1 融合多源信息

多源信息能够帮助构建更加精准的知识表示，经典的知识表示学习模型往往仅关注知识图谱自身的三元组结构信息，而忽略了蕴含在多源信息中的丰富知识。这些多源信息包括：类型信息（实体类型，关系类型等），文本描述信息以及关系路径等，本章中将讨论如何将各种外部融合进知识表示空间中共同学习。

$$L = \sum_{(h,r,t)\in\mathcal{S}} \sum_{(h',r,t')\in\mathcal{S}'} \max(0, f(h,r,t)+\gamma-f(h'+r-t')) \tag{1}$$

1.1 类型信息

所指的类型包括实体类型和关系类型。Guo 等人 [1] 提出语义平滑嵌入模型（Semantically Smooth Embedding, SSE），语义平滑基于的思想是属于同一语义类型的实体在嵌入空间中距离比较近。SSE 利用两种流行学算法 Laplacian eigenmaps 和 Locally linear embedding 来约束这种平滑性假设。Lacian eigenmaps 要求一个实体与它具有同类型的实体在嵌入空间中距离更近，这种约束形式用公式表示为：

$$\Re_1 = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \|\mathbf{e}_i - \mathbf{e}_j\|_2^2 \, w_{ij}^{(1)} \tag{2}$$

其中 $\mathbf{e}_i, \mathbf{e}_j$ 分别是实体 e_i, e_j 的嵌入表示，$w_{ij}^{(1)}$ 代表两个实体之间的邻接矩阵，当两个实体 e_i 和 e_j 属于同一类型时，$w_{ij}^{(1)} = 1$，反之为 0。Locally linear embedding 要求一个实体可以由它临近的实体经过线性组合表示出来。这里临近的实体就是指属于同一个类别的实体，这种约束形式用公式表示为：

$$\Re_2 = \sum_{i=1}^{n} \left\| \mathbf{e}_i - \sum_{e_j\in\mathbb{N}(e_i))} w_{ij}^{(2)} \mathbf{e}_j \right\|_2^2 \tag{3}$$

其中 $\mathbb{N}(e_i)$ 代表与实体 e_i 类型相同的 K 个实体集合，集合的个数 K 是一个超参数。当某个实体 e_j 在集合 $\mathbb{N}(e_i)$ 中，则 $w_{ij}^{(2)}$ 为 1，反之为 0。将 $\Re_1$ $\Re_2$ 加到最大间隔方法里作为整个模型损失函数的一个正则化项从而达到约束嵌入空间语义平滑的作用。实验表明，SSE 模型学习后的嵌入空间中，语义类型一致的实体很好的聚在了一起。

SSE 模型的缺点在于默认所有实体只有一个类型而且每一个类型没有层次特征。而现实世界中的实体不仅有多个类型而且每一个类型是有层次结构的。如图 1 所示 从图中可以看到头实体 William

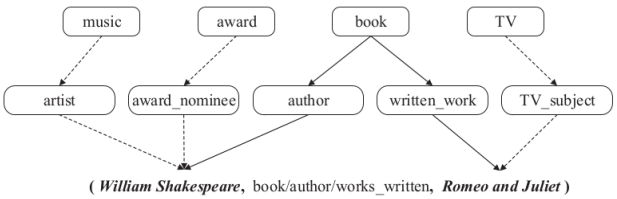

图 1 应该是引用哪个？

Shakespeare 存在很多实体类型，而且每一个实体类型都有子类型。基于以上问题，Xie 等人[6] 提出一种融合实体层次类型信息的模型 TKRL（type-embodied knowledge representation learning）。模型的能量函数定义为：

$$E(h,r,t) = \|M_{r,h}h + r - M_{rt}t\| \tag{4}$$

其中 $M_{r,h}, M_{r,t}$ 分别表示的是头实体和尾实体的类型映射矩阵，是指头实体和尾实体在关系 r 下应该凸显哪一个类型。如图 1 中的例子头实体 William Shakespeare 在 book/author/works_written 关系下应该凸显 book/author 实体类型。为了达到这个目的，TKRL 为每一个子类型也设置了一个映射矩阵，使用这些子类型的映射矩阵构建层次类型映射矩阵。假设每一个实体的类别集合是 c，有 n 个类别，$c = \{c_1, c_2, \cdots, c_n\}$。$c_i$ 是实体的第 i 个类别，而每一个类别又是分层的，有很多个子类型，假设有 m 个子类型，$c_i = \{c_i^1, c_i^2, \cdots, c_i^m\}$，其中 c_i^j 代表实体的第 i 个类型的第 j 个子类型。此时头实体类型映射矩阵表示为：

$$M_{rh} = \alpha_1 M_{c_1^h} + \alpha_2 M_{c_2^h} + \cdots + \alpha_n M_{c_n^h} \tag{5}$$

其中 $M_{c_i^h}$ 代表头实体在它的第 i 个类型下的映射矩阵，每一个 α_i 就是权重，表示的就是实体在当前第 i 个类型下的凸显程度，尾实体类型映射矩阵计算方式同理。为了进一步的处理层次类型信息，TKRL 采用两种层次编码器。第一种是递归层次编码器：

$$M_{c_i^h} = \prod_{i=1}^{m} M_{c_i^h}^{(i)} = M_{c_i^h}^{(1)} M_{c_i^h}^{(2)} \cdots M_{c_i^h}^{(m)} \tag{6}$$

第二种是加权层次编码器：

$$\begin{aligned} M_{c_i^h} =& \sum_{i=1}^{m} \beta_i M_{c_i^h}^{(i)} \\ =& \beta_1 M_{c_i^h}^{(1)} + \beta_2 M_{c_i^h}^{(2)} + \cdots + \beta_m M_{c_i^h}^{(m)} \end{aligned} \tag{7}$$

其中 $M_{c_i^h}^{(j)}$ 代表的是头实体的第 i 个类型的第 j 个子类型的映射矩阵。

实体类型信息还能在模型中作为类型限制，帮助模型学习更好的知识表示。由于在嵌入空间中相似类型的实体距离较近，这使得模型很难区分这些实体，从而在知识图谱补全或者三元组分类任务上效果不好。为了解决这个问题，Krompaß 等人[2] 使用硬类型限制控制负例三元组的生成，具体的就是强制替换的实体与原来的实体具有同样的类型，但是这样可能会打乱嵌入空间中原有的语义分布。TKRL 模型采用一种软类型限制（Soft Type Contraint，STC）机制，STC 的策略是以某一定的比例选取那些与被替换实体有相同类型的实体，如下公式

$$P(\widetilde{e} \in E_c) = \frac{(k+1)\,|E_c|}{|E| + k\,|E_c|} \tag{8}$$

其中 $E_c \in E$ 代表的一个实体集合，其中所有实体都有类型 c，k 是一个超参数，含义是选取在 E_c 集合中的实体作为负例三元组的几率是选取那些不在 E_c 集合中的实体几率的 k 倍。

除了考虑实体类型信息外，Zhang 等人[8] 提出一种利用关系层次结构的知识表示模型 HRS（Hierarchical Relation Structure）。HRS 将知识图谱中的关系划分为一个三层的层次结构，顶层关系 r_c 是由语义相近的关系聚类而成，中间层指特定的关系 r'，底层关系 r_s 是将每一个关系再进一步的划分成多个子关系。在训练时，对于每一个三元组 (h, r, t)，其中关系 r 的嵌入表示 $\mathbf{r}$ 为三层关系的嵌入表示之和 $\mathbf{r} = \mathbf{r}_c + \mathbf{r}' + \mathbf{r}_s$。HRS 模型的损失函数定义为

$$L = L_{Orig} + L_{HRS} \tag{9}$$

其中 L_{Orig} 采用经典模型 TransE，TransR 等的最大间隔方法，L_{HRS} 作为正则化项约束模型学习层次关系的表示。

$$L_{HRS} = \lambda_1 \sum_{\mathbf{r}_c \in \mathcal{C}} \|\mathbf{r}_c\|_2^2 + \lambda_2 \sum_{\mathbf{r}' \in \mathcal{R}} \|\mathbf{r}'\|_2^2 + \lambda_3 \sum_{\mathbf{r}_s \in \mathcal{S}} \|\mathbf{r}_s\|_2^2 \tag{10}$$

其中 $\mathcal{C}$，$\mathcal{R}$，$\mathcal{S}$ 分别代表关系聚类集合，关系集合，子关系聚类集合。此外在知识图谱中某些关系代表的是实体的属性信息，而属性与关系的区别在于属性有明显的一对多关系。Lin 等人提出一种知识表示模型 KR-EAR，其中 EAR 分别指实体，属性，关系（entities，attributes，relations）。KR-EAR 将所有的关系分离出关系与属性，然后再建模实体表示。

1.2 文本描述

知识图谱中的很多实体都是带有描述信息的，如图所示。这些描述文本是对实体相关信息的详细描

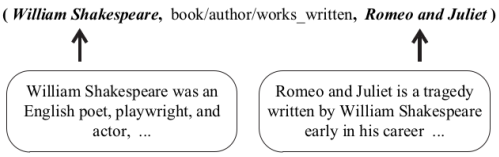

图 2　引用那个呢

述，这些文本可以认为是对知识图谱中结构化信息的补充，利用这些实体的文本信息有助于建模知识图谱的语义关系从而得到更加精准的知识表示。此外那些仅仅基于知识图谱结构化信息的知识表示模型无法处理不在知识图谱中的实体（out of KG），而联合文本嵌入的方式可以做到互补使得模型学习到那些在文本中出现而不在知识图谱中的实体。

Wang 等人[4] 首先提出联合知识图谱和文本的知识表示学习模型。整个模型分成三个部分：知识模型、文本模型、对齐模型。知识模型基于平移模型的思想学习实体和关系的表示，文本模型利用 Skip-gram 模型的思想学习词向量，最后对齐模型实现知识空间和文本空间对齐，对齐原则是利用实体名称或者维基百科锚文本。模型的损失函数为这三个子模块之和

$$\mathcal{L} = \mathcal{L}_K + \mathcal{L}_T + \mathcal{L}_A \tag{11}$$

其中 $\mathcal{L}_K$，$\mathcal{L}_T$，$\mathcal{L}_A$ 分别是知识模型、文本模型、对齐模型的损失函数。

$$\begin{aligned} \mathcal{L}_K = -\sum_{(h,r,t)} & \log \Pr(h|r,t) \\ & + \log \Pr(t|h,r) + \log \Pr(r|h,t) \end{aligned} \tag{12}$$

知识模型用来学习知识图谱中三元组的表示，

以Pr(h|r,t) 为例。

$$\Pr(h|r,t) = \frac{\exp z(h,r,t)}{\sum_{\widetilde{h}} \exp z(\widetilde{h},r,t)} \tag{13}$$

$$z(h,r,t) = b - \frac{1}{2}\left\|\mathbf{h}+\mathbf{r}-\mathbf{t}\right\|_2^2 \tag{14}$$

文本模型用来学习实体描述文本中单词的表示，基于 Skip-gram 的思想，定义文本模型的损失函数如下：

$$\mathcal{L}_T = -\sum_{(w,v)\in\mathcal{C}} \log \Pr(w|v) \tag{15}$$

其中 $\Pr(w|v)$ 代表两个单词在滑动窗口内的共现概率，$\mathcal{C}$ 代表滑动窗口内共现单词的集合。

$$\Pr(w|v) = \frac{\exp z(w,v)}{\sum_{\widetilde{w}} \exp z(\widetilde{w},v)} \tag{16}$$

$$z(w,v) = b - \frac{1}{2}\left\|\mathbf{w}-\mathbf{v}\right\|_2^2 \tag{17}$$

但是由于实体名称歧义性较大的问题利用实体名称对齐的原则对打乱文本原有的语义空间，

$$\mathcal{L}_A = \sum_{(w,v)\in\mathcal{C},v\in\mathcal{A}} \log \Pr(w|e_v) \tag{18}$$

其中 $\mathcal{A}$ 表示锚文本的集合，e_v 代表实体描述文本中单词 v 所对应的在锚文本中的实体。利用维基百科锚文本对齐的原则依赖于特定的数据源，这使得这种方式无法应用到其它数据源领域。为了解决以上问题，Zhong 等人[9] 在以上的知识表示学习模型的基础上做了改进。在对齐模型的损失函数 $\mathcal{L}_A$，利用实体描述文本作为对齐原则，认为实体的表示向量应当与描述文本单词的词向量尽可能接近。

$$\mathcal{L}_A = -\sum_{e\in E}\sum_{w\in D_e} \log \Pr(w|e) + \log \Pr(e|w) \tag{19}$$

E 代表知识图谱中的实体集合，D_e 代表实体 e 的描述文本。

以上两个模型考虑的是单词级别的文本信息而没有利用整个文本的语序语义信息。Xie 等人[7] 提出一种融合实体描述的知识表示模型（Description-embodied knowledge representation learning，DKRL）。DKRL 在 TransE 模型的基础上融合实体描述的文本信息，为每一个实体设置两种知识表示。第一种是基于知识图谱结构化信息的表示，可以随机初始化或者利用 TransE 模型预训练好的实体表示。第二种是基于描述的表示，使用连续词袋（continuous bag-of-words，CBOW）模型和卷积神经网络（Convolutional neural network，CNN）模型从文本中构建。给定一三元组 (h,r,t)，DKRL 模型的能量函数定义如公式：

$$\begin{aligned} E(h,r,t) =& ||\mathbf{h}_S + \mathbf{r} - \mathbf{t}_S|| + ||\mathbf{h}_S + \mathbf{r} - \mathbf{t}_D|| \\ &+ ||\mathbf{h}_D + \mathbf{r} - \mathbf{t}_S|| + ||\mathbf{h}_D + \mathbf{r} - \mathbf{t}_D|| \end{aligned} \tag{20}$$

其中 $\mathbf{h}_S/\mathbf{h}_D$ $\mathbf{t}_S/\mathbf{t}_D$ 分别代表头实体/尾实体的基于结构的表示和基于实体描述的表示，$\mathbf{r}$ 代表关系的知识表示。通过混合项 $||\mathbf{h}_S+\mathbf{r}-\mathbf{t}_D||$，$||\mathbf{h}_D+\mathbf{r}-\mathbf{t}_S||$，$||\mathbf{h}_D+\mathbf{r}-\mathbf{t}_D||$ 的限制，DKRL 模型将实体的两种不同的表示向量映射到同一个语义空间中。

Wang 等人[5] 提出一种增强型文本知识嵌入模型（Text-enhanced knowledge embedding，TEKE）。TEKE 不仅将文本信息融入到知识图谱的空间中，并且每一个关系对于不同的头实体和尾实体都有不同的表示，这样可以更好的解决一对多，多对一以及多对多的关系问题。首先使用实体连接工具标注出文本中是知识图谱中实体的单词并且基于这些实体单词构建共现网络（co-occurrence network）。对于文本中的实体 e，定义它的上下文集合为 $n(e)$，它的文本向量表示为 $\mathbf{n}(e)$，$\mathbf{n}(e)$ 是 $n(e)$ 中所有单词词向量的加权求和，权重为两个实体单词共同出现的频次归一化后的值。给定一个三元组 (h,t,r)，对于两个实体 h,t 的上下文集合 $n(h)$，$n(t)$ 中共同出现的单词往往反映着两个实体之间的关系，定义 $n(h,t) = n(h) \cap n(t)$ 作为两个实体上下文的匹配集合，其向量表示为 $\mathbf{n}(h,t)$，它是 $n(h,t)$ 中所有实体匹配表示的加权求和，可以看出当 h 或 t 不同时关系的表示也是不同的，因此 TEKE 模型可以很好的处理 1-to-N，N-to-1，N-to-N 关系。最后将文本的语义空间映射到知识图谱的语义空间，见公式。

$$\begin{aligned} \widehat{h} &= \mathbf{n}(h)A + \mathbf{h} \\ \widehat{r} &= \mathbf{n}(h,t)B + \mathbf{r} \\ \widehat{t} &= \mathbf{n}(t)A + \mathbf{t} \end{aligned} \tag{21}$$

其中 A,B 是映射矩阵，$\mathbf{h},\mathbf{r},\mathbf{t}$ 是偏置项，它们都是可学习的参数，词向量利用 Word2Vec 工具预训练。模型的得分函数定义为 $f(h,r,t) = ||\widehat{h}+\widehat{r}-\widehat{t}||_2^2$，损失函数采用最大间隔函数（公式 1）。

1.3 关系路径

Lin 等人[3] 提出 PTransE 模型，在 TransE 模型的基础上将两个实体之间的多步关系路径看做两个实体之间相连的关系。由于两个实体之间存在大量的关系路径，PTransE 模型采用路径约束资源分配（path-constraint resource allocation，PCRA）算法用来提取出可以用来建模两个实体之间关系的可靠的关系路径。PTransE 模型的优化目标分为两部分：第一部分是利用 TransE 模型建模两个实体之间直接相连的关系；第二部分是建模两个实体之间多关系路径。定义 $P(h,t)=\{p_1,p_2,\cdots,p_N\}$ 代表实体 h 和 t 之间的多关系路径的集合，模型的能量函数定义如下其中 $E(h,r,t)$ 即为公式（1）中 TransE 模型的能量函数 $f(h,r,t)$。给定头实体 h 和尾实体 t 下的某一多条关系路径 $p=(r_1,r_2,\cdots,r_l)$，r_i 代表路径 p 上的第 i 个关系，l 是路径上关系的个数。$E(h,p,t)$ 代表给定实体 h 和 t 以及多跳关系路径 p 的能量函数。

$$E(h,p,t)=\|\mathbf{p}-\mathbf{r}\| \tag{22}$$

其中 $\mathbf{r}$ 代表 h 和 t 之间直接相连的关系 r 的向量表示，$\mathbf{p}$ 代表 h 和 t 之间的多跳关系路径 p 的向量表示。PTransE 模型采用三种方式计算 $\mathbf{p}$。

$$\begin{aligned}\mathbf{p}&=\mathbf{r}_1+\mathbf{r}_2+\cdots+\mathbf{r}_l \quad \text{ADD}\\ \mathbf{p}&=\mathbf{r}_1\cdot\mathbf{r}_2\cdot\cdots\cdot\mathbf{r}_l \quad \text{MUL}\\ \mathbf{c}_i&=f(W[\mathbf{c}_{i-1};\mathbf{r}_i]) \quad \text{RNN}\end{aligned}$$

ADD 表示将路径上所有关系的向量表示相加，MUL 表示相乘，RNN 代表利用循环神经网络，此时 $\mathbf{p}$ 为最后时刻的隐藏状态。顺序有问题，不需要介绍 $E(h,P,t)$，直接 $E(h,p,t)$，然后过渡到 $L(h,p,t)$，最后解释 $R(p|h,t)$

References

[1] Shu Guo et al. “Semantically Smooth Knowledge Graph Embedding”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*. The Association for Computer Linguistics, 2015, pp. 84–94. DOI: 10.3115/v1/p15-1009. URL: https://doi.org/10.3115/v1/p15-1009.

[2] Denis Krompaß, Stephan Baier, and Volker Tresp. “Type-Constrained Representation Learning in Knowledge Graphs”. In: *The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part I*. Ed. by Marcelo Arenas et al. Vol. 9366. Lecture Notes in Computer Science. Springer, 2015, pp. 640–655. DOI: 10.1007/978-3-319-25007-6_37. URL: https://doi.org/10.1007/978-3-319-25007-6%5C_37.

[3] Yankai Lin et al. “Modeling Relation Paths for Representation Learning of Knowledge Bases”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*. Ed. by Lluís Màrquez et al. The Association for Computational Linguistics, 2015, pp. 705–714. DOI: 10.18653/v1/d15-1082. URL: https://doi.org/10.18653/v1/d15-1082.

[4] Zhen Wang et al. “Knowledge Graph and Text Jointly Embedding”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1591–1601. DOI: 10.3115/v1/D14-1167. URL: https://www.aclweb.org/anthology/D14-1167.

[5] Zhigang Wang and Juan-Zi Li. “Text-Enhanced Representation Learning for Knowledge Graph”. In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial In-*

telligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016. Ed. by Subbarao Kambhampati. IJCAI/AAAI Press, 2016, pp. 1293–1299. URL: http://www.ijcai.org/Abstract/16/187.

[6] Ruobing Xie, Zhiyuan Liu, and Maosong Sun. "Representation Learning of Knowledge Graphs with Hierarchical Types". In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. Ed. by Subbarao Kambhampati. IJCAI/AAAI Press, 2016, pp. 2965–2971. URL: http://www.ijcai.org/Abstract/16/421.

[7] Ruobing Xie et al. "Representation Learning of Knowledge Graphs with Entity Descriptions". In: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. AAAI' 16. Phoenix, Arizona: AAAI Press, 2016, pp. 2659–2665.

[8] Zhao Zhang et al. "Knowledge Graph Embedding with Hierarchical Relation Structure". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Ed. by Ellen Riloff et al. Association for Computational Linguistics, 2018, pp. 3198–3207. DOI: 10.18653/v1/d18-1358. URL: https://doi.org/10.18653/v1/d18-1358.

[9] Huaping Zhong et al. "Aligning Knowledge and Text Embeddings by Entity Descriptions". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*. Ed. by Lluís Màrquez et al. The Association for Computational Linguistics, 2015, pp. 267–272. DOI: 10.18653/v1/d15-1031. URL: https://doi.org/10.18653/v1/d15-1031.