Computer Science and Application 计算机科学与应用, 2021, 11(5), 1401-1410
Published Online May 2021 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2021.115143

Hans汉斯

基于双向LSTM的文本情感倾向分类

李金宇，王晓晔，彭　宪，田　昊，吉智豪，罗一宁，李金泳

天津理工大学计算机科学与工程系，天津
Email: 18744749014@163.com

收稿日期：2021年4月21日；录用日期：2021年5月17日；发布日期：2021年5月25日

摘　要

随着互联网普及，在网络上出现了大量带有个人主观性的文本，这些文本含有大量情感相关信息和个人的主观观点，目前通常的卷积网络处理这些文本无法将信息进行关联，处理起来无法达到想要的效果，判断文本情感倾向不够准确，所以本文使用国内百度开发的PaddlePaddle框架，构建双向LSTM (Long Short-Term Memory)网络从众多文本信息和数据中准确而高效地分析出文本中所蕴含的情感，并判断情感极性，对情感倾向做出分类。实验中对美食评论信息进行情感预测，首先利用Embedding来计算出词向量，通过双向LSTM提取特征和融合，借助softmax函数构建分类器，获得文本信息的情感倾向，实验结果较为理想。

关键词

情感分析，PaddlePaddle，LSTM

Text Sentiment Classification Based on Bilateral LSTM

Jinyu Li, Xiaoye Wang, Xian Peng, Hao Tian, Zhihao Ji, Yining Luo, Jinyong Li

Department of Computer Science and Engineering, Tianjin University of Technology, Tianjin
Email: 18744749014@163.com

Received: Apr. 21st, 2021; accepted: May 17th, 2021; published: May 25th, 2021

Abstract

With the popularity of the Internet, a large number of personally subjective texts appear on the Internet. These text messages contain a large amount of emotional related information and personal subjective opinions. Some of these information are useless and may cause information explosion. At present, the usual convolutional network processing these texts cannot associate the

文章引用: 李金宇, 王晓晔, 彭宪, 田昊, 吉智豪, 罗一宁, 李金泳. 基于双向LSTM的文本情感倾向分类[J]. 计算机科学与应用, 2021, 11(5): 1401-1410. DOI: 10.12677/csa.2021.115143

information, and the processing cannot achieve the desired effect, so we use the PaddlePaddle developed by Baidu in China based on the bidirectional LSTM (Long Short-Term Memory) network built by us to obtain a large amount of text information, to analyze the emotion contained in the text accurately and efficiently from the data and judge the polarity of the emotion, classify the emotion, and apply it in practice. The model first uses Embedding to calculate the word vector, then uses the two-way LSTM to extract features and fusion, and finally uses the softmax function to construct a classifier to obtain the emotional tendency of the text information.

Keywords

Emotion Analysis, PaddlePaddle, LSTM

1. 引言

文本情感分析是指判断一段文本所表达的情绪状态。文本可以是一个句子，一个段落或一个文档。情绪状态可以是两类，如(正面，负面)，(高兴，悲伤)；也可以是三类，如(积极，消极，中性)等等。情感分析的应用场景十分广泛，如把用户在购物网站(亚马逊、天猫、淘宝等)、旅游网站、电影评论网站上发表的评论分成正面评论和负面评论；或为了分析用户对于某一产品的整体使用感受，抓取产品的用户评论并进行情感分析等。

在文本情感分析领域，早期做出研究贡献的有 Turney 和 Pang，他们运用了多种方法探测商品评论和电影影评的两极观点。此研究是建立在文档级所进行的分析。另一种文档意见的分类方式可以是多重等级的，Pang 和 Snyder 延伸了早先的基础两极意见研究，将电影影评分类并预测为 3 至 4 星的多重级别，而 Snyder 就餐馆评论做了个深度分析，从多种不同方面预测餐馆的评分，比如食物、气氛等等(在一个 5 星的等级制度上)。尽管在大多数统计方面的分类方式中，“中性”类是经常被忽略的，因为“中性”类的文本经常是处于一个两极分类的边缘地带，但是很多研究者指出，在每个两极化问题当中，都应该识别出三个不同的类别。进一步地说，一些现有的分类方式例如 Max Entropy 和 SVMs 可以证明，在分类过程中区分出“中性”类可以帮助提高分类算法的整体准确率。

现有的文本情感分析的途径大致可以集合成四类：关键词识别[1]、词汇关联、统计方法[2]和概念级技术[3]。关键词识别是利用文本中出现的清楚定义的影响词，例如“开心”、“难过”、“伤心”、“害怕”、“无聊”等等，来影响分类。词汇关联除了侦查影响词以外，还赋予词汇一个和某项情绪的“关联”值。统计方法调控机器学习中的元素，比如潜在语意分析，SVM (support vector machines)、词袋等等。一些更智能的方法意在探测出情感持有者(保持情绪状态的那个人)和情感目标(让情感持有者产生情绪的实体)。要想挖掘在某语境下的意见，或是获取被给予意见的某项功能，需要使用到语法之间的关系。语法之间互相的关联性经常需要通过深度解析文本来获取。与单纯的语义技术不同的是，概念级的算法思路权衡了知识表达的元素，比如知识本体、语意网络，因此这种算法也可以探查到文字间比较微妙的情绪表达。

现在处理长文本常见的为 Stack-LSTM (双向栈式 LSTM)，由文献[4]提出。它是 LSTM 的变种，由三层 LSTM 模型堆栈而成，并由正反向 LSTM 结合输出，经最大池化后连接 softmax 构建分类器。

综上所述，为了更好地联系文本上下文信息，本文基于双向栈式 LSTM (长短时记忆网络)网络进行改进，并使用国内的 PaddlePaddle，通过大量实验最后减少栈数并加入 dropout 层使文本情感分析的准确率有明显提升。本文的主要贡献：

1) 对原先的栈式双向 LSTM 网络进行改进，结合正反向 LSTM 获取到更加完整的上下文信息。

2) 在全连接层后加入 dropout 层，丢弃一些神经元，使模型训练减少过拟合现象，提高文本分类的准确率。

3) 将栈式双向 LSTM 网络与优化后的双向 LSTM 网络进行对比，优化过后的网络准确率更高，更具有竞争性。

2. 相关工作

2.1. Embedding 计算词向量

在自然语言处理任务中，词向量(Word Embedding)是表示自然语言里单词的一种方法，即把每个词都表示为一个 N 维空间内的点，即一个高维空间内的向量。通过这种方法，实现把自然语言计算转换为向量计算。

一个句子中词的顺序往往对这个句子的整体语义有重要的影响。因此，在刻画整个句子的语义信息过程中，不能撇开顺序信息。如果简单粗暴地把这个句子中所有词的向量做加和，会使得我们的模型无法区分句子的真实含义，例如：我不爱吃你做的饭。你不爱吃我做的饭。

一个有趣的想法，把一个自然语言句子看成一个序列，把自然语言的生成过程看成是一个序列生成的过程。例如对于句子“我，爱，人工，智能”，这句话的生成概率 P (我，爱，人工，智能)可以被表示为 P (我，爱，人工，智能) = P (我｜<s>)∗P (爱｜<s>，我)∗P (人工｜<s>，我，爱)∗P (智能｜<s>，我，爱，人工)∗P (</s>｜<s>，我，爱，人工，智能)。

其中<s>和</s>是两个特殊的不可见符号，表示一个句子在逻辑上的开始和结束。

上面的公式把一个句子的生成过程建模成一个序列的决策过程，这就是香农在 1950 年左右提出的使用马尔可夫过程建模自然语言的思想。使用序列的视角看待和建模自然语言有一个明显的好处，那就是在对每个词建模的过程中，都有机会去学习这个词和之前生成的词之间的关系，并利用这种关系更好地处理自然语言。如图 1 所示，生成句子“我，爱，人工”后，“智能”在下一步生成的概率就变得很高了，因为“人工智能”经常同时出现。通过考虑句子内部的序列关系，我们就可以清晰地区分“我不爱吃你做的菜”和“你不爱吃我做的菜”这两句话之间的联系与不同了。

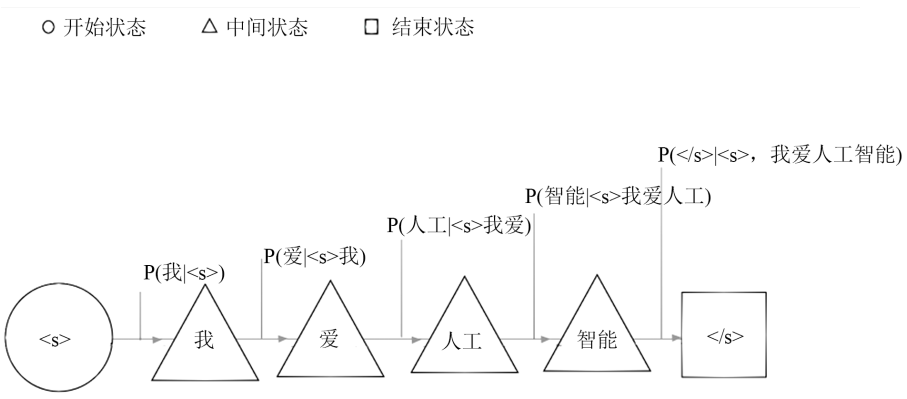

Figure 1. Word modeling learning process
图 1. 词建模学习过程

2013 年，Mikolov 提出的经典 word2vec 算法就是通过上下文来学习语义信息[5]。

本文通过使用国内百度开发的最新的开源深度学习框架 PaddlePaddle (https://www.paddlepaddle.org.cn/)中的 embedding 接口来计算词向量，embedding 是一种 distributed representation，它的出现是相对于 one-hot 的表示法。embedding 在自然语言处理任务中获得了很大的成功，所以也常被翻译为“词向量”。但是，作为一类表示学方法，我们可以为所有离散的输入学习对应的 embedding 表达，并不不局限于自然语言处理任务中的词语。它将高度稀疏的离散输入嵌入到一个新的实向量空间，对抗维数灾难，使用更少的维度，编码更丰富的信息。embeding 层可以理解为从一个矩阵中选择一行，一行对应着一个离散的新的特征表达，是一种取词操作，它的主要目的将词转为相应的向量，这样一个词就可以映射到高维空间，一个常见的做法就是计算词与词之间的距离，因为词被表示成了向量，计算距离就是简单的计算两个向量间的距离，通过这种向量计算获得的值通常都可以看作是两个词之间的相似度。

2.2. 长短期记忆网络(LSTM)

相比于简单的循环神经网络，LSTM 增加了记忆单元 c、输入门 i、遗忘门 f 及输出门 o。这些门及记忆单元组合起来大大提升了循环神经网络处理长序列数据的能力。由于对于较长的序列数据，循环神经网络的训练过程中容易出现梯度消失或爆炸现象。所以 LSTM 能够解决这一问题。他由 Hochreiter S、Schmidhuber J 提出[6]。

LSTM 通过给简单的循环神经网络增加记忆及控制门的方式，增强了其处理远距离依赖问题的能力。类似原理的改进还有 Gated Recurrent Unit (GRU) [7]，其设计更为简洁一些。这些改进虽然各有不同，但是它们的宏观描述却与简单的循环神经网络一样，即隐状态依据当前输入及前一时刻的隐状态来改变，不断地循环这一过程直至输入处理完毕。若将基于 LSTM 的循环神经网络表示的函数记为 F，则其公式为 $h_t = F\left(x_t, h_{t-1}\right)$，$F$ 由下列公式组成：

$$i_t = \sigma\left(W_{xi} + W_{hi} + W_{ci}C_{t-1} + b_i\right) \tag{1}$$

$$f_t = \sigma\left(W_{xf}X_t + W_{hf}h_{t-1} + W_{cf}C_{t-1} + b_f\right) \tag{2}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \tag{3}$$

$$o_t = \sigma W_{xc}X_t + W_{ho}h_{t-1} + W_{co}c_t + b_o \tag{4}$$

$$h_t = o_t \odot \tanh\left(c_t\right) \tag{5}$$

其中 i_t，f_t，c_t，o_t 分别表示输出门，遗忘门，记忆单元及输出门的向量值，带角标的 W 及 b 为模型参数，tanh 为双曲正切函数，⊙表示逐元素(elementwise)的乘法操作。输入门控制着新输入进入记忆单元 c 的强度，遗忘门控制着记忆单元维持上一时刻值的强度，输出门控制着输出记忆单元的强度。三种门的计算方式类似，但有着完全不同的参数，它们各自以不同的方式控制着记忆单元 c，如图 2 所示。

对于正常顺序的循环神经网络，h_t 包含了 t 时刻之前的输入信息，也就是上文信息。同样，为了得到下文信息，我们可以使用反方向(将输入逆序处理)的循环神经网络。结合构建深层循环神经网络的方法(深层神经网络往往能得到更抽象和高级的特征表示)，我们可以通过构建更加强有力的基于 LSTM 的栈式双向循环神经网络，来对时序数据进行建模。通常以三层为例子，如图 3 所示，奇数层 LSTM 正向，偶数层 LSTM 反向，高一层的 LSTM 使用低一层 LSTM 及之前所有层的信息作为输入，对最高层 LSTM 序列使用时间维度上的最大池化即可得到文本的定长向量表示(这一表示充分融合了文本的上下文信息，并且对文本进行了深层次抽象)，最后我们将文本表示连接到 softmax 构建分类模型。

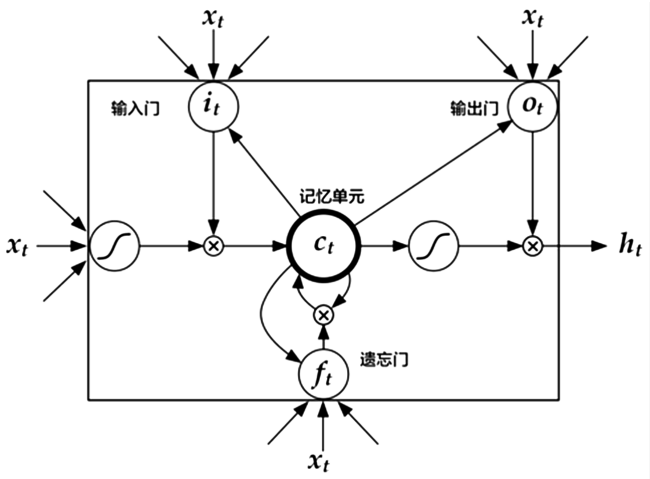

Figure 2. LSTM at time t
图 2. 时刻 t 的 LSTM

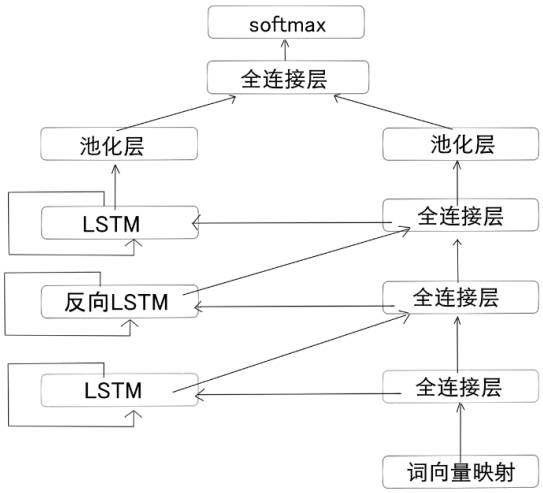

Figure 3. Stacked bidirectional LSTM network for text classification
图 3. 栈式双向 LSTM 网络用于文本分类

3. 模型设计

本文提出了基于栈式双向 LSTM 网络而改进的单层双向 LSTM 网络，再计算词向量后将其分别输入到正向 LSTM 和反向 LSTM 层，并经过隐藏层后加入了 dropout 层，最后连接 softmax 构建分类器。

3.1. Dropout 层

dropout 是指深度学习训练过程中，对于神经网络训练单元，按照一定的概率暂时将其从网络中移除，在设计网络时，设定的每层神经元代表一个学习到的中间特征(即几个权值的组合)，网络所有神经元共同作用来表征输入数据的特定属性(如图像分类中，表征所属类别)。当相对于网络的复杂程度(即网络的表达能力、拟合能力)而言数据量过小时，出现过拟合，显然这时各神经元表示的特征相互之间存在许多重复和冗余。dropout 的直接作用是减少中间特征的数量，从而减少冗余，即增加每层各个特征之间的正交性。dropout 对于解决过拟合问题有着简单有效的特点，文献[8]便探讨了关于 dropout 在过拟合问题上的

作用。本文通过多次调整丢弃概率大小，发现将丢弃概率设置为 0.7 时效果最好，dropout 随机将一些神经元输出设置为 0，其他的仍保持不变。

3.2. LSTM 网络变形及模型结构

本文将栈式双向 LSTM 网络进行了略微修改，改进后的网络层数缩减，保留一个正向 LSTM，一个反向 LSTM 网络，并分别对正反 LSTM 序列使用时间维度上的最大池化即可得到文本的定长向量表示(这一表示充分融合了文本的上下文信息，并且对文本进行深层次抽象)，在加入了 dropout 层后连接至 softmax 构建分类模型，如图 4。使最终的准确率有所提升。

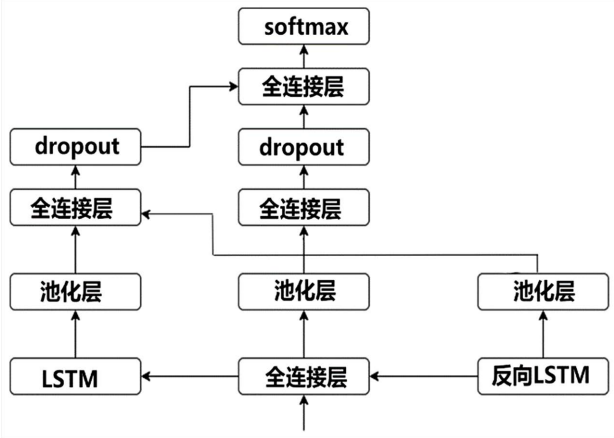

Figure 4. Modified two-way LSTM network for sentiment classification
图 4. 修改后的双向 LSTM 网络用于情感分类

本文的模型主要包含以下三个部分：

计算词向量：当用于分类的文本数据量较大，需要从大量数据中计算出词向量。在计算词向量之前，我们需要获取语料，并且将语料中的字转换为 id，构建词典。一个向量对应一个词，并且通过 embedding 来构建一个二维矩阵。

特征提取：使用了 LSTM 作为特征提取部分模型，它在阅读长序列时，拥有遗忘机制，能很好的处理长文本。比如识别一段语音，X 为其中一句话，我们在识别这句话时，会利用上一句话的信息帮助识别。假设上一句话的信息包含主题的性别，但此时这句话的信息中出现了新的性别，这时候“遗忘门”就起作用了，它会删去上句话中旧的主题性别，同时“输入门”会更新新的主题性别。这样，当前信息状态即可得到一句新的输入。最终我们通过“输出门”决定输出哪部分信息，考虑到主题后可能出现的动词，它可能会输出主题的单复数信息，以便知道如何与动词结合在一起。通过对前期信息有选择的记忆和遗忘，LSTM 实现了对相关信息的长期记忆，从而提取了时间特征。

LSTM 网络训练层：通过正反 LSTM 网络获取两个方向上的信息，同时正反两层都连接相同的输入层，加入 dropout 层随机丢弃一些神经元，提高了网络的泛化能力，包含了三层隐藏层，并最后接入 softmax 构建分类器。

4. 实验

实验采用了国内百度开发的先进的开源框架——PaddlePaddle，它有着很方便的数据接口和模型定义接口。对比比较底层的谷歌深度学习框架 TensorFlow，它直接搭建了深度学习模型的高层。基于它进行

深度学习模型的开发，只需要关注模型的高层结构，不用考虑复杂的底层代码编写的各种问题。

这次实验我们用了两组不同模型，一种是常见的栈式双向 LSTM 网络，它由三层 LSTM 构成，高层 LSTM 网络将得到低层 LSTM 网络的信息。一种是我们改进的 LSTM 网络，仅分别有一层正反 LSTM 网络，测试两组不同的模型对文本情感倾向性分类的准确度。

4.1. 实验数据

语料来自我们从网站上爬取的各种食品评论，以及 PaddlePaddle 官网中他人发布的 chnsenticorp 食品评价数据集，其中 35,696 条作为训练数据集，1200 条作为测试数据集，这些评价分为正面评价和负面评价，如图 5，用于判断一种美食的好吃与否。并且将这些数据集进行预处理，如表 1，包括去除停用词、电话号码、QQ 号、特殊符号等噪声数据。去除这些噪声数据能很好的提高模型的准确率，避免受到干扰。并将数据集乱序，能很好的防止过拟合的问题。

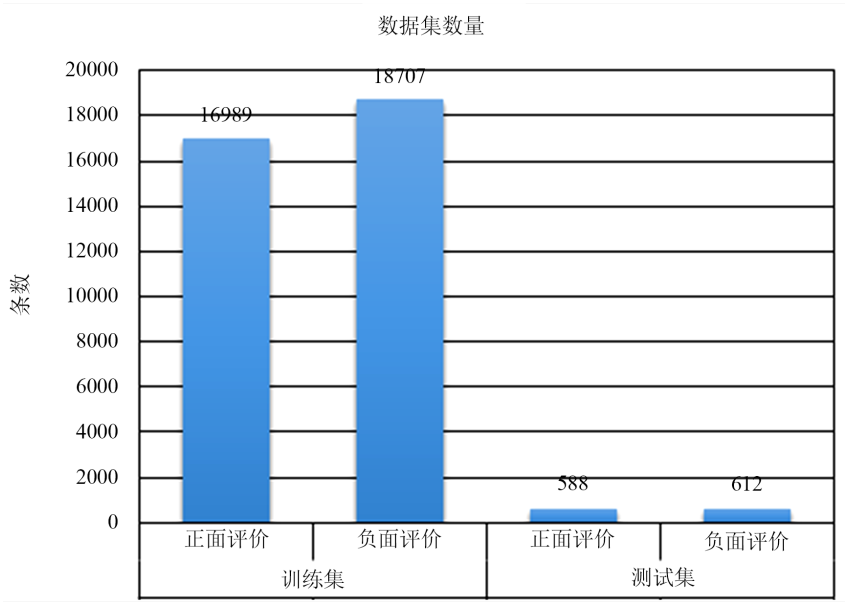

Figure 5. Number of data sets
图 5. 数据集数量

Table 1. Data preprocessing
表 1. 数据预处理

初始数据(1 代表正面，0 代表负面)	预处理后数据
其他还好，就是价格有点贵-----1	其他还好，就是价格有点贵 1
1918894561:服务太差，没人理，当客人是透明的，出品太咸，上菜速度慢。@广发餐厅下次不会再来…… 0	服务太差，没人理，当客人是透明的，出品太咸，上菜速度慢。下次不会再来 0
环境不错，各道菜味道都挺好，份量多，最忘不了就是红米肠了^_^ 1	环境不错，各道菜味道都挺好，份量多，最忘不了就是红米肠了 1

4.2. 参数调整及两组模型对比

先通过 jieba 分词技术将文本进行分词[9]，构建词典，然后计算生成词向量，在训练模型时，通过调整词向量的维度，计算词向量的更新方式，调整 dropout 层的丢弃率以及学习速率来调整参数。如图 6。

栈式总体参数设置		非栈式总体参数设置	
参数键值	参数值	参数键值	参数值
Embedding size	128	Embedding size	128
Pool_type	max	Pool_type	max
Learning	0.0005	Dropout_prob	0.7
Batch size	128	Learning	0.0005
Epoch	30	Batch size	128
Optimizer	Adagrad	Epoch	30
		Optimizer	Adagrad

Figure 6. Training parameter settings
图 6. 训练参数设置

在常见的栈式双向 LSTM 模型中，我们通过调整参数，增加栈的层数，最高准确率也才达到 88%；于是我们想到了修改网络结构，将 dropout 层丢弃率修改为 0.7，多次迭代后发现最高准确率能达到 91%，明显优于第一种 LSTM 模型。且优化后的非栈式模型 loss 整体更低，表明收敛性和收敛速度更快，如图 7、图 8。

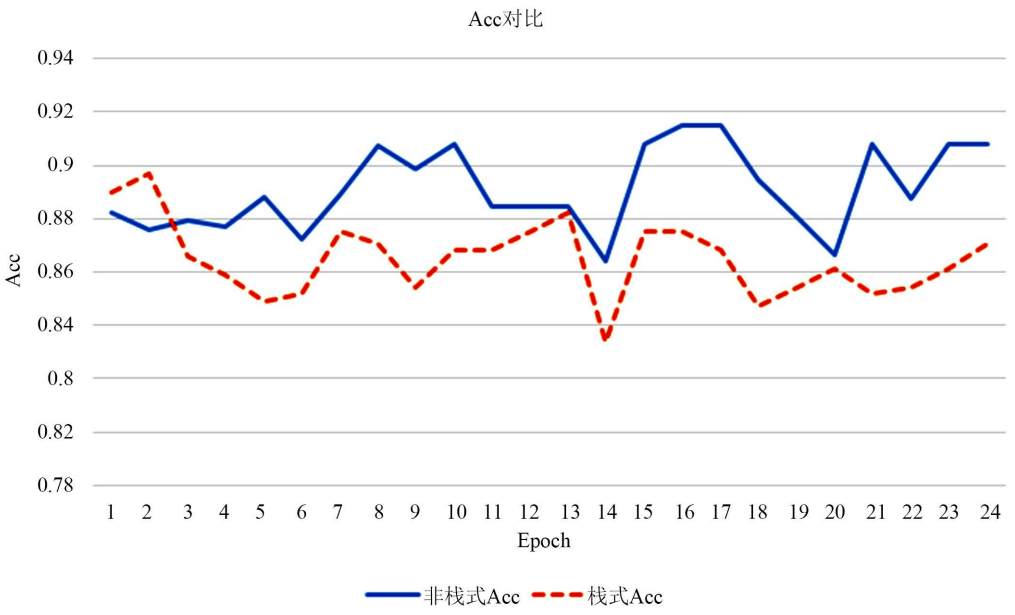

Figure 7. Comparison of model accuracy after optimization
图 7. 优化后模型准确度对比

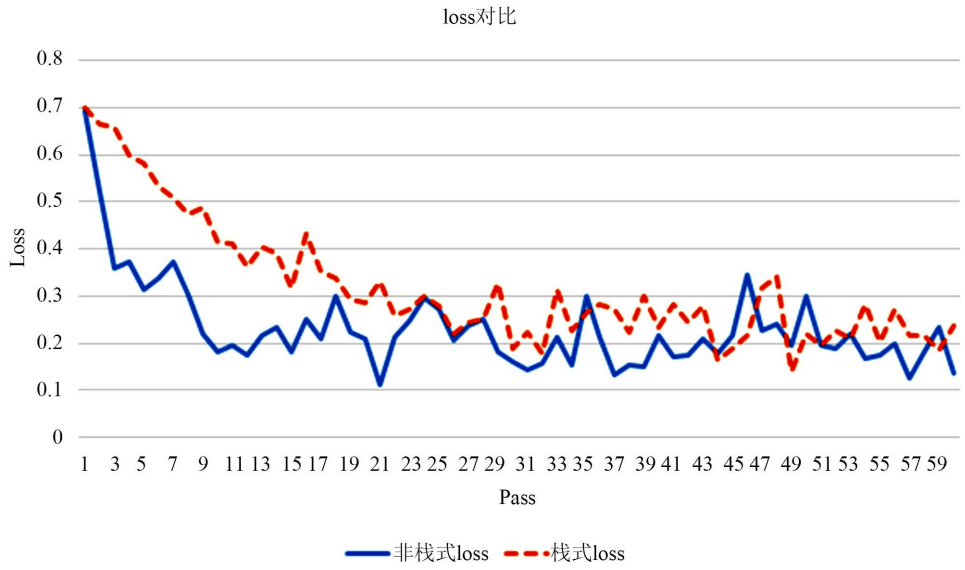

Figure 8. Model loss comparison after optimization
图 8. 优化后模型 loss 对比

4.3. 实验结果分析

通过两组实验对比，分析两种 LSTM 模型的准确度数据。

第一组常见的三层栈式双向 LSTM 网络高层的 LSTM 继承了低层的 LSTM 的信息，最后一个输出记忆，作为整个句子的语义信息，并直接把这个向量作为输入，送入一个分类层进行分类，从而完成对情感分析问题的神经网络建模，拥有较高的准确率。

第二组优化的网络分别将向量输入到正反 LSTM 模型中，并经过两层隐藏层后进入 dropout 层丢弃一些神经元，并最后接入 softmax 层构建分类器，整体拥有更高的准确率。训练步数在 10 步 loss 趋于平缓，表明其拥有更好的收敛性并且收敛速度更快。

5. 结束语

本文在基于三层栈式双向 LSTM 网络上进行了优化，使用 PaddlePaddle 提出了一种新的 LSTM 网络结构，训练正反 LSTM 模型，得到完整上下文信息。通过进行对比不同结构，更好地完成了文本情感倾向性分类任务，能更好地解决评论信息的情感分类化，和文本情感分析中的上下文联系问题。本次实验也有待完善，在数据集选择上，我们应该选用更多种类的数据集来完善提高模型，并提高测试数据集的数量。后续我们将改进这些部分，提高我们的准确度以及研究更多结构去构建分类器。

基金项目

大学生创新创业训练计划项目(202010060159)。

参考文献

[1] Xin, K. and Bublé, M. (2021) On Extracting Keywords from Long-and-Difficult English Sentences for Smart Sentiment Analysis. *Internet Technology Letters*, **4**, e226. https://doi.org/10.1002/itl2.226

[2] Ahmad, S.R., Yusop, N., Bakar, A.A., *et al*. (2017) Statistical Analysis for Validating ACO-KNN Algorithm as Feature Selection in Sentiment Analysis. *AIP Conference Proceedings* **1891**, Article ID: 020018. https://doi.org/10.1063/1.5005351

[3] Saif, H., *et al.* (2014) SentiCircles for Contextual and Conceptual Semantic Sentiment Analysis of Twitter. *The Semantic Web*: *Trends and Challenges*: 11*th International Conference*, *ESWC* 2014, Anissaras, Crete, Greece, 25-29 May 2014, 83-98. https://doi.org/10.1007/978-3-319-07443-6_7

[4] Dyer, C., Ballesteros, M., Ling, W., *et al*. (2015) Transition-Based Dependency Parsing with Stack Long Short-Term Memory. *Computer Science*, **37**, 321-332.

[5] Dey, R. and Salemt, F.M. (2017) Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks. 2017 *IEEE* 60*th International Midwest Symposium on Circuits and Systems* (*MWSCAS*), Boston, MA, 6-9 August 2017, 1597-1600. https://doi.org/10.1109/MWSCAS.2017.8053243

[6] Hochreiter, S. and Schmidhuber, J. (1997) Long Short-Term Memory. *Neural Computation*, **9**, 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[7] Rong, X. (2014) word2vec Parameter Learning Explained. arXiv:1411.2738 [cs.CL]

[8] Srivastava, N., Hinton, G., Krizhevsky, A., *et al*. (2014) Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, **15**, 1929-1958.

[9] 祝永志, 荆静. 基于 Python 语言的中文分词技术的研究[J]. 通信技术, 2019, 52(7): 1612-1619.