Differentiating Concepts and Instances for Knowledge Graph Embedding

摘要

大多数传统知识嵌入方法将实体（概念和实例）和关系编码为低维语义空间中的向量，同样忽略了概念和实例之间的差异。在本文中，我们通过区分概念和实例提出了一种名为TransC的新型知识图嵌入模型。具体而言，TransC将知识图中的每个概念编码为一个球体，将每个实例编码为同一语义空间中的一个矢量

介绍

以往的方法都忽略了区分概念和实例，并将两者都视为实现简化的实体。因此，以前工作将导致以下两个缺点：

- 缺乏有效的概念表示
- 缺乏传递性

为了解决这些问题，文中提出了一种名为TransC的新型翻译嵌入模型。在TransC中，每个概念被编码为球体，并且每个实例被编码为同一语义空间中的向量，并且相对位置被用于对概念和实例之间的关系进行建模。更具体地说，instanceOf关系自然地通过检查实例向量是否在概念领域内来表示。

相关工作

- Translation-based Model： TransE TransH TransR/CTransR TransD 其他
- Bilinear Model： RESCAL DisMult HolE ComplEx
- External Information Learning Model : TEKE DKRL

问题

$Kg = \{C, I, R, S\}$

C和I分别表示概念和实例集。关系集R可以形式化为 $R = \{r_e, r_c\} \bigcup R_l$,其中 r_e是instanceOf关系。r_e 是subClassOf关系。R_l是实例关系集，因此三组S 可以分为三个不相交的子集。

1. `InstanceOf` triple set $\mathcal{S}_e = \{(i, r_e, c)_k\}_{k=1}^{n_e}$, where $i \in \mathcal{I}$ is an instance, $c \in \mathcal{C}$ is a concept, and n_e is the size of $\mathcal{S}_e$.

2. `SubClassOf` triple set $\mathcal{S}_c = \{(c_i, r_c, c_j)_k\}_{k=1}^{n_c}$, where $c_i, c_j \in \mathcal{C}$ are concepts, c_i is a sub-concept of c_j, and n_c is the size of $\mathcal{S}_c$.

3. Relational triple set $\mathcal{S}_l = \{(h, r, t)_k\}_{k=1}^{n_l}$, where $h, r \in \mathcal{I}$ are head instance and tail instance, $r \in \mathcal{R}_l$ is an instance relation, and n_l is the size of $\mathcal{S}_l$.

对于每个概念$c \in C$ 学习 $s(p, m)$球体 p是中心 m是半径

instanceOf-subClassOf transitivity

$$(i, r_e, c_1) \in S_e \wedge (c_1, r_c, c_2) \in S_c \rightarrow (i, r_e, c_2) \in S_e, \quad (5)$$

subClassOf-subClassOf transitivity

$$(c_1, r_c, c_2) \in S_c \wedge (c_2, r_c, c_3) \in S_c \rightarrow (c_1, r_c, c_3) \in S_c. \quad (6)$$

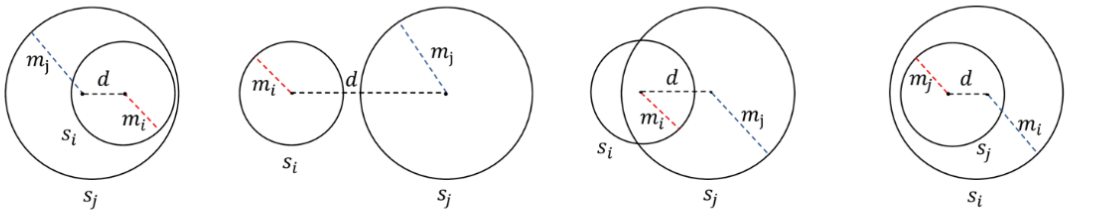
方法

S中有三种三元组，分别为它们定义不同的损失函数

- 实例

$$f_e(i, c) = ||\mathbf{i} - \mathbf{p}||_2 - m.$$

- 子类 四种情况
-



(a) $d < |m_i - m_j| \wedge m_i < m_j$ (GOAL) (b) $d \geq |m_i + m_j|$ (c) $|m_i - m_j| \leq d < |m_i + m_j|$ (d) $d < |m_i - m_j| \wedge m_i \geq m_j$

$$d = ||\mathbf{p}_i - \mathbf{p}_j||_2. \quad (8)$$

$$f_c(c_i, c_j) = ||\mathbf{p}_i - \mathbf{p}_j||_2 + m_i - m_j. \quad (9)$$

$$f_c(c_i, c_j) = ||\mathbf{p}_i - \mathbf{p}_j||_2 + m_i - m_j. \quad (10)$$

$$f_c(c_i, c_j) = m_i - m_j. \quad (11)$$

- 关系

$$f_r(h, t) = ||\mathbf{h} + \mathbf{r} - \mathbf{t}||_2^2. \quad (12)$$

训练

based ranking loss for `instanceOf` triples:

$$\mathcal{L}_e = \sum_{\xi \in \mathcal{S}_e} \sum_{\xi' \in \mathcal{S}'_e} [\gamma_e + f_e(\xi) - f_e(\xi')]_+, \quad (13)$$

where $[x]_+ \triangleq \max(0, x)$ and γ_e is the margin separating positive triplets and negative triplets. Similarly, for `subClassOf` triples, we will have a ranking loss:

$$\mathcal{L}_c = \sum_{\xi \in \mathcal{S}_c} \sum_{\xi' \in \mathcal{S}'_c} [\gamma_c + f_c(\xi) - f_c(\xi')]_+, \quad (14)$$

and for relational triples, we will have a ranking loss:

$$\mathcal{L}_l = \sum_{\xi \in \mathcal{S}_l} \sum_{\xi' \in \mathcal{S}'_l} [\gamma_l + f_r(\xi) - f_r(\xi')]_+. \quad (15)$$

Finally, we define the overall loss function as linear combinations of these three functions:

$$\mathcal{L} = \mathcal{L}_e + \mathcal{L}_c + \mathcal{L}_l. \quad (16)$$

实验

数据集

以前的大多数工作都使用FB15K和WN18（Bordes等，2013）进行评估。但是这两个数据集不适合我们的模型，因为FB15K主要由实例组成，WN18主要包含概念。因此，我们使用另一个流行的知识图YAGO（Suchanek等，2007）进行评估，其中包含来自WordNet的许多概念和来自维基百科的实例。

steps:

(1) We randomly select some relational triples like (h, r, t) from the whole YAGO dataset as our relational triple set $\mathcal{S}_l$.

(2) For every instance and instance relation existed in our relational triples, we save it to construct instance set $\mathcal{I}$ and instance relation set $\mathcal{R}_l$ respectively.

(3) For every `instanceOf` triple (i, r_e, c) in YAGO, if $i \in \mathcal{I}$, we save this triple to construct `instanceOf` triple set $\mathcal{S}_e$.

(4) For every concept existed in `instanceOf` triple set $\mathcal{S}_e$, we save it to construct concept set $\mathcal{C}$.

(5) For every `subClassOf` triple (c_i, r_c, c_j) in YAGO, if $c_i \in \mathcal{C} \wedge c_j \in \mathcal{C}$, we save this triple to construct `subClassOf` triple set $\mathcal{S}_c$.

(6) Finally, we achieve our triple set $\mathcal{S} = \mathcal{S}_e \cup \mathcal{S}_c \cup \mathcal{S}_l$ and our relation set $\mathcal{R} = \{r_e, r_c\} \cup \mathcal{R}_l$.

(1) For every `instanceOf` triple (i, r_e, c) in valid and test dataset, if (c, r_c, c_j) exists in training dataset, we save a new `instanceOf` triple (i, r_e, c_j).

(2) For every `subClassOf` triple (c_i, r_c, c_j) in valid and test dataset, if (c_j, r_c, c_k) exists in training dataset, we save a new `subClassOf` triple (c_i, r_c, c_k).

(3) We add these new triples to valid and test dataset of YAGO39K to get M-YAGO39K.

链路预测

Experiments	Link Prediction					Triple Classification(%)			
Metric	MRR		Hits@N(%)			Accuracy	Precision	Recall	F1-Score
	Raw	Filter	1	3	10				
TransE	0.114	0.248	12.3	28.7	51.1	92.1	92.8	91.2	92.0
TransH	0.102	0.215	10.4	24.0	45.1	90.8	91.2	90.3	90.8
TransR	0.112	0.289	15.8	33.8	56.7	91.7	91.6	91.9	91.7
TransD	0.113	0.176	8.9	19.0	35.4	89.3	88.1	91.0	89.5
HolE	0.063	0.198	11.0	23.0	38.4	92.3	92.6	91.9	92.3
DistMult	**0.156**	0.362	22.1	43.6	66.0	93.5	93.9	93.0	93.5
ComplEx	0.058	0.362	29.2	40.7	48.1	92.8	92.6	**93.1**	92.9
TransC (unif)	0.087	**0.421**	28.3	50.0	69.2	93.5	94.3	92.6	93.4
TransC (bern)	0.112	0.420	**29.8**	**50.2**	**69.8**	**93.8**	**94.8**	92.7	**93.7**

Table 2: Experimental results on link prediction and triple classification for relational triples. Hits@N uses results of "Filter" evaluation setting.

元组分类

Datasets	YAGO39K				M-YAGO39K			
Metric	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
TransE	82.6	83.6	81.0	82.3	71.0↓	81.4↓	54.4↓	65.2↓
TransH	82.9	83.7	81.7	82.7	70.1↓	80.4↓	53.2↓	64.0↓
TransR	80.6	79.4	**82.5**	80.9	70.9↓	73.0↓	66.3↓	69.5↓
TransD	83.2	84.4	81.5	82.9	72.5↓	73.1↓	71.4↓	72.2↓
HolE	82.3	86.3	76.7	81.2	74.2↓	81.4↓	62.7↓	70.9↓
DistMult	**83.9**	**86.8**	80.1	**83.3**	70.5↓	86.1↓	49.0↓	62.4↓
ComplEx	83.3	84.8	81.1	82.9	70.2↓	84.4↓	49.5↓	62.4↓
TransC (unif)	80.2	81.6	80.0	79.7	**85.5**↑	**88.3**↑	81.8↑	85.0↑
TransC (bern)	79.7	83.2	74.4	78.6	85.3↑	86.1↑	**84.2**↑	**85.2**↑

Table 3: Experimental results on `instanceOf` triple classification(%).

Datasets	YAGO39K				M-YAGO39K			
Metric	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
TransE	77.6	72.2	89.8	80.0	76.9↓	72.3↑	87.2↓	79.0↓
TransH	80.2	76.4	87.5	81.5	79.1↓	72.8↓	92.9↑	81.6↑
TransR	80.4	74.7	91.9	82.4	80.0↓	73.9↓	92.9↑	82.3↓
TransD	75.9	70.6	88.8	78.7	76.1↑	70.7↑	89.0↑	78.8↑
HolE	70.5	73.9	63.3	68.2	66.6↓	72.3↓	53.7↓	61.7↓
DistMult	61.9	68.7	43.7	53.4	60.7↓	71.7↑	35.5↓	47.7↓
ComplEx	61.6	71.5	38.6	50.1	59.8↓	65.6↓	41.4↑	50.7↑
TransC (unif)	82.9	77.1	93.7	84.6	83.0↑	77.5↑	**93.1**↓	84.7↑
TransC (bern)	**83.7**	**78.1**	**93.9**	**85.2**	**84.4**↑	**80.7**↑	90.4↓	**85.3**↑

Table 4: Experimental results on `subClassOf` triple classification(%).

未来的工作

（1）球体是一个在语义空间中表示概念的简单模型，但由于它过于幼稚，它仍然有一些限制。我们将尝试找到一个更具表现力的模型而不是球体来表示概念。 （2）概念在不同的三元组中可能有不同的含义。我们将尝试使用几个典型的实例向量作为概念的中心来表示概念的不同含义