

知识推理技术简介

技术管理部 李罗政

2019年12 月27日

内容提要

- 从知识图谱到知识推理
- 知识推理技术概览：
 - 基于规则的推理
 - 基于分布式表示的推理
- 知识推理应用举例和落地展望

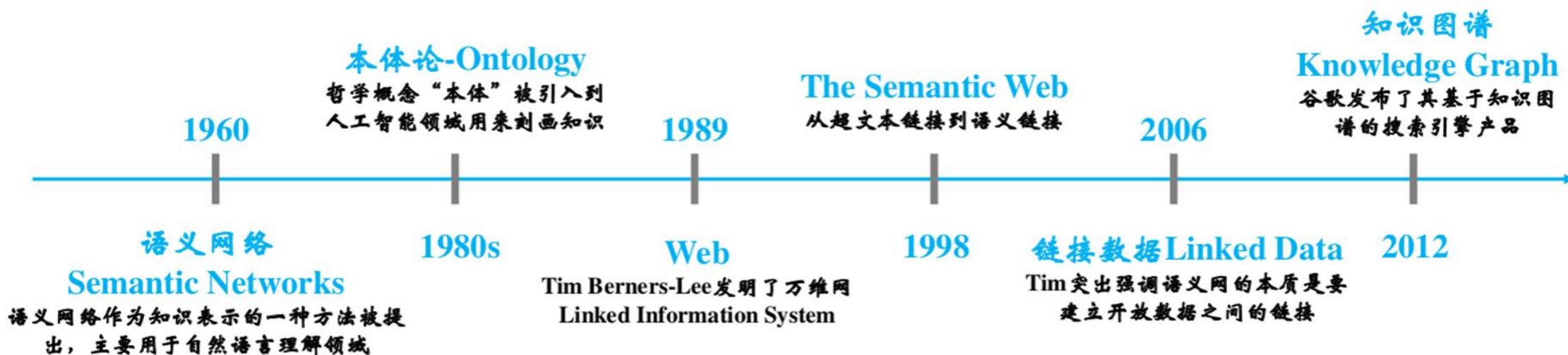
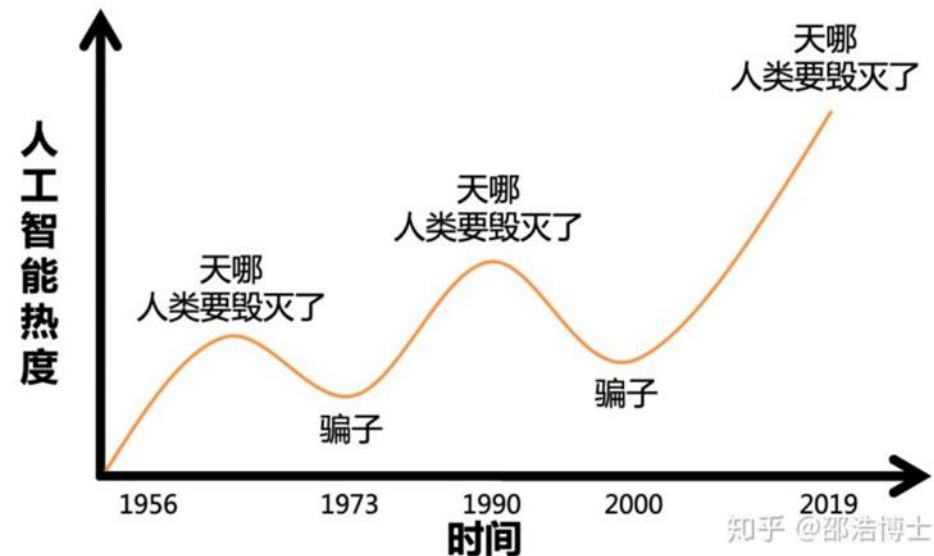
早期的知识工程

人工智能的三条此起彼伏的主线：

机器学习（经验主义）

神经网络（连接主义）

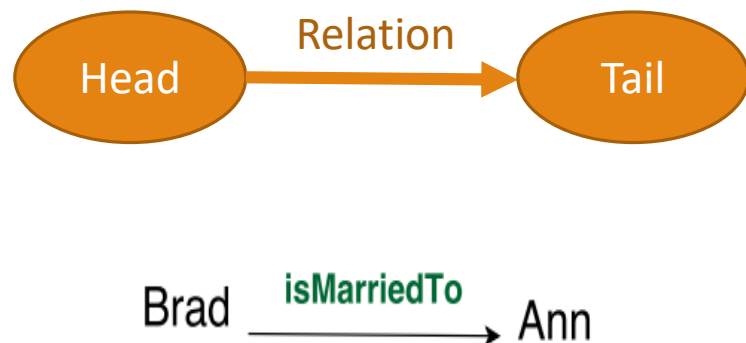
知识工程（符号主义）



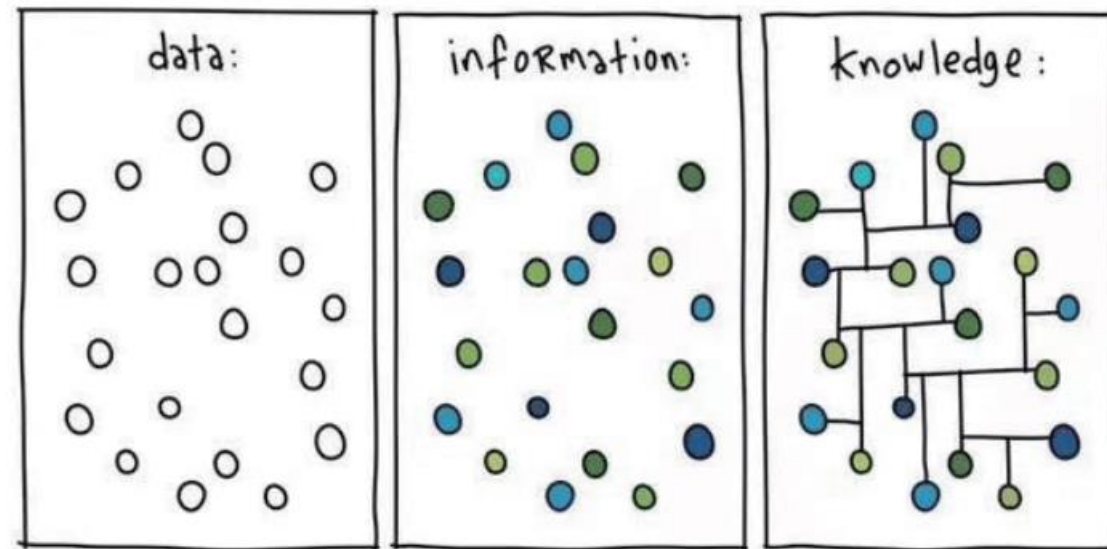
知识图谱

2012年Google提出知识图谱的概念，即用计算机符号来表示人脑中的知识。

逐步发展、简化，形成了今天最常见的三元组形式：



- 三元组形式下，知识的表示方法大为简化，可计算下随之增强。
- 因此，知识图谱不是一种新的知识表示方法，而是知识表示在工业界的大规模知识应用



知识图谱的应用

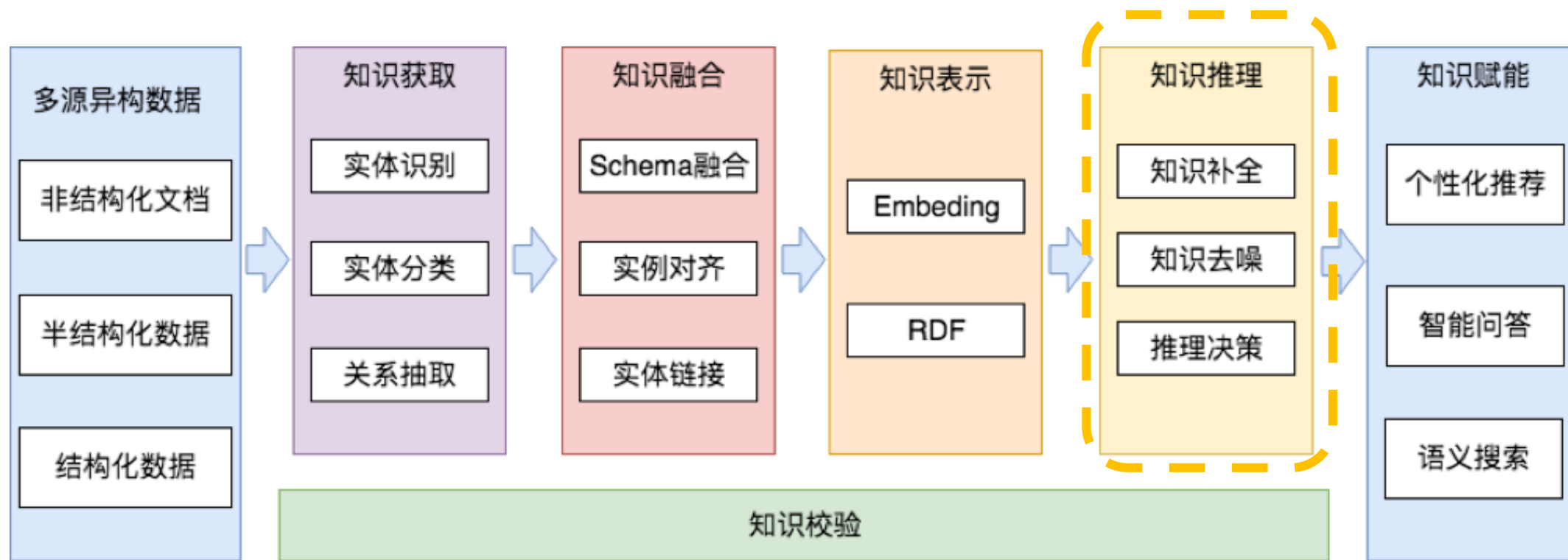
全球各大公司在知识图谱领域的布局



2020年.....

从知识图谱到知识推理

知识图谱应用中的关键技术：



知识推理的应用

- 知识图谱的去噪和补全：

已存在的知识库中，可能存在很多错误的关系连接（噪声）。很多原本正确的关系连接，也可能发生丢失。

- 推理决策：

利用推理技术可以结合知识图谱与现实世界的知识进行关系或属性的预测，挖掘隐含信息

例如，基于人物知识库和事件知识库中的关联分析和规则学习，进行敏感人物判断，可以辅助人工，优化效率

应用场景举例

例如：从wiki百科抽取的小型公众人物知识库，存在大量的关系和属性的缺失，字段和记录数如下：

pageId	predicate	value
1	简介	克利斯登·奈加特（挪威语:KristenNygaard，
1	中文名	克利斯登·奈加特
1	外文名	KristenNygaard
1	性别	男
1	出生地	挪威,奥斯陆
1	出生日期	1926-8-27
1	逝世日期	2002-8-10
1	领域/行业/研究方向	电脑科学
1	维基百科页面URL	https://zh.wikipedia.org/wiki/克利斯登·奈加
1	主要成就/代表作等	图灵奖(2001)
1	主要成就/代表作等	IEEE约翰·冯·诺伊曼奖章(2002)
1	主要成就/代表作等	圣奥拉夫指挥官勋章
1	主要成就/代表作等	诺伯特·维纳社会与专业责任奖

字段	记录数
人物	12406
出生地	11293
国籍	7532
配偶	4627
主要成就/代表作等	4946
领域/行业/研究方向	1952

知识推理

推理：通过各种方法获取新的知识或者结论。

在知识图谱中，推理需要满足以下特点：

- 可满足性(satisfiability)，检验某一实例或本体的可满足性；

一个不可满足的本体
 $\text{Man} \sqcap \text{Women} \sqsubseteq \perp$
 $\text{Man}(\text{Allen})$
 $\text{Women}(\text{Allen})$

- 分类(classification)，计算新的概念包含关系；

$\text{Mother} \sqsubseteq \text{Women}$
 $\text{Women} \sqsubseteq \text{Person}$ \longrightarrow $\text{Mother} \sqsubseteq \text{Person}$

- 实例化(materialization)

计算新的类实例信息

$\text{Mother}(\text{Alice})$
 $\text{Mother} \sqsubseteq \text{Women}$ \longrightarrow $\text{Women}(\text{Alice})$

计算新的二元关系

$\text{has_son}(\text{Alice}, \text{Bob})$
 $\text{has_son} \sqsubseteq \text{has_child}$ \longrightarrow $\text{has_child}(\text{Alice}, \text{Bob})$

知识推理技术

两个大类：

- **基于规则(符号)的推理：**

可以容纳一定程度的语义信息，准确、可解释性强但效率低下，规则构建复杂；

- **基于分布式表示的推理：**

更多利用了知识图谱的结构信息，易于使用和计算，但可解释性差，容易引入噪声

基于规则的知识推理

- 一阶逻辑：只是个体的量化，没有个体集合的量化

例如：张三是学生，李四也是学生”，可以表示为：

$ISSTUDENT(张三) \wedge ISSTUDENT(李四)$

- 描述逻辑：基于本体，

TBox: 术语集，用于描述类和属性之间的相互关系

ABox: 断言集，是关于应用领域的实例断言，表明个体的属性或个体间的相互关系（实例之间的相互关系）

Abox上的推理：Tableaux算法

基于规则的知识推理

规则的公式化表示:

$$\alpha \text{ head} \leftarrow \text{body},$$

其中 α 表示信心指数, 也可以写成以下的形式:

$$\alpha r(x, y) \leftarrow R_1 \wedge R_2 \wedge R_3 \dots \wedge R_n,$$

例如 规则:

$$(X, \text{hasMother}, Y) \leftarrow (X, \text{hasParent}, Y), (Y, \text{gender}, \text{Female})$$

其支持实例 (support) 为:

$$\begin{aligned} &(\text{Bob}, \text{hasMother}, \text{Ann}) \leftarrow \\ &(\text{Bob}, \text{hasParent}, \text{Ann}), (\text{Ann}, \text{gender}, \text{Female}) \end{aligned}$$

规则推理的技术难点

- 给定的知识图谱，数据包含实例的fact knowledge, 而schema knowledge 通常欠缺

E.g.

$(\text{Bob}, \text{hasParent}, \text{Ann}), (\text{Ann}, \text{gender}, \text{Female}) \xrightarrow{?} (\text{X}, \text{hasMother}, \text{Y}) \leftarrow (\text{X}, \text{hasParent}, \text{Y}), (\text{Y}, \text{gender}, \text{Female})$

- 寻找规则的支持实例时，搜索空间巨大:

$$\alpha r(x, y) \leftarrow R_1 \wedge R_2 \wedge R_3 \dots \wedge R_n,$$

搜索空间: $(NE * NE)^n * (NR)^n$, 假如一个知识图谱中有100个实体, 10种关系, 规则的链长为3, 搜索空间约为 10^{15} .

AMIE, AMIE+

- 从知识图谱中挖掘Horn rule
- RDF型的知识图谱
- 使用策略剪枝和近似算法来缩小搜索空间
- 挖掘到的规则形式单一，适用于一阶逻辑

Algorithm 1 Rule Mining

```
1: function AMIE(KB  $\mathcal{K}$ ,  $minHC$ ,  $maxLen$ ,  $minConf$ )
2:    $q = [r_1(x, y), r_2(x, y) \dots r_m(x, y)]$ 
3:    $out = \langle \rangle$ 
4:   while  $\neg q.isEmpty()$  do
5:      $r = q.dequeue()$ 
6:     if  $AcceptedForOutput(r, out, minConf)$  then
7:        $out.add(r)$ 
8:     end if
9:     if  $length(r) < maxLen$  then
10:       $R(r) = Refine(r)$ 
11:      for all rules  $r_c \in R(r)$  do
12:        if  $hc(r_c) \geq minHC \ \& \ r_c \notin q$  then
13:           $q.enqueue(r_c)$ 
14:        end if
15:      end for
16:    end if
17:  end while
18:  return  $out$ 
19: end function
```

Reference: Galárraga L, Teflioudi C, Hose K, et al. Fast rule mining in ontological knowledge bases with AMIE+ [J]. The VLDB Journal—The International Journal on Very Large Data Bases, 2015, 24(6): 707-730.

AMIE, AMIE+

适用范围: **Horn rule**, 在A-Box上的事实集, 发掘规则

(1) 连通性 (connected): 即连接的两个规则之间要 (至少) 共享一个共同实体。比如以下的例子, 就是不连通的:

$$\text{diedIn}(x, y) \Rightarrow \text{wasBornIn}(w, z)$$

(2) 封闭性 (closed): 即每个实体在一个规则中, 至少出现两次。比如下面的例子, 就不是封闭的规则:

$$\text{diedIn}(x, y) \Rightarrow \text{wasBornIn}(x, z)$$

(3) 允许存在反身 (reflexive) 规则, 即在body中包含有head relation, 比如:

$$\text{isMarriedTo}(x, z) \ \& \ \text{hasChild}(z, y) \Rightarrow \text{hasChild}(x, y)$$

AMIE, AMIE+

核心思想：通过对规则进行剪枝，以达到缩小搜索空间的目的

（1）Count Projection Queries：首先通过head-coverage 阈值，筛选出规则

（2）Dangling Atom Operator（添加悬挂边）：

（3）Closed Atom Operator（添加闭合边）：

```
1 SELECT r, COUNT(livesIn(x,y)) WHERE livesIn(x,y) & marriedTo(x,z) & r(X,Y)
2 SUCH THAT COUNT(livesIn(x,y)) >= k
3 with r(X,Y) in {r(x,w), r(z,w), r(w,x), r(w,z)}
```

（4）Instantiated Atom Operator（添加实例边）：

Neural LP

- 利用端到端模型学习一阶逻辑规则
- 把符号运算转化为矩阵运算
- 通过神经网络（RNN）解决搜索空间的问题
- 但在知识图谱较大时，计算量剧增
- 可解释性

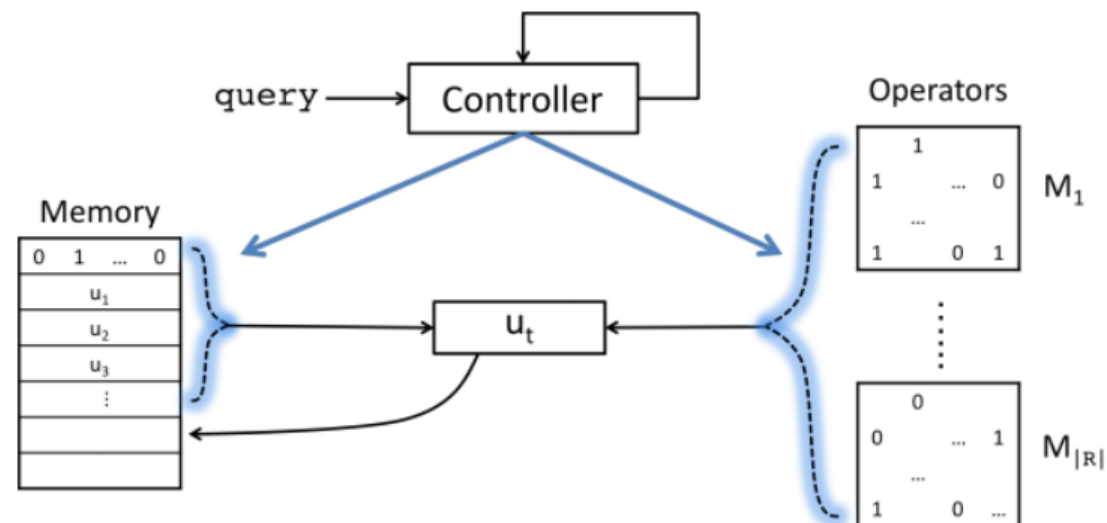


Figure 2: The neural controller system.

Reference: Yang F, Yang Z, Cohen W W. Differentiable learning of logical rules for knowledge base reasoning[C]//Advances in Neural Information Processing Systems. 2017: 2319-2328.

Neural LP

把符号运算转化为矩阵（张量）运算:

每一种关系 R ，对应为矩阵 $|E|*|E|$ 的 $(0,1)$ 矩阵 M_R . 实体 x 用one-hot 的形式进行表示

$$r(y, x) \leftarrow R_1(y, z) \wedge R_2(z, x), \quad \longrightarrow \quad s = M_{R1} \cdot M_{R2} \cdot v_x$$

于是，对于给定的rule head $r(y, x)$, 可以转化为优化问题, 训练出对应的参数 (α, β)

$$s = \sum_l (\alpha_l (\Pi_{k \in \beta_l} M_{R_k} v_x)), \quad \text{score}(y | x) = v_y^T s$$
$$\max_{\{\alpha_l, \beta_l\}} \sum_{\{x, y\}} \text{score}(y | x) = \max_{\{\alpha_l, \beta_l\}} \sum_{\{x, y\}} v_y^T \left(\sum_l (\alpha_l (\Pi_{k \in \beta_l} M_{R_k} v_x)) \right)$$

分布式表示下的知识推理

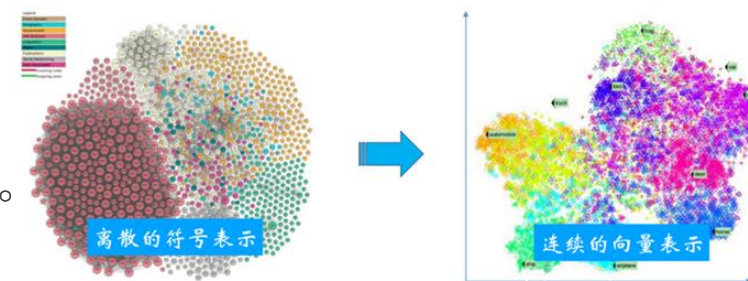
核心思想： 把每个实体（**entity**）和向量（**relation**）表示为高维空间的向量，然后对于每个三元组（ h, r, t ），通过模型计算出每个三元组的评分函数（优化函数） $f(h, r, t)$ 。然后对于知识图谱中不存在的三元组，根据评分的高低，推断是否存在。实体和关系的向量式嵌入有多重方法，常用的包括以下两种：

●Trans系列：

受词向量表示方法的启发，将知识库中实体之间的关系看成是实体间的某种平移，并用向量表示。

●矩阵分解系列：

实体看作向量，关系看作矩阵。知识库三元组构成一个大的张量。



Trans系列

TransE:

$$f(h, r, t) = ||\mathbf{h} + \mathbf{r} - \mathbf{t}||$$

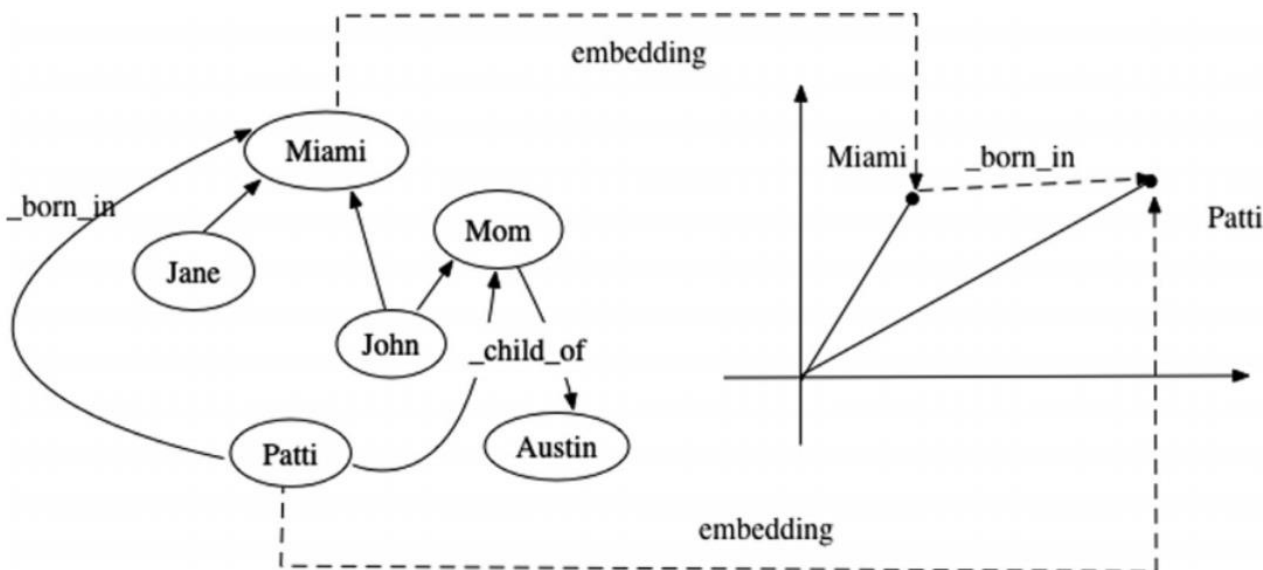


Fig. 2. Translations operating on the low-dimensional embeddings of the entities from knowledge graph.

Algorithm 1 Learning TransE

input Training set $S = \{(h, \ell, t)\}$, entities and rel. sets E and L , margin γ , embeddings dim. k .

- 1: **initialize** $\ell \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each $\ell \in L$
- 2: $\ell \leftarrow \ell / \|\ell\|$ for each $\ell \in L$
- 3: $\mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each entity $e \in E$
- 4: **loop**
- 5: $\mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$ for each entity $e \in E$
- 6: $S_{batch} \leftarrow \text{sample}(S, b)$ // sample a minibatch of size b
- 7: $T_{batch} \leftarrow \emptyset$ // initialize the set of pairs of triplets
- 8: **for** $(h, \ell, t) \in S_{batch}$ **do**
- 9: $(h', \ell, t') \leftarrow \text{sample}(S'_{(h, \ell, t)})$ // sample a corrupted triplet
- 10: $T_{batch} \leftarrow T_{batch} \cup \{((h, \ell, t), (h', \ell, t'))\}$
- 11: **end for**
- 12: Update embeddings w.r.t.
$$\sum_{((h, \ell, t), (h', \ell, t')) \in T_{batch}} \nabla [\gamma + d(\mathbf{h} + \ell, \mathbf{t}) - d(\mathbf{h}' + \ell, \mathbf{t}')]_+$$
- 13: **end loop**

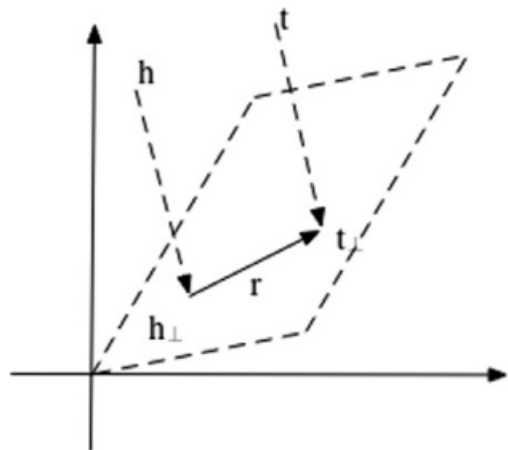
TransE 模型形式简单，但许多关系可能并不适用，例如：

当 r 为夫妻关系时， $r(A, B)$, $r(B, A)$ 均成立，但目标函数难以描述。

$r = 0$?

改进的Trans模型

TransH: 对于每个关系都建立一个超平面，将原有的实体向量投影到超平面，再进行向量计算



Simple illustration of TransH (Wang et al., 2014c).

TransR: 实体和向量分别在不同的向量空间表示。进行向量计算时，先将实体向量进行矩阵投影： $\mathbf{h}_r = \mathbf{M}_r \mathbf{h}$, $\mathbf{t}_r = \mathbf{M}_r \mathbf{t}$

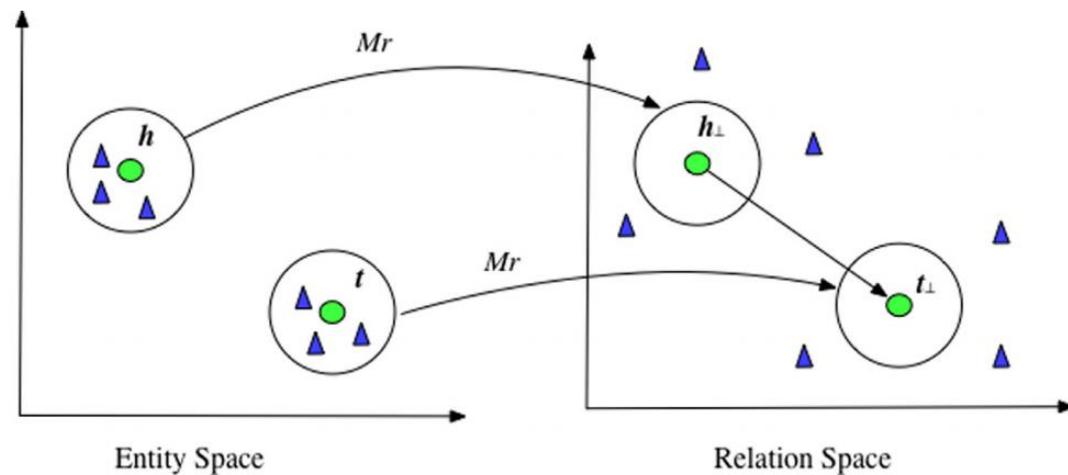


Fig. 5. Simple illustration of TransR (Lin et al., 2015b).

矩阵分解模型: RESCAL

$f(h, r, t) = 1$, if (h, r, t) exist

$$\phi_{linearmap} = sim(\mathbf{v}_s \mathbf{M}_r, \mathbf{v}_o)$$

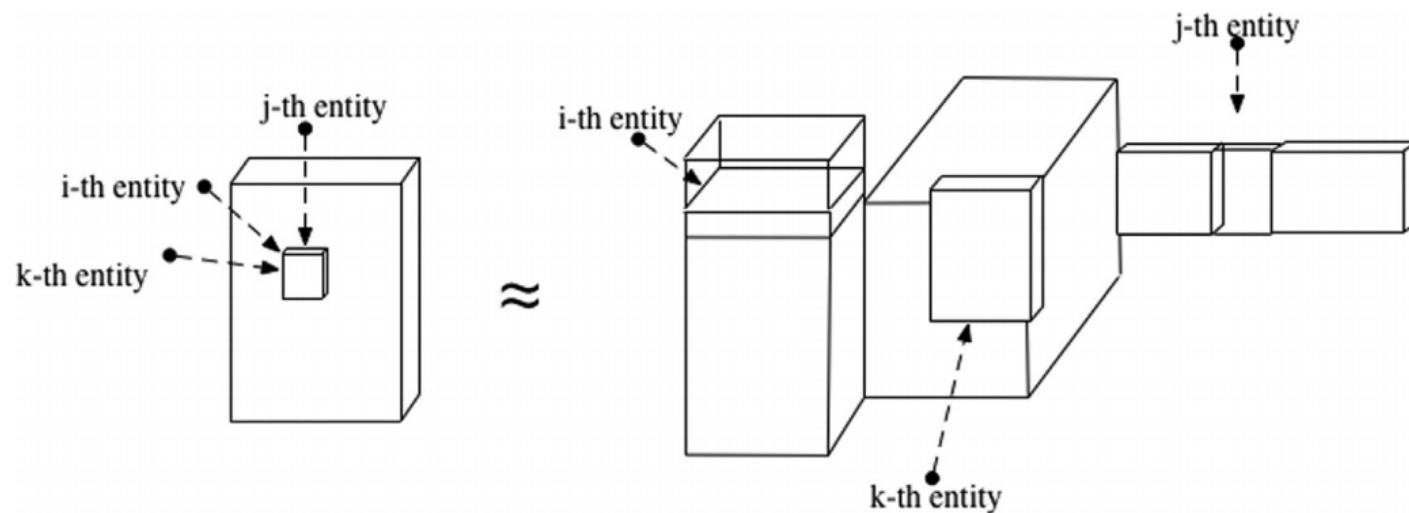


Fig. 3. Simple illustration of RESCAL.

矩阵分解模型: RESCAL

缺点： 相比于TransE， 模型的参数更多，对于大型知识图谱，参数爆炸式增长。

优点： 更多的参数下，关系矩阵可以表示复杂的关系。如下实例，不同形式的矩阵可以表示不同类型的关系。

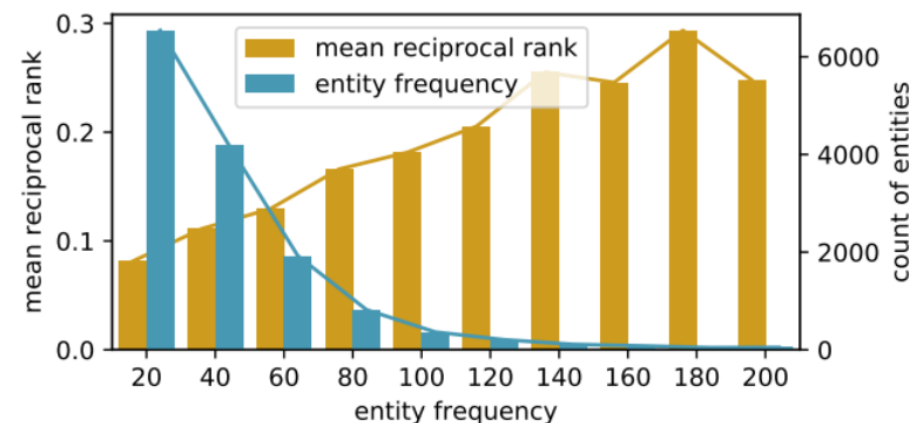
Object Property Axioms	Rule Form	According to Linear Map Assumption	Rule Conclusion
ReflexiveOP(r)	(x, r, x)	$\mathbf{v}_x \mathbf{M}_r = \mathbf{v}_x$	$\mathbf{M}_r = \mathbf{I}$
SymmetricOP(r)	$(y, r, x) \leftarrow (x, r, y)$	$\mathbf{v}_y \mathbf{M}_r = \mathbf{v}_x; \mathbf{v}_x \mathbf{M}_r = \mathbf{v}_y$	$\mathbf{M}_r \mathbf{M}_r = \mathbf{I}$
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)	$(x, r_2, y) \leftarrow (x, r_1, y)$	$\mathbf{v}_x \mathbf{M}_{r_2} = \mathbf{v}_y, \mathbf{v}_x \mathbf{M}_{r_1} = \mathbf{v}_y$	$\mathbf{M}_{r_1} = \mathbf{M}_{r_2}$
subOP(r_1, r_2)	$(x, r_2, y) \leftarrow (x, r_1, y)$	$\mathbf{v}_x \mathbf{M}_{r_2} = \mathbf{v}_y, \mathbf{v}_x \mathbf{M}_{r_1} = \mathbf{v}_y$	$\mathbf{M}_{r_1} = \mathbf{M}_{r_2}$
inverseOP(r_1, r_2)	$(x, r_1, y) \leftarrow (y, r_2, x)$	$\mathbf{v}_x \mathbf{M}_{r_1} = \mathbf{v}_y, \mathbf{v}_y \mathbf{M}_{r_2} = \mathbf{v}_x$	$\mathbf{M}_{r_1} \mathbf{M}_{r_2} = \mathbf{I}$
subOP(OPChain(r_1, r_2), r)	$(y_0, r, y_2) \leftarrow (y_0, r_1, y_1), (y_1, r_2, y_2)$	$\mathbf{v}_{y_0} \mathbf{M}_r = \mathbf{v}_{y_2}, \mathbf{v}_{y_0} \mathbf{M}_{r_1} = \mathbf{v}_{y_1}, \mathbf{v}_{y_1} \mathbf{M}_{r_2} = \mathbf{v}_{y_2}$	$\mathbf{M}_{r_1} \mathbf{M}_{r_2} = \mathbf{M}_r$

技术难点

分布式表示可以极大地提升可计算性，但该类方法的使用依然非常受限于训练所用知识图谱的质量和结构

因此，由于**稀疏性**，模型在不同数据集上的表现，差异很大

而实际应用中，数据的稀疏性和噪声都要远高于FB15K-237



	WN18	WN18RR	FB15	FB15-237
Entities	40943	40559	14951	14505
Relations	18	11	1345	237
Triples	141442	86835	483142	272115
Testset	5000	3134	59071	20466

Approach	WN18			WN18RR			FB15			FB15-237		
	h@1	h@10	MRR	h@1	h@10	MRR	h@1	h@10	MRR	h@1	h@10	MRR
SimpleE [Kazemi and Poole, 2018]	93.9	94.7	94.2				66.0	83.8	72.7			
ConvE [Dettmers <i>et al.</i> , 2018]	93.5	95.5	94.2	39	48	46	67.0	87.3	74.5	23.9	49.1	31.6
ComplEx-N3 [Lacroix <i>et al.</i> , 2018]		96	95		57	48		91	86		56	37
R-GCN+ [Schlichtkrull <i>et al.</i> , 2018]	69.7	96.4	81.9				60.1	84.2	69.6	15.1	41.7	24.9
CrossE [Zhang <i>et al.</i> , 2019]	74.1	95.0	83.0				63.4	87.5	72.8	21.1	47.4	29.9
AMIE+ [Galárraga <i>et al.</i> , 2015]	87.2	94.8		35.8	38.8		64.7	85.8		17.4	40.9	
RuleN [Meilicke <i>et al.</i> , 2018]	94.5	95.8		42.7	53.6		77.2	87.0		18.2	42.0	

小结

基于分布式表示的推理和基于规则的推理，二者各有优点。在最新的研究中，研究者尝试着将二者的优点加以结合，以提高知识表示和推理效果

Methods	Advantages	Disadvantages	Representative work
rule-based inference	capture hidden semantic information in KGs, improve the accuracy of knowledge reasoning significantly; simulate human reasoning ability, which makes it possible to incorporate priori knowledge to assist in reasoning	rules are not easy to obtain; rules with noise can mislead reasoning	MLNs FOIL PSL PRA
distributed representation-based inference	make full use of structural information existing in KGs; simple, easy to transfer to large-scale KGs	only consider the constraints that satisfy the KG facts, and the deeper compositional information is not considered, which limits the reasoning ability	RESCAL SE TransE TransH TransR TransG

Reference: Chen X, Jia S, Xiang Y. A review: Knowledge reasoning over knowledge graph[J]. Expert Systems with Applications, 2019: 112948.

应用举例：

小型wiki公众人物知识库， 稀疏性超过FB15K-237：

实体： 64485

关系： 23

三元组： 91733

出现次数	实体数
1	59629
2	3593
3	738
4	292
5	154
6	94
7	73
8	55

直接使用TransE 模型：

metric:	MRR	MR	hit@10	hit@3
hit@1				
l(raw):	0.004235	12627.238281	0.005505	0.004033
0.002017				
r(raw):	0.165218	17476.750000	0.214749	0.173434
0.139532				
averaged(raw):	0.084726	15051.994141	0.110127	0.088734
0.070775				
l(filter):	0.006783	11762.375000	0.009484	0.006541
0.003706				
r(filter):	0.167191	17476.294922	0.215676	0.175015
0.142094				
averaged(filter):	0.086987	14619.334961	0.112580	0.090778
0.072900				
0.112580				
0.11257971823215485				

应用举例:

PCA confidence: $conf_{pca}(\vec{B} \Rightarrow r_h(x, y)) :=$

$$\frac{supp(\vec{B} \Rightarrow r_h(x, y))}{\#(x, y) : \exists z_1, \dots, z_m, y' : \vec{B} \wedge r_h(x, y')}$$

例如: $R : livesIn(x, y) \Rightarrow wasBornIn(x, y)$ 规则的PCA confidence 为 0.5

<i>livesIn</i>	<i>wasBornIn</i>
(Adam, Paris)	(Adam, Paris)
(Adam, Rome)	(Carl, Rome)
(Bob, Zurich)	

在wiki知识库上, 我们检验规则, ‘配偶(x, y) & 性别(x, M) => 性别(y, not M)’

得到的Pca_conf= 0.2229

应用举例:

使用**AMIE+**算法挖掘的一阶规则:

```
[[['父母(x, y) <= 子女(y, x)', 0.31746031746031744],  
  ['师长(x, y) <= 配偶(y, x)', 0.11111111111111111],  
  ['目前所在地(x, y) <= 国籍(x, y)', 0.13035204567078973],  
  ['原名(x, y) <= 原名(y, x)', 0.9902912621359223],  
  ['子女(x, y) <= 父母(y, x)', 0.38461538461538464],  
  ['配偶(x, y) <= 师长(y, x)', 0.1],  
  ['配偶(x, y) <= 配偶(y, x)', 0.656934306569343],  
  ['职务(x, y) <= 职务(y, x)', 1.0],  
  ['国籍(x, y) <= 目前所在地(x, y)', 0.13035204567078973]]
```

知识推理的落地展望

知识推理 vs. 人类推理

- 以统计学习为基础的算法，擅长“发现”，但并不能“发现”数据以外的知识
- 需要结合人类的知识对知识图谱加以校验

推理和查询（SQL等）有何不同？

- 推理的“查询条件”更为复杂
- 推理中的规则可以人为编写，也可以从知识图谱中进行挖掘发现
- 本质相同，都以计数为基础

最“实用”的推理算法？

- 规则和分布式表示相结合
- 形成知识表示和推理的闭环结构，互相正向促进，以解决高噪声的问题

感谢聆听！ 欢迎提问

