

文章编号: 1003-0077(2021)01-0104-09

## 基于神经网络融合标签相关性的多标签情感预测研究

陈 玮, 林雪健, 尹 钟

(上海理工大学 光电信息与计算机工程学院, 上海 200093)

**摘 要:** 近年来, 多标签分类任务(MLC)受到了广泛关注。传统的情感预测被视为一种单标签的监督学习, 而忽视了多种情感可能在同一实例中共存的问题。以往的多标签情感预测方法没有同时提取文本的局部特征和全局语义信息, 或未考虑标签之间的相关性。基于此, 该文提出了一种基于神经网络融合标签相关性的多标签情感预测模型(Label-CNN\_LSTM\_Attention, L-CLA), 利用 Word2Vec 方法训练词向量, 将 CNN 和 LSTM 相结合, 通过 CNN 层挖掘文本更深层次的词语特征, 通过 LSTM 层学习词语之间的长期依赖关系, 利用 Attention 机制为情义词特征分配更高的权重。同时, 用标签相关矩阵将标签特征向量补全后与文本特征共同作为分类器的输入, 考察了标签之间的相关性。实验结果表明, L-CLA 模型在重新标注后的 NLP&CC2013 数据集上拥有较好的分类效果。

**关键词:** 多标签分类; 情感预测; 神经网络

**中图分类号:** TP391

**文献标识码:** A

## Neural Network Based Multi-label Sentiment Analysis via Tag Fusion

CHEN Wei, LIN Xuejian, YIN Zhong

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

**Abstract:** In recent years, multi-label-classification-task(MLC) has been widely concerned. Traditional sentiment analysis is regarded as a single label supervised learning, while ignoring the problem that multiple sentiments may coexist in the same instance. This paper proposes a multi-label sentiment analysis model (Label-CNN\_LSTM\_Attention, L-CLA) to fuse the labels via neural network. With the Word2Vec as the input, CNN and LSTM are combined, with the CNN layer dealing with deep word features in the text and the LSTM layer capturing the long-term dependence between words. The attention mechanism is adopted to assign higher weight to the affective words, and the label correlation matrix is integrated to pad the label feature vector as part of the input. The experimental results show that the L-CLA model has a good classification effect on the retagged NLP&CC2013 data set.

**Keywords:** multi-label classification; sentiment analysis; neural network

## 0 引言

情感是人们对客观事物认知的心理表现, 是表达思想的关键。随着微博、豆瓣、Twitter 等社交软件在互联网上的普遍使用, 大量的新闻和评论等数据均可以在网络上获得<sup>[1]</sup>。对社交媒体内容中所表达的情感进行检测和分析, 有利于在商业、公共卫生、社会福利等方面的应用<sup>[2]</sup>。

情感预测的任务是确定对一个目标或主题的态度, 态度可以是一种极性(积极或消极), 也可以是一种情感状态, 如喜悦、愤怒或悲伤<sup>[3]</sup>。以往大多数研究是判断社交媒体内容中所包含的正面或负面态度, 如商品评论中, 顾客对于某个产品是否满意, 或者在影评中, 观众对于影片持精彩或乏味的态度。关于判断情感状态的工作, 大多只关注单标签分类, 将情感文本在多个类别中划分为一个情感类别, 而忽视了多种情感类别可能在一个文本实例中共存的

收稿日期: 2020-06-05 定稿日期: 2020-08-28

基金项目: 国家自然科学基金青年基金(61703277)

情况,这显然与现实不符<sup>[4]</sup>。例如,一个老人有三个儿子,但老了没有一个人照顾他,这可能导致读者悲伤、愤怒或者其他不同的情感。

近年来,多标签分类在很多领域都有了广泛的应用,如文本分类、图像标注、生物信息学领域<sup>[5]</sup>等。目前,针对微博文本进行多标签情感分类的研究尚处于起步阶段<sup>[6]</sup>。第二届自然语言处理与中文计算会议(NLP&CC2013)首次提出了关于中文微博的细粒度情感预测任务,要求为每一个微博文本实例从一组关联的标签中选取多个情感标签。2017年,孙松涛等人<sup>[7]</sup>利用卷积神经网络(convolution neural network,CNN)模型将微博句子中的词向量合成句子向量,并将这些句子向量作为特征训练多标签分类器,在NLP&CC2013数据集上完成微博的多标签情感分类。该方法只提取了文本的局部特征,而没有考虑文本的全局语义信息。2019年,罗锋等人<sup>[8]</sup>提出了一种基于分层注意力的长短时记忆网络(long short term memory networks,LSTM)模型,其使用注意力机制对文本进行表示,之后使用LSTM进行多标签分类,但该方法忽略了文本的局部特征。王莹等人<sup>[9]</sup>在2019年使用FastText模型对爬取的新闻文本数据集进行情感预测,实验结果表明,FastText模型的准确率高于传统的支持向量机、逻辑回归等机器学习方法,速度优势比神经网络模型更加明显。2020年,孙小婉等人<sup>[10]</sup>利用Transformer的多头注意力机制同时获取文本的全局特征和特定方面相关的部分信息,在一定程度上解决了CNN难以获取全局语义信息和RNN训练速度过慢且单词间的依赖程度随着距离增加而衰弱的问题,在Amazon美食评论数据集和Booking酒店评论数据集上完成细粒度情感的分析,但该方法未能考虑到标签的相关性。

在多标签情感预测分类任务中,除了要考虑文本的局部特征和全局语义信息以外,还需要对标签之间的相关性进行充分的建模。姚源林等人<sup>[11]</sup>在2014年提出一种结合上下文特征和篇章特征的多标签微博情感分类方法,该方法以句子为单位,首先对文本中的每个句子进行初始情绪分类,然后通过句子之间的情绪转移关系特征考察每个句子之间的情绪标签相关性,最终在NLP&CC2013数据集上进行实验。但该方法忽视了句子的局部特征。Huang等人<sup>[12]</sup>在2016年提出通过计算标签的余弦相似度来学习标签特征的多标签分类,通过判别不同标签是否能共享标签之间特征的方法考察标签的

相关性。2019年,杨涛等人<sup>[13]</sup>在进行多标签文本分类预测时,通过对所有标签的词向量空间进行 $k$ 近邻检索,将与网络输出向量余弦距离最近的前 $k$ 个标签作为预测的多标签,验证了该方法在标签语义扩展性上的可行性。因此,利用不同标签之间的相关性可有效地提高多标签学习的性能。

现有的多标签情感分类方法有些没有同时提取文本的局部特征和全局语义信息,有些没有充分考虑标签之间的相关性,针对这两个问题,本文提出了一个(Label-CNN\_LSTM\_Attention,L-CLA)模型。利用CNN提取文本中词语的局部特征,并将其与词向量组合,作为LSTM的输入。LSTM作为一种特殊的RNN模型,能够较好地解决长依赖问题<sup>[14]</sup>。在提取词语局部特征和词向量的长期依赖关系后,注意力机制将根据每个文本片段的贡献,在不同的时间步生成文本特征向量,组合成完整的文本特征表达式。同时,为了解决标签之间的相关性,利用标签相关矩阵将标签矩阵补全,与文本特征表达式拼接。最终,在输出层进行文本情感分类,将提取的文本特征与所对应的标签矩阵共同作为分类器输入,增强了分类的精确性。

## 1 L-CLA 模型

为了使表述更加清晰,本文引入了一些符号来描述多标签情感分类的任务。给定一个有 $n$ 个单词的文本 $X = \{x_1, x_2, x_3, \dots, x_n\}$ 和所有标签的集合 $L = \{L_1, L_2, \dots, L_l\}$ ,这里 $l$ 为总的标签数目。提出的模型如图1所示。该模型包含文本特征提取和标签补全两大部分。利用Word2Vec训练得到文本词向量,CNN提取文本 $X$ 中每个词的N-gram表示 $p_j$ , $p_j$ 与词向量 $w_j$ 结合作为LSTM层的输入,产生隐藏状态 $h_j$ 。最后引入注意力机制,在步长 $t$ 内生成文本特征表达式 $S'$ 。同时,由于重新标注的NLP&CC2013数据集中存在只拥有一个情感标签的文本,为了提高多标签分类的性能,根据标签相关性对只拥有一个情感标签的文本进行标签补全,将补全后的标签向量与文本特征表达式拼接,在输出层使用softmax函数进行分类,获取文本所包含的情感类别。

### 1.1 文本特征的提取

文本特征的提取过程由词嵌入层、CNN层和

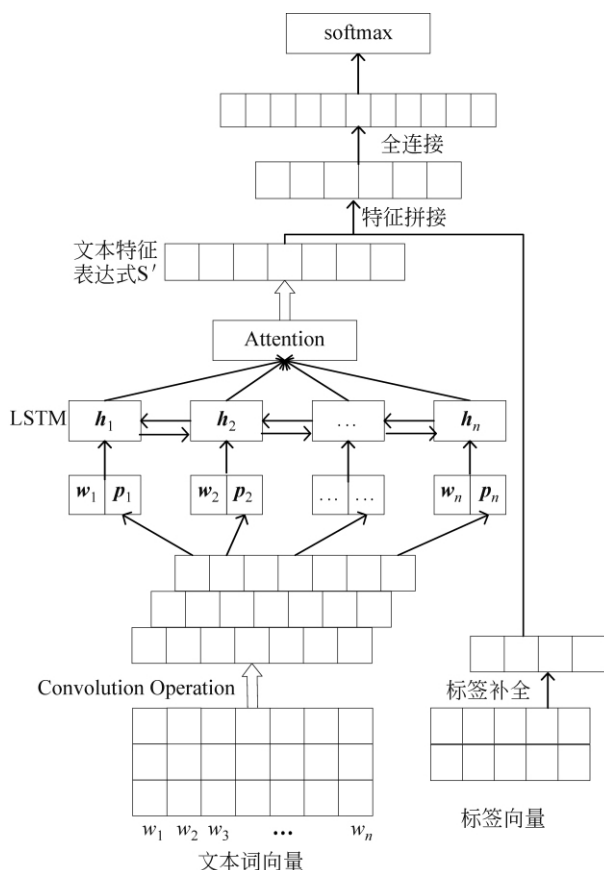


图1 基于神经网络融合标签相关性的模型

LSTM层构成。CNN具有很强的局部语义特征提取能力,已经在文本分类领域得到了广泛的使用。而LSTM可以很好地捕获长期的距离依赖特征<sup>[15]</sup>。为了更好地提取局部特征和全局语义信息,本文将CNN和LSTM结合起来提取文本信息。下面详细说明各个层的具体工作。

### 1.1.1 词嵌入矩阵

该层的目的是将输入的句子转化成词向量,本文使用Word2Vec训练的词向量代替CNN词嵌入层转化的向量。Word2Vec在2013年被谷歌提出,可以很好地在向量空间中找到句子中相似的词<sup>[16]</sup>。将数据进行简单的预处理后,利用预先训练好的Word2Vec模型将文本 $X$ 转换成词向量 $W = \{w_1, w_2, w_3, \dots, w_n\}$ 。由于输入的句子长度不同,因此有必要设置向量的最大维度,如果句子所映射成的向量维度未满足最大维度,则用零向量填充。

### 1.1.2 CNN层

该层使用CNN提取文本 $X$ 中每个词 $x_j$ 的N-gram表示 $p_j$ , $x_j$ 是文本中的第 $j$ 个单词。当CNN用于提取特征时,通常是卷积核和maxpooling层一起工作来生成最重要的局部特征。然而,池操作在

很大程度上降低了特征位置信息,所以我们仅应用卷积不进行池操作,保持了原句的顺序性,使得LSTM层能更好地学习长依赖关系。令 $m \in R^{k \times d}$ 表示长度为 $k$ 、向量维数为 $d$ 的卷积核。对于句子中任何一个词向量 $w_j$ ,都有一个窗口大小为 $k$ 的矩阵,如式(1)所示。

$$W_{j:j+k-1} = [w_j; w_{j+1}; \dots; w_{j+k-1}] \quad (1)$$

其中, $W_{j:j+k-1} \in R^{k \times d}$ ,卷积核 $m$ 依次经过 $S$ ,得到相应的特征映射 $e \in R^{n-k+1}$ ,每一个特征映射 $e_j$ 的计算如式(2)所示。

$$e_j = f(w_{j:j+k-1}^T \cdot m + b) \quad (2)$$

其中, $\cdot$ 是点积, $b$ 是偏置参数, $f$ 是一个非线性函数,使用Relu作为激活函数。使用 $h$ 个窗口大小为 $k$ 的卷积核,我们可以得到 $h$ 个相同长度的特征映射。然后对特征映射进行拼接和转置得到 $W$ ,如式(3)所示。

$$W = [e_1; e_2; \dots; e_i; \dots; e_h]^T \quad (3)$$

其中, $e_i$ 是指由第 $i$ 个核卷积生成的特征映射。令 $p_j$ 为 $W \in R^{(n-k+1) \times h}$ 的第 $j$ 行,是 $h$ 个不同核在 $w_j$ 上运行产生的 $h$ 维向量 $w_{j:j+k-1}$ ,它可以看作是带有 $x_j$ 局部语义信息的N-gram表示。

### 1.1.3 LSTM层

循环神经网络RNN在训练过程中很容易产生长依赖问题,导致梯度消失和梯度爆炸的问题,而LSTM有效解决了这两个问题<sup>[17]</sup>。LSTM包括输入门 $X_t$ 、输出门 $O_t$ 、遗忘门 $f_t$ 、储存层 $C_t$ 以及隐藏层 $h_t$ 。 $W_f, W_i, W_c, W_o$ 和 $b_f, b_i, b_c, b_o$ 分别是每个门的权值矩阵和偏置参数,结构如图2所示。

储存层 $C_t$ 保存 $t$ 时刻的信息。遗忘门 $f_t$ 根据输入 $X_t$ 和 $h_{t-1}$ ,通过sigmoid激活函数判断上一个储存层 $C_{t-1}$ 中哪些信息需要保留,哪些信息需要去除。该层激活函数的判断结果为0或1,分别代表完全去除或完全保留。输入门 $i_t$ 包含两部分信息,一部分是使用包含sigmoid函数的输入门确定哪些信息将要被加入到当前的储存层中,另一部分是使用tanh函数产生新的候选向量 $\tilde{C}$ ,经过输入门和遗忘门之后,将上一层的储存层信息 $C_{t-1}$ 更新为 $C_t$ 。最后,输出层通过sigmoid函数确定哪些信息需要输出,哪些信息还需要经过tanh激活函数对细胞状态进行处理。

在本层中,将CNN层提取的 $p_j$ 与单词向量 $w_j$ 组合为序列 $S$ 作为LSTM的输入,结合上一层所产生的隐藏状态 $h_{t-1}$ ,产生每个词所对应的隐藏状态 $h_t$ ,如式(4)、式(5)所示。

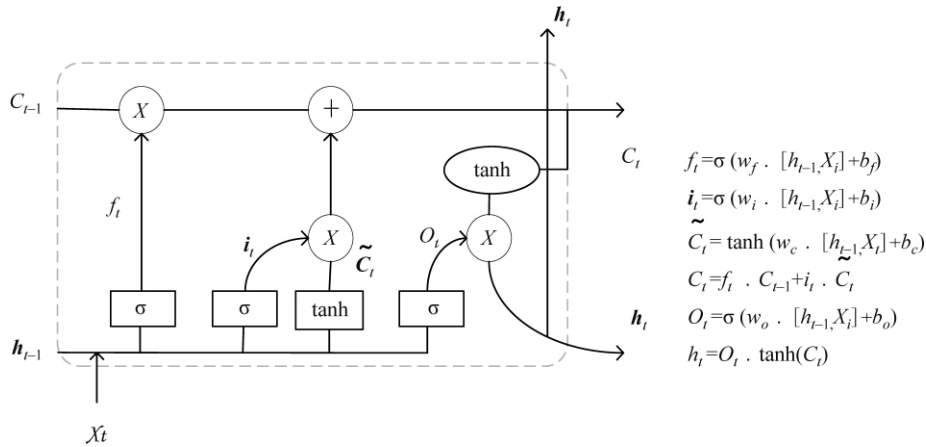


图2 LSTM 结构

$$\mathbf{S} = [g_1, g_2, g_3, \dots, g_n] \quad (4)$$

$$\mathbf{g}_j = [w_j, p_j] \quad (5)$$

使用 LSTM 和 Attention 机制计算每一个词的隐藏状态,如式(6)所示。

$$\mathbf{h}_j = \text{LSTM}(\mathbf{h}_{j-1}, \mathbf{g}_j) \quad (6)$$

#### 1.1.4 注意力机制

对于一个句子而言,并不是所有的单词都同等重要。句子中的一些情意信号,如情意词,通常对句子的情感极性起决定性作用<sup>[18]</sup>。因此,我们利用注意力机制来提取较重要的文本特征表示  $c_t$ 。在步长  $t$  内,注意力机制分配给文本不同部分的权重  $\alpha_{t,j}$ ,文本特征表示  $c_t$  计算如式(7)所示。

$$c_t = \sum_{j=1}^n \alpha_{t,j} \mathbf{h}_j \quad (7)$$

其中,  $\alpha_{t,j}$  是特征向量的一个权重,  $\mathbf{h}_j$  为隐藏状态。为了计算  $\alpha_{t,j}$ ,我们先要利用一层前馈神经网络计算得到  $e_{ij}$  作为  $\mathbf{h}_j$  的一个表示,计算如式(8)所示。

$$e_{ij} = f(\mathbf{W}_s \mathbf{h}_j + \mathbf{b}_s) \quad (8)$$

其中,  $f$  为非线性变换函数,  $\mathbf{W}_s \in \mathbf{R}^{1 \times d_h}$  和  $\mathbf{b}_s \in \mathbf{R}$  分别是权重矩阵和偏置参数,  $d_h$  为  $\mathbf{h}_j$  的维数。权重  $\alpha_{t,j}$  的计算如式(9)所示。

$$\alpha_{t,j} = \frac{\exp(e_{ij})}{\sum_j \exp(e_{ij})} \quad (9)$$

最后在时间步长  $t$  内,  $c_t$  组成文本特征表达式  $\mathbf{S}'$ ,如式(10)所示。

$$\mathbf{S}' = \{c_1; c_2; c_3; \dots; c_t\} \quad (10)$$

## 1.2 标签矩阵的补全过程

在文本实例“太令人生气了。中山南二路宛平路,我的很多长辈都住在那里,土方车真的很吓人,车身沉重,刹车不便,司机视野也不好,正常人看到

都会有意识地避让的,但是有很多土方车闯红灯、抢黄灯已经习惯了,遵纪守法的不乱穿马路的老百姓是不是应该去团购钢铁侠的铠甲啊。”中,该实例只被标注为“愤怒”的标签,但在该语境下,“愤怒”和“恐惧”情感共存的可能性很大。因此,考虑到标签之间的相关性,有必要对仅有一个标签的语料进行标签补全,考察与其相关性较大的标签是否符合该语料所表达的情感,这样有利于提高分类的精确性。

本文通过将标签矩阵  $\mathbf{Y}$  与一个标签相关矩阵  $\mathbf{A}$  相乘,达到补全标签矩阵的目的,即对标签矩阵中的标签向量进行线性变换,得到一个新的标签矩阵。通过欧氏距离度量的方法找到标签之间的距离信息,用标签之间的距离来描述标签集合中标签的相关性,最终得到标签相关矩阵  $\mathbf{A}$ 。标签矩阵  $\mathbf{Y}$  的定义如式(11)所示。

$$\mathbf{Y} = \{L_1, L_2, \dots, L_l\} \quad (11)$$

其中,  $l$  为标签总数目,为了进行标签补全,利用 one-hot 编码对单个标签进行编码。如果文本实例中存在情感标签  $L_1$ ,即  $\mathbf{Y} = [1, 0, \dots, 0]$ 。标签相关性矩阵如式(12)所示。

$$\mathbf{A} = \begin{bmatrix} 1 & w_{12} & \dots & w_{1l} \\ w_{21} & 1 & \dots & w_{2l} \\ \vdots & \vdots & \ddots & \vdots \\ w_{l1} & w_{l2} & \dots & 1 \end{bmatrix} \quad (12)$$

其中,  $\mathbf{A} \in \mathbf{R}^{l \times l}$ 。  $w_{ij}$  ( $1 \leq i \leq l, 1 \leq j \leq l$ ) 的值由欧氏距离得出,具体计算方法如式(13)、式(14)所示。

$$z = \sqrt{\sum_{i=1}^l \sum_{j=1}^l (x_i - x_j)^2 + (y_i - y_j)^2} \quad (13)$$

$$w_{ij} = \frac{1}{1+z} \quad (14)$$

式(13)中,  $(x_i, y_i)$  为标签集合中第  $i$  个标签

的向量,  $(x_j, y_j)$  为标签集合第  $j$  个标签的向量。 $z$  为第  $i$  个标签与第  $j$  个标签的欧氏距离。式(14)中,  $w_{ij}$  为第  $i$  个标签和第  $j$  个标签的相关性值。由于  $z$  为非负数, 将其取倒数使  $w_{ij}$  置于  $(0, 1)$  之间。任意两标签之间的相关性越大, 则  $z$  的值越小,  $w_{ij}$  的值越大。标签补全过程如式(15)所示。

$$\mathbf{Y}^* = \mathbf{Y}\mathbf{A} \quad (15)$$

其中,  $\mathbf{Y}^*$  中数值较大的列所代表的标签与训练文本所标注标签具有较大的相关性, 反之数值较小的列所代表的标签与标注标签之间的相关性较小。

以重新标注后的 NLP&CC2013 数据集为例, 将 8 种情感类别按照“无”“厌恶”“喜好”“高兴”“恐惧”“悲伤”“愤怒”“惊讶”的顺序排列, 则“愤怒”所代表的原始标签向量为  $\mathbf{Y}=[0, 0, 0, 0, 0, 0, 1, 0]$ , 直观上, 该语料中出现了“吓人”一词, “愤怒”可能与“恐惧”具有很强的标签相关性, 因此, 计算标签之间的相关性矩阵  $\mathbf{A}$ 。如式(16)所示。

$\mathbf{A} =$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0.05 & 0.02 & 0.69 & 0.57 & 0.46 & 0.41 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0.46 & 0.04 & 0.01 & 0.58 & 0.46 & 1 & 0.31 \\ 0 & 0.41 & 0.52 & 0.47 & 0.54 & 0.56 & 0.31 & 1 \end{bmatrix} \quad (16)$$

标签“愤怒”与标签“无”“厌恶”“喜好”“高兴”“恐惧”“悲伤”“愤怒”“惊讶”的相关性值依次为 0, 0.46, 0.04, 0.01, 0.58, 0.46, 1, 0.31。因此, 在该实例中标签“愤怒”与标签“恐惧”的相关性值最高, 两个标签很可能共存, 将原始标签补全后得到补全向量  $\mathbf{Y}^*=[0, 0, 0, 0, 1, 0, 1, 0]$ 。

将得到的文本特征向量  $\mathbf{S}'$  和补全后的标签矩阵拼接起来, 如式(17)所示。

$$\mathbf{S}_i = \mathbf{S}' \oplus \mathbf{Y}^* \quad (17)$$

最后利用 softmax 函数进行分类, 如式(18)所示。

$$P(i) = \frac{e^{z_i}}{\sum_1^C e^{z_i}}, i=1, \dots, C \quad (18)$$

其中,  $Z$  是上一层的输出,  $C$  是维度,  $P(i)$  为标签概率。

## 2 实验结果和分析

### 2.1 实验环境

实验操作系统为 Ubuntu16.04.3 LTS、8G 内

存、CPU 为酷睿 i5-7200U、4G 显卡。编程语言为 Python 3, 开发工具为 Pycharm, 使用的深度学习框架为 Tensorflow-1.10。

### 2.2 数据集

由于标注好的多标签情感分析数据集较少, 本文选用了 NLP&CC2013 微博情感分析数据集, 该数据集中情感被分为 8 个类别, 分别为无、厌恶、喜好、高兴、恐惧、悲伤、愤怒和惊讶, 标注好的微博语料有 14 000 条, 每个微博语料有一个或两个情感标签, 训练集有 4 000 条语料, 测试集有 10 000 条语料。但在标注好的测试集中, 情感标签的分布不均衡, 无情感标签的语料有 5 000 多条。因此, 将该数据集进行重新标注。

#### 2.2.1 数据集标注准则

由于每个人对情感判别的标准不一致, 容易出现截然不同的标注结果。因此, 本文在标注过程中, 采用三人独立标注、协同交叉验证的方式, 即每个人先独立标注一部分语料, 之后随机从三个标注人员的标注语料中抽取相同数目的文本语料, 交叉进行评估, 若标注准确率在 90% 以上, 则此次抽取的文本语料标注结果有效。

#### 2.2.2 数据集标签分布

重新标注后的 NLP&CC2013 数据集标签分布如表 1 所示。表中数据为该两种情感标签占标签总数的百分比。本文不区分文本中情感的主次关系, 只对文本中包含哪些情感标签进行探究。从表 1 可以看出, 仅包含一个标签的文本约占三分之一, 对于这部分标签要进行标签补全操作, 不同极性的情感标签共存的情况较少, 如“厌恶”和“喜好”, “高兴”和“愤怒”等, 同极性的情感标签共存的情况较多, 如

表 1 情感标签分布统计(%)

	无	厌恶	喜好	高兴	恐惧	悲伤	愤怒	惊讶
无	2.4	4.3	4.2	5.7	3.7	5.4	4.5	4.2
厌恶	—	—	0.3	0.2	4.8	4.6	5.9	3.3
喜好	—	—	—	9.8	0	0.3	0.1	3.7
高兴	—	—	—	—	0.1	0.1	0	3.5
恐惧	—	—	—	—	—	5.5	6.2	4.1
悲伤	—	—	—	—	—	—	3.5	4.7
愤怒	—	—	—	—	—	—	—	4.9
惊讶	—	—	—	—	—	—	—	—

“喜好”和“高兴”，“厌恶”和“愤怒”等。

最终,对重新标注后的 NLP&CC2013 数据集进行划分并实验,其中训练集 12 000 条语料,验证集 1 000 条语料,测试集 1 000 条语料。

## 2.3 实验设置

### 2.3.1 文本预处理

主要包括文本标准处理和分词。标准处理包括通过正则去噪,将繁体转化成简体等操作。分词则通过结巴分词包对重新标注后的 NLP&CC2013 数据集进行分词处理。最后,建立词典实现文本预处理。建立词典的过程中过滤掉出现次数小于 5 的词,词典中未出现的词用<unk>代替。

### 2.3.2 词向量训练

词嵌入模型 Word2Vec 利用词语与上下文的关系将词语转化为一个低维实数向量,能有效区分一词多义或多词一义<sup>[19]</sup>。本文中使用 Word2Vec 中的 Skip-gram 模型。文本词向量的大小默认值为 256,标签词向量的大小默认为 128,词向量上下文最大距离 window 默认值为 8,在训练语料的过程中,可以根据语料的大小动态调整该值。随机梯度下降法中迭代的最大次数默认值为 5,对于大语料,可以增大这个值。

### 2.3.3 参数的设置

根据本文提出的方法 L-CLA,在参数的选择过程中,通过调优,当 CNN 层的参数设置如表 2 所示时,LSTM 层的参数如表 3 所示,此时取得了较好的分类效果。

表 2 CNN 层参数设置

参数	值
词向量维度	200
滑动窗口大小	3,4,5
滑动窗口数量	128
激活函数	RELU
Pooing 方法	无
Epoch	20
Optimizer	Adam

表 3 LSTM 层参数设置

参数	值
词向量维度	200
学习率	0.001
隐藏层大小	256

续表

参数	值
损失函数	交叉熵
Epoch	20
Optimizer	Adam

## 2.4 评价指标

为了评价 L-CLA 的性能,采用了四种广泛使用的多标签学习的性能评估指标<sup>[20]</sup>,包括汉明损失(HL),1-错误率(OE)、排序损失(RL)和平均精度(AP)。

**汉明损失(hamming loss, HL)**: 用来评估预测标签与相关真实标签不一致的比例,该指标取值越小,则预测的结果越准确,计算如式(19)所示。

$$HL = \frac{1}{N} \sum_{i=1}^N \frac{1}{Q} |h(x_i) \Delta y_i| \quad (19)$$

其中,  $\Delta$  代表预测标签集合与真实标签集合之间的对称差,  $N$  为测试集样本数目,  $h(x_i)$  表示样本  $x_i$  的预测结果,  $Q$  为标签空间的维度,  $y_i$  表示对应的真实标签。

**1-错误率(one-error, OE)**: 用来评估预测结果中第一个标签不在真实标签集合中的概率,计算如式(20)所示。

$$OE = \frac{1}{N} \sum_{i=1}^N [[\arg_{y_j} \min Y_i \in Z_i]] \quad (20)$$

其中,  $[[[]]]$  表示当预测正确时结果返回 0,预测错误时结果返回 1,  $Y_i$  为预测标签结果,  $Z_i$  为真实标签结果,  $\arg_{y_j} \min Y_i$  表示样本位靠前的标签,  $N$  为测试集样本数目。

**排序损失(ranking loss, RL)**: 用来评估预测结果标签集中的排序序列中,错误标签出现在正确标签之前的平均次数。计算如式(21)所示。

$$RL = \frac{1}{N} \times \sum_{i=1}^N \frac{|\{(l_k, l_j) | f_k(x_i) \leq f_j(x_i), (l_k, l_j) \in y_i \times \overline{y_i}\}|}{|y_i| |\overline{y_i}|} \quad (21)$$

其中,  $\overline{y_i}$  为集合  $y_i$  在标签空间中的补集,  $f()$  为多标签分类器所对应的实值函数,  $l_k$  为预测标签,  $l_j$  为真实标签,  $N$  表示测试集样本数目。

**平均精度(average precision, AP)**: 用来度量预测结果标签中正确排序的平均次数。计算如式(22)所示。

$$AP = \frac{1}{N} \sum_{i=1}^N \frac{1}{|y_i|} \sum_{l_k \in y_i} \frac{|\{l_j \mid \text{rank}(x_i, l_j) \leq \text{rank}(x_i, l_k), l_j \in y_i\}|}{\text{rank}(x_i, l_k)} \quad (22)$$

其中,  $y_i$  为标签空间集合,  $l_k$  为预测标签,  $l_j$  为真实标签,  $x_i$  为预测标签概率,  $N$  表示测试集样本数目。平均精度的值越大, 说明预测的结果越准确。

## 2.5 实验结果

为评估 CNN 与 LSTM 结合的模型性能, 将其与单独的 CNN 和单独的 LSTM 进行对比, 在训练过程中, 设置模型提前停止条件, 当验证集 val\_acc 三轮不再增加时保存此模型, 三个模型的测试集对比结果如图 3 所示, 可以看出, CNN 与 LSTM 相结合的模型在测试集上表现优异。

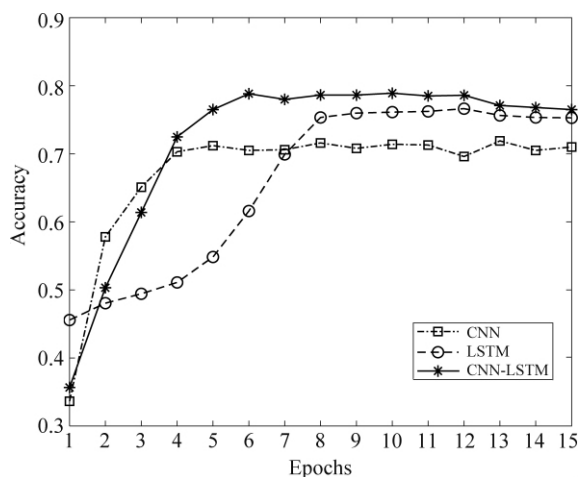


图 3 CNN、LSTM 与 CNN-LSTM 准确率对比

为了评估 L-CLA 模型的性能, 将其与 Unigram+Context、FastText、Transformer 模型、CNN-LSTM 四种多标签分类算法作对比。

(1) **结合上下文和篇章特征的多标签分类方法 (Unigram+Context)**: 该方法利用基分类器对微博文本进行分析得到每个句子的初始情感分类结果, 然后利用相邻句子的转移概率对句子情感类别进行修正。

(2) **FastText 模型**: 该模型相比于传统的词嵌入方法, 增加了字符 n-gram 的嵌入<sup>[21]</sup>, 能够包含单词的局部特征, 具有非常快的训练速度。模型的输入是词序列, 使用层次 softmax 函数计算预定义类上的概率, 利用交叉熵计算损失。

(3) **Transformer 模型**: 该模型包含编码器和解码器两部分, 不同于传统的 CNN 和 RNN, 利用 Self-Attention 可以有效解决 NLP 任务中中长距离依

赖特征的问题。

(4) **CNN-LSTM 结合方法**: 相比于本文方法, 少了标签矩阵的补全过程, 未考虑标签之间的相关性。

5 种方法的对比结果如图 4 所示。可以看出, 在测试集上, 使用词语一元文法特征作为主要特征, 并结合上下文进行微博情感预测的方法效果要低于神经网络的方法, 准确率在 69% 左右。CNN 与 LSTM 相结合的方法准确率达到 75%。FastText 的收敛速度较快, 准确率要低于 CNN 和 LSTM 相结合的方法, 达到 71%。而 Transformer 方法的收敛速度相比于其他方法较慢, 最终的准确率达到 72%。而本文方法 L-CLA 在 CNN 和 LSTM 相结合的基础上添加标签相关信息的补全过程, 能够略微提升准确性, 最终准确率达到 78%。

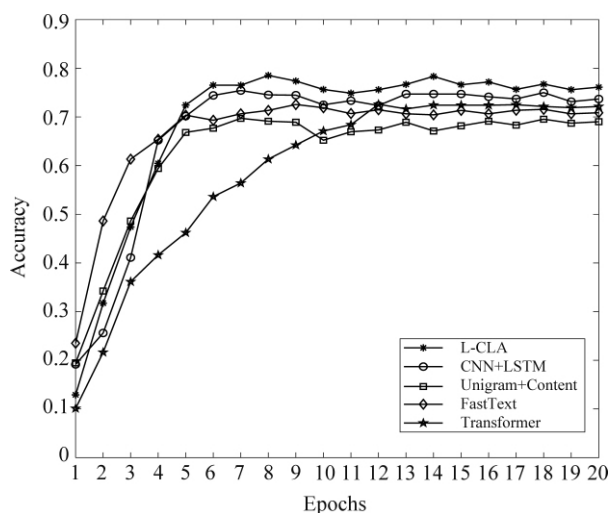


图 4 五种模型准确率对比

表 4 中, HL 代表汉明损失, OE 代表 1-错误率, RL 代表排序损失, AP 代表平均精度。“↓”代表值越小分类效果越好, “↑”代表值越大分类效果越好。从表 4 可以看出, 本文提出的 L-CLA 模型在平

表 4 实验对比结果

模型	HL ↓	OE ↓	RL ↓	AP ↑
Unigram+Context	0.263	0.282	0.237	0.692
FastText	0.242	0.327	0.264	0.713
Transformer	0.238	<b>0.212</b>	0.205	0.724
CNN-LSTM	0.240	0.296	<b>0.191</b>	0.749
L-CLA	<b>0.223</b>	0.268	0.275	<b>0.783</b>

均精度(AP)和汉明损失(HL)方面表现较为优异。相对于使用神经网络方法而言,使用一元特征并结合上下文的方法在各指标上有所欠缺。FastText方法能够利用 N-gram 获取局部词序信息,Transformer 方法能够解决句子长依赖问题,深度学习方法 CNN-LSTM 既提取了局部特征,又提取了全局语义信息,能够较为准确地完成多标签情感分类任务(MLC)。这四种方法均没有考虑到标签之间的相关性,而本文提出的 L-CLA 模型能够将文本局部特征和全局语义信息相结合,还考虑了标签之间的相关性。实验结果表明,L-CLA 具有较好的多标签分类效果。

### 3 结论

本文提出了一种解决多标签分类任务的方法 L-CLA,该方法结合 CNN 和 LSTM 神经网络能够有效地提取局部特征和全局语义信息,利用 Attention 机制为情意词分配更高的权重,同时利用标签相关矩阵补全了每个文本实例所对应的标签矩阵,考察了标签之间的相关性,即一个文本中只标注了一个情感标签,那么与其相关性较大的情感标签是否适合于该文本,对单个情感标签进行标签补全有利于提高多标签分类的精确性。实验证明,本文的模型在重新标注的 NLP&CC2013 数据集上表现较好。

### 参考文献

- [1] Jabreel M, Moreno A. A deep learning based approach for multi-label emotion classification in tweets[J]. *Applied Sciences*, 2019, 9(6): 1123.
- [2] 湛志群, 鞠婷. 基于 BERT 和双向 LSTM 的微博评论倾向性分析研究[J]. *情报理论与实践*, 2020, 43(8): 173-177.
- [3] Saif M Mohammad. Sentiment analysis: Detecting valence, emotions, and other affectual states from text [J]. Woodhead Publishing, 2016: 201-237.
- [4] Kozareva, Zornitsa, Navarro Borja, et al. UA-ZBSA: A headline emotion classification through web information [J]. *Association for Computational Linguistics*, 2007: 334-337.
- [5] Read J, Pfahringer B, Holmes G, et al. Classifier Chains for multi-label classification [C]// *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, Berlin, Heidelberg, 2009.
- [6] 史绍亮, 文益民, 缪裕青. 中文微博情感分类的简单多标签排序算法 [J]. *计算机应用*, 2015, 35(10): 2721-2726.
- [7] 孙松涛, 何炎祥. 基于 CNN 特征空间的微博多标签情感分类 [J]. *工程科学与技术*, 2017, 49(03): 162-169.
- [8] 罗锋. 基于深度学习的文本情绪多标签分类方法研究 [D]. 太原: 山西大学硕士学位论文, 2019.
- [9] 王莹. 基于深度学习的文本分类研究 [D]. 沈阳: 沈阳工业大学硕士学位论文, 2019.
- [10] 孙小婉. 基于深度学习的可解释性情感分析模型研究 [D]. 长春: 吉林大学硕士学位论文, 2020.
- [11] 姚源林. 面向微博文本的情绪分析方法研究 [D]. 哈尔滨: 哈尔滨工业大学硕士学位论文, 2014.
- [12] Huang J, Li G, Huang Q, et al. Learning Label-specific features and class-dependent labels for multi-label classification [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2016, 28(12): 3309-3323.
- [13] 杨涛. 基于标签相关性的文本多标签分类算法的研究 [D]. 北京: 北京工业大学硕士学位论文, 2019.
- [14] Kaladevi P, Thyagarajah K. Integrated CNN and LSTM-DNN-based sentiment analysis over big social data for opinion mining [J]. *Behaviour & Information Technology*, 2019.
- [15] Rehman A U, Malik A K, Raza B, et al. A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis [J]. *Multimedia Tools And Applications*, 2019, 78(18): 26597-26613.
- [16] An H W, Moon N. Design of recommendation system for tourist spot using sentiment analysis based on CNN-LSTM [OL]. <https://doi.org/10.1007/s12652-019-01521-w>.
- [17] Sundermeyer M, Alkhouli T, Wuebker J, et al. Translation modeling with bidirectional recurrent neural networks [C]// *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Doha: Association for Computational Linguistics, 2014: 14-25.
- [18] X Zhou, X Wan, J Xiao. Attention-based LSTM network for cross-lingual sentiment classification [C]// *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016: 247-256.
- [19] 彭俊利, 谷雨, 张震, 等. 融合改进型 TC 与 Word2Vec 的文档表示方法 [J/OL]. *计算机工程*, 2020: 1-7 [2020-06-07]. <https://doi.org/10.19678/j.issn.1000-3428.0056370>.
- [20] Zhang M L, Zhou Z H. A review on multilabel learning algorithms [J]. *IEEE Trans. Knowledge Data Eng.*, 2014, 26(8): 1819-1837.



[21] 赵希明. 基于深度学习的微博情感分析研究[D].北

京:北方工业大学硕士学位论文,2019.



陈玮(1964—),硕士,副教授,主要研究领域为模式识别与智能系统,智能信息处理。

E-mail: chenwei@usst.edu.cn



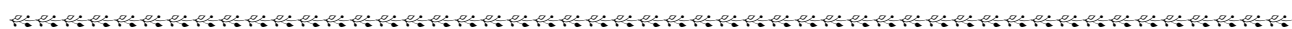
林雪健(1996—),通信作者,硕士研究生,主要研究领域为情感分析。

E-mail: xjbbqcl@126.com



尹钟(1988—),博士,副教授,主要研究领域为生物医学信号处理,情感计算。

E-mail: yinzhong@usst.edu.cn



(上接第 103 页)



王鹏宇(1997—),本科生,主要研究领域为人工智能,信息检索。

E-mail: wpymango@163.com



张敏(1977—),通信作者,博士,副教授,主要研究领域为个性化推荐,信息检索,用户建模。

E-mail: z-m@tsinghua.edu.cn



马为之(1992—),博士,助理研究员,主要研究领域为个性化推荐与用户建模。

E-mail: mawz12@hotmail.com