

文章编号: 1003-0077 (2017) 00-0000-00

基于多语义融合的反讽识别

摘要: 反讽是一种复杂的语言现象, 被广泛应用于社交媒体当中。如何让计算机具有识别反讽的能力成为了自然语言处理研究领域的热门研究内容之一。该文针对反讽识别中缺乏上下文语境信息和修辞表达信息的问题, 提出了基于多语义融合的反讽识别方法。该方法采用 ELMo 从大规模反讽文本中训练得到领域词嵌入表示, 并融合基于词性和基于风格信息的语义表示, 使用双向长短期记忆网络和卷积神经网络进行反讽识别。实验结果表明, 所提出模型能够从多个维度提取反讽文本的潜在语义特征, 在公开数据集 IAC 上的实验性能有显著提升。

关键词: 反讽识别; 多语义融合; 神经网络; ELMo;

中图分类号: TP391

文献标识码: A

Irony Recognition based on Multiple Semantic Fusion

Abstract: Irony is a complex linguistic phenomena that are widely used in social media. With the rapid development of internet, how to make a computer recognize irony has become one of the hot topics in the field of natural speech processing. Existing irony detection approaches suffer from lack of context and rhetoric information. To address this problem, irony recognition based on multiple semantic fusion is proposed. We use ELMo to train domain vector from large-scale ironic texts, and then combine the semantic representation based on syntactic structure and style information. Bi-direction Long Short-term Memory network and Convolutional Neural Network are used to recognize irony. Experimental results show that our proposed model can capture the latent semantic features of irony texts from multiple dimensions. The experimental performances of IAC dataset are significant improved.

Key words: irony detection; multiple semantic fusion; neural network; ELMo

0 引言

反讽^[1]是社交媒体中常见的修辞表达方式, 其所要表达的真实意义和字面含义相反, 是一种带有强烈感情色彩的修辞格, 通常含有否定、讽刺以及嘲弄的含义^[2]。随着社交媒体和人工智能的快速发展, 如何利用计算机技术识别反讽, 理解蕴含在反讽字面语义背后的真实情感表达成为了自然语言处理领域备受关注的研究内容之一。反讽识别在反语与讽刺理论的驱动下, 以认知语

言学为基础构建计算模型, 从而赋予了计算机从更深层次理解人类情感的能力, 为实现更加智能友好的人机交互环境奠定了研究基础。从应用角度来说, 反讽识别的深入研究能够促进情感分析的发展, 提高情感分析任务的性能。总之, 反讽识别在情感分析系统、人机对话系统、翻译系统中都有着广泛的应用前景。

反讽识别任务被归结为有监督的文本分类问题, 目标是判别一个句子是否包含反讽的含义。现有的反讽识别方法主要分为基于规则的方法^[3]、基于统计的方法^[4]和基于深度学习方法^[5-6]。

收稿日期: 2017-03-16; 定稿日期: 2017-04-26

六号

基金项目: 基金名 (基金号); 基金名 (基金号)

六号, 核实准确完整的基金名称

基于规则的方法以反讽的理论为支撑,能够反应讽刺语句的语言学特点,但是规则的制定需要领域专家,且迁移性较差。基于统计的方法以当前的语料为基础,能够反应反讽语句的统计学特征,但是特征的构建需要耗费大量的时间和精力,泛化能力较弱。深度学习方法以大规模语料训练得到的词嵌入表示为基础,能够反应反讽语句的高维潜在语义信息,但是传统的词嵌入表示不能反应同一单词在不同上下文语境中的差异,此外,目前的深度学习方法缺乏对讽刺语句修辞特征的特征,这使得将深度学习方法应用于反讽识别领域时,其性能受到了一定的限制。

本文提出了一种基于多语义融合的神经网络模型(EPSN)来识别反讽。考虑到反讽的语境特点,采用语言模型的词嵌入表示方法(ELMo)^[7]在大规模讽刺语料库上进行训练,得到了反讽相关的词嵌入表示。考虑到反讽的词性特点,对反讽语句的词性特征进行了嵌入表示。基于反讽的风格特点,从字符角度对反讽语句进行了建模。在多维语义表示的基础上,分别采用了双向长期记忆神经网络(Bi-LSTM)和卷积神经网络(CNN)提取反讽语句的高维潜在语义特征,从而对反讽语句进行识别。本文的贡献如下:

- 针对反讽语句同一单词在不同上下文中语境时其词嵌入表示差异较大的问题,本文采用了 ELMo 对大规模的反讽语料库进行了训练,得到了针对反讽语料的领域词嵌入表示。
- 针对反讽语句修辞表达的特点,本文采用了词性向量表示,并使用了 Bi-LSTM 提取反讽语句的词性特征。此外,本文采用了 CNN 提取反讽语句的字符向量特征,使模型能够获取反讽语句的修辞特征,从而进一步提高了模型识别反讽的性能。
- 本文提出的融合语义、词性和风格等多种特征的反讽识别模型在公开数据集 IAC_v2 的三个子数据集上进行了实验,实验结果表明,本文提出的基于多语义融合的反讽识别模型能够有效的提取隐藏在反讽语句背后的语义特征,提高反讽识别的性能。

1 相关工作

反讽作为一种修辞现象,受到语言学家和认知心理学家的广泛关注。随着情感分析技术的深入研究,反讽识别成为了自然语言处理领域的热点问题之一。对于反讽识别研究,我们根据使用的方法不同,从基于规则、基于统计的传统机器学习方法和基于深度学习的方法三个方面对前人的工作进行梳理。

基于规则的方法尝试通过制定一系列规则来捕捉反讽的语义特征。Veale 等^[8]指出反讽和幽默明喻的强关联性,构建了词汇选择、句法结构、语义等多种规则识别反讽。Maynard^[9]认为 hashtag 是推特上识别反讽的一项关键特征,并对反讽和情感文本同时进行分类识别。Bharti^[10]等提出了基于语法分析的词汇生成算法和基于感叹词的是否出现等规则来识别反讽。Riloff^[11]等提出了一种根据规则自动学习文本中的正负情感短语句的反讽识别方法。基于规则的反讽需要领域专家制定大量的规则,且规则之间容易出现冲突,其迁移性较差。

基于统计的传统机器学习方法通常基于词袋模型,根据反讽语言特点构建多种反讽相关特征,并使用传统机器学习方法进行反讽识别。Farias 等^[12]利用多种情感词典,挖掘句子中的情感特征,并基于语义相似度和情感相关性等语义特征识别反讽。Bamman 等^[13]指出推特中的作者特征,回复的文本内容等上下文特征对反讽识别起到了关键作用。Rayes 等^[14]对比了朴素贝叶斯和决策树算法在平衡和非平衡反讽数据集上的性能。Joshi 等人^[15]探索了 4 类词向量特征在反讽识别中的性能,作者指出词向量的引入能够大幅度提升性能。基于特征的传统机器学习方法需要耗费大量时间精力人工构建特征,且模型的泛化能力较弱。

深度学习方法是利用神经网络结构对输入的反讽语句进行逐层筛选和提取,获得具有表征能力的特征,从而对反讽文本进行识别。Amir 等^[16]对社交媒体中的用户信息进行了建模,构建了用户嵌入式表示,并采用 CNN 学习用户信息和上下文信息进行反讽识别。Zhang 等^[17]采用 Bi-LSTM 自动提取反讽文本中的句法和语义特

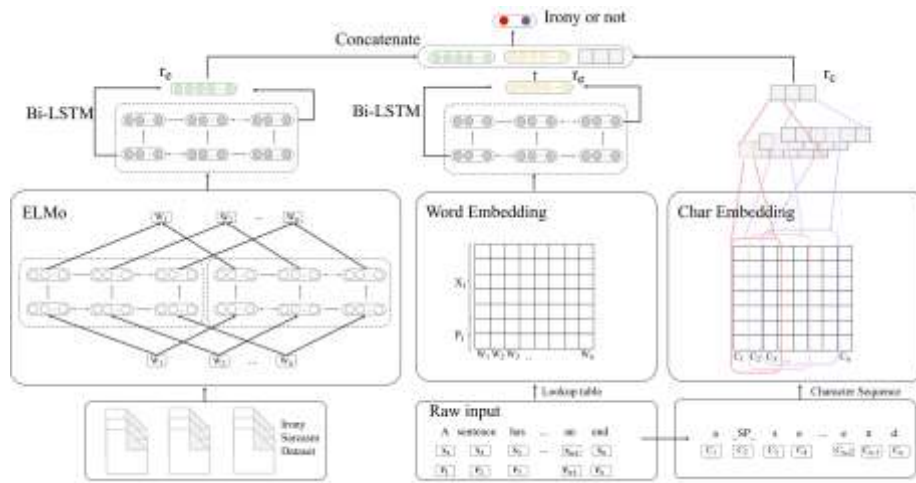


图 1 多语义融合的神经网络模型(EPSN)框架结构图

征, 并将目标推文的历史推文作为上下文信息引入模型进行反讽识别。Ghosh 等^[18]利用推特的上下文信息、用户信息和时间信息作为 CNN 和 LSTM 的输入, 实验结果表明, 该方法使得反讽识别的性能得到了大幅度的提升。Tay 等人^[19]指出反讽通常由于积极和消极的情感冲突或字面意与比喻意的情景矛盾而产生, 作者构建了一种基于注意力机制的神经网络模型识别反讽。张庆林等^[20]提出了一种对抗学习框架, 在模型训练的过程中应用对抗的思想生成样本, 从而提高了分类器的鲁棒性和泛化能力。目前, 传统的词嵌入表示已经不能满足现有反讽识别研究, 同时, 采用深度学习方法对于反讽语句修辞特征的建模还有待探索。

2 基于多语义融合的反讽识别

本小节将介绍本文提出的多语义融合的神经网络模型(EPSN)框架结构。基于多语义融合的反讽识别模型通过 ELMo 训练得到的词嵌入表示、反讽语句的词性嵌入表示和字符的嵌入表示等多种语义表示学习反讽语句的潜在语义特征, 从而提升反讽识别的性能。

多语义融合的神经网络模型(EPSN)框架结构如图 1 所示。首先, 通过 ELMo 模型, 利用大规模反讽语料库训练反讽领域相关的 ELMo 词嵌入表示。其次, 得到反讽语句的词性标签, 将它与普通词嵌入表示联合表示反讽语句的词性信息。通过 Bi-LSTM 模型提取与反讽强相关的语义特征。

再次, 将反讽语句表示为字符嵌入式表示, 利用 CNN 模型提取反讽语句的风格特征。最后, 将三部分语义特征融合得到反讽语句的最终表示并进行反讽识别。在接下来的部分将详细介绍每部分的具体实现。

2.1 基于语境的反讽语义表示

传统的词嵌入表示, 如 Word2Vec、GloVe 等, 给自然语言处理任务的性能带来了巨大的提升空间。但是它们通常是上下文无关的词嵌入表示。例如 Word2Vec 训练时, 使用单层的神经网络模型, 通过目标单词周围的若干个单词来预测目标单词, 但是周围单词的顺序并不重要, 预测结果可以认为只是统计意义上的最大概率。换句话说, 传统的词嵌入表示与句法结构, 特定语境、语义等关联较小。而反讽语句中, 语境和语义至关重要。例如:

Visits to a dentist are fun!

“fun”表示“有趣的”, 通常该单词表示积极的情感, 而且大多数情况下包含“fun”的句子不包含反讽的含义。但是该句子中, “看牙医”的情景让人们普遍认为是一件不愉快的事情, 因此该句为一个反讽语句。反讽语句的语境对反讽识别起着非常重要的作用, 本文希望得到的词嵌入表示包含语境信息, 从而更好的表征反讽语句。因此, 本文采用了 ELMo 词嵌入表示。

语言模型的词嵌入表示 (ELMo)

ELMo 是一种新型的深度语境化的词嵌入表示, 能够反应单词的复杂语义和语言环境信息。ELMo 的词嵌入表示是整个输入语句的函数, 它

采用了双向语言模型来对语句信息进行建模。

对于一个给定句子的单词序列 $S = (t_1, t_2, \dots, t_N)$, 利用目标单词 t_k 之前的单词序列 $(t_1, t_2, \dots, t_{k-1})$, 即前向的语言模型计算 t_k 的概率, 计算公式如下:

$$p(t_1, t_2, \dots, t_N) = \prod_{k=1}^N p(t_k | t_1, t_2, \dots, t_{k-1}) \quad (1)$$

类似的, 后向的语言模型公式如下:

$$p(t_1, t_2, \dots, t_N) = \prod_{k=1}^N p(t_k | t_{k+1}, t_{k+2}, \dots, t_N) \quad (2)$$

ELMo 采用多层的 LSTM 计算上下文无关的单词表示 x_k^{LM} 。对于输入 x_k^{LM} , 在每个位置 k , 每一层 LSTM 的隐层输出一个上下文无关的表示 \vec{h}_{kj}^{LM} , $j=1, 2, \dots, L$, 顶层的 LSTM 的输出 $\vec{h}_{k,L}^{LM}$ 通过 Softmax 来预测下一个单词 t_{k+1} 。结合前向和后向的语言模型, 计算最大似然估计的公式如下:

$$\sum_{k=1}^N (\log(p(t_k | t_1, \dots, t_{k-1}; \theta_x, \vec{\theta}_{LSTM}, \theta_S)) + \log(p(t_k | t_{k+1}, \dots, t_N; \theta_x, \vec{\theta}_{LSTM}, \theta_S)) \quad (3)$$

对于每一个单词, L 层的语言模型计算得到 $2L+1$ 的表示:

$$R_k = \{x_k^{LM}, \vec{h}_{kj}^{LM}, \vec{h}_{k,L}^{LM} | j=1, \dots, L\} \\ = \{h_{k,L}^{LM} | j=0, \dots, L\} \quad (4)$$

用以下公式计算特定任务的词嵌入表示:

$$ELMo_k^{task} = E(R_k; \theta^{task}) = \gamma^{task} \sum_{j=0}^L s_j^{task} h_{kj}^{LM}$$

由于 EIMo 学习词嵌入表示的过程不仅仅是通过目标词的窗口, 而是从整个语料库学习, 因此通过 EIMo 学习到词嵌入表示能够更好的表征特定任务的信息。本文采用了 Khodak 等^[21]构建的大规模反讽语料库进行模型的预训练, 得到反讽领域相关的词嵌入表示, 该词嵌入表示包含了反讽语料的复杂语义特征, 能够根据不同语境反应单词的语义信息。

2.2 基于词性信息的反讽语义表示

反讽与幽默相似, 是一种文体^[22], 通常具有特殊的词性表达。通过词嵌入表示, 模型能够自

动的学习反讽语句的语义特征, 但是获取的词性信息有限。

词性信息可以通过句子成分分析获取。利用斯坦福大学的句法分析工具, 将句子中的单词表示为对应的词性标签。为了使模型能够自动的学习词性标签的结构特征, 我们将词性标签转化为低维稠密向量表示。对于目标语句 $S = (x_1, x_2, \dots, x_N)$, $x_i \in \mathbb{R}^d$, x_i 为单词的词嵌入表示, d 为词嵌入表示的维度。对于句子 S , 词性标签的嵌入表示为 $P = (p_1, p_2, \dots, p_N)$, $p_i \in \mathbb{R}^{d_p}$, d_p 为词性嵌入表示的维度。为了更好的利用词性信息表征反讽语句, 本文对二者进行了融合。对于每个单词 x_i 和其词性 p_i , 计算 $x_i' = x_i \oplus p_i$, 从而得到带有词性特征的句子表示。

2.3 基于风格信息的反讽语义表示

风格是一种独特的表达方式, 很多情况下, 正是风格特征使得人们能够区分反讽文本^[2]。Reyes^[23]指出风格是指多次出现的文本元素序列, 这些反复出现的文本元素使得文本表现出相对稳定的特征, 从而使得人们有可能识别出文本的反讽特性。该文献指出字符级的词袋模型 (character n-grams) 能够有效的提高反讽识别的性能。

反讽语句常采用一些修辞方法表达出反讽的效果, 如押韵、重复字符、重复的标点符号等。例如:

You have a looooot of nerve !!!!!

该句是一个反讽的语句, 句子采用了比喻的方法表达了反讽。同时, 语句中采用了不规范的拼写形式“looooot”和多个感叹号加强了反讽的效果。

为了使模型能够捕获反讽语句的风格信息, 本文对反讽语句的字符进行建模, 将句子表示成字符的序列并最终得到句子的字符序列的嵌入表示。对于目标语句中的每个单词 x_i , 其字符嵌入表示为 $x_i = (c_1, c_2, \dots, c_M)$, 其中 $c_i \in \mathbb{R}^{d_c}$, d_c 为字符嵌入表示的维度。

2.4 混合神经网络模型

为了更好的提取反讽文本的特征, 结合上文所述的基于语境、基于词性、基于风格的语义信

息, 本文提出了一种基于 Bi-LSTM 和 CNN 的混合神经网络架构。该架构主要由三部分组成: 对于基于语境的反讽语义表示和融合词性特征的反讽语义表示, 二者均具有明显的时间序列特征, 句子的语义与上下文信息紧密相关, 因此本文分别使用 Bi-LSTM 提取二者的潜在语义信息用于反讽识别。对于基于风格信息的反讽语义表示, 使用 Bi-LSTM 会造成长期遗忘问题, 且结构化的风格信息更有助于反讽的识别, 因此本文采用了 CNN 模型。

双向长短期记忆网络 (Bi-LSTM)

本文利用 Bi-LSTM 对基于语境和基于词性信息进行建模。LSTM 由 Hochreiter 等^[24]提出, 它被广泛应用于自然语言处理的任务当中。LSTM 采用门机制和细胞的状态更新提取文本的有效信息。LSTM 包括输入门 i_t 、遗忘门 f_t 、输出门 o_t 和细胞 c_t 。可以用如下公式形式化的描述 LSTM:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + V_i c_{t-1}) \quad (5)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + V_f c_{t-1}) \quad (6)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_{t-1}) \quad (7)$$

$$c_t = f_t \square c_{t-1} + i_t \tanh(W_c x_t + U_c h_{t-1}) \quad (8)$$

$$h_t = o_t \square \tanh(c_t) \quad (9)$$

其中 x_t 为当前输入向量, σ 为 sigmoid 函数, \square 表示按元素乘, $W_{\{i,f,o,c\}}$, $U_{\{i,f,o,c\}}$, $V_{\{i,f,o,c\}}$ 为参数, h_t 为隐层状态向量。Bi-LSTM 分别从句子的前后两个方向使用 LSTM, 然后合并二者的输出作为句子最终的表示。

利用 Bi-LSTM 对基于语境的语义信息进行特征提取, 得到反讽语句的语境的高维语义表征 r_e , 同样, 使用 Bi-LSTM 对基于词性的信息进行特征提取, 得到词性的高维语义特征 r_p 。

卷积神经网络 (CNN)

本文利用卷积神经网络提取反讽语句的风格特征。对于反讽语句的字符嵌入表示, 采用不同大小的卷积核进行卷积操作。卷积核为 $\omega \in \square^{ld}$, l 为卷积窗口大小, 最终得到输出特征 v_i , 其形式化表示如下:

$$v_i = f(\omega h_{i+l-1} + b) \quad (10)$$

其中 f 为激活函数, h_{i+l-1} 表示字符序列中第 i 列到 $i+l-1$ 列。

利用 CNN 对基于风格信息的语义特征进行特征提取, 得到反讽语句风格信息的高维语义特征 r_s 。

2.5 损失函数

综上所述, 本文融合基于语境信息、基于词性信息、基于风格信息的反讽语义特征, 采用 Softmax 函数进行反讽的识别, 其形式化描述如下:

$$z_{out} = \text{Soft max}(W[r_e; r_p; r_s] + b) \quad (11)$$

其中, $z_{out} \in \square^C$ 为预测概率, C 为类别, W 和 b 为参数。

模型基于反向传播算法采用端到端的方式进行训练, 本文采用了期望交叉熵作为损失函数。

$$\text{loss}(\theta) = -\sum_i \sum_j y_i^j \log y'_j + \lambda \|\theta\|^2 \quad (12)$$

其中, y 为真实标签, y' 为预测标签, i, j 分别为句子的编号和类别编号, λ 为正则化参数, θ 为超参数。

3 实验与实验结果分析

本小节首先对实验数据和实验设置进行介绍, 然后详细对比分析了本文提出模型和基线模型的性能, 最后通过消融实验证明所提出模型各个部分的有效性。

表 1 IAC_v2 数据集统计信息

数据集	样本数量	平均	平均
		句子长度	句子数量
GEN	3260	52	2.9
RQ	850	70	4.5
HYP	582	63	4.1

3.1 数据集及评价标准

本文使用了在线论辩数据集的 v2 版本 (IAC_v2)^[26]。IAC_v2 包含三种类型的反讽文本, 包括: 普通类型 (GEN), 反问类型 (RQ)

表 2 本文提出方法 EPSN 和其他基线方法的性能对比

方法	GEN			HYP			RQ		
	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)
Oraby	74.0	71.0	77.0	70.0	71.0	68.0	65.0	68.0	63.0
Poira	72.6	77.5	68.2	-	-	-	-	-	-
S-SVM gauss	74.9	64.7	88.9	68.2	66.1	71.2	69.2	67.0	71.8
T-Log.Reg.L1:	72.9	73.7	71.9	72.4	69.5	69.4	72.4	72.4	72.7
CNN-Adv	76.4	76.7	-	65.4	65.4	-	71.1	71.1	-
CNN-AT	76.5	76.5	-	70.4	70.4	-	72.3	73.2	-
EPSN	79.4	75.9	83.2	72.7	72.3	73.1	73.7	74.1	73.2

和夸张类型 (HYP)。三个数据集均为正负例 1:1 的均衡语料。数据集详细信息见表 1。

本文采用被广泛应用与自然语言处理任务的准确率 (Acc)、精确率 (P)、召回率 (R) 和 F1-score (F1) 来衡量实验结果。

3.2 实验设置

ELMo 训练采用了默认参数。ELMo 输出的词嵌入表示维度为 1024。针对基于 ELMo 的语义信息和基于词性信息, 本文采用了 Bi-LSTM 自动的学习文本特征。传统的词嵌入表示采用了 GloVe 训练得到的 300 维向量。词性标签的嵌入表示采用了随机初始化方法, 维度为 50。Bi-LSTM 隐层节点个数为 150, dropout 为 0.35。针对基于风格信息的反讽语义表示, 本文采用了 CNN 模型。字符的嵌入表示采用随机初始化方法, 维度为 300。CNN 模型的节点个数为 100, 窗口大小为 [3,4,5], dropout 为 0.35。在训练过程中本文采用了学习率递减策略和早停机制。本实验采用了 5 折交叉验证。

3.3 实验结果分析

不同模型的对比

为了对比本文提出的基于多语义融合的反讽识别方法的有效性, 本文采用如下基线方法:

Oraby 等方法^[26]: 采用 N-gram 和 word2Vec 作为特征, 使用 SVM 作为分类器。

S-SVM gauss^[27]: 采用潜在语义分析 (LSA) 提取反讽文本的高维语义特征, 使用支持向量机模型 (SVM) 进行反讽识别。

T-Log.Reg.L1^[27]: 采用潜在语义分析提取反讽文本特征的高维语义特征, 使用逻辑回归模型 (Log.Reg) 进行反讽识别。

Poira 等方法^[28]: 分别对情感、情绪和个人特征进行预训练, 并采用了多个 CNN 的深度神经网络模型。

CNN-Adv^[20]: 采用对抗生成的思想, 生成对抗样本参与模型训练。

CNN-AT^[20]: 基于领域迁移的对抗学习得到的反讽识别模型。

EPSN: 本文提出的基于多语义融合的神经网络模型。

表 2 中列出了本文提出的基于多语义融合的神经网络模型和前人工作的详细对比。表 2 的结果表明:

(1) 基于词袋模型的传统机器学习方法性能较低, 只考虑词频和简单的语义信息显然无法有效的提取反讽的特征。

(2) 采用 LSA 可以将词和反讽文本映射到潜在语义空间, 可以从一定程度上提高传统机器学习方法的性能。Poira 提出的深度神经网络模型在 GEN 上取得了最高的精确率, 在应用了对抗生成和迁移的方法之后, 反讽识别的性能在三个数据集上均有了提升, 这表明基于词嵌入表示的深度学习方法能够有效的提取反讽的潜在语义特征, 提升反讽识别的性能。

(3) 本文提出的方法 EPSN 在 GEN 和 HYP 两个数据集上的 F1 值提升较明显, 相对于 CNN-AT, 在 GEN 上提升 2.9%, 在 HYP 上提升 2.3%, 而对于 RQ 数据提升较小, 约为 1.4%, 这说明模型在大数据集上能够更好的学习的反讽语义特征, 而对于 RQ 数据集, 其数据规模较小, 采用传统的机器学习方法就能得到比较有竞争力的性能。此外, EPSN 模型在三类反讽数据集上

都取得最高的 F1 值, 说明引入语境信息、词性信息和风格信息能够从不同侧面反映了反讽的语义特征, 提高模型识别反讽语句的性能, 同时也显示出 EPSN 模型具有较好的泛化能力。

不同词嵌入表示对比

为了验证基于语境的反讽语义表示的有效性, 本文对比了不同维度和不同训练方法得到的词嵌入表示的性能。对比过程采用 Bi-LSTM 提取高维语义特征, 参数均采用相同设置。实验结果如下表:

表 3 不同词向量对模型性能的影响

词向量	GEN (F1 %)	HYP (F1 %)	RQ (F1 %)
W2V-100	72.4	70.1	70.3
W2V-300	73.7	70.2	70.2
GloVe-100	72.5	70.3	70.2
GLoVe-300	72.7	70.2	70.4
ELMo-1024	75.6	71.3	72.1

表 3 中列出了使用 Word2Vec 词嵌入表示^[29]和 GloVe 词嵌入表示^[30]的实验结果, 数字表示维度。实验结果表明, Word2Vec 和 GloVe 词嵌入表示在三个数据上的性能区别不大, 而 ELMo 在三个数据集上性能提升明显, 均有 1%-3%的提升。说明采用 ELMo 在大规模反讽数据上训练得到的词嵌入表示能够更好的表达反讽数据中单词的分布式语义表示, 且 ELMo 能够根据不同的上下文赋予单词不同语义特征, 这也使得基于 ELMo 的反讽语义表示能够包含语境信息, 从而提升了反讽识别性能。

不同语义表示的对比

为了验证基于词性信息和风格信息的对模型的影响, 本文对比了引入以上两类语义特征对神经网络模型的影响。EN 表示仅使用 ELMo 信息, EPN 表示使用了 ELMo 和词性信息, ESN 表示使用了 ELMo 和风格信息。

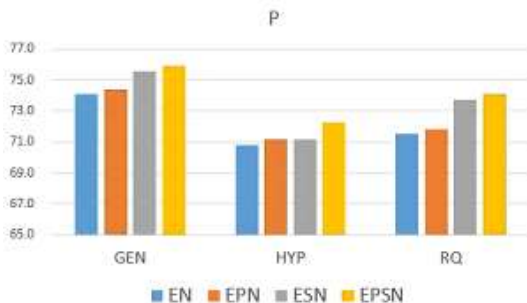


图 2 多语义融合对模型精确率的影响

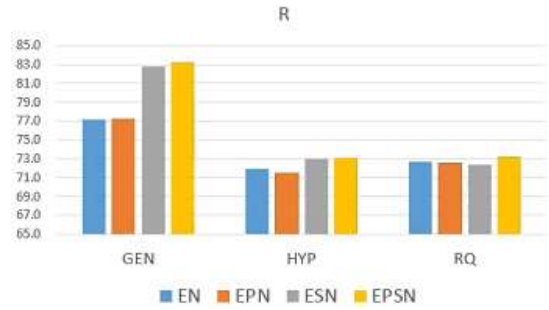


图 3 多语义融合对模型召回率的影响

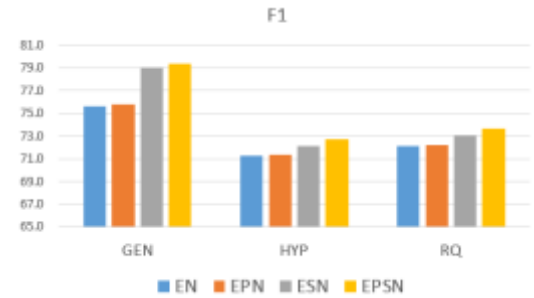


图 4 多语义融合对模型 F 值的影响

多语义融合模型对不同评价标准的影响如图 2-4 所示。首先, 从图中我们发现, 词性信息对三种类型的反讽数据的评价标准影响较小, 这主要是由于 ELMo 得到的语义信息已经包含了部分词性信息, 因此引入词性信息后, 模型的性能只有很小幅度的提升。其次, 如图 3 所示, 引入风格信息后, GEN 数据集的召回率有了极大提升, 提升幅度接近 6%, 精确率也有小幅度提高, 这表明普通类型的反讽语句包含更多的诸如押韵、重复等修辞手法。此外, 基于风格信息的反讽语义特征在 HYP 和 RQ 数据集上也能够促使性能提升。风格信息的提取对模型性能的提升起到重要的作用。最后, 基于多语义融合的模型在三种类型的反讽语料上都取得了最佳的性能, 说明本文提出的模型能够从语境信息、词性信息、风格等多个维度学习反讽语句的潜在语义特征。

4 总结与展望

本文针对反讽语句缺乏语境信息的问题, 采用 ELMo 从大规模反讽语料训练得到反讽领域相关的词嵌入表示, 该表示能够根据反讽语句不同的上下文赋予单词不同的语义信息。此外, 根据反讽语句的词性信息和风格特点对反讽语句进行信息嵌入式表示, 分别使用 Bi-LSTM 和 CNN 提

取词性特征和风格特征用于反讽识别。在公开反讽数据集 IAC_v2 的三个子集上的实验结果表明, 本文提出的基于多语义融合的神经网络模型能够明显地提高反讽识别性能, 取得了目前已知的最佳性能。

反讽存在多种类型, 不同类型的反讽其语义特点相差较大, 很难用统一的模型解决所有类型的反讽识别问题。在接下来的工作中, 将进一步探索更多反讽相关的特征并将它们融入到神经网络模型当中。此外, 反讽和幽默、双关等具有许多相似之处, 如何进行领域迁移和联合学习也是值得关注的方向。

参考文献

- [1] Far á s D I H, Patti V, Rosso P. Irony detection in Twitter: The role of affective content[J]. ACM Transactions on Internet Technology (TOIT), 2016, 16(3): 19.
- [2] Joshi A, Bhattacharyya P, Carman M J. Automatic sarcasm detection: A survey[J]. ACM Computing Surveys (CSUR), 2017, 50(5): 73.
- [3] Carvalho P, Sarmiento L, Silva M J, et al. Clues for detecting irony in user-generated contents: oh...!! it's so easy;-[C]//Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion. ACM, 2009: 53-56.
- [4] Kreuz R J, Caucci G M. Lexical influences on the perception of sarcasm[C]//Proceedings of the Workshop on computational approaches to Figurative Language. Association for Computational Linguistics, 2007: 1-4.
- [5] Bamman D, Smith N A. Contextualized sarcasm detection on twitter[C]//Ninth International AAAI Conference on Web and Social Media. 2015.
- [6] 刘龙飞, 杨亮, 张绍武, 等. 基于卷积神经网络的微博情感倾向性分析[J]. 中文信息学报, 2015, 29(6): 159-165.
- [7] Peters M E, Neumann M, Iyyer M, et al. Deep contextualized word representations[J]. arXiv preprint arXiv:1802.05365, 2018.
- [8] Veale T, Hao Y. Detecting ironic intent in creative comparisons[C]//ECAI. 2010, 215: 765-770.
- [9] Maynard D G, Greenwood M A. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis[C]//LREC 2014 Proceedings. ELRA, 2014.
- [10] Bharti S K, Babu K S, Jena S K. Parsing-based sarcasm sentiment recognition in twitter data[C]//Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015. ACM, 2015: 1373-1380.
- [11] Riloff E, Qadir A, Surve P, et al. Sarcasm as contrast between a positive sentiment and negative situation[C]//Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013: 704-714.
- [12] Far á s D I H, Patti V, Rosso P. Irony detection in Twitter: The role of affective content[J]. ACM Transactions on Internet Technology (TOIT), 2016, 16(3): 19.
- [13] Bamman D, Smith N A. Contextualized sarcasm detection on twitter[C]//Ninth International AAAI Conference on Web and Social Media. 2015.
- [14] Reyes A, Rosso P, Veale T. A multidimensional approach for detecting irony in twitter[J]. Language resources and evaluation, 2013, 47(1): 239-268.
- [15] Joshi A, Tripathi V, Patel K, et al. Are Word Embedding-based Features Useful for Sarcasm Detection?[J]. EMNLP, 2016.
- [16] Amir S, Wallace B C, Lyu H, et al. Modelling context with user embeddings for sarcasm detection in social media[J]. CONLL, 2016.
- [17] Zhang M, Zhang Y, Fu G. Tweet sarcasm detection using deep neural network[C]//Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers. 2016: 2449-2460.
- [18] Ghosh A, Veale T. Magnets for Sarcasm: Making sarcasm detection timely, contextual and very personal[C]//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017: 482-491.
- [19] Tay Y, Tuan L A, Hui S C, et al. Reasoning with Sarcasm by Reading In-between[J]. ACL, 2018.
- [20] 张庆林, 杜嘉晨, 徐睿峰. 基于对抗学习的讽刺识别研究[J]. 北京大学学报 (自然科学版), 2019, 55(1): 29-36.
- [21] Khodak M, Saunshi N, Vodrahalli K. A large self-annotated corpus for sarcasm[J]. arXiv preprint arXiv:1704.05579, 2017.
- [22] Liu L, Zhang D, Song W. Exploiting Syntactic Struc-

-
- tures for Humor Recognition[C]//Proceedings of the 27th International Conference on Computational Linguistics. 2018: 1875-1883.
- [23] Reyes A, Rosso P, Veale T. A multidimensional approach for detecting irony in twitter[J]. *Language resources and evaluation*, 2013, 47(1): 239-268.
- [24] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.2
- [25] Graves A, Jaitly N, Mohamed A. Hybrid speech recognition with deep bidirectional LSTM[C]//2013 IEEE workshop on automatic speech recognition and understanding. IEEE, 2013: 273-278.
- [26] Oraby S , Harrison V , Reed L , et al. Creating and Characterizing a Diverse Corpus of Sarcasm in Dialogue[J]. 2017.
- [27] Poria S , Cambria E , Hazarika D , et al. A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks[J]. 2016.
- [28] Di Gangi M A , Lo Bosco, Giosu é Pilato G . Effectiveness of data-driven induction of semantic spaces and traditional classifiers for sarcasm detection[J]. *Natural Language Engineering*, 2019, 25(2):257-285.
- [29] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space. In *ICLR*, pages 1-12, 2013.
- [30] 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 (EMNLP). 2014: 1532-154

