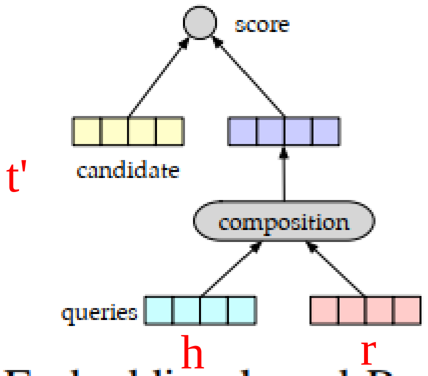
基于规则的知识图谱补全

凌静

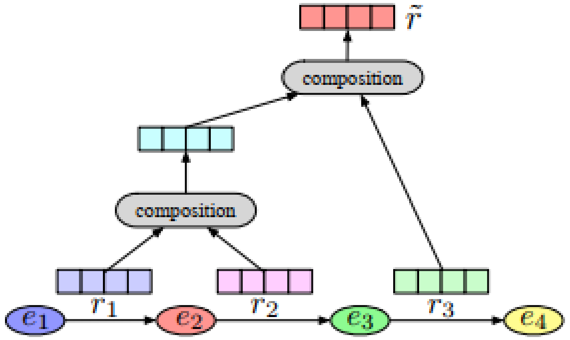
2020/11/26

KG补全方法&相关任务

- **Embedding-based Models**
- **Rule-based Reasoning**
- Relation Path Reasoning
- RL-based Path Finding



(a) Embedding-based Ranking.



(b) Relation paths [57].

- Link Prediction
    - 社交网络熟人/相似用户推荐；生物领域相互作用蛋白质预测
- Triple Classification
    - 图谱精化
- Rule Mining
- Meta Relational learning

《A Survey on Knowledge Graphs: Representation, Acquisition and Applications》

方向

- 基于统计的规则挖掘方法
  - AMIE3
- 在嵌入中注入逻辑规则以提高推理能力（联合学习或迭代训练）
  - IterE
- 神经模型和符号模型结合，以端到端方式进行基于规则的推理
  - ExpressGNN

Fast and Exact Rule Mining with AMIE 3

Jonathan Lajus, Luis GalÃ ¿arraga, and Fabian M. Suchanek.
In ESWC 2020.

概念&AMIE流程

rule: α *head* $\leftarrow$ *body* $\quad (\mathrm{X}, \mathit{hasMother}, \mathrm{Y}) \leftarrow (\mathrm{X}, \mathit{hasParent}, \mathrm{Y}), (\mathrm{Y}, \mathit{gender}, \mathrm{Female})$

grounding: $(\mathrm{Bob}, \mathit{hasMother}, \mathrm{Ann}) \leftarrow (\mathrm{Bob}, \mathit{hasParent}, \mathrm{Ann}), (\mathrm{Ann}, \mathit{gender}, \mathrm{Female})$

规则挖掘应用： fact prediction/data and ontology alignment/error detection

AMIE3：提出一系列剪枝&加速策略,可以高效率地精确计算置信度，无需采样、提前结束或近似计算

- Integer-based in memory database

将内存中的数据库迁移到一个基于整数的系统中，其中实体和关系被映射到一个整数空间，并由原始数据类型int表示。

$$support(R) = |\{p : (\mathcal{K} \wedge R \models p) \ \wedge \ p \in \mathcal{K}\}|$$

$$confidence(R) = \frac{support(R)}{support(R) + |\{p : (\mathcal{K} \wedge R \models p) \wedge p \in cex(R)\}|}$$

$$pca\text{-}conf(\boldsymbol{B} \Rightarrow r(x,y)) = \frac{support(\boldsymbol{B} \Rightarrow r(x,y))}{|\{(x,y) : \exists y' : \boldsymbol{B} \wedge r(x,y')\}|}.$$

$$hc(\boldsymbol{B} \Rightarrow r(x,y)) = \frac{support(\boldsymbol{B} \Rightarrow r(x,y))}{|\{(x,y) : r(x,y) \in \mathcal{K}\}|}$$

Algorithm 1: AMIE

Input: a KB: $\mathcal{K}$, maximum rule length: l, head coverage threshold: $minHC$, confidence threshold: $minC$

Output: set of Horn rules: $rules$

```
q = [⊤ ⇒ r_1(x, y), ⊤ ⇒ r_2(x, y) ... ⊤ ⇒ r_m(x, y)]
rules = ⟨⟩
while |q| > 0 do
    R = q.dequeue()
    if closed(R) ∧ pca-conf(R) ≥ minC ∧ betterThanParents(R, rules) then
        rules.add(r)
    if length(R) < l ∧ pca-conf(R_c) < 1.0 then
        for each rule R_c ∈ refine(R) do
            if hc(R_c) ≥ minHC ∧ R_c ∉ q then
                q.enqueue(r_c)
return rules
```

实验结果

- 500Go of RAM
- RuDiK&AMIE3

Table 6: Performances and output of Ontological Pathfinding (OP), RuDiK and AMIE 3. *: rules with support $\geq$ 100 and CWA confidence $\geq$ 0.1.

Dataset	System	Rules	Runtime
Yago2s	OP (their candidates)	429 (52*)	18min 50s
	OP (our candidates)	1 348 (96*)	3h 20min
	RuDiK	17	37min 30s
	AMIE 3	97	**1min 50s**
	AMIE 3 (support=1)	1 596	7min 6s
DBpedia 3.8	OP (our candidates)	7 714 (220*)	> 45h
	RuDiK	650	12h 10min
	RuDiK + types	650	11h 52min
	AMIE 3	5 084	**7min 52s**
	AMIE 3 (support=1)	132 958	32min 57s
Wikidata 2019	OP (our candidates)	15 999 (326*)	> 48h
	RuDiK	1 145	23h
	AMIE 3	8 662	**16h 43min**

Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning

W. Zhang, B. Paudel, L. Wang, J. Chen, H. Zhu, W. Zhang, A. Bernstein, and H. Chen,
in WWW, 2019, pp. 2366–2377.

嵌入方法&规则方法

- 嵌入学习：稀疏实体的编码能力较差
- 规则学习：效率问题（搜索空间与关系的数量成指数关系）

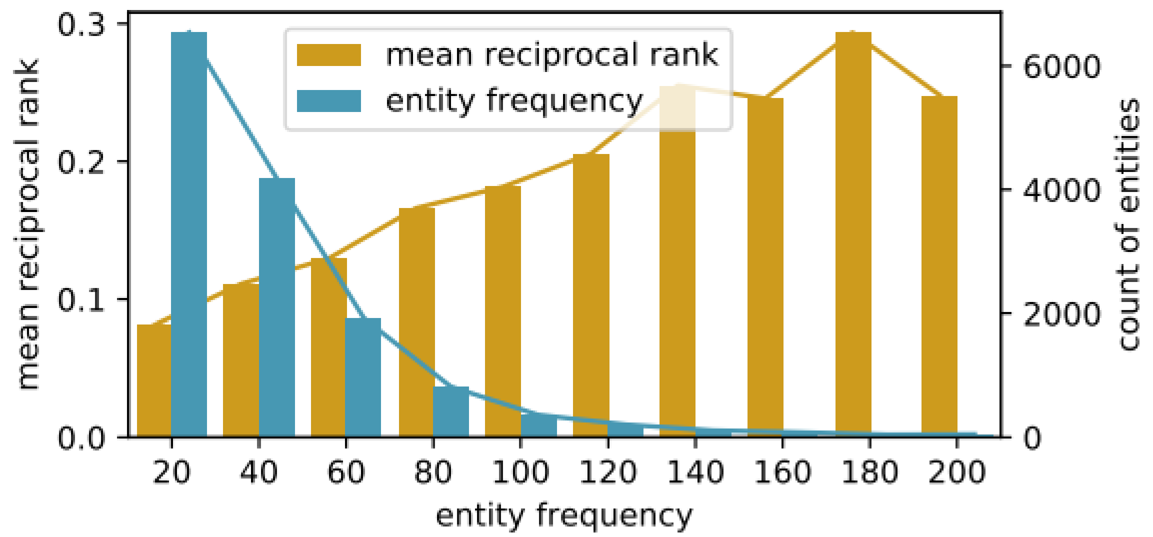


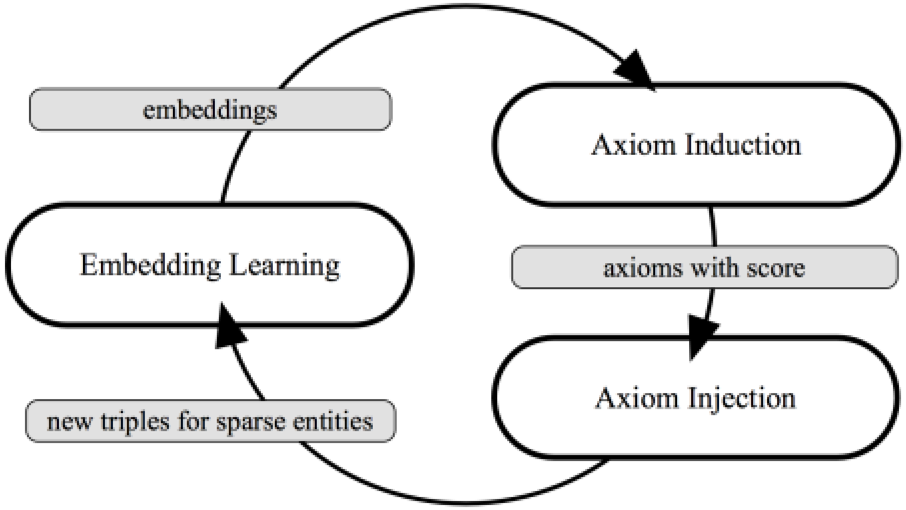
Figure 1: The numbers and mean reciprocal rank of different frequency entities based on ANALOGY results on FB15k-237.

研究问题&方法

问题:嵌入和规则是否可以同时学习，使它们互补?

三个部分:

1. 嵌入学习
2. 公理归纳
3. 公理注入



嵌入学习

- Translation-based assumption
  - 实体->向量；关系->向量； $v_s + v_h = v_o$
- **linear map assumption √**
  - 实体->向量；关系->矩阵； $v_s M_r = v_o$

$$\min L_{embedding} = \frac{1}{n} \sum_{((s,r,o),l_{sro}) \in \mathcal{I}} [-l_{sro} \log(\phi(s,r,o)) - (1 - l_{sro}) \log(1 - \phi(s,r,o))]$$

$$\phi(s,r,o) = sim(\mathbf{v}_s \mathbf{M}_r, \mathbf{v}_o) = \sigma(\mathbf{v}_s^\top \mathbf{M}_r \mathbf{v}_o)$$

表达的是必要条件，满足等式不是一定满足某个性质。
比如$M_{r1}=M_{r2}$表达包含（可以是等价也可以是包含），
已知(a,r1,b)成立,预测(a,r2,b), r1可能是兄弟关系也可能是
家人关系，如果r1是家人的话，预测就错误了

公理归纳

OWL2本体语言

关系的性质

Object Property Axioms	Rule Form	According to Linear Map Assumption	Rule Conclusion
ReflexiveOP(r)	(x, r, x)	$\mathbf{v}_x\mathbf{M}_r = \mathbf{v}_x$	$\mathbf{M}_r = \mathbf{I}$
SymmetricOP(r)	$(y, r, x) \leftarrow (x, r, y)$ r:兄弟	$\mathbf{v}_y\mathbf{M}_r = \mathbf{v}_x;\ \mathbf{v}_x\mathbf{M}_r = \mathbf{v}_y$	$\mathbf{M}_r\mathbf{M}_r = \mathbf{I}$
TransitiveOP(r)	$(x, r, z) \leftarrow (x, r, y), (y, r, z)$	$\mathbf{v}_x\mathbf{M}_r = \mathbf{v}_z;\ \mathbf{v}_x\mathbf{M}_r = \mathbf{v}_y,\ \mathbf{v}_y\mathbf{M}_r = \mathbf{v}_z,$	$\mathbf{M}_r\mathbf{M}_r = \mathbf{M}_r$
EquivalentOP(r_1, r_2)	$(x, r_2, y) \leftarrow (x, r_1, y)$ r1:配偶 r2:夫妻	$\mathbf{v}_x\mathbf{M}_{r_2} = \mathbf{v}_y,\ \mathbf{v}_x\mathbf{M}_{r_1} = \mathbf{v}_y$	$\mathbf{M}_{r_1} = \mathbf{M}_{r_2}$
subOP(r_1, r_2)	$(x, r_2, y) \leftarrow (x, r_1, y)$ r1:兄弟 r2:家人（继承）	$\mathbf{v}_x\mathbf{M}_{r_2} = \mathbf{v}_y,\ \mathbf{v}_x\mathbf{M}_{r_1} = \mathbf{v}_y$	$\mathbf{M}_{r_1} = \mathbf{M}_{r_2}$
inverseOP(r_1, r_2)	$(x, r_1, y) \leftarrow (y, r_2, x)$ r1:丈夫 r2:妻子	$\mathbf{v}_x\mathbf{M}_{r_1} = \mathbf{v}_y,\ \mathbf{v}_y\mathbf{M}_{r_2} = \mathbf{v}_x$	$\mathbf{M}_{r_1}\mathbf{M}_{r_2} = \mathbf{I}$
subOP(OPChain(r_1, r_2), r)	$(y_0, r, y_2) \leftarrow (y_0, r_1, y_1), (y_1, r_2, y_2)$	$\mathbf{v}_{y_0}\mathbf{M}_r = \mathbf{v}_{y_2},\ \mathbf{v}_{y_0}\mathbf{M}_{r_1} = \mathbf{v}_{y_1},\ \mathbf{v}_{y_1}\mathbf{M}_{r_2} = \mathbf{v}_{y_2}$	$\mathbf{M}_{r_1}\mathbf{M}_{r_2} = \mathbf{M}_r$

- step1：枚举r代入规则
- step2：对于不完整规则，随机选k个r的三元组,枚举其中的每个头尾实体的相关关系作为r' or r''
- step3：对每条规则计算其support,大于一的加入公理池中

$$\min k \quad s.t.\ 1 - \frac{C^k_{(1-p)N}}{C^k_N} > t$$

- 公理置信度打分

$$s_a(F) = \|\mathbf{M}_1^a - \mathbf{M}_2^a\|_F \qquad s_a = \frac{s_{max}(t) - s_a(F)}{s_{max}(t) - s_{min}(t)}$$

Frobenius范数

公理注入

- 过滤与稀疏实体无关的三元组
- 为每条规则设置了预测三元组的**数量上限**

$$sparsity(e) = 1 - \frac{freq(e) - freq_{min}}{freq_{max} - freq_{min}}$$

$$\pi(a \wedge b) = \pi(a) \cdot \pi(b)$$

$$\pi(a \vee b) = \pi(a) + \pi(b) - \pi(a) \cdot \pi(b)$$

$$\pi(\neg a) = 1 - \pi(a)$$

$$\pi(a \Rightarrow b) = \pi(\neg a \vee b)$$

$$g:\ (s_1, r_1, o_1) \wedge (s_2, r_2, o_2) \Rightarrow (s^a, r^a, o^a):$$

$$\pi(g) = 1 - \pi(s_1, r_1, o_1)\pi(s_2, r_2, o_2) + \pi(s_1, r_1, o_1)\pi(s_2, r_2, o_2)\pi(s^a, r^a, o^a)$$

$$\pi(s^a, r^a, o^a) = \frac{\pi(g) - 1 + \pi(s_1, r_1, o_1)\pi(s_2, r_2, o_2)}{\pi(s_1, r_1, o_1)\pi(s_2, r_2, o_2)}$$

//认为知识库中的三元组绝对正确

$$\pi(s^a, r^a, o^a) = s_a$$

数据集&规则评测

Table 3: Statistics of datasets.

Dataset	#E	#R	#Train	#Valid	#Test
WN18-sparse	40,943	18	141,442	3624(72.48%)	3590(71.8%)
WN18RR-sparse	40943	11	86835	1609(53.03)	1661(52.9%)
FB15k-sparse	14,951	1,345	483,142	18544(37.08%)	22013(37.26%)
FB15k-237-sparse	14541	237	272115	10671(60.8%)	12454(60.85%)

验证集和测试集仅保留与稀疏实体相关的三元组

HQr:HC>0.7的规则

Table 6: Rule evaluation results.

	WN18-sparse			WN18RR-sparse			FB15k-sparse			FB15k-237-sparse		
	time	HQr	%	time	HQr	%	time	HQr	%	time	HQr	%
AMIE+	4.98s	16	11.4%	3s	2	5.71%	428s	1820	4.4%	66s	470	1.9%
IterE	**1.63s**	**20**	**20.2%**	**0.75s**	**6**	**19.3%**	**26.49s**	**11375**	**17.6%**	**4.72s**	**653**	**11.8%**

$$HC(rul) = \frac{\#(e,e') : support(rul) \wedge head(rul)(e,e')}{\#(e,e') : head(rul)(e,e')}$$

结果

	WN18-sparse					FB15k-sparse				
	MRR (filter)	MRR (raw)	Hit@1 (filter)	Hit@3 (filter)	Hit10 (filter)	MRR (filter)	MRR (raw)	Hit@1 (filter)	Hit@3 (filter)	Hit10 (filter)
TransE[3]	41.8	33.5	10.2	71.1	84.7	39.8	25.5	25.8	48.6	64.5
DistMult[47]	73.8	55.8	59.3	87.5	93.1	60.0	32.4	**61.8**	65.1	75.9
ComplEx[35]	91.1	67.7	89.0	93.3	94.4	61.6	32.7	54.0	65.7	76.1
ANALOGY[23]	**91.3**	67.5	89.0	93.4	94.4	62.0	33.1	54.3	66.1	76.3
IterE(ANALOGY)	90.1	67.5	87.0	93.1	**94.8**	61.3	35.9	52.9	66.2	76.7
IterE(ANALOGY) + **axioms**	**91.3**	**78.9**	**89.1**	**93.5**	**94.8**	**62.8**	**38.8**	55.1	**67.3**	**77.1**

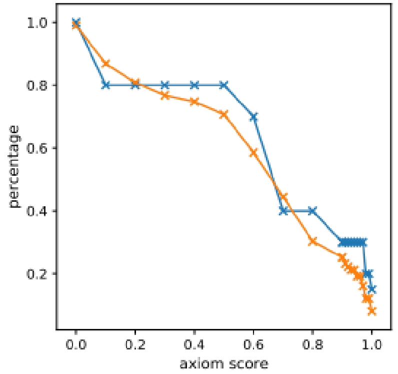
	WN18RR-sparse					FB15k-237-sparse				
	MRR (filter)	MRR (raw)	Hit@1 (filter)	Hit@3 (filter)	Hit10 (filter)	MRR (filter)	MRR (raw)	Hit@1 (filter)	Hit@3 (filter)	Hit10 (filter)
TransE[3]	14.6	12.4	3.4	24.7	28.8	23.8	15.6	16.4	26.1	38.5
DistMult[47]	25.5	20.8	23.8	26.0	22.5	20.4	12.9	12.8	22.6	36.2
ComplEx[35]	25.9	21.4	24.6	26.2	28.6	19.7	13.3	12.0	21.7	35.4
ANALOGY[23]	19.8	13.3	24.6	27.5	28.7	19.8	13.9	12.3	21.4	34.9
IterE(ANALOGY)	27.2	22.7	25.0	**28.1**	**31.4**	20.7	14.0	13.1	22.8	36.2
IterE(ANALOGY) + **axioms**	**27.4**	**25.7**	**25.4**	**28.1**	**31.4**	**24.7**	**18.6**	**17.9**	**26.2**	**39.2**

$$\mathrm{MRR} = \frac{1}{Q}\sum_{Q}\frac{1}{rank_Q}$$

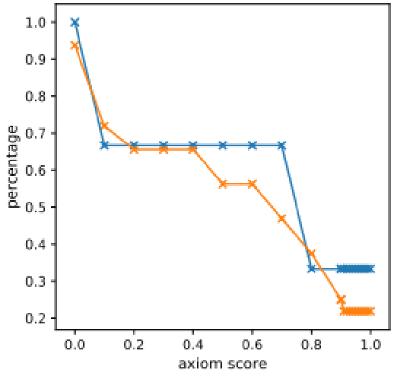
$$rank(s,r,o) = \frac{1}{2}\left(rank_s(?,r,o) + rank_o(s,r,?)\right)$$

Hit@n：正确三元组排名在前n的占比

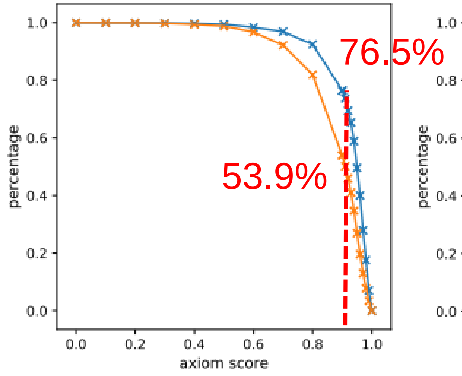
结果



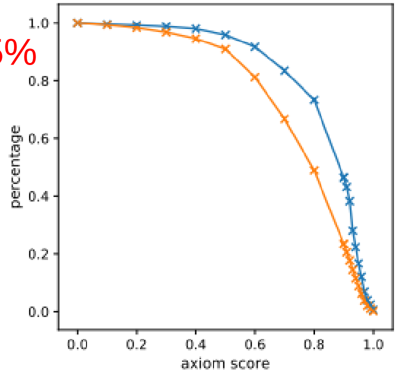
(a) **WN18-sparse**



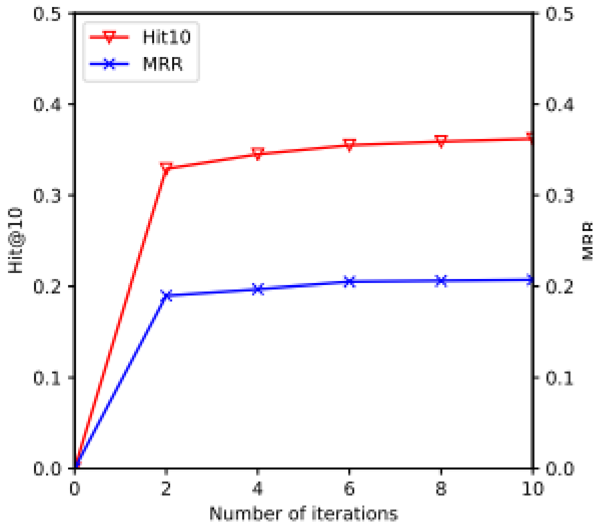
(b) **WN18RR-sparse**



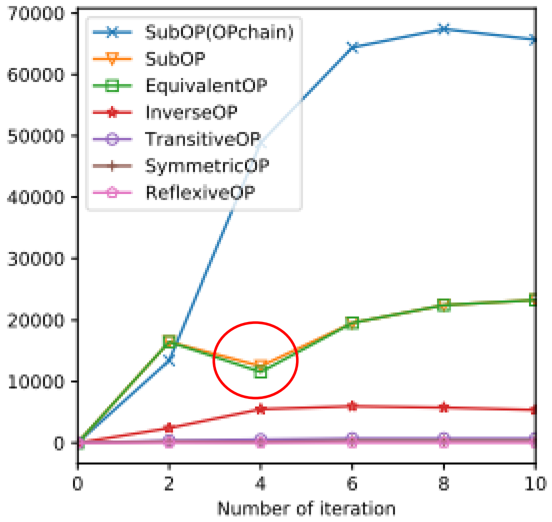
(c) **FB15k-sparse**



(d) **FB15k-237-sparse**



(a)



(b)

意味着在前一次迭代中获得高分的公理在下一次迭代中可能得到较低的分数。（嵌入影响规则）

x：公理分数阈值　　　　黄线：公理占比

y：满足阈值的公理/总数　　蓝线：高质量公理占比（HC>0.7）

EFFICIENT PROBABILISTIC LOGIC REASONING WITH GRAPH NEURAL NETWORKS

Y. Zhang, X. Chen, Y. Yang, A. Ramamurthy, B. Li, Y. Qi, and L. Song,
in ICLR, 2020, pp. 1–20.

ExpressGNN

- explore the combination of MLNs and GNNs, and use graph neural networks for variational inference in MLN
- 马尔可夫逻辑网络(Markov logic network，MLNs)结合逻辑规则和概率图形模型，给出一件事为真的概率而非直接判断它的对错
- 推理

Table 3: Performance on FB15K-237 with varied training set size.

Model	MRR					Hits@10				
	0%	5%	10%	20%	100%	0%	5%	10%	20%	100%
MLN	-	-	-	-	0.10	-	-	-	-	16.0
NTN	0.09	0.10	0.10	0.11	0.13	17.9	19.3	19.1	19.6	23.9
Neural LP	0.01	0.13	0.15	0.16	0.24	1.5	23.2	24.7	26.4	36.2
DistMult	0.23	0.24	0.24	0.24	0.31	40.0	40.4	40.7	41.4	48.5
ComplEx	0.24	0.24	0.24	0.25	0.32	41.1	41.3	41.9	42.5	51.1
TransE	0.24	0.25	0.25	0.25	0.33	42.7	43.1	43.4	43.9	52.7
RotatE	0.25	0.25	0.25	0.26	0.34	42.6	43.0	43.5	44.1	53.1
pLogicNet	-	-	-	-	0.33	-	-	-	-	52.8
ExpressGNN-E	**0.42**	**0.42**	0.42	0.44	0.45	53.1	53.1	53.3	55.2	57.3
ExpressGNN-EM	**0.42**	**0.42**	**0.43**	**0.45**	**0.49**	**53.8**	**54.6**	**55.3**	**55.6**	**60.8**

Table 4: Zero-shot learning performance on FB15K-237.

Model	MRR	Hits@10
NTN	0.001	0.0
Neural LP	0.010	2.7
DistMult	0.004	0.8
ComplEx	0.013	2.2
TransE	0.003	0.5
RotatE	0.006	1.5
ExpressGNN-E	0.181	29.3
ExpressGNN-EM	**0.185**	**29.6**

CoDEx: A Comprehensive Knowledge Graph Completion Benchmark

Tara Safavi, Danai Koutra,
in EMNLP 2020.

CoDEx

构建过程:

1.人工定义13个领域的实体和关系类型种子，使用snowball sampling收集初始三元组(Wikidata)

2.根据实体的度进行筛选（最小度k分别为15/10/5），得到三个大小不同的数据集

3.为避免train/test泄露，枚举所有关系对r和r'，(h,r,t)和（t,r',h)重叠超过50%的关系会被删除

symmetry?

附加信息:

- 实体类型:来自Wikidata的 “instance of” 和 “subclass of” 关系
- Wikidata 标签和描述（多语言）
- 实体和实体类型的多语言介绍
- 阿拉伯语, 德语, 英语, 西班牙语, 俄罗斯语, 汉语

	$\|E\|$	$\|R\|$	Triples $E \times R \times E$					Multilingual coverage					
			Train (+)	Valid (+)	Test (+)	Valid (-)	Test (-)	ar	de	en	es	ru	zh
CoDEx-S	2,034	42	32,888	1827	1828	1827	1828	77.38	91.87	96.38	91.55	89.17	79.36
CoDEx-M	17,050	51	185,584	10,310	10,311	10,310	10,311	75.80	95.20	96.95	87.91	81.88	69.63
CoDEx-L	77,951	69	551,193	30,622	30,622	-	-	67.47	90.84	92.40	81.30	71.12	61.06

Hard Negatives

- 使用模型预测KG中三元组的尾实体，选满足以下条件的三元组作为候选实体:
  - rank top10
  - 不在原KG中
  - 预测的尾实体和原三元组尾实体的实体类型相同
- 人工标注(仅CoDEx-S和CoDEx-M)

RESCAL
TransE
ComplEx
ConvE
TuckER

Table 3: Selected examples of hard negatives in CoDEx with explanations.

Negative	Explanation
(*FrÃľdÃľric Chopin*, *occupation*, *conductor*)	Chopin was a pianist and a composer, not a conductor.
(*Georgia*, *diplomatic relation*, *Russia*)	Georgia and Russia broke diplomatic relations in 2008.
(*Lesotho*, *official language*, *American English*)	English, not American English, is an official language of Lesotho.
(*Senegal*, *part of*, *Middle East*)	Senegal is part of West Africa.
(*Simone de Beauvoir*, *field of work*, *astronomy*)	Simone de Beauvoir's field of work was primarily philosophy.
(*Vatican City*, *member of*, *UNESCO*)	Vatican City is a UNESCO World Heritage Site but not a member state.

关系模式分析及模型评估&负样本的应用

	CoDEx-S	CoDEx-M	CoDEx-L
Symmetry	17.46%	4.01%	3.29%
Composition	10.09%	16.55%	31.84%

if X was founded by Y, and Y’ s country of citizenship is Z, then the country [i.e., of origin] of X is Z

founded(X,Y) && country(Y,Z) => country(X,Z)

	CoDEx-S			CoDEx-M			CoDEx-L		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
RESCAL	0.404	0.293	0.623	0.317	0.244	0.456	0.304	0.242	0.419
TransE	0.354	0.219	0.634	0.303	0.223	0.454	0.187	0.116	0.317
ComplEx	**0.465**	**0.372**	**0.646**	**0.337**	**0.262**	**0.476**	0.294	0.237	0.400
ConvE	0.444	0.343	0.635	0.318	0.239	0.464	0.303	0.240	0.420
TuckER	0.444	0.339	0.638	0.328	0.259	0.458	**0.309**	**0.244**	**0.430**

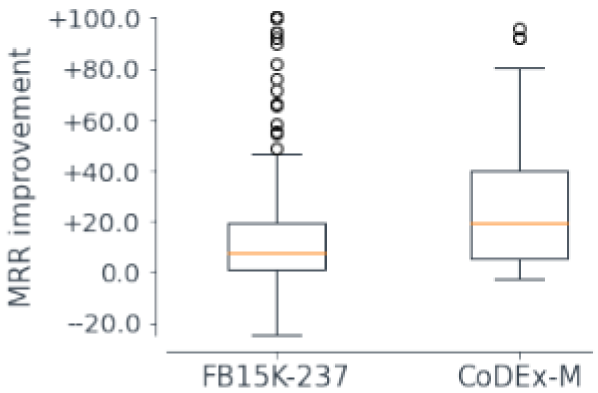
一些模型在CoDEx数据集上的效果

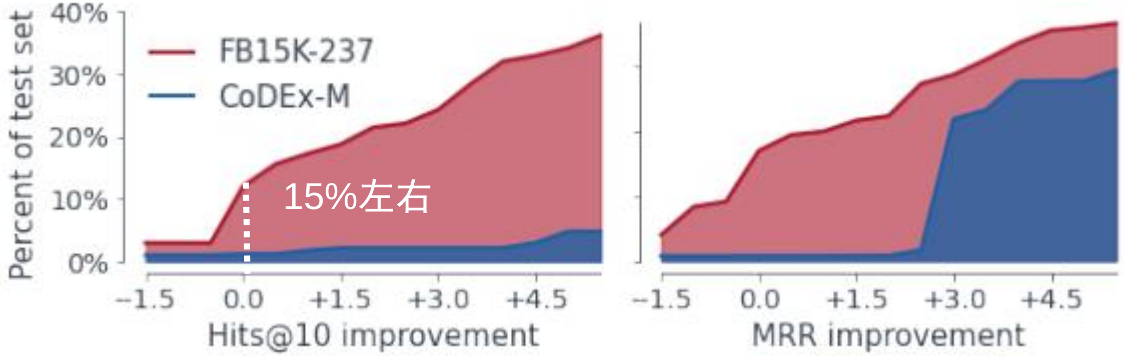
	CoDEx-S						CoDEx-M					
	Uniform		Relative freq.		Hard neg.		Uniform		Relative freq.		Hard neg.	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
RESCAL	0.972	0.972	0.916	0.920	**0.843**	**0.852**	0.977	0.976	0.921	0.922	0.818	0.815
TransE	0.974	0.974	0.919	0.923	0.829	0.837	**0.986**	**0.986**	0.932	0.933	0.797	0.803
ComplEx	**0.975**	**0.975**	**0.927**	**0.930**	0.836	0.846	0.984	0.984	0.930	0.933	0.824	0.818
ConvE	0.972	0.972	0.921	0.924	0.841	0.846	0.979	0.979	**0.934**	**0.935**	**0.826**	**0.829**
TuckER	0.973	0.973	0.917	0.920	0.840	0.846	0.977	0.977	0.920	0.922	0.823	0.816

三元组分类任务效果对比

CoDEx vs FB15K-237

	Baseline	Embedding	Improvement
FB15K-237	0.236	0.356	+0.120
CoDEx-M	0.135	0.337	+0.202





baseline:对于(h,r,?)预测，统计所有object位置上出现的实体，计算相对频率进行打分，去掉已经存在于KG中的三元组，计算rank的MRR

图：improvement的累积分布函数
横坐标:improvement value
纵坐标:累积占比

感谢！