

# 基于词嵌入融合和循环神经网络的中英文隐喻识别<sup>\*</sup>

苏传东 黄孝喜 王荣波 谌志群 毛君钰 朱嘉莹 潘宇豪

(杭州电子科技大学认知与智能计算研究所 杭州 310018)

**摘要:**【目的】针对自然语言中普遍存在的隐喻现象,提出一种基于词向量融合和循环神经网络(RNN)的中英文隐喻识别方法。【方法】通过本文提出的词嵌入融合算法将文本映射到词向量空间作为神经网络的输入,以RNN作为编码器,注意力机制和池化技术作为特征提取器,最后利用Softmax计算文本为隐喻的概率。【结果】基于词嵌入融合的隐喻识别方法的准确率和F1值比基于普通词嵌入的方法在英文隐喻识别任务上可以提高11.8%和6.3%,在中文隐喻识别任务上可以提高8.9%和7.8%。【局限】由于长距离依存问题,本文方法在句式复杂的长文本上隐喻识别效果不稳定。【结论】基于词嵌入融合和RNN的模型在隐喻识别问题上表现非常好,说明词嵌入融合可以提高神经网络对隐喻的识别能力。

**关键词:** 隐喻识别 深度学习 词嵌入 循环神经网络

**分类号:** TP391

**DOI:** 10.11925/infotech.2096-3467.2019.0828

**引用本文:** 苏传东, 黄孝喜, 王荣波等. 基于词嵌入融合和循环神经网络的中英文隐喻识别[J]. 数据分析与知识发现, 2020, 4(4): 91-99.(Su Chuandong, Huang Xiaoxi, Wang Rongbo, et al. Identifying Chinese / English Metaphors with Word Embedding and Recurrent Neural Network[J]. Data Analysis and Knowledge Discovery, 2020, 4(4): 91-99.)

## 1 引言

隐喻是自然语言中非常常见的语言现象,隐喻实质是源域到目标域之间的概念映射,也是一种认知方式<sup>[1]</sup>。根据句法结构和文字用法的特征,常见的隐喻主要有三类:名词性隐喻、动词性隐喻和形容词性隐喻<sup>[2]</sup>。“共产党是红太阳”是名词性隐喻,源域是“红太阳”这个名词,目标域是“共产党”,将红太阳映射到共产党,象征着共产党的温暖。“阳光融化了我”是动词性隐喻,源域是“融化”这个动词,目标域是“阳光”,将融化映射到阳光,强调太阳非常温暖。“炽热的心”是形容词性隐喻,源域是“炽热”这个形容词,目标域是“心”,将炽热映射到心,说明内心非常向往某件事物。而“发如雪”这样本体和喻体都在句中且通过“像”、“如”等关联词进行关联的情况为

明喻,汉语概念上的隐喻是和明喻相对的概念。在“阳光融化了我”和“炽热的心”这两句中本体和喻体不能直接找出,需要根据语境推测,即汉语意义上的隐喻。然而隐喻的英语“Metaphor”是个广义的概念,明喻是使用关联词的特殊类型的隐喻<sup>[3]</sup>。本文的隐喻识别技术针对这个广义上的隐喻。

自然语言处理(Natural Language Processing, NLP)领域的科学家对隐喻这种语言现象越来越关注<sup>[4]</sup>。对隐喻进行理解和处理的计算机技术称为隐喻计算,隐喻计算包括隐喻识别、隐喻解释、隐喻生成。这些都是隐喻理解的基础技术,然而这些隐喻计算问题尚未被有效解决,严重制约机器翻译、问答系统、信息检索等自然语言理解(Natural Language Understanding, NLU)下游任务的发展<sup>[5]</sup>。例如

通讯作者: 黄孝喜, ORCID: 0000-0003-4483-3664, E-mail: huangxx@hdu.edu.cn。

<sup>\*</sup>本文系教育部人文社会科学研究规划基金项目“融合深度神经网络模型的汉语隐喻计算研究”(项目编号: 18YJA740016)和国家社会科学基金重大项目“汉语隐喻的逻辑表征与认知计算”(项目编号: 18ZDA290)的研究成果之一。

“She devour this novel.”,谷歌翻译系统<sup>①</sup>将其翻译为“她吞噬了这部小说。”,显然谷歌翻译系统仅仅翻译了字面含义,没有理解这句话的动词隐喻含义,“吞噬”应该映射到目标域“小说”,隐含意义为沉迷于小说。要解决隐喻现象这个自然语言理解的难题,还有许多工作要做。

近年来深度学习技术在NLP领域影响非常大,在多种NLP任务上取得了很好的效果,但在隐喻计算问题上的研究不够深入。本文使用深度学习方法解决中英文隐喻识别问题。

(1)提出基于词嵌入的语义优先中断理论,论述在词嵌入空间捕获隐喻特征的可能性。

(2)提出词嵌入融合方法,强化词嵌入对隐喻的表征能力。

(3)设计一种基于循环神经网络的隐喻识别器,通过注意力机制和池化技术加强隐喻特征提取能力。

## 2 相关工作

最初的隐喻识别工作主要基于符号规则。Wilks<sup>[6]</sup>提出语义中断理论和优先选择模型,在优先选择模型中隐喻会发生语义中断,从而判断存在隐喻现象。Fass<sup>[7]</sup>基于语义优先中断理论,提出修正语义学,并开发隐喻解释系统Met\*,由于语料库的限制,该系统在解释复杂的隐喻现象时表现不好。Neuman等<sup>[8]</sup>通过WordNet中的实体节点与边的关系识别隐喻,其原理是隐喻表达与字面表达在WordNet实体关系中有不同的模式,在与实体相关的名词性隐喻上识别效果很好,缺点是在动词性隐喻和形容词性隐喻识别上效果欠佳。

随着语料库的发展与丰富,隐喻识别工作逐渐集中在语料统计学。Shutova等<sup>[9]</sup>根据语义优先中断理论提出针对动词隐喻的聚类算法,无监督地学习隐喻意与字面意的区别,这种方法比基于WordNet的动词性隐喻识别效果更好,但只局限于动词性隐喻的识别。Hovy等<sup>[10]</sup>通过依存句法为隐喻表达建立句法树,提取句法树的语义特征,利用支持向量机(SVM)和条件随机场(CRF)建模,发现SVM的隐喻识别效果不错。Rai等<sup>[11]</sup>通过CRF对语

义特征、上下文特征和情感特征进行建模,证明这些额外的特征可以提高隐喻识别效果。Tsvetkov等<sup>[12]</sup>将独立语言的语义特征通过向量空间进行潜在语义分析(LDA)以提取词之间的关系特征,并通过随机森林(RF)识别多语平行语料中的隐喻。

深度学习技术极大地促进了NLP的进展,Kalchbrenner等<sup>[12]</sup>将卷积神经网络(CNN)应用于NLP,提出动态CNN(DCNN)以适用于不同长度的文本。Graves等<sup>[13]</sup>将RNN应用于NLP,相比于CNN,RNN更加符合自然语言本身递归循环的性质。将深度学习技术应用于隐喻计算方面的工作相对较少,Do Dinh等<sup>[14]</sup>使用多层感知机(MLP)识别隐喻,相对于传统机器学习方法效果较好。Bizzoni等<sup>[15]</sup>利用词向量强大有效的表征能力结合基础神经网络架构处理隐喻,在名词性隐喻和形容词性隐喻的识别任务中表现很好。Rei等<sup>[16]</sup>利用有监督的语义相似网络进行隐喻识别,在网络添加门控机制以提取语义特征,该方法比普通的前馈神经网络(FNN)效果更好。上述隐喻识别方法的缺点是神经网络结构比较简单,还可以向更复杂的神经网络架构扩展。

汉语隐喻研究的起步较晚,大部分研究集中在符号规则 and 传统机器学习。王治敏等<sup>[17]</sup>利用最大熵(ME)模型识别中文名词性隐喻。李斌等<sup>[18]</sup>结合ME和CRF的方法识别中文明喻。黄孝喜<sup>[19]</sup>使用依存句法为不同隐喻模式建模,提出一种模式匹配算法识别中文隐喻。将深度学习应用于中文隐喻方面的工作非常少,黄孝喜等<sup>[20]</sup>结合CNN和SVM的方式识别中英文隐喻,并取得了较好的效果。

综上,隐喻识别的主要工作集中在传统机器学习和简单的神经网络,并且大部分研究集中在英文隐喻上。因此亟需中文隐喻识别技术上的突破。本文针对中英文隐喻识别问题,使用词嵌入融合技术和更复杂的神经网络来建模,进一步拓展和改进已有方法。

## 3 研究方法

### 3.1 基于词嵌入的语义优先中断理论

词嵌入的本质是将词表示成在分布式语义空间

<sup>①</sup><https://translate.google.cn>.

中的向量形式,语义越相近的词在向量空间的距离越小。Mikolov等<sup>[21]</sup>提出经典的词嵌入模型 Word2Vec,包括两种训练词嵌入的建模方法,分别是连续词袋模型 CBOW 和跨语法模型 Skip-gram。CBOW 通过上下文来预测中心词,而 Skip-gram 相反,通过中心词来预测上下文。通过 Word2Vec 模型可以得到具有语义特征的向量形式的词嵌入,设  $E_{w2v}$  是由 Word2Vec 训练得到的词嵌入向量集合,  $E_{w2v}(w_i)$  表示单词  $w_i$  在  $E_{w2v}$  中的向量。Mikolov 等<sup>[22]</sup>指出词向量在经过向量运算后的结果向量依然保持语义特征,例如“国王”减去“男人”的词向量等于“皇后”减去“女人”的词向量,如图 1 所示。

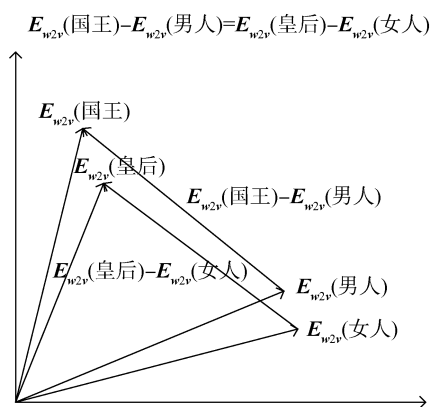


图 1 词嵌入向量运算示意  
Fig.1 Schematic Diagram of Word Embedding Vector Operation

根据语义优先中断理论的原理和词嵌入的特性提出基于词嵌入的语义优先中断理论,即词嵌入可以包含隐喻的概念并且符合语义优先中断理论。例如在隐喻句“发如雪”中“发”和“雪”是两个语义相差很大的概念,在词向量空间中会触发语义中断,“白”和“雪”是经常搭配的词,在词向量空间中存在相近的关系,“头发”与“雪”并非语义相近,但“头发”和“白”的词向量和会比“头发”与“雪”的距离更近,如图 2 所示。“头发”与“白”的词向量相加会减少“头发”与“雪”的词向量距离。源域“雪”与目标域“头发”通过隐喻意“白”的关联,从而在词嵌入空间中符合一定的特征规律,即隐喻句和非隐喻句在词嵌入空间会有不同的特征模式,隐喻识别实质是在词嵌入空间中的模式识别问题。

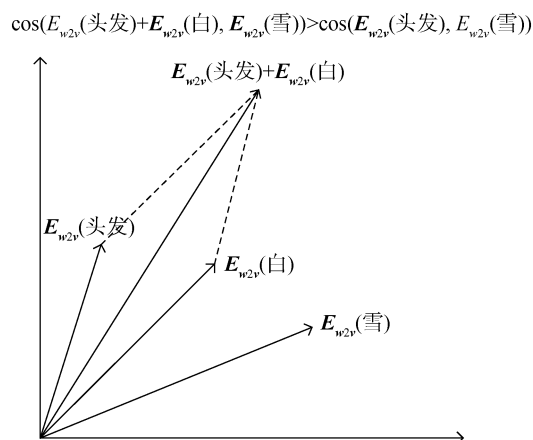


图 2 通过词嵌入解释隐喻示意  
Fig.2 Metaphor Interpretation with Word Embeddings

### 3.2 基于词嵌入融合隐喻特征加强

根据基于词嵌入的语义优先中断理论,隐喻句和非隐喻句在词嵌入空间会有不同的特征模式。这种模式在经过不同语料库训练的词嵌入中会有一定区别,例如维基百科、百度百科语料的语言风格会偏向官方严谨;推特、微博语料的语言风格会偏向轻松大众化;财经新闻类语料的语言风格会偏向金融领域。通过不同语料训练而来的词嵌入的隐喻特征也会有一定区别,笔者认为将这些通过不同语料训练的词向量融合后的词嵌入的隐喻特征更加精确。这就像不同人对同一事物有不同理解,将这些人的理解总结在一起会使结论更加客观。Coates 等<sup>[23]</sup>提出元嵌入的概念,元嵌入是最基础的词嵌入,通过对元嵌入的线性平均可以增强词嵌入的性能。将这个理论进行泛化,使之更加通用并且可以对隐喻特征进行增强。令  $E_A(w)$ ,  $E_B(w)$  分别为词  $w$  在词嵌入集合  $A$ ,  $B$  中的词向量,且两个词向量可以拥有不同的维度,如果在一个隐喻句中两个词  $w_1$  和  $w_2$  的搭配触发语义中断,假设隐喻识别器  $M$  可以捕获这个语义中断进而识别隐喻,可以描述如公式(1)和公式(2)所示。

$$M(E_A(w_1), E_A(w_2)) = \text{TRUE} \quad (1)$$

$$M(E_B(w_1), E_B(w_2)) = \text{TRUE} \quad (2)$$

根据基于词嵌入的语义优先中断理论,语义中断由隐喻特征决定,隐喻特征又由词向量间的距离

决定,通过 $L_2$ 范数衡量向量间的距离,词 $w_1$ 和 $w_2$ 在词嵌入空间 $A$ 和 $B$ 中的距离分别如公式(3)和公式(4)所示。

$$D_A(w_1, w_2) = \|E_A(w_1) - E_A(w_2)\|_2 \quad (3)$$

$$D_B(w_1, w_2) = \|E_B(w_1) - E_B(w_2)\|_2 \quad (4)$$

将词嵌入空间 $A$ 和 $B$ 中对应词向量融合映射到空间 $C$ ,具体过程如公式(5)和图3所示。

$$E_C(w) = mE_{A_2}(w) + nE_{B_2}(w) \quad (5)$$

其中, $E_{A_2}(w)$ 、 $E_{B_2}(w)$ 分别是 $E_A(w)$ 、 $E_B(w)$ 补零对齐后的向量, $m$ 和 $n$ 是两个向量的融合系数, $d_1$ 和 $d_2$ 分别是 $w$ 在初始词向量空间 $A$ 和 $B$ 中的维度, $k_1$ 和 $k_2$ 分别是 $E_A(w)$ 、 $E_B(w)$ 的补零个数, $E_{A_2}(w)$ 是 $E_A(w)$ 前 $k_2$ 个数值, $E_{B_2}(w)$ 是 $E_B(w)$ 后 $k_1$ 个数值, $mE_{A_2}(w) + nE_{B_2}(w)$ 是 $E_{A_2}(w)$ 和 $E_{B_2}(w)$ 的非零数值相加部分。

由于目前开源的经过大规模语料预训练的词嵌入的维度大部分都是300维,本文主要研究两种典

$$\begin{aligned} D_C(w_1, w_2) &= \left\| \begin{bmatrix} E_A(w_1) \\ E_B(w_1) \end{bmatrix} - \begin{bmatrix} E_A(w_2) \\ E_B(w_2) \end{bmatrix} \right\|_2 = \left\| (E_{A_2}(w_1) + E_{B_2}(w_1)) - (E_{A_2}(w_2) + E_{B_2}(w_2)) \right\|_2 \\ &= \left\| (E_{A_2}(w_1) - E_{A_2}(w_2)) - (E_{B_2}(w_1) - E_{B_2}(w_2)) \right\|_2 = \left\| (E_A(w_1) - E_A(w_2)) - (E_B(w_1) - E_B(w_2)) \right\|_2 \\ &= \sqrt{(D_A(w_1, w_2))^2 + (D_B(w_1, w_2))^2 - 2D_A(w_1, w_2)D_B(w_1, w_2)\cos(\theta)} \\ &= \sqrt{(D_A(w_1, w_2))^2 + (D_B(w_1, w_2))^2 - 2D_A(w_1, w_2)D_B(w_1, w_2)(0)} \\ &= \sqrt{(D_A(w_1, w_2))^2 + (D_B(w_1, w_2))^2} \end{aligned}$$

其中, $\theta$ 是 $E_A(w)$ 和 $E_B(w)$ 的夹角,因为 $E_A(w)$ 和 $E_B(w)$ 在这种情况下是正交的,所以 $\cos(\theta)$ 值为零,因此 $w_1$ 和 $w_2$ 在词嵌入融合空间 $C$ 中的距离是在 $A$ 、 $B$ 中距离平方和的根,这表明词 $w_1$ 和 $w_2$ 在空间 $C$ 的距离由在空间 $A$ 和 $B$ 中的距离决定,隐喻识别器 $M$ 如果能识别词 $w_1$ 和 $w_2$ 在空间 $A$ 和 $B$ 中触发语义中断而产生的隐喻,那么 $M$ 仍然可以识别词 $w_1$ 和 $w_2$ 在词嵌入融合空间 $C$ 中触发语义中断而产生的隐喻,定义如公式(7)所示。

$$M(E_C(w_1), E_C(w_2)) = \text{TRUE} \quad (7)$$

(2) 第二种典型情况,令 $k_1 = k_2 = 0$ ,  $m = n = 0.5$ ,融合后的词嵌入定义如公式(8)所示。

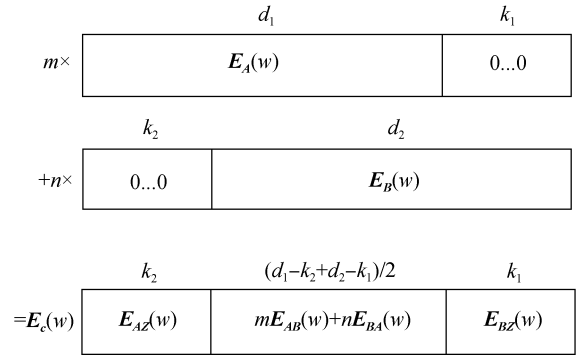


图3 词嵌入融合示意

Fig.3 Schematic Diagram of Word Embedding Combination

型情况下的词嵌入融合,因此设 $d_1 = d_2 = 300$ 。

(1) 第一种典型情况,令 $k_1 = k_2 = 300$ ,  $m = n = 1$ ,则融合后的词嵌入如公式(6)所示。

$$E_C(w) = [E_A(w), E_B(w)]^{-1} \quad (6)$$

词 $w_1$ 和 $w_2$ 在词嵌入空间 $C$ 中的距离推导过程如下所示。

$$E_C(w) = [0.5 \times E_A(w), 0.5 \times E_B(w)]^{-1} \quad (8)$$

在这种情况下 $w_1$ 和 $w_2$ 在词嵌入融合空间 $C$ 中的距离正比于在 $A$ 、 $B$ 中距离平方和的根,因此也符合同样结论,即隐喻识别器 $M$ 仍然可以识别在词嵌入融合空间 $C$ 中的隐喻特征。两个以上的词嵌入融合可以看作不断地用两个词向量递归融合,结果也符合上述规律,推导过程相似,这里不再赘述。

综上,词嵌入融合后依然包含隐喻特征,且将多个词嵌入融合后会使融合词嵌入拥有更丰富的语义信息,因此基于词嵌入融合的隐喻特征增强是理论上可行的。

### 3.3 基于循环神经网络的隐喻识别器

RNN因其递归的结构对于序列结构数据特别



有效,自然语言正是这样的序列数据,并且语言本身具有递归结构的性质。然而,普通的RNN有梯度消失和梯度爆炸的问题,RNN的变体长短时记忆(LSTM)能够有效缓解这些问题。LSTM是带有门控机制的RNN,分别是输入门 $i_t$ 、遗忘门 $f_t$ 、记忆门 $c_t$ 和输出门 $o_t$ ,其中 $t$ 为时序,LSTM的原理公式(9)-公式(13)所示<sup>[13]</sup>。

$$f_t = \sigma_f(x_t W_{xf} + h_{t-1} W_{hf} + \omega_{cf} \odot c_{t-1} + b_f) \quad (9)$$

$$i_t = \sigma_i(x_t W_{xi} + h_{t-1} W_{hi} + \omega_{ci} \odot c_{t-1} + b_i) \quad (10)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(x_t W_{xc} + h_{t-1} W_{hc} + b_c) \quad (11)$$

$$o_t = \sigma_o(x_t W_{xo} + h_{t-1} W_{ho} + \omega_{co} \odot c_t + b_o) \quad (12)$$

$$h_t = o_t \odot \sigma_h(c_t) \quad (13)$$

其中, $\sigma$ 代表Sigmoid函数, $\odot$ 为Hadamard乘积, $h_t$ 是隐藏状态。双向LSTM(Bi-LSTM)使用两个LSTM分别处理正序和逆序的语言序列,最后将输出结果拼接在一起,因此Bi-LSTM能够兼顾语言的上下文语境,更加符合自然语言的语境特征<sup>[13,24]</sup>。使用Bi-LSTM构建隐喻识别器,笔者设计的端到端的基于词嵌入融合和循环神经网络的隐喻识别器(Metaphor Recognizer Based on Word Embedding Combination and Recurrent Neural Network, ECMR)如图4所示。

其中,词嵌入层和词嵌入融合层的作用是语义特征表示,将自然语言转化为计算机可以理解的离散数值形式,语义特征中包含隐喻特征。将这些语义特征表示输入双层Bi-LSTM,再通过注意力层、最大池化层和平均池化层过滤不相干的语义特征,提取与隐喻相关的特征。其中注意力层使用的是在机器翻译任务中效果非常好的Bahdanau注意力,其本质是语义对齐<sup>[25]</sup>。最后通过Sigmoid函数进行分类,输出目标语句为隐喻句的概率,设置一个隐喻阈值,概率超过这个阈值输出为1表示该语句为隐喻句,反之输出为0则表示非隐喻句。隐喻识别任务的训练目标函数是交叉熵损失函数,如公式(14)所示。

$$J = -\frac{1}{N} \sum_{i=0}^N (y'_i \log(y_i) + (1 - y'_i) \log(1 - y_i)) \quad (14)$$

其中, $N$ 代表训练文本样本数量, $y'_i$ 代表第 $i$ 个文本样本是否为隐喻的真实标签, $y_i$ 是隐喻识别器根据输入文本样本的预测值。通过训练不断优化这个损失函数 $J$ 。

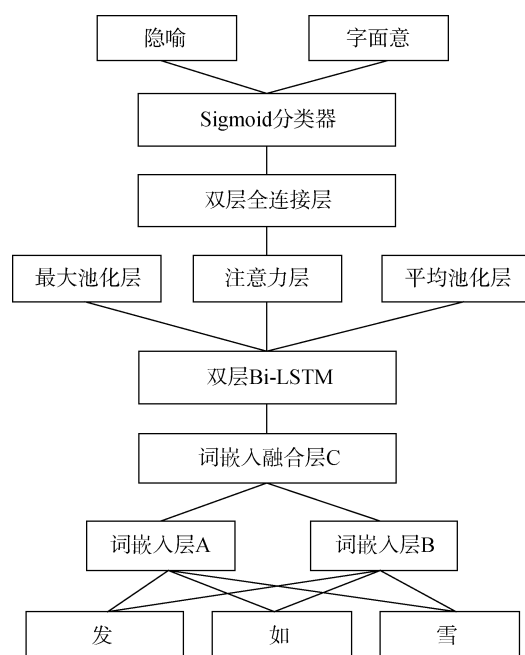


图4 基于循环神经网络的隐喻识别器架构

Fig.4 Architecture of Metaphor Recognizer Based on RNN

## 4 实验及结果分析

### 4.1 中英文隐喻数据集

在隐喻研究中合适的高质量数据集是非常重要的,本文的隐喻识别研究分别在中英文数据集上进行。英文隐喻数据集比较丰富,动词性隐喻数据集选择标注质量较高且数据类别相对平衡的隐喻数据集TroFi,该数据集由Birke等<sup>[26]</sup>标注,有3737条英文数据,隐喻数据占43%;形容词性隐喻选择由Tsvetkov等<sup>[2]</sup>标注的AN隐喻数据集,包含1768条标注质量较高的形容词名词短语结构的数据,隐喻数据占50%。中文隐喻研究起步较晚,因此高质量的标注数据集相对较少,采用TroFi数据集和AN数据集的英译中版本,通过谷歌翻译系统对TroFi数据集和AN数据集进行翻译,然而翻译系统有时不能正确理解隐喻从而产生错误的翻译,因此对初步翻译的语料库进行人工检查纠正。统计发现,大部分的句子可以得到正确翻译,AN中译数据集(AN\_CN)约3%的句子翻译出现问题,TroFi中译数据集(TroFi\_CN)约1%的句子翻译出现问题,例如将“hard poverty”的隐喻意“艰难的贫困状态”翻译为字

面意“硬贫困”，笔者将这些句子纠正以保证英文的隐喻句对应的中文翻译仍然是隐喻句。中英文隐喻数据集的训练集、验证集、测试集划分比例分别为80%、10%和10%。

#### 4.2 实验细节和超参数设置

本文在中文隐喻方面使用三个由 Word2Vec 的 Skip-gram 算法预训练的词嵌入,分别由百度百科、搜狗新闻和微博的语料训练。在英文隐喻方面使用三个相差较大的词向量训练算法预训练的词嵌入,分别是 GloVe<sup>[27]</sup>、FastText<sup>[28]</sup>、Paraphrase<sup>[29]</sup>。使用的预训练词嵌入维度都是300维。英文文本使用 SpaCy 框架(<https://spacy.io>)进行大小写转换、词干提取、去除标点等数据清洗工作以提高词映射到词向量的概率,并且缓解词表溢出(Out Of Vocabulary, OOV)问题。中文文本需要分词处理再进行词向量映射,使用 Jieba 框架(<https://github.com/fxsjy/jieba>)完成中文分词。将文本转化为词嵌入后进行比率为0.3的空间 Dropout 正则化操作防止过拟合,再通过双层 Bi-LSTM 提取初级语义特征,其中第一层 Bi-LSTM 有256个神经元,第二层 Bi-LSTM 有128个神经元;分别用注意力层、最大池化层和平均池化层提取不同特征的高级语义,再将这些含有高级语义特征的输出拼接在一起通过双层全连接层来提取隐喻特征,第一层全连接层有64个神经元,激活函数为整流线性单元(ReLU),并且通过比率为0.1的 Dropout 正则化操作,第二层全连接层只有一个神经元,激活函数为 Sigmoid 函数,输出该文本为隐喻的概率。训练隐喻识别器的优化器是能够自适应学习率的 Adam 优化器,初始学习率设置为0.001。

#### 4.3 实验结果对比与分析

一共进行两组实验:英文隐喻识别和中文隐喻识别。为表述简洁,定义以下符号:MR 代表笔者设计的使用普通词嵌入隐喻识别器,EC-MR 代表使用词嵌入融合的词嵌入融合隐喻识别器, $G$  代表 GloVe 词嵌入, $F$  代表 FastText 词嵌入, $P$  代表 Paraphrase 词嵌入, $B$  代表由百度百科语料训练的 Word2Vec 词嵌入, $S$  代表由搜狗新闻语料训练的 Word2Vec 词嵌入, $W$  代表由微博语料训练的 Word2Vec 词嵌入。 $M1(A, B)$  代表在  $k_1 = k_2 = 300, m = n = 1$  参数下词嵌入  $A$  和词嵌入  $B$  融合; $M2(A, B)$  代表在  $k_1 = k_2 = 0, m = n = 0.5$  参数下

词嵌入  $A$  和词嵌入  $B$  融合;当  $M1$  和  $M2$  超过两个参数时表示按顺序融合多个词嵌入。为防止偶然因素的干扰,实验采用十折交叉验证来充分利用训练集,并且通过运行10次实验取平均值获得最终结果。使用在隐喻识别任务常用的评测指标准确率(Accuracy)、精确率(Precision)、召回率(Recall)和 F1 值评测隐喻识别器性能,其中准确率和 F1 值综合考虑了精确率和召回率两项指标,因此准确率和 F1 值更具有参考意义。本文的对比系统是 Gao 等<sup>[30]</sup>使用单层 Bi-LSTM 构建的利用隐喻序列标注特征的 SEQ 模型和 Rei 等<sup>[16]</sup>使用浅层神经网络构建的用于隐喻识别的监督相似网络(SSN),这两个系统在隐喻识别任务中取得了非常优越的结果。

英文隐喻识别的结果如表1所示,各个评测指标最佳结果用加粗表示。

表1 隐喻识别器在英文隐喻识别任务上的表现(%)

Table 1 Performance of Metaphor Recognizer in English Task (%)

模型	词嵌入方式	数据集	准确率	精确率	召回率	F1 值
MR	$G$	TroFi	65.3	56.1	88.5	67.7
MR	$F$	TroFi	62.0	58.5	83.6	67.3
MR	$P$	TroFi	63.6	54.0	85.9	66.3
EC-MR	$M1(G, P)$	TroFi	<b>73.8</b>	66.3	75.6	70.7
EC-MR	$M1(G, F, P)$	TroFi	71.1	63.3	73.1	67.9
EC-MR	$M2(G, P)$	TroFi	66.8	56.9	84.6	68.0
EC-MR	$M2(G, F, P)$	TroFi	64.2	54.3	<b>88.6</b>	67.3
SEQ	-	TroFi	73.7	<b>68.7</b>	76.4	<b>72.0</b>
MR	$G$	AN	84.2	76.0	95.0	84.3
MR	$F$	AN	80.9	79.5	77.5	78.5
MR	$P$	AN	83.1	76.6	90.0	82.8
EC-RNN	$M1(G, P)$	AN	83.1	77.8	87.5	82.4
EC-RNN	$M1(G, F, P)$	AN	82.0	74.0	92.5	82.2
EC-RNN	$M2(G, P)$	AN	84.3	75.0	<b>97.5</b>	<b>84.8</b>
EC-RNN	$M2(G, F, P)$	AN	<b>86.5</b>	86.8	82.5	84.6
SSN	-	AN	82.9	<b>90.3</b>	73.8	81.1

基于词嵌入融合方式的隐喻识别器准确率和 F1 值普遍优于基于普通词嵌入的隐喻识别器,准确率和 F1 值最高可以提高 11.8% 和 6.3%。实验结果符合预期,在 TroFi 数据集的实验中, $M1$  融合方式的词嵌入融合可以有效提升准确率,其中  $M1(G, P)$  融合方式的准确率和基于额外词法信息的 SEQ 模型相当,但隐喻识别器的优势是不依赖任何额外构建

的语言学特征。 $M1$  融合方式带来准确率提升的原因是这种融合方式会增加词嵌入的维度,比普通词嵌入携带更多语义特征,但随着词嵌入的维度增加会带来召回率降低,比如  $M1(G, F, P)$  融合方式,这是由于词嵌入维度的增加会给隐喻识别器的隐喻特征提取带来困难。在 AN 数据集上的实验表明,  $M2$  融合方式效果比  $M1$  融合方式要好,  $M2$  融合方式通过平均两个词嵌入混合不同词嵌入的语义特征,会携带比普通词嵌入更多的语义特征且不会增加词嵌入维度,准确率和 F1 值都超过 SSN 模型,这是因为 SSN 模型仅仅使用浅层神经网络,隐喻特征提取能力不足。从两个英文数据集上的实验结果来看,隐喻识别器在 TroFi 数据集的表现总体要差于在 AN 数据集上的表现,这是因为 AN 数据集是简单的形容词名词短语结构数据,各个测评指标都较高,而 TroFi 数据集是完整的句子,句子的长度较长且结构复杂,给隐喻识别器带来隐喻特征提取上的难度,造成测评指标普遍偏低。例如在句子“Jaime Rosenthal said he was stepping down as President Azcona’s economic adviser, but staying on as vice president.”中,隐喻识别器不能很好地识别“stepping down as President Azcona’s economic adviser”这个隐喻现象,这是因为长句句式结构复杂且语义优先中断通过较长的词组触发,给隐喻识别器带来干扰。

中文隐喻识别的结果如表 2 所示,各个评测指标最佳结果用加粗表示。在中文数据集上的实验结果和英文数据集类似,准确率和 F1 值最高可以提高 8.9% 和 7.8%,  $M1(B, S)$  融合方式在 TroFi\_CN 数据集上的综合表现最好,  $M2(B, S, W)$  融合方式在 AN\_CN 数据集上表现最好。隐喻识别器在中文数据集上的各项测评指标比英文数据集要差一点,这是因为对英文数据集的翻译会造成部分语义信息的损失,且中文数据集数据处理上还有分词这一步骤,分词的正确率本身达不到百分之百,造成的误差会进一步影响到隐喻识别的效果。

综合中英文隐喻识别实验结果,得到三个主要的结论:

(1) 基于词嵌入融合方式的隐喻识别器各个评测指标普遍优于基于普通词嵌入的方式,这表明融合词嵌入方式带来的额外语义特征有助于提升模型

表 2 隐喻识别器在中文隐喻识别任务上的表现(%)

Table 2 Performance of Metaphor Recognizer in Chinese Task (%)

模型	词嵌入方式	数据集	准确率	精确率	召回率	F1
MR	$B$	TroFi_CN	58.3	50.0	96.1	65.8
MR	$S$	TroFi_CN	59.9	51.6	85.9	64.1
MR	$W$	TroFi_CN	58.8	50.3	91.4	66.4
EC-MR	$M1(B, S)$	TroFi_CN	<b>61.5</b>	<b>52.1</b>	94.9	<b>67.3</b>
EC-MR	$M1(B, S, W)$	TroFi_CN	61.0	51.7	<b>96.2</b>	67.2
EC-MR	$M2(B, S)$	TroFi_CN	59.9	50.1	93.6	66.1
EC-MR	$M2(B, S, W)$	TroFi_CN	59.4	50.7	92.3	65.5
MR	$B$	AN_CN	84.3	82.5	82.4	82.5
MR	$S$	AN_CN	77.5	71.7	82.4	76.7
MR	$W$	AN_CN	84.3	82.6	84.5	83.3
EC-MR	$M1(B, W)$	AN_CN	85.4	84.6	82.5	83.5
EC-MR	$M1(B, S, W)$	AN_CN	85.4	86.5	80.0	83.1
EC-MR	$M2(B, W)$	AN_CN	85.4	82.9	<b>85.0</b>	84.0
EC-MR	$M2(B, S, W)$	AN_CN	<b>86.4</b>	<b>86.8</b>	82.6	<b>84.5</b>

的隐喻识别能力。

(2) 对比隐喻识别器和 SEQ 模型的实验结果发现,仅使用词嵌入特征的隐喻识别器就能达到依赖其他语言学特征的隐喻识别器相当的效果,这表明词嵌入特征隐含丰富的语言学特征,深度学习模型有能力挖掘这些特征来进行深层语义分析。

(3) 对比隐喻识别器和 SSN 模型的实验结果发现,基于深度神经网络的隐喻识别器比基于的浅层神经网络拥有更强的隐喻识别能力,这表明深度神经网络拥有更强大的隐喻特征提取能力。

## 5 结 语

本文提出基于词嵌入的语义优先中断理论,论述了在词嵌入空间捕获隐喻特征的可行性。同时提出通过词嵌入融合的方法增强隐喻特征,阐述了基于词嵌入融合的隐喻识别器识别语言中隐喻现象的数学原理并构建了一个基于循环神经网络的隐喻识别器。在中英文隐喻数据集上的实验结果表明该隐喻识别器具有很好的隐喻识别效果,且基于词嵌入融合方式的隐喻识别器效果优于基于普通词嵌入的隐喻识别器。

根据中英文隐喻识别实验结果,隐喻识别方法和模型还有很多需要优化的地方。针对复杂句式长



语句的隐喻识别效果差,未来将使用词法、句法、语义等更多的语言学特征来辅助隐喻识别器识别复杂句式长语句中的隐喻现象。在语料库上,未来将针对中文隐喻处理数据集的短板构建一个大规模高质量的中文隐喻语料库以促进中文隐喻处理技术的发展。在模型结构上,未来将尝试构建效果更优性能更好的隐喻识别器,基于注意力机制的Transformer<sup>[31]</sup>模型和预训练语言模型<sup>[32]</sup>在多项NLP任务上的卓越效果也给笔者带来很大启示,未来会将这些先进的领域前沿技术应用到隐喻计算相关任务上。

## 参考文献:

- [1] Lakoff G, Johnson M. Conceptual Metaphor in Everyday Language[J]. The Journal of Philosophy, 1980, 77(8): 453-486.
- [2] Tsvetkov Y, Boytsov L, Gershman A, et al. Metaphor Detection with Cross-Lingual Model Transfer[C]// Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. 2014, 1: 248-258.
- [3] McArthur T, Lam-McArthur J, Fontaine L. Oxford Companion to the English Language[M]. Oxford University Press, 2018.
- [4] Shutova E. Design and Evaluation of Metaphor Processing Systems[J]. Computational Linguistics, 2015, 41(4): 579-623.
- [5] Mao R, Lin C, Guerin F. Word Embedding and WordNet Based Metaphor Identification and Interpretation[C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. 2018: 1222-1231.
- [6] Wilks Y. A Preferential , Pattern-seeking, Semantics for Natural Language Inference[A]// Ahmad K, Brewster C, Stevenson M. Words and Intelligence I[M]. Springer, 2007: 83-102.
- [7] Fass D. Met\*: A Method for Discriminating Metonymy and Metaphor by Computer[J]. Computational Linguistics, 1991, 17(1): 49-90.
- [8] Neuman Y, Assaf D, Cohen Y, et al. Metaphor Identification in Large Texts Corpora[J]. PLoS One, 2013, 8(4): e62343.
- [9] Shutova E, Sun L, Korgonen A. Metaphor Identification Using Verb and Nouns Clustering[C]// Proceedings of the 23rd International Conference on Computational Linguistics. 2010: 1002-1010.
- [10] Hovy D, Srivastava S, Jauhar S K, et al. Identifying Metaphorical Word Use with Tree Kernels[C]// Proceedings of the 1st Workshop on Metaphor in NLP. 2013: 52-57.
- [11] Rai S, Chakraverty S, Tayal D K. Supervised Metaphor Detection Using Conditional Random Fields[C]// Proceedings of the 4th Workshop on Metaphor in NLP. 2016: 18-27.
- [12] Kalchbrenner N, Grefenstette E, Blunsom P. A Convolutional Neural Network for Modelling Sentences[OL]. arXiv Preprint, arXiv: 1404.2188.
- [13] Graves A, Schmidhuber J. Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures [J]. Neural Networks, 2005, 18(5-6):602-610.
- [14] Do Dinh E L, Gurevych I. Token-Level Metaphor Detection Using Neural Networks[C]// Proceedings of the 4th Workshop on Metaphor in NLP. 2016: 28-33.
- [15] Bizzoni Y, Chatzikyriakidis S, Ghanimifard M. “Deep” Learning: Detecting Metaphoricity in Adjective-Noun Pairs[C]// Proceedings of the 2017 Workshop on Stylistic Variation. 2017: 43-52.
- [16] Rei M, Bulat L, Kiela D, et al. Grasping the Finer Point: A Supervised Similarity Network for Metaphor Detection[OL]. arXiv Preprint, arXiv:1709.00575.
- [17] 王治敏, 王厚峰, 俞士汶. 基于机器学习方法的汉语名词隐喻识别[J]. 高技术通讯, 2006, 17(6): 575-580. (Wang Zhimin, Wang Houfeng, Yu Shiwen. Chinese Nominal Metaphor Recognition Based on Machine Learning[J]. Chinese High Technology Letters, 2006, 17(6): 575-580.)
- [18] 李斌, 于丽丽, 石民, 等. “像”的明喻计算[J]. 中文信息学报, 2008, 22(6): 27-32. (Li Bin, Yu Lili, Shi Min, et al. Computation of Chinese Simile with “Xiang” [J]. Journal of Chinese Information Processing, 2008, 22(6): 27-32.)
- [19] 黄孝喜. 隐喻机器理解的若干关键问题研究[D]. 杭州: 浙江大学, 2009. (Huang Xiaoxi. Research on Some Key Issues of Metaphor Computation[D]. Hangzhou: Zhejiang University, 2009.)
- [20] 黄孝喜, 李晗雨, 王荣波, 等. 基于卷积神经网络与 SVM 分类器的隐喻识别[J]. 数据分析与知识发现, 2018, 2(10): 77-83. (Huang Xiaoxi, Li Hanyu, Wang Rongbo, et al. Recognizing Metaphor with Convolution Neural Network and SVM[J]. Data Analysis and Knowledge Discovery, 2018, 2(10): 77-83.)
- [21] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[OL]. arXiv Preprint, arXiv: 1301.3781.
- [22] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and Their Compositionality[C]// Proceedings of the 26th International Conference on Neural Information Processing Systems. 2013: 3111-3119.
- [23] Coates J, Bollegala D. Frustratingly Easy Meta-Embedding—Computing Meta-Embeddings by Averaging Source Word Embeddings[OL]. arXiv Preprint, arXiv:1804.05262.
- [24] Graves A, Fernández S, Schmidhuber J. Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition [C]// Proceedings of the 15th International Conference on Artificial Neural Networks: Formal Models and Their Applications. Springer, 2005: 799-804.
- [25] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[OL]. arXiv Preprint, arXiv:1409.0473.
- [26] Birke J, Sarkar A. A Clustering Approach for Nearly Unsupervised Recognition of Nonliteral Language[C]// Proceedings of the 11th



Conference of the European Chapter of the Association for Computational Linguistics. 2006.

- [27] Pennington J, Socher R, Manning C. GloVe: Global Vectors for Word Representation[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. 2014: 1532-1543.
- [28] Joulin A, Grave E, Bojanowski P, et al. Bag of Tricks for Efficient Text Classification[OL]. arXiv Preprint, arXiv:1607.01759.
- [29] Wieting J, Bansal M, Gimpel K, et al. From Paraphrase Database to Compositional Paraphrase Model and Back[J]. Transactions of the Association for Computational Linguistics, 2015, 3: 345-358.
- [30] Gao G, Choi E, Choi Y, et al. Neural Metaphor Detection in Context[OL]. arXiv Preprint, arXiv:1808.09653.
- [31] Vaswani A, Shazeer N, Parmar N, et al. Attention is All You Need [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017: 5998-6008.
- [32] Radford A, Narasimhan K, Salimans T, et al. Improving Language Understanding by Generative Pre-training[OL]. <https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf>.

### 作者贡献声明:

苏传东:设计研究方案,撰写论文;  
 苏传东,黄孝喜:提出研究思路,设计研究方案,论文最终版本修订;  
 王荣波,湛志群:修改论文,总结实验结论,论文最终版本修订;  
 苏传东,毛君钰:设计实验方案,采集、清洗和分析实验数据,中英文

数据集整理;

朱嘉莹,潘宇豪:进行实验,记录实验过程,分析实验结果。

### 利益冲突声明:

所有作者声明不存在利益冲突关系。

### 支撑数据

支撑数据由作者自存储,E-mail:suchuandong@hdu.edu.cn。

- [1] 苏传东. TroFi\_english.csv. TroFi 动词隐喻数据集.
- [2] 苏传东. TroFi\_chinese.csv. TroFi 动词隐喻数据集中译版本.
- [3] 苏传东. AN\_english.csv. AN 形容词隐喻数据集.
- [4] 苏传东. AN\_chinese.csv. AN 形容词隐喻数据集中译版本.
- [5] 苏传东. GloVe\_840B\_300d.txt. GloVe 词嵌入.
- [6] 苏传东. Paragram\_sl999\_300d.txt. Paragram 词嵌入.
- [7] 苏传东. Fasttext\_2M\_300d.vec. FastText 词嵌入.
- [8] 苏传东. Word2Vec\_baidubaike.bigram-char. 百度百科语料训练的 Word2Vec 词嵌入.
- [9] 苏传东. Word2Vec\_sogounews.bigram-char. 搜狗新闻语料训练的 Word2Vec 词嵌入.
- [10] 苏传东. Word2Vec\_weibo.bigram-char. 微博语料训练的 Word2Vec 词嵌入.

收稿日期:2019-07-12

收修改稿日期:2019-09-12

## Identifying Chinese / English Metaphors with Word Embedding and Recurrent Neural Network

Su Chuandong Huang Xiaoxi Wang Rongbo Chen Zhiquan Mao Junyu Zhu Jiaying Pan Yuhao  
 (Institute of Cognitive and Intelligent Computing, Hangzhou Dianzi University, Hangzhou 310018, China)

**Abstract:** [Objective] This paper proposes a method to recognize Chinese and English metaphors with word vector combination and recurrent neural network (RNN), aiming to identify the ubiquitous metaphors from natural languages. [Methods] First, we mapped texts to the word vectors as inputs of the neural network with the help of word-embedding combination algorithm. Then, we used the RNN as encoder, and took the attention mechanism and the pooling technique as feature extractor. Finally, we utilized Softmax to calculate the probability of the text was a metaphor. [Results] The accuracy and F1 of the proposed method with English texts improved by 11.8% and 6.3%, compared with traditional method based on vanilla word embedding. For Chinese tasks, the accuracy and F1 of the proposed method also improved by 8.9% and 7.8%. [Limitations] Due to the long-distance dependence issue, our method could not effectively recognize metaphors in long texts with complex sentences. [Conclusions] The proposed model significantly improves the neural network's ability to recognize metaphors.

**Keywords:** Metaphor Recognition Deep Learning Word Embedding Recurrent Neural Network