>	<b>99.92</b>	<b>83.10</b>	<b>94.90</b>	<b>67.31</b>	99.61	<b>83.32</b>

Method	FB15k-237							
	v1		v2		v3		v4	
	AP	H@10	AP	H@10	AP	H@10	AP	H@10
Neural-LP	69.64	52.92	76.55	58.94	73.95	52.90	75.74	55.88
DRUM	69.71	52.92	76.44	58.73	74.03	52.90	76.20	55.88
RuleN	75.24	49.76	88.70	77.82	91.24	87.69	91.79	85.60
GraIL	84.69	64.15	90.57	81.80	91.68	82.83	94.46	89.29
CoMPILE	85.50	67.64	91.68	82.98	<b>93.12</b>	84.67	<b>94.90</b>	87.44
SNRI	<b>86.69</b>	<b>71.79</b>	<b>91.77</b>	<b>86.50</b>	91.22	<b>89.59</b>	93.37	<b>89.39</b>

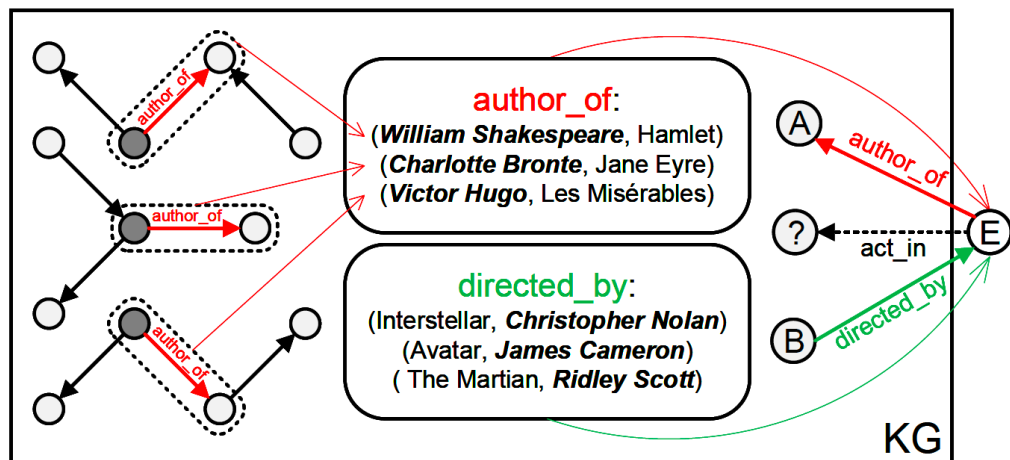
Method	WN18RR	
	v1	v4
SNRI w/o NRF	86.96	82.26
SNRI w/o NRP	85.91	82.01
SNRI w/o MI	84.84	82.43
SNRI	87.23	83.32



# Inductive推理- CFGA-AAAI22

## 动机

1. 以往GNN方法从查询三元组的局部子图中聚合邻域信息，但忽略了所有已知三元组在整个KG上更一般（广义）的关系语义。
2. 为了满足归纳场景，怎么样更好地表示查询三元组中某个明确（具体）的关系语义？



### • 关系语义

拥有相同关系的实体通常具有相似的语义，例如  $\langle X, \text{author\_of}, Y \rangle$ ，X是作者，Y是书籍。

### • 广义关系语义

一个实体通常拥有多个关系，该实体可能具有多个语义。同时，一个关系和多个实体有联系。融合了这些实体的关系，被认为是具有了广义的关系语义。

### • 具体关系语义

给定  $\langle E, \text{act\_in}, ? \rangle$ ，此时具体关系指的是查询关系 **act\_in**。

# Inductive推理- CFGA-AAAI22

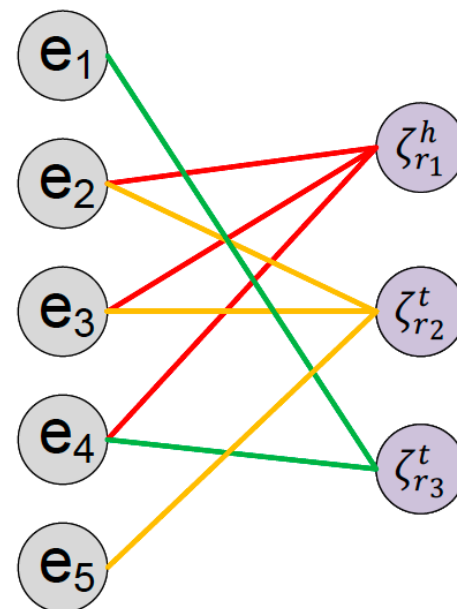
## 创新

问题1. GNN从查询三元组的局部子图中聚合邻域信息，但忽略了所有已知三元组在整个KG上更一般（广义）的关系语义。

方法1：为了得到广义的关系语义，利用超图将整个KG上与关系联系的多个实体信息聚合起来

问题2. 怎么样更好地表示查询三元组中某个具体（明确）的关系语义？

方法2: 为了得到具有明确语义的实体表示，针对特定的查询关系，设计了一个细粒度的关系生成器。

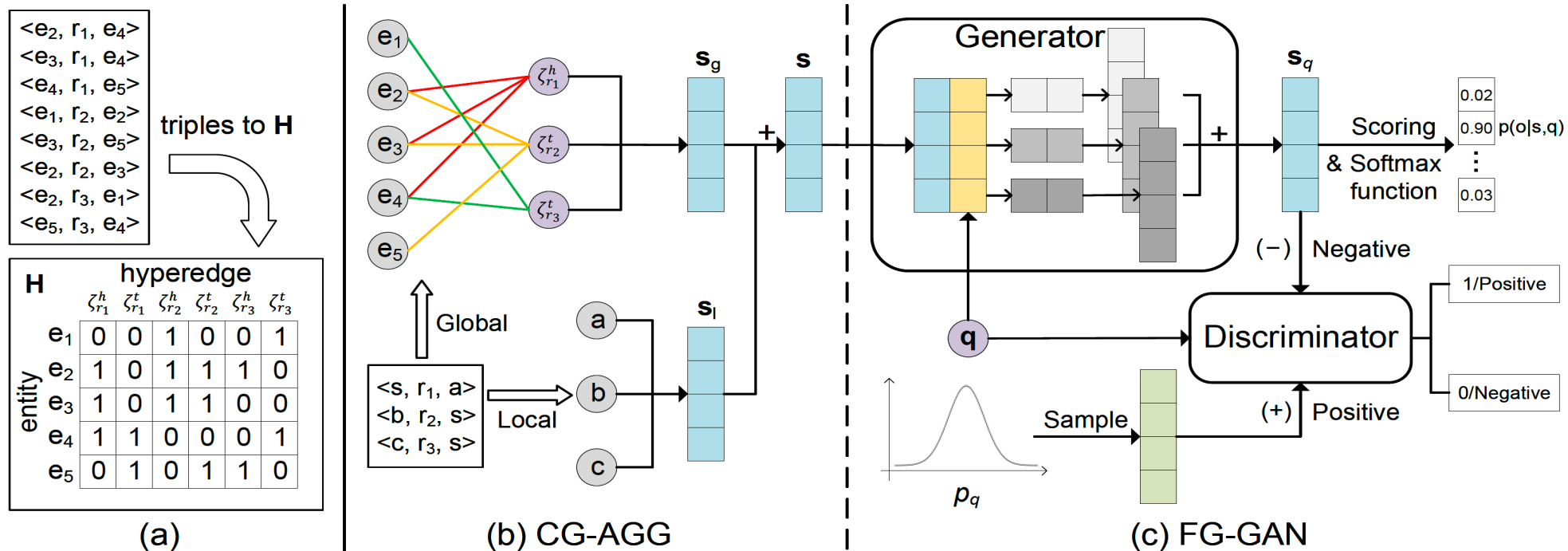


# Inductive推理- CFGA-AAAI22

## CFGA模型

**CG-AGG**：粗粒度聚合器（融合多个关系），利用超图GNN捕捉实体的全局信息，利用传统GNN捕捉实体的局部信息，得到粗粒度的实体表示

**FG-GAN**：细粒度生成器（聚焦查询关系），利用GAN生成查询关系分布，得到细粒度的实体表示



# Inductive推理- CFGA-AAAI22

## 模块一: 粗粒度聚合器 ( CG-AGG )

### 1. 全局聚合器

基于整个KG，计算节点的全局表示。将每个关系 $r$ 拆成两条超边 $\zeta_r^h$ 和 $\zeta_r^t$ ，分别表示与 $r$ 相连的头实体集合和尾实体集合。

那么超图的关联矩阵 $H$ ：

$$H_{ij} = \begin{cases} 1, & \text{if } e_i \in \zeta_j \\ 0, & \text{otherwise.} \end{cases}$$

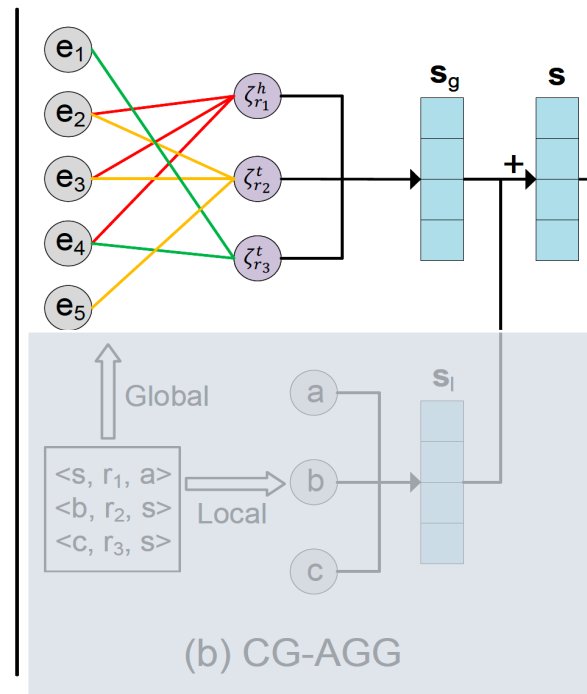
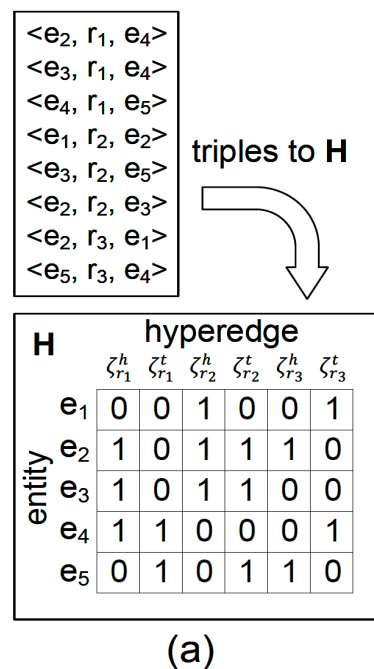
接着，将每个集合中的实体表示加权求和，计算所有超边的向量表示 $\mathbf{Z} \in \mathbb{R}^{2|R| \times d}$ 。

$$\mathbf{Z} = \mathbf{W}\mathbf{D}^{-1}\mathbf{H}^T\mathbf{E}$$

然后，对于每个节点 $s$ ，再将相邻的超边聚合，计算节点 $s$ 的

全局向量表示 $s_g \in \mathbb{R}^d$ ：

$$s_g = \sigma \left( \frac{1}{\sum_j h_{s,j}} \mathbf{h}_s^T \mathbf{Z} \Theta_g \right)$$



# Inductive推理- CFGA-AAAI22

## 模块一: 粗粒度聚合器 ( CG-AGG )

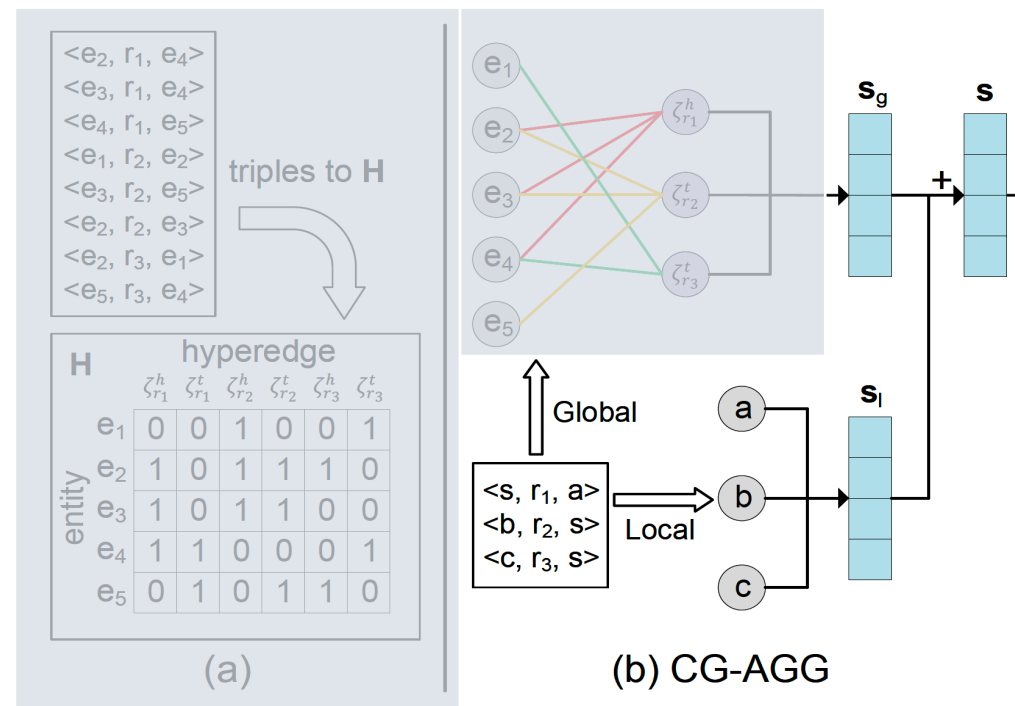
### 2. 局部聚合器

基于查询三元组的局部子图，计算节点s的局部表示，方式与面向Inductive任务的GNN相似。具体地，对于节点s，将其一跳邻居实体加权求和得到局部表示 $s_1$ ：

$$s_1 = \sigma \left( \frac{1}{\sum_i n_{s,i}} \mathbf{n}_s^T \mathbf{E} \Theta_1 \right)$$

最后，将全局表示 $s_g$ 和局部表示 $s_1$ 相加，得到节点s的粗粒度表示：

$$\mathbf{s} = A(s) = \alpha \mathbf{s}_1^T + (1 - \alpha) \mathbf{s}_g^T$$





# Inductive推理- CFGA-AAAI22

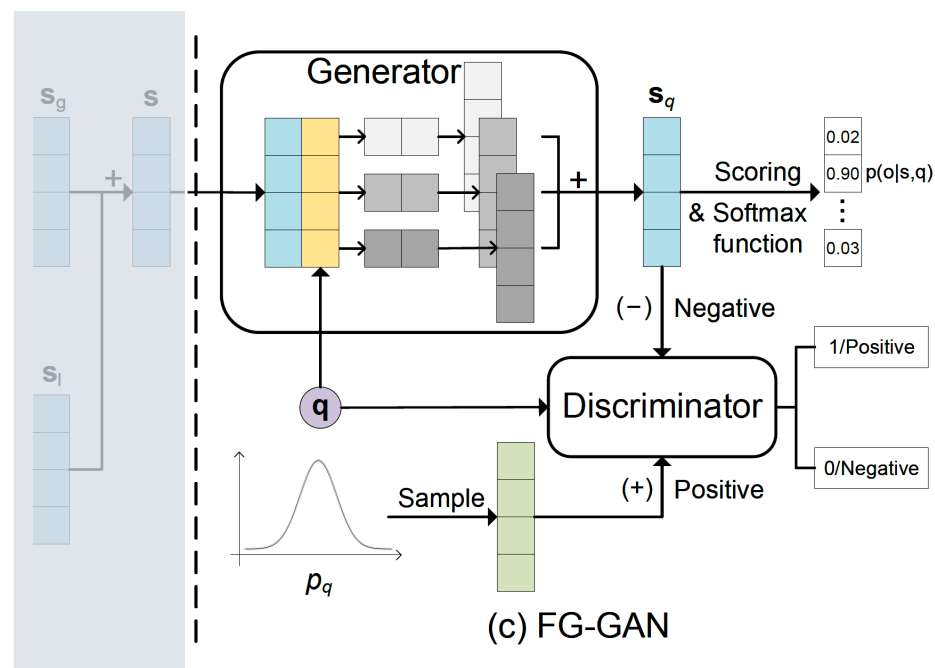
## 模块二: 细粒度生成器 ( FG-GAN )

### 1. 细粒度生成器的作用

给定查询三元组 $\langle s, q, ? \rangle$ ，基于具体的查询关系 $q$ ，得到**实体的细粒度表示**。本文以粗粒度的实体表示和查询关系 $q$ 为输入，利用生成对抗网络GAN生成关系 $q$ 的对应分布，将其视为实体的细粒度表示。

### 2. 利用生成模型的动机

本文将**每个关系看作是一个分布**，相同的关系具有相同的分布。同一关系下的实体通常具有某些相似性，即**拥有相同关系的实体表示应该和这个关系具有相同的分布**。因此，GAN中生成器G生成实体分布，判别器D区分该表示是来自先验分布（正）还是来自生成器（负）。



# Inductive推理- CFGA-AAAI22

## 模块二: 细粒度生成器 ( FG-GAN )

### 3. 生成对抗模型GAN

具体地，在给定查询三元组，本文使用生成器G利用CNN来获得以查询关系q为条件的**实体细粒度表示**，

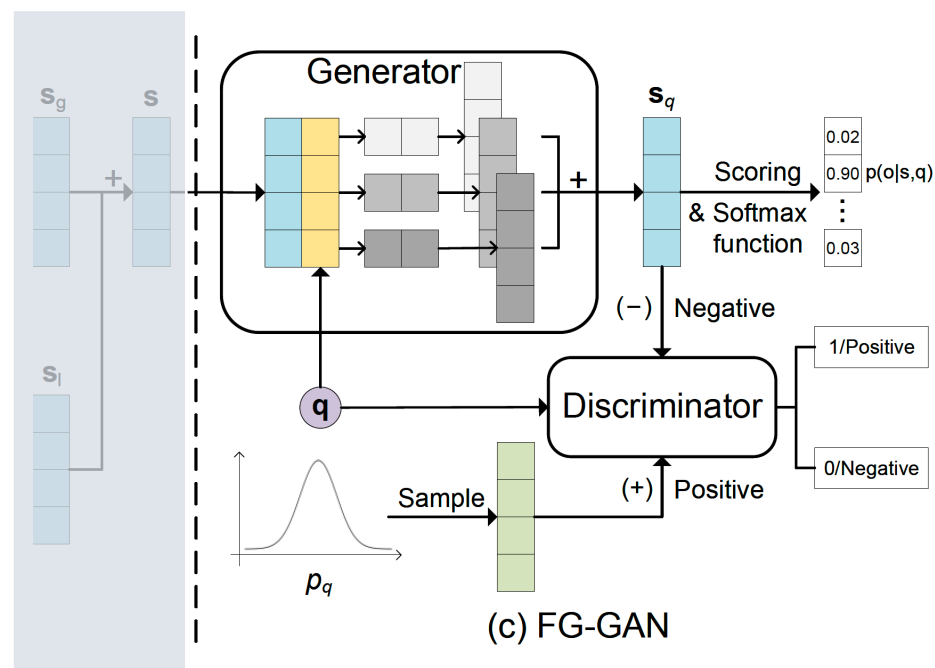
$$s_q = G(A(s), q) = \sigma \left( \sum_{i=1}^K ([s; \mathbf{R}^T \mathbf{q}] * \Omega_i) \right)$$

判别器D区分该表示是来自先验分布（正）还是来自生成器（负）。

$$D(\mathbf{x}, q) = \text{Sigmoid}(\mathbf{W}_1(\mathbf{W}_0(\mathbf{x}||\mathbf{q}) + b_0) + b_1)$$

最终，训练目标期望查询关系的先验分布和具有该关系的实体的分布相同。

$$\min_{G,A} \max_D V(G, A, D) = \mathbb{E}_{\mathbf{z} \sim p_q} [\log D(\mathbf{z}, q)] + \mathbb{E}_{s \sim p_{data}} [\log(1 - D(G(A(s), q), q)].$$



# Inductive推理- CFGA-AAAI22

## 实验结果

Model	FB15k-237-Sub				FB15k-237-Obj				NELL-995-Sub				NELL-995-Obj			
	Hits@N				Hits@N				Hits@N				Hits@N			
	MRR	1	3	10	MRR	1	3	10	MRR	1	3	10	MRR	1	3	10
MEAN	<u>0.376</u>	<u>27.9</u>	<u>42.2</u>	<u>56.6</u>	0.128	6.6	13.6	25.5	0.296	17.3	33.9	57.4	0.253	17.3	28.3	40.6
LAN	0.359	25.6	40.9	56.1	<u>0.138</u>	7.5	<u>15.1</u>	<u>26.3</u>	<u>0.313</u>	<u>17.5</u>	<u>37.8</u>	<u>60.7</u>	0.250	16.0	29.7	<u>42.6</u>
ELPE	0.371	26.6	41.8	53.3	<u>0.126</u>	6.4	13.0	24.1	<u>0.287</u>	16.7	32.7	57.1	0.248	16.2	27.5	40.2
OOS	0.325	23.8	37.1	48.9	0.132	<u>8.1</u>	14.3	23.7	0.263	16.6	28.6	51.4	<u>0.290</u>	<u>22.0</u>	<u>32.3</u>	42.1
CFAG	<b>0.393</b>	<b>29.5</b>	<b>44.0</b>	<b>57.5</b>	<b>0.156</b>	<b>9.4</b>	<b>17.0</b>	<b>27.6</b>	<b>0.360</b>	<b>23.9</b>	<b>41.9</b>	<b>62.0</b>	<b>0.312</b>	<b>23.8</b>	<b>34.2</b>	<b>45.9</b>

Model	MRR	Hits@1	Hits@3	Hits@10
CFAG w/o A	0.348	23.6	39.7	56.4
CFAG w/o G	0.342	23.5	38.3	56.8
CFAG w/o A, G	0.334	22.3	37.8	54.4
CFAG	<b>0.360</b>	<b>23.9</b>	<b>41.9</b>	<b>62.0</b>

Table 3: Ablation results on NELL-995-Sub.

Model	MRR	Hits@1	Hits@3	Hits@10
CFAG r/w TransE	0.350	23.6	40.4	58.7
CFAG r/w DistMult	0.319	21.0	36.1	55.9
CFAG	<b>0.360</b>	<b>23.9</b>	<b>41.9</b>	<b>62.0</b>

Table 6: Experimental results on NELL-995-Sub with different generators.

# 复杂逻辑查询- GNN-QE -ICML22

## 动机

- 1.传统的符号方法遍历一个完整的知识图来提取答案，这为每一步提供了良好的解释，但真实世界中KG是不完整的。
2. 神经网络方法通常利用几何表示来学习复杂一阶逻辑查询的向量表示，但推理过程缺乏可解释性。

**Natural query:** What musical instruments did Minnesota-born Nobel Prize winners play?

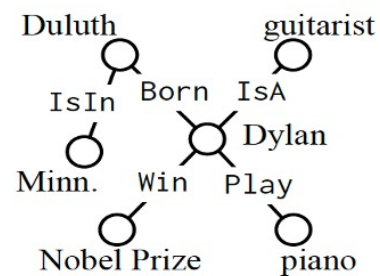
**EPFO query:**  $Q[A] \triangleq ?A : \exists V s.t. \text{Born}(\text{Minnesota}, V) \wedge \text{Win}(\text{NobelPrize}, V) \wedge \text{Play}(V, A)$

一阶逻辑查询由常量、变量、关系运算和逻辑运算组成。

该任务的**核心问题**包括：

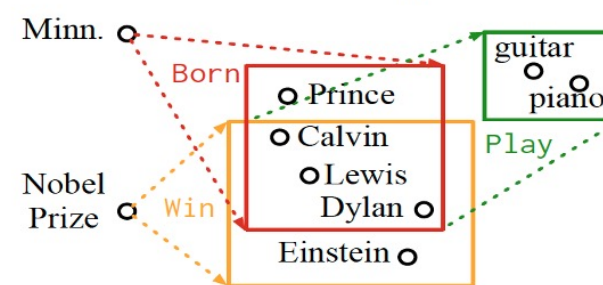
- 1.常量和变量是KG实体，如何表示实体？
- 2.关系运算 $R(a, b)$ 是二元函数，如何表示关系运算？
3. 逻辑运算包括 $\wedge$ ， $\vee$ ， $\neg$ ，如何表示逻辑运算？

### Forms of Knowledge



Symbolic Method

### Reasoning



Geometric Operation

# 复杂逻辑查询- GNN-QE -ICML22

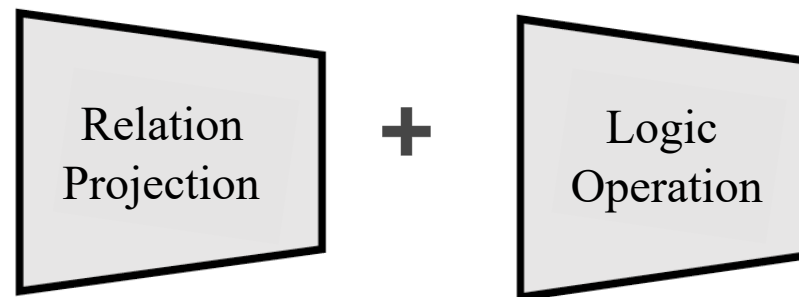
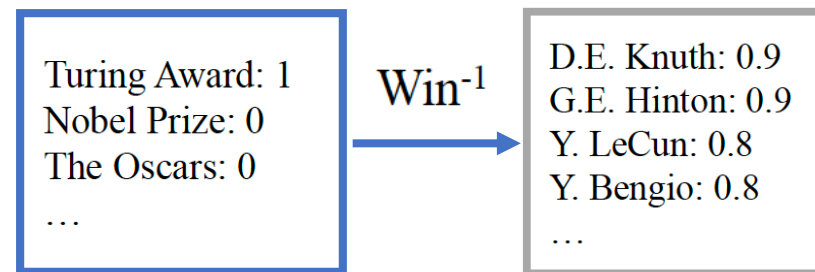
## 创新

问题1. 神经网络方法中间变量不透明，推理过程缺乏可解释性

方法1: 将每个节点用一个模糊集表示，集合中的元素是KG上所有实体，并额外用概率值表示实体的置信度

问题2: 如何处理复杂逻辑查询的两类操作，包括关系投影和逻辑操作 ( $\wedge \vee \neg$ )

方法2: 将复杂一阶逻辑 (FOL) 查询分解为模糊集上的表达式，并且为这两类操作定义了新的模型



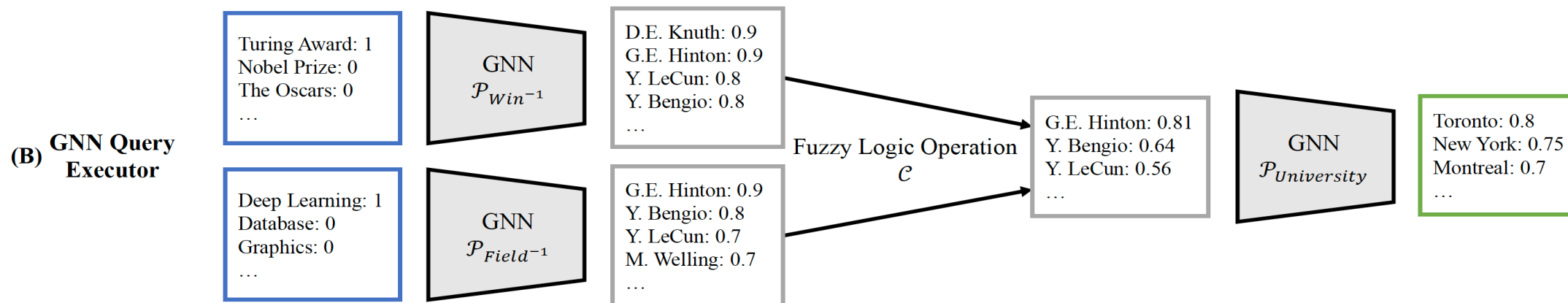
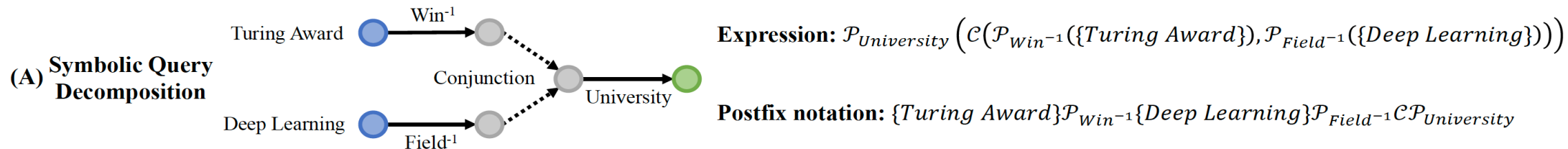
# 复杂逻辑查询- GNN-QE -ICML22

## GNN-QE模型

**符号查询分解**：将一阶逻辑查询转换为基本运算的表达式（类比算术表达式的后缀形式）

**GNN查询执行**：学习一个图神经网络模型，通过执行表达式来检索答案

$$q = v. \exists u: Win(u, Turing\ Award) \wedge Field(u, Deep\ Learning) \wedge University(u, v)$$



# 复杂逻辑查询- GNN-QE-ICML22

## 模块一: 符号查询分解

### 1. 一阶逻辑查询转换为表达式

- (1) 一阶逻辑查询包括常量、变量、关系运算和逻辑运算。
- (2) 针对FOL的常量和变量, 采用模糊集表示  $\mathcal{A} = (\mathcal{U}, x)$ , 缩写为  $\mathcal{x}$ 。
- (3) 针对FOL的关系运算和逻辑运算, 定义了模糊集上的四种操作符。
- (4) 通过这些操作符, 将查询语句转换成表达式, 如下图所示。

**Relation Projection:**  $\mathcal{P}_q(\mathcal{x})$

**Conjunction:**  $\mathcal{C}(\mathcal{x}, \mathcal{y})$

**Disjunction:**  $\mathcal{D}(\mathcal{x}, \mathcal{y})$

**Negation:**  $\mathcal{N}(\mathcal{x})$

- ① 自然语句: Which universities do the Turing Award winners of deep learning work in?
- ② FOL查询语句:  $Q = V?. \exists V: Win(Turing Award, V) \wedge Field(Deep Learning, v) \wedge University(V, V?)$
- ③ 表达式:  $\mathcal{P}_{University} \left( \mathcal{C} \left( \mathcal{P}_{Win^{-1}}(\{Turing Award\}), \mathcal{P}_{Field}^{-1}(\{Deep Learning\}) \right) \right)$
- ④ 后缀表达式:  $\{Turing Award\} \mathcal{P}_{Win^{-1}} \{Deep Learning\} \mathcal{P}_{Field}^{-1} \mathcal{C} \mathcal{P}_{University}$

# 复杂逻辑查询- GNN-QE-ICML22

## 模块二: GNN查询执行

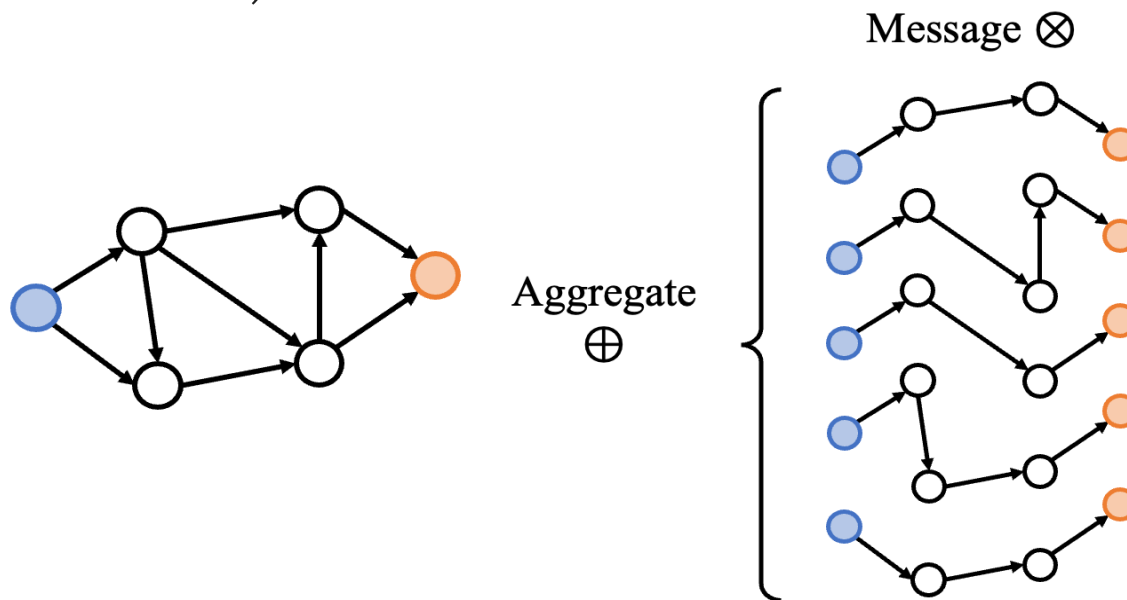
### 1. 关系运算操作(NBFnet模型)

为了解决不完备知识图上的复杂查询，本文学习了一个图神经模型来执行关系投影 $\mathcal{P}_q(x)$ 。

$$\mathbf{h}_v^{(0)} \leftarrow \text{Indicator}(u, v, q)$$
$$\mathbf{h}_v^{(t)} \leftarrow \text{Aggregate} \left( \left\{ \text{Message} \left( \mathbf{h}_z^{(t-1)}, (z, r, v) \right) \mid (z, r, v) \in \mathcal{E}(v) \right\} \right)$$

#### NBFnet模型核心思想：

1. 给定头尾实体，利用深度遍历算法得到多条路径
2. 在每条路径上执行Message操作
3. 将多条路径进行聚合执行Aggregate操作





# 复杂逻辑查询- GNN-QE-ICML22

## 模块二: GNN查询执行

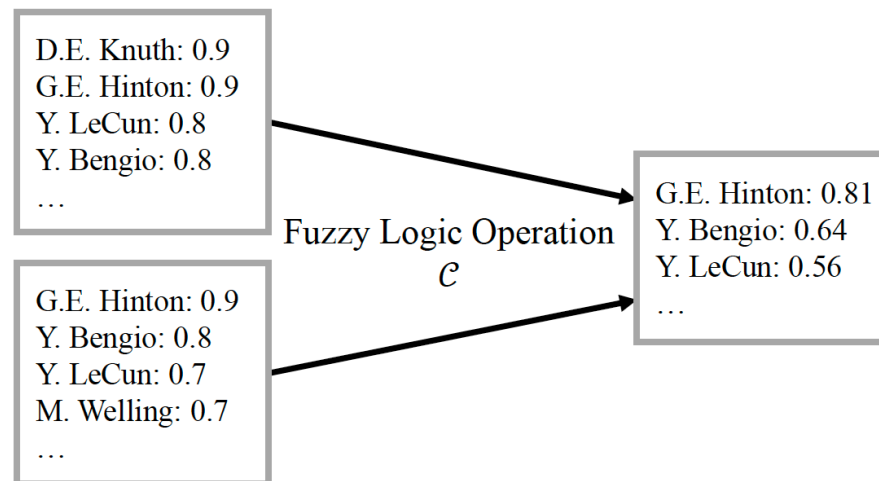
### 2. 逻辑运算操作

逻辑运算(即合取 $\mathcal{C}(x, y)$ 、析取 $\mathcal{D}(x, y)$ 和否定 $\mathcal{N}(x)$ )将多个关系投影结果粘合在一起,并为下一个关系投影生成输入模糊集。注意,不同于以往的表示学习方法,这些运算是在模糊集上执行的。

$$\mathcal{C}(x, y) = x \odot y$$

$$\mathcal{D}(x, y) = x + y - x \odot y$$

$$\mathcal{N}(x) = 1 - x$$



# 复杂逻辑查询- GNN-QE-ICML22

## 实验结果

Model	avg <sub>p</sub>	avg <sub>n</sub>	1p	2p	3p	2i	3i	pi	ip	2u	up	2in	3in	inp	pin	pni
FB15k																
GQE	28.0	-	54.6	15.3	10.8	39.7	51.4	27.6	19.1	22.1	11.6	-	-	-	-	-
Q2B	38.0	-	68.0	21.0	14.2	55.1	66.5	39.4	26.1	35.1	16.7	-	-	-	-	-
BetaE	41.6	11.8	65.1	25.7	24.7	55.8	66.5	43.9	28.1	40.1	25.2	14.3	14.7	11.5	6.5	12.4
CQD-CO	46.9	-	<b>89.2</b>	25.3	13.4	74.4	78.3	44.1	33.2	41.8	21.9	-	-	-	-	-
CQD-Beam	58.2	-	<b>89.2</b>	54.3	28.6	74.4	78.3	58.2	67.7	42.4	30.9	-	-	-	-	-
ConE	49.8	14.8	73.3	33.8	29.2	64.4	73.7	50.9	35.7	55.7	31.4	17.9	18.7	12.5	9.8	15.1
<b>GNN-QE</b>	<b>72.8</b>	<b>38.6</b>	88.5	<b>69.3</b>	<b>58.7</b>	<b>79.7</b>	<b>83.5</b>	<b>69.9</b>	<b>70.4</b>	<b>74.1</b>	<b>61.0</b>	<b>44.7</b>	<b>41.7</b>	<b>42.0</b>	<b>30.1</b>	<b>34.3</b>
FB15k-237																
GQE	16.3	-	35.0	7.2	5.3	23.3	34.6	16.5	10.7	8.2	5.7	-	-	-	-	-
Q2B	20.1	-	40.6	9.4	6.8	29.5	42.3	21.2	12.6	11.3	7.6	-	-	-	-	-
BetaE	20.9	5.5	39.0	10.9	10.0	28.8	42.5	22.4	12.6	12.4	9.7	5.1	7.9	7.4	3.5	3.4
CQD-CO	21.8	-	<b>46.7</b>	9.5	6.3	31.2	40.6	23.6	16.0	14.5	8.2	-	-	-	-	-
CQD-Beam	22.3	-	<b>46.7</b>	11.6	8.0	31.2	40.6	21.2	18.7	14.6	8.4	-	-	-	-	-
FuzzQE	24.0	7.8	42.8	12.9	10.3	33.3	46.9	26.9	17.8	14.6	10.3	8.5	11.6	7.8	5.2	5.8
ConE	23.4	5.9	41.8	12.8	11.0	32.6	47.3	25.5	14.0	14.5	10.8	5.4	8.6	7.8	4.0	3.6
<b>GNN-QE</b>	<b>26.8</b>	<b>10.2</b>	42.8	<b>14.7</b>	<b>11.8</b>	<b>38.3</b>	<b>54.1</b>	<b>31.1</b>	<b>18.9</b>	<b>16.2</b>	<b>13.4</b>	<b>10.0</b>	<b>16.8</b>	<b>9.3</b>	<b>7.2</b>	<b>7.8</b>

# 目录

---

- 基于图神经网络的知识图谱推理
  - Transductive学习
  - Inductive学习
  - 复杂逻辑查询
- 大模型下的知识图谱审视
  - LLM和知识图谱的关系
  - LLM对知识图谱的影响
  - LLM和知识图谱的未来发展

# 大模型下的知识图谱审视

## 1. LLM和知识图谱的关系

- 从概念的角度

**知识图谱KG** ( Knowledge Graph ) 是对真实世界的概念及其关系的表示，通常以图数据结构的形式表示。它旨在提供一种结构化和有组织的方式来表示和查询有关世界的信息。

**大语言模型LLM** ( Large Language Model ) 是一种语言生成模型，旨在基于它所训练的数据中的模式和关系来生成类人文本。它对所生成信息中的基本概念和关系没有深入的理解。



请问ChatGPT有用到知识图谱吗？



是的，作为一个自然语言处理模型，我可以利用知识图谱来丰富我的回答和提供更准确的信息。知识图谱是一个结构化的知识库，它将实体、概念和关系组织在一起，形成一个图形化的表示。在回答问题或提供信息时，我可以利用知识图谱来查找相关实体和概念，并提供关于它们的详细信息。这有助于提高我的准确性和响应速度。

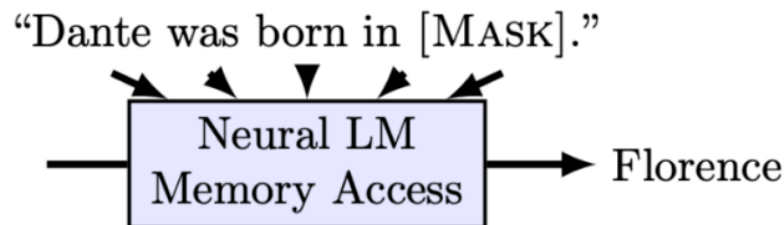
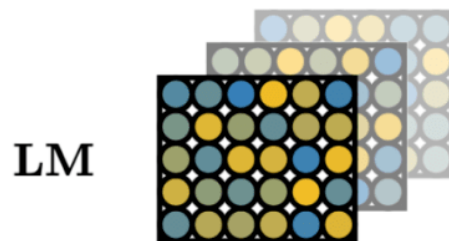
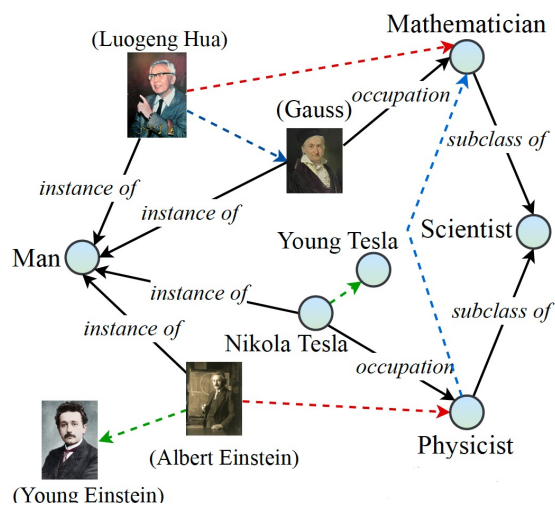
# 大模型下的知识图谱审视

## 1. LLM和知识图谱的关系

- 从知识的角度

知识图谱，**结构化知识库**，知识图谱通过三元组储存知识，知识图谱显式地储存知识。相当于将知识和利用知识的方式解耦。显式地储存知识也有助于归因溯源，提高模型行为的可解释性。

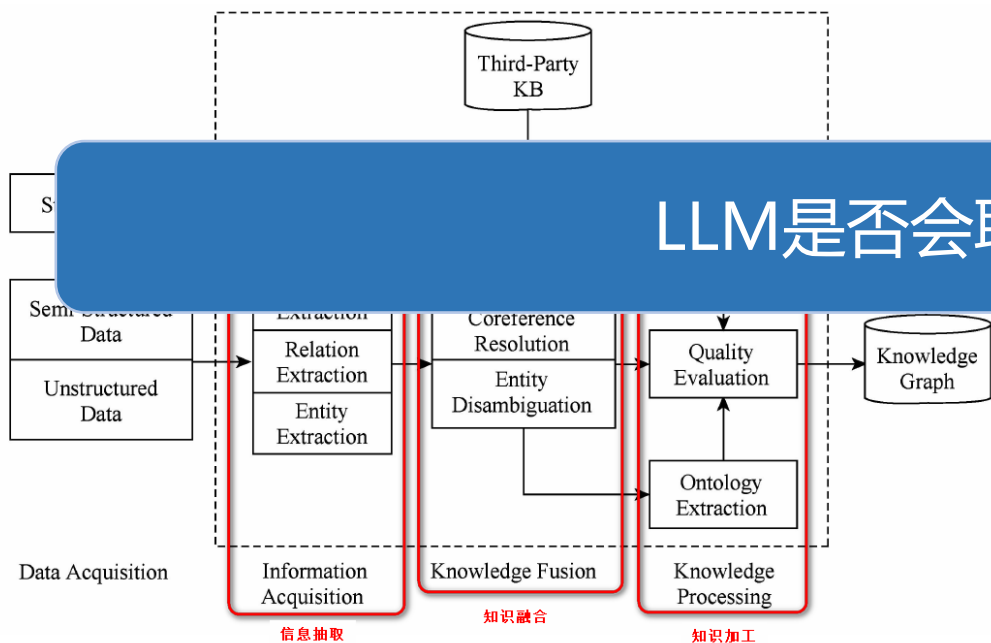
大模型，**参数化知识库**，大模型通过网络参数储存知，大模型隐式的储存知识。语言模型除了学习到语言学知识外，还学到了大量世界性知识（或称事实知识）。大模型决策过程难归因、解释、溯源。



# 大模型下的知识图谱审视

## 2. LLM对知识图谱的影响

以ChatGPT为代表的大语言模型，在与人直接交互的问答场景表现更为优异。而知识图谱作为此前搜索领域主要应用的技术，必然受到了巨大的挑战，比如知识抽取，知识融合，知识推理与计算，以及上层的问答、搜索、推荐其实都会受到影响。



LLM是否会取代知识图谱？

输入文本: "The company Apple is located in Cupertino. Its CEO Tim Cook is known for his leadership skills.", 请识别出文本中出现的实体, 并进行实体消歧。

Tim Cook	人物
----------	----

对于实体消歧, 可以使用如下的方式进行:

- "Apple"可以指代多个不同的实体, 包括: 苹果公司 (Apple Inc.)、苹果电脑公司 (Apple Computer Inc.) 等。根据上下文, "Apple"在这里指的很可能是苹果公司, 因此可以将其消歧为"Apple Inc."。
- "Cupertino"可以指代多个不同的地点, 但在这个例子中, 由于上下文中提到了"Apple", 可以通过知识库等方式将其消歧为"Cupertino, California"。
- "Tim Cook"可以指代多个不同的人物, 但由于上下文中提到了"Apple"的CEO, 可以将其消歧为"Tim Cook, CEO of Apple Inc."。

# 大模型下的知识图谱审视

## 3. LLM和知识图谱的未来发展

- 角度1: 大模型在知识图谱上的表现

通用知识图谱中的知识获取、知识融合等模块很可能被大模型替代，而**基于知识图谱的推理**则是目前能够与大模型共存的。此外，**垂直领域的知识图谱**仍然有其发展空间，比如面向金融、医药等特定KG、面向智慧城市和智慧交通的时空KG。



- 现阶段大模型的局限性

缺乏对其生成文本的上下文和背景知识的理解  
无法在不同概念之间进行推理和建立逻辑联系  
缺乏该领域专家的深度和细微差别

- 融入知识图谱的优势

访问丰富的背景信息和上下文，更深入理解和上下文的文本  
以结构化方式组织概念，提高模型的推理能力  
关联不同人物的画像特征，提供特性化的推荐和响应

# 大模型下的知识图谱审视

## 3. LLM和知识图谱的未来发展

- 角度2: 知识存储方式的选择

参数化存储和形式化存储在准确性和数据量两方面的比较。现在来看模型存储的数据量可能是超过了常见的知识图谱的，另外知识图谱的准确性在很多细分领域是要比模型高的。当参数化存储的知识的准确性和数据量超过形式化存储的时候，形式化存储方式就显得落后了。

符号主义和连接主义各自在不同的历史时期都扮演着重要的角色，它们提供了以符号为主和以连接为主的两种不同认知模型，同时它们也都具有各自的优点和局限。未来研究的重点将更多地放在如何将这两种范式结合起来，以实现更好的认知和智能行为。连接主义盛行，符号主义依旧存在。所以知识图谱不会消失，只是在它擅长的领域继续存在。



# 总结

---

- 基于图神经网络的知识图谱推理
  - Transductive学习
  - Inductive学习
  - 复杂逻辑查询
- 大模型下的知识图谱审视
  - LLM和知识图谱的关系
  - LLM对知识图谱的影响
  - LLM和知识图谱的未来发展

# 后续讨论班介绍

---

- 第一周：图神经网络在金融领域的应用-石逢钊
- 第二周：图神经网络赋能的知识图谱研究与应用-刘瑜
- 第三周：图网络在生物化学领域的应用-周玉晨
- 第四周：图网络在社会网络中的应用-宋传承
- 第五周：图神经网络在推荐系统中的应用-吴咏萱