一种基于成对字向量和噪声鲁棒学习的同义词挖掘算法

张浩宇 [1, 2] 王 戟 [1]

摘 要 同义词挖掘是自然语言处理中一项重要任务. 为了构建大规模训练语料, 现有研究利用远程监督、点击图筛选等方式抽取同义词种子, 而这几种方式都不可避免地引入了噪声标签, 从而影响高质量同义词挖掘模型的训练. 此外, 由于大量实体词所具有的少样本特性、领域分布差异性和预训练词向量训练目标与同义词挖掘任务的不一致性, 在同义词挖掘任务中, 词级别的预训练词向量很难产生高质量的实体语义表示. 为解决这两个问题, 提出了一种利用成对字向量和噪声鲁棒学习框架的同义词挖掘模型. 模型利用预训练的成对字向量增强实体语义表示, 并利用自动标注的噪声标签通过交替优化的方式, 估计真实标签的分布并产生伪标签, 希望通过这些改进提升模型的表示能力和鲁棒性. 最后, 使用 WordNet 分析和过滤带噪声数据集, 并在不同规模、不同领域的同义词数据集上进行了实验验证. 实验结果和分析表明, 该同义词挖掘模型在各种数据分布和噪声比例下, 与有竞争力的基准方法相比, 均提升了同义词判别和同义词集合生成的效果.

关键词 同义词挖掘, 噪声标签学习, 自然语言处理, 成对字向量, 信息抽取

引用格式 张浩宇, 王戟. 一种基于成对字向量和噪声鲁棒学习的同义词挖掘算法. 自动化学报, 2023, **49**(6): 1181−1194

DOI 10.16383/j.aas.c210004

A Synonym Mining Algorithm Based on Pair-wise Character Embedding and Noisy Robust Learning

ZHANG Hao-Yu[1, 2] WANG Ji[1]

Abstract Synonym mining is an important task in natural language processing. In order to construct large-scale training corpus, existing studies extract synonym seeds using distant supervision and click graph filtering, which inevitably introduce noisy labels, thus affecting the training of high-quality synonym mining models. In addition, due to the few-shot and domain-distribution-shift property of most entity words, and the inconsistency between the training objective of the pre-trained word embeddings and the synonym mining task, it is difficult for the pre-trained word embeddings in the synonym mining task to produce high-quality entity semantic representations. To address these two issues, this paper proposes a synonym mining model that utilizes pair-wise character embeddings and a noise robust learning framework. The model uses pre-trained pair-wise character embeddings to enhance the entity semantic representations, estimate true label distribution and generate pseudo-labels through a joint optimization process. We want to improve the representation ability and robustness of the model through these improvements. Finally, we use WordNet to analyze and filter noisy datasets and conduct the experiments on synonym datasets of different sizes and domains. The experimental results show that the proposed synonym mining model improves the synonym set-instance classification and set generation performances compared to competitive benchmark methods under different data distribution and noise ratios.

Key words Synonym mining, noisy label learning, natural language processing, pair-wise character embedding, information extraction

Citation Zhang Hao-Yu, Wang Ji. A synonym mining algorithm based on pair-wise character embedding and noisy learning. *Acta Automatica Sinica*, 2023, **49**(6): 1181−1194

实体词一般用来表示客观存在并可以相互区别的事物, 如人名、机构名、地名等专有名词或有意义的时间等. 挖掘实体词间的同义词、反义词、上/下位词等词汇关系, 对于计算机有效理解词汇语义十分重要. 在这些实体词词汇关系中, 同义词指意义相同或相近的实体词, 其主要特征是词汇在语义上相同或相似. 同义词挖掘是自然语言处理领域一项重要任务, 它的目标是从无结构文本中识别出所

收稿日期 2021-01-04 录用日期 2021-06-06
Manuscript received January 4, 2021; accepted June 6, 2021
国家重点研发计划 (2017YFB1001802), 国家自然科学基金 (91948303, 62032024) 资助
Supported by National Key R & D Program (2017YFB1001802) and National Natural Science Foundation of China (91948303, 62032024)
本文责任编委 刘洋
Recommended by Associate Editor LIU Yang
1. 国防科技大学高性能计算国家重点实验室 长沙 410072 2. 军事科学院国防科技创新研究院人工智能研究中心 北京 100071
1. State Key Laboratory of High Performance Computing, National University of Defense Technology, Changsha 410072 2. Artificial Intelligence Research Center, Defense Innovation Institute, Beijing 100071

有的实体同义词, 能为很多下游与实体相关的任务(如知识库补全、知识库问答、实体分类、实体链接、搜索扩充、文档摘要等)[1−5] 提供有用的信息. 近年来, 基于深度神经网络的方法在同义词挖掘研究中取得了不错效果, 但在领域特定数据集上, 受限于带标签训练数据的不足. 为了解决这个问题, 已有的研究尝试从无结构化文本中提取一些训练种子, 其中比较典型的方式有通过实体链接的方式利用已有的知识图谱进行远程监督构造数据[6] 和基于搜索引擎、电子邮件等场景下搜索点击图筛选构造训练数据[7−8]. 远程监督是利用已有数据构建同义词挖掘数据集的一类重要和常见方法, 很多同义词挖掘领域的基准数据集是利用远程监督进行构建的[6, 9].

尽管远程监督方法能够有效地构造出大规模的领域带标签训练数据, 但是这些数据中包含着一定比例的噪声标签, 在一定程度上影响了同义词挖掘模型的学习效果, 也是亟待解决的问题. 在解决此类问题时, 噪声学习是当前研究中较为主流的一种研究思路. 噪声学习主要研究如何利用鲁棒的模型设计、目标函数设计以及先验知识减少包含噪音的标签在训练时对深度神经网络的影响. 近期, 在计算机视觉、自然语言处理等领域出现了一些研究噪声对神经网络学习过程影响的分析以及噪声鲁棒学习方法的研究[10−12]. 例如, 基于噪声学习的方法在同样使用远程监督构造大规模监督数据的细粒度实体识别和关系抽取领域被广泛应用[12−14]. 而在同义词挖掘领域, 远程监督构造数据集的方法也被广泛应用, 且至少在以下两个过程中会引入噪声标签: 1) 在实体链接的过程中, 由于实体链接器的错误, 而引入标签噪声; 2) 知识库本身的节点同义词信息包含错误. 尽管存在着这样的噪声标签问题, 但已有的同义词挖掘方法往往忽略了该问题或只专注于解决如何在远程监督构建数据过程中去噪[6], 较少关注如何在学习过程中去噪.

另一方面, 实体词语义表示学习也是同义词挖掘的一个难点. 实体词尤其是领域实体词具有稀疏性质, 在大规模通用语料中出现次数很少. 预训练词向量的目标是最大化每个单词与其近邻单词的条件概率, 词汇语义关系判断只是其副产物. 例如, Word2Vec 等工作[15−16] 捕获了词汇间的相似性关系, “北京”和“广州”的表示向量较为相似, 但这类预训练词向量无法直接有效应用到同义词相似度判别. Fei 等[17] 研究实验结果也表明, 词级别的预训练词向量在实体同义词挖掘任务上的表示能力受到一定的限制. 为了解决词表示能力的不足, 现有的很多同义词挖掘研究[6−7, 18] 选择挖掘各种统计特征增强词汇表示 (例如语义标签、上/下文模板、共现频率等), 但较少有从语义表示向量角度进行的探索.

为了解决上述两类问题, 本文提出一种结合了噪声鲁棒学习框架和成对字表示向量的方法. 该方法分别利用成对字向量来增强实体词的表示能力, 利用噪声学习框架缓解噪声标签的影响. 在实验过程中, 利用 WordNet[19] 对含噪声数据集进行了分析和过滤. 本文的主要贡献如下:

1) 利用 WordNet 对远程监督得到的数据进行噪声比例和噪声特点分析, 并过滤了测试集中的噪声, 以得到更加准确的评判结果. 针对性地缓解了之前同义词挖掘基准数据集本身的质量问题, 同时统计了与 WordNet 中的同义词数据相比, 多个基准数据集的噪声数据比例.

2) 引入了成对字向量和噪声学习的框架, 针对当前的同义词集合生成方法的两类问题 (实体词的表示问题和对噪声标签的鲁棒性不足问题) 进行了针对性改进. 针对实体词的表示问题, 对成对词向量方法进行了字符级别的扩展; 针对对噪声标签的鲁棒性不足问题, 设计了一种“估计−矫正”框架, 进行伪标签分布的学习与矫正.

3) 通过在 3 个不同规模、不同领域数据集上进行的实验, 对模型效果进行了定性和定量分析比较. 实验分析和结果表明, 本文方法能够有效提升在各个规模、各个领域的效果. 进一步的辅助实验分析表明, 本文提出的改进方法能够有效地学习到更好的实体词语义表示, 并在不同比例的噪声训练数据集上, 具有更高的鲁棒性.

1 噪声数据下的同义词集合生成模型

本文提出一种在噪声数据下能够有效学习的同义词分类模型 NL-P2V (Noisy learning-pair to vector), 它由基础模型、成对字向量表示的增强模块和噪声学习框架组成, 整体结构如图 1 所示. 图 1 中左半部分为第 1.3 节中融合的成对字向量模块; 右半部分为基准模型结构, FC 代表全连接层.

1.1 问题定义和任务描述

本文任务是在利用知识图谱, 以远程监督方式构建的实体同义词数据集上, 通过对模型和算法的改进, 提升实体同义词集合生成的效果. 其中, 知识图谱是由一系列 <头实体, 关系, 尾实体> 三元组构成的关系网络. 在实体同义词挖掘任务中, 远程监督方法通过将无标记语料中的实体指称识别, 并链接到知识图谱中对应节点的方式, 对齐多个实体

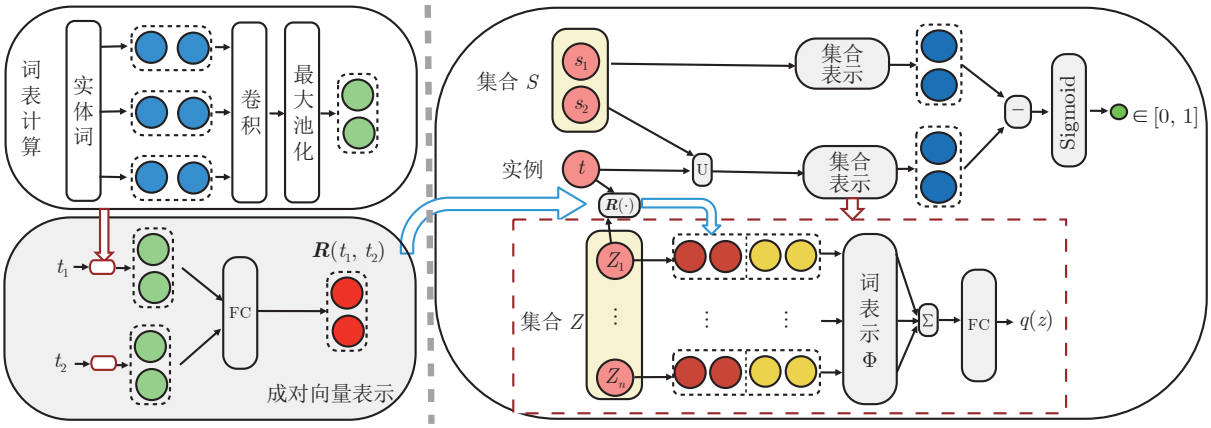


图 1　模型结构图

Fig. 1　The model architecture

指称, 并将这些实体词归入同一个同义词集合.

本文方法的基础是文献 [9] 提出的一种集合−实例分类器的结构. 这种方法将同义词挖掘形式化为基于集合表示学习的二分类任务, 并利用知识图谱以远程监督的方式构造训练数据. 在本文实验中, 直接使用文献 [9] 的远程监督构建的数据.

1) 基于知识图谱远程监督的同义词集合生成. 一个同义词集合 $S = \{S_1, \cdots, S_{|s|}\}$ 是指一个单词或短语的集合, 其中的每个元素都对应着同一个现实世界的实体. 给定一个无结构化文本语料 $\mathcal{D}$、从中抽取出的实体词库 $\mathcal{V}$ 和一个知识图谱 $\mathcal{G}$, 则知识图谱远程监督下的同义词集合生成任务的目标是根据从 $\mathcal{D}$ 和 $\mathcal{G}$ 中抽取的信息, 将 $\mathcal{V}$ 中的所有实体词聚类为若干个同义词集合.

2) 集合−实例同义词分类. 同义词集合生成任务可以被分解为若干个集合−实例同义词分类任务. 后者的输入是一个已有的同义词集合 S 和一个实体词库中的实体词 t, 分类器输出该实体词属于同义词集合同义词的概率 $p(S, t) \in [0, 1]$.

1.2　基础方法

本文进行的两项改进与基础模型结构无关, 能够扩展到其他模型结构上.

1.2.1　分类器结构

集合−实例同义词分类器的一个性质就是顺序无关性, 即分类的结果不受集合内元素顺序的影响. 为了保证顺序无关性, 集合−实例分类器的实现方式有以下两种: 1) 分别计算集合中的元素和实例的相似度, 并进行信息聚合. 但这种方式忽略了集合内元素互相之间的联系. 2) 借鉴集合表示学习的思想[20], 使用一个集合表示学习模块, 来输出集合 Z 的表示得分 $q(Z) \in \mathbf{R}$, 可以认为这个得分体现了该集合中元素的内聚程度和完整度.

使用的分类器在集合表示模块的基础上, 首先计算输入集合 S 的表示得分 $q(S)$, 然后将实例单词 t 加入到 S 中, 得到一个新的集合 $S \cup t$, 最后对比这 2 个得分, 并通过激活函数计算最终 t 属于集合 S 的同义词输出概率:

$$p(S, t) = \sigma(q(S \cup t) - q(S)) \tag{1}$$

1.2.2　集合表示学习模块

模型中的集合表示学习模块的过程如下: 首先, 给定一个具有 n 个词的集合 $Z = \{z_1, \cdots, z_i, \cdots, z_n\}$, 经过词向量层, 将集合中每个实体词 z_i 映射为其对应的词向量 $x_i \in \mathbf{R}^{d_w}$; 然后, 该词向量表示经过一个两层全连接网络得到词表示 $\{\Phi(x_1), \cdots, \Phi(x_n)\}$, $\Phi(x_n) \in \mathbf{R}^{d_p}$; 接着, 将所有词表示经过向量求和得到初始集合表示 $v(Z) = \sum_{i=1}^{n} \Phi(x_i) \in \mathbf{R}^{d_s}$, 由于向量求和是一个具有顺序不变性的操作, 因此整个集合的表示生成过程也保证了顺序不变的性质; 最后, 由一个三层全连接网络将集合的向量表示转换为集合得分 $q(Z) = g(v(Z))$. 由于本文主要研究噪声学习策略和成对字表示的增强效果, 因此直接使用了与文献 [9] 相同的基础结构. 以图卷积神经网络[21]为代表的图神经网络的聚合操作, 对于中心节点的邻域是顺序无关的, 具备置换不变性.

1) 训练. 首先给定 N 个集合−实例−标签三元组 $< S^i, t^i, y^i >|_{i=1}^{N}$ 作为训练数据, 其中 $y^i \in \{0, 1\}$. 由具有参数 θ 的分类器给出的分类概率为 $p(S^i, t^i)$, 则使用最大化对数似然概率的方式进行参数更新, 即如式 (2) 所示的损失函数. 在训练时, 从构成训练集的同义词集合 $S^1, S^2, \cdots, S^{|s|}$ 中构造正样本, 即对于一个训练集合 S^k, 随机选择其中一个

子集 $\hat{S}^k$ 和一个不属于该子集中的元素 $t^k \in S^k - \hat{S}^k$ 构造三元组 $< \hat{S}^k, t^k, 1 >$. 然后从所有训练词库 $\mathcal{V}-S^k$ 中不在当前集合的词中, 随机采样 K 个词作为负样本.

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} -y^i \log_2(p(S^i, t^i)) \tag{2}$$

2) 集合生成. 在利用集合−实例分类器进行集合生成的过程中, 利用贪心策略即当某一时刻有 k 个集合 $\{S^1, \cdots, S^k\}$ 时, 遍历 $\mathcal{V}$ 中尚未被选择到任何一个集合中的单词, 最终获得与其中某一集合 S^i 相似度最高的单词 t, 通过判断 $p(s^i, t)$ 是否超过预先设定的分类阈值, 来决定将 t 放入集合 S^i 中或单独构成一个新的同义词集合 S^{k+1}. 该过程由任意选择某个单词构成初始集合 S^1 开始, 到所有 $\mathcal{V}$ 中的单词均被选入了某个集合结束.

1.3 成对字向量表示增强

文献 [17, 22] 研究表明, 仅使用预训练的词向量作为同义词表示往往不足以表达出实体词的完整语义, 原因有以下 2 点: 1) 通常使用跳字模型或连续词袋模型[15, 23] 等范式训练的预训练词向量的训练目标, 是最大化每个单词与其近邻单词的条件概率. 而同义词关系判别要求根据单词或单词与其上/下文最小化同义词间的距离且最大化非同义词间的距离, 这两个任务目标之间存在偏移. 2) 由于实体词 (特别是领域数据中的实体词) 具有稀疏特性且可能出现语义偏移, 使用以词为单元的语义表示并直接在通用领域预训练的词向量, 可能受限于目标词出现的频率. 基于卷积神经网络、循环神经网络、基于自注意力的变换器网络等模型的字级别或子词级别词汇表示模块, 在包括机器翻译[24]、语言模型[25]、问答系统[26] 等方向, 已经得到了较为广泛应用, 展示出了在各类任务中, 字/子词级别的词汇表示对词向量建模能力的提升效果.

因此, 根据文献 [27] 的成对词向量思想, 并将其推广到字级别. 本文提出一种利用成对向量表示学习的思想, 在字级别上对实体表示进行增强的方法.

成对字向量模块的输入是两个实体词 $t_1 = \{t_{11}, t_{12}, \cdots, t_{1i}, \cdots, t_{1m_1}\}$、$t_2 = \{t_{21}, t_{22}, \cdots, t_{2j}, \cdots, t_{2m_2}\}$ 和一个上/下文句子 $c = \{c_1, \cdots, c_i, \cdots, c_n\}$, 其中 t_{1i} 和 t_{2j} 是构成实体词的每个字符, m_1 和 m_2 分别是构成两个实体词的字符长度, c_i 是上/下文中的一个单词. 在应用到同义词集合生成模型时, 成对字向量的输出为特征向量 $\boldsymbol{R}(t_1, t_2)$, 其表达的语义是两个实体词之间的相似程度.

1.3.1 成对字向量模块结构

$$x = \frac{E(t_1)}{||E(t_1)||}, \qquad y = \frac{E(t_2)}{||E(t_2)||} \tag{3}$$

$$\boldsymbol{R}(x, y) = MLP^4[x; y; x \odot y] \tag{4}$$

首先, 两个单词中的每个字符 t_{ij} 会根据预先抽取的字符词典 $\mathcal{V}_c$ 以及对应的字向量矩阵映射到字向量 $\boldsymbol{e}_{ij} \in \mathbf{R}^{d_c}$, 其中 d_c 是字向量的维度. 而后两个字向量列表被输入一个具有 d_v 个卷积核的二维卷积层, 其中每个卷积核的二维卷积窗口大小为 $d_c \times w_c$, 卷积将抽取窗口 w_c 范围内的局部信息得到上/下文感知的字表示. 对字向量 $\boldsymbol{e}_{ij}$ 来说, 首先将得到向量表示 $\hat{\boldsymbol{e}}_{ij} \in \mathbf{R}^{d_v}$. 然后, 卷积层的输出被输入一个最大池化层, 并得到两个单词的字级别表示 $E(t_1) \in \mathbf{R}^{d_v}$ 和 $E(t_2) \in \mathbf{R}^{d_v}$. 接着, 经过归一化得到两个输入单词的最终表示 x 和 y. 这两个单词表示被输入到多层全连接网络, 并得到两个单词间的相似度表征 $\boldsymbol{R}$. 式 (4) 中, $[\cdot;\cdot]$ 表示向量拼接, $\odot$ 表示按元素乘, MLP 表示具有一个隐藏层的全连接神经网络. 最后, 使用双向长短时记忆网络 (Long short term memory, LSTM) 对上/下文 c 的词向量列表进行编码, 并使用注意力池化进行聚合, 得到上/下文的向量表示 $\boldsymbol{C}(c)$, 其中 $\boldsymbol{W}$、$\boldsymbol{h}$ 是参数矩阵和参数向量:

$$c_i = E_w(c_i) \tag{5}$$

$$\{h_1, \cdots, h_n\} = BiLSTM(\{c_1, \cdots, c_n\}) \tag{6}$$

$$w_i = \frac{\mathrm{e}^{kh_i}}{\sum\limits_{j=1}^{n} \mathrm{e}^{kh_j}} \tag{7}$$

$$\boldsymbol{C}(c) = \sum_{i=1}^{n} w_i \boldsymbol{W} \boldsymbol{h}_i \tag{8}$$

成对字向量的输入是单词中的字符序列, 由卷积层、最大池化以及归一化模块进行处理后得到该单词的上/下文感知的字表示, 对于顺序不同的字符序列, 由卷积层保证获得的语义向量输出差异.

1.3.2 预训练

使用第 1.3.1 节介绍的结构来计算成对字向量, 并进行无监督式的预训练. 在预训练时, 原始的 P2V (Pair to vector) 是利用语料中的词对共现信息来构造正样本. 对于正样本 $< t_1, t_2, c >$ 尽量增大其表示向量 $\boldsymbol{R}(t_1, t_2)$ 与 $\boldsymbol{C}(c)$ 的相似度; 对于负样本, 则减少两个向量表示的相似度, 即目标函数为正样本的向量表示的点积与负样本向量表示的负点积之和.

在本文实验中, 抽取 3 个数据集的训练集部分

的同义词对, 从 P2V 预训练所使用的维基百科语料中选择实体词所出现过的句子. 并将每一对实体同义词 t_1、t_2 以及 t_1 所出现的上/下文 c 作为一个正样本, 以随机抽取的 K_p 个非同义词实体 $\bar{t}_n$ 来构成负样本, 则最终预训练时的损失函数为:

$$\mathcal{L}_{P2V} = \log_2 \sigma(\boldsymbol{R}(t_1, t_2) \cdot \boldsymbol{C}(c)) + \sum_{j=1}^{K_p} \log_2 \sigma(-\boldsymbol{R}(t_1, \bar{t}_j) \cdot \boldsymbol{C}(c)) \tag{9}$$

1.3.3　成对字向量增强的同义词分类模型

在预训练完成后, 训练同义词模型时使用成对字向量表示增强. 输入集合 S 和实例 t, 此时使用预训练后的部分成对字向量网络, 包括字级别词表示模块 $E(\cdot)$ 以及词相似度评判模块 $\boldsymbol{R}(\cdot, \cdot)$. 通过计算 S 中每一个元素 s_i 与实例 t 的相似度 $f(s_i, t) = \boldsymbol{R}(E(s_i), E(t)) \in \mathbf{R}^{d_v}$, 并把该表示拼接到原本的实体词表示上作为最终的实体词表示, 即 $[f(s_i, t); x_i] \in \mathbf{R}^{d_p+d_v}$. 利用预训练的成对字向量, 缓解了词向量目标函数与任务不一致和无法解决少样本特性 2 个问题. 其中, 成对语义表示 P2V 预训练所起的作用, 主要是缓解词表示目标函数与任务不够一致. 而与成对词向量相比, 成对字向量能够更好地处理词库外词以及稀疏词, 本文在第 2 节将会对此进行实验和评价.

1.4　噪声鲁棒学习

在使用远程监督数据进行训练的过程中, 使用交叉熵损失函数会受到噪声标签的影响, 从而降低模型的预测效果. 虽然文献 [28] 尝试使用标签平滑或置信惩罚这类提升鲁棒性的正则方法来处理, 但对于噪声标签学习来说, 通常效果不够. 本文提出一种基于隐变量判别和交替优化框架的噪声学习模块来解决这一问题.

本文将包含噪声标签的三元组标记为 $< S, t, y >$, 将用来估计正确标签的隐向量标记为 $\hat{y}$.

1) 隐变量分布估计. 在基础模型上, 首先设计一个辅助判别器来估计隐向量 $\hat{y}$, 该模块输出为 $h(S, t) = p'(\hat{y}|S, t) = \sigma(q'(S \cup t) - q'(S))$, 其中 $q'(\cdot)$ 为集合表示学习模块, 与主分类器共享词表示学习的过程. 使用单独的集合表示处理阶段为 $q'(S) = g'(v(S))$. 在训练时, 由于隐变量 $\hat{y}$ 是没有标签信息的, 因此无法完全通过监督训练的方式来训练辅助判别器. 但噪声标签即使包含 15% ~ 25% 的噪声信息, 但仍然具有很多正确的样本类别信息. 利用这一先验, 通过式 (10) 减少辅助分类器损失比率进行训练, 这可以看作是减少学习率, 以防止辅助判别器在训练时过快地拟合噪声标签. 在使用辅助判别器时, 直接利用辅助判别器给出的分类概率, 将其作为对正确标签隐变量的估计.

$$\mathcal{L}_e(\theta') = (1-\alpha) \cdot \mathcal{L}(\theta) + \alpha \cdot MLE(p'(\hat{y}|S, t), y) \tag{10}$$

式中, MLE (Maximum likelihood estimation) 表示极大似然估计.

2) 交替优化框架. 为有效利用估计到的隐变量分布, 引入了一个名为“估计–矫正”交替优化的框架. 将优化步骤 1) 称为交替优化中的“估计”步骤, 其目标是估计正确标签的分布. 而在“矫正”步骤中, 利用当前参数的辅助判别器, 估计到的隐变量分布用如下方式来计算伪标签和损失函数:

$$H(p') = -\sum_{i=1}^{B} p'(\hat{y}^i|\cdot) \log_2 p'(\hat{y}^i|\cdot) \tag{11}$$

$$\bar{y} = (1 - H) \cdot \hat{y} + H \cdot y \tag{12}$$

$$\mathcal{L}_{c_1}(\theta') = MLE(p(y|S, t), \bar{y}) \tag{13}$$

式中, B 是训练时的批次样本数量, $H(\cdot)$ 表示该批次样本的隐变量估计熵. 使用批次熵 (而不是单个样本熵) 来避免预测概率在单个样本上的差异造成训练震荡. $\bar{y}$ 表示由隐变量分布计算出的伪标签, 熵越低, 表示模型当前总体的置信度越高即向隐变量偏移较多.

3) 时间衰减的 KL (Kullback-Leibler) 散度约束. 式 (10) 和式 (13) 的损失函数和优化框架倾向于奖励辅助判别器的高置信度. 但这种形式的“估计–矫正”目标函数在实验中会趋向使隐变量向伪标签靠近, 从而和预测分布 $p(\cdot)$ 趋同. 因此, 度量了预测概率和隐变量概率分布的 KL 散度, 并将其作为一个惩罚项, 修正当前阶段的目标函数为 $\mathcal{L}_c(\theta')$:

$$\mathcal{KL}(p||p') = -\sum_{i=0}^{1} p(y=i) \log_2 \frac{p'(\hat{y}=i)}{p(y=i)} \tag{14}$$

$$\delta = \exp(-\mathcal{KL}(p||p')) \tag{15}$$

$$\mathcal{L}_c(\theta') = (1-t) \cdot \delta \cdot \mathcal{L}_{c_1} + t \cdot \mathcal{L}_{c_1} \tag{16}$$

式中, $t \in [0, 1]$ 表示当前实验进行的阶段在每个训练批次更新. 因为实验中没有使用提前停止策略且训练轮数确定, 因此 t 不是一个超参数. 在最终的目标函数下, 训练的“矫正”阶段会在训练前期惩罚与预测分布过于相似的隐变量估计分布, 从而防止隐变量分布与预测分布趋同.

2　实验与评价

本节通过实验, 验证在同义词挖掘任务中, 与

若干有竞争力的基准方法相比, 在噪声标签假设条件下, 基于噪声学习与成对字向量模型在同义词集合分类和集合生成任务性能上的提升.

此外还设计了多个辅助实验, 来验证本文提出的 2 个改进模块能否有效缓解词语义表示问题与噪音标签问题. 在成对字向量模块方面, 通过消融实验, 验证了在噪声学习模块基础上增加成对字向量模块带来的性能提升, 并通过词表示可视化对比定性分析词表示学习效果. 此外, 通过消融实验, 分析了噪声鲁棒学习模块为基础模型带来的单独性能提升效果, 验证了噪声学习模块在不同噪声比例下的影响. 通过超参数调优实验, 验证了噪声学习模块超参数对性能的影响. 通过这 3 个实验对噪声学习模块的有效性进行定量判断. 本节还引入了中文同义词数据集, 以验证模型在不同语言上的效果.

2.1 数据集与评价指标

与文献 [6, 9] 相同, 本文使用了此领域中通用的 3 个公开基准数据集[1]. 在此基础上, 为了测试在中文语料上的效果, 本文在中文同义词词林语料 (CILIN)[2] 上进行了相应同义词挖掘实验.

数据集 Wiki 是维基百科上导出的数据集, 其中使用 Freebase 作为知识图谱. 训练集和测试集数据从 100 000 篇维基百科文章中抽取实体并链接到知识图谱上产生. 数据集 NYT 通过将 2013 年 119 000 篇 *New York Times* 语料文章中的实体与 Freebase 中的节点对齐而生成. PubMed 是一个医疗领域同义词数据集, 它通过将 150 万篇医疗论文摘要链接到统一医学语言系统知识图谱上生成.

扩展版同义词词林数据集由哈尔滨工业大学实验室在原版基础上进行扩展, 包含 77 343 条实体词, 且经过了人工校验. 在本文实验中, 随机抽取扩展版同义词词林语料中的 500 个单词集合作为测试集, 使用同义词词林的其他同义词集合作为训练集进行训练与验证, 并在测试集合上评估方法的同义词集合生成效果.

在数据集获取了实体链接对齐所得到的同义词集合后, 进行了实体级别的训练集与测试集的随机切分, 最终获得的测试集中没有与训练集重合的单词, 保证了测试效果的准确, 并提升了数据集的难度. 表 1 展示了 4 个数据集的详细统计信息.

实验分为以下两个部分设置, 针对这两个部分, 分别引入了通用的评价指标.

[1] 数据集下载 URL: http://bit.ly/SynSetMine-dataset

[2] 同义词词林语料 URL: http://ir.hit.edu.cn/demo/ltp/Sharing_Plan.htm

表 1 数据集统计信息
Table 1 Dataset statistics

数据集	Wiki	NYT	PubMed	CILIN
文档	100 000	118 664	1 554 433	—
句子	6 839 331	3 002 123	15 051 203	—
训练集单词	8 731	2 600	72 627	75 614
训练集同义词集合	4 359	1 273	28 600	17 317
测试集单词	891	389	1 743	2 237
测试集同义词集合	256	117	250	500

1) 对于同义词集合生成任务. 与文献 [9] 相同, 使用 3 种可以反映聚类质量的评价指标: 调整兰德系数 (Adjusted Rand index, ARI)、Fowlkes-Mallows 指数 (Fowlkes-Mallows index, FMI) 和归一化互信息 (Normalized mutual information, NMI)[29] 作为评价指标.

2) 对于集合−实例同义词分类任务. 使用二分类预测结果的精确率与召回率的调和平均 (F1) 值和正例样本的准确率作为评判指标.

2.2 实验设置与实现细节

本文使用与文献 [9] 相同的超参数设置, 使用训练集 5 折验证来进行超参数调优. 在成对字向量预训练部分, 除了本节列出的超参数, 其他超参数选择和预训练过程均与文献 [27] 相同. 在 CILIN 数据实验中, 由于使用了预训练的 300 维词向量, 因此对其他一些相关超参数 (如字级别表示维度、字向量维度等) 进行了调整. 本文唯一通过网格搜索进行超参数选择的超参数是字级别表示维度. 表 2 为本文在实验中所使用的超参数设置, 其中字向量维度 d_c、卷积窗口大小 w_c、字级别表示维度 d_v 是成对字向量模块引入的超参数, 辅助判别器损失比率 α 是噪声学习模块所引入的超参数. 除表 2 内容外, 本文使用一个两层的神经网络作为词表示模块, 使用一个三层的神经网络作为集合表示模块, 这部分网络结构与文献 [9] 也完全相同. 在额外引入的成对字编码模块中, 使用了单层的卷积−池化网络对成对字级别信息进行处理.

如第 1.4 节所言, 使用实体链接来获取同义词种子, 会在数据集中引入相当比例的噪声, 而且由于测试集中也存在这些噪声, 因此在一定程度上影响了测试的准确性. 为了缓解这个问题, 使用 WordNet[19][3] 来过滤数据集中的噪声, 并额外增加一些缺失的同义词, 将这样处理后的测试集称为干净样本

[3] WordNet 下载 URL: https://wordnet.princeton.edu/

表 2　超参数设置
Table 2　Hyper-parameter settings

数据集	Wiki	NYT	PubMed	CILIN
词向量维度 d_w	50	50	50	300
词级别表示维度 d_p	250	250	250	250
集合表示隐单元维度 d'_s	500	500	500	500
学习率	0.0001	0.0001	0.0003	0.0003
训练轮数	800	500	50	50
负样本采样数量 K	50	20	50	70
批大小	64	32	32	32
随机失活比例	0.5	0.3	0.3	0.3
字向量维度 d_c	50	50	50	150
卷积窗口大小 w_c	5	5	5	5
字级别表示维度 d_v	24	24	24	50
辅助判别器损失比率 α	0.15	0.15	0.15	0.15

集合. 详细的处理过程以及训练集和测试集中的噪声比例在第 2.4 节给出.

2.3　基准方法

用来对比的基准方法有以下 6 种:

1) K-均值 (K-means) 是一种无监督的聚类算法. 输入每一个候选词的词向量, 输出 K 个集合. 在每个数据集上, K 被设置为测试集正确同义词集合的数量.

2) 鲁汶 (Louvain)[30] 是一种社区发现算法. 输入是一张图, 输出是从图中发现的社区聚类. 计算过程为: a) 以所有候选词为图的节点, 若候选词词向量之间的相似度超过阈值, 则增加一条候选词间的边; b) 在训练集上进行阈值的调优.

3) 集合扩展 (SetExpan)[31] 是一种两阶段的无监督方法. 计算过程为: a) 按照词向量相似度找到每个词的 k 近邻; b) 输入 k 近邻图, 使用 Louvain 计算同义词集合.

4) 约束 K-means 是一种增加了显式约束的半监督 K-means 聚类变种, 其输出的聚类必须符合约束项. 支持向量机−鲁汶算法 (Support vector machine + Louvain, SVM + Louvain) 是一种两阶段半监督方法. 计算过程为: a) 使用 SVM 判别两个候选词是否为同义词 (粗筛选); b) 将构建的图使用 Louvain 进一步聚类. 其中 SVM 分类器使用训练集进行训练.

5) 成对判别器 (L2C)[32] 是一种监督式聚类方法. 使用神经网络学习一个成对同义词判别器, 并使用这个成对判别器生成同义词集合.

6) 集合−实例判别器 (SynSetMine)[9] 是一种有监督的基于集合表示学习的集合−实例分类方法. 集合生成时, 基于分类器的预测使用贪心策略进行. 这是本文方法的基准.

文本 NL-P2V 方法与 SynSetMine 的区别如下: 1) SynSetMine 是本文方法的基准集合生成方法. NL-P2V 在其基础上, 针对实体词表示问题, 增加了成对字向量模块. 2) 针对噪声标签问题, 增加了噪声学习模块.

2.4　数据清洗与噪声分析

如第 1.4 节所述, 很多利用远程监督的方法会造成噪声标签问题. 而在同义词挖掘任务中, 产生噪声和遗漏同义词对的原因主要有以下 3 点: 1) 将实体链接到同一个实体的所有词都作为同义词, 而实体链接器训练和预测语料的分布差异较大; 2) 遗漏了链接到两个同义词实体的候选词, 这些候选词也应当是同义词; 3) 由于领域知识图谱的节点缺失, 造成无法有效链接.

为了验证 3 个英文数据集中的同义词质量和覆盖率, 使用 WordNet 对其进行过滤. 使用的过滤方法所依赖的假设条件比较宽松: 在 3 个数据集上, 当 2 个单词都存在于 WordNet 中时, WordNet 相较于远程监督的结果, 在统计意义上更为准确. 主要原因有以下 3 点 : 1) WordNet 的英文同义词库被 George[19] 整理. 从其他一些领域研究现状看, 会将人类标注者标注的数据看作基本准确[33]. 2) 如第 1.4 节所述, 远程监督所获得的数据包含了一定比例的噪声, 且没有经过校验. 3) 假设条件较为苛刻, 缩小了过滤范围.

给定原始的同义词集合 $C = \{C_1, C_2, \cdots, C_N\}$ 和所有候选词库 $\mathcal{V}$, 其中候选词库中的单词与所有集合中的单词是等价的. 记 WordNet 中的词库为 $\mathcal{V}_w$, WordNet 中单词 w_i 的同义词集合为 $syn(w_i)$. 过滤和增广数据的策略如下:

1) 将同义词集合转换为同义词对. 转换结果只包含所有正样本的同义词对 $\{\cdots, < w_i, w_j >, \cdots\}$.

2) 同义词对 $< w_i, w_j >$ 过滤规则. 若两个单词都在 WordNet 的词库中即 $w_i \in \mathcal{V}_w$ 且 $w_j \in \mathcal{V}_w$, 且两个单词都不在 WordNet 中对方的同义词集合中, 则将其过滤 $w_j \notin syn(w_i)$, $w_i \notin syn(w_j)$.

3) 增强策略. 对单词 w_i, 若另一个非同义词单词同时出现在数据集词库和 WordNet 词库中, 即 $w_j \in \mathcal{V}$, $w_j \in \mathcal{V}_w$, 且在 WordNet 中有 $w_j \in syn(w_i)$, $w_i \in syn(w_j)$, 则将 $< w_i, w_j >$ 增加到同义词对中.

由过滤和增广数据策略可以看出, 在噪声过滤实验中判断噪声同义词对的标准是很严格的. 用以上策略分别在 3 个数据集的训练集和测试集进行训练, 但实验中未使用处理后的训练集, 所有方法的训练都使用了原始的带噪声训练集. 使用处理后的干净测试集对方法效果进行衡量. 如表 3 所示, 在 Wiki 和 PubMed 数据集上, 都存在不同比例的噪声标签和数据缺失情况. 例如在 Wiki 数据集的训练集和测试集上, 噪声样本对的比例分别为 20.0% 和 26.6%, 遗漏样本对的比例分别为 8.6% 和 28.6%. 在 PubMed 数据集上, 能够过滤的噪声数据比例最低, 为 4.6%. 这可能是由于 PubMed 数据集主要由特定领域的实体组成.

表 3 数据集噪声比例

Table 3 Noise data percentage on datasets

统计类别	Wiki		PubMed	
	训练集	测试集	训练集	测试集
原始词对	4372	635	44027	1493
噪声样本对	875	169	2740	70
遗漏样本对	380	182	12851	331
干净词对	3877	648	54138	1754
原始集合数量	4359	256	28600	250
干净集合数量	3327	228	25761	259

由于对实验中所使用的数据集进行了修改, 现对数据清洗实验的设计思路总结如下: 实验中, 使用的 3 个同义词挖掘数据集是通过远程监督的方式以实体链接产生同义词数据, 不可避免地会带来噪声. 因此, 使用 WordNet 以一种保守的策略进行数据清洗, 即当且仅当两个单词同时在 WordNet 的词库中且它们之间 (直接或间接) 的同义词关系与原始数据集中不同时, 才会进行修正并统计. 其中隐含的假设是 WordNet 中的同义词准确率要高于 3 个远程监督形成的同义词数据集. 表 4 中的开源基准方法 SynSetMine 和改进方法 NL-P2V 都是在同一清洗后的测试集上进行的, 因此实验效果是比较公平的.

2.5 实验结果及分析

表 4 列出了所有基准方法和本文 NL-P2V 方法的实验效果. 其中所有方法都被在测试集上执行 5 次, 报告平均性能并标注了标准偏差值即表 4 中 (± std). 其中, 带“∗”上标的方法在原始测试集 (含噪声) 上进行实验, 不带“∗”上标的方法在使用 WordNet 过滤噪声后的测试集上进行实验. NL-P2V w/o P2V 表示基准方法加上噪声学习而不包含成对字向量方法的模型变种. NL-Word-P2V 表示将 NL-P2V 中的成对字向量替换为成对词向量模块后的模型变种. 3 个英文数据集上词向量的维度设置均与表 2 中字级别表示维度 d_v 一致.

1) 集合生成效果

集合生成任务利用已经训练好的集合–实例同

表 4 实验结果 (%)

Table 4 Main experimental results (%)

方法	Wiki			NYT			PubMed		
	ARI (± std)	FMI (± std)	NMI (± std)	ARI (± std)	FMI (± std)	NMI (± std)	ARI (± std)	FMI (± std)	NMI (± std)
K-means*	34.35 (± 1.06)	35.47 (± 0.96)	86.98 (± 0.27)	28.87 (± 1.98)	30.85 (± 1.76)	83.71 (± 0.57)	48.68 (± 1.93)	49.86 (± 1.79)	88.08 (± 0.45)
Louvain*	42.25 (± 0)	46.48 (± 0)	92.58 (± 0)	21.83 (± 0)	30.58 (± 0)	90.13 (± 0)	46.58 (± 0)	52.76 (± 0)	90.46 (± 0)
SetExpan + Louvain*	44.78 (± 0.28)	44.95 (± 0.28)	92.12 (± 0.02)	43.92 (± 0.90)	44.31 (± 0.93)	90.34 (± 0.11)	58.91 (± 0.08)	61.87 (± 0.07)	92.23 (± 0.15)
约束 *K*-means*	38.80 (± 0.51)	39.96 (± 0.49)	90.31 (± 0.15)	33.80 (± 1.94)	34.57 (± 2.06)	87.92 (± 0.30)	49.12 (± 0.85)	51.92 (± 0.83)	89.91 (± 0.15)
SVM + Louvain*	6.03 (±0.73)	7.75 (± 0.81)	25.43 (± 0.13)	3.64 (± 0.42)	5.10 (± 0.39)	21.02 (± 0.27)	7.76 (± 0.96)	8.79 (± 1.03)	31.08 (± 0.34)
L2C*	12.87 (± 0.22)	19.90 (± 0.24)	73.47 (± 0.29)	12.71 (± 0.89)	16.66 (± 0.68)	70.23 (± 1.20)	—	—	—
SynSetMine*	56.43 (± 1.31)	57.10 (± 1.17)	93.04 (± 0.23)	44.91 (± 2.16)	46.37 (± 1.92)	90.62 (± 1.53)	74.33 (± 0.66)	74.45 (± 0.64)	94.90 (± 0.97)
SynSetMine	54.52 (± 1.23)	54.87 (± 1.08)	92.80 (± 0.20)	47.33 (± 1.84)	47.96 (± 2.07)	90.16 (± 1.29)	71.61 (± 0.66)	72.20 (± 0.60)	94.38 (± 0.60)
NL-P2V	63.01 (± 1.06)	63.54 (± 0.98)	93.92 (± 0.12)	50.72 (± 1.63)	52.88 (± 2.10)	91.66 (± 1.02)	75.54 (± 0.88)	75.65 (± 0.56)	94.98 (± 0.49)
NL-Word-P2V	61.31 (± 0.94)	61.18 (± 0.76)	93.70 (± 0.41)	49.13 (± 1.07)	51.69 (± 1.71)	91.21 (± 0.45)	74.67 (± 0.96)	74.58 (± 0.50)	95.02 (± 0.46)
NL-P2V w/o P2V	56.09 (± 1.01)	56.34 (± 0.83)	93.13 (± 0.31)	49.04 (± 1.43)	50.02 (± 1.79)	91.07 (± 0.57)	73.48 (± 0.92)	73.49 (± 0.47)	94.47 (± 0.56)

义词分类器, 将所有测试集中的候选词依次聚合到若干个同义词集合中, 并和正确的集合进行相似度对比. 表 4 展示了不同方法在 3 个数据集测试集上的实验效果. 由表 4 可知: 1) 无监督的聚类方法 (如 K-means、Louvain 等) 由于没有使用监督信息, 在同义词集合生成任务上的效果不佳. 而作为基准的有监督统计机器学习方法 (如约束 K-means、SVM + Louvain 和 L2C) 并不总是能够收益于监督信息的加入. 前者由于有效利用了监督信息作为聚类约束, 取得了一定的效果提升, 但后者可能受限于 SVM 的表达能力, 反而造成了性能的大幅度下降. 2) 另一个作为基准的骨干模型 SynSetMine 由于能够直接捕捉集合级别的特征, 因此在所有基准模型中获得了最佳效果. 无论是在原始的噪声测试集, 还是在利用 WordNet 过滤后的测试集上, 该方法都获得了近似的效果, 表明以该模型作为基准模型, 已经具有强大的表示学习能力. 综上所述, 表 4 实验结果表明, 通过引入噪声学习和成对字向量表示, 本文提出的 NL-P2V 方法进一步大幅度地提升了 3 个集合相似度指标, 且在所有数据集上所有指标的提升幅度都非常显著. 提升的主要来源是噪声学习对于噪声标签影响的缓解, 以及成对字向量对于实体这种本质上具有少样本学习特征的单词在表示能力上的进一步提升.

2) 消融实验

为了验证每个模块对于模型性能的提升贡献, 本文还进行了消融实验, 将完整模型与两个模型变种 NL-Word-P2V 和 NL-P2V w/o P2V 进行性能比较, 如表 4 下半部分所示. 由表 4 可以看出: 1) 单独的噪声学习模块相对于 SynSetMine 方法平均能带来 1.9% 的 ARI 提升, 验证了噪声学习模块对于噪声数据下鲁棒性的增强效果; 2) 通过对加入成对词向量的模型变种 NL-Word-P2V 和完整模型 NL-P2V 在 3 个数据集上的集合生成性能和使用成对字向量相比较, 成对词向量能够带来平均 2.39% 的 ARI 提升和 2.54% 的 FMI 提升. 成对字向量相对于成对词向量的提升原因主要有以下 2 点: 1) P2V 是一种用于跨句匹配的成对语义表示学习方法, 主要解决的是词义匹配的问题. 而通过扩展到字级别, 使 P2V 能够额外考虑字符级别的词义关系匹配. 2) 模型中的词表示是通过成对语义表示与词向量拼接得到的, 而成对字向量蕴含了更多词汇补充信息.

3) 集合−实例同义词分类效果

集合−实例同义词分类是对某个单词是否属于一个已有的同义词集合的预测. 为了分析模型在这个子任务上的效果, 本文在 PubMed 数据集上进行了预处理, 将测试集中的所有同义词集合转换为了 3 486 个集合−单词对, 其中包括 1 743 个正例样本和 1 743 个负例样本. 本文比较了 3 种方法在这个任务上的效果: 1) Shen 等[9] 使用的基准方法即将集合中每个单词与目标单词进行相似性评估, 并使用均值作为最终预测的概率; 2) Shen 等[9] 提出的集合编码方法; 3) 本文提出的噪声鲁棒的成对字编码增强方法 NL-P2V. 最终结果如图 2 所示, 在不同规模的集合−实例分类下, 绝大多数情况下 F1 和准确率均有提升, 验证了本文提出的噪声鲁棒方法和成对字编码模块的效果.

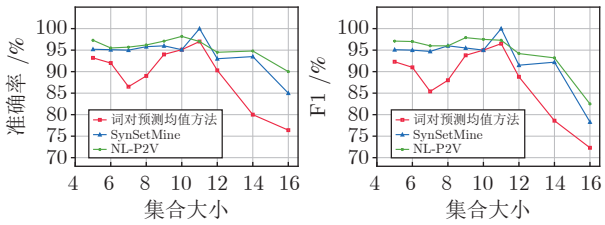


图 2　不同集合大小下模型性能对比

Fig. 2　Model performances on samples with different set size

4) 中文同义词集合分类实验

为了测试本文方法在不同语言尤其是在中文语料上的适用性, 在哈尔滨工业大学提出的扩展同义词词林语料上进行了训练, 并与基准方法进行了性能对比. NL-P2V 方法中的成对字向量模块的参数并没有经过预训练, 而是随机初始化并在训练时更新, 其他超参数设置见表 2. 由于经过人工大量校准, 同义词词林语料上噪声较少, 因此本文进一步随机在训练集上增加了 5% 和 10% 的噪声比例. 表 5 列出了本文方法与 4 种噪声方法的对比实验结果. 由表 5 可以看出, 本文提出的 NL-P2V 能够应用于中文语料, 并在包含随机噪声的训练数据集上表现出一定的鲁棒性. 增加了成对字向量和噪声学习后, NL-P2V 方法在不同训练噪声下的 CILIN 数据集上仍然能够取得最高约 2.02% 的 NMI 指标提升和约 1.96% 的 FMI 指标提升. 虽然引入的两个模块能够起到一定效果, 但受限于训练噪声类型, 噪声学习模块无法在所有 CILIN 数据的训练噪声比例下带来稳定的提升.

5) 中、英文数据性能差异分析

需要注意的是, 本文在 CILIN 数据集上的性能提升低于在 3 个英文同义词数据集, 且性能提升表现与英文数据集并不一致. 为了分析中、英文数据集上的性能差异来源, 本文选择了 NYT 和含 10% 噪声的 CILIN 数据集, 对测试样本根据所在同义词

表 5 CILIN 实验结果 (%)
Table 5 Experimental results on CILIN (%)

方法	训练噪声比例	ARI	FMI	NMI
SynSetMine	0	17.07	17.97	71.94
NL-P2V	1	20.26	20.73	73.97
SynSetMine	2	17.02	17.57	73.34
NL-P2V	3	17.01	17.96	73.36
SynSetMine	3	14.28	15.80	75.00
NL-P2V	5	16.24	16.91	74.01

集合大小归为 5 类, 并分别记录每个数据集上、每类样本的集合−实例同义词分类效果相比于 SynSetMine 方法的提升幅度, 结果如图 3 所示. 由图 3 可以看出: 1) 无论对于中文还是英文数据集来说, 集合词数量超过一定规模后, 分类效果会迅速下降. 这可能是由于过大的同义词集合包含的词义概念较多, 学习难度较大. 2) 在中文 CILIN 数据集上, 对规模较大集合的提升幅度更小. 说明本文进行的改进在应用于此类样本时, 无法达到有效提升. 从总体结果看, 在中文 CILIN 数据集上提升幅度较小的主要原因有: 1) 由于本文并未进行预训练, 因此成对字向量模块无法学习到较好的补充语义表示; 2) 增加的噪声为随机噪声, 其噪声类型和远程监督引入的噪声有较大区别, 影响了噪声学习模块的效果.

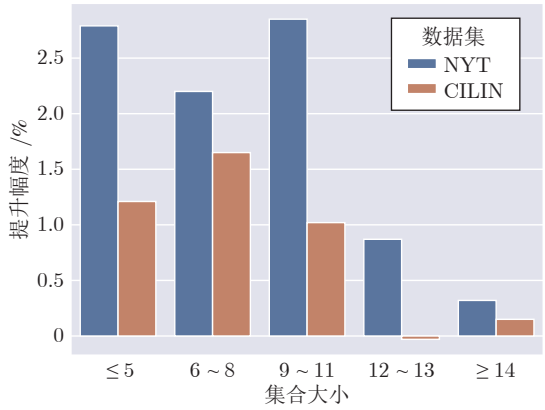


图 3 不同集合大小的中、英文数据性能效果对比
Fig. 3 Comparison of performance enhancement in Chinese and English data with different set sizes

6) 方法效率分析

本文所有实现和改进都基于 Pytorch 框架和原始 SynSetMine 代码库[4]. 所有训练和预测都是在单个 1080Ti GPU 上完成, 本文方法与 SynSetMine 基准方法的效率对比如表 6 所示. 由表 6 可以看出, 相比于原始基准方法, 本文提出的噪声学习方法并没有额外降低很多效率, 是一种轻量级的增强方法; 而成对字向量的方法对性能提升幅度较大, 但是会成倍增加执行的时间消耗.

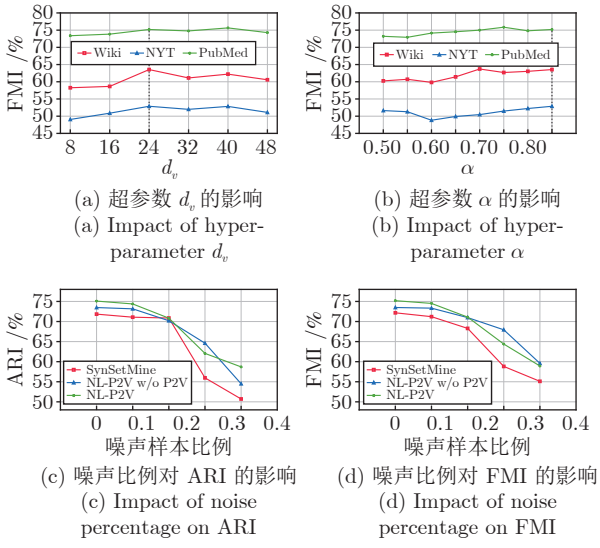
[4] 源代码开放于: https://github.com/mickeystroller/SynSetMine-pytorch

表 6 效率对比
Table 6 Efficiency comparison

方法	训练			集合预测		
	Wiki (h)	NYT	PubMed (h)	Wiki (s)	NYT (s)	PubMed (s)
K-means	—	—	—	1.82	0.88	2.95
Louvain	—	—	—	3.94	20.59	74.60
SynSetMine	7.7	77 min	3.6	3.57	1.24	19.11
NL-P2V w/o P2V	8.2	80 min	4.9	3.60	1.18	20.58
NL-P2V	18.1	2.9 h	7.1	6.47	2.69	27.04

7) 超参数的影响

由于在噪声鲁棒学习和成对字向量模块中分别引入了辅助判别器的损失比率 α 和字级别表示维度 d_v 两个超参数. 本文对这两个超参数进行了网格搜索, 搜索结果见图 4(a)、图 4(b). 由于本文在进行实验时并没有按照网格搜索的最优超参数来调优, 因此表 4 中最终模型的结果某些指标可能要低于图 4 中的最佳结果. 另外, 由图 4 可以看出, 超参数 α 对性能有影响, 但在 0.70 ~ 0.85 取值区间内, 在 3 个数据集上始终都能显著超出基准模型.



(a) 超参数 d_v 的影响
(a) Impact of hyper-parameter d_v

(b) 超参数 α 的影响
(b) Impact of hyper-parameter α

(c) 噪声比例对 ARI 的影响
(c) Impact of noise percentage on ARI

(d) 噪声比例对 FMI 的影响
(d) Impact of noise percentage on FMI

图 4 超参数以及训练集噪声比例的影响分析
Fig. 4 Analysis of the impact on hyper-parameters and training set noise percentage

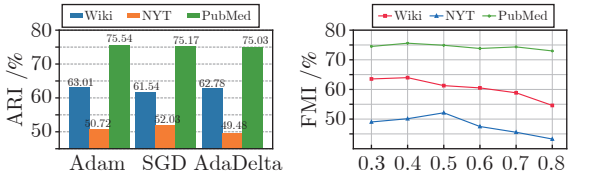
8) 训练集不同噪声比例的影响

为了进一步探究噪声对于模型的影响, 本文通过在过滤后的训练集上增加随机噪声的方式训练不同的模型, 并在干净测试集上进行测试实验. 噪声比例为 10% ~ 40%. 图 4(c)、图 4(d) 给出本文模型

不同变种以及基准模型的实验结果, 表明使用噪声鲁棒和成对字向量有利于缓解训练数据中的噪声标签对于模型性能的影响. ARI 和 FMI 两项指标下降的幅度说明在这两个模块中, 噪声鲁棒模块对于缓解噪声标签带来的影响更加重要.

9) 优化器与集合生成策略的影响

为了进一步探究其他超参数对于同义词集合生成效果的影响, 本文分别使用不同的优化器, 生成时使用不同的集合生成策略, 并比较模型在这些不同超参数下的性能. 如图 4 所示, 本文分别尝试了 Adam、随机梯度下降 (Stochastic gradient descent, SGD) 和 AdaDelta 优化器进行训练, 并在同义词集合生成时, 分别设置最低判别阈值 θ 为 0.3、0.4、0.5、0.6、0.7、0.8 的策略进行集合生成, 比较 NL-P2V 方法在不同优化器和生成策略下的性能波动. 其中 SGD 优化器设置学习率为 0.01, AdaDelta 优化器学习率和超参数为默认参数. 分别记录生成的同义词集合的平均 ARI 和 FMI 值, 来测试这 2 种超参数对性能的影响. 由图 5(a) 可知, 不同的优化器会对集合生成性能产生轻微影响, 且 3 种优化器中 Adam 优化器的平均效果最好; 而不同判别阈值下的集合生成策略会对性能有较大影响. 当判别阈值超过 0.5 时, 效果较差.



(a) 不同优化器的影响　(b) 集合生成策略阈值 θ 的影响

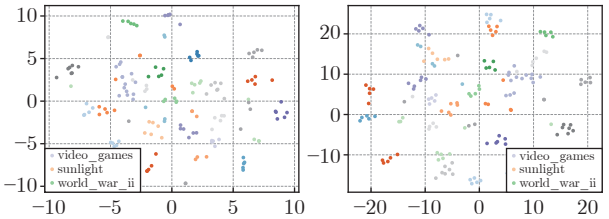
(a) Impact of different optimizers　(b) Impact of set generation policy threshold θ

图 5　不同优化器和生成策略下集合生成效果

Fig. 5　Model performances with different optimizers and set generation policy

10) 词表示的低维可视化对比

为了验证本文引入的成对字向量模块的效果, 分别导出原始词表示和由成对字向量模块增强的词表示. 使用 Wiki 测试集上集合数量超过 4 的同义词集合样本作为可视化数据. 样本中包含 28 个集合共 155 个实体词. 使用 T-分布随机近邻嵌入算法[34] 进行降维并根据集合分组进行散点图绘制, 选择并标记了 3 个单词所在的集合作为示例, 可视化效果如图 6 所示. 由图 6 可以看出, 在大多数情况下, 加入成对字向量增强后的词表示实体同义词类别内聚集程度更高. 该实验说明了引入成对字向量后, 对模型在语义表示学习上的帮助和提升效果显著. 本文使用一个样例, 从直觉上说明成对词向量和成对字向量的差异. 例如对同义单词“檀香山”和“檀岛”. 若使用成对词向量, 受限于词库大小, 这两个出现频率较低的实体词会因其是词库外词而计算得到接近随机的语义表示. 使用成对字向量模块后, 因为字库或者字符库的覆盖率更高, 两实体词可以从其组成的字中获得一些相对有意义的语义信息, 因此影响了最终计算的词表示聚类效果.



(a) 传统的单独词表示　(b) 成对字向量增强词表示

(a) Traditional word representations　(b) Pair-wise character vector enhanced (word) representations

图 6　加入成对字向量之前和之后词表示可视化对比

Fig. 6　Visualization of word distributed representations with or without pair-wise character vector embeddings

3　相关工作

3.1　实体同义词发现

之前大多数实体同义词发现工作都集中在从 (半) 结构化数据中构建训练种子和发现新实体同义词, 如网络表[35] 和查询日志的点击图[8]. 本文直接从无结构化的文本语料中挖掘实体同义词的集合, 扩大了可应用范围.

已有方法在提取同义词时, 大都利用了共现统计信息[36]、文本模式匹配[37]、分布式相似度[17, 22, 38] 或它们的组合[6] 来提取同义词. 然而, 这些方法只能判别同义词对或查询实体的候选同义词排名, 因为没有考虑集合中同义词之间的关联, 无法生成高质量的同义词集合. 一些研究试图进一步处理候选同义词排名列表或建立同义词对图, 然后应用图聚类技术得到同义词的集合[39]. 然而, 这些两阶段方法都存在噪声累积问题. 文献 [9] 利用集合−实例分类器进行同义词集合生成, 缓解了之前工作所面临的集合表示学习和同义词集合无法生成的问题. 本文针对之前同义词挖掘方法所忽略的噪声学习和预训练词表示不足的情况, 进行了相应改进和探索.

3.2　噪声学习

噪声学习是如何在训练数据存在部分含噪声的错误标签的情况下, 进行有效学习的一类问题. 在经典机器学习领域, 噪声学习问题被广泛研究, 例

如利用统计分类器来识别噪声标签. 近几年, 也有很多基于神经网络的方法使用神经网络来处理噪声标签问题, 例如在计算机视觉、自然语言处理等应用领域引入远程监督构造训练数据任务中, 也存在着相当比例的噪声, 例如远程监督条件下的信息抽取和命名实体分类问题. 在这些问题上, 如何避免噪声标签的影响即各类噪声学习的方法, 也被大量应用.

根据近年来的研究[40], 本文将噪声问题根据性质总结归类如下: 对应一个 N 分类问题, 输入特征 x、噪声标签 $\tilde{y}$ 即真实标签 y 属于 N 种类别中的一种或多种. 根据噪声产生的方式, 可以分为人工合成噪声和真实噪声, 通常分别对应着类级别的噪声和实例级别噪声[14]. 前者的噪声标签只与标签 y 相关, 后者与 $< x, y >$ 相关. 根据减少噪声影响的方法, 可以将基于深度神经网络的噪声学习分为以下 5 类: 1) 从模型的角度避免噪声的影响. 例如增加辅助结构来预测噪声转移矩阵. 但此类方法对噪声的先验假设依赖性强. 2) 设计鲁棒的损失函数避免噪声的影响. 例如对称交叉熵函数等. 但有研究表明[11], 这类方法对于复杂噪声不一定有效, 而且会增加收敛时间. 3) 基于干净样本识别的方法. 例如在训练集的一个准确样本子集上进行训练, 并通过样本选择或者样本权重学习、标签修正等策略减少噪声样本的影响[41]. 但这种方式依赖预先定义有足够规模的准确样本集合. 4) 利用正则化项的方式减少对噪声样本的过拟合, 例如 MixUp 和 MentorMix[14] 等. 此类方法灵活轻量可以搭配其他方法使用, 但通常在噪声数据下提升幅度不大. 5) 根据任务先验减少噪声的影响. 例如在远程监督信息抽取任务中[42], 采用注意力机制将一个包中的句子和候选关系计算权重并忽略权重较低的实例.

本文设计的通过隐变量分布估计和交替优化的学习框架, 其优势是不需要额外划分准确样本集合, 且不需要噪声先验知识.

4 结束语

同义词挖掘是自然语言处理和信息抽取中的重要方向, 在知识库补全、搜索扩展等下游任务上有重要作用, 得到了广泛的研究与关注. 在前人研究的基础上, 本文工作聚焦于在远程监督产生的种子数据下的同义词挖掘任务, 基于远程监督过程中的实体链接带来的噪声标签问题, 构建了噪声鲁棒的同义词挖掘模型. 该方法通过引入隐变量估计和交替优化的噪声学习框架来修正噪声标签, 进一步使用字级别的成对向量表示增强模型的实体表示能力. 通过在 3 个规模不同的领域数据集和中文扩展版同义词词林数据集上与几种基准方法的实验对比, 结果表明, 本文方法在包括同义词聚类、同义词集合判别以及不同比例噪声训练在内的一系列实验中, 都表现出了显著提升效果. 在后续工作中, 将继续研究噪声鲁棒模块的一些细节问题, 如噪声鲁棒模块的标签分布学习效果等. 另外, 在本文工作中, 对成对词向量和成对字向量未进行更加深入的实验比较, 这是今后需要补充的工作. 此外, 探索和研究其他形式构造的同义词训练种子数据中噪声的分布和对模型学习过程的影响, 探索噪声学习在其他噪声数据下的效果, 以及进一步尝试将噪声学习方法应用在其他自然语言处理的任务上, 是未来要做的一项工作.

References

1 Azad H K, Deepak A. Query expansion techniques for information retrieval: A survey. *Information Processing & Management*, 2019, **56**(5): 1698−1735

2 Gui T, Ye J, Zhang Q, Zhou Y, Gong Y, Huang X. Leveraging document-level label consistency for named entity recognition. In: Proceedings of the 29th International Joint Conference on Artificial Intelligence. Virtual Event: 2020. 3976−3982

3 Zhang H, Cai J, Xu J, Wang J. Complex question decomposition for semantic parsing. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy: ACL, 2019. 4477−4486

4 Rao Zi-Yun, Zhang Yi, Liu Jun-Tao, Cao Wan-Hua. Recommendation methods and systems using knowledge graph. *Acta Automatica Sinica*, 2021, **47**(9): 2061−2077
(饶子昀, 张毅, 刘俊涛, 曹万华. 应用知识图谱的推荐方法与系统. 自动化学报, 2021, **47**(9): 2061−2077)

5 Hou Li-Wei, Hu Po, Cao Wen-Lin. Automatic Chinese abstractive summarization with topical keywords fusion. *Acta Automatica Sinica*, 2019, **45**(3): 530−539
(侯丽微, 胡珀, 曹雯琳. 主题关键词信息融合的中文生成式自动摘要研究. 自动化学报, 2019, **45**(3): 530−539)

6 Qu M, Ren X, Han J. Automatic synonym discovery with knowledge bases. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Halifax, Canada: ACM, 2017. 997−1005

7 Wang Z, Yue X, Moosavinasab S, Huang Y, Lin S, Sun H. SurfCon: Synonym discovery on privacy-aware clinical data. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage, USA: ACM, 2019. 1578−1586

8 Li C, Zhang M, Bendersky M, Deng H, Metzler D, Najork M. Multi-view embedding-based synonyms for email search. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. Paris, France: ACM. 575−584

9 Shen J, Lyu R, Ren X, Vanni M, Sadler B, Han J. Mining entity synonyms with efficient neural set generation. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Honolulu, Hawaii, USA: AAAI, 2019. 249−256

10 Song H, Kim M, Park D, Lee J. Learning from noisy labels with

deep neural networks: A survey [Online], available: https://arxiv.org/abs/2007.08199, July 22, 2020

11 Arazo E, Ortego D, Albert P, O′Connor N E, McGuinness K. Unsupervised label noise modeling and loss correction. In: Proceedings of the 36th International Conference on Machine Learning. Long Beach, USA: PMLR, 2019. 312−321

12 Zhang H, Long D, Xu G, Zhu M, Xie P, Huang F, et al. Learning with noise: Improving distantly-supervised fine-grained entity typing via automatic relabeling. In: Proceedings of the 29th International Joint Conference on Artificial Intelligence. Virtual Event: IJCAI, 2020. 3808−3815

13 Chen B, Gu X, Hu Y, Tang S, Hu G, Zhuang Y, et al. Improving distantly-supervised entity typing with compact latent space clustering. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, USA: ACL, 2019. 2862−2872

14 Jiang L, Huang D, Liu M, Yang W. Beyond synthetic noise: Deep learning on controlled noisy labels. In: Proceedings of the 37th International Conference on Machine Learning. Virtual Event: PMLR, 2020. 4804−4815

15 Mikolov T, Sutskever I, Chen K, Corrado G S, Dean J. Distributed representations of words and phrases and their compositionality. In: Proceedings of the 27th Annual Conference on Neural Information Processing Systems. Lake Tahoe, USA: NIPS, 2013. 3111−3119

16 Li Xiao-Tao, You Shu-Juan, Chen Wei. An algorithm of semantic similarity between words based on word single-meaning embedding model. *Acta Automatica Sinica*, 2020, **46**(8): 1654−1669
(李小涛, 游树娟, 陈维. 一种基于词义向量模型的词语语义相似度算法. 自动化学报, 2020, **46**(8): 1654−1669)

17 Fei H, Tan S, Li P. Hierarchical multi-task word embedding learning for synonym prediction. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage, USA: ACM, 2019. 834−842

18 Roth M, Upadhyay S. Combining discourse markers and cross-lingual embeddings for synonym-antonym classification. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, USA: ACL, 2019. 3899−3905

19 George A M. WordNet: A lexical database for English. *Communications of the ACM*, 1995, **38**(11): 39−41

20 Zaheer M, Kottur S, Ravanbakhsh S, Póczos B, Salakhutdinov R, Smola A J. Deep sets. In: Proceedings of the Annual Conference on Neural Information Processing Systems. Long Beach, USA: NIPS, 2017. 3391−3401

21 Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. In: Proceedings of the 5th International Conference on Learning Representations. Toulon, France: ICLR, 2017.

22 Hazem A, Daille B. Word embedding approach for synonym extraction of multi-word terms. In: Proceedings of the 11th International Conference on Language Resources and Evaluation. Miyazaki, Japan: ELRA, 2018. 297−303

23 Devlin J, Chang M W, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, USA: ACL, 2019. 4171−4186

24 Banar N, Daelemans W, Kestemont M. Character-level transformer-based neural machine translation. In: Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval. Seoul, South Korea: ACM, 2020. 149−156

25 Miyamoto Y, Cho K. Gated word-character recurrent language model. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. Austin, USA: ACL, 2016. 1992−1997

26 Lukovnikov D, Fischer A, Lehmann J, Auer S. Neural network-based question answering over knowledge graphs on word and character level. In: Proceedings of the 26th International Conference on World Wide Web. Perth, Australia: ACM, 2017. 1211−1220

27 Joshi M, Choi E, Levy O, Weld D S, Zettlemoyer L. Pair2Vec: Compositional word-pair embeddings for cross-sentence inference. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, USA: ACL, 2019. 3597−3608

28 Pereyra G, Tucker G, Chorowski J, Kaiser L, Hinton G E. Regularizing neural networks by penalizing confident output distributions. In: Proceedings of the 5th International Conference on Learning Representations. Toulon, France: ICLR, 2017.

29 Nguyen X V, Epps J, Bailey J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research*, 2010, **11**: 2837−2854

30 Blondel V, Guillaume J, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008, **2008**(10): Article No. 10008

31 Shen J, Wu Z, Lei D, Shang J, Ren X, Han J. SetExpan: Corpus-based set expansion via context feature selection and rank ensemble. *Machine Learning and Knowledge Discovery in Databases*, 2017, **1**: 288−304

32 Hsu Y, Lv Z, Kira Z. Learning to cluster in order to transfer across domains and tasks. In: Proceedings of the 6th International Conference on Learning Representations. Vancouver, Canada: ICLR, 2018.

33 Xu P, Barbosa D. Neural fine-grained entity type classification with hierarchy-aware loss. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans, Louisiana, USA: ACL, 2018. 16−25

34 Van Der Maaten L. Accelerating T-SNE using tree-based algorithms. *The Journal of Machine Learning Research*, 2014, **15**(1): 3221−3245

35 He Y, Chakrabarti K, Cheng T, Tylenda T. Automatic discovery of attribute synonyms using query logs and table corpora. In: Proceedings of the 25th International Conference on World Wide Web. Montreal, Canada: ACM, 2016. 1429−1439

36 Liu X, Wang L, Zhang J, Yin J, Liu H. Global and local struc-

ture preservation for feature selection. *IEEE Transactions on Neural Networks and Learning Systems*, 2013, **25**(6): 1083−1095

37 Grigonyte G, Cordeiro J, Dias G, Moraliyski R, Brazdil P. Paraphrase alignment for synonym evidence discovery. In: Proceedings of the 23rd International Conference on Computational Linguistics. Beijing, China: ACL, 2010. 403−411

38 Wang Ya-Shen, Huang He-Yan, Feng Chong, Zhou Qiang. Conceptual sentence embeddings based on attention mechanism. *Acta AutomaticA Sinica*, 2020, **46**(7): 1390−1400
(王亚珅, 黄河燕, 冯冲, 周强. 基于注意力机制的概念化句嵌入研究. 自动化学报, 2020, **46**(7): 1390−1400)

39 Ustalov D, Panchenko A, Biemann C. Automatic induction of synsets from a graph of synonyms. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver, Canada: ACL, 2017. 1579−1590

40 Tang C, Liu X, Li M, Wang P, Chen J, Wang L, et al. Robust unsupervised feature selection via dual self-representation and manifold regularization. *Knowledge-based Systems*, 2018, **145**: 109−120

41 Wang X, Hua Y, Kodirov E, Robertson N M. ProSelfLC: Progressive self label correction for training robust deep neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Virtual Event: CVPR, 2021.

42 Lin Y, Shen S, Liu Z, Luan H, Sun M. Neural relation extraction with selective attention over instances. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin, Germany: ACL, 2016. 2124−2133

张浩宇 军事科学院国防科技创新研究院人工智能研究中心助理研究员. 2020 年获得国防科技大学博士学位. 主要研究方向为自然语言处理, 知识图谱.
E-mail: zhanghaoyu10@nudt.edu.cn
(**ZHANG Hao-Yu** Lecturer at the Artificial Intelligence Research Center, Defense Innovation Institute. He received his Ph.D. degree from National University of Defense Technology in 2020. His research interest covers natural language processing and knowledge graph.)

王 戟 国防科技大学计算机学院教授. 1995 年获得国防科技大学博士学位. 主要研究方向为软件方法学, 高可信与智能软件技术. 本文通信作者.
E-mail: wj@nudt.edu.cn
(**WANG Ji** Professor at the College of Computer, National University of Defense Technology. He received his Ph.D. degree from National University of Defense Technology in 1995. His research interest covers software methodology, high confidence and intelligent software technologies. Corresponding author of this paper.)