结合词向量与传统分类模型的微博情感分析

江洲钰 2015K8009908010

选题：题目14，设计实现一种基于文本内容/情感的文本自动分类方法

具体目标：实现一个分类器，完成对微博文本的简单二元分类，即分为正面、负面。

一、引言与综述

目前以微博为代表的网络社区蓬勃发展，随之而来的是大量的用户评论数据，面向微博文本的情感分析成为舆情监测的重要手段，具有特别的意义。情感分析可以视为文本分类中的一个特例，传统分类方法如朴素贝叶斯、支持向量机等应用广泛，最近兴起的基于卷积神经网络的深度学习方法也有开始应用的例子[1]。

微博情感分析的主要流程分为微博语料获取与预处理、特征选择、分类模型建立、情感分类结果输出。

由于中文微博情感分析发展时间较短，开源的已标注的语料库并不多，其中较权威的是第二届CCF自然语言处理与中文计算会议开放的中文微博情感分析评测数据[2]。但语料库时效性有限，若不使用开源语料库，可以采取爬虫技术爬取实时微博文本，但需要人工标注。

对于特征选择问题的思路可以分为基于词典和基于统计。基于词典的方法需要预先建立情感词典和相应规则集，优点是可以面向微博文本自身特点调整词典，针对性比较好，并且需要的计算资源也较小，缺点是微博新词的产生导致词典的不断更新，文献[3]中建立了一套微博情感词典和规则集，获得了较好结果。基于统计的方法则不需要人工设定的词典，而是从大规模语料中利用机器学习的方法得出规律性的特征，优点是涵盖面广，更新也无需太多人力，缺点是对于语料库的规模有要求，在没有合适的微博已标注语料的情况下针对性不好。由于情感词典建立较困难，现在基于统计的方法应用更多，如文献[4]中根据PMI-IR算法结合语料库统计信息，并采用C4.5算法进行分类；文献[5]中通过提取情感词特征使用SVM对文本进行情感分类。

对于分类模型建立的思路可以分为利用传统分类方法和利用深度学习方法。利用传统分类方法，即朴素贝叶斯、支持向量机（SVM）、随机森林等，一般基于词袋模型，而词袋模型本质上是一种词频统计，忽略了语义联系。利用深度学习方法一般基于词向量模型（word2vec），词向量模型是将词作为特征，并映射到K维向量空间，词转化为向量后可以定义距离以挖掘深层的语义联系。

二、本文工作要点

微博文本不同于正式文本，作为在网络社区上的文本，具有不规范性、流行性、符号混杂性等特点，具体总结了以下四点：

（1）长度不定，但最长不超过140字，一般在80-100字间；

（2）句式简单，一般来说表达情绪很直白，一个或几个关键词就能决定整段文本的情感；

（3）用词不规范，出现大量谐音词，但体裁相近，且会集中出现大量流行词汇；

（4）常用表情符号来辅助表达情绪。

2、4给处理上带来方便，特别是文本化的表情符号能够增加情绪词权重；而1、3是挑战，需要数据量尽量大，涵盖更多的语言现象。

而在中期报告中也提出，由于句子中与情绪表达无关的文本的存在，直接使用分类的方法对于长文本不适用，需要进行过滤，并且传统分类方法忽略了词间的语义联系，使得训练集的规模需求变大，在没有大的微博语料库的情况下也很难取得好的效果。

因此，本文考虑结合词向量和传统分类方法，可以在某个大规模语料库上训练词向量表，继而用微博文本中词向量的平均获得整个文本的向量，用向量作为输入来进行分类。同时，当输入不再需要完整的文本时，过滤无用文本也就成为可能，删除连词、介词等并不会对句子中的关键信息有影响。词向量模型照顾到了语义联系，也符合情绪本质上是语义层面的特点；而传统分类方法训练速度较深度学习方法快，消耗计算资源较小，效果也没有绝对的差距，因此适合在个人电脑上运行。

三、具体方法与流程

1.实验数据

维基百科中文语料（https://dumps.wikimedia.org/zhwiki/latest/zhwiki-latest-pages-articles.xml.bz2）：用于训练词向量表

第二届CCF自然语言处理与中文计算会议中文微博情感分析样例数据：本项目微博语料来源

中科院计算所中文自然语言处理开放平台发布的中文停用词表StopWord.txt：用于过滤无用文本

2.训练流程

首先由维基百科中文语料获取词向量表，借助模块为python中开源的genism，其中包括了针对维基百科语料训练词向量的方法。

之后对微博语料进行预处理，下载的样例数据中并不是简单的正负二元分类，其分类包括：anger 愤怒、disgust 厌恶、fear 恐惧、happiness 快乐、like 喜好、sadness 悲伤、surprise 惊讶、none 中性。我将anger 愤怒、disgust 厌恶、fear 恐惧、sadness 悲伤归为负面，happiness 快乐、like 喜好归为正面，最终获得各1029句的正负平衡语料，命名为neg.txt和pos.txt。

对正负语料进行分词，借助模块为jieba（结巴分词）；使用停用词表进行文本清洗；最终对照词向量表并取平均值获得句子的特征向量。获得的特征向量维数为400，为节约计算资源、加快速度，使用PCA分析作图，发现100维即可包含几乎全部信息，因此降维至100。

随机划分训练集与验证集为95：5。验证集有102句，其中负面48句，正面54句。

使用三种分类模型进行比较研究，为SVM、BP神经网络、随机森林。分别在训练集上训练得到模型，调用模型在验证集上进行验证。

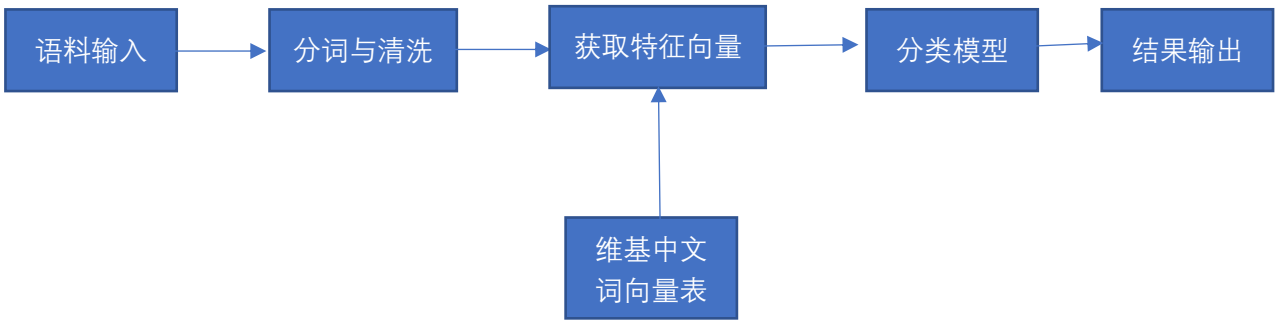

图 1：方法流程

四、实验结果与分析

1.评价指标

类别	属于此类	不属于此类
实际属于此类	TP	FN
实际不属于此类	FP	TN

表 1：混淆矩阵

TP、TN 表示分类结果与实际标签一致的文本数，FN、FP 表示分类结果与实际标签不一致的文本数。

准确率：P=TP/(TP+FP)

召回率：R=TP/(TP+FN)

F1 值：F1=2P*R/(P+R)

2.实验结果

在验证集上测试结果如下表：

模型	类别	准确率	召回率	F1 值
SVM	0	0.61	0.62	0.62
	1	0.66	0.65	0.65
BP 神经网络	0	0.56	0.71	0.62
	1	0.66	0.50	0.57
随机森林	0	0.65	0.69	0.67
	1	0.71	0.67	0.69

表 2：验证结果

3.实验结果分析

（1）比较来看，随机森林效果最好，但差距不大，说明在这样较小的训练集上没有模型占绝对优势。调参时发现增加 BP 的迭代次数的提升效果并不明显，但改变隐藏层层数有较大影响；SVM 中优化核函数是关键；随机森林中要适当减少“树”的数目，增加“树”效果不好。

（2）训练集的规模影响很大。当减少 100 句训练集句子，三种模型的 F1 值普遍下降近 10 个百分点，其中 SVM 受影响最大，BP 神经网络受影响最小。说明这些模型仍然很依赖训练集规模，而本项目中语料库规模本就不够大，应是导致最后验证效果一般的原因之一。

（3）正负验证效果不平衡。SVM 与随机森林均是正面文本效果好，推测有两方面原因：1.在整理语料时，负面有四类情绪而正面有两类，负面情绪词显得更复杂，导致在训练集不够大的情况下涵盖不全面，效果会差一些；2.负面文本中有一类表达忧伤的，表意含蓄，提取特征也困难，导致结果变差。但 BP 神经网络的结果是负面优于正面，不清楚其中原因。

（4）适用于批量处理，而非单条微博分析。每次分析时文本均需经过预处理与向量化，其中最消耗时间的是词向量表的 I/O，待分析文本的规模对耗时影响不大，因此适用于一次处理大量数据，处理单条微博则相较用文本直接分类的方法慢很多。

四、总结

本文针对中文微博情感分析任务，提出了结合词向量和传统分类模型的方法，适用于个人电脑端的批量处理。通过整理出的正负语料，比较研究了 SVM、BP 神经网络、随机森林三种分

类模型的效果，发现差距不大，但模型效果受到训练集规模很大影响。如果能有规模更大的微博标注语料，效果应当能有进一步提升。
中期报告中提到的问题，未解决的是由于微博用语更新迭代速度快，在几年前的语料上训练得到的模型对于现在的微博文本可能不适用。在本文处理框架下，若没有快速更新的微博语料库，则难有好的解决方式。

五、参考资料

部分代码参照了 GitHub 上的开源项目“利用 Python 进行中文情感极性分析”：https://github.com/AimeeLee77/senti_analysis
[1]梁军,柴玉梅,原慧斌等.基于深度学习的微博情感分析[J].中文信息学报,2014,28(5):155-161.
[2] http://tcci.ccf.org.cn/conference/2012/pages/page10_dl.html
[3]王志涛，於志文，郭斌，路新江.基于词典和规则集的中文微博情感分析[J].计算机工程与应用，2015,51（8）：218-225.
[4]周剑峰，阳爱民等.基于二元搭配词的微博情感特征选择[J].计算机工程，2014,40(6):162-165.
[5]杨经，林世平.基于 SVM 的文本词句情感分析[J].计算机应用与软件.2011,28(09):225-228.
[6]张强，陶皖，王海燕.微博情感分析综述[J].安庆师范大学学报（自然科学版）,017,23(4):68-74.
[7]李锐,张谦,刘嘉勇.基于加权 word2vec 的微博情感分析[J].通信技术,2017,50(3):502-506.
[8]斯坦福大学自然语言处理第七课“情感分析（Sentiment Analysis）”
[9]阳爱民,周咏梅,周剑峰. 中文微博语料情感类别自动标注方法[J].计算机应用,2014,34（8）：2188-2191
[10]宗成庆.统计自然语言处理(第 2 版).清华大学出版社