



中山大學
SUN YAT-SEN UNIVERSITY

本科生毕业论文（设计）

Undergraduate Graduation Thesis (Design)

题目 Title: 公众情感分类：传统模型与神经网络模型

院系
School (Department): 数据科学与计算机学院

专业
Major: 软件工程（移动信息工程）

学生姓名
Student Name: 李昕

学号
Student No.: 12353103

指导教师(职称)
Supervisor (Title): 饶洋辉(讲师)

时间：2016年4月20日

Date: Apr. 20 2016

附表一：毕业论文（设计）开题报告

论文（设计）题目：公众情感分类研究：传统模型与神经网络模型

（简述选题的目的、思路、方法、相关支持条件及进度安排等）

开题报告：公众情感分析的目标数据主要是在线新闻文本和社交网络文本。在线新闻数据集的重要特征就是包含有用户投票信息。这些投票信息客观地反映了用户在浏览新闻过后产生的情绪，是属于文本之外的重要特征。然而过去的一些工作并没有重视这个特征。在这篇文章中，我们正是利用了这个特征来计算文档在训练过程中的重要性。情感投票越集中在 1 个或少数几个类上，说明新闻文本的情感特征越突出，从而就具有更大的价值；而对于哪些投票比较分散的文章，我们认为他们是一些噪声文档，因为人去浏览这些文档，都会产生不同的情绪，机器更是如此，所以这些文档对于训练分类器意义不大。我们在文章中提出了 RPWM 和 WMCM 模型来对文档重要性进行区分。另一方面，由互联网用户产生的社交网络文本无法提供情感投票信息，所以我们尝试从提取有意义的特征这个方向去解决问题。收到卷积滤波器的启发，我们提出了 BiCNN 来更好地学习文本的表达。实验结果验证了这 3 个模型的有效性
整个毕业设计安排如下：

2015.5-2015.6 选定毕业题目

2015.7-2015.11 阅读人工智能以及数据挖掘方面的书籍和论文，思考并确定最终毕设方向及论文题目

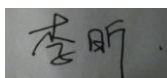
2016.12 撰写开题报告，根据数据集确定代码基本构架

2016.1-2016.2 完成实验代码，得到 baseline 及本文实验结果

2016.3-2016.4 撰写毕业论文，并不断完善修改

2016.5 毕业答辩

学生签名：

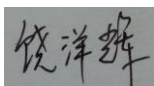


2016 年 04 月 20 日

指导教师意见：

1、同意开题（√） 2、修改后开题（ ） 3、重新开题（ ）

指导教师签名：

Handwritten signature in black ink on a grey rectangular background, reading '饶洋辉'.

2016 年 04 月 20 日

附表二、毕业论文过程检查情况记录表

指导教师分阶段检查论文的进展情况（要求过程检查记录不少于3次）：

第1次检查

学生总结：

- (1) 调整了一些写作不规范的地方(标点符号，字体，措辞，段落对齐方式，行距不一致等)；
- (2) 订正了公式 (3.9)中的乱码错误；
- (3) 调整了部分预定义的符号。

指导教师意见：

上述问题已改正，下一步建议增加数据集，完善实验内容。

第2次检查

学生总结：

增加了社交网络文本数据集，增加了实验内容以及新的基准算法。

指导教师意见：

上述问题已改正。




第3次检查

学生总结：

通过与老师讨论，解决了神经网络训练速度较慢的问题，网络收敛性不好的问题。

指导教师意见：

上述问题已改正。

	<p>学生签名：  2016 年 04 月 20 日</p> <p>指导教师签名：  2016 年 04 月 20 日</p>
<p>总体完成 情况</p>	<p>指导教师意见：</p> <p>为降低噪声训练文档对公众情感分类的影响，本文提出了基于“情感集中度”和“情感熵”的读者视角的加权模型（RPWM）以及加权多标签情感分类模型（WMCM）；对于社交媒体文本，由于无法引入投票信息来过滤掉噪声文档，本文提出了一种浅层的卷积神经网络 BtCNN 来更好地学习文档特征。总体而言，本文所提出的模型创新性较强，实验数据集及结果分析也较完善、优秀。</p> <p>1、按计划完成，完成情况优（√） 2、按计划完成，完成情况良（ ） 3、基本按计划完成，完成情况合格（ ） 4、完成情况不合格（ ）</p> <p>指导教师签名：  2016 年 04 月 20 日</p>

附表三、毕业论文答辩情况

答辩人		专业	
论文（设计）题目			
答辩小组成员			
<p>答辩记录：</p>			

记录人签名：

年 月 日

学术诚信声明

本人所提交的毕业论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料均真实可靠。除文中已经注明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的作品或成果。对本论文的研究作出重要贡献的个人和集体，均已在文中以明确的方式标明。本毕业论文的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

本人签名： 李昕 日期： 2016.03

论文题目：公众情感分类研究：传统模型与神经网络模型

专 业：软件工程（移动信息工程）

学生姓名：李昕

学 号：12353103

指导教师：饶洋辉 讲师

摘 要

随着社交媒体服务的发展，许多用户开始通过新闻，博客，微博等来表达自己的意见和情感倾向。通过计算机自动分析这些自然语言文本中包含的情感信息的方法称为公众情感分析。公众情感分析的目标数据主要是包含读者投票信息的在线新闻文本，以及包含投票信息的社交网络文本(例如：tweet, 微博, facebook, 微信朋友圈等)。对于在线新闻的情感分析，在之前的工作中，每一篇训练文档的权重都是相同的，也就是说，只有文档的语义特征能够影响到分类决策，但有的文档对于人来说都是很难辨别的，计算机就更加难以辨别，所以这类文档对于训练分类器是没有意义的，即，这是一篇噪声文档。

为了降低噪声文档的影响，我们充分利用了“情感投票分布”这一文档的外部特征，提出了基于“情感集中度”和“情感熵”的读者视角的加权模型(RPWM)以及加权多标签情感分类模型(WMCM)，其中“情感集中度”用来衡量一篇文档在训练时的重要性而“情感熵”则是从统计的角度计算了文档在多个情感类上的困惑度。同时，我们使用“主题”信息来识别同一个词中可能包含的不同情感。对于社交媒体文本，由于无法引入投票信息来过滤掉噪声文档，我们提出了一种浅层的卷积神经网络 BtCNN 来更好地学习文档特征。BtCNN 以短语层级的词向量作为输入，避免了一词多义和同义词引起的歧义，同时相邻词组成的短语又间接引入了句子结构信息。另外，因为 BtCNN 的结构简单，所以也在一定程度上降低了参数过拟合的可能性。实验结果表明，我们提出的 WMCM, RPWM 模型，BtCNN 模型在在线新闻文本数据集和社交网络文本数据集上分别取得了很好的效果，验证了模型的有效性。

关键词：情感分析；公众情感检测；情感集中度；多标签分类；卷积神经网络

Title: Social emotion classification: traditional approaches and neural model
Major: Software Engineering (Mobile Information Engineering)
Name: Xin LI
Student ID: 12353103
Supervisor: Assist Prof. RAO Yanghui

Abstract

With the extensive growth of social media services, many users express their feelings and opinions through news articles, blogs and tweets/microblogs. Method using computer to automatically analyze emotional information contained in the text is called social emotion analysis. It mainly focuses on mining valuable information from online news articles and texts in the social network. As with the analysis of online news documents, previous works treat training document equally, that is, only the semantic features of document will affect the classification decision, however, it's hard for people to discriminate some documents, let alone the computer itself. So, these documents are useless when training an emotional classifier.

Based on this viewpoint, we propose RPWM and WMCM, which make full use of external features. RPWM, which is based on “emotional entropy”, computes the perplexity of document by leveraging the statistical information and WMCM induces concept “emotional concentration” to measure the importance of document. For social media text, we try to improve the classification performance by learning good features over documents since no external features is available. We propose BtCNN, a variant of convolutional neural network to detect the feature in the document. The input of BtCNN is distributed word vectors, which maps word to low dimensional vector space. Besides, phrases composed by two neighboring words introduce the order information indirectly. BtCNN does not have deep and complexed architectures, which prevents network from over-fitting effectively. The experiment results proves the effectiveness of the proposed models RPWM, WMCM and BtCNN.

Keywords: Sentiment Analysis, Social Emotion Classification, Emotional Concentration, Multi-label Classification

目 录

摘 要.....	I
ABSTRACT.....	II
第 1 章 引言.....	1
1.1 选题背景与意义.....	1
1.2 本文的研究内容与主要工作.....	3
1.3 本文的论文结构与章节安排.....	3
第 2 章 相关工作.....	4
2.1 情感分类.....	4
2.2 特定领域的应用.....	6
2.3 多标签分类.....	6
第 3 章 在线新闻文本的情感分类模型	8
3.1 通用框架.....	8
3.2 符号定义.....	9
3.3 文档权重的估计方法.....	10
3.4 语义层面的词/文档关联	12
3.5 情感预测.....	14
第 4 章 社交网络文本的情感分类模型	16
4.1 符号定义.....	16
4.2 网络结构与文档特征学习.....	18
4.3 网络训练与情感预测.....	21
第 5 章 在线新闻文本数据集实验	23
5.1 数据集.....	23
5.2 实验设计	24
5.3 与基准算法进行对比	26
第 6 章 社交网络文本情感分类实验	29

6.1 数据集	29
6.2 实验设计	30
6.3 与基准算法对比	32
第 7 章 总结与展望	35
参考文献.....	36
致 谢.....	42
附 录.....	43

第 1 章 引言

1.1 选题背景与意义

随着 Web 2.0 技术的蓬勃发展，互联网上产生了大量的用户参与的，对于诸如人物，事件，产品等有价值的评论信息，这些以文本形式存在的信息表达了在线用户的各种情感色彩以及意见倾向[1]。然而，海量数据的累积逐渐提高了人工处理这些评论信息的成本，人们开始迫切需要计算机来辅助处理大规模的包含情感信息的数据，情感分析技术在这样的背景下应运而生。情感分析，又称意见挖掘，是一门研究在线用户的情感倾向，意见以及对事物的看法的学科[2]。早期的情感分析[3]主要使用有监督的学习方法来区分用户评论的好坏。实验结果表明算法并不会优于传统的文本分类算法，同时在情感预测上效果不是很好。为了提升情感分析系统的性能，许多监督，非监督以及半监督相结合的方法[4]被提出来进行跨领域的情感分类。我们这篇文章的关注点在于公众情感分类，即分析舆论走势以及社交媒体上的用户情感变化。公众情感分类的目标数据包括在线新闻文本和社交网络上产生的用户文本。

挖掘在线新闻的情感信息起源于 SemEval-2007 的第 14 个任务“affective text analysis”[5]。这个任务要求参赛者利用几个预定义的情感标签(“高兴”，“难过”，“感动”等)对新闻标题进行自动标注。比赛中涌现了很多优秀的系统，作为其中性能最优的系统，SWAT[6]采用了有监督的方法从训练集中学得了“词-情感”的映射关系，然后将词和情感之间的关系作为分类预测的重要特征去判断测试集中新闻标题的情感信息。鲍胜华等人在 2009 年和 2012 年分别提出了 Emotion-term(ET)[7]模型和 Emotion-topic(ETM)[8]模型。其中，ET 对词和情感的关系进行了建模。SWAT 和 ET 都是从词层空间去推断文本的情感信息，并在当时都取得了不错的效果，但由于相同的词在不同的语境下表示的含义不同，从而导致词中包含的主观信息和情感倾向性存在差异[9]。基于自然语言歧义性的考虑，ETM 模型在词层与情感层中增加了一层“主题层”。“主题”层的每一个单元都是一个主题，模型中的主题信息通过 LDA[10]来学到，然后这些主题信息用

来区别相同词语中所表达的不同语义。受 ETM 模型的启发，最近涌现了一些主题相关的情感分类模型[11][12]，这些模型都是尝试将文本映射到主题特征空间，然后再计算与情感信息的关联度。这些模型都一定程度上弥补了一词多义和同义词带来的性能下降，但是它们没有能够区分训练文档之间的区别。有些训练文档对于人来说都是很难判断其中的情感的，这些文档对于机器来说就是没有训练价值的，训练时将他们与其他文档视为相同，会无形中引入噪声。结果表明，没有区分训练样本重要性的基准算法的性能波动很大，尤其是在样本数很少或是文本特征有限时[13]。

基于最近的工作的缺点，我们提出了两个“训练样本权重不一致”的模型：读者视角的加权情感分类模型(RPWM)和多标签加权情感分类模型(WMCM)，这两个模型都对训练文档进行了加权，同时使用了主题提取的方法来减少语言歧义引起的分类性能下降。

分析社交网络上用户的情感信息是公众情感分类的又一个重要的方向。Agarwal 等人[46]利用多种 twitter 相关的特征来对用户的 tweet 进行情感极性分析。Jiang 等人[47]则是基于图的优化算法来分析 tweet 对于包含在其中的实体所表现出的倾向性。Tang[50]等人通过构造 twitter 相关的情感词典来提升情感分类的性能。近年来，随着计算能力的提高以及理论研究的突破，基于深度学习的模型开始应用到社交网络相关的情感分类中。Dong[48]等人使用句法解析器将句子转换为树状结构，然后使用递归神经网络来对从短语的层面对文档进行情感分析。Tang[49]等人基于神经语言模型学习 twitter 相关的词向量，学习到的词向量之后作为特征被输入到有监督分类器中。此外，在他们 15 年的工作[51]中，长短期记忆模型(LSTM)被用来进行基于实体的 twitter 情感分类。工作[47][48][51]主要是分析带标注实体的 tweet 文本的情感信息，而[49][50]是通过大规模外部语料来增加情感分类相关的特征。在这篇论文中，我们基于 Kim 等人的工作[52]提出了一种浅层的卷积神经网络(BtCNN)用于情感分析，与之前的工作不同的是，BtCNN 将句子中相邻的两个词视为短语，增强相邻词在上下文中的语义联系，也就是说，模型既不需要句法解析来引入句子结构信息，也不用从外部提取其他特征，而是直接通过输入预训练的词向量来计算短语向量，然后在短语的基础上通过卷积来对文档进行建模。

1.2 本文的研究内容与主要工作

我们在这篇文章中的主要工作如下：

- 对于包含情感投票的在线新闻文本，我们提出了“情感熵”和“情感集中度”两个概念来估计每个训练文本的权重。同时引入主题层，在主题特征的基础上辨别同一个词中可能包含的多种主观性情感。
- 对于不包含情感投票的社交媒体文本，我们提出了一种浅层神经网络 BtCNN 来对文本进行建模，同时使用预训练的词向量来避免自然语言的歧义。
- 对照实验中，我们实现了一些基于主题与基于单词的情感分类模型，同时在在线新闻数据集上对 RPWM 和 WMCM 模型进行评测，在社交网络文本数据集上对 BtCNN 进行评测，最后的实验结果验证了我们提出的这 3 种模型的有效性

1.3 本文的论文结构与章节安排

本文共分为 7 章，章节内容安排如下：在第 2 章，我们介绍了情感分析与多标签分类最近的研究进展与相关工作；接着，在第三章与第四章，我们会分别阐述我们提出的 RPWM，WMCM 模型和 BtCNN 模型。由于在新闻文本与社交网络文本数据集上使用的基准方法与对照方法有所不同，所以我们分两章来介绍实验。第 5 章介绍 RPWM 与 WMCM 模型在新闻文本数据集上与其他基准方法的对照，第 6 章我们用 BtCNN 与一些经典的算法进行对比。最后，我们在第 7 章总结了提出的算法，并展望了未来的工作。

第 2 章 相关工作

这一节中，我们首先介绍了情感分类的最新的研究进展；然后我们列举了一些跟研究相关的实际应用；最后，我们阐述了多标签分类的研究进展。

2.1 情感分类

在最近的一段时间里，情感分类的研究主要集中在产品或者电影评论的分类 [14][15]。Das 和 Chen[16]于 2007 年在工作[17]生成的特定领域的情感词典的基础上，构建了一个跨领域的词典。然而，这个词典在情感分类上的性能严重地依赖于选取的关键词。例如：在股票市场中，句子 “It is not a bear market” 的情感极性是正向的，但是由于 “not” 的存在，基于上面所说的词典的方法往往会认为这句话含有消极的情绪，从而会在一定程度上降低系统的性能。传统的分类方法，例如：朴素贝叶斯，支持向量机和最大熵，被用来提高工作[3]中的效果，但是在文本分类上的效果并不理想。另一方面，研究人员开始将用户的客观信息加入到文本主观情感分析中。Li 等人[18]将文本与 “用户-词” 关系结合起来进行 “主题” 建模。Tan[19]和 Hu[20]等人则是引入 twitter 用户之间的关系，对 tweets 进行情感分析。

对于在线新闻文本的情感分析研究起源于 2007 年 SemEval 语义评测竞赛的第 14 个任务[5] “affective text analysis”。任务的评测数据集由一些手工标注过的新闻标题构成，这些标题来自与 Google News 或者 CNN。任务的目的是构造一个分类系统，从读者的角度来挖掘文本中含有的情感信息。基于这个数据集进行评测的工作涌现了很多。较早期的工作主要通过分析词的极性来对整个文本的情感倾向性进行预测。SWAT[6]主要基于 unigram 模型，系统在计算词的情感极性时，是通过计算每一个包含它的新闻标题在相应情感类上的倾向程度的平均值得到的。Emotion-term(ET)模型[7]则可以视为朴素贝叶斯的变种，系统首先根据训练集得到的情感投票频率计数来随机采样一种情感 e ，然后再根据情感-词的分布对词进行采样，最后采用朴素贝叶斯方法计算后验概率 $P(e|d)$ 。总的来说，这些基于词层空间的方法因为没有解决自然语言歧义的问题，所以在实际的应用中不是太广泛。

最近，一些情感主题模型[8]和多标签主题模型[11][12]被提出并应用到情感分类的任务中。这些方法都是通过引入一个隐层(主题层)来对一个单词可能有多种情感极性的情况进行建模。然而，由于训练样本中可能存在噪声，这些主题模型的性能会出现明显的波动[13]。

另一方面，社交媒体文本的情感分类研究主要分为实体相关与非实体相关的研究。实体相关的情感分类方法需要手动标注或者使用命名实体识别工具来找出文本中的实体，然后分析整个文本对于实体的情感倾向性[47]。这类文本的内容一般是社交网络上的用户对于某一个事物看法或间接。随着近年来 twitter, facebook, 新浪微博等社交网络平台的兴起，越来越多的研究人员开始在社交媒体文本上进行实体相关的情感分类研究。Jiang[47]等人在 2011 年的工作中，通过 3 步来对文本进行情感分析。首先判断文本对于目标实体的是否包含主观情感，然后对于包含用户情绪的文本再进行情感倾向性分类，最后引入外部的推文拓展较短的用户文本并通过图优化算法进行情感分类。Dong 等人[48]提出了一种递归神经网络的变种 AdaCNN。这个网络在二叉句法树的结构上，递归地对输入的词向量进行线性相加，构造短语层级的向量，并用生成的短语向量来表征整篇文档。同时，他们还引入了词性标注，目标实体与上下文的依赖关系等特征来提升分类性能。Vo[53]等人使用预训练词向量和情感词典来从目标实体的上下文以及整篇 tweet 中抽取丰富的特征，然后通过神经池化函数(neural pooling function)对抽取的特征进行过滤。过滤之后的特征被输入到分类器中进行情感检测。Tang[52]等人在没有外部解析工具以及语料库的情况下，使用长短期记忆模型(LSTM)对目标实体上下文进行语义建模。非实体相关的情感分类研究主要是对文本本身进行倾向性判断。这个方向的工作是对社交网络文本的整体情感倾向性进行分析。Tang[54]等人的系统使用束搜索(beam search)对推文进行切割，切割的结果经过情感排序之后被用来进行情感预测。工作[55]提出了一种卷积神经网络的变种 CharSCNN。CharSCNN 使用了两个卷积神经网络架构来对词和句子进行建模。本文中提出的 3 个模型 WMCM, RPWM 以及 BtCNN 都是对文本的整体倾向性进行分析，而不是对文本中具体的某一个实体或概念进行分析。

2.2 特定领域的应用

上述与情感和情绪相关的技术已经被应用到了多个领域当中。Li 等人[21]搭建了一套股票价格预测的系统，这个系统整合了 6 种情感模型，通过这 6 种不同的模型，系统可以自动分析市场行情并预测股票价格。他们的后续工作[22]中还验证了将情感分析技术引入到摘要模型中可以进一步提高系统预测股市行情的能力。

随着网络媒体服务的发展，许多方法开始侧重于社交网络平台的情感分析[23]。Rill 等人[24]将 2013 年德国议会选举前后在 twitter 上出现的近 400 万条 tweet 作为训练语料设计了一个检测突发性政治主题的系统。这个系统不仅能够比 Google Trends 更早地预测政治事件走向，而且能够在 tweet 出现之后立即抽取相应地主题。此外，Bell 等人[25]提出了一种通过分析人使用微博的记录来实现人与计算机的自动交互的方法。他们从社交网络的丰富文本和用户关系中提取了大量重要的特征，极大地提升了人机对话系统的性能。

2.3 多标签分类

从多类别样本中学习有价值的信息是近年来机器学习的又一个热点话题。Ghamrawi 和 McCallum[26]发现条件随机场多分类模型能够通过检测多个类的共现模式(co-occurrence pattern)挖掘多个类标之间的依赖关系。为了解决多标签分类问题中模型输出标签数目不可控的问题，Zhang 等人[27]在传统 KNN 方法的基础上提出了 ML-KNN 方法。Vens 等人[28]在他们的文章中讨论了多种决策树生成的方法，这些方法旨在解决层次多标签分类的问题。同时，他们也通过实验进一步探索了这个方法在基因工程中的应用。为了提高网页分类与网页标签推荐等于网页相关的任务的性能，Tang 等人[29]开发了一个名为 MetaLabeler 的系统，这个系统能够自动为输入网页贴上一系列相关的标签，避免了大量的人为参与和交叉验证。

在最近的进展中，Read 等人的工作[30]表明基于二元关系(binary relevance-based)的方法在不依靠大规模数据集的情况下能够取得不错的预测性能。Dembczynsky 等人[31]以两种标签依赖关系为例详细介绍了发现这种依赖关

系的方法。Montanes 等人[32]提出了另一种称为依存二元关系学习的对标签的依赖关系进行建模的方法。Hong 等人[33]通过将多标签分类模型集成到分类器簇上，开发了一个基于多个专业架构的多标签分类框架。更多关于多标签分类的研究进展可以在工作[34]和[35]中找到。

第 3 章 在线新闻文本的情感分类模型

在这个章节中，我们会介绍这篇文章提出的前两个用于在线新闻文本情感分类的模型，即，读者视角加权情感分类模型(RPWM)和多标签加权情感分类模型(WMCM)。我们首先列举出了模型中的常用记号。然后，我们详细阐述了估计训练样本权重以及将词与文档关联起来的方法。最后，我们介绍了对在线新闻文本中包含的情感进行预测的过程。

3.1 通用框架

我们提出 RPWM 模型和 WMCM 模型是为了解决训练语料中可能存在噪声文档的问题，同时模型也能够辨别同一个词所表达的不同含义。系统框图以及整体流程见图 3-1。

首先，在抽取主题模块，我们使用了非监督的主题模型来对整个语料进行分析。

然后，我们根据训练数据里的情感投票分布来估计训练文档的权重或贡献度。这个过程主要在计算情感集中度/情感熵和文档加权两个模块中进行。

第三，通过抽取的主题信息来计算测试集文档中的词语训练样本(文档)在语义层面的相关度；接着，使用贝叶斯推断预测整篇测试文档的情感极性。最后，我们使用了 *F1* 指标来评测系统性能。

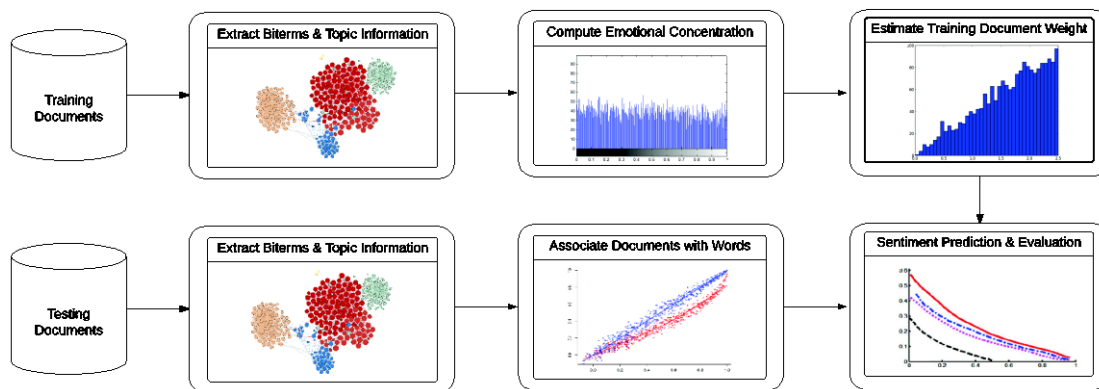


图 3-1 在线新闻文本情感分析系统框架

3.2 符号定义

为了能够更方面的介绍我们所提出的模型，我们首先在这里定义文章中频繁用到的记号：

在线新闻文档集 D 中包含了 $|D|$ 篇新闻，这些新闻一共包含了 W 个不同的词（英文单词或中文单词）。用户在浏览新闻文档之后，可以根据自身感受对 E 个不同的情感标签进行投票。假设预定义的情感标签有：joy, anger, fear, sad 和 surprise(在这个例子中， $E=5$)，在阅读新闻 d 之后，有 1 个用户投给了 joy, 2 个用户投给了 anger, 3 用户投给了 fear, 4 个用户投给了 surprise, 同时没有用户投票给 sad, 那么 d 的情感投票分布可以表示为 $\{1, 2, 3, 0, 4\}$ 。

在我们提出的两个模型中，我们使用了 K 个主题来对整个新闻语料的语义进行建模。参数 θ_d 和 ϕ_w 分别代表文档 d 与单词 w 的主题分布，超参数 α 是与主题相关的分布的狄利克雷先验参数，直观上可以认为是在训练之前，每个主题中已有的文档数目或是单词的实例个数。

在主题提取的过程中，我们除了使用 LDA[10]，还使用了 Cheng 等人在 2014 年提出的 Biterem Topic Model (BTM) [36] 来学习文档和词的主题表达。BTM 通过捕捉双词(同一个滑窗中的任意两个词)在整个语料库中的共现关系来解决短文本上下文信息不足的问题。我们将滑窗的大小记为 λ ，整个语料库中存在的双词个数为 B 。 B 会随着 λ 的增长而增长。给定一篇文档 $d = \{w_1, w_2, w_3\}$ ，如果 $\lambda = 2$ ，那么 d 中包含的双词有 $\{w_1, w_2\}$ 和 $\{w_2, w_3\}$ ，而当 λ 取 3 时，将会生成 $\{w_1, w_2\}$ ， $\{w_2, w_3\}$ 和 $\{w_1, w_3\}$ 这 3 个双词。表格 3-1 中列出了第 3 章中的常用记号。

表 3-1 第 3 章常用记号

符号	符号定义
D	训练文档集
K	主题个数
E	情感标签的个数
W	语料库词汇表大小
θ_d	文档 d 的主题分布

φ_w	单词 w 的主题分布
α	θ_d 和 φ_w 的狄利克雷先验因子
B	文档集生成的双词的个数
λ	BTM 模型滑窗的大小
D_e	情感类 e 中包含的训练文档集

3.3 文档权重的估计方法

在以往的倾向性分类任务中，规定每篇文档只能传达出一种情感(即：正面或负面)。而新闻门户网站所提供的情感类往往不是互斥的，也就是说，同一篇新闻可以让用户产生多种情感。假设有两篇文章 d_1 和 d_2 的情感投票总数相同，他们的投票序列分别是： $\{10,0,0,0,0\}$ 和 $\{3,2,2,2,1\}$ ($E=5$)。我们首先将这些投票数分布归一化，这样就得到两个标准的概率分布 $P(e|d)$ ： $\{1,0,0,0,0\}$ 和 $\{0.3,0.2,0.2,0.2,0.1\}$ 。如果根据多数服从少数的原则，那么 d_1 和 d_2 会被归为同一个类，但是很显然，对于第一个类来说， d_1 是更重要的，因为 d_1 在情感上的歧义程度基本没有(其他类的投票都为 0)而 d_2 还有可能传达第二个类，第三个类的情感信息，所以在学习分类器的过程中， d_2 可能就会引入噪声。基于这样的想法，我们提出了两个概念来表征训练文档的重要性。第一个是 RPWM 中的“情感熵”(emotional entropy)。第二个是 WMCM 中的“情感集中度”(emotional concentration)。

“情感熵”这一概念源自数学中的“熵”。后者用来描述一个离散型随机变量的混乱程度[37]，变量的概率分布越均匀，熵值越大，随机变量越混乱。对于每一篇文档的情感投票，我们同样可以将他视为一个取值为 e_1 到 e_E 的离散型随机变量，如果情感投票分布的熵值越大，代表每一个情感类上的投票越平均，对于机器来说，就越不容易识别这篇文档真正的情感；相反的，如果情感投票分布的

熵值越接近于 0，则说明情感投票只集中在少数几个甚至 1 个类上，在这几个类上的情感特征会更突出，也就是说，这篇文档是训练分类器的高质量样本。基于这样的想法，我们在 RPWM 模型中加入“情感熵”来衡量训练文档的重要性，公式如下：

$$P_{RPWM}(d) = 1 - Entropy_d \quad (0.1)$$

$$Entropy_d = -\sum_{i=1}^E P(e_i | d) * \log_E(P(e_i | d)) \quad (0.2)$$

其中， $Entropy_d$ 表示文章 d 的情感熵， $P(e_i | d)$ 表示在给定文档 d 的情况，情感投票出现在类 e_i 中的概率。另外，我们将对数的底数设置为 E ，保证了情感熵的值在 0 到 1 之间， $P_{RPWM}(d)$ 表征的是文档 d 在训练分类器时的重要性。

“情感集中度”则是根据情感投票的统计信息来计算一篇文档的重要程度。我们先对一篇文档的情感投票数进行升序排序，即令 $v_1 \leq v_2 \leq \dots \leq v_E$ ， \bar{v} 为这些投票数的算数平均值。与熵类似，当 $v_1 = v_2 = \dots = v_E = \bar{v}$ ，文档的情感集中度取到最小值，即代表文档的“情感困惑度”最高，而当 $v_1 = v_2 = \dots = v_{E-1} = 0, v_E = E * \bar{v}$ 时，“情感集中度”取到最高值，因为此时文档 d 只会引起一种读者情感。为了计算一般化的“情感集中度”的值，我们参考了 [38] 中计算累计百分比 F 和 Q 的公式。在我们提出的加权多标签分类模型(WMCM)中， F 跟 Q 的定义如下：

$$F_i = \frac{i}{E} (i = 1, \dots, E) \quad (0.3)$$

$$Q_i = \frac{\sum_{j=1}^i v_j}{E * \bar{v}} (i = 1, \dots, E) \quad (0.4)$$

计算完 F 跟 Q 值之后，我们用他们来估计每一个情感类 e 中文档的“情感集中度”，公式如下：

$$P(d | e) = \frac{\sum_{i=1}^{E-1} (F_i - Q_i)}{\sum_{i=1}^{E-1} F_i} \quad (0.4)$$

我们提出的情感集中度有以下几个性质：

- a. 当 $F_i = Q_i (i=1, \dots, E)$ 时, $P(d|e)=0$, 文档 d 在类 e 上的情感集中度为最小值, 此时文档中含有的噪声最大。
- b. 当 $Q_i = 0 (i=1, \dots, E-1)$ 并且 $F_E = Q_E$ 时, $P(d|e)=1$, 文档 d 在类 e 上的情感集中度达到最大值, 情感特征突出, 对于训练分类器而言是最好的样本。

在上文中也说过, “情感熵” 主要作用于离散型随机变量, 而 “情感集中度” 能够同时应用在连续型和有序的离散型随机变量上。因此, “情感集中度” 这项指标能够用来在情感排序问题[39]中衡量文档的重要性。“情感集中度” 越高, 表示文档在训练过程中的重要性越大, 当所有用户都把投票投给同一个情感类时, “情感集中度” 达到最大值, 文档在所属情感类中的重要程度也达到最高。

3.4 语义层面的词/文档关联

这一小节的目的在于估计训练文档 d 与测试集文档中的词 w 的联合概率。最直接的方法就是基于单词层面的共现关系来计算这个概率值, 但由于同一个词的不同实例可能发达不同的情感(即: 情感歧义性), 我们选择加入主题层, 从语义的层面去分析词和文档之间的关系。

工作[40][41]将 LDA[10], PLSA[42]等主题模型应用到情感分析中, 以此来降低单词歧义性对系统性能的影响。在 LDA 模型中, 文档中的每个词将会以以下的方式来生成:

- 从狄利克雷分布 $Dir(\alpha)$ 中随机采样一个文档-主题分布 θ_d
- 对于文档 d 中的每一个词 w :
 - 从多项分布 $Multinomial(\theta_d)$ 中随机采样一个主题 z
 - 从多项式分布 $Multinomial(\varphi_z)$ 随机采样一个单词 w, w 即为生成的单词

Gibbs Sampling 等[43]统计方法被用来学习有意义的主题。我们在读者视角的加权模型(RPWM)中使用了 LDA[10]来学习主题。RPWM 中估计文档-主题分布的方法如下:

$$\theta_d^{(z)} = \frac{n_d^{(z)} + \alpha}{n_d + K * \alpha} \quad (0.5)$$

θ_d 为文档 d 的主题分布, n_d 是文档 d 中的单词的数目, $n_d^{(z)}$ 代表文档 d 中被分派给主题 z 的单词实例的个数, 平滑参数 α 目的是为了避开 0 概率的出现

因为 LDA 的主题学习过程中比较依赖于上下文, 而短文本又有明显的上下文信息缺乏的特点, 所以我们在多标签加权分类模型中使用了另一种主题模型 **Biterm Topic Model(BTM)**[36]来学习主题信息。BTM 通过引入“双词”自动增加文本的上下文信息, 在滑窗内的任意两个词都可以组成一个“双词”, 即, 这两个词是有语义关联的。在迭代过程中, BTM 是针对双词进行主题指派, 因此 WMCM 模型中的文档-主题分布的估计方法与 RPWM 中略有不同:

$$\theta_d^{(z)} = \frac{n_d^{(z)} + \alpha}{B_d + K * \alpha} \quad (0.6)$$

其中, B_d 为文档 d 所生成的双词的个数, $n_d^{(z)}$ 代表文档 d 所生成的双词中被分派到主题 z 的双词的数目。

主题信息提取完成之后, 我们进行词语和文档的关系分析。给定一篇文档 d 中包含有 100 个单词, 某一个单词 w 在整个语料库中的实例有 100 个, 一共有 4 个主题来对整个文档集进行建模。假设 d 中被分派到主题 1,2,3,4 下的单词个数分别为 10,20,30,40; w 的实例中被分派到主题 1,2,3,4 下的个数也为 10,20,30,40。通过统计信息, 我们可以知道文档 d 主要表达的是主题 4 的含义; 而 w 在主题 4 中出现得最频繁。如果我们将主题 4 实例化为“machine learning”, 那么根据刚才的统计信息可以推出: 文档 d 的主要内容是跟“machine learning”相关的, 而单词 w 在大部分情况下(40%)也是与“machine learning”有关, 这说明, 单词 w 与文档 d 在主题空间中的距离是很接近的。基于这个想法, 我们分两步来估计单词 w 和文档 d 之间的关系。

首先, 我们先估计一个单词的不同实例在各个主题上的分布:

$$\varphi_w^{(z)} = \frac{n_w^{(z)} + \alpha}{\sum_{z=1}^K (n_w^{(z)} + \alpha)} \quad (0.7)$$

其中 $n_w^{(z)}$ 表示单词 w 被分派到主题 z 中的实例个数。平滑参数 α 可以避免 0

概率的出现。需要注意的是，在我们的 WMCM 模型中，虽然主题分派是针对双词来进行的，但为双词分派主题之后，单个词的主题分派也就完成了，所以公式 (3.7) 对于 WMCM 模型中的 BTM 来说同样适用。

然后，我们在主题空间下(维度为 K)，以余弦相似度来衡量单词 w 与文档 d 之间的相关度，即，单词 w 与文档 d 的联合概率：

$$P(d, w) = \frac{\theta_d * \varphi_w}{|\theta_d| * |\varphi_w|} \quad (0.8)$$

3.5 情感预测

给定一篇测试文档 \hat{d} ，他被分类为某种情感的概率为：

$$P(e | \hat{d}) \propto P(e) * P(d | e) \quad (0.9)$$

$P(e)$ 为情感类 e 出现的概率，根据最大似然估计可以得到：

$$P(e) = \frac{|D_e| + \beta}{|D| + E * \beta} \quad (0.10)$$

D_e 为情感类 e 中包含的训练文档的集合， D 为所有训练文档的集合，平滑参数 β 的作用是避免 0 概率值的出现。

然后，我们基于条件独立性假设计算先验概率 $P(\hat{d} | e)$ ，即：

$$P(\hat{d} | e) = \prod_{w \in \hat{d}} P(w | e) \quad (0.11)$$

在模型 RPWM 和 WMCM 中， $P(w | e)$ 直观上表示 w 与情感类 e 的相关程度，这个值与 w 与 e 的中文档 d 的相关程度有关。在之前的方法中， w 与 e 的相关程度可以通过 w 与 e 中的所有文档相关程度的和的平均值来计算，因为训练文档可能存在噪声，所以我在这里引入 3.3 节中估计的训练文档权重。对于 RPWM 模型而言：

$$P(w | e) \propto \sum_{d \in e} P_{RPWM}(d) * P(w | d) \propto \sum_{d \in D_e} P_{RPWM}(d) * P(w, d) \quad (0.11)$$

由于“文档集中度”是基于某一个类 e 的，所以 WMCM 模型中计算 $P(w|e)$ 的公式如下：

$$\begin{aligned} P(w|e) &= \sum_{d \in D_e} P(d|e) * P(w|d) \\ &\propto \sum_{d \in D_e} P(d|e) * P(w,d) \end{aligned} \quad (0.11)$$

其中，测试集中的词 w 与训练文档 d 的联合概率 $P(w,d)$ 可以通过公式(3.8) 来求得。

RPWM 模型和 WMCM 模型通过对训练文档进行加权，有效地减小了噪声文档对于系统性能的影响，进一步提升了分类器的分类准确率。

第 4 章 社交网络文本的情感分类模型

在这一节中，我们会介绍本文中提出的第 3 个模型—BtCNN。本章第一部分会列出 BtCNN 中用到的记号，然后在第二部分，我们重点阐述网络结构以及如何利用 BtCNN 来更好地学习文本特征，最后一个部分，我们会介绍如何训练 BtCNN 以及如何用 BtCNN 来预测文本的情感。

4.1 符号定义

为了方便介绍我们的模型，我们在这里定义本章中的常用记号。

在训练与测试的过程中，我们将文档集 D 划分为 D_{train} ， D_{val} 以及 D_{test} ，分别代表训练集，验证集与测试集。给定一篇文档 d 以及它的单词列表 $\{w_1, w_2, \dots, w_n\}$ ，为了增强相邻词在上下文的联系，我们将相邻词组成短语来表示一篇文档，这样可以得到对应的词语列表 P_d ，即 $P_d = \{(w_1, w_2), (w_2, w_3), \dots, (w_{n-1}, w_n)\}$ ，因此，文档 d 的长度 $N_d = |P_d|$ 。假设输入的词向量维度为 K ，输入的单词 $w_i (i = 1, \dots, n)$ 对应的词向量为 $x_i (i = 1, \dots, n)$ ，我们对每个短语包含词的词向量进行线性相加，得到表示短语的向量 p_i ，记两个词向量对应的系数为 a_1, a_2 。通过相邻词构成短语之后，我们可以得到 N_d 个短语及其向量表示，这些向量构成输入文档的矩阵表示 $d_{mat} \in R^{N_d * K}$ ，滤波器 f 在文档矩阵 d_{mat} 上进行卷积操作，滤波器高度为 $height_f$ ，宽度为 $width_f$ 。这里，我们借鉴了 Kim[52] 等人提出的 CNN 网络中的结构，将滤波器宽度 $width_f$ 设置为为文档矩阵的宽度(或短语向量的维度 K)，即认为每一个词向量都是最小的信息单元，不可能从词向量的部分元素中获取更细粒度的特征。卷积之后添加偏置向量 b ，然后再输入非线性层，非线性层的激活函数记为 h ，常见的 h 有 \tanh ， relu 和 sigmoid 。非线性层的输出就是滤波器 f 在文档矩阵中

提取的特征向量，记为 o_f 。在网络中可能会有多个滤波器，记滤波器的数目为 N_f 。由于每篇文档的长度不同，卷积之后得到的特征向量 o_f 的长度也不同。所以 o_f 还需要输入池化层(pooling layer)，多个滤波器产生的 o_f 通过池化层，输出的结果拼接在一起就构成了文档的特征向量 d_f 。表格 4-1 中列出了第 4 章中的常用符号。

表格 4-1 第 4 章常用记号

符号	符号定义
E	情感类别个数
D	文档集
D_{train}	训练文档集
D_{val}	验证文档集
D_{test}	测试文档集
d	文档 d
d_{mat}	文档 d 的矩阵表示
P_d	文档 d 短语列表
N_d	单词 d 中包含的短语个数
a_1, a_2	词向量线性叠加系数
K	词向量维度
f	卷积滤波器
$height_f, width_f$	滤波器 f 的高度与宽度
b	偏置向量
h	非线性激活函数
o_f	卷积特征向量
N_f	滤波器数目

d_f 文档特征向量

4.2 网络结构与文档特征学习

这一小节，我们将会介绍我们提出的 BtCNN 的网络结构以及如果利用这个网络结构去学习文本的特征向量。

在卷积之前，我们要对词向量进行处理，如 4.1 节中所说，我们将文档 d 中相邻的两个词的词向量进行线性叠加，构成短语(phrase)的向量表示形式：

$$p_i = a * x_i + b * x_{i+1} (i = 1, \dots, n-1) \quad (4.1)$$

其中 x_i 是文档 d 中的第 i 个词向量， p_i 表示文档 d 中的第 i 个短语的向量表示。这样做的好处是：增强相邻词之间的语义联系，同时在没有使用外部工具的情况下引入句子顺序信息。把这两个词组合起来获取更多的局部特征，虽然卷积滤波器也有提取上下文特征的性质，但是相邻的两个词更有可能产生语义上的联系，所以 BtCNN 的输入不是词向量，而是词向量线性组合之后得到的短语向量。鉴于 Kim[52]等人的工作在文本分类上取得的良好效果，我们模仿了他们的网络设计方法，也就是将滤波器 f 的宽度 $width_f$ 设置为输入的词向量的维度 K 。这样的设置方法与卷积神经网络在图像分类中的设置略有不同，因为图像矩阵中，每一行是若干个可分的像素点，单个像素携带有图像特征，而文档矩阵中，每一行代表一个词(或短语)，是不可再分的特征单元，所以捕捉词(短语)向量的局部特征是没有意义的，这是我们采用这样的设置方法的出发点。经过设置之后，BtCNN 对文档 d 的矩阵表示 d_{mat} 进行一维卷积，并通过激活函数来向滤波器提取的特征中引入非线性因素：

$$c_i = h(f \cdot d_{mat}(i:i+f_h-1) + b) \quad (4.2)$$

其中， $d_{mat}(i:i+f_h-1)$ 表示文档矩阵 d_{mat} 的第 i 行到第 $i+f_h-1$ 行。 b 为偏置向量， c_i 表示滤波器 f 从文档矩阵中提取的第 i 个特征值。Kim[52]等人在他们的卷积神经网络中选择 ReLU[56]作为非线性函数，将小于 0 的特征值全部截断为 0，它的函数形式如下：

$$h_{\text{ReLU}}(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (4.3)$$

这样做可以有效地降低网络前向传播时的计算复杂度，同时提高模型的泛化能力[57]。但是，将特征值设为 0 也阻碍梯度的反向传播。所以，我们在 BtCNN 中采用了随机弱化的 ReLU(Randomized Leaky Rectified Linear Unit)。一方面，弱化的 ReLU 不会产生 0，也就不会增加优化目标函数的难度，另外，引入的随机性可以降低过拟合的可能性。RReLU 的函数表示如下：

$$h_{\text{RReLU}}(x) = \begin{cases} x, & \text{if } x \geq 0 \\ kx, & \text{otherwise} \end{cases} \quad (4.4)$$

$$k \approx \text{Uniform}(l, u), l < u \text{ and } l, u \in [0, 1] \quad (4.5)$$

k 的取值服从均匀分布 $\text{Uniform}(l, u)$ 。卷积的结果加入非线性元素就得到了滤波器从当前文本中提取的特征向量 c_f ，它的维度为 $N_d - \text{height}_f + 1$ 。可以看到，向量的长度会随着文档长度(短语个数)的变化而变化，为了使网络能够处理变长的句子，我们采用了 Collobert 等人[58]提出的最大值池化(one-max-pooling)，也就是取特征向量 c_f 中的最大值来作为滤波器从文本中提取的最终特征：

$$c_{f-\text{pool}} = \max\{c_1, c_2, \dots, c_{N_d - \text{height}_f + 1}\} \quad (4.6)$$

$c_{f-\text{pool}}$ 为池化层的输出，是一个标量。我们将 N_f 个滤波器对应的特征拼接起来，得到文档的向量表示 d_f 。 d_f 被输入到 softmax 层中，进行多标签情感分类。

Softmax 层与输出层之间是全连接的，仍然存在不少的参数(连接两个层的边的权重)，在训练数据有限的情况下容易过拟合，所以我们采用了 Srivastava[59]等人提出的节点失效(Dropout)的方法，在池化层与 softmax 层之间加入了失效层(Dropout Layer)。这个方法是通过在前向传播时随机将节点权重暂时性的设置为 0 来实现网络的正则化。Dropout 层对特征向量的操作如下：

$$d_{f-\text{drop}} = d_f \cdot r \quad (4.7)$$

$$r_i \approx \text{binomial}(p), (i = 1, 2, \dots, N_d - \text{height}_f + 1) \quad (4.8)$$

d_{f-drop} 为 Dropout 层输出的文档向量， r 是长度与 d_f 相同的 0-1 向量，他的每一个元素都是从二项分布 $binomial(p)$ 中生成， p 表示单次伯努利试验的成功概率 (即，取值为 1 的概率)。随机让节点失效，连接他的边(参数)也随之暂时消失，从而降低了训练时参数过拟合的风险。需要注意的是，节点失效只是暂时性的，下一轮前向传播时，节点的权重有可能恢复，同时，我们只在训练 BtCNN 时加入 Dropout Layer，测试时，顶层文本特征不会发生改变。BtCNN 的网络架构如图 4-1。为了更形象的介绍我们的 BtCNN，我们在图 4-2 中画出了文本特征的提取以及类标的预测过程。

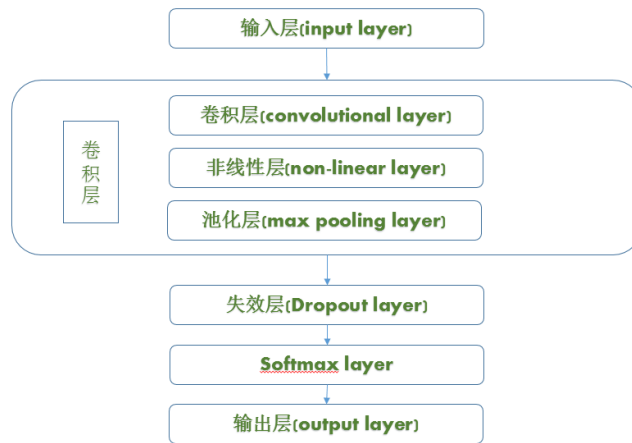


图 4-1 BtCNN 网络架构

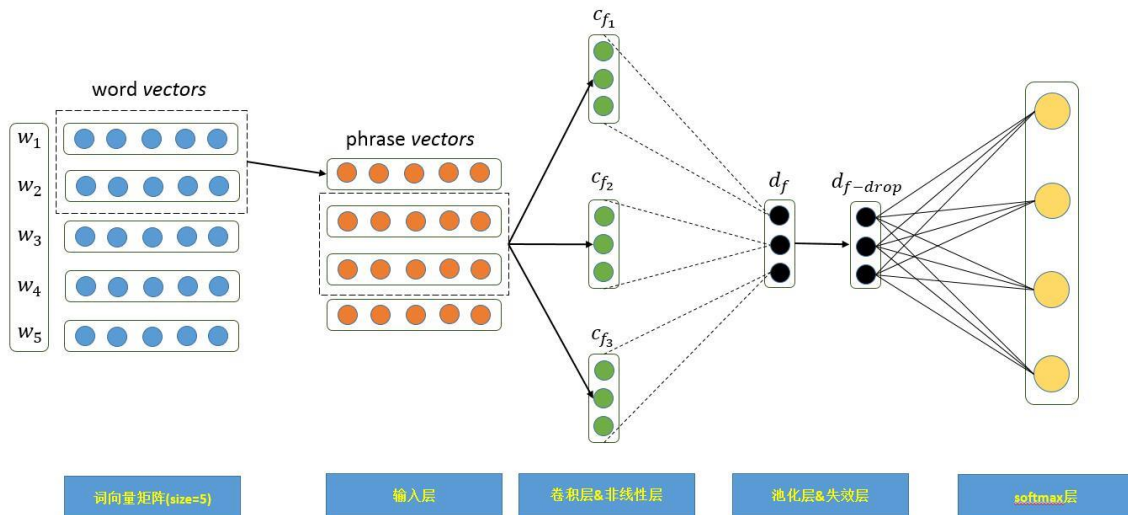


图 4-2 BtCNN 系统流程(假设输入文档中的单词数 N_d 为 5，词向量的长度 K 为 5，使用 3 个滤波器进行卷积($N_f = 3$)，滤波器的高度 $height_f$ 均为 2，预定义的情感类别个数为 4)

4.3 网络训练与情感预测

在训练的过程中，对于输入的文档 d ，softmax layer 会输出文档被分到各个类的概率，即 $P(e_i | d)(i=1, \dots, E)$ ，我们采用交叉熵来作为 BtCNN 网络优化的目标函数。计算公式如下：

$$J(\theta) = -\frac{1}{|D_{train}|} \sum_{d \in D_{train}} \sum_{i=1}^E t_d^i * p(e_i | d) \quad (4.9)$$

出于简洁的目的，我们将训练过程中的参数统一表示为 θ ， $J(\theta)$ 为 BtCNN 网络的训练损失函数。 $t_d \in R^E$ 是一个独热(one-hot)向量，1 的位置就是文档 d 对应的真实类标。 $p(e_i | d)$ 是通过 BtCNN 网络计算出的值，表示文档 d 被分类为第 i 个类的概率。训练网络的目标就是要寻找合适的参数 θ 使得目标函数 $J(\theta)$ 的值最小。优化过程中，我们将训练文档集平均分成若干个文档子集，然后依次在每一个文档子集上计算损失函数的值，通过梯度下降更新参数 θ 。这些文档子集被称为 mini-batch。为了使得更新后的参数 θ 更接近最优解，我们对参数的每一个维度使用不同的学习速率 (learning rate)，论文中使用的方法是 nesterov momentum[60][61]。momentum 的优点在于：如果某一个维度的参数一直朝同一个方向变化，那么就增大这个维度的动态学习速率；反之，则减小学习速率；这个方法能够加快梯度下降的速度，当参数在最优点两侧变化时，相应的动态学习速率会减少，从而使得 θ 更加靠近最优解。第 t 次迭代的参数更新过程如下：

$$\Delta\theta_t = mu * \Delta\theta_{t-1} - \varepsilon * \nabla J(\theta_{t-1}) \quad (4.10)$$

$$\theta_t = \theta_{t-1} + \Delta\theta_t \quad (4.11)$$

其中常数 ε 为学习速率， mu 也是常数，它的大小控制参数变化值增加的快慢， $\Delta\theta_t$ 为第 t 次迭代的参数更新值(在正常的梯度下降中， $\Delta\theta_t = -\varepsilon * \nabla J(\theta_{t-1})$ ，如果 $\Delta\theta_{t-1}$ 与梯度的负方向一致，那么参数的变化值就会加快，反之，则会减少)，Bengio 等人[61]为了进一步加快模型收敛的速度，先超前估计了参数经过本次迭代之后出现在损失平面上的位置，然后根据参数在这个超前位置处的梯度，更新当前的

参数值，所以公式(4.10)就变成了：

$$\Delta\theta_t = \mu * \Delta\theta_{t-1} - \varepsilon * \nabla J(\theta_{t-1} + \mu * \Delta\theta_{t-1}) \quad (4.12)$$

公式(4.12)(4.11)就构成了我们的 BtCNN 网络中使用的 nesterov momentum 优化方法。另外，在测试的过程中，卷积层的非线性激活函数对于负特征值的处理与公式(4.4)有所不同。

$$h_{rrelu}(x_{test}) = \begin{cases} x_{test}, & \text{if } x_{test} \geq 0 \\ 0.5, & \text{otherwise} \end{cases} \quad (4.13)$$

同样与训练网络时不同的是，失效层(Dropout layer)不会作用在测试集文本的特征上，失效层的作用是避免参数过拟合而不是学习更有用的特征，所以失效层对于测试集来说没有意义。至此，我们的 BtCNN 网络介绍完毕。

第 5 章 在线新闻文本数据集实验

在这一节中，我们会详细介绍实验中所用的在线新闻文本数据集，实验参数设定以及与基准算法的对比过程。

5.1 数据集

为了验证我们提出的 RPWM 和 WMCM 模型的有效性和普适性，我们使用了以下两个数据集：

- 1) *SemEval*. 这是 SemEval-2007 竞赛的官方数据集，其中包含了 1250 条新闻标题，这些新闻标题来自于 Google News, CNN 以及其他门户网站。任务中预定义了 6 种情感标签(“anger”, “disgust”, “fear”, “joy”, “sad”, “surprise”), 每篇文档在 6 种情感上的投票都经过人为的处理。在去掉了 4 篇投票数为 0 的标题之后，我们使用其中的前 246 篇文本作为训练集，后 1000 篇文档作为测试集。
- 2) *SinaNews*. 这是一个包含了 4570 篇新闻的文档集。新闻来源于新浪新闻 [11]。这个数据集中的预定义情感标签有 8 个(“感动”, “同情”, “无聊”, “愤怒”, “开心”, “难过”, “惊讶” 和 “温暖”), 每篇文本在各个情感标签上的投票均来自于新闻读者。经过预处理之后，整个文档集中包含 1975153 个词和 325434 个用户情感投票。为了避免新闻背景的相似性对实验结果的干扰,我们使用 1 月至 2 月的新闻作为训练集(共计 2342 篇), 3 月到 4 月的新闻作为测试集(共计 2228 篇)。

关于数据集的更多统计信息已经列举在表格(5-1)中。

表格 5-1 数据集的统计信息

数据集	情感标签	文档数目	投票数目
<i>SemEval</i>	anger	87	12042
	disgust	42	7634
	fear	194	20306
	joy	441	23613
	sad	265	24039
	surprise	217	21495
<i>SinaNews</i>	感动	749	41798
	同情	225	23230
	无聊	273	21995
	愤怒	2048	138167
	高兴	715	43712
	难过	355	37162
	惊喜	167	11386
	温馨	38	7986

5.2 实验设计

这个部分中，我们实现了一些的基准算法，来与我们提出了 RPWM 和 WMCM 模型进行对比：

- 1) *SWAT: SemEval-07 task 14*[5]中性能最优秀的系统之一。它使用了 unigram 的语言模型来标注标题中含有情感色彩的部分，同时也计算了每一个词在各个情感类别上的得分[5][6]。
- 2) *Emotion Term Method(ET)*: ET 直接对词和情感标签之间的关系进行建模[7]。ET 在朴素贝叶斯条件独立性假设的基础上，认为文档中的词是从情感标签中经过两步采样之后生成。ET 与朴素贝叶斯的区别在于，ET 在计算先验概率 $P(e)$ 和 $P(w|e)$ 时，考虑了情感投票的信息。

- 3) *Emotion Topic Model(ETM)*: ETM 在 ET 的基础上, 引入了一层主题层, 通过 LDA 学习主题分布, 将词与情感标签关联起来[8]。ETM 的参数遵循论文[8]中的原始设定。
- 4) *Multi-label supervised topic model(MSTM) and Sentiment latent topic model(SLTM)*[12]: MSTM 和 SLTM 都是包含了两重生成过程的模型。MSTM 首先从语料中学习主题分布, 然后通过学到的主题分布去对情感标签进行抽样; SLTM 则相反, 它从情感投票分布中学习主题分布, 然后在生成语料中的所有词。

表格(5-2)中列出了与主题模型相关的参数。需要注意的是, RPWM 模型和 WMCM 模型中的主题参数(即 LDA 和 BTM 的主题参数)设定略有不同, 它们在 *SemEval* 和 *SinaNews* 上也不同。Gibbs Sampling 的迭代次数被设置为 1000。另外, 我们将所有实验中的平滑参数 β 固定为 0.01, 因为实验表明, β 的取值对于结果并不会造成很大的影响。在 WMCM 中, 还需要设定滑窗 λ 的大小, 不同数据集文档的平均长度不同, 相应的 λ 值也不一样。ETM, MSTM, SLTM 等基准算法的设定都与原始论文中的设定相同。

表格 5-2 模型参数设定

模型	参数	<i>SemEval</i>	<i>SinaNews</i>
RPWM(LDA)	α	0.05	50/K
	β	0.01	0.01
WMCM(BTM)	α	50/K	50/K
	β	0.01	0.01
	λ	2	15

与之前提到的一样, 新闻文本情感分析的目的在于通过预测未知文本的情感(即: $P(e|\hat{d})$)来挖掘在线用户的情感偏好。 $P(e|\hat{d})$ 的值越大, 说明文档越有可能对用户产生相应的情感, 引起读者共鸣。为了验证模型的有效性, 我们预测了这个条件分布, 并与真实的情感投票分布进行比较。更详细地说, 我们将条件概

率最大的情感标签作为输出标签，而通过真实的用户投票分布，我们可以确定文档最有可能表达的几种情感标签。如果系统输出标签在这几种情感标签中，那么就认为这次预测是正确的，否则，就是错误的。在文章中，我们用 $Pred_{d@n}$ 来表示系统对于单个测试文档 d 的预测情况：

$$Pred_{d@n} = \begin{cases} 1, e_p \in Top_{d@n} \\ 0, otherwise \end{cases} \quad (5.1)$$

其中， e_p 是系统输出的预测标签， $Top_{d@n}$ 是根据真实情感投票分布选出的得分最高的标签集合， n 为集合中标签的数目。在评价整体预测的性能时，我们使用微平均的 $F1$ 指标。 $F1$ 指标中，准确率跟召回率所占权重相同，微平均则是计算整体 $F1$ 指标时，不考虑类的边界，只考虑样本是否预测正确[44]，根据表格(5-1)中的统计信息可以知道，数据集中的每个类别在数目上是不平衡的，所以在这里使用宏平均的 $F1$ 指标并不能客观地描述分类器的性能。 $F1$ 指标值的计算基于 $Pred_{d@n}$ 。另外，我们将 n 的值设为 0，意味着只有当预测的标签正好就是真实投票分布中得分最高的标签时，预测才是正确的。具体计算公式如下：

$$F1_{micro} = \frac{\sum_{d \in D_{test}} Pred_{d@1}}{|D_{test}|} \quad (4.2)$$

$F1_{micro}$ 的值越大，代表系统的性能的分类预测性能就越好。

5.3 与基准算法进行对比

除了词层空间的模型 SWAT 和 ET，我们还跟 ETM, MSTM, SLTM 等引入了主题层的模型进行对比。因为在学习主题分布的过程中，主题的数目也是一个重要的影响因素，所以，我们依照[7][8]工作中的做法，保持其他参数不变的情况下，将主题数由 2 变化到 30，同时观察系统的预测性能。实验结果表明，我们提出的 RPWM 模型在平均指标上取得了最好的效果，而 WMCM 模型的预测性能最稳定，同时在 $F1$ 值上与 RPWM 模型的相应值十分接近。图(5-1)显示了基准算法与我们提出的 RPWM 和 WMCM 模型在不同主题数下的性能变化。

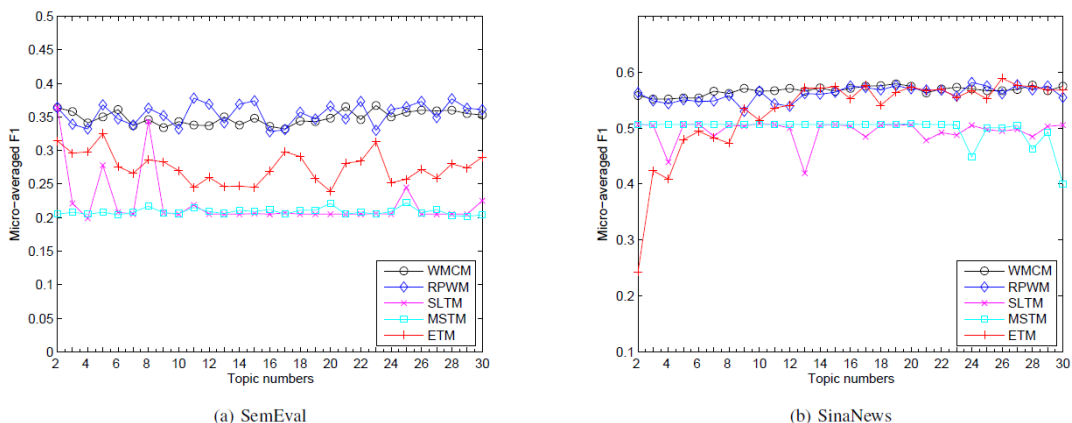


图 5-1 RPWM 和 WMCM 模型在不同主题数下的性能变化

WMCM 模型在 *SemEval* 数据集上的平均预测准确率相较于 SWAT, ET, ETM, MSTM 和 SLTM 而言提高了 11.14%, 12.57%, 26.91%, 67.13%, 57.93%, 在 *SinaNews* 数据集上则是分别提升了 12.05%, 32.21%, 7.24%, 13.85%, 14.61%。SWAT 系统在这两个数据集上的 *F1* 指标值分别为 31.40%和 50.63%，ET 的则是 31.00%和 42.91%。RPWM 模型在平均预测准确率上取得了最高值，它在 *SemEval* 上的 *F1* 指标值为 36.47%，在 *SinaNews* 上的 *F1* 指标值则是 56.12%。从图(4-1)中还可以观察到，RPWM 模型的稳定性不如 WMCM 模型，也就是说，WMCM 模型的性能受主题数变化的影响更小。

为了验证 WMCM 模型的稳定性，我们对它和 RPWM 以及其他基准算法进行 2 组统计实验。第一组实验是基于方差来评价算法的稳定性；第二组实验则是基于均值。统计显著的阈值我们设定为 0.05。

首先，我们进行 *F-test*，我们对方差进行分析，以此来评估同方差性的假设。由于 SWAT 和 ET 没有引入主题模型，所以他们的性能不会随主题数目的变化而变化。因此，*F-test* 中没有包含这两个基准算法。表格(5-3)列出了 WMCM 模型与 RPWM 以及其他基准算法在 *F-test* 上的 *p* 值。结果表明他们在方差上的差异是统计显著的(所有的 *p* 值都小于 0.05)，也就是说，在主题数不同的情况下，WMCM 模型的平均预测性能比 RPWM 以及其他基准算法都要稳定。

表格 5-3 WMCM 模型与其他模型在 *F-test* 上的 *p* 值

Models	<i>SemEval</i>	<i>SinaNews</i>
RPWM	0.0105	0.0023
SLTM	8.4E-11	5.2E-7

<i>MSTM</i>	0.0001	3.0E-8
<i>ETM</i>	3.2E-5	0.0000

接着，我们将对 WMCM 模型和其他的模型进行 *t-test*，*t-test* 可以对平均性能的差值进行评估。表格(5-4)列出了每组试验的 *p* 值。结果显示 WMCM 模型的预测性能是远远超过 SLTM, MSTM, SWAT, ETM 和 ET 的，因为相对应的 *t-test* 的 *p* 值都小于 0.05(平均性能差距是统计显著的)。即使相比于平均性能最好的模型，WMCM 模型在 *SemEval* 上的预测准确率也是能够十分接近 RPWM 的(*p* 值大于 0.05，统计不显著)。此外，WMCM 模型在 *SinaNews* 数据集上的性能是优于 RPWM 模型的

表格 5-4 WMCM 模型与其他模型在 *t-test* 上的 *p* 值

Models	<i>SemEval</i>	<i>SinaNews</i>
<i>RPWM</i>	0.0928	0.0363
<i>SLTM</i>	2.3E-17	2.7E-19
<i>MSTM</i>	1.8E-43	7.5E-17
<i>ETM</i>	8.0E-19	0.0080
<i>ET</i>	1.7E-18	2.8E-37
<i>SWAT</i>	2.9E-17	2.1E-27

第 6 章 社交网络文本情感分类实验

在这一章，我们会介绍社交网络文本数据集以及 BtCNN 模型在不同数据集上的性能优势，同时我们也会分析模型性能好与不好的原因

6.1 数据集

在社交网络文本情感分类部分，我们使用了以下两个数据集：

- (1) *tweet*: *tweet* 数据集包含了 4242 个来自 *twitter* 文本。每一篇文本分别对应一个正向情感的得分以及负面情感的得分。为了方便分类，我们在预处理时通过正负面的情感得分来为文本设置类标。在这个数据集中，我们定义了 3 种类标：*positive*，*negative* 和 *neutral*。当文档的正向情感得分大于负面情感得分时，我们认为这是这篇 *tweet* 是 *positive* 的；当正负面得分相同时，我们认为文档的情感为 *neutral*，如果以上两种条件都不满足，那么文本传递的情感信息是负面的，即 *negative*。通过预处理之后，我们得到 1340 篇 *positive* 文档，949 篇 *negative* 文档以及 1953 篇 *neutral* 文档，数据集中句子的最大长度为 34，平均长度为 16.8，一共出现了 15704 个不同的词。在实验中，为了训练一个拟合样本点能力较好的神经网络，我们选择每个类中的前 60% 的样本作为训练集，后 20% 的样本作为测试集，其余的作为验证集。
- (2) *isear*: *isear* 数据集是 ISEAR(International Survey on Emotion Antecedents and Reactions) 项目中构造的用于情感分类的数据集。数据集中包含了 7588 个句子，全部是通过问卷的形式来生成的，参与问卷的一共有 1096 个来自不同背景的人，他们对问卷中的内容进行理解，然后根据自己的感受选择相应的情感标签，组织方提供的情感标签有 7 类：*anger*, *disgust*, *fear*, *joy*, *sadness*, *shame* 和 *guilt*。经过处理以及滤除特殊符号后，数据集中最长的句子长度为 59，平均长度为 21.6。与 *tweet* 数据集中类似，我们从每个类的样本中选择前 60% 作为训练集，后 20% 作为测试集，其余

作为网络训练时的验证集。

我们将相邻的两个词作为短语是合理，那不相邻的两个词有没有可能组成短语呢？为了验证我们的猜想，我们放松了构造短语的条件，我们设定窗口大小 win ，在当前的词窗口范围内的词，都是可能与之构成短语的词。BtCNN 网络中的短语就是窗口大小 win 取 2 时，从文档中生成的短语。当 win 的大小为 0 或 1 时，抽取的短语就是文本中的单词。数据集的详细统计信息见表格 6-1。需要注意的是，当 win 取 0 或 1 时，最大长度，平均长度与词汇表大小均是基于文档中的单词来计算，其他情况下则是基于文档中包含的短语来计算这些值。由于使用了词向量(<https://code.google.com/archive/p/word2vec/>)，所以，词向量数表示能够在预训练的词向量中找到对应项的单词数。

表格 6-1 社交网络文本数据集统计信息

数据集	窗口大小	单词数	词向量数	词汇表大小	最大长度	平均长度	文档数目	训练	测试
isear	-			8980	59	21.57			
	2			55506	58	20.57			
	3	8980	8118	113099	115	40.14	7588	4550	1523
	4			165299	171	58.71			
	5			210810	229	76.30			
tweet	-			-	34	16.86			
	2			47346	33	15.86			
	3	15704	8906	92457	65	30.71	4242	2544	851
	4			133177	96	44.58			
	5			169383	126	57.55			

6.2 实验设计

在这个部分，我们实现了一些基于传统机器学习与神经网络的基准算法，算法描述如下：

- (1) SVM-words: 通过 unigram 语言模型提取文本中的特征，然后将特征输入到线性 SVM 分类器中进行情感预测。
- (2) SVM-phrase: 同 1 类似，不同之处在于模型基于短语(即，相邻的两个词)

来提取文本特征

- (3) **SVM-word2vec**: 使用 Mikolov 等人训练的词向量[63]来表示文章中的每一个词，然后将文档中包含的所有词的词向量相加，再取平均值就构成了文档的特征向量。文档特征向量会被输入到线性 SVM 分类器中进行分类。注，对于在预训练词向量中无法找到对应项的词，我们随意初始化一个相同维度的向量作为他的特征表达。
- (4) **SVM-phrase2vec**: 与 BtCNN 构造短语的方式相同，我们将相邻的两个词构成短语，而短语在向量空间中的表示通过将两个词的词向量作线性相加得到。之后文档特征向量被输入到线性 SVM 中进行情感分类。
- (1)(2)(3)(4)所使用的线性 SVM 的相关参数见表格 6-2。
- (5) **CNN1**: Kim[52]等人在 2014 年提出的一个卷积神经网络架构，网络包含一个卷积层，一个池化层以及一个全连接层，卷积层采用多个高度不同的滤波器来提取不同粒度的特征。网络训练过程中，他们采用了不需要设置学习速率的 adadelta[62]来寻找最优的参数。这个网络是目前文本分类领域性能最好的几个工作之一。
- (6) **CNN2**: 与(5)中使用的网络相同，只是输入换成 phrase vectors。
- (7) **Fully-Connected-NN1**: 我们自己实现的一个全连接神经网络。整个网络一共包含 3 层：输入层，隐藏层以及 softmax 层。网络的输入为预训练的词向量，优化目标函数主要基于随机梯度下降的方法
- (8) **Fully-Connected-NN2**: 基本设置与(5)中相同，不过输入变为短语向量。短语向量通过(3)中提到的线性相加的方法计算。

表格 6-2 svm 参数

核函数	损失函数	训练方式	惩罚项	松弛变量系数
线性核	Hinge loss	One-vs-one	L2 norm	1.0

另外，为了更好的进行对照试验，我们将神经网络中用到的相关参数也列举在了表格 6-3 中

表格 6-3 神经网络相关参数

模型	激活函数	滤波器宽度	滤波器个数	学习速率	Batch_size	失效概率	隐藏层节点数	最大迭代次数	(a_1, a_2)
CNN	ReLU	3,4,5	3	-	75	0.5	-	50	-
FC-NN	Tanh	-	3	0.01	75	-	300	50	-
BtCNN	RReLU	3,4,5	3	0.1	75	0.5	-	50	(1, 1)

注：神经网络的输入均为 Mikolov[63]等人预训练的词向量，向量维度 K 均为 300。为了方便起见，构造短语向量时的叠加系数被设置为(1,1)，即两个词向量的和即为对应的短语向量。

6.3 与基准算法对比

在 6.1 已经介绍过，训练数据集中每个类的样本数比较平均，所以我们在评测系统性能时，使用了微平均(micro-avg)这一衡量标准。实验结果显示，我们提出的 BtCNN 模型在两个数据上均取得了最好的性能。对照试验结果如表格 6-4。

表格 6-4 对照试验性能比较

数据集	模型	micro-avg
isear	SVM-words	53.3%
	SVM-phrase	44.78%
	SVM-word2vec	53.11%
	SVM-phrase2vec	51.54%
	CNN1	61.49%
	CNN2	61.78%
	Fully-Connected-NN1	50.04%
	Fully-Connected-NN2	48.72%
	BtCNN	62.04%
tweet	SVM-words	58.87%
	SVM-phrase	49.11%
	SVM-word2vec	58.63%
	SVM-phrase2vec	57.11%
	CNN1	60.51%

CNN2	61.10%
Fully-Connected-NN1	52.07%
Fully-Connected-NN2	50.02%
BtCNN	62.16%

从表格中可以看出，我们的 BtCNN 都表现出了最好的性能，根据表格 6-1 中的统计信息，我们也可以了解到 BtCNN 所耗费的资源相对于另外两种神经网络模型来说是最少的(模型文本长度最短)。为了合理地解释性能的提升，我们从以下几个方面来进行分析：

- a. 是否包含神经网络？从表格 6-4 的结果中，我们可以看到，BtCNN 与 CNN 相对于 svm 的方法都有明显的提升，但是全连接的神经网络性能却很糟糕，所以神经网络的结构并不是性能提升的一个关键因素。
- b. 是否使用预训练的词向量？预训练的词向量能够将单个词映射到低维度的向量空间，可以在一定程度上解决一词多义和同义词的问题。但是 svm 方法的实验结果，使用词做特征并不会比使用词向量做特征的系统性能差。
- c. 是否使用卷积滤波器？卷积滤波器将滤波器宽度之内的词进行卷积，然后加入非线性元素，这一方法能够有效地学习文本的上下文特征。实验结果也表明，使用了卷积滤波器的 CNN 与 BtCNN 在性能上都有很大的优势。
- d. 是否使用 phrase vector？在试验中，我们发现了一个有趣的现象。对比 CNN1 与 CNN2，我们发现使用 phrase vector 作为输入的 CNN2 性能有了微小的提升，而比较 SVM-word2vec 和 SVM-phrase2vec 的时候，出现了相反的结果，使用了 phrase vector 的后者反而出现了性能下降，经过分析之后，我们得出的结论是，卷积滤波器造成了这个现象的产生。对于基于 SVM 的方法，上下文的信息并没有关联在一起，构造短语之后，文本中的特征反而减少了，使得性能有所下降；而对于使用了卷积的 CNN，因为卷积操作将上下文联系在一起了，反而学到了更多深层的语义特征，所以准确率有了一定的提高

综上所述，我们的 BtCNN 对于分类准确率的提高来自于：(1)使用了卷积滤波器；(2)将输入变成了 phrase vector，使得一些深层的特征能够在卷积过程中被挖掘出来；(3)简单的网络结构，避免的参数超调；(4)使用了 nesterov momentum，使得迭代停止时的参数更接近最优解。

第7章 总结与展望

情感分析对于在线服务提供商来说是非常有用的。分析用户的情感能够帮助理解用户的偏好，对事物的看法以及情感倾向程度，从而使得服务商能够推送更相关的产品，提供更个性化的服务。多标签分类是一种将文档与情感关联起来的基本方法。然而，传统的算法[45]大多都是将每一个训练样本的权重视为相同，因此，这些模型在训练的过程中很有可能会引入噪声。不同于这些方法，我们在进行在线新闻的情感分析时提出的 RPWM 和 WMCM 模型加入了对训练样本质量的估计，从数据集的统计信息科学的对每一篇训练文档进行加权。

对于没有情感投票的文本，我们提出了 BtCNN 模型来更好地学习文本特征，简单的结构不仅避免了参数过拟合，也降低了训练时间；卷积滤波器的使用有效的提取了上下文的特征。

在未来，我们将会沿着以下的方向继续我们的工作：

- 1) 我们会继续分析超参数对于主题提取过程的影响，同时去探索一些有效的方法以实现自动设置超参数。
- 2) 这篇文章中，我们在统计信息的基础上，通过“情感熵”和“情感集中度”来估计了文档的权重，这些权重可以通过其他有监督的方法来学到，我们会继续实验以找到可行的方法。
- 3) 我们会进一步优化卷积神经网络，目前最大的问题仍然是参数过多，同一组参数的泛化能力不强，对于不同的数据集不能产生很好的性能，下一步的目的是降低网络对于超参数的依赖。
- 4) 除了优化模型本身，我们还计划将他们应用到其他领域中，例如：股票预测，电影或者音乐推荐等。

参考文献

- [1] 赵妍妍, 秦兵, 刘挺. 文本情感分析. 软件学报, 21(8), 2010, 1834-1848
- [2] B. Liu, Sentiment analysis and opinion mining, Synthesis lectures on human language technologies, 5(1), 2012, 1~167
- [3] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL Conference on Empirical Methods in Natural Language Processing, Philadelphia, ACL, 2002, 79~86
- [4] K. Kim, J. Lee, Sentiment visualization and classification via semi-supervised nonlinear dimensionality reduction, Pattern Recognition, 47(2), 2014, 758~768
- [5] C. Strapparava, R. Mihalcea, Semeval-2007 task 14: Affective text. In Proceedings of the 4th International Workshop on Semantic Evaluations, Prague, ACL, 2007, 70~74
- [6] P. Katz, M. Singleton, R. Wicentowski, Swat-mp: the semeval-2007 systems for task 5 and task 14. In Proceedings of the 4th international workshop on semantic evaluations, Prague, ACL, 2007, 308~313
- [7] S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, Y. Yu, Joint emotion-topic modeling for social affective text mining, In Proceedings of 9th IEEE International Conference on Data Mining, Miami, IEEE, 2009, 699~704
- [8] S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, Y. Yu, Mining social emotions from affective text, IEEE Transactions on Knowledge and Data Engineering, 24(9), 2012, 1658~1670
- [9] C. Quan, F. Ren, An exploration of features for recognizing word emotion, In Proceedings of the 23rd International Conference on Computational Linguistics, Stroudsburg, ACL, 2010, 922~930
- [10] D. M. Blei, A. Y. Ng, M. I Jordan, Latent dirichlet allocation, The Journal of machine Learning research, 3(1), 2003, 993~1022
- [11] Y. Rao, Q. Li, L. Wenyin, Q. Wu, X. Quan, Affective topic model for social emotion detection, Neural Networks, 58, 2014, 29~37
- [12] Y. Rao, Q. Li, L. Wenyin, Sentiment topic models for social emotion mining. Information Sciences, 266, 2014, 90~100
- [13] Y. Rao, J. Lei, L. Wenyin, Q. Li, M. Chen, Building emotional dictionary for sentiment analysis of online news, World Wide Web, 17(4), 2014, 723~742
- [14] L. Zhuang, F. Jing, X. Y. Zhu, Movie review mining and summarization. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management,

- Arlington, ACM, 2006, 43~50
- [15] M. Hu, B. Liu, Mining and summarizing customer reviews. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data mining, Seattle, ACM, 2004, 168~177
- [16] S. R. Das, M. Y., Chen, Yahoo! for Amazon: Sentiment extraction from small talk on the web. *Management Science*, 53(9), 2007, 1375~1388
- [17] J. Yi, T. Nasukawa, R. Bunescu, W. Niblack, Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques, In Proceedings of 3rd IEEE International Conference on Data Mining, Melbourne, IEEE, 2003, 427~434
- [18] F. Li, S. Wang, S. Liu, M. Zhang, Suit: A supervised user-item based topic model for sentiment analysis, In 28th AAAI Conference on Artificial Intelligence, Québec City, AAAI, 2014, 1636~1642
- [19] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, P. Li, User-level sentiment analysis incorporating social networks, In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data mining, San Diego, ACM, 2011, 1397~1405
- [20] X. Hu, L. Tang, J. Tang, H. Liu, Exploiting social relations for sentiment analysis in microblogging, In Proceedings of the 6th ACM International Conference on Web Search and Data Mining, Rome, ACM, 2013, 537~546
- [21] X. Li, H. Xie, L. Chen, J. Wang, X. Deng, News impact on stock price return via sentiment analysis, *Knowledge-Based Systems*, 69, 2014, 14~23
- [22] X. Li, H. Xie, Y. Song, S. Zhu, Q. Li, F. L. Wang, Does Summarization Help Stock Prediction? A News Impact Analysis, *IEEE Intelligent Systems*, 30(3), 2015, 26~34
- [23] A. Montejo-Ráez, M. C. Díaz-Galiano, F. Martínez-Santiago, L. A. Ureña-López, Crowd explicit sentiment analysis, *Knowledge-Based Systems*, 69, 2014, 134~139
- [24] S. Rill, D. Reinel, J. Scheidt, R. V. Zicari, Politwi: Early detection of emerging political topics on twitter and the impact on concept-level sentiment analysis, *Knowledge-Based Systems*, 69, 2014, 24~33
- [25] D. Bell, T. Koulouri, S. Lauria, R. D. Macredie, J. Sutton, Microblogging as a mechanism for human-robot interaction, *Knowledge-Based Systems*, 69, 2014, 64~77
- [26] N. Ghamrawi, A. McCallum, Collective multi-label classification, In Proceedings of the 14th ACM International Conference on Information and Knowledge Management, ACM, 2005, 195~200
- [27] M. L., Zhang, Z. H., Zhou, ML-KNN: A lazy learning approach to multi-label learning, *Pattern recognition*, 40(7), 2007, 2038~2048
- [28] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, H. Blockeel, Decision trees for hierarchical multi-label classification, *Machine Learning*, 73(2), 2008, 185~214

- [29] L. Tang, S. Rajan, V. K. Narayanan, Large scale multi-label classification via metalabeler, In Proceedings of the 18th International Conference on World wide web, Madrid, ACM, 2009, 211~220
- [30] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, *Machine learning*, 85(3), 2011, 333~359
- [31] K. Dembczyński, W. Waegeman, W. Cheng, E. Hüllermeier, On label dependence and loss minimization in multi-label classification, *Machine Learning*, 88(1-2), 2012, 5~45
- [32] E. Montañes, R. Senge, J. Barranquero, J. R. Quevedo, J. J. del Coz, E. Hüllermeier, Dependent binary relevance models for multi-label classification, *Pattern Recognition*, 47(3), 2014, 1494~1508
- [33] C. Hong, I. Batal, M. Hauskrecht, A generalized mixture framework for multi-label classification, In Proceedings of the SIAM International Conference on Data Mining, Vancouver, NIH Public Access, 2015, 712~720
- [34] E. Gibaja, S. Ventura, A tutorial on multilabel learning, *ACM Computing Surveys (CSUR)*, 47(3), 2015, 52
- [35] M. L., Zhang, Z.H. Zhou, A review on multi-label learning algorithms, *IEEE Transactions on Knowledge and Data Engineering*, 26(8), 2014, 1819~1837
- [36] X. Cheng, X. Yan, Y. Lan, J. Guo, Btm: Topic modeling over short texts, *IEEE Transactions on Knowledge and Data Engineering*, 26(12), 2014, 2928~2941
- [37] J. Garai, Entropy is a Mathematical Formula, arXiv preprint physics/0301048, 2003
- [38] P. Giudici, *Applied data mining: statistical methods for business and industry*, John Wiley & Sons, 2005
- [39] K. H. Y. Lin, H. H. Chen, Ranking reader emotions using pairwise loss minimization and emotional distribution regression, In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Honolulu, ACL, 2008, 136~144
- [40] C. Lin, Y. He, Joint sentiment/topic model for sentiment analysis, In Proceedings of the 18th ACM Conference on Information and Knowledge Management, Hong Kong, ACM, 2009, 375~384
- [41] S. Li, L. Huang, R. Wang, G. Zhou, Sentence-level Emotion Classification with Label and Context Dependence, In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, Beijing, ACL, 2015, 1045~1053
- [42] T. Hofmann, Probabilistic latent semantic indexing, In Proceedings of the 22nd annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Berkeley, ACM, 1999, 50~57
- [43] T. L. Griffiths, M. Steyvers, Finding scientific topics, *Proceedings of the National Academy of Sciences*, 101(suppl 1), 2004, 5228~5235

- [44] C. D. Manning, R. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge, Cambridge university press
- [45] P. N. Tan, M. Steinbach, V. Kumar, Introduction to Data mining, Pearson Addison Wesley, Boston, 2006
- [46] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, R. Passonneau, Sentiment analysis of twitter data, In Proceedings of the workshop on languages in social media, Portland, ACL, 2011, 30~38
- [47] L. Jiang, M. Yu, M. Zhou, X. Liu, T. Zhao, Target-dependent twitter sentiment classification, In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, ACL, 2011, 151~160
- [48] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, K. Xu, Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification, In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, ACL, 2014, 49~54
- [49] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, B. Qin, Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification, In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics , Baltimore, ACL, 2014, 1555~1565
- [50] D. Tang, F. Wei, B. Qin, M. Zhou, T. Liu, (2014, August), Building Large-Scale Twitter-Specific Sentiment Lexicon: A Representation Learning Approach, In Proceedings of 25th International Conference on Computational Linguistics, Dublin, 2014, 172~182
- [51] D. Tang, B. Qin, X. Feng, T. Liu, Target-Dependent Sentiment Classification with Long Short Term Memory, arXiv preprint arXiv:1512.01100, 2015
- [52] Y. Kim, Convolutional neural networks for sentence classification, arXiv preprint arXiv:1408.5882, 2014
- [53] D. T., Vo, Y. Zhang, Target-dependent twitter sentiment classification with rich automatic features, In Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Morgan Kaufmann, 2015, 1347~1353
- [54] D. Tang, B. Qin, F. Wei, L. Dong, T. Liu, M. Zhou, A joint segmentation and classification framework for sentence level sentiment classification, IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23(11), 2015, 1750~1761
- [55] C. N., dos Santos, M. Gatti, Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts, In Proceedings of 25th International Conference on Computational Linguistics, 2014, Baltimore, ACL, 2014, 69~78
- [56] V. Nair, G. E., Hinton, Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning, Haifa, ACM, 2010, 807~814
- [57] X. Glorot, A. Bordes, Y. Bengio, (2011). Deep sparse rectifier neural networks. In International Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, JMLR,

- 2011, 315~323
- [58] R. Collobert, J. Weston, A unified architecture for natural language processing: Deep neural networks with multitask learning, In Proceedings of the 25th International Conference on Machine learning, Helsinki, ACM, 2008, 160~167
- [59] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, The Journal of Machine Learning Research, 15(1), 2014, 1929~1958
- [60] D. E., Rumelhart, G. E., Hinton, R. J., Williams, Learning representations by back-propagating errors, Nature, 323, 1986, 533~536
- [61] Y. Bengio, N. Boulanger-Lewandowski, R. Pascanu, Advances in optimizing recurrent networks, In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, IEEE, 2013, 8624-8628
- [62] M. D., Zeiler, ADADELTA: an adaptive learning rate method, arXiv preprint arXiv:1212.5701, 2012
- [63] T. Mikolov, I. Sutskever, K. Chen, G. S., Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, In Advances in neural information processing systems, Lake Tahoe, MIT Press, 2013, 3111~3119

相关的科研成果目录

X. Li, H. R. X., Y. H. Rao, Y. J. Chen, H. Huang, X. B. Liu and F. L. Wang, “Weighted multi-label classification model for sentiment analysis of online news,” in the 2016 International Conference on Big Data and Smart Computing (BigComp), pp. 215-222, 2016.

X., Li, Y. H. Rao, Y. J. Chen, H. Huang, X. B. Liu, “Social emotion classification via reader perspective weighted model,” in the 30th AAAI Conference on Artificial Intelligence (AAAI), 2016, poster paper, accepted.

致 谢

由衷感谢我的导师饶洋辉老师，本文是在他的指导下完成的。

李 昕

2016 年 04 月

附 录

论文附录依次用大写字母“附录 A、附录 B、附录 C……”表示，附录内的分级序号可采用“附 A1、附 A1.1、附 A1.1.1”等表示，图、表、公式均依此类推为“图 A1、表 A1、式 A1”等。

（注：对于一些不宜放在正文中的重要支撑材料，可编入毕业论文的附录中。包括某些重要的原始数据、详细数学推导、程序全文及其说明、复杂的图表、设计图纸等一系列需要补充提供的说明材料。如果毕业设计(论文)中引用的实例、数据资料，实验结果等符号较多时，为了节约篇幅，便于读者查阅，可以编写一个符号说明，注明符号代表的意义。附录的篇幅不宜太多，一般不超过正文。）

毕业论文成绩评定记录

指导教师评语：

为降低噪声训练文档对公众情感分类的影响，本文提出了基于“情感集中度”和“情感熵”的读者视角的加权模型（RPWM）以及加权多标签情感分类模型（WMCM）；对于社交媒体文本，由于无法引入投票信息来过滤掉噪声文档，本文提出了一种浅层的卷积神经网络 BtCNN 来更好地学习文档特征。总体而言，本文所提出的模型创新性较强，实验数据集及结果分析也较完善、优秀。

成绩评定：优秀

指导教师签名：



2016 年 04 月 20 日

答辩小组或专业负责人意见：

成绩评定：

签名（章）：

年 月 日

院系负责人意见：

成绩评定：

签名（章）：

年 月 日