高技术通讯 2024 年 第 34 卷 第 5 期:453-462

doi:10.3772/j.issn.1002-0470.2024.05.002

基于多维语义特征与层次注意力机制的讽刺识别①

宋留静②*** 赵泽方*** 马宇翔*** 申罕骥* 李 俊③***

(*中国科学院计算机网络信息中心 北京 100190)

(**中国科学院大学 北京 100049)

(***河南大学计算机与信息工程学院 开封 475004)

摘 要 讽刺是一种复杂的语言表达方式,在日常交流中发挥着重要作用。随着人工智能和社交网络的快速发展,讽刺识别已成为自然语言处理领域的热点研究课题之一。现有的讽刺识别研究往往从单一维度对讽刺文本特征进行表示,忽视了讽刺文本特征的细微差异及其重要程度。本文将讽刺识别视为文本分类任务,在特征提取阶段,将讽刺文本根据其不一致性特征、情感特征、句法结构特征和风格特征进行多维语义特征表示。在特征融合阶段,针对不同维度特征对整体特征贡献和关联程度不同,采用层次注意力机制调整不同讽刺语言学特征对模型整体性能的影响。实验结果表明,所提出的模型能够从多个维度提取讽刺文本的潜在语义特征,其在公开数据集 IAC、Tweets 和 Reddit 上的实验性能均有明显提升。

关键词 讽刺识别;自然语言处理;多维语义表示;层次注意力机制

讽刺[1]是一种微妙的语言表达方式,其所要表达的情感倾向和字面含义往往相反,在日常生活中被广泛应用。近年来,随着微博、推特等社交媒体的快速发展,越来越多的用户倾向于通过讽刺的话语分享他们的个人观点,这种隐式情感表达给文本情感分析带来巨大挑战。如何使用计算机自动识别讽刺文本,成为自然语言处理领域备受关注的研究热点之一。并且,讽刺识别课题的深入研究对提高情感分析性能、问答系统和机器翻译等领域都具有重要意义。

讽刺识别任务被看做是一种文本分类问题,目标是在社交网络中,判断一条言论是否包含讽刺的含义[2]。传统的讽刺识别方法主要基于语言学特点,以讽刺理论为依据人工构建特征和规则[3],需要依赖领域专家耗费大量的时间和精力,并且规则模型的迁移性能较差。随着人工智能快速发展,基于特征的机器学习方法[4-6]在讽刺识别任务中得到应用,该方法以当前的数据集为基础,能够学习预测讽刺文本,但是其泛化性能较弱。目前,深度学习在自然语言处理领域取得重大突破,有研究者将其应用于讽刺识别中[7-9],该方法首先采用词嵌入表示单词的语义信息,然后通过神经网络学习深层次特征。但是现有工作往往从单一维度对讽刺文本进行表示,难以表达讽刺文本细微的特征差异,同时忽略了不同维度的特征对文本分类的影响程度不同,这可能导致目前的深度学习模型性能受到限制。

为了解决上述问题,本文提出了一种基于多维语义特征和层次注意力机制的讽刺识别模型。首先考虑到讽刺文本的语言特点,采用词嵌入语言模型表示方法 GloVe(global vectors for word representa-

① 国家重点研发计划(2019YFB1405801),中国科学院对外合作重点项目(241711KYSB20180002)和河南省重点研发与推广专项(222102210040)资助项目。

② 女,1993 年生,博士生;研究方向:自然语言处理,情感分析;E-mail:songliujing@cnic.cn。

③ 通信作者,E-mail:lijun@cnic.cn。

(收稿日期:2023-04-05)

tion)[10],在大规模的讽刺语料库里进行训练,得到讽刺相关的词嵌入语义表示。其次针对讽刺文本不一致性的特点,采用双向长短时记忆神经网络(bi-directional long short-term memory,Bi-LSTM)和自注意力机制提取讽刺文本的语义特征。再次针对讽刺文本的情感特征和句法结构特征的特点,在语义表示的基础上,均采用图卷积神经网络(graph convolutional network,GCN)提取其语义特征。然后针对讽刺文本风格特点,采用字符级嵌入表示和卷积神经网络(convolutional neural networks,CNN)提取其高维潜在语义特征。最后采用层次注意力机制,将不同维度的语义特征进行融合,从而进行讽刺识别。综上所述,本文的贡献有以下 3 点。

(1)针对讽刺文本的多种语言学特点,采用词嵌入表示和深度学习算法,分别从讽刺文本的不一致特征、情感特征、上下文特征和风格特征 4 个维度对讽刺文本进行特征表示。

(2)针对不同维度特征在讽刺识别模型中的重要程度不同,采用层次注意力机制,为不同的维度分配合理的权重,从而进一步提高模型识别讽刺的性能。

(3)提出的基于多维语义特征和层次注意力机制的讽刺识别模型,在公开数据集 IAC、Tweets 和 Reddit 上分别进行了实验,结果表明,本文方法能够有效提升讽刺识别的性能。

1 相关工作

随着讽刺在社交网络中的广泛应用以及文本情感分析问题的深入研究,越来越多的学者对讽刺识别产生了浓厚的兴趣,讽刺识别成为了自然语言处理领域的热点研究问题之一。根据讽刺识别所使用方法的不同,将前人的研究大致分为 4 类:基于规则的方法、基于特征的机器学习方法、基于神经网络的深度学习方法和基于注意力机制的讽刺识别方法。

早期的讽刺识别研究大多采用基于规则的方法,通过制定一系列固定模式的规则来获取讽刺文本的语义特征。Carvalho 等人[3]使用用户评论中的口头或手势线索,如表情符号、拟声表达、特殊标点符号等进行讽刺识别。Riloff 等人[2]提出了一种自举算法,通过自动学习文本中的正负情感词进行讽刺识别。Maynard 和 Greenwood[11]将推特上的 hashtag 视为讽刺的主要特征,从而同时对情感和讽刺文本进行分类识别。虽然基于规则的方法可以在特定的文本或场景中取得一定的效果,但很难进一步扩展和推广。

基于特征的机器学习方法通常使用词袋模型,构建多种语义特征作为讽刺文本特征,并且使用传统的机器学习方法进行讽刺识别。GonzálezIbánez 等人[12]将单词和语用因素作为特征,通过支持向量机来识别讽刺。Reyes 等人[6]通过 n-gram 语言模型找到具有讽刺信息的单词,并分别使用朴素贝叶斯分类器和决策树算法在平衡和非平衡讽刺数据集上进行对比实验。Bamman 等人[13]和 Joshi 等人[5]试图通过收集语言外信息来捕捉上下文的不一致性。Farías 等人[4]利用标点符号频率、推文长度、大写字符数量和情感特征等结构特征进行讽刺识别。传统的机器学习方法需要复杂的特征设计,耗时长且需要专业的知识背景,因此模型泛化性能较弱。

基于神经网络的深度学习方法主要采用词嵌入表示,如 Word2Vec、GloVe 等方法,并使用 CNN、循环神经网络(recurrent neural network,RNN)、图神经网络(graph neural network,GNN)等神经网络方法提取特征,从而对讽刺文本进行识别。Amir 等人[7]、Kim 等人[14]和 Das 等人[15]对用户信息进行嵌入式表示,采用 CNN 模型学习文本内容进行讽刺识别。Ghosh 等人[16]使用 CNN 和 Bi-LSTM 对推文的用户信息、时间信息和上下文信息进行建模。Liang 等人[17]和 He 等人[18]使用 GCN 获取讽刺文本的全局特征。Huang 和 Carley[19]使用图注意力网络来构建讽刺文本的依存树。Lou 等人[20]进一步使用 GCN 对讽刺文本的情感特征和上下文特征进行表示,从而进行讽刺识别。实验结果表明,基于神经网络的深度学习方法使讽刺识别的性能得到明显提升。

近年来,基于注意力机制的讽刺识别方法得到广泛的应用,其关键思想是根据特征的重要程度不同,能够全局捕捉联系,从而获取长期依赖关系。Tay 等人[9]将注意力模型用于讽刺识别,强调讽刺

是由积极和消极情感或字面意和隐藏意的不一致性产生的,通过构建多维内部注意力循环网络,对文本中每个词对的相似度进行建模,从而获取不一致性信息。Zhang 等人[21]使用迁移学习和基于注意力的 Bi-LSTM 自动获取上下文不一致性信息进行讽刺识别。Kumar 等人[8]和 Duan 等人[22]将注意力机制引入神经网络来识别讽刺评论,实验结果显示,注意力机制明显提高了识别性能。

2 基于多维语义的讽刺识别方法

如图 1 所示,本文提出基于多维语义特征与层次注意力机制的神经网络模型,其进行讽刺识别的主要步骤为:(1)将文本内容进行词嵌入表示,采用 Bi-LSTM 和自注意力机制提取文本的不一致性特征;(2)通过情感词典获取情感图和词嵌入表示,通过 GCN 提取文本的情感特征;(3)通过依赖树获取句法结构图和词嵌入表示,通过 GCN 提取文本的句法结构特征;(4)将文本进行字符级的词嵌入表示,通过 CNN 和注意力机制提取风格特征;(5)为了更好区分不同讽刺特征在讽刺识别任务中的贡献程度,本文采用层次注意力机制对不同维度的特征分配权重,从而进行讽刺识别。

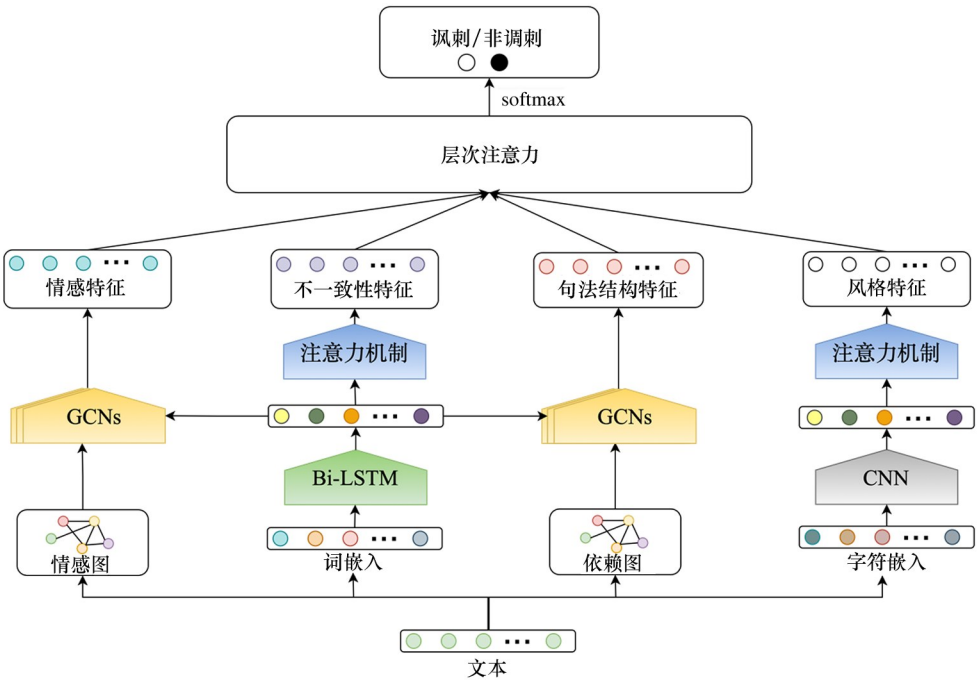

图 1 基于多维语义特征的层次注意力神经网络模型

2.1 多维语义特征表示

现有的深度学习算法试图挖掘数据集更深层的语义特征,多维语义特征表示有利于提高神经网络的学习性能。因此,本文模型构造了多维语义特征表示,使得神经网络模型能够更充分地学习讽刺文本潜在的语义特征。本节将介绍讽刺文本的不一致性特征、情感特征、句法结构特征和风格特征的表示及特征提取方法。

2.1.1 不一致性特征表示

语境和语义的不一致性是讽刺文本表达的重要因素。Riloff 等人[2]将讽刺文本的不一致性分为 4 类:情感语境不一致性、语义语境不一致性、历史语境不一致性和会话语境不一致性。大量研究表明,不一致性特征在讽刺识别中起重要作用。

本文基于分布式假设将单词映射为低维稠密向量,并同时保持了单词的语义信息。设文本的单词序列为 $\boldsymbol{S}=\{w_i\}_{i=1}^{n}$,$n$ 为句子的长度,将句子中的每个单词 w_i 嵌入到 m 维向量表示中,则句子的向量表示为 $\boldsymbol{X}=\{\boldsymbol{x}_i\}_{i=1}^{n}$,其中 $\boldsymbol{x}_i\in R^m$,$\boldsymbol{X}\in R^{m\times n}$。

模型利用 Bi-LSTM 提取语句的潜在语义特征。LSTM 能够对文本语义上的长距离依赖关系进行建模,而 Bi-LSTM 能够从正反 2 个方向分别提取文本

的潜在语义特征,并且融合两部分的语义信息。在每个时间戳 t, 正向和反向的 LSTM 分别对输入词向量 $\boldsymbol{x}_t$ 的处理过程为

$$\overrightarrow{\boldsymbol{h}_t} = \text{LSTM}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t) \tag{1}$$

$$\overleftarrow{\boldsymbol{h}_t} = \text{LSTM}(\boldsymbol{h}_{t+1}, \boldsymbol{x}_t) \tag{2}$$

其中, $\boldsymbol{h}_t$ 表示 t 时刻的隐藏向量, $\boldsymbol{x}_t$ 为 t 时刻输入的词向量。将每个时间戳正反 2 个方向的隐藏向量拼接起来就得到 Bi-LSTM 单个时间戳的输出,记作 $\boldsymbol{h}_t = [\overrightarrow{\boldsymbol{h}_t}, \overleftarrow{\boldsymbol{h}_t}] R^{2d_h}$, d_h 表示隐藏向量的维度。因此,特征表示层能够得到讽刺文本的潜在语义特征,记为

$$\boldsymbol{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_n\} = \text{Bi-LSTM}(\boldsymbol{x}) \tag{3}$$

模型采用注意力机制提取每 2 个词对之间的不一致性语义关系。计算公式如下所示。

$$\boldsymbol{w}_{ij} = \tanh(\boldsymbol{W}^{\mathrm{T}} \boldsymbol{h}_i + \boldsymbol{b}) \tag{4}$$

$$\boldsymbol{\alpha}_{ij} = \frac{\exp(\boldsymbol{w}_{ij})}{\sum_{j=1}^{n} \exp(\boldsymbol{w}_{ij})} \tag{5}$$

$$\boldsymbol{q}^i = \sum_{j=1}^{n} \boldsymbol{\alpha}_{ij} \boldsymbol{h}_i \tag{6}$$

其中, $\boldsymbol{W}$ 为权重参数, $\boldsymbol{b}$ 为偏置项, tanh 为非线性激活函数, $\boldsymbol{\alpha}_{ij}$ 为注意力权重。所有参数均采用随机初始化并且在训练过程中动态更新, $\boldsymbol{q}^i$ 表示不一致性特征注意力层输出向量。

2.1.2 情感特征表示

讽刺文本中的情感词对讽刺识别存在显著的影响。在通常情况下,讽刺文本带有强烈的主观色彩,语句中大多包含具有强烈情感倾向性的单词。因此,在许多讽刺识别的研究中常采用情感词典的方式提取情感词作为讽刺文本的重要特征。

为了更好地使得模型学习到文本中的情感词语和文本语义之间关系特征,本文采用情感词典 SenticNet 构建情感图并获得邻接矩阵 $\boldsymbol{A}_{ij}^a \in R^{n \times n}$, 单词的情感得分为

$$\boldsymbol{A}_{ij}^a = \text{abs}(S(w_i) - S(w_j)) \tag{7}$$

其中 $S(w_i) \in [-1, 1]$ 表示从 SenticNet 检索到的单词 w_i 的情感得分,如果情感词典中不包含 w_i, 则 $S(w_i) = 0$。

在得到情感邻接矩阵后,本文将其和上节中文本特征一同输入到 GCN 架构中,可获得情感特征表示:

$$\boldsymbol{g}_a^l = f(\tilde{\boldsymbol{A}}^a \boldsymbol{g}^{l-1} \boldsymbol{W}_a^l + \boldsymbol{b}_a^l) \tag{8}$$

其中, $\boldsymbol{g}^{l-1} \in R^{n \times 2d_n}$ 是由通过前面的 GCN 层所得到的隐藏层表示,然而第 1 层 GCN 的初始特征为通过 Bi-LSTM 学习到的上下文特征 $\boldsymbol{g}^0 = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_n\}$; f 为非线性激活函数 ReLU, $\tilde{\boldsymbol{A}}_i^a = \boldsymbol{A}_i^a / (\boldsymbol{E}_i - 1)$ 是规范化邻接矩阵,其中 $\boldsymbol{E}_i = \sum_{j=1}^{n} \boldsymbol{A}_{i,j}^a$ 是情感邻接矩阵 $\boldsymbol{A}_i^a$ 的度; $\boldsymbol{W}_a^l \in R^{2d_n \times 2d_n}$ 为训练参数; $\boldsymbol{b}_a^l \in R^{2d_n}$ 为偏置项。

2.1.3 句法结构特征表示

讽刺是一种复杂的语言学现象,句子成分较为复杂,如果缺乏对文本句法结构的利用,将难以对目标文本是否存在讽刺作出正确判断。因此,句法结构特征是讽刺识别的关键特征。

本文使用基于依存关系树来构建句法结构图,并得到句法结构邻接矩阵为

$$\boldsymbol{A}_{ij}^d = \begin{cases} 1 & \text{如果 } \varphi(w_i, w_j) \\ 0 & \text{其他} \end{cases} \tag{9}$$

其中 $\boldsymbol{A}^d \in R^{n \times n}$, $\varphi(w_i, w_j)$ 表示句子的依存树中单词 w_i 和单词 w_j 之间的关系。受文献[20]启发,本文通过构建无向图来构造句法结构图,因此 $\boldsymbol{A}_{i,j} = \boldsymbol{A}_{j,i}$, 且当 $i = j$ 时 $\boldsymbol{A}^d = 1$。

同情感特征表示类似,可得到句法结构特征表示为

$$\boldsymbol{g}_d^l = f(\tilde{\boldsymbol{A}}^d \boldsymbol{g}^{l-1} \boldsymbol{W}_d^l + \boldsymbol{b}_d^l) \tag{10}$$

2.1.4 风格特征表示

讽刺是一种文学体裁,通常具有独特的风格特征。文献[23]指出文本中某些重复的字符或者重复的标点符号使得文本表现出相对稳定的风格特征。在很多情况下,正是由于文本的风格特征从而产生了讽刺的效果。例如:

You have a looooooooot of never !!!!!

上述示例是一个讽刺文本,该文本采用字符重复的方式表现讽刺效果。文本中的单词"looooooooot"是一个不规范的拼写形式,字符"!"也被重复了多次,这种刻意的字符重复是讽刺文本的重要特征。为了更好地获取风格特征,本文将文本表示成字符序列的形式,对于每个字符,采用唯一的编码将其相

互对应,字符表包括 26 个英文字母,0 ~ 9 共 10 个数字以及 34 个常用符号,共计 70 个字符。

在风格特征的向量表示层中,对于每个目标文本,本文首先将其表示成字符的序列,然后使用字符的嵌入(character embedding)表示将文本中的每个字符都映射到高维语义空间。与词嵌入表示相同,设文本语句的单词序列为 $S = \{w_1, w_2, \cdots, w_n\}$,$n$ 为句子的长度。此时,对于每个单词 w_i,其字符向量表示为 $\boldsymbol{w}_i = \{l_1, l_2, \cdots, l_p\}$,$l_i \in R^{d_s}$,$p$ 为单词的长度,d_s 为字符向量的维度。本文将采用随机初始化的方法对字符向量进行初始化。

在风格的特征提取层中,由于风格特征并没有明显的时间序列特征,文本的语义与上下文信息也没有紧密相关。若本文使用 Bi-LSTM 会造成长期遗忘的问题,并且结构化的风格信息更加有助于讽刺文本的表示,因此,本文采用 CNN 的神经网络模型来提取文本的风格特征。卷积层利用窗口大小为 h 的卷积核来提取文本局部的风格特征,计算公式如式(11)所示。

$$\boldsymbol{c}_t = f(\boldsymbol{W} l_{i:i+l-1} + \boldsymbol{b}) \tag{11}$$

其中,$\boldsymbol{c}_t$ 为输出的风格特征向量,f 为非线性激活函数 ReLU,$\boldsymbol{W}$ 为参数,$\boldsymbol{b}$ 为偏置项,$l_{i:i+l-1}$ 表示第 i 个单词到第 $i + L - 1$ 列。在实验中,本文使用多个卷积核,拼接得到风格特征表示为 $\boldsymbol{h}^s$。

在风格特征注意力层中,为能够对携带明显语义信息的字符给予更多的关注,本文在提取风格特征时引用注意力机制,计算公式如下:

$$\boldsymbol{w}_{ij} = \tanh(\boldsymbol{W}^{\mathrm{T}} \boldsymbol{h}_i^s + \boldsymbol{b}) \tag{12}$$

$$\boldsymbol{\alpha}_{ij} = \frac{\exp(\boldsymbol{w}_{ij})}{\sum_{j=1}^{n} \exp(\boldsymbol{w}_{ij})} \tag{13}$$

$$\boldsymbol{q}^s = \sum_{j=1}^{n} \boldsymbol{\alpha}_{ij} \boldsymbol{h}_i^s \tag{14}$$

其中,$\boldsymbol{W}$ 为权重参数,$\boldsymbol{b}$ 为偏置项,tanh 为非线性激活函数,$\boldsymbol{\alpha}_{ij}$ 为注意力权重。所有参数均采用随机初始化并且在训练过程中动态更新,$\boldsymbol{q}^s$ 表示风格特征注意力层输出向量。

2.2 层次注意力机制

由于不同的讽刺语言学特征对讽刺文本的关联程度不同,为了更好地为不同维度的特征分配权重,本文采用层次注意力机制来调整不同语言学特征对于讽刺识别性能的影响,公式为

$$\boldsymbol{w}_j = \tanh(\boldsymbol{W}^{\mathrm{T}} V_j + \boldsymbol{b}) \tag{15}$$

$$\beta_j = \frac{\exp(\boldsymbol{w}_j)}{\sum_{j=1}^{4} \exp(\boldsymbol{w}_j)} \tag{16}$$

$$\boldsymbol{q} = \sum_{j=1}^{4} \beta_j \boldsymbol{k}_j,\ k_j \in \{\boldsymbol{q}^i, \boldsymbol{g}_a^l, \boldsymbol{g}_d^l, \boldsymbol{q}^s\} \tag{17}$$

其中,$\boldsymbol{W}$ 为参数矩阵,$\boldsymbol{b}$ 为偏置项,tanh 为非线性激活函数,V_j 为不同的句子特征表示,β_j 为不同特征的注意力权重。所有参数采用随机初始化并且在训练中动态更新。$\boldsymbol{q}^i$ 为不一致性特征表示,$\boldsymbol{g}_a^l$ 为情感特征表示,$\boldsymbol{g}_d^l$ 为句法结构特征表示,$\boldsymbol{q}^s$ 为风格特征表示。$\boldsymbol{q}$ 为文本的最终特征表示。

2.3 讽刺分类

文本采用了层次注意力机制方法融合了文本不一致特征、情感特征、句法结构特征和风格特征,随后采用 Softmax 函数进行讽刺识别,其形式化表示如下:

$$v = \tanh(\boldsymbol{W}' \boldsymbol{q} + \boldsymbol{b}') \tag{18}$$

$$\hat{\boldsymbol{y}} = \mathrm{softmax}(\boldsymbol{W} \boldsymbol{v} + \boldsymbol{b}'') \tag{19}$$

其中,$\boldsymbol{W}'$、$\boldsymbol{W}''$ 为权重参数,$\boldsymbol{b}'$、$\boldsymbol{b}''$ 为偏置项。最终向量 $\boldsymbol{v}$ 作为 Softmax 函数的输入,被用来预测目标文本是否为讽刺文本,$\hat{\boldsymbol{y}}$ 为预测标签。

本文模型是基于反向传播算法与端到端的方式进行训练,并且采用期望交叉熵作为损失函数,表达公式如下:

$$loss = -\sum_i \sum_j y_i^j \log \hat{\boldsymbol{y}}_i^j + \lambda \| \theta \|^2 \tag{20}$$

其中 y 为真实的标签,i、j 分别为文本的编号和类别编码,λ 为正则化参数,θ 为超参数。

3 实验结果与分析

本节首先介绍实验数据、评价指标、实验的设置和基线方法,然后详细对比分析本文提出的模型和基线模型的讽刺识别性能,最后通过实验结果分析了本文方法的有效性。

3.1 数据集评价指标

为了评估本文所提出的模型,本文在来自 3 个知名来源的 6 个基准数据集上进行了实验。为了保证公平,本文实验数据集的统计和处理与文献[9,20]

保持一致。数据集规模见表 1。

表 1 讽刺数据集统计信息

数据集	训练集		测试集	
	讽刺	非讽刺	讽刺	非讽刺
IAC-V1	862	859	97	94
IAC-V2	2 947	2 921	313	339
Tweets-1	282	1 051	35	113
Tweets-2	23 456	24 387	2 569	2 634
Reddit-1	5 521	5 607	1 389	1 393
Reddit-2	6 419	6 393	1 596	1 607

IAC(Internet argument corpus)数据集来自于社交媒体政治辩论的论坛,它包括 2 个版本,分别表示为 IAC-V1(https://nlds.soe.ucsc.edu/sarcasm1)和 IAC-V2(https://nlds.soe.ucsc.edu/sarcasm2)。

Tweets 数据集由 Riloff[2] 和 Ptáček[24] 提供,本文使用 Twitter API 以及所提供的 tweet IDs(http://api.twitter.com/)获取推文。

Reddit 数据集,本文使用文献[25]所提供的 Reddit(http://nlp.cs.princeton.edu/SARC)的 2 个子数据集(movies 和 technology)进行讽刺识别。

为了更直观地和基线方法进行比较,本文采用了被广泛接受并应用于文本分类任务中的精确率(Precision)、召回率(Recall)、F1-score(F1)精确率(Acc.)作为评价指标。

3.2 参数设置

本文所有实验均在 NVIDIA Tesla V100S GPU,PyTorch3.6[26] 上运行。在训练过程中,词向量采用 GloVe 进行初始化,维度为 300。GCN 的层数均设置为 3 层,隐藏层表示的维度为 300。L_2 正则化系数为 0.01,采用 Adam 优化器,学习率为 0.001,批大小(batch size)为 32,丢弃率(dropout)为 0.3。为了防止过度拟合,本文采用了学习率递减策略和早停机制。

3.3 基线方法

本文使用下述基线方法进行对比实验。

NBOW[9] 是一个简单的神经词袋基线,对所有的词嵌入模型进行求和,并将求和后的向量传递到一个简单的逻辑回归层。

CNN 是一个具有最大池化层的普通卷积神经网络。

GRNN[21] 模型采用双向门控循环单元(bidirectional gated recurrent unit, Bi-GRU)提取讽刺文本局部句法和语义信息。

CNN-LSTM-DNN[27] 是由 CNN、LSTM 和深度神经网络堆叠的方式进行讽刺识别。

ATT-LSTM[28] 是一个基于注意力的 LSTM 模型,它对 LSTM 编码器的所有隐藏状态均采用了注意力机制。

SIARN[9] 和 MIRAN[9] 使用内部注意力机制克服了序列神经网络的局限性,提取目标文本语义特征。SIARN 采用的一维内部注意力机制,MIRAN 采用的是多维内部注意力机制。

SAWS[29] 采用加权片段的自注意力机制模型进行讽刺识别,克服了以往模型在判断由片段不一致引起的讽刺时效率较差的问题。

ADGCN[20] 模型基于 GCN 神经网络,通过将情感信息和依赖信息进行交互建模,将 Bi-LSTM 的隐藏层输出作为 GCN 的初始化输入。

3.4 实验结果分析

3.4.1 不同模型对比

表 2~4 展示了 6 个基准数据集在本文模型上的实验结果,研究结论如下。

(1)基于神经网络的深度学习方法性能明显优于基于特征的机器学习方法。NBOW 表现性能较差,但和初期的神经网络方法相比,其性能甚至高于 CNN-LSTM-DNN 方法,并且传统的机器学习方法一般准确率高,召回率低,更好地说明了基于特征的机器学习方法更加依赖人工构造特征的质量,且泛化能力较差。

(2)基于注意力机制的模型性能表现较好。起初的神经网络模型没有使用注意力机制,仅获取语句的局部语义信息,不足以识别复杂的讽刺表达之间的长期依赖关系,因此证明了注意力机制的有效性。

表 2　在 IAC 2 个数据子集上的实验结果

模型	IAC-V1				IAC-V2			
	精确率/%	召回率/%	F1/%	准确率/%	精确率/%	召回率/%	F1/%	准确率/%
NBOW	57.17	57.03	57.00	57.51	66.01	66.03	66.02	66.09
CNN	58.21	58.00	57.95	58.55	68.45	68.18	68.21	68.56
GRNN	56.21	56.21	55.96	55.96	62.26	61.87	61.21	61.37
CNN-LSTM-DNN	55.50	54.60	53.31	55.96	64.31	64.33	64.31	64.38
ATT-LSTM	58.98	57.93	57.23	59.07	70.04	69.62	69.63	69.96
SLARN	63.94	63.45	60.52	62.69	72.17	71.81	71.85	72.10
MIARN	63.88	63.71	63.18	63.21	72.92	72.93	72.75	72.75
SAWS	66.22	65.65	65.60	66.13	73.25	73.40	73.43	73.55
ADGCN	68.08	68.08	68.06	68.06	76.96	76.98	76.97	76.99
本文	**72.61**	**72.61**	**72.59**	**72.60**	**78.41**	**78.26**	**78.32**	**78.33**

表 3　在 Tweets 2 个数据子集上的实验结果

模型	Tweets-1(Riloff)				Tweets-2(Ptáček)			
	精确率/%	召回率/%	F1/%	准确率/%	精确率/%	召回率/%	F1/%	准确率/%
NBOW	71.28	62.37	64.13	79.23	80.02	79.06	79.43	80.39
CNN	71.04	67.13	68.55	79.48	82.13	79.67	80.39	81.65
GRNN	66.32	64.74	65.40	76.41	82.06	81.02	82.43	82.20
CNN-LSTM-DNN	69.76	66.62	67.81	78.72	79.65	79.12	79.20	79.94
ATT-LSTM	69.76	66.62	67.81	78.72	81.62	81.45	81.56	81.56
SLARN	73.82	73.26	73.24	82.31	82.62	82.51	82.59	82.59
MIARN	73.34	68.34	70.10	80.77	82.34	82.72	82.78	82.78
SAWS	74.69	74.08	74.34	81.72	83.25	83.40	83.43	83.55
ADGCN	74.81	76.22	75.45	81.75	83.85	83.85	83.85	83.86
本文	**80.18**	**76.27**	**78.20**	**83.91**	**84.19**	**84.19**	**84.19**	**84.30**

表 4　在 Reddit 2 个数据子集上的实验结果

模型	Reddit-1(Movies)				Reddit-2(Technology)			
	精确率/%	召回率/%	F1/%	准确率/%	精确率/%	召回率/%	F1/%	准确率/%
NBOW	67.33	66.56	66.82	67.52	65.45	65.62	65.52	66.55
CNN	65.97	65.97	65.97	66.24	65.88	62.90	62.85	66.80
GRNN	66.16	66.16	66.16	66.42	66.56	66.73	66.66	67.65
CNN-LSTM-DNN	68.27	67.87	67.95	68.50	66.14	66.73	65.74	66.00
ATT-LSTM	68.11	67.87	67.94	68.37	68.20	68.78	67.44	67.22
SLARN	69.59	69.48	69.52	69.84	69.35	70.05	69.22	69.57
MIARN	69.68	69.37	69.54	69.90	68.97	69.30	69.09	69.91
SAWS	71.79	71.77	71.76	71.77	72.50	72.45	72.45	72.48
ADGCN	74.48	74.58	74.47	74.48	75.59	75.59	75.58	75.59
本文	**75.22**	**75.22**	**75.16**	**75.22**	**76.33**	**76.33**	**76.24**	**76.14**

(3)该模型在 Tweets 数据集上都取得了最佳性能。当文本语句过长或过短时,性能明显下降,这表明当文本缺少信息或者冗余信息时,应该进行更深一步的研究。

(4)本文所提出的模型在 6 个数据集上均取得了最优的性能。本模型能够提取文本的不一致性特征、情感特征、句法结构特征和风格特征,采用层次注意力机制调节不同输入对提取特征的影响,而且能够调整不同维度语言特征对讽刺识别的影响,从而有效提高了讽刺识别的性能。

3.4.2　不同语义表示对模型性能的影响

为了验证不同语义特征对模型整体性能的影响,本文对比了不同语义特征的实验性能,对比过程中参数均采用相同的设置,实验结果如表 5 所示,其中 incongruity 表示不一致性特征,affective 表示情感特征,dependency 表示句法结构特征,stylize 表示风格特征。实验数据为每个数据集的 Acc. 值。

表 5 所显示的实验结果表明,本文所提出的多维语义特征表示方法模型,能够更好地获取不同维度的讽刺文本的特征,性能要优于其他模型;风格特征对整体模型性能影响最小,这可能是数据集中缺乏具有明显风格特征的文本,针对这类数据集,直接提取风格特征对模型的性能影响较为有限;值得注意的是,情感特征对模型整体性能的影响较大,说明在讽刺识别任务中,情感信息尤为重要。

表 5　不同语义特征表示对模型性能的影响

模型	IAC-V1	IAC-V2	Tweets-1	Tweets-2	Reddit-1	Reddit-2
本文	**72.60**	**78.33**	**83.91**	**84.08**	**75.22**	**76.14**
w/o incongruity	71.94	77.80	83.10	83.93	74.24	75.83
w/o affective	70.15	76.93	82.15	82.36	73.88	74.12
w/o dependency	71.55	77.69	82.86	83.85	74.10	75.69
w/o stylize	72.23	78.12	83.26	84.00	75.22	76.14

3.4.3　层次注意力机制对模型性能的影响

本文对比了是否使用层次注意力机制(hierarchical attention mechanism, HAM)对模型性能的影响。*w/o* HAM 表示在特征融合阶段没有使用层次注意力机制,而是将多维语义特征直接进行拼接,从而进行讽刺识别。实验数据为每个数据集的 Acc. 值。

实验结果如表 6 所示,结果表明,采用层次注意力机制不仅可以调节不同讽刺语义特征的权重,而且能够调整不同讽刺语言学特征和讽刺语句的关联程度,使得模型识别讽刺的性能得到提高。

表 6　层次注意力机制对讽刺识别性能的影响

模型	IAC-V1	IAC-V2	Tweets-1	Tweets-2	Reddit-1	Reddit-2
本文	**72.60**	**78.33**	**83.91**	**84.08**	**75.22**	**76.14**
w/o HAM	71.83	77.61	83.12	83.77	74.10	74.98

3.4.4　可视化实验

为了直观地表达不同语义特征表示对模型性能的影响,本文针对不一致特征和风格特征的注意力机制进行可视化实验。图 2 和图 3 中展示了本文所提出的模型在不一致性特征和风格特征学习到的讽刺/非讽刺示例的注意力权重。为了可视化注意力权重,本文使用了热图将句子中的每个词语相应的权重值关联起来,通过颜色的变化来显示权重的大小,较高的权重值对应着更重要的单词,而较低的权重值则表示相对不重要的单词。

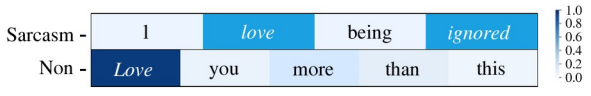

图 2　不一致性特征注意力可视化

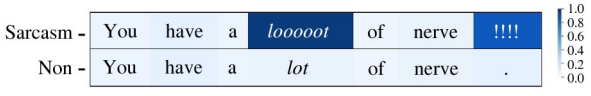

图 3　风格特征注意力可视化

4 结 论

本文针对讽刺文本缺乏多维语义特征表示的问题,从不一致性特征、句法结构特征、情感特征和风格特征4个维度对讽刺特征进行表示。为了更好地权衡不同维度的特征对模型性能的影响,采用层次注意力机制进行讽刺识别。在公开数据集IAC、Tweets和Reddit上的实验结果表明,本文提出的基于多维语义特征与层次注意力神经网络模型能够明显提高讽刺识别性能,取得了该数据集目前已知的最佳性能。

参考文献

[1] GIBBS R W. On the psycholinguistics of sarcasm [J]. Journal of Experimental Psychology: General, 1986,115(1):3.

[2] 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. Seattle, USA: ACL, 2013: 704-714.

[3] CARVALHO P, SARMENTO 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. Hong Kong, China: Association for Computing Machinery, 2009:53-56.

[4] FARÍAS 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):1-24.

[5] JOSHI A, TRIPATHI V, PATEL K, et al. Are word embedding-based features useful for sarcasm detection? [EB/OL]. (2016-10-04)[2023-03-06]. https://arxiv.org/pdf/1610.00883/pdf.

[6] REYES A, ROSSO P, VEALE T. A multidimensional approach for detecting irony in twitter[J]. Language Resources and Evaluation, 2013,47:239-268.

[7] AMIR S, WALLACE B C, LYU H, et al. Modelling context with user embeddings for sarcasm detection in social media[EB/OL]. (2016-07-05)[2023-03-06]. https://arxiv.org/pdf/1607.00976.pdf.

[8] KUMAR A, NARAPAREDDY V T, SRIKANTH V A, et al. Sarcasm detection using multi-head attention based bidirectional LSTM[J]. IEEE Access, 2020,8:6388-6397.

[9] TAY Y, TUAN L A, HUI S C, et al. Reasoning with sarcasm by reading in-between. [EB/OL]. (2018-05-08)[2023-03-06]. https://arxiv.org/pdf/1805.02856/pdf.

[10] PENNINGTON J, SOCHER R, MANNING C D. Glove: global vectors for word representation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing . Doha, Qatar: ACL, 2014:1532-1543.

[11] MAYNARD D G, GREENWOOD M A. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis[C]//The 2014 Language Resources and Evaluation Conference. Reykjavik, Iceland: ELRA, 2014:1-6.

[12] GONZÁLEZ-IBÁNEZ R, MURESAN S, WACHOLDER N. Identifying sarcasm in twitter: a closer look[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, USA:ACL, 2011:581-586.

[13] BAMMAN D, SMITH N. Contextualized sarcasm detection on twitter[C]//Proceedings of the International AAAI Conference on Web and Social Media. Austin, USA: AAAI Press, 2015:574-577.

[14] KIM Y. Convolutional neural networks for sentence classification[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha, Qatar: ACL, 2014:1746-1751.

[15] DAS D, CLARK A J. Sarcasm detection on flick using a CNN[C]//Proceedings of the 2018 International Conference on Computing and Big Data. Charleston, USA: Association for Computing Machinery, 2018:56-61.

[16] 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. Copenhagen, Denmark: ACL, 2017:482-491.

[17] LIANG B, LOU C, LI X, et al. Multi-modal sarcasm detection with interactive in-modal and cross-modal graphs [C]//Proceedings of the 29th ACM International Conference On Multimedia. Virtual, China:ACM, 2021:4707-4715.

[18] HE S, GUO F, QIN S. Sarcasm detection using graph convolutional networks with bidirectional LSTM[C]// Proceedings of the 3rd International Conference on Big Data Technologies. Qingdao, China: ACM, 2020:97-101.

[19] HUANG B, CARLEY K M. Syntax-aware aspect level sentiment classification with graph attention networks. [EB/OL]. (2019-09-05)[2023-03-06]. https://arxiv.org/pdf/1909.02606.pdf.

[20] LOU C, LIANG B, GUI L, et al. Affective dependency graph for sarcasm detection[C]//Proceedings of the 44th

International ACM SIGIR Conference on Research and Development in Information Retrieval. Virtual, Canada: Association for Computing Machinery, 2021:1844-1849.
[21] 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. Osaka, Japan: ACL, 2016:2449-2460.
[22] DUAN S, ZHAO H. Attention is all you need for Chinese word segmentation. [EB/OL]. (2020-10-06)[2023-03-06]. https://arxiv.org/pdf/1910.14537/pdf.
[23] REYES A, ROSSO P, BUSCALDI D. From humor recognition to irony detection: the figurative language of social media [J]. Data & Knowledge Engineering, 2012,74:1-12.
[24] PTÁČEK T, HABERNAL I, HONG J. Sarcasm detection on czech and english twitter[C]// Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics. Dublin, Ireland: ACL, 2014:213-223.
[25] KHODAK M, SAUNSHI N, VODRAHALLI K. A large self-annotated corpus for sarcasm. [EB/OL]. (2018-03-22)[2023-03-06]. https://arxiv.org/pdf/1704.05579.pdf.
[26] PASZKE A, GROSS S, MASSA F, et al. Pytorch: an imperative style, high-performance deep learning library [EB/OL]. (2019-12-03)[2023-03-06]. https://arxiv.org/pdf/1912.01703.pdf.
[27] GHOSH A, VEALE T. Fracking sarcasm using neural network[C]// Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. San Diego, USA:CASS, 2016:161-169.
[28] YANG Z, YANG D, DYER C, et al. Hierarchical attention networks for document classification[C]// Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego, USA: ACL, 2016:1480-1489.
[29] PAN H, LIN Z, FU P, et al. Modeling the incongruity between sentence snippets for sarcasm detection [M]. ECAI 2020:IOS Press, 2020:2132-2139.

Sarcasm recognition based on multi-dimensional semantic features and hierarchical attention mechanism

SONG Liujing[*,**], ZHAO Zefang[*,**], MA Yuxiang[***], SHEN Hanji[*], LI Jun[*,**]
([*]Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190)
([**]University of Chinese Academy of Sciences, Beijing 100049)
([***]School of Computer and Information Engineering, Henan University, Kaifeng 475004)

Abstract

Sarcasm is a complex language expression that plays an important role in everyday communication. With the rapid development of artificial intelligence and social networks, making computers to automatically recognize sarcasm has become one of the hot research topics in the field of natural language processing. Existing research on sarcasm recognition often expresses samantic features from a single dimension, ignoring the subtle differences and importance of samantic features. This paper treats sarcasm recognition as a kind of natural language classification task, in the feature extraction stage, the sarcasm text is represented by multi-dimensional semantic features according to its inconsistency features, affective features, dependency structure features and style features. In the feature fusion stage, the hierarchical attention mechanism is used to adjust the impact of different samantic linguistic features on the overall performance of the model in view of the different contribution and correlation degree of different dimension features to the overall feature. The experimental results show that the proposed model can extract the latent semantic features of satirical text from multiple dimensions, bring a significant improvement on public datasets IAC, Tweets and Reddit.

Key words: sarcasm recognition, natural language processing, multi-dimensional semantic, hierarchical attention mechanism