



面向社交媒体评论的上下文语境讽刺检测模型

韩 虎^{1,2}, 赵启涛¹, 孙天岳¹, 刘国利¹

(1. 兰州交通大学 电子与信息工程学院, 兰州 730070; 2. 甘肃省人工智能与图形图像工程研究中心, 兰州 730070)

摘 要: 讽刺是日常交际中一种常见的语用现象,能够丰富说话者的观点并间接地表达说话者的深层含义。讽刺检测任务的研究目标是挖掘目标语句的讽刺倾向。针对讽刺语境表达变化多样以及不同用户、不同主题下的讽刺含义各不相同等特征,构建融合用户嵌入与论坛主题嵌入的上下文语境讽刺检测模型。该模型借助 ParagraphVector 方法的序列学习能力对用户评论文档与论坛主题文档进行编码,从而获取目标分类句的用户讽刺特征与主题特征,并利用一个双向门控循环单元神经网络得到目标句的语句编码。在标准讽刺检测数据集上进行的实验结果表明,与传统 Bag-of-Words、CNN 等模型相比,该模型能够有效提取语句的上下文语境信息,具有较高的讽刺检测分类准确率。

关键词: 自然语言处理;上下文语境讽刺检测;深度学习;ParagraphVector 模型;双向门控循环单元模型

开放科学(资源服务)标志码(OSID):



中文引用格式: 韩虎,赵启涛,孙天岳,等.面向社交媒体评论的上下文语境讽刺检测模型[J].计算机工程,2021,47(1):66-71.

英文引用格式: HAN Hu, ZHAO Qitao, SUN Tianyue, et al. Contextual sarcasm detection model for social media comments[J]. Computer Engineering, 2021, 47(1): 66-71.

Contextual Sarcasm Detection Model for Social Media Comments

HAN Hu^{1,2}, ZHAO Qitao¹, SUN Tianyue¹, LIU Guoli¹

(1. School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China; 2. Gansu Provincial Engineering Research Center for Artificial Intelligence and Graphic and Image Processing, Lanzhou 730070, China)

[Abstract] Sarcasm is a common pragmatic phenomenon in daily communication that enriches the views of speakers and indirectly expresses their deep meaning. The research goal of sarcasm detection task is to mine the sarcasm tendency of target sentences. As the contexts and expressions of sarcasm is diverse, and the meaning of sarcasm varies according to users and topics, this paper proposes a contextual sarcasm detection model fusing users' embedding and forum topic embedding. The model uses the sequence learning ability of ParagraphVector method to encode the documents of user comments and forum topics to obtain the satirical features of users and topic features of the target sentence. Then a Bi-directional-Gated Recurrent Unit (Bi-GRU) neural network is used to obtain the sentence code of the target sentence. Experimental results on the standard sarcasm detection dataset show that compared with traditional Bag-of-Words, CNN and other models, this model can effectively extract the contextual information of sentences, and has a higher accuracy of sarcasm detection and classification.

[Key words] Natural Language Processing (NLP); contextual sarcasm detection; deep learning; ParagraphVector model; Bi-directional-Gated Recurrent Unit (Bi-GRU) model

DOI: 10.19678/j.issn.1000-3428.0056606

0 概述

如何从社交媒体评论中挖掘用户的情感倾向已成为当前国内外学者的研究重点之一^[1]。由于部分用户在评论语句中频繁使用讽刺表达,并且讽刺语

具有反情感极性的特点,因此针对评论语句的讽刺检测任务成为文本情感分类中一个重要研究领域。与普通的文本情感分类任务不同,讽刺检测问题需要额外的上下文语境信息来补充目标语句实现分类。例如句子:“I am sure Hillary would have done

基金项目: 国家自然科学基金(61562057); 国家社会科学基金(17BXW071); 甘肃省科技计划项目(18JR3RA104)。

作者简介: 韩 虎(1977—),男,副教授、博士,主研方向为机器学习、数据挖掘;赵启涛、孙天岳、刘国利,硕士研究生。

收稿日期: 2019-11-15 **修回日期:** 2020-01-07 **E-mail:** 2328558048@qq.com

that.”,如果不考虑语境,则无法判断它的情感极性,但如果它的作者与Hillary政治倾向相反或者此语句的主题背景具有明显的消极倾向,便会将它与消极极性相联系,其讽刺倾向也愈加明显。

在以往的研究中,很多学者将讽刺检测问题等同于简单的文本分类任务。文献[2]分析了用户在葡萄牙一家在线报纸上发表的评论,发现表情符号、笑的拟声表达以及大量的标点符号、引号等暗示讽刺含义。文献[3]研究表明,人们可以利用同一语句中的明显情绪对比作为言语讽刺的标记。但这种方法无法完全地挖掘语句的讽刺倾向,存在明显的缺陷。例如只给出文本:“Great idea!”或“what a hardship.”,如果不了解说话者的语境(说话者是谁,话题与谁有关以及内容如何与前后语境相关联),则便无法精准预测说话者的意图。因此,讽刺检测任务需要上下文语境信息。

随着研究的不断深入,有部分学者尝试将作者信息嵌入到目标语句向量中参与分类。文献[4]发现Twitter评论中的上下文特征对讽刺文本分类的作用明显,比如作者的历史推文、好友回复信息以及作者评论其他帖子的内容,均对分类的精度有较大影响。文献[5]通过将相似用户投射到嵌入空间的邻近区域来构建用户嵌入。文献[6]在建模用户嵌入时采用更复杂的模型,通过作者个性特征与文体特征的CCA^[7]融合向量来表示用户嵌入,并增加语句的论坛主题特征,在Reddit论坛的标准语料库SARC^[8]上测试并表现出领先的性能。但是Hazarika的模型涉及复杂的特征工程需要耗费大量的资源。

本文构建面向社交媒体评论的上下文语境讽刺检测模型,利用ParagraphVector^[9]方法建模用户与主题嵌入,运用BiGRU模型^[10]建模目标语句,通过实验定性比较模型各组件对分类精度的影响,并根据SARC数据集对模型性能进行验证。

1 相关工作

1.1 上下文语境讽刺检测

基于上下文语境的讽刺检测是讽刺检测领域的细分任务,目的是通过挖掘目标语句的衍生特征来补充判断语句的讽刺倾向。神经网络在自然语言处理领域取得了显著进展,特别是在文本情感分析等子领域的成功应用,推动了学术界将研究方向从依赖于人工提取特征的传统方法迁移到深度学习方法^[11-12]。

近年来,许多学者将神经网络引入到讽刺检测任务中,并取得了比传统方法更好的结果。文献[13]使用预先训练的卷积神经网络来提取语句特征进行讽刺分类。文献[14]使用双向门控递归神经网络来捕获推文的句法和语义信息,并利用池化操作从历史推文中自动地捕获上下文特征,实验结果表明,使用神经网络提取特征提高了讽刺检测的准确性。文献[15]采用卷积

神经网络和长短时记忆(Long Short-Term Memory, LSTM)网络构建的混合模型有效提高了分类精度。文献[16]提出注意力机制和长短时记忆网络的混合神经网络模型对文本进行特征提取。

大部分基于上下文语境的讽刺检测任务是在神经网络模型的基础上进行的,文献[4-6]先后将用户嵌入和论坛主题嵌入引入讽刺检测任务,通过增加额外的上下文语境信息来扩展模型。受文献[4-6]启发,本文提出一种将ParagraphVector方法与BiGRU网络相结合的模型用于基于上下文语境的讽刺检测任务。

1.2 ParagraphVector模型

ParagraphVector模型利用词、句嵌入算法得到单词和句子相应的低维、稠密的分布式表示,有助于后续利用更加成熟的深度神经网络算法对文本进一步处理^[17]。ParagraphVector模型是在Word2vector^[18]的基础上提出的,是一种无监督且不定长文本的连续分布式向量表示方法。

1.3 GRU模型

GRU(Gated Recurrent Unit)模型是长短时记忆网络的改进模型,具有结构简单、参数较少、收敛性更好等优点。相比于其他的循环神经网络,GRU模型运算更加简洁。

GRU中每个单元的更新方式如下:

$$r_t = \sigma(W_r \times [h_{t-1}, x_t]) \quad (1)$$

$$z_t = \sigma(W_z \times [h_{t-1}, x_t]) \quad (2)$$

$$\tilde{h}_t = \tanh(W \times [r_t \times h_{t-1}, x_t]) \quad (3)$$

$$h_t = (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t \quad (4)$$

其中, r_t 、 z_t 、 \tilde{h}_t 和 h_t 分别表示 t 时刻的重置门状态、更新门状态、候选激活状态和激活状态, h_{t-1} 表示 $t-1$ 时刻的隐层状态。

2 基于上下文语境的讽刺检测模型

本文提出的上下文语境讽刺检测模型结构如图1所示,主要包括4个部分:

1)用户嵌入网络。本文将每个用户的历史评论整合为用户文档,通过ParagraphVector模型学习用户文档得到用户嵌入。

2)论坛主题编码。类似于对用户文档的编码,本文将每个论坛的历史主题推文整合为此论坛的主题文档,同样使用ParagraphVector模型学习每个论坛的主题特征。

3)语句编码模型。利用Bi-GRU网络训练目标分类语句,提取句法特征,输出待分类语句的隐层表。

4)分类输出层。将1)~3)部分得到的向量连接,通过一个Softmax层得到最终分类结果。

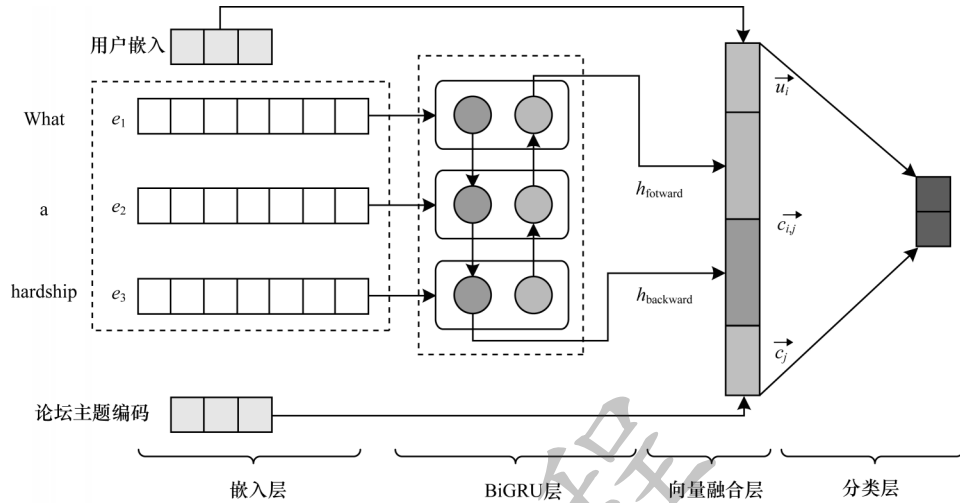


图1 基于上下文语境的讽刺检测模型

Fig.1 Contextual-based sarcasm detection model

本文提出的讽刺检测模型将用户信息、主题信息和目标分类句分别建模,最终组合分类,利用这种建模方式,既可以充分挖掘各模块的语义信息,也能保留三者之间的联系。

2.1 任务定义

此项工作的目的是检测在线论坛(Reddit)评论语句的讽刺倾向。论坛中的每条评论都与唯一的评论用户和评论论坛主题相关联,假定所有的用户构成一个用户集合 $U=\{u_1, u_2, \dots, u_N\}$, 所有的论坛主题构成论坛集合 $T=\{t_1, t_2, \dots, t_{N_t}\}$ 。对于给定的由第 i 个用户 u_i 在第 j 个论坛主题 t_j 上的评论文本 $C_{i,j}$, 本文的任务是判断 $C_{i,j}$ 是否具有讽刺倾向。例如句子“Obama wants people to die from the shutdown so he can blame it on the GOP”, 其语境信息如表1所示。

表1 目标句语境信息

Table 1 Context information of target sentence

任务定义	语境信息
用户	FeuNoir
	An essential classic.<END>
	They were clearly just confused souls who strayed too far from God!<END>
用户文档	I thought it was libraries and roads and shit.<END>
	Shit,everyone here can agree on that tho.<END>
	Those are just minor details anyway.<END>
	...
主题	block user
	Stop attacking her and going off-topic you incompetent pathetic troll.<END>
	Says the one whose user name has two misspellings in it.<END>
主题文档	Yeah they really helped out Haiti.<END>
	And Bernie's reach is far left.<END>
	and then Obama vetos it for some reason.<END>
	They've already picked their candidate.<END>
	...
讽刺倾向	1

由表1可知,目标句所属的用户与主题分别为“FeuNoir”和“block user”,此用户与主题的历史评论数据若干条,用标识符<END>组合为用户文档和主题文档。本文提取用户文档和主题文档的特征来辅助检测目标句的讽刺倾向。

2.2 用户嵌入

用户嵌入试图从用户的历史评论帖子中捕捉与讽刺倾向相关的用户特征,其基本思想是将特征相似的用户投射到嵌入空间的邻近区域。本文采用目前较为常见的文本向量模型 PV-DM (Distributed Memory Model of Paragraph Vectors) 将用户文档表示为段落向量从而实现用户嵌入。PV-DM 模型通过将段落向量与词向量相结合来预测下一个词向量,段落向量与词向量是训练过程的副产物,具体步骤如下:

1)输入层。将每个用户文档映射为一个段落向量存储在段落矩阵 D 中,每个单词映射为词向量存储在词矩阵 C 中。

2)投影层。将词矩阵 C 中的词向量与段落矩阵 D 中的段落向量进行拼接,用于预测 $P(w_{t+c}|w_{t-c}, \dots, w_{t+c}; d) = \frac{e^{y(w_i; d)}}{\sum_i e^{y_i}}$ 。

3)输出层。使用层次 Softmax 求得 $P(w_{t+c}|w_{t-c}, \dots, w_{t+c}; d) = \frac{e^{y(w_i; d)}}{\sum_i e^{y_i}}$ 的概率分布,如式(5)所示,同时通过神经网络的反向传播获得误差梯度,并运用随机梯度下降法迭代更新模型的参数以及段落向量和词向量。

$$P(w_{t+c}|w_{t-c}, \dots, w_{t+c}; d) = \frac{e^{y(w_i; d)}}{\sum_i e^{y_i}} \quad (5)$$

其中,词矩阵 C 中的每一个单词与输出层的节点相对应,而每一个 y_i 对应词 i 未标准化的对数概率:

$$y = b + Uh(w_{t-c}, \dots, w_{t+c}, d; C, D) \quad (6)$$

其中, b 和 U 均为 Softmax 参数, $h(\cdot)$ 表示将词向量与段落向量相连接。训练结束后,用户文档被映射

为定长的向量,即段落向量。使用训练好的PV-DM模型可将任意长度的用户文档 u_i 映射成为用户嵌入向量 u_i 。

2.3 论坛主题嵌入

类似于用户能够影响目标语句的讽刺倾向,本文假定目标语句所属的论坛主题能够影响语句的讽刺倾向。例如,对于政治或体育主题下的评论通常比自然灾害评论更容易具有讽刺意味。本文采用与编码用户嵌入相同的方法编码论坛主题嵌入,使用ParagraphVector模型将Reddit论坛下的每个主题文档 t_j 表示为特征向量 t_j 。

2.4 语句特征

本文使用Bi-GRU模型对目标分类语句建模,即在标准GRU的基础上增加反向的GRU,使得当前时刻的输出能与前一时刻的状态和后一时刻的状态都产生联系,通过将正向GRU与反向GRU相连接形成双向GRU网络。在每一时刻,模型的输入会同时提供给两个方向相反的GRU网络,输出则由这两个单向的GRU共同决定,计算方式如下:

$$\vec{h}_t = \text{GRU}(x_t, \vec{h}_{t-1}) \quad (7)$$

$$\overleftarrow{h}_t = \text{GRU}(x_t, \overleftarrow{h}_{t-1}) \quad (8)$$

$$h_t = w_t \cdot \vec{h}_t + v_t \cdot \overleftarrow{h}_t + b_t \quad (9)$$

其中,GRU(\cdot)表示能够把词向量编码成对应GRU隐层状态的非线性变换函数, w_t 表示 t 时刻双向GRU所对应的前向隐层状态 \vec{h}_t 的权重, v_t 表示 t 时刻双向GRU所对应的反向隐层状态 \overleftarrow{h}_t 的权重, b_t 表示 t 时刻隐层状态所对应的偏置。将目标分类语句的词向量表示为Bi-GRU网络的输入,Bi-GRU网络的主要目的是对输入的文本向量进行文本深层次特征的提取。通过Bi-GRU网络特征提取后,可以更充分地学习上下文之间的关系,提取目标分类文本特征向量 $C_{i,j}$ 。

2.5 模型训练

在提取到评论 $C_{i,j}$ 的文本表示向量 $C_{i,j}$ 、用户 u_i 的用户嵌入向量 u_i 以及论坛 t_j 的主题特征向量 t_j 后,将这3个向量连接起来形成统一的文本表示 $\hat{C}_{i,j} = [C_{i,j} \oplus u_i \oplus t_j]$ 。其中, \oplus 表示连接。最后,使用Softmax函数将 $\hat{C}_{i,j}$ 转换为概率输出,具体公式如下:

$$\hat{y} = \text{Softmax}(W_o \hat{C}_{i,j} + b_o) \quad (10)$$

其中, W_o 是权重矩阵, b_o 是偏置。利用反向传播算法来训练模型,在训练过程中,通过全连接层的每一次输出最小化交叉熵损失函数对模型进行优化:

$$\text{Loss} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^2 y_{i,j} \ln \hat{y}_{i,j} + \lambda \|\theta\|^2 \quad (11)$$

其中, N 是训练集中的所有评论, $y_{i,j}$ 为实际类别, $\hat{y}_{i,j}$ 为模型预测类别, $\lambda \|\theta\|^2$ 为交叉熵正则项。

3 实验结果与分析

3.1 实验数据与实验平台

本文使用Reddit论坛的标准讽刺检测数据集SARC^[7]对模型进行验证,在实验中考虑了SARC数据集的3种变体,分别是Main balanced、Main imbalanced和Politics,数据集信息如表2所示。本文的实验环境信息如下:操作系统为Windows 10;CPU为Inter core™i7-7700 CPU @ 3.60 GHz;显卡为GTX 1080;内存为16.0 GB。实验采用Google的TensorFlow深度学习框架,使用Python语言编程实现。

表2 SARC数据集统计

Table 2 Statistics of SARC datasets

数据集	训练集		测试集	
	评论数	讽刺占比/%	评论数	讽刺占比/%
Main balanced	77 351	50	32 333	50
Main imbalanced	25 784	25	10 778	25
Politics	6 834	50	1 703	50

3.2 评价指标

本文使用准确率Accuracy作为评价模型分类结果的指标。

$$\text{Accuracy} = \frac{T}{N} \quad (12)$$

其中, T 表示模型正确预测评论类别的数量, N 表示样本总数。

3.3 实验参数

在本文的实验中,数据集采用了文献[19]提出的Glove词向量来初始化实验数据的词向量,实验中的许多超参数都需要设置和调整,超参数的更新都是在每一次训练完成后,根据实验的精度和损失手动调整的。经过若干次迭代,实验的超参数如表3所示。

表3 实验参数

Table 3 Experimental parameter

参数	参数值
词向量维度	300
Bi-GRU隐藏层节点数	200
学习率	0.05
训练迭代次数	100
Mini-batch Size	4 096
Dropout rate	0.5
L2-norm regularization	0.5

3.4 对比实验

将本文提出的模型与以下常见的模型进行对比:

1) Bag-of-Words:词袋模型是自然语言处理领域常用的建模文本方法,即以单词作为矢量空间的维度,以文本中出现的单词频率作为文本矢量对应维度的值。

2)CNN:文献[20]提出的卷积神经网络模型使用独立评论句子作为模型的输入,没有嵌入用户与主题信息,是最基础的卷积神经网络模型。

3)Bi-GRU:文献[10]提出的Bi-GRU网络模型是LSTM网络的变体,该模型能够解决句子间的长时依赖问题。

4)CNN-SVM:文献[13]提出的模型使用预训练的CNN网络提取历史语句特征,并与目标分类句特征共同作为SVM模型的输入进行分类。

5)CUE-CNN:文献[5]提出的基于段落向量模型与卷积神经网络的分类模型,首先通过段落向量对用户文档进行建模,然后将用户嵌入与用于对目标句建模的CNN的特征向量相结合得到最终的待分类表示。

6)CASCADE:文献[6]将论坛主题建模引入讽刺检测任务,用户嵌入和论坛主题编码与CNN相结合形成分类向量。

3.5 结果分析

本文在SARC数据集上进行7组模型的对比实验,实验分类准确率如表4所示。

表4 实验分类准确率

Table 4 Accuracy of experimental classification

模型	Main balanced	Main imbalanced	Politics
Bag-of-Words	0.63	0.68	0.59
CNN	0.64	0.68	0.60
Bi-GRU	0.66	0.68	0.64
CNN-SVM	0.68	0.69	0.65
CUE-CNN	0.70	0.71	0.68
CASCADE	0.77	0.79	0.74
本文模型	0.78	0.79	0.73

从表4的实验结果可以看出,Bag-of-Words模型的性能最差,这是因为Bag-of-Words模型忽略了文本中词语的先后次序,而且缺少词语的语义信息。对于所有应用神经网络模型的方法,基础的CNN模型表现最差,对于没有进行上下文嵌入的模型,CNN模型的表现稍逊于Bi-GRU模型,这体现出循环神经网络模型对于处理序列化数据的优势。但上述3种基础模型相比于嵌入语境信息的模型性能较差,这说明只对目标语句建模的方法不能完全挖掘语句的讽刺倾向,因此在模型中加入上下文语境信息很有必要。对于在模型中加入语境信息的情况,CNN-SVM模型在Main balanced、Main imbalanced和Politics 3个数据集上相较基础的CNN模型准确率提升了4%、1%和5%。CUE-CNN模型引入用户嵌入,CASCADE在CUE-CNN的基础上引入了主题嵌入,这2种模型都是利用CNN作为提取内容语句特征的基础模型。本文的模型使用段落向量模型无监督地学习用户嵌入与论坛主题嵌入,并利用Bi-

GRU模型提取内容语句特征。实验结果表明,本文模型的准确率优于除CASCADE的其他模型,这说明用户嵌入与主题嵌入对目标句讽刺倾向检测的重要性,以及本文模型能够提取对讽刺检测有用的用户与主题特征。本文模型与CASCADE的准确率相近,这是由于本文对内容语句的编码采用了比处理序列数据更具优势的Bi-GRU模型,Bi-GRU相比CNN具有更可靠的记忆单元,能够提取更复杂的语句特征信息,并且本文模型简化了CASCADE的用户嵌入方式,避免了建模用户嵌入时大量的特征工程工作。实验结果表明,本文模型优于目前基于上下文语境讽刺检测的其他模型。

3.6 模型分析

为进一步分析模型各组件对语句讽刺检测程度的影响,本文从数据集中抽取一些典型的句子进行分类结果对比分析,如表5所示。从表5结果可以看出,例句1、例句2和例句3由于缺乏上下文语境信息而导致预测错误,但是例句1、例句2在仅加入用户嵌入后模型预测结果便有了明显的改善,而例句3由于长度较短且缺乏明显的情感词,需要对其所属论坛主题明确后才能得到正确的预测。说明本文提出的使用ParagraphVector模型对用户以及主题建模是可行的,并且能够达到较好的效果。例句4在基础Bi-GRU模型(Bi-GRU+U)和增加用户嵌入(Bi-GRU+U+P)的模型中都能得到正确的预测,但当模型增加主题嵌入时得出错误的结果,说明更复杂的模型不一定能得到正确的预测效果。

表5 数据集中典型句分析

Table 5 Typical sentence analysis in dataset

评论	是否 讽刺	模型讽刺比率		
		Bi-GRU	Bi-GRU+U	Bi-GRU+U+P
1.Obama wants people to die from the shutdown so he can blame it on the GOP.	是	0.23	0.70	0.88
2.That's just in bad taste cmon man.	否	0.82	0.34	0.25
3.he UNK he's a Racist.	是	0.32	0.42	0.89
4.Why not use a tag that label them for what they are:lies and propaganda?	否	0.22	0.25	0.78

4 结束语

本文提出一种结合用户嵌入、论坛主题嵌入与内容编码的上下文语境讽刺检测模型。该模型运用ParagraphVector模型与Bi-GRU模型分别编码上下文语境信息和内容信息,有效提高讽刺检测分类准确率。在标准讽刺检测数据集上的实验结果验证了该模型的有效性。对于讽刺检测这类特殊的文本分类问题,需要增加用户嵌入与主题嵌入等语境信息,

但由于用户文档与主题文档的训练样本较少,实验过程中会出现过拟合、分类错误等问题,下一步将对此进行改进,以得到更好的分类效果。

参考文献

- [1] PANG B, LEE L. Opinion mining and sentiment analysis [J]. Foundations and Trends in Information Retrieval, 2008, 2(1/2): 1-13.
- [2] 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. New York, USA: ACM Press, 2009: 53-56.
- [3] RILOFF E, QADIR A, SURVE P, et al. Sarcasm as contrast between a positive sentiment and negative situation [C]//Proceedings of 2013 IEEE Conference on Empirical Methods in Natural Language Processing. Washington D. C., USA: IEEE Press, 2013: 704-714.
- [4] BAMMAN D, SMITH N A. Contextualized sarcasm detection on twitter [C]//Proceedings of the 9th International AAAI Conference on Web and Social Media. [S. l.]: AAAI Press, 2015: 457-468.
- [5] AMIR S, WALLACE B C, LYU H, et al. Modelling context with user embeddings for sarcasm detection in social media [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1607.00976.pdf>.
- [6] HAZARIKA D, PORIA S, GORANTLA S, et al. CASCADE: contextual sarcasm detection in online discussion forums [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1805.06413>.
- [7] HOTELLING H. Relations between two sets of variates [J]. Biometrika, 1936, 28(3/4): 321-377.
- [8] KHODAK M, SAUNSHI N, VODRAHALLI K. A large self-annotated corpus for sarcasm [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1704.05579.pdf>.
- [9] LE Q, MIKOLOV T. Distributed representations of sentences and documents [C]//Proceedings of IEEE International Conference on Machine Learning. Washington D. C., USA: IEEE Press, 2014: 1188-1196.
- [10] CHO K, VAN MERRIENBOER B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1406.1078.pdf>.
- [11] MIKOLOV T, SYTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality [J]. Neural Information Processing Systems, 2013, 26: 3111-3119.
- [12] LECUN Y, BEBGIO Y, HINTON G. Deep learning [J]. Nature, 2015, 521(7553): 436-444.
- [13] PORIN S, CAMBRIA E, HAZARIKA D, et al. A deeper look into sarcastic tweets using deep convolutional neural networks [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1610.08815.pdf>.
- [14] ZHANG Meishan, ZHANG Yue, FU Guohong. Tweet sarcasm detection using deep neural network [C]//Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers. Washington D. C., USA: IEEE Press, 2016: 2449-2460.
- [15] GHOSH A, VEALE T. Fracking sarcasm using neural network [C]//Proceedings of the 7th IEEE Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. Washington D. C., USA: IEEE Press, 2016: 161-169.
- [16] TAY Y, TUAN L A, HUI S C, et al. Reasoning with sarcasm by reading in-between [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1805.02856.pdf>.
- [17] GOLDBERG Y. Neural network methods for natural language processing [J]. Human Language Technologies, 2017, 10(1): 288-309.
- [18] BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model [J]. Journal of Machine Learning Research, 2003, 3: 1137-1155.
- [19] PENNINGTON J, SOCHER R, MANNING C. GloVe: global vectors for word representation [C]//Proceedings of 2014 IEEE Conference on Empirical Methods in Natural Language Processing. Washington D. C., USA: IEEE Press, 2014: 1532-1543.
- [20] KIM Y. Convolutional neural networks for sentence classification [EB/OL]. [2019-10-10]. <https://arxiv.xilesou.top/pdf/1408.5882.pdf>.

编辑 索书志