基于混合专家模型的词语上下位关系判别方法

曾楠, 谢志鹏

引用本文

曾楠, 谢志鹏. 基于混合专家模型的词语上下位关系判别方法[J]. 计算机科学, 2023, 50(2): 285-291.

ZENG Nan, XIE Zhipeng. Mixture-of-Experts Model for Hypernymy Discrimination[J]. Computer Science, 2023, 50(2): 285-291.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.211200066

基于混合专家模型的词语上下位关系判别方法

曾　楠　　谢志鹏
复旦大学计算机科学技术学院　上海 200438
(nzeng19@fudan.edu.cn)

摘　要　词语的上下位关系判别是自然语言处理中一项基础且具有挑战性的任务。传统的有监督方法通常采用单个模型在整个语义空间中对所有上下位词对进行全局建模,并取得了一定的效果。然而,上下位关系的分布式语义表征具有相当的复杂性,在语义空间的不同区域中往往具有不同的表现,使得全局模型难以学习。针对此问题,文中提出了基于混合专家的上下位关系判别方法。该模型基于分而治之的策略,将语义空间划分为多个子空间,每个子空间对应一个局部专家(模型),局部专家(模型)关注它们自己的子空间,并采用门控机制决定空间的分割和专家的混合。实验结果表明,这种专家混合模型在公开数据集上的性能优于传统的全局模型。

关键词:上下位关系判别;混合专家;局部模型
中图法分类号　TP391

Mixture-of-Experts Model for Hypernymy Discrimination

ZENG Nan and XIE Zhipeng
School of Computer Science,Fudan University,Shanghai 200438,China

Abstract　Hypernymy discrimination is an essential and challenging task in NLP. Traditional supervised methods usually model all the hypernymies in the global semantic space,which has achieved fair performance. However,the distributed semantic representation of hypernymies is rather complex,and their manifestations may differ significantly in different areas of the semantic space,making it difficult to learn the global model. This paper employs the mixture-of-experts framework as a solution. It works on the basis of a divide-and-conquer strategy,which divides the semantic space into multiple subspaces,and each subspace corresponds to a local expert(model). A number of localized experts(models) focus on their own domains(or subspaces) to learn their specialties,and a gating mechanism determines the space partitioning and the expert aggregation. Experimental results show that the mixture-of-experts model outperforms the traditional global ones on public datasets.

Keywords　Hypernymy discrimination,Mixture-of-Experts,Local model

1　引言

上下位关系是一种重要的词汇语义关系,用于描述词语之间的层次性隶属关系。例如,在词对"玫瑰-花"中,"玫瑰"是"花"的下位词(Hyponym),"花"则是"玫瑰"的上位词(Hypernym)。迄今为止,大多数知识图谱的组织架构都是依赖is-a(上下位)关系作为其骨架,这包括实体与概念(Instance-of)关系(例如"西班牙-国家")、类与子类(Subclass-of)关系(例如"哺乳动物-动物")等。上下位关系的词汇知识已经被广泛应用于各项自然语言理解的下游任务,包括但不限于分类体系构建[1]、文本蕴含识别[2-3]和问答[4]等。正是因为上下位关系的重要应用价值,研究人员提出了上下位词对判别任务,其目标是从众多的词对(包括同义词、反义词、共上位词等)中区分出上下位词对,并对其展开了广泛的研究。

近年来,随着深度学习的发展,词向量作为词语分布式语义的一种表征方法,在自然语言处理中占据了基础性的地位[5]。上下位关系判别问题往往被转换成对词向量关系的判定。一种方法是利用词向量的拼接(Concatenation)或差(Difference)作为词对的表征,然后训练出一个支持向量机判别器来判定词对关系[6-7]。另一种方法则利用了双线性变换来建模上下位词的非对称性以判别上下位词对[8]。此外,Rei等[9]提出了有向相似度神经网络来度量两个词的上下位

到稿日期:2021-12-05　返修日期:2022-05-01
基金项目:国家重点研发计划(2018YFB1005100);国家自然科学基金(62076072)
This work was supported by the National Key Research and Development Program of China(2018YFB1005100) and National Natural Science Foundation of China(62076072).
通信作者:谢志鹏(xiezp@fudan.edu.cn)

关系。这些方法都是在全局语义空间中对上下位关系进行整体建模，并取得了较好的判别效果。

然而，上下位关系在分布式语义空间中的表示是复杂的，它在语义空间的不同区域（或不同的子空间）中往往具有不同的表现，从而产生了复杂的决策边界，全局模型的学习较为困难。针对此问题，本文提出了一种混合专家模型进行求解。该方法不是试图去学习出一个全局判别模型，而是将语义空间划分为多个子空间（或区域），并在每个子空间中对应学习出一个局部专家。局部专家仅仅聚焦于对应的子空间，而局部专家的集成则覆盖了整个语义空间。

本文第 2 节介绍了上下位词对判别的相关工作；第 3 节介绍了使用混合专家模型的动机，并给出了该方法的详细描述；第 4 节介绍了实验设置以及实验结果；最后总结全文。

2 相关工作

由于上下位关系在自然语言理解中的重要性，研究人员对上下位词对判别做了大量的研究。在早期，人们通过手工构造了一些词汇知识库，如 WordNet[10]，然而人工构造这样一个分类体系是耗时耗力的过程，并且在范围和领域上也是有限的。因此，研究人员提出了多种方法来从数据中自动识别上下位词，大致可以分为以下几类。

2.1 基于模式匹配的方法

传统的基于模式匹配的方法是，当一个词对 (x,y) 同时出现在一个句子中且路径满足一定的模式（例如，[y] such as [x]）时，则认为 (x,y) 存在上下位关系。Hearst[11] 通过人工构造一些模式来检测上下位词的关系，而这种模式的覆盖率往往较低。为了提高模式的覆盖率，文献[12]采用了自举法找出更多的匹配模式，文献[13]利用 LSTM 编码句法分析树中的路径（词汇句法模式）来区分上下位词对。此外，Shi 等[14]使用异构信息网络增加了其他的上下文表示。在基于模式的方法中，且上下位词必须同时出现在一个句子中，否则无法判断词对是否具有上下位关系。基于模式的方法的潜在缺点是覆盖率低，且抽取结果的召回是有限的。

2.2 基于无监督的方法

为了解决模式匹配方法覆盖率低的问题，基于分布式语义的方法利用两个词的分布表示建模了词对具有上下位关系的可能性。分布式语义模型最初是用分布假设[15]来度量词语之间的语义相似性，其假设是词的语义由其上下文决定（通过上下文表示目标词的语义）。为了在分布式语义模型中识别出上下位关系，建模上下位关系的非对称性，许多研究者提出了非对称的相似性度量方法。这些方法大多数依赖于分布包含假说（DIH）[16-17]：下位词的上下文特征是上位词上下文特征的子集。

Shwartz 等[18]分别利用基于窗口的上下文和基于依存的上下文，采用非对称的度量方法来区分上下位词。此外，相关的非对称的相似度衡量方法还有 Weeds-Precision[16]，ClarkeDE[19]，balAPinc[17] 等。这些方法通过计算下位词的上下文特征在上位词的上下文特征中被包含的百分比（不同的上下文特征的权重不同）来建模分布包含假说。无监督的方法不依赖于现有的大规模标记的上下位词，适用于低资源语言。但是，无监督方法通常不如有监督方法准确和有效。

2.3 基于有监督的方法

传统的有监督方法使用两个词的分布向量作为特征，如 Concat 模型、Diff 模型[20] 等。给定词对 (x,y)，$\boldsymbol{x}$ 和 $\boldsymbol{y}$ 分别表示词语 x 和 y 的词向量。Baroni 等[6]将两个词向量拼接 $concat(\boldsymbol{x}\oplus\boldsymbol{y})$ 作为特征，训练全局的高斯核函数的支持向量机（SVM）模型。其他一些研究者[7,21-22]将 $diff(\boldsymbol{y}-\boldsymbol{x})$ 作为特征来训练不同的全局判别器，以判别上下位词对。

近年来，由于上下位词的语义层次结构，研究者提出了一些学习上下位词词向量的方法[14,23]。Shi 等[14] 利用 max-margin 神经网络学习上下位词的词向量。Nguyen 等[23]提出在文本和 WordNet 概念层次上联合训练层次词向量，以进行上下位词对判别。Glavas 等[8]利用上下位的非对称性以及特定词嵌入的转换来捕捉上下位关系。Dash 等[24] 利用严格偏序网络建模上下位关系的非对称性和传递性。

这些有监督的模型都是在一个全局的模型中区分上下位词，但在整个分布语义空间中，上下位词在空间的不同区域（不同子空间）的表现是不同的，学习统一的全局模型比较困难，因此本文提出了基于混合专家的上下位判别方法，将语义空间划分为多个子空间，每个子空间有专门的判别器。不同的词对对应不同的子空间，选择不同的专家进行判别。本文模型可针对不同的词汇语义关系进行设计[25]。

3 本文方法

给定训练数据集 H，其包含有 N 个词对的训练集，$H=\{(x^{(n)},y^{(n)})\}_{n=1}^{N}$，$t^{(n)}$ 表示第 n 个词对的真实标签，当 $t^{(n)}=1$ 时，表示 $y^{(n)}$ 是 $x^{(n)}$ 的上位词。为了比较全局空间和局部子空间中词对的分布，本文在两种空间上分别做了 t-SNE 降维可视化[26]。此外，由于 Turney 等[20] 的研究表明 $(\boldsymbol{y}-\boldsymbol{x})$ 是区分上下位词的重要特征，因此本文利用 $(\boldsymbol{y}-\boldsymbol{x})$ 在训练数据集的全局语义空间上做了 t-SNE 降维可视化，如图 1 所示。图 1 中，每个样本点都表示了一个词对 (x,y)，蓝色点表示 y 是 x 的上位词，红色点表示 (x,y) 是非上下位词对，当不同词汇语义关系词对混在一起时，在一个全局的空间中，利用特征 $(\boldsymbol{y}-\boldsymbol{x})$ 对上下位词对和非上下位词对进行划分非常困难，具有不同词汇语义关系的词对之间重叠度大，决策边界不清晰。

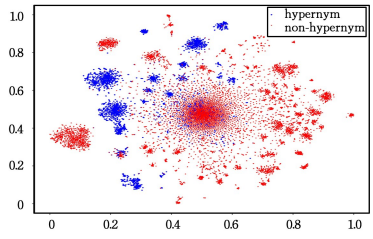


图 1 全局空间中 $(\boldsymbol{y}-\boldsymbol{x})$ 的 t-SNE 可视化（电子版为彩图）

Fig. 1 t-SNE visualization of $(\boldsymbol{y}-\boldsymbol{x})$ in global space

为了观察数据在局部模型中的分布，本文利用下位词词向量对样本空间使用 K-means 聚类，聚类的簇数 $cluster=$

16，并将每个子空间内样本的（$\boldsymbol{y}-\boldsymbol{x}$）特征使用 t-SNE 进行降维可视化。我们从中选出了样本分布较为稠密的 4 个子空间分布图，如图 2 所示。

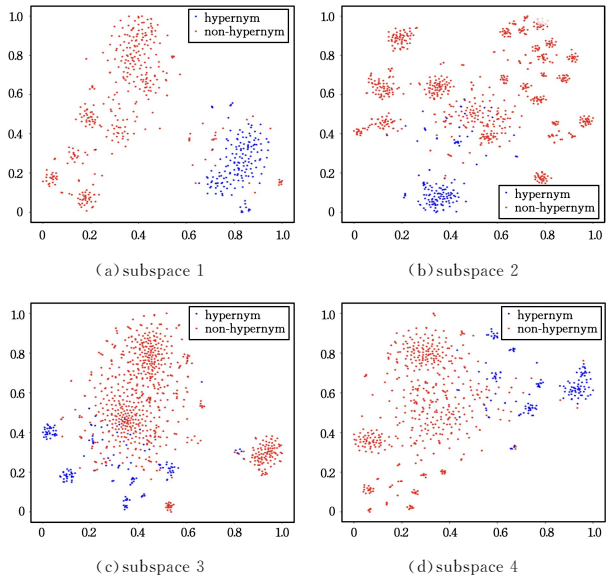


图 2　局部子空间中（$\boldsymbol{y}-\boldsymbol{x}$）的 t-SNE 可视化图

Fig. 2　t-SNE visualization of ($\boldsymbol{y}-\boldsymbol{x}$) in local subspaces

图 2 的 4 个划分子空间中，上下位词对与非上下位词对重叠较少，决策边界清晰，且将全局语义空间划分为多个子空间后，子空间对应的样本数减少，因此在单个子空间中训练专家判别器会更加容易，这与混合专家模型[27]分而治之的思想是完全一致的。

由于在全局空间中很难找出统一的判别器对上下位关系进行判别。因此本文提出了基于混合专家的词语上下位关系判别（MoE-HD）的框架。图 3 给出了模型框架，它采用分而治之的方式，将语义空间划分为若干子空间，每个子空间由一个特定的专家负责，每个专家会对输入的词对进行判别。此外，由门控机制来确定联合决定最终判定结果的专家。

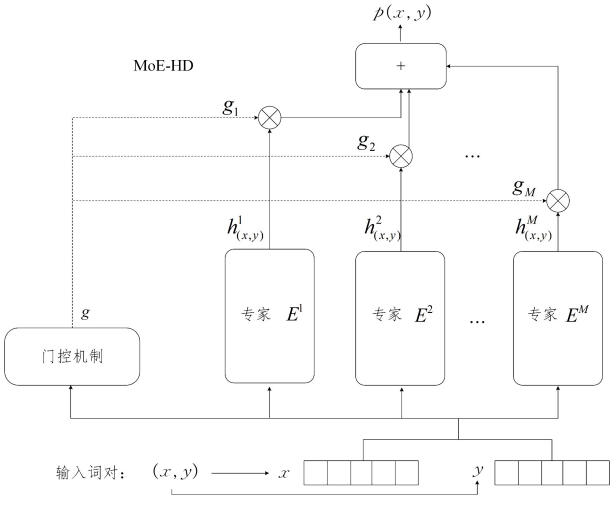


图 3　MoE-HD 模型架构

Fig. 3　Architecture of MoE-HD

3.1　局部专家

在 MoE-HD 模型架构中，局部专家都是同质的，它们拥有相同的网络结构但参数不同。给定一个词（x，y）作为输入，每个专家E^i计算（x，y）是上下位词对的未归一化的可能性$h^i(x,y)$。

局部专家分别采用了词向量拼接 $concat(\boldsymbol{x}\oplus\boldsymbol{y})$[6]（Concat 模型）和 $diff(\boldsymbol{y}-\boldsymbol{x})$[7]（Diff 模型）作为上下位关系的判别特征，其中词向量 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 的维度为d_e。

在 Concat 模型中，关系向量$\boldsymbol{r}_c$为：

$$\boldsymbol{r}_c=\boldsymbol{x}\oplus\boldsymbol{y} \tag{1}$$

接着，关系向量$\boldsymbol{r}_c$经过含有一个隐层的 MLP 来获得（x，y）是上下位词对的分数$h^i_c(x,y)$：

$$h^i_c(\boldsymbol{x},\boldsymbol{y})=(\boldsymbol{m}^i_{c\_o})^{\mathrm{T}}\cdot\mathrm{ReLU}(\boldsymbol{r}_c\cdot\boldsymbol{M}^i_{c\_h}+\boldsymbol{b}^i_{c\_h})+b^i_{c\_o} \tag{2}$$

其中，MLP 的隐藏层有d_{c_h}个单元，$\boldsymbol{M}^i_{c_h}$的维度是 2 $d_e\times d_{c\_h}$，$\boldsymbol{b}^i_{c_h}$和$\boldsymbol{m}^i_{c_o}$是两个长度为d_{c_h}的向量，$b^i_{c_o}$是偏置。

类似地，在 Diff 模型中，关系向量$\boldsymbol{r}_d$和$h^i_d(w_1,w_2)$分别为：

$$\boldsymbol{r}_d=\boldsymbol{y}-\boldsymbol{x} \tag{3}$$

$$\boldsymbol{h}^i_d(\boldsymbol{x},\boldsymbol{y})=(\boldsymbol{m}^i_{d\_o})^{\mathrm{T}}\cdot\mathrm{ReLU}(r_d\cdot\boldsymbol{M}^i_{d\_h}+\boldsymbol{b}^i_{d\_h})+\boldsymbol{b}^i_{d\_o} \tag{4}$$

其中，MLP 的隐藏层有d_{d_h}个单元，$\boldsymbol{M}^i_{d_h}$的维度是$d_e\times d_{d_h}$，$\boldsymbol{b}^i_{d_h}$和$\boldsymbol{m}^i_{d_o}$是两个长度为d_{d_h}的向量，$b^i_{d_o}$是偏置。

3.2　门控机制

假定在 MoE-HD 模型中有 M 个局部专家，对于每个输入的词对（x，y），我们会得到 M 个上下位词分数 $\boldsymbol{h}=[h^i(x,y)]_{1\leqslant i\leqslant M}$，其中$h^i(x,y)$为第 i 个专家的输出。因此，最后需要融合得出上下位词检测的最终分数。

在 MoE-HD 模型中，最终的分数是通过 M 个局部专家产生的分数加权求和得到的。

$$s(x,y)=\boldsymbol{g}^{\mathrm{T}}\cdot\boldsymbol{h} \tag{5}$$

其中，g 是在 M 维的单纯形上表示专家对最终成绩的贡献比例。门控机制用于计算特定词对(x,y)对应的 $\boldsymbol{g}$，实现专家的动态混合。

$$\boldsymbol{g}=\mathrm{softmax}(\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{M}_g) \tag{6}$$

其中，$\boldsymbol{M}_g$是门控机制的参数矩阵，且$\boldsymbol{M}_g\in\mathbb{R}^{d_e\times M}$，$\boldsymbol{M}_g$的第 i 列表示第 i 个专家的表示向量，同时下位词的词向量与表示向量的乘积为第 i 个专家的注意力分数。对注意力分数进行 softmax 得到权重向量 $\boldsymbol{g}$，权重向量 $\boldsymbol{g}$ 实现了对专家的选择，由于参数矩阵和专家的参数都是随机初始化的，同时本文采用了下位词词向量选择子空间，因此不同词对选择的子空间不同，且在训练过程中会强化对子空间的选择以及优化专家的效果。

在上下位关系中，下位词的数量总是远大于上位词的数量，一个上位词往往对应多个下位词，但下位词往往只对应一个上位词（不考虑传递闭包）。为了更准确地划分语义子空间，本文利用下位词词向量作为选择语义子空间的特征。

为了避免表示向量的模长过长而导致当特征向量与其方向一致（相反）时，专家的权重非常大（小），进而导致在专家选择时极度增大（压缩）模长较长的专家权重，因此本文对专家的表示向量进行了归一化，使得专家的权重完全由特征向量和专家表示向量的方向决定。

3.3 模型预测和损失函数

给定词对(x,y)，对最后的得分计算 sigmoid 函数值，得到其上下位关系的概率：

$$p(x,y)=\sigma(s(x,y)) \tag{7}$$

对于含有 N 个词对的训练集 $H=\{(x^{(n)},y^{(n)})\}_{n=1}^{N}$，模型采用了交叉熵损失函数：

$$L=\frac{1}{N}\sum_{n=1}^{N}[t^{(n)}\log p^{(n)}+(1-t^{(n)})\log(1-p^{(n)})] \tag{8}$$

其中，$p^{(n)}=p(x^{(n)},y^{(n)})$是模型对第 n 个词对的上下位关系预测概率。

算法的训练过程如算法 1 所示。

算法 1 基于混合专家的词语上下位关系判别算法的训练过程

输入：词语 x 和 y 的词向量 $\mathbf{x},\mathbf{y}$ 以及标签 t，迭代次数 T

输出：M 个局部专家$\{E^i:0\leqslant i<M\}$，门控参数矩阵$\mathbf{M}_g$

1. 初始化：随机初始化专家 $\mathbf{E}=[E^i]_{1\leqslant i\leqslant M}$，门控参数矩阵$\mathbf{M}_g$，损失函数 $L(E^1,E^2,\cdots,E^M,\mathbf{M}_g)$
2. for it=0 to it=T−1 do
3. 生成特征$\mathbf{r}_c=\mathbf{x}\oplus\mathbf{y}$(或特征$\mathbf{r}_d=\mathbf{y}-\mathbf{x}$)
4. for i=0 to i=M−1 do
5. 将特征$\mathbf{r}_c$(或$\mathbf{r}_d$)经过专家得到非归一化概率$h_{x,y}^i=MLP^i(\mathbf{r}_c)$ ($h_{x,y}^i=MLP^i(\mathbf{r}_d)$)
6. end for
7. 计算 M 个专家的注意力权重 $\mathbf{g}=\mathrm{softmax}(\mathbf{x}^T\cdot\mathbf{M}_g)$
8. 基于上下位词对分数$h_{x,y}^i$和门控机制计算上下位关系概率$p^{(n)}=\sigma(\mathbf{g}^T\cdot\mathbf{h})$
9. 计算损失 $L=\mathrm{BinaryCrossEntropy}(p^{(n)},t^{(n)})$
10. 更新专家E^i和门控参数矩阵$\mathbf{M}_g$
11. end for

4 实验结果和分析

4.1 数据集

本文在 HyperNET[13]数据集上评估本文方法。Shwartz 等从多个外部的知识图谱抽取了只有直接关系的成对词语(即没有传递闭包)，构造了这个用于词语上下位关系检测的数据集。Levy 等[28]发现，在训练集和测试集之间存在明显的词汇重叠的情况下，采用词汇分布式表示进行词汇-语义关系判别的有监督模型容易出现过度拟合。在这种情况下，模型倾向于学习单个单词的属性(例如，一个单词是典型的上位词)，而不是单词之间的关系。因此，这些数据集的测试结果是对模型真实性能过于乐观的估计。

为了减小词汇记忆性的影响，Levy 等提出将数据集进行拆分，在训练集、测试集之间都不存在词汇重叠。然而，模型在词汇分割设置中的性能是对模型真实性能过于悲观的估计，在一个现实场景中，模型可能会预测一些与训练集词汇有重叠的词对。真正的模型性能可能介于随机分割的性能和词汇分割的数据集的性能之间，因此我们在两种分割方式中测试了性能。

表 1 列出了实验数据集的大小。此外，在每个数据集中具有上下位关系的词对的比例大概是 20%。

表 1 数据集

Table 1 Datasets

Datasets	Train	Val.	Test
HypeNet(Rnd)	49475	3534	17670
HypeNet(Lex)	20335	1350	6610

4.2 基线模型

本文将基于混合专家的局部模型与下列传统全局模型进行比较。1)Concat 模型[20]，利用词对的词向量的拼接 $\boldsymbol{x}\oplus\boldsymbol{y}$ 作为特征，经过两层的 MLP 模型，隐层激活函数为 ReLU 函数，隐层单元数为d_e。2)Diff 模型[20]，利用词对的词向量的差 $\boldsymbol{y}-\boldsymbol{x}$ 作为特征，经过两层的 MLP 模型，隐层激活函数为 ReLU 函数，隐层单元数为$\lfloor d_e/2\rfloor$。3)HypeNET Integrated 模型[13]，集成了依存路径以及词向量信息，利用 LSTM 编码连接上下位词对的路径，将下位词词向量、路径编码、上位词词向量拼接作为特征，经过一个单层的网络进行二分类。4)DUAL-T 模型[8]，利用一对张量将上下位通用词向量映射为特定词向量，建模上下位词的非对称性，最后基于特定词向量的双线性点积得到上下位关系的得分。

4.3 实验设置

本文使用了 300 维的 FastText 词向量[29]。在研究上下位关系时，研究人员关于词向量是否做归一化有不同的看法[8,30]。我们对上下位词模长的差值进行了统计，统计结果如图 4 所示。

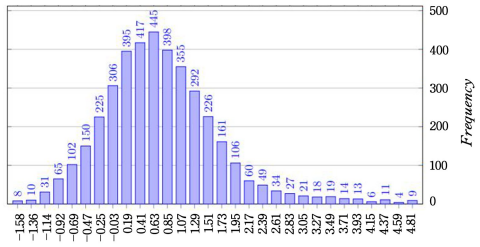


图 4 下 $\|\boldsymbol{y}\|-\|\boldsymbol{x}\|$ 直方图

Fig. 4 $\|\boldsymbol{y}\|-\|\boldsymbol{x}\|$ histogram

我们发现，75.6%的下位词都长于上位词，可能的原因是下位词更加具体，在语料中出现频次低，模长偏长，而上位词更加通用，在语料中出现频次高，模长偏短。另外，我们发现，对于下位词模长短于上位词模长的词对，很多是需要特定的知识或者是作为特定实体才会判定为上下位词，如在(ambition, album)词对中，词语“ambition”在多数场景下并不以专辑的名称出现。除这类词对外，模长信息对于上下位关系判别任务具有重要意义，因此不对模长进行归一化处理。

模型优化使用了 AdamW 算法[31]，初始学习速率设置为 1×10^{-3}，批大小为 64。局部专家数量 M 是依据验证集在范围$\{2^i\}_{1\leqslant i\leqslant 6}$内的网格搜索来确定的，最终实验采用了 $M=16$。

我们运行算法 10 次，同时记录平均的精确率(P)、召回率(R)和 f1 分数(F1)，计算式如下：

$$P=\frac{TP}{TP+FP} \tag{9}$$

$$R=\frac{TP}{TP+FN} \tag{10}$$

$$F1=\frac{2\times P\times R}{P+R} \tag{11}$$

其中，TP 表示正确预测为正例的样本数，TF 表示正确预测为负例的样本数，FP 为错误预测为正例的样本数，FN 表示错误预测为负例的样本数。

本文所有实验均采用表 2 所列的实验环境。

表 2　实验环境

Table 2　Experimental environment

操作系统	Ubuntu16.04
CPU	AMD Ryzen Threadripper 1950X 16-Core Processor
GPU	GTX 1080Ti-11GB
Python	3.7.0
Pytorch	1.8.1
RAM	64 GB

4.4　实验结果比较

表 3 列出了将基于混合专家的上下位关系判别模型(C-MoE，D-MoE)与 Concat 模型[20]、Diff 模型[20]、HypeNET Integrated[13](HypeIn)以及 DUAL-T[8]模型的效果进行比较的结果。C-MoE 模型达到了最优的效果，此外，C-MoE 相比 Concat 模型[20]以及 D-MoE 相比 Diff 模型[20]在去除记忆性(Lex)数据集和未去除记忆性(Rnd)数据集上的效果都更优，这说明在切分为多个语义子空间之后，让每个专家只专注于自己局部子空间的上下位关系区分，对模型的效果有显著提升。C-MoE 和 Concat 在未去除记忆性(Rnd)数据集上基本持平，可能的原因是以 $concat(\boldsymbol{x}\oplus\boldsymbol{y})$ 为特征的模型严重受词汇记忆性的影响，在训练模型时只是单纯记住了某个词总是为上位词或者下位词。而以 $concat(\boldsymbol{x}\oplus\boldsymbol{y})$ 为特征的模型的效果优于以 $diff(\boldsymbol{y}-\boldsymbol{x})$ 为特征的模型，在一定程度上是因为以 $concat(\boldsymbol{x}\oplus\boldsymbol{y})$ 为特征的模型很少关注两个词向量之间的交互，而更多地关注单个词向量，容易受词汇记忆性的影响。

表 3　不同模型的对比实验结果

Table 3　Experimental results comparison between our model and baseline models

Method	Lex			Rnd		
	P	R	$F1$	P	R	$F1$
HypeIn[13]	80.9	61.7	70.0	91.3	**89.0**	90.1
DUAL-T[8]	70.5	78.5	74.3	**93.3**	82.6	87.6
Concat[20]	**81.1**	80.5	80.8	92.7	**88.9**	**90.8**
C-MoE	80.8	**83.8**	**82.2**	**92.9**	88.8	**90.8**
Diff[20]	**81.3**	73.9	77.4	**92.4**	85.6	88.8
D-MoE	81.0	**80.2**	**80.6**	91.9	**88.2**	**90.0**

4.5　超参数分析

专家数量在本文模型中是重要的超参数，本文研究了在不同专家数下模型的表现。

为了减轻词汇记忆性对模型的影响，超参数分析是在去除记忆性(Lex)数据集上进行的。图 5 给出了在测试集上不同专家数对结果的影响，可以发现，C-MoE 和 D-MoE 都在专家数量 $M=16$ 时达到最优。当专家数较小时，模型结果较差，随着专家数的增加，模型效果上升，专家数增加到一定程度时，模型效果又开始下降。可能的原因是，当专家数较少时，子空间中样本的分布仍然比较复杂，判别器难以区分，而专家数过多时，每个专家所管控的样本数又太少，导致专家的泛化性能差，因此最后结果下降。

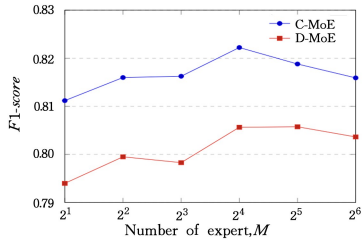


图 5　不同的专家数对结果的影响

Fig. 5　Effect of different number of experts on results

4.6　算法性能分析

本节以 C-MoE 模型为例，分析模型的空间复杂度和时间复杂度。其中，M 表示专家数，d_e 表示词向量的维度。

4.6.1　空间复杂度

空间复杂度包括两部分：总参数量和各层输出特征。单个专家由两个全连接层构成，其参数量为 $(2\ d_e\times d_e+d_e)+(d_e\times 1+1)=2\ d_e^2+2\ d_e+1$，专家各层输出特征为 d_e+1；门控参数量为 $d_e\times M$，门控输出特征为 M；而空间复杂度 $=M\times$(单个专家参数量＋单个专家各层输出特征)＋门控参数量＋门控输出特征 $=O(Md_e^2)$，因此 C-MoE 模型的空间复杂度与专家数成线性关系，与词向量维度成二次关系。

4.6.2　时间复杂度

利用浮点运算次数来衡量模型的时间复杂度，模型时间复杂度拆分为 M 个单个专家、门控、加权求和 3 部分的浮点运算次数，分别记为 $Time_{单个专家}$，$Time_{门控}$，$Time_{加权求和}$。$Time_{单个专家}=(2\ d_e+2\ d_e)\times d_e+(d_e+d_e)\times 1=4\ d_e^2+2\ d_e$，$Time_{门控}=(d_e+d_e-1)\times M$，$Time_{加权求和}=M+M-1$，模型的时间复杂度 $=M\times Time_{单个专家}+Time_{门控}+Time_{加权求和}=O(Md_e^2)$。C-MoE 模型的时间复杂度与专家数成线性关系，与词向量维度成二次关系。

4.7　动态专家选择

不同的词对所在的局部子空间不同，因此采用门控机制实现了专家动态混合。为了验证动态选择局部专家的必要性，本文模拟两种静态选择专家的方式，具体做法是将门控机制移除，局部专家分别用固定权重集成(C-W，D-W)以及等权重(C-Sum，D-Sum)的方式集成。固定权重集成是在决策层使用参数权重矩阵将专家判定结果(训练可调，测试固定)融合在一起，参数矩阵的维度为 $M\times 1$。在测试阶段每个样本对专家的选择是相同的，如式(12)所示：

$$s(x,y)=\boldsymbol{w}^{\mathrm{T}}\cdot\boldsymbol{h} \tag{12}$$

等权重集成是将专家判定结果直接相加，作为最终判定结果，如式(13)所示：

$$s(x,y)=sum(\boldsymbol{h}) \tag{13}$$

从表 4 可以看到，固定权重集成和等权集成都使得模型的效果下降，说明多个全局专家模型的简单集成并不能带来模型效果的提升，而将语义空间拆分成多个子空间，不同局部专家管控不同子空间内的样本，才能更好地拟合样本分布，提升模型效果。

表 4 动态专家选择与静态专家集成结果

Table 4 Dynamic expert selection and static expert integration experiments

Method	Lex			Rnd		
	P	*R*	*F*1	*P*	*R*	*F*1
C-W	79.9	82.7	81.2	92.7	**88.8**	90.7
C-Sum	78.5	80.4	79.3	**93.0**	88.5	90.7
C-MoE	**80.8**	**83.8**	**82.2**	92.9	**88.8**	**90.8**
D-W	80.4	71.5	75.6	**92.1**	85.7	88.8
D-Sum	79.0	72.8	75.7	91.5	85.5	88.4
D-MoE	**81.0**	**80.2**	**80.6**	91.9	**88.2**	**90.0**

对比固定权重集成和等权集成，前者的效果略优于后者。这说明不同专家的判别效果不同，在判别时表现好的专家取更大的权重，模型表现更优。但固定权重集成的效果仍然差于以混合专家方式集成专家，说明了表现好的全局专家并不适用于所有样本，还是需要根据样本所在的子空间选择适当的专家适配。

4.8 错误分析

为了进一步分析产生错误的原因，我们在 C-MoE 模型预测的测试集中截选了一部分错误的样例，如表 5 所列。

表 5 错误样例

Table 5 Error samples

word1	word2
room	novel
dominion	novel
oblivion	orphanage
trauma	fracture
metro	novel
norman	soprano
merrick	place
cosmos	flower
fishbone	band
bell	typeface

(1)表 5 上半部分是错误预测为上下位关系的样例，其中关于 novel 的词对都预测为上下位关系，可能原因是模型只记住了 novel 为典型上位词，这也体现了以两个词向量作为输入特征，模型受词汇记忆性的影响会非常严重。

(2)表 5 下半部分是错误预测为非上下位关系的样例，由于上下位词大都是名词，而名词受到词汇多义性的影响最大，尤其是以罕见语义出现时，常湮没在常见语义中，使得判断困难。例如，在词对(fishbone，band)中，只有“fishbone”作为“鱼骨头乐队”时，上下位关系才成立。

结束语 上下位关系是一种重要的词汇语义关系，被广泛应用于文本蕴含推理、自动问答等下游任务中。本文针对在全局模型中区分词语上下位关系较为困难的问题，提出了一种基于混合专家的上下位关系判别方法，实验结果证明了本文模型的有效性。

在错误样例的启发下，在未来的研究中，我们可以参照如下思路进行进一步的研究：1)构建不受词汇记忆性影响的模型，并将混合专家的思想应用到模型上，通过消除记忆性影响得到更好的结果；2)通过考虑词汇的多义性(如采用多义词向量)，增强词向量的表示，从而达到更好的效果。

参 考 文 献

[1] NAVIGLI R, VELARDI P, FARALLI S. A graph-based algorithm for inducing lexical taxonomies from scratch[C]// Twenty-Second International Joint Conference on Artificial Intelligence. Barcelona: IJCAI/AAAI, 2011: 1872-1877.

[2] LAN Y, JIANG J. Embedding WordNet knowledge for textual entailment[C]// Proceedings of the 27th International Conference on Computational Linguistics: Santa Fe. New Mexico: ACL, 2018: 270-281.

[3] CHEN Q, ZHU X, LING Z H, et al. Neural natural language inference models enhanced with external knowledge[C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne. Australia: ACL, 2018: 2406-2417.

[4] HUANG Z, THINT M, QIN Z. Question classification using head words and their hypernyms[C]// Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Honolulu: ACL, 2008: 927-936.

[5] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[C]// 1st International Conference on Learning Representations. Scottsdale: ICLR, 2013: Workshop Poster.

[6] BARONI M, BERNARDI R, DO N Q, et al. Entailment above the word level in distributional semantics[C]// Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics. Avignon: ACL, 2012: 23-32.

[7] ROLLER S, ERK K, BOLEDA G. Inclusive yet selective: Supervised distributional hypernymy detection[C]// Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. Dublin: ACL, 2014: 1025-1036.

[8] GLAVAŠ G, PONZETTO S P. Dual tensor model for detecting asymmetric lexico-semantic relations[C]// Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen: ACL, 2017: 1757-1767.

[9] REI M, GERZ D, VULIĆ I. Scoring lexical entailment with a supervised directional similarity network[C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Volume 2: Short Papers. Melbourne: ACL, 2018: 638-643.

[10] MILLER G A. WordNet: a lexical database for English [J]. Communications of the ACM, 1995, 38(11): 39-41.

[11] HEARST M A. Automatic acquisition of hyponyms from large text corpora[C]// The 15th International Conference on Computational Linguistics. Nantes: ACL, 1992: 539-545.

[12] KOZAREVA Z, HOVY E. A semi-supervised method to learn and construct taxonomies using the web[C]// Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Massachusetts: ACL, 2010: 1110-1118.

[13] SHWARTZ V, GOLDBERG Y, DAGAN I. Improving hypernymy detection with an integrated path-based and distributional method[C]// Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: ACL, 2016: 2389-2398.

[14] SHI Y, SHEN J, LI Y, et al. Discovering hypernymy in text-rich heterogeneous information network by exploiting context granu-

larity[C]// Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing: ACM, 2019: 599-608.

[15] HARRIS Z S. Distributional structure[J]. Word, 1954, 10(2/3): 146-162.

[16] WEEDS J, WEIR D. A general framework for distributional similarity[C]// Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. Sapporo: ACL, 2003: 81-88.

[17] KOTLERMAN L, DAGAN I, SZPEKTOR I, et al. Directional distributional similarity for lexical inference[J]. Natural Language Engineering, 2010, 16(4): 359-389.

[18] SHWARTZ V, SANTUS E, SCHLECHTWEG D. Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection[C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Valencia: EACL, 2016: 65-75.

[19] CLARKE D. Context-theoretic semantics for natural language: an overview[C]// Proceedings of the Workshop on Geometrical Models of Natural Language Semantics. 2009: 112-119.

[20] TURNEY P D, MOHAMMAD S M. Experiments with three approaches to recognizing lexical entailment[J]. Natural Language Engineering, 2015, 21(3): 437-476.

[21] WEEDS J, CLARKE D, REFFIN J, et al. Learning to distinguish hypernyms and co-hyponyms[C]// Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics. Dublin: ACL, 2014: 2249-2259.

[22] FU R, GUO J, QIN B, et al. Learning semantic hierarchies via word embeddings[C]// Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Baltimore: ACL, 2014: 1199-1209.

[23] NGUYEN K A, KÖPER M, WALDE S S, et al. Hierarchical embeddings for hypernymy detection and directionality[C]// Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen: ACL, 2017: 233-243.

[24] DASH S, CHOWDHURY M F M, GLIOZZO A, et al. Hypernym detection using strict partial order networks[C]// Proceedings of the AAAI Conference on Artificial Intelligence. Online: AAAI, 2020: 7626-7633.

[25] XIE Z, ZENG N. A Mixture-of-Experts Model for Antonym-Synonym Discrimination[C]// Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). Online: ACL, 2021: 558-564.

[26] VAN DER MAATEN L, HINTON G. Visualizing data using t-SNE[J]. Journal of Machine Learning Research, 2008, 9(11): 2579-2605.

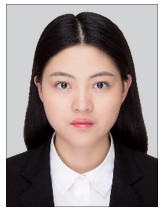
[27] SHAZEER N, MIRHOSEINI A, MAZIARZ K, et al. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer[C]// 5th International Conference on Learning Representations. Toulon: OpenReview. net, 2017: 1-29.

[28] LEVY O, REMUS S, BIEMANN C, et al. Do supervised distributional methods really learn lexical inference relations? [C]// Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Denver: ACL, 2015: 970-976.

[29] BOJANOWSKI P, GRAVE E, JOULIN A, et al. Enriching word vectors with subword information[J]. Transactions of the Association for Computational Linguistics, 2017, 5: 135-146.

[30] WANG C, HE X. Birre: learning bidirectional residual relation embeddings for supervised hypernymy detection[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: ACL, 2020: 3630-3640.

[31] LOSHCHILOV I, HUTTER F. Fixing weight decay regularization in adam[J/OL]. CoRR, 2017, abs/1711. 05101: 1-14. https://www.doc88.com/p-9029673865620.html.



ZENG Nan, born in 1997, postgraduate. Her main research interests include nature language processing and lexical-semantic relation.

XIE Zhipeng, born in 1976, Ph.D, associate professor, Ph.D supervisor, is a member of China Computer Federation. His main research interests include data mining, machine learning and natural language processing.

（责任编辑：喻藜）