分类号 密级

UDC

学 位 论 文

基于深度学习框架的中文情感分析方案

的设计与实现

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 作者姓名： | 筴硕 | | | | |
| 指导教师： | 薛定宇 教授 | | | | |
|  | 东北大学信息科学与工程学院 | | | | |
| 申请学位级别： | 硕士 | 学科类别： | | | 工学 |
| 学科专业名称： | 模式识别与智能系统 | | | | |
| 论文提交日期： | 2017年12月 | | | 论文答辩日期： | 2017年12月 |
| 学位授予日期： | 2018年 1 月 | | 答辩委员会主席： | |  |
| 评阅人： |  | | | | |

东 北 大 学

2017年12月

**A Thesis in Pattern Recognition and Intelligent System**

**The Design and Implementation of Chinese Texts’ Sentiment Analysis Program Based on Deep Learning Framework**

By Sean Jie

Supervisor: Professor DingYu Xue

**Northeastern University**

**December 2017**

# 独创性声明

本人声明，所呈交的学位论文是在导师的指导下完成的。论文中取得的研究成果除加以标注和致谢的地方外，不包含其他人己经发表或撰写过的研究成果，也不包括本人为获得其他学位而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名：

日 期：

学位论文版权使用授权书

本学位论文作者和指导教师完全了解东北大学有关保留、使用学位论文的规定：即学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人同意东北大学可以将学位论文的全部或部分内容编入有关数据库进行检索、交流。

作者和导师同意网上交流的时间为作者获得学位后：

半年 □ 一年□ 一年半□ 两年□

学位论文作者签名： 导师签名：

签字日期： 签字日期：

基于深度学习框架的中文情感分析方案的设计与实现

# 摘要

随着机器学习，特别是深度学习这几年的逐步兴起，其发展程度甚至达到了一个前所未有的高度。现如今，各行各业都在试图通过引入机器学习的人工智能技术来提高生产效率、改善产品性能、解放劳动力。作为利用深度学习等机器学习技术得到了广泛的进步的领域包括有：图像处理、计算机视觉、语音的识别与合成、自然语言处理技术。这些领域不仅受到了社会各界的广泛关注，也一直是整个学术界研究的热点。

近年来，随着互联网的不断发展，谷歌、Facebook、Twitter、百度、阿里巴巴、腾讯、新浪、网易、豆瓣、知乎等公司旗下的一系列互联网产品已经变成了人们生活中必不可少的组成部分。人们每天都会花费大量的时间在这些应用上，同时也会同时留下大量的评论文本数据，而这些留下的评论无疑则会体现着用户本人的态度。那么对这些评论进行情感分析有助于企业或是机构了解用户本人对于该产品、事件、现象、主题等内容的主观看法以及积极还是消极的态度，进而无论是对于下一步判断用户情感倾向还是改善服务质量都有很大的帮助。

本文首先将手动截取得到的数据样本构建成数据库并分割成样本集，测试集和交叉验证集，将样本集通过Word2Vec神经网络进行词向量化以数字化，然后对向量化以后的样本利用长短期记忆神经网络进行训练，用来区分不同的情感态度（正向、中性、负向）。之后利用交叉验证集来调整本文涉及到的神经网络的参数，并且对传统的长短期记忆进行一系列优化。最终利用测试数据集对整个神经网络进行验收，并完成整个方案的设计与改进。

关键词：自然语言处理，深度学习，词向量化，长短期记忆神经网络

# Abstract

In recent years, machine learning, especially deep learning is becoming hotter, its popularity has even become an unreachable height. Many different industries are seeking ways to use this new technology to improve manufacture abilities and their products. his new skill has already been proved advanced in image processing, computer vision, sound recognition and synthesis, and neural language processing. The whole society is always drawn attention on these areas, and the academic community has been focusing on these areas.

Nowadays, with the continuous development of the Internet, Google, Facebook, Twitter, Baidu, Alibaba, Tencent, Sina, Netease, Douban, Zhihu and many other companies and their applications are becoming indispensable in every day’s life. While spending many time on these apps, people write hundreds of thousands of comments on them, which undoubtedly reflect users’ attitudes. Making sentiment analysis on these comment texts can help us to know user’s opinions, positive or negative, on products, events, phenomena, or theme, in order to judge people’s emotional tendencies or improve productions.

In this paper, we first construct the database based on the data samples that are manually download, and divide them into sample sets, test sets, and cross validate sets. Then we digitized them through a Word2Vec neural network. Then a Long Short Memory Neural Work will be used to be trained to distinguish from different emotional attitudes (positive, neutral, negative). Then the cross validate sets will be used to adjust the parameters or constructor of our network. Finally, the test sets will be used to check the whole neural network and finish the whole system.

**Key words**: NLP, deep learning, word2vec, LSTM

# 目 录

[独创性声明 I](#_Toc499891667)

[摘要 II](#_Toc499891668)

[Abstract III](#_Toc499891669)

[目录 IV](#_Toc499891670)

[第1章 绪论 - 1 -](#_Toc499891671)

[1.1 研究背景及意义 - 1 -](#_Toc499891672)

[1.1.1 研究背景 - 1 -](#_Toc499891673)

[1.1.2 课题研究意义及目的 - 2 -](#_Toc499891674)

[1.2 国内外研究现状 - 2 -](#_Toc499891675)

[1.3 本课题的研究内容及章节安排 - 7 -](#_Toc499891676)

[1.3.1 主要研究内容 - 7 -](#_Toc499891677)

[1.3.2 论文结构 - 8 -](#_Toc499891678)

[第2章 评论文本数据集的建立 - 11 -](#_Toc499891679)

[2.1 评论简介 - 11 -](#_Toc499891680)

[2.2 情感分析语料库 - 12 -](#_Toc499891681)

[2.3 数据集的建立 - 13 -](#_Toc499891682)

[2.3.1 中文情感挖掘语料库 - 13 -](#_Toc499891683)

[2.3.2 数据集的扩充 - 14 -](#_Toc499891684)

[2.3.2 数据集的存储 - 16 -](#_Toc499891685)

[2.4小结 - 17 -](#_Toc499891686)

[第3章 深度学习神经网络模型 - 19 -](#_Toc499891687)

[3.1 循环神经网络 - 19 -](#_Toc499891688)

[3.1.1 神经网络简介 - 19 -](#_Toc499891689)

[3.1.2 循环神经网络结构 - 21 -](#_Toc499891690)

[3.1.3 循环神经网络的前向传播 - 22 -](#_Toc499891691)

[3.1.4 循环神经网络的反向传播 - 24 -](#_Toc499891692)

[3.1.5 梯度消失和梯度爆炸 - 25 -](#_Toc499891693)

[3.2 长短期记忆神经网络 - 26 -](#_Toc499891694)

[3.2.1 长短期记忆神经网络的门结构 - 27 -](#_Toc499891695)

[3.2.2 长短期记忆神经网络模型的建立 - 28 -](#_Toc499891696)

[3.3 模型效果分析 - 29 -](#_Toc499891697)

[3.4小结 - 30 -](#_Toc499891698)

[第4章 中文情感分析方案的设计 - 33 -](#_Toc499891699)

[4.1 分词系统 - 33 -](#_Toc499891700)

[4.2 词向量化 - 35 -](#_Toc499891701)

[4.2.1 Word2Vec - 35 -](#_Toc499891702)

[4.2.2 两种语言网络模型结构 - 36 -](#_Toc499891703)

[4.3 双向长短期记忆神经网络 - 37 -](#_Toc499891704)

[4.4 深度学习框架选择 - 39 -](#_Toc499891705)

[4.5小结 - 41 -](#_Toc499891706)

[第5章 参数优化与训练过程 - 43 -](#_Toc499891707)

[5.1 初始化状态 - 43 -](#_Toc499891708)

[5.2 Skip-gram模型参数选择 - 45 -](#_Toc499891709)

[5.2.1 负采样率 - 45 -](#_Toc499891710)

[5.2.2 降采样阈值选择 - 46 -](#_Toc499891711)

[5.2.3 词向量维度 - 47 -](#_Toc499891712)

[5.3 Bi-LSTM模型参数选择 - 48 -](#_Toc499891713)

[5.3.1 神经网络层数和神经单元 - 48 -](#_Toc499891714)

[5.4 训练过程参数选择 - 49 -](#_Toc499891715)

[5.4.1 大规模数据的梯度下降 - 49 -](#_Toc499891716)

[5.4.2 学习率的选择 - 51 -](#_Toc499891717)

[5.5 正则化 - 52 -](#_Toc499891718)

[5.5.1 Dropout算法 - 53 -](#_Toc499891719)

[5.5.2 循环Dropout算法 - 54 -](#_Toc499891720)

[5.5.3 循环Dropout率的设定 - 55 -](#_Toc499891721)

[5.6 实验结果与分析 - 55 -](#_Toc499891722)

[5.7小结 - 58 -](#_Toc499891723)

[第6章 总结与展望 - 60 -](#_Toc499891724)

[6.1 工作总结 - 60 -](#_Toc499891725)

[6.2 问题与展望 - 60 -](#_Toc499891726)

[参考文献 - 62 -](#_Toc499891727)

[致谢 - 65 -](#_Toc499891728)

# 绪论

本章为全文的绪论，将从宏观的角度介绍本文研究内容的意义和背景，并介绍国内外的主流的情感分析方法，介绍了先进十分火爆的机器学习技术以及这些技术是如何改善情感分析效果的。在本章节的最后，介绍了整篇文章的主要研究内容和章节安排。

## 1.1 研究背景及意义

### 1.1.1 研究背景

大互联网时代，无论是日常生活用的微信、Facebook，购物时候离不开的淘宝、京东、亚马逊，还是搜索时候必须使用的百度、谷歌，甚至连聚会看电影吃饭都不得不用到的美团、大众点评、豆瓣，这些大公司和他们的应用产品已经成为人们生活之中必不可少的重要组成部分，这些大量的应用程序之间又存在人与人之间的交互，一个人对其中某家店铺或电影的评价往往会影响到其他人的行为，这些评论的文本数据错综复杂且均与人们的生活息息相关，但如果试图通过人工的方式来鉴别，则如同在浩瀚的海洋中驾驶小舟前行，根本无法全部的遍历任务。那么如何利用这些海量的文本数据，对这些文本数据做出情感分析，分析说话人的态度究竟是怎么样的，他对于某一个事件、某一主题、某一现象、甚至是某一部电影究竟是持有怎样的观点，乐观、悲观、还是中立的态度。这种基于人工智能方式的情感分析可以很大程度节约人力成本，减少不必要的重复体力劳动。

1946年，冯·诺依曼和他的现代计算机问世以来，人们就逐渐发现，在一些重复计算的事情上，计算机做得比人好得多得多。而关于让计算机理解人类语言则伴随了了整个计算机的发展史，最早可以追溯到1950年，大名鼎鼎的图灵先生发表的《计算机与智能》一文中对于人工智能会将从自然语言处理方面实现突破[1]，这篇文章中提到的问题也就成了日后天下闻名的图灵测试。而后心理学家弗兰克·罗森布拉特和他的划时代的感知机的[2]横空出世正式宣告了神经网络的面世。而后在语言学天才诺姆·乔姆斯基的研究下[3-5]，自然语言处理的计算机处理方式由试图让计算机理解复杂的语法与文法规则转变为了基于统计学的语言模型。而后信息论的创始人香农通过对“信息论”的发展与完善[6]，以及马尔科夫条件随机场的思想为机器翻译注入了新的活力[7]。深度学习时代，自然语言处理领域迎来了爆发期，特别是循环神经网络(Recruit Neural Network)和长短期记忆神经网络(Long Short Time Memory)在处理时间序列上被证明有着非常不错的效果，因此也为情感分析提供了技术支持。

而国内关于自然语言处理的研究起步较晚。一方面是由于我国发展计算机事业很晚，国内的专家学者以及研究人员出现了短暂的断层真空期，另外一方面的原因是由于汉语本身博大精深，其语法和表达方式都与拉丁语系的英语、法语、德语有较大差异[8]，故我国自然语言处理方面的研究与应用扔落后于欧美国家。

### 1.1.2 课题研究意义及目的

那么究竟什么才是情感分析呢？文本数据根情感又有怎么样的关系呢？学术界对于情感有着明确的定义[9]，情感是一种感觉，由态度、情感、观点三部分组成，它是评论人的主观印象，而非事实。通常可分为支持／反对，喜欢／不喜欢，好／坏等相对立的观点，对于某些复杂的情况还会有介于二者之间的中立态度存在。



图1.1新浪新闻下面的评论样图

Fig.1.1 Samples of comments from sina news

对于这些评论文本进行情感分析和观点挖掘将对于我们实际生活有很大的帮助。例如，公司或者相关机构可以通过大家的评论来检验某款产品的市场反馈是积极的还是消极的；可以通过检验邮件的内容来判断该邮件安全与否；群众对于某一个政治事件、或者活动的观点是怎么样的，诸如此类不胜枚举。

在当前的现实生活中，已经有不少公司利用一些诸如机器学习、数据统计的方法来进行情感分析来指导生产了。例如阿里巴巴公司的淘宝产品就有针对用户评论的销量分析系统，谷歌公司有针对邮件的垃圾邮件过滤系统，IMDB公司的电影网站上有专门的评论分析器可以通过观众的评论及打分预测电影的票房。

## 1.2 国内外研究现状

自然语言处理技术发展的初期阶段，主流方式是通过基于统计学的词法分析器[10]模型，其方法是利用马尔可夫随机场和信息熵的方式来确定最大信息熵，来确定统计语言模型，然后进行文本挖掘、情感分析工作。

上世纪80年代开始，反向传播算法(即Back Propagation算法，简称BP) [11]，多层感知器(Multilayer Perceptron) [12]的出现以及基各种传统的机器学习模型陆续问世，例如Boosting [13]、最大熵法[14]，支持向量机 [15]等。这些传统机器学习模型大都可看作是简易的浅层神经网络，通常含有一层隐含层或不含隐含层。这些浅层模型的出现直接使得语言建模的准确度提高了几个量级，很多诸如机器翻译，情感分析工具就是在那一时期发现出来的。

而多伦多大学的Hinton教授，早在1985年，就已经提出了通过用于学习概率分布的深层神经网络的方法[16]。而Hinton在2012年的ImageNet大赛上，他和他的学生组成的团队利用卷积神经网络(Convolutional Neural Network,即CNN)，通过准确率以压倒性的态势战胜了所有的参赛者方式[17]，使得深度学习大放异彩。也从此揭开了深度学习火热浪潮的序幕。而在此之后在谷歌，Facebook，百度，亚马逊等大公司的推动下，各大院校研究所纷纷设立专门研究机构，深度学习的热浪席卷了图片处理，计算机视觉，语音合成与识别，当然也包括自然语言处理领域。一时间，各种模型被提出，各种改进方案如同百花齐放一般引发了一场不大不小的人工智能浪潮。而CNN的原作者，同时也是Hinton教授的学生的Yann LeCun先生在加入了Facebook公司出任人工智能实验室主任之后又对CNN做了进一步的探索及完善。在Hinton和LeCun的令人震惊的工作至上，众多学者发现CNN也可应用于自然语言处理领域[18] ，特别是诸如语义分析，垃圾邮件检测和话题分类之类的简单分类任务。同时又有一些改进的方案被提了出来[19-20] ，这些改进方案通过修改神经网络结构或增加层数，可以完成语义聚类的任务。Kim.Y等人发明了利用词袋表征，直接对one-hot向量进行卷积运算的方式的方式对文本数据进行CNN的训练，可以用来预测文字区域上下文的内容，也取得了不错的效果[21-22]。但是利用CNN来进行自然语言处理还有几个待解决的问题：首先，由于卷积神经网络中卷积运算和池化运算会丢失某些语句的顺序信息，因此不适用于某些顺序标签任务的语义处理任务，而中文由于有各种讽刺，倒装，反语等修辞手法的存在，使用CNN有时会造成巨大的歧义，导致网络无法收敛，例如“此地安能居住，其人好不悲伤”，如果采用不同的读法，则会有完全两种不同的意思。这句的意思可以是“此地安，能居住。其人好，不悲伤”，这句话就是明显的褒义存在，但是如果是“此地安能居住？其人好不悲伤！”则明显含有消极的意义。此外选择CNN模型在训练的过程当中需要选择许多个超参数：输入特征(word2vec, Glove, one-hot)，卷积滤波器的尺寸和数量，池化策略（最大值，平均值），当然也包括激励函数(Sigmoid, ReLU, tanh)的选择。不同的超参数的选择会对CNN模型的性能和稳定性上产生很大的影响。



图1.2 卷积神经网络用来语句分类

Fig.1.2 CNN for sentence classification

随后出现的循环神经网络(Recruit Neural Network, RNN)则是另外一种热门的神经网络结构。RNN最早的提出可以追溯到1989年，包括Hinton教授在内的几名教授,以亚历山大•魏贝尔教授为主的几名教授联合提出的时间延迟神经网络(Time Delay Neural Network, TDNN) [23]。这种神经网络结构上十分类似传统的神经网络，但是每个神经元可以近似的处理输入的一个子集，对于不同时刻的输入数据，该神经单元有不同的权重。从某种意义上讲，这与CNN算法十分相似，同样的，都是都是每个神经单元对某一个子集进行相同的操作，而不是对整个输入进行操作。

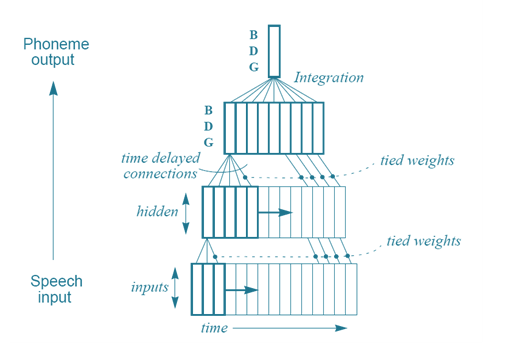


图1.3 时间延迟神经网络

Fig.1.3 Time Delay Neural Network

然而TDNN本质上还是顺序的输入-输出结构，目前为止所有的神经网络都是由前向后正想传递的，这就意味着每一层神经单元的输出仅能作为下一层的输入。后来在深度学习领域另一位泰山北斗，加拿大科学家约书亚•本吉奥教授等人的杰出工作与不懈努力下[24-25]，循环神经网络(RNN)终于在2012年由亚历克斯•格雷夫教授在他的书中正式做出来完整的总结表述[26]。循环神经网络与其他神经网络最大的不同之处在于，循环神经网络中的每一层的神经单元的输入不仅有来自上一层神经单元的输出，还有来自同一层神经单元的输出。换言之就是神经网络的结构是会有同层之间的传递。这样的结构有一个好处就是神经网络可以展现出记忆性，因为隐含层的输入不仅包含输入层的输入，同时还包含隐含层前一时刻的输入。这样的特性使得RNN在处理时间序列的数据上面有着非常好的效果，而RNN的问世也使得TDNN推出了历史的舞台，从此RNN接管了NLP领域中的诸如词向量表达，词性标注，机器翻译，文本生成等的绝大部分任务，而且很多公司纷纷表明RNN在这些工作中取得了很好的效果。

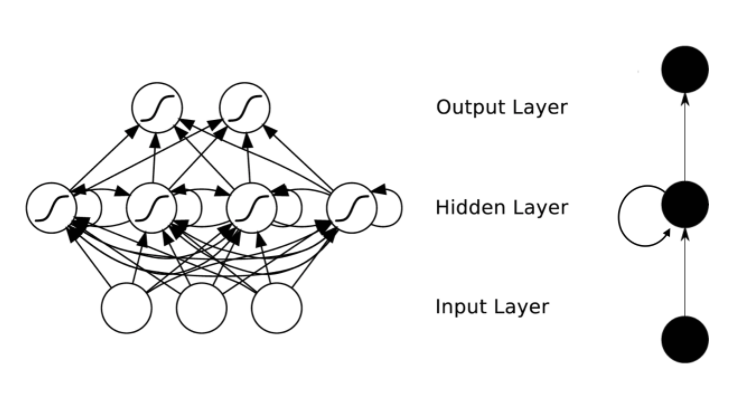


图1.4循环神经网络结构

Fig.1.4 Structure for Recruit Neural Network

然而在训练RNN的过程当中，研究者们逐渐发现，RNN神经网络或者取得非常杰出的效果，要么根本无法收敛，训练RNN似乎变成了运气的事情，但是在学术界，有关于运气的事情往往都是不存在。学者们对深入的具体原因进行数学层面的探究[27]。

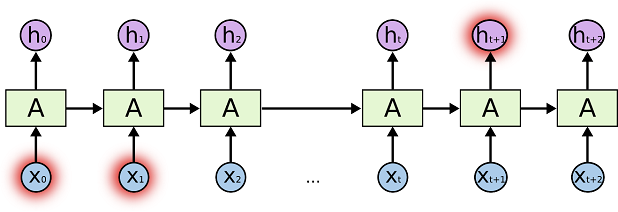


图1.5 用于预测的循环神经网络

Fig.1.5 Predicting job using RNN

因为RNN的结构设计的目的是使的神经网络内部的神经单元全部可以实现连接，形成一个循环的网络。那么神经网络模型结构中的输出层的神经单元的误差函数就要传回到第一层的隐含层中的神经单元，而传递的路径就是从后向前的每一层的神经网络中的神经单元进行横向传递，又由于每一层的网络都有各自的权重，如果这个权重是一个很大的数，那么经过几次传递，原始的误差函数就变成了一个无穷大的数，这也是著名的梯度爆炸(Exploding Gradient )现象。那么如果网络权重不是很大的数呢？假设权重是一个接近于1的数，假设是一个有五个隐含层的RNN网络，每层有20个神经单元，那么一共就有100个神经单元，也就有100个权重值。假设权重是1.1、误差函数是1，那么1.1的一百次方等于多少呢？答案是13780.61233，相对于原始的1来讲，这个数字也就等同于无穷大了；如果权重小于1呢？在刚刚的基础上重新假设权重是0.9，0.9的100次方是。这个值与原始的1相比也就相当于不存在了，那么这一现象也就是与梯度爆炸相对应的梯度弥散(Vanishing Gradient)现象了。

为了解决RNN中普遍存在的梯度弥散和梯度爆炸现象，约翰开普勒林茨大学的赫克多教授，在1998年提出了一个RNN的改进方案[28]，这个改进方案引入了门的设计理念，在每一个RNN神经单元之中加入了输入门、输出门与遗忘门。门的概念有效的解决了梯度弥散和梯度爆炸现象，而这一改进方案后来有了自己的名字，长短期记忆神经网络(Long Short Time Memory Neural Network, LSTM),而LSTM的问世，使得自然语言处理领域的大多数问题都有了不二的选择。在NLP社区之中，一直广为流传的一句话就是：“没有什么NLP问题是LSTM解决不了的。”自此，NLP领域的技术条件就已经发展的比较完备，剩下的就是结合应用场景实际应用，其中一些诸如谷歌[29]，Facebook，百度等大公司都取得了不错的应用效果。针对情感分析，谷歌公司推出的谷歌邮箱可以针对邮件内容自动过滤垃圾邮件；IMDB公司提供的电影评论分析功能，可以分析影评人的评价态度分析影片热度，进而预估票房走势；Facebook也在美国大选期间通过舆论分析来分析各个候选人的支持度，进而预测得票数。



图1.6 IMDB热度预测

Fig.1.6 Popularity predicting on IMDB

然而国内的自然语言处理还处于起步阶段，这些先进的技术还没有得到良好的应用。这是由多方面的原因造成的：一是汉语与拉丁系的英语、德语、法语、西班牙语、意大利语等欧美主流体系仍有很大不同，在NLP发展到今天这样的一个程度，英汉／汉英的翻译准确率也远远达不到英法／法英的准确度。其二国内掌握有大量文本数据的公司并未采取公开数据的方式，大家纷纷各自成立研机构究闭门造车，这样也从某种程度上抑制了汉语情感分析的发展研究。第三就是我国计算机发展比较晚，基础比较弱，计算机类的人才储备较少，科研中坚力量不足。国内的目前的情感分析技术有：使用分词技术对文本分词然后利用词典进行关键句分析[30]；利用诸如支持向量机、最大熵法、高斯分布、决策树等传统机器学习方法[31-33]进行分析；利用TF-IDF等词频统计学方法进行分析[34] ；也有利用CNN、RNN、LSTM等深度学习方法进行情感分析的[35-36]。

在现阶段中文文本情感分析领域，明显使用规则统计，或是使用频率分析等简单的方法的效果可以比深度学习的方法好很多。因为规则统计、或是频率分析有人多人工的工作蕴含在其中，可以完成某些对人来讲较为简单，但是对电脑较为困难的工作。举个简单的例子，“我家的洗衣机声音很大。”这句话可能就是贬抑的，但是“我家的音响声音很大。”这句话可能就是褒义的了。通过人工设定词典的方式，就可以避免神经网络在学习的过程中出现的歧义现象。但是这并不能成为阻碍科学进步，技术发展的前进道路的理由。因为科研的道路从来就没有捷径，也从来不是按照可以用的标准来发展的，所以本文还是要对中文文本情感分析进行诸如深度学习等先进技术的尝试与科研探索。

## 1.3 本课题的研究内容及章节安排

### 1.3.1 主要研究内容

语言的意义是对于自然界中某种事物，或人类的某些行为进行编码。之所以有不同的语言是因为在历史的发展过程当中各种文明互相对某种事物和人类行为的表述不同。按照计算机的描述就是看作是对自然界中的某种事物和人类行为进行编码，而看到这些文字的人在大脑里对文字进行解码，获得事件本身，而语言的不同在计算机里可以用编码的方式不同来解释。

由于计算机从被创造出来伊始，其存储结构决定了计算机只能存储0和1组成的二进制序列。因此无论是什么样的深度学习算法，都只能处理数字化得文本序列，而人类语言都是有单词或汉字之类的基本语言单位组成，同计算机语言之间有着天然的鸿沟，为了消除这一鸿沟，则需要对文本信息进行数字化，用来得到可以被计算机是别的文本信息的数字化表达，即词向量[37]。

本文中，将首先将得到的文本数据集进行分类，分成正向类(Positive)与负向类(Negative)两类，之后将所有的数据集进行不同的词向量化尝试，以分析不同的词向量方法的特点。然后利用深度学习框架进行对神经网络的学习。最后再利用分类器得到最终的结果，以下为本文的技术路线：

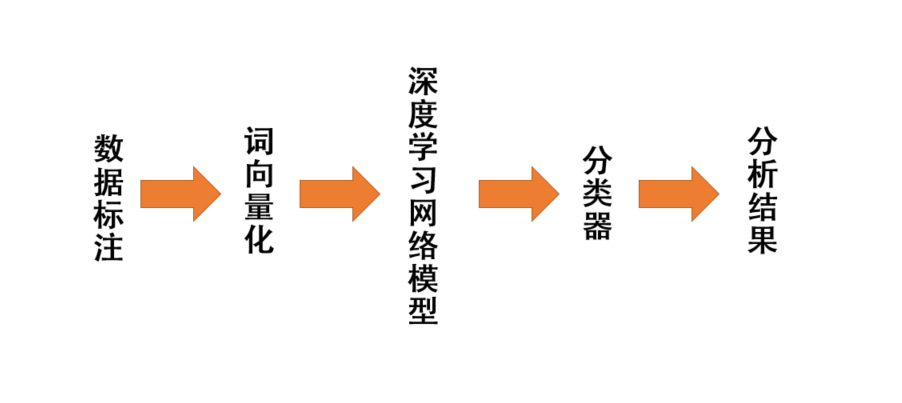


图1.7论文框架

Fig.1.7 Paper framework

常规的情感分析和观点挖掘分类结果有两类和三类两种，两类就是传统的正向类与负向类，三类就是在两类的基础上增加了中立类。IMDB在其网站公布的数据就有这两类。鉴于中立类的存在将会有可能导致神经网络无法收敛或发生过拟合现象，因此本文将不设中立类分类结果。二分类的划分参考IMDB的划分方式，以10分为标准，5分以下包含5分的评论作为负向类，超过5分的评论则被统计为正向类。



图1.8 IMDB的评论及分类

Fig.1.9 Comments and classification on IMDB

通过以上方式，本文获得了特征明确并且拟合难度低的数据样本。NLP领域的深度学习框架与图像处理领域的深度学习框架有很大的不同，其表现为层数少，神经单元多，因此在后续的文章之中，本文将会对不同的神经网络结构进行探讨。

### 1.3.2 论文结构

本文的研究为深度学习以及自然语言处理中的情感分析，全文共分为六个章节，以下为每个章节的简介：

第一章本文分为研究背景及意义，国内外研究现状和主要研究内容以及对本文行文的安排做了介绍。其中本文大致的介绍了深度学习的发展历史，粗略的了解了自然语言处理发展历史的三个阶段，介绍了中外自然语言处理领域的研究进展并分析了大致的原因。以及本文的研究目标和论文的大致结构。

第二章本问讨论的工作主要为为文本数据集的建立。本文通过使用开源数据集的方式来建立样本数据集，对于样本量较小的问题，本文采取爬虫技术，抓取互联网上的一些评论文本作为样本数据集的补充。之后本文将数据集以6:2:2的比例划分为样本集，交叉验证集和测试集。

第三章本文探讨了几种比较流行的适用于自然语言处理的神经网络，将这几种网络分别从数学上进行推导与结构分析，分析其特点。并提出了适合本文的一种双向神经网络模型。

第四章本文介绍了本方案的流程，以及各个流程的作用和实现方式，并对每个方案进行优缺点分析，以及本文的最终选择。

第五章本文提出了一种完整的情感分析方案的实现流程，以及训练神经网络当中需要注意到的一些细节，包括神经网络结构、参数优化和训练方式等，通过对比试验确定了选择的参数。并分析结果。

第六章为总结与展望，本章内容为对前五章内容的总结，分析本系统的优点与不足，并展望下一步的工作目标。

# 评论文本数据集的建立

本章节将介绍什么是评论，以及评论的社会意义。然后介绍国内外在进行情感分析时都采用了哪些数据集和文本集，以及本文将采取哪种数据集。针对数据集中数据太少，样本容量不够现象，本章将介绍通过网络爬虫的方式抓去电影评论数据，扩充本文的样本容量。

## 2.1 评论简介

评论是指主体发出者针对实物进行主观或客观的自我印象阐述，其阐述可以使正面的，也可以是负面的，引起简单易懂，更可以抒发发出者本人的情感，故此已经广泛的为大众所使用。按照评论对象的不同又可以分为新闻评论，电影评论，使用评论，商品评论等。互联网时代，评论具有结构间断，传播时间短，传播速度快等一系列特点，已经成为了一种协作手段，一种传播力量，一种社会存在。



图2.1 新闻评论实例

Fig.2.1 The samples of news comments

针对评论最早的分析是早在2002年由当时还在康奈尔大学读博士的Bo Pang提出的[38]。他的第一篇关于情感分析的开山之作，使用了电影评论数据，通过对比统计学方法与机器学习的方法（朴素贝叶斯，最大熵分类和支持向量机）进行比较，分宜电影评论文情感分类情况，分析了机器学习技术明显优于人工基准的方法，并最后总结了研究这一问题更具有挑战性的因素。他们的这一篇论文已经面世，立刻受到了社会各界的广泛关注和认可，Bo Pang本人也在毕业后出任谷歌工程师也继续从事了这一行业的研究，写出了诸多观点挖掘与情感分析的论文，在这里就不一一表述了。

使用电影评论数据作为情感分析的数据集在此之后似乎成为了一个标准，其原因有三：一机器学习中的监督学习需要样本是标记完成的，也就是分类过的，而诸如评论之类的文本其褒贬性很难用客观评价，同样的一句话，可能不同人看了之后感受不一。而电影评论由于同时还会引入打分机制，这样就可以有一个与评论人感受一致的评分标准，可以作为样本的分类原则，因此使用电影评论数据可以减少人工标注样本的复杂任务。二是由于诸如产品评价、用户感受之类的文本数据对于这些公司有着保密的特点，从对产品的评价就可以了解到该产品的销量进而得知公司的一些不便公开的情况，因此大多数公司往往不愿意公开其收集到的产品评论数据。三电影评论数据易于获取，电影作为人们日常休闲娱乐无法离开的一部分，各大电影评分或售票网站均有开放评论，其评论数据量十分巨大且易于获取。



图2.2 某购物网站上的评价截图

Fig.2.2 An example of comment on a shopping website

上图所示的就是某知名购物网站上的评论截图，客服人员正在对于负面的评论进行解释与说明。但是如果评论的数量足够多的时候，人工成本将会十分巨大。

## 2.2 情感分析语料库

情感分析语料库在学术界有很多不同的版本，但是大多数都是英语评论的，他们之中经常被使用的有：

（1）康奈尔大学提供的影评数据集，该数据集由持有肯定态度与否定态度的电影评论各1000条组成，另外还有标注有褒贬极性的句子各5331句，标注有主客观标签的句子各5000句。

（2）伊利诺伊大学芝加哥分校提供的产品评论数据集，该数据集包含了亚马逊和Cnet网站上下载到的关于五种产品（包括两个牌子的数码相机，MP3播放器，手机和DVD播放器）的评论，这些评论被详细的标注了评价对象、情感分析极性等信息。

（3）麻省理工大学的餐厅评价集，这些餐厅评论语料分别按照菜品、就餐环境，服务质量，价钱和整体体验五个方面被标注了五个等级，共计有4488篇短评。

（4）Janyce Wiebe等人开发的MPQA(Multiple-Perspective QA)库，该库包含了535篇不同角度的新闻及评论，这些评论被深度的标注了。标注者手工的为每个句子标注出了一些诸如情感信息、观点持有者、评价对象、主管表达以及其强度等信息。

（5）IMDB提供的影评集，包含有IMDB平台的的电影评论50000条，这些电影评论按照评论时的打分被分为不同的两个情感分类。

国内由于对情感分析的研究起步较晚，目前还没有像英文语料库这么多的的语料库，国内比较成功的中文语料库有当时就读于中科院计算所的谭松波博士整理并公开的大规模评论集，该评论集主要由三部分组成，分别是酒店评论，笔记本电脑评论于书记评论，其中酒店评论共有正负类各5000篇、笔记本电脑和书籍评论各有平衡资料正负类各2000篇。

## 2.3 数据集的建立

2.3.1 中文情感挖掘语料库

从之前的章节中可以知道，适用于中文情感挖掘的语料库并不是很多。因此本文选择谭松波博士的酒店类评论语料集ChnSentiCorp，笔记本电脑的语料集由于重复率太高，样本可用度小不予采纳，而书籍评论由于长度往往超出限制，并不符合一般短评的特点，也不予以采纳。

深度学习算法之所以取得了这么多的成果,学术界和工业界主流的意见认为共有三方面的原因，第一是海量的数据，第二是高性能计算硬件的产生，第三时大规模并行计算方式的提出，这三者共同造就了深度学习今日的火爆。而现在仅有这一万条数据是远远满足不了深度学习神经网络的需求的，数据量小对于神经网络而言，往往达不到可以使神经网络收敛的样本规模，也就是整个网络模型时浅拟合的。对此常规的解决方案有两种，一是扩大样本集，二是采取对抗生成模型(Generative Adversarial Networks，GANs)。其中对抗生成模型是一种可以从小样本集中不断模仿样本数据，然后通过判别器判断，最终获得一个较大规模样本数据的生成型网络[39]。GANs虽然自提出以来就受到了广泛的关注和研究，但是至今未见在NLP领域有较为成功的应用实例，故本文采取简单的方法，扩大样本集规模。

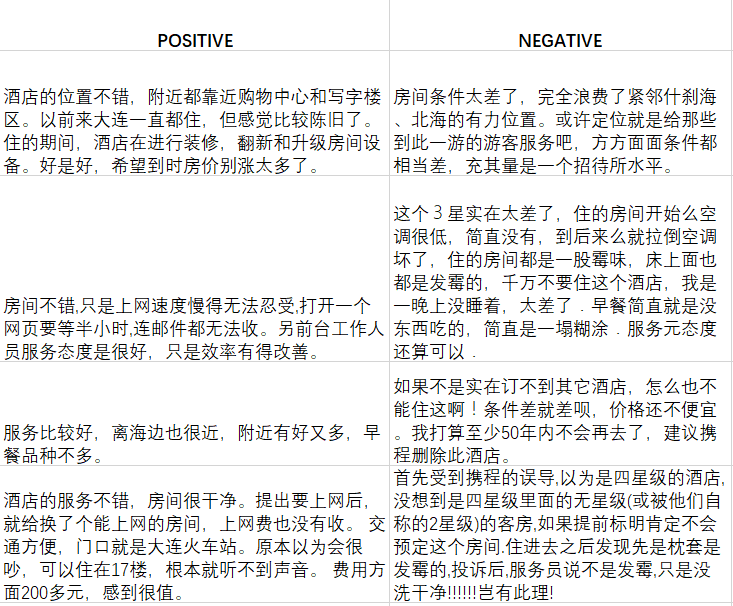


图2.3 中文语料库评论截图

Fig.2.3 The examples of ChnSentiCorp dataset

2.3.2 数据集的扩充

豆瓣是广受好评的中文社区网站，该网站以书影音为主，提供关于书籍、电影、音乐作品等信息，无论描述还是评论都由用户提供。该网站每日活跃量超过2亿。该网站的豆瓣电影板块，始终被认为是最大最权威的电影分享评论社区，该社区收录了超过百万条的影片和影人信息，在每一部电影下面，都会有影迷们的短评和打分，累计已经有超过上亿条的点评信息。其中每一条点评都由作者、影片、影评、标题、评分五部分组成。

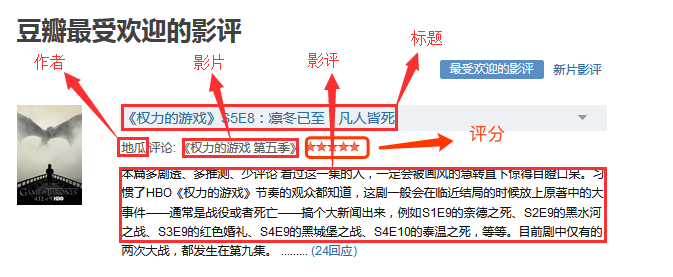


图2.4 豆瓣电影点评样例

Fig.2.4 A Sample of comments on Douban Movie

如上图所示，该截图是一个经典的豆瓣影评的组成部分。我们之所以看到网页所示的样子，是浏览器将文件渲染成了这个样子。如果想要读取出网也之中的数据，就需要拿到对应的资源文件，也就是HTML文件。

图2.5 对豆瓣电影评论页部分HTML信息

Fig.2.5 Part of html on Douban Movie

通过向对应的URL发送HTTP请求的方式，可以获得目标地址返回的响应。然后对于响应返回的信息进行解析，这就是爬虫的工作原理。

在的到的数据之中，可以整理出更多的信息，包括发表时间和用户名，用户等级和用户最近的登陆时间等等。但是这些信息并不是本文必须的信息。



图2.6 豆瓣电影评论页HTML包含所有信息

Fig.2.6 All information of html on Douban Movie

对这些信息用正则的手段进行过滤，则可以得到本文关心的评论文本和对应分数。其中评分为1分和2分的，将会被标记为负向类，4分5分的则会被标记为正向类，3分的评论，由于本系统设计的是二分类，故不纳入本次的样本集。至此，本文需要的样本集就已经完全建立成功，建立成功之后，数据应该被打乱，然后以6:2:2的比例分割成为训练集，交叉验证集和测试集，其大小分别为：训练集正负各一万五千条，交叉验证集正负各五千条，测试集正负各五千条。

2.3.2 数据集的存储

在获取到这些信息以后，下一要解决的是对这些海量的文本信息进行存储。以下是几种存储方案：

（1）将每条文本信息以.txt文本信息的格式存储在本地。

（2）将文本信息写入本地的关系型数据库MySQL，其中分别以正向类和负向类作为主键，并且还要对不同的样本数据随机分配一个id以区分不同的评论作为主键建表。

（3）将文本信息写入云端非关系性数据库MongoDB。

由于无论用哪种方案都需要对整个样本集进行遍历，下面我们就采用相同配置的云服务器，分别对几种方案下的文本数据进行遍历，分析遍历所需时间：

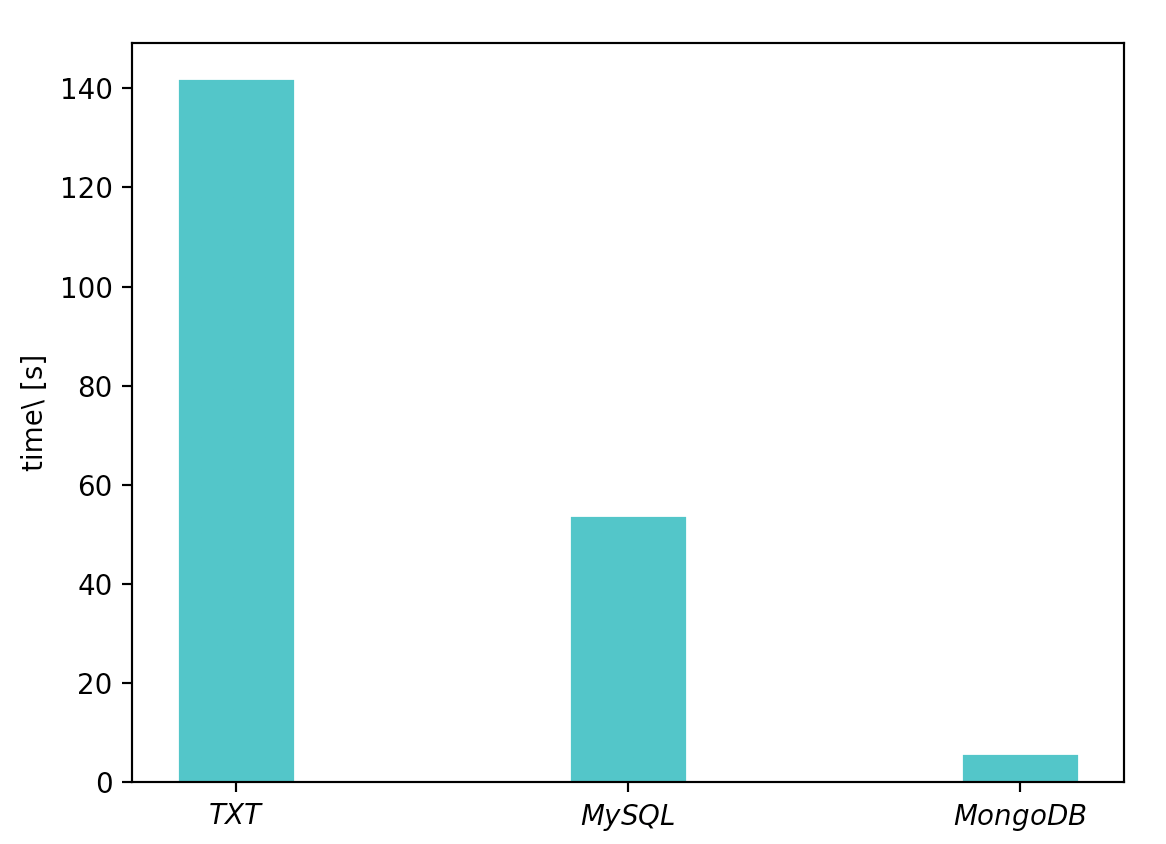


图2.7 几种不同存储方式的执行时间

Fig.2.7 Time of using different database to store data

从上图可以看出基于分布式的文件存储数据库MongoDB在执行时间上有着无可匹敌的速度，并且由于它的非关系行型结构的存储关系，并不需要将正向类和负向类样本分开存储，也提升了不少的性能空间。

## 2.4小结

本章介绍了情感分析系统的由来，以及国内外使用较多的文本集，以及这些文本集的特点。之后针对数据集太小的问题，本文通过爬虫访问的方式，获取最大中文电影社区豆瓣电影的评论数据。之后完整的建立起本问的情感分析中文文本样本集，最后本文随机的方式将样本集分为训练集、交叉验证集和测试集三个子集。

# 深度学习神经网络模型

本章将通过数学推导的方式详细介绍循环神经网络的工作流程，以及梯度下降方式等细节信息。接下来将从结构上分析循环神经网络面临的问题，以及梯度现象和梯度爆炸现象产生的数学原因。随后介绍了循环神经网络的升级结构，长短期记忆网络，并仔细介绍长短期记忆神经网络的门结构。最后提出了适合本文的情感分析的双向长短期记忆神经网络模型。

## 3.1 循环神经网络

循环神经网络(Recurrent Neural Networks，RNN)源自于美国加州理工大学物理学家J. J. Hopfield教授在1982年提出的单层反馈神经网络，人们后来成这种反馈网络为霍普菲尔德(Hopfield)神经网络，虽然Hopfield本人利用该模型解决了旅行商(TSP)问题，但是由于霍普菲尔德网络实现起来十分困难，在提出时期，并没有合适的应用场景，因此该网络结构也与1986年前后被全联接神经网络以及其他传统的机器学习方法所取代。然而随着人工智能的逐渐发展，学者们逐渐发现，传统机器学习方法十分依赖人工特征的提取，而诸如自然语言处理、图像识别、计算机视觉当中特征提取存在着诸多限制。全联接的神经网络也存在着诸如参数太多，参数调节困难等诸多瓶颈，因此诞生了更有效的循环神经网络结构，这些结构也在语音识别、机器翻译以及时间序列预测上取得了不错的效果。

### 3.1.1 神经网络简介

介绍循环神经网络之前，需要补充介绍神经网络的一些基础知识，神经网络的基本组成结构叫做神经元，其结构如图3.1：



图3.1单个神经元

Fig.3.1 Single neuron

上图中的神经元是以一个及截距为+1的输入值的运算单元，对应的公式为：

(3.1)

而其中的为激活函数，激活函数即使神经元本身对输入之和进行的某些线性或非线性运算，可以似的神经网络有着更好的拟合能力，常用的激活函数有Sigmoid函数：

(3.2)

实际应用中我们常用的激活函数还有双曲正切函数，公式如下：

(3.3)

以及线性校正单元激活函数：

(3.4)

以下是sigmoid和tanh函数的曲线图：

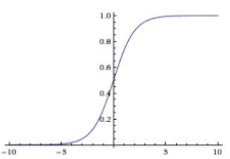
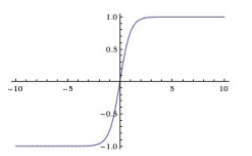
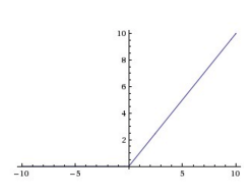
  

图3.2 sigmoid(左), tanh（中）函数和ReLU函数（右）曲线图

Fig.3.2 The curves of sigmoid(left) , tanh(middle) and ReLU(right)

从上图可以看出，sigmoid非线性激活函数映射到实数的区间为[0,1]，tanh非线性函数映射到实数的区间为[-1,1], ReLU函数映射到实数区间为[0, x]，其中x是函数的输入。对于神经网络，每一个隐含层的神经单元都需要加入激活函数的神经单元，这些激活函数可以弥补神经网络线性表达力不足的缺点，增加非线性因素。

以上介绍的是单个神经单元和几种不同的激活函数，将多个神经元被组合链接，并且之拥有了一定的层次结构的时候，神经网络的模型就搭建好了，下图即为具有一个隐含层的神经网络模型。



图3.3神经网络模型

Fig.3.3 The model of neural network

其对应的公式如下：

(3.5)

(3.6)

(3.7)

(3.8)

隐含层的优势之处在于：隐含层可以在数据之中寻找特征，并传递给下一层进行操作，而不用处理大量的原始数据，更不用处理噪声的干扰。通过链式法则，可以轻易知道，神经网络通过调节权重似的误差最小化，还可以通过计算先前层的神经网络的导数的方式来调节他们的权重。简而言之，可以通过微积分的方式，讲训练集的误差放置在先前隐含层的神经单元，在有更多隐含层的情况下，则还可以额进一步分裂误差。通过反向传递的方式，就可以改变神经网络中所有神经单元的权重，并最终选择最合适的权重值。

### 3.1.2 循环神经网络结构

和卷积神经网络中过滤器参数的共享概念类似，循环神经网络中，循环神经网络结构中的参数在不同时刻也是共享的。神经网络通过不同时刻的输入共享网络结构的方式，把时间概念引入整个神经网络循环之中，而这正是卷积神经网络和其他神经网络做不到的。通过在神经网络的结构里引入循环的概念，使得神经网络本身可以保持一段时间的信息，就如同神经网络有了时间的记忆。

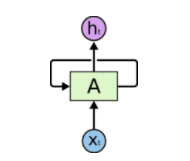


图3.4 循环神经网络经典结构示意图

Fig.3.4 A classic structure for RNN

上图所示的神经网络展示了一个典型的循环神经网络。在循环神经网络之中，一个非常重要的概念就是时刻。循环神经网络会对于每一个时刻的输入结合当前时刻模型的状态给出一个输出。从图3.4上可以看出，RNN的主体结构A的输入出了有来自于输出层的之外，还有一个来自于当前时刻的状态，这样就形成了一个循环。在每一个时刻，RNN的主体结构A除了会读取t时刻的输入并输出一个之外，还会把A当前的状态从当前时刻传递到下一时刻，也就是从当前步传递到下一步。理论上，循环神经网络可以看作上市统一神经网络结构被无限次复制的结构，但是由于目前的计算性能以及优化角度等限制，目前的循环神经网络无法被真正的无限循环。在实际应用当中，往往会将循环体进行进一步的展开，于是得到图3. 5所示的结构。



图3.5 循环神经网络展开结构示意图

Fig.3.5 A sketch map of an unrolled RNN structure

从图3. 5可以更加清楚的看到RNN在每一个时刻都有一个输入，然后根据RNN当前的状态提供一个输出。而RNN当前时刻的状态是根据上一时刻的状态和当前时刻的输入共同决定的。由这样的神经网络结果可以得出结论，即RNN十分擅长解决时间序列的输入问题。对于一个数列数据，则可以按照时间前后的顺序，将不同时刻的数据（一般是按照时间戳从小到大传入）传入到神经网络之中，用来预测下一时刻的序列数据，或者是当前时刻神经网络的处理数据。

### 3.1.3 循环神经网络的前向传播

RNN的模型有着很多的变种，每一种的框架都是如前面介绍的，但是具体的神经元的链接细节则不尽相同，这里本文以主流的传统RNN为例，介绍它的前向传递的过程，以及数学推导。



图3.6 RNN结构图

Fig.3.6 A structure chart of RNN

上图左边所示的RNN模型，按照时间序列展开，则得到上图的右边部分，这是在时刻，RNN神经网络的内部神经元展开图。其中：

1. 表示时刻，神经网络的训练样本输入值。
2. 表示时刻，该神经单元的状态值，共同决定。
3. 表示时刻，该神经单元输出值，决定。
4. 表示时刻，神经网络的代价函数。
5. 表示时刻，神经网络的训练样本的真实值。
6. U、W、V这三个矩阵则是模型的参数矩阵，表达着各个神经单元之间的连接权重，在整个RNN神经网络之中，这三个矩阵是共享的。

在任意时刻 共同决定:

(3.9)

其中为RNN的激活函数，一般都会采用tanh函数，关于激活函数的具体公式可以参考前一节内容。b是该层神经网络的偏置节点，也被称为截距。那么在时刻，模型的输出为:

(3.10)

最终在时刻，得到的预测输出为：

(3.11)

其中是神经网络的输出层的激活函数，由于RNN常被用来解决分类问题，则这里一般都是softmax回归，那么此时的损失函数为：

(3.12)

### 3.1.4 循环神经网络的反向传播

有了RNN前向传播的的数学推导，那么就可以对RNN进行反向传播对数学推导。RNN的反向传播与传统的神经网络一样，通过一轮轮的梯度下降算法迭代，找到合适的RNN神经网络参数，也就是U、W、V、b、c。由于RNN是基于时刻而不是数据集，RNN的反向传播优势也叫BPTT (back propa-gation through time)，另外，RNN的参数是所有位置共享的，因此每次更新都是更新的全局参数。

对于一个完整的RNN模型，由于每一时刻都有一个损失函数，那么最终的损失函数为：

(3.13)

其中V和c的偏导数比较容易直接通过求偏导数得到：

= (3.14)

= (3.15)

而由于某一时刻的梯度损失要有当前时刻的梯度损失和时刻的梯度损失两部分共同决定。因此W、U、b的偏导数则没有那么容易计算，先定义时刻隐藏状态的梯度为：

(3.16)

则根据链式求导法则可以知道：

(3.17)

对于，也就是最终时刻，由于之后没有其他序列连接了，则有：

(3.18)

然后就可以计算出W、U、b的偏导数了：

= (3.19)

= (3.20)

= (3.21)

### 3.1.5 梯度消失和梯度爆炸

就如在前文之中提到过的那样，RNN在NLP领域取得了广泛的应用，其原则是由于RNN可以识别一段文字中的相互关系。但是RNN的训练还存在着一些问题，很多学者发现RNN的模型难以训练，举个例子：

“红烧肉的做法是首先将五花肉洗净切块。锅中热油。加入生抽、老抽以及葱、姜、蒜。炒至所有的肉颜色均匀即可出锅。出锅前加鸡精。”

这样的一段话输入到RNN之中，然后利用RNN判断这段菜谱做的是什么菜也就是这段文字的主题，RNN可能给出的答案是鱼香肉丝，而真正的答案在那段描述中的第一句就已经指出是红烧肉。这是由于这道菜的名字出现在文章的第一句。而网络根据后面间隔较近的几句话得出了这道菜是鱼香肉丝，简单的RNN无法捕捉到间隔很远的信息，其本质原因可以由计算公式3.19分析，其中的意味着正在对一个向量取向量函数的偏导数，其结果是一个雅可比矩阵，其元素是该向量在各个方向上的导数。

这个雅可比矩阵的2范数，其绝对值的上确界是1，因为激活函数tanh会把所有的直映射到-1到1之间的范围里。



图3.7 tanh激活函数和偏导数

Fig.3.7 Tanh and derivative

可以看到激活函数tanh在两端的偏导数都是0，当愈加趋紧两端的时候，曲线的梯度变化越小，也可以说曲线越平坦。在神经网络里面，这就意味着对应的神经元是更加饱和的，这些神经元具有0梯度或是无穷小的梯度，在链式求导法则的过程中，由于是有多个神经元相连接，也就是多个无穷小的数或0相乘，网络模型中的梯度值则会迅速的收敛，在几个步长之后完全消失。直观上讲，这就意味着来自“遥远”的步骤的剃度贡献为0，这些值完全不会影响正在进行的学习，这也是为什么刚刚的例子中第一句即展示了菜谱的名字，却对结果没有任何影响。而这也就是著名的剃度消失现象。

那么同样的，还可以考虑如果雅可比矩阵的元素的值很大的情况。这样将会得到一个无穷大的数字，这就是梯度爆炸现象。而梯度爆炸现象和梯度消失现象统称为剃度渐变问题。其中梯度消失现象比较著名，以为梯度爆炸现象的出现，往往会伴随着一个无穷大的数，这样程序就会报错。另外使用阈值修剪法，则可以简单有效的解决剃度爆炸问题。在发生梯度消失的情况下，其过程往往无法轻易察觉，因此这些问题也阻碍了RNN的发展。

## 3.2 长短期记忆神经网络

由于RNN存在着不能对长期的内容形成记忆，也就是梯度消失现象，后来学者们逐步构造出了长短期记忆神经网络。长短期记忆神经网络(Long Short Term Memory, LSTM)是一种特殊的循环神经网络，这种网络结构可以学习长期的依赖关系，用来解决RNN中普遍存在的剃度消失问题。

大多数的RNN神经网络都是拥有复用的神经网络模块，常规的RNN服用的模块有着非常简单的结构。例如一个tanh层激活函数的神经单元。



图3.8 标准RNN结构中复用的模块

Fig.3.8 The repeating module in a standard RNN

LSTM也是基于这样的基础，但是服用的模块结构却不尽相同。与传统的RNN中的简单tanh激活函数的神经单元相比。LSTM的神经单元拥有四个神经单元结构，他们之间存在着相互作用的逻辑关系。

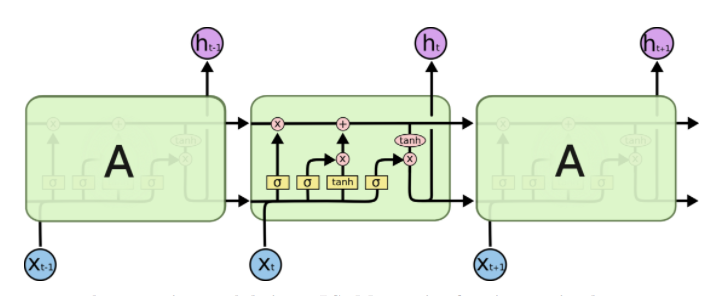


图3.9 标准LSTM结构中复用的模块

Fig.3.9 The repeating module in a standard LSTM

LSTM的核心思想是单元状态，水平的传递过程贯穿了整个结构的顶部。细胞单元利用门的概念来控制信息的通过与否，或者是增加、减少流通的信息。由于Sigmoid函数可以把输入映射到0和1这个区间，那么LSTM的神经单元就利用Sigmoid激活函数，来控制留通过的信息量。每一个的LSTM单元都包含有三个门的结构来控制单元状态，它们分别是输入门、遗忘门与输出门。

### 3.2.1 长短期记忆神经网络的门结构

LSTM中神经单元的第一步就是判断信息流之中的哪部分信息是需要丢弃的，而这一过程则需要“遗忘门”来完成。“遗忘门”根据 和的状态，利用Sigmoid函数，把单元状态中的每一个数字映射到0和1的区间上，0则代表完全遗忘，1代表完全保留。其对应的结构图如下：

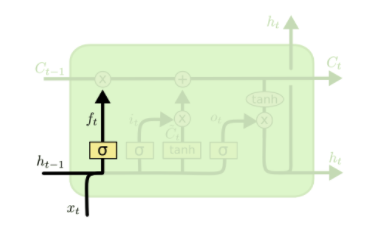


图3.10遗忘门结构

Fig.3.10 The construction of forget gate layer

在本文设计的系统之中，这种设计模式有着很好的效果。在语言模型之中，可以根据之前的文本信息来预测下一个单词或全文的主题。神经网络的细胞单元可能包含了当前文本的信息或主题，当网络接收到了新的内容的时候，“遗忘门”则可以帮助实现对过时信息的过滤,。

接下来的步骤则是决定应该在细胞状态中储存什么样的信息。这样的一个过程由两步组成，第一步是由一个有Sigmoid激活函数的“输入门”来决定要升级那些值，然后与此同时，由一个有tanh激活函数的神经单元产生需要被添加进状态之中的新的单元状态，第二步则是把这两部分融合起来，产生新的单元状态。

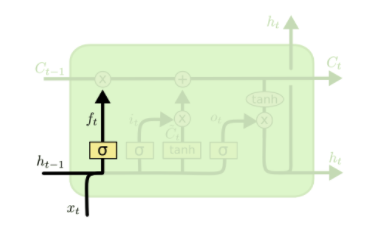


图3.11输入门结构

Fig.3.11 The construction of input gate layer

对应的在本文的方案之中设计的模型，这一结构可以实现神经单元状态里面的新主题添加功能，用以替换网络模型中需要被遗忘的旧主题。

最后，则是该结构的输出过程，利用Sigmoid函数可以决定单元状态中的那些部分用来输出，然后再通过一个tanh层进行过滤，通过将tanh的映射为0到1之间的结果与Sigmoid门的输出相成，可以得到最终的输出，这样的结构被称为“输出门”。

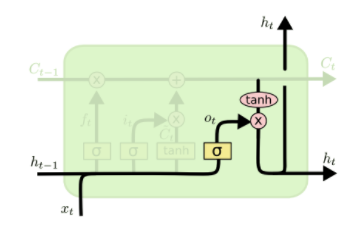


图3.12输出门结构

Fig.3.12 The construction of output gate layer

在语言模型之中，输出层可以用来控制最后的输出。由于本文用到的神经网络模型中存储的是文本数据的主题，而最终的输出却是一个情感倾向，这就需要这种门的结构。

### 3.2.2 长短期记忆神经网络模型的建立

长短期记忆神经网络的建立有很多种方式，并且各种不同的变种网络结构在不同的应用场景都取得了不错的效果。在本系统中，对于某一句子中的第t个词，将其对应的词向量模型作为输入，其前一时刻，也就是上一句话的输出，以及上一个神经网络的状态，神经网络的输出是。其中和都用零向量进行初始化。

无论是电影评论还是产品评论，最终的情感倾向都是依赖于整段文字。这也就意味着在时间戳的时刻，神经网络不仅依赖于先前的文字，还依赖于t时刻之后的文字。采取双向LSTM(Bidirectional LSTM, Bi-LSTM)的连接结构则可以考虑到双向的序列处理[40]。该结构采取两个LSTM网络并行的方式，其中一个神经网络进行正序处理，另一个进行倒序处理，则在时刻的输出可以代表前向LSTM与后向LSTM的连接。

在语句层面，由于每一句具有一个时间戳，因此每一个句子就是在每一个时间时刻网路处理的最小单位。因此如下图所示的两个LSTM网络分别接受从第一个和最后一个语句的词向量。在输出层面，则将两个网络结构的输出连接起来并输入到最终的softmax层，输出每个句子的情感分布。

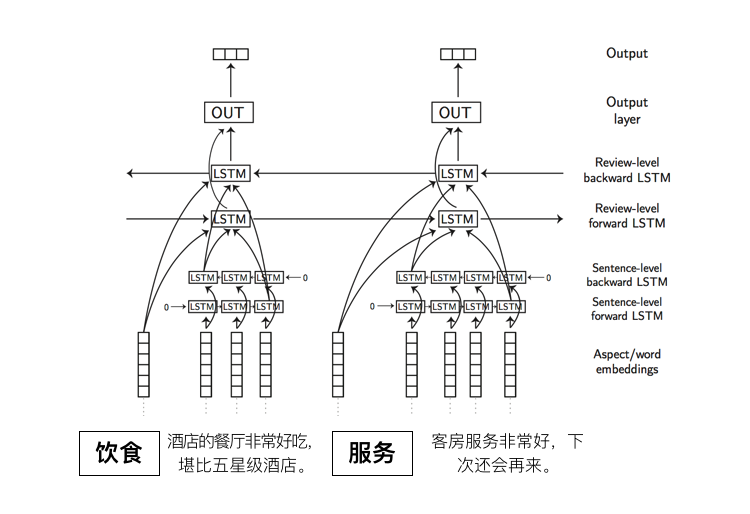


图3.13 双向LSTNM情感分析神经网络模型

Fig.3.13 The model of Bi-LSTM neural network

## 3.3 模型效果分析

在验证算法的时候，不应该采用本文设计的方案中将要使用的的数据集，因为这样有可能导致可能存在的过拟合现象。在验证双向LSTM算法的时候，Sebastian Ruder博士等人通过利用SemEval-2016的评论集进行了验证[41]。

该数据集包含有餐厅、酒店、笔记本电脑、手机和相机五个方面的评论数据，包含英语、西班牙语、法语、俄语、荷兰语、土耳其语、中文以及阿拉伯语在内的八种语言。共记11个领域的情感分析任务。每个数据集包含300-400条评论，1250-6000个句子。每个语句都标注了一个或者多个领域的情感价值。本文对语料数据集进行了处理，分别对不同语言的不同领域的数据应用几种主流的常见的深度学习模型，以来观测他们的准确度。

由于比较的算法的神经网络模型不尽相同，本文并没有强制要求CNN和RNN有相同的层数和神经单元数，更没有要求所有的模型均采用相同的激活函数，所有算法都在大致相近的时间之内达到神经网络稳定收敛的状态，这样结果可以看作在固定时间内该网络模型的最佳表现。本文采取固定所有神经网络模型的输入和输出维度的方式。对于输入，不同的神经网络模型均采用了相同的方法进行了归一化处理，所有的神经网络模型均采用相同的梯度下降法，每一个神经网络模型都采用相同的正则化系数，范数正则的方式。唯一不同之处在于CNN由于网络层数很多，可以自由应应用droupout法进行正则处理，而几种LSTM神经网络模型由于本身结构的结构限制，没有采用相同的droupout率，关于droupout正则的部分，会在后面应用的部分有详细地介绍和分析，在这里不重复赘述。通过做到以上对比试验，本文将分析几种不同的神经网络模型在情感分析上的表现。

表3.1 不同网络模型的准确率

Table 3.1 The results of different models

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Language | Domain | Best | CNN | LSTM | Bi-LSTM |
| English | Restaurants | 88 | 82.7 | 85.3 | 88 |
| Spanish | Restaurants | 83.9 | 70.9 | 76.6 | 83.9 |
| French | Restaurants | 78.6 | 77.1 | 77.3 | 78.6 |
| Russian | Restaurants | 77.6 | 72.2 | 77.6 | 74.1 |
| Dutch | Restaurants | 79.8 | 76.1 | 73.2 | 79.8 |
| Turkish | Restaurants | 82.7 | 79.9 | 79.8 | 82.7 |
| Arabic | Hotels | 82.7 | 81 | 78.1 | 82.7 |
| English | Laptops | 81.8 | 80 | 80.2 | 81.8 |
| Dutch | Phones | 82.3 | 80.2 | 73.2 | 82.3 |
| Chinese | Cameras | 80.8 | 78.2 | 78.9 | 80.8 |
| Chinese | Phones | 73.1 | 72.4 | 73.1 | 72.7 |

以上结果即为在不同的数据集上利用不同的模型，采用相同的正则化系数，梯度，学习率和dropout规则的情况下，进行控制变量的结果，与原文的结果虽然略微有些不同，但是大抵是一致的。至于不同的原因，是因为本文与原模型采取了部分的不一致的结果。

从结果上可以知道，在情感分析领域，通过分层结构的双向长短期记忆神经网络有着比其他模型更好的结果。在模型数量不足或是人工设计不足的情况下，双向长短期记忆神经网络的表现更是超过了大多数同类的技术水平。

## 3.4小结

在本章节，本文从数学角度剖析了循环神经网络和它的改进算法，长短期记忆神经网络。并且从结构上对长短期记忆神经网络的门结构做了深度的解读，以及它们的设计理念。在传统长短期记忆网络的基础上，本文根据应用场景的特点，提出了双向长短期记忆神经网络的方案，并且在不同文本数据上进行不同算法的比对实验，得出了在情感分析领域，双向长短期记忆神经网络的优越性以及可行性。

# 第4章 中文情感分析方案的设计

本章将就如何将介绍文本信息是如何转变为数字化信息的过程进行讨论。先分析了中文在情感分析任务中面临的分词问题，并对比几种常用的分词方案。对于下一步将分词结果转变为数字化的步骤，本篇将介绍一种新型的词向量化方式，并实际验证几种不同词向量化方式的效果，并作出本文方案采取的选择。在本片的最后，我们还将讨论几种常用的深度学习框架，并选择适合本文的框架。

## 4.1 分词系统

在中文乃至亚洲自然语言处理领域，一直有一个独有的问题阻碍着中文自然语言处理的发展，那就是分词。分词系统在英语、法语和德语等拉丁语系中是不存在的，因为这些语言词和词之间有着明确的分隔符，也就是空格的存在，可以对这些语言的模型执行进行处理。而以中文为主的一些亚洲语言，还有日语、汉语、蒙古语是没有这样的分隔符的。以中文为例，中文自然语言处理的最小单位是词，而中文在存储的过程之中的最小单位却是汉字。那么如何将这些独立的汉字分割成词就是中文自然语言处理的一个先行工作。由于汉语具有可以用很少的字表达很复杂观点的特点，因此分词也是从古至今一直在进行的一项研究，古代诸子百家对相同古文的不同注解就是最初的分词模型，而注解的意义就是消除歧义，而歧义则是不同的分词造成的产物。

分词系统在自然语言处理的过之中至关重要，因为分词的结果直接影响算法的好坏，进而会影响整个系统设计的成功与否。中文的分词算法研究始于二十世纪八十年代，北京航空航天大大学计算机系的梁南元教授设计的CDWS(Chinese Distinguishing Word System)书面汉语分词系统拉开了国内对中文分词的研究序幕[42]，该系统也是第一个较为成功的中文分词系统，具有开创性的引领意义。这种方法也被称为基于词典的方法，其实现原理就是按照字的前后顺序从左到右逐渐扫描，遇到词典里面的词语就标记出来，遇到复合的词语就匹配成最长的词典中的词语，遇到字典中没有的字符串则进行拆分。例如“东北大学是一座拥有着悠久历史的现代化大学。”按照这种方法逐字扫描的方式其分词就过就是“东北大学／是／一座／拥有／悠久／历史／的／现代化／大学”。这种方法可以用来解决大概百分之八十的简单中文分词问题，如果遇到复杂的语义信息，这种方法就会无能为力，例如“南京市长江大桥”会被分词为“南京／市长／江大桥”，这样的分词方式，凡是长江流过的城市的市长都会变成江大桥了，另外“发展中国家”也会被分词为“发展／中国／家”。

在梁南元教授工作的基础上，此后的二十年之内很多的改进算法分别问世。这些算法有正相匹配法、逆向匹配法、最长匹配法与最短匹配法，还有一些算法试图把这些改进算法融合起来。但是这些的本质没有发生变化，还是词典法。词典法思路简单，易于实现，但是其分词效果好坏十分依赖于词典的质量和数量，另外对于一些新兴的网络流行语反应不敏感，词典中没有收录则会发生分词失败的现象。清华大学电子工程系的郭进博士是把统计学方法引入到中文分词系统的第一人，并且取得了巨大的成功[43]。统计学习的分词方法的思路是利用马尔可夫条件随机场和动态规划的思想，把所有可能的分词方式及其概率都罗列出来，则概率最大的模型就是最有可能想到得到的分词结果。基于统计学的自然语言处理中文分词法经过不断的完善，逐步的发展来的算法之中比较成功的有最大熵分词模型和N-Gram元分词法。所谓N-Gram就是基于N-1阶马尔可夫模型，认为文本中的第i个词仅与前面N个词有关，与更前面的词无关，更与后面的词无关。基于统计学的分词方法虽然可以有效的消除歧义、可以有效的识别出未记录在词典中的词语，但是其有着运算了巨大的缺点特别是在N-Gram中的N取一个较大值的时候，分词算法的运算了就可以说得上十分巨大了。

如何既可以做到词典法的快速又可以兼容统计学方法的消除歧义呢。在这里，本文的方案则是采用的方法是将二者结合起来的方法，力争做到博采众家之所长。IK中文分词器是一款开源的中文分词工具包。设计之初，它是基于Apache基金会下Lucene项目为主题，结合词典分词法和文法分析算法的中文分词组建，但随着不断的发展完善，IK分词已经变成了一个独立的拥有“争相迭代最细粒度切分算法”的JAVA组件，可以实现分词功能和简单的歧义消除。而ElasticSearch是另外一个基于Lucene的搜索云服务器，它的特点是搜索速度非常快，可以横向扩展，十分适合对海量的数据进行统计工作和分类聚类工作。正是由于ElasticSearch的这个特点，基于ElasticEearch的IK分词器可以对常规的词典分析结果进一步进行统计学分析，而且基于云端的设计可以通过Http的方法通过json格式进行文本信息传递，方式简单，易于实现，故现在被很多公司所采用。

为了验证本文中的方法的效果，采用语言联盟(Linguistic Data Consortiu-m, LDC)的分词条目，该组织1992年在宾夕法尼亚大学由几名学生组织创立。该网站提供了可供分词的语料库，共计四万多条语句，并附有分词之后的结果。本文在相同的服务器上分别运行几种常见的中文分词器，对于统计结果与标准答案作为比较，分析该算法的分词准确度，以及运行时间。对于有一些文本，语料库中的分词是粗粒度分词，而算法执行为细粒度分词，例如预料集中的“北京大学”被分词算法分割为“北京／大学”，那么本文也依然认为分词算法是准确的。另外由于部分算法是机遇云解决方案的，另外一些算法则是本地计算，为了统一标准本文统一使用阿里云16核处理器、64GB内存的阿里ECS云服务器。

表4.1 不同分词器的分词结果

Table 4.1 The results of different analyzers

|  |  |  |  |
| --- | --- | --- | --- |
| Analyzer | Language | Accuracy/Percent | Time/Min |
| CJK | 汉语 | 32.1 | 22.3 |
| IK | 亚洲语言 | 87.2 | 61.1 |
| paoding | 汉语 | 78.6 | 27.7 |
| MM | 亚洲语言 | 77.6 | 43.1 |
| imdict | 亚洲语言 | 78.1 | 25.5 |
| IK+ES | 亚洲语言 | 93.5 | 26.5 |

通过对比结果可以知道CJK算法较为简单，分词结果也不是很理想，但是其特点是分词速度最快，imdict和MM由于受到自带字典数量较小的限制，分脆准确率不是很理想，MM分词则是最为中庸，而IK分词取得了非常好的分词效果，但是分词时间过长。采用基于ElasticSearch的IK分词对IK算法有进一步提升，其主要原因是减少了对磁盘的I/O操作，故速度有所提升。

## 4.2 词向量化

在前面的工作中，已经完成了对数据集中的数据进行文本分词的工作，现在文本都是以词的形式存在着。计算机虽然可以对文本信息进行显示，但是如果要涉及到计算机的计算就无能为力了。众所周知，计算机是通过二进制的方式进行计算的，因此下一步工作就是把此刻分散的文本形式的词转化成计算机可以识别的数字化表示。

4.2.1 Word2Vec

传统的词向量方式是由Hinton教授提出的one-hot表示方式。其设计理念是利用N位寄存器的状态分别对N个状态进行编码。举个例子如果想要对“东北大学信息学院院长”进行编码。这里有三个特征属性：东北大学，信息学院，院长。

假设对应的词典如下：

学校：[“大连理工大学”，“东北大学”，“辽宁大学”，“大连海事大学”]

学院：[“计算机学院”，“数学学院”，“信息学院”，“冶金学院”]

职务：[“院长”，“党委书记”，“所长”，“副所长”，“办公室主任”]

那么此时学校是四维的，学院也是四维对，职务是五维的。对应上文的样本[“东北大学”，“信息学院”，“院长”]中的“东北大学”对应的编码就是[0，1，0，0]，“信息学院”对应的编码是[0，0，1，0]，院长对应的编码则是[1，0，0，0，0]。则最后的“东北大学信息学院院长”完整的编码就是[0，1，0，0，0，0，1，0，1，0，0，0，0]。

虽然已经成功的将文字样本转化成了离散化的数字信号，可以传入计算机进行计算，但是把这种方法应用于大规模文本显然是不现实的，因为大规模文本意味着词典容量也会很大，特别是针对于情感分析的文本其容量往往是十分巨大的，而采用one-hot的表示方式则意味着这段话的向量表示中有很多位是0，这会带来几个显而易见的问题：第一、输入维度过大，会导致机器学习中常见的问题，维度灾难。第二，由于向量中大多数元素都是0，也就意味着后续的特征矩阵十分稀疏，稀疏矩阵在运算过程极易引发各种问题，不仅是数学上的无穷大，还包括计算机编程语言中的空指针问题。

针对于这个问题，谷歌的研究员Tomas Mikolov带领的团队在2013年发表的两篇论文[44-45]开创了新的时代。通过深度神经网络的方式可以把一段语句文本映射到固定维的空间上，变成形如[0.123, 0.564, 0876, … , 0.231]的固定维度向量。而去他们的进一步研究表明，这些向量之间符合现行运算规则，例如国王的词向量减去男人的词相连加上女人的词向量与王后的词向量十分接近，这里采用的词向量计算方式就是余弦定理。同样德国的词向量减去柏林的词向量加上法国的词向量与巴黎的词向量十分接近。在下图所示的词向量空间之中，利用PCA降维的方式把观测各个国家和首都的词向量坐标，可以看到这一现象。

他们的工作通过简单的神经网络结构把文本映射到固定维度的空间上取得了十分重大的意义，可以说把自然语言处理带上了一个新的高度。这样既可以避免由one-hot表示方式引起的维度灾难，还可以利用固定维度的词向量对文本进行进一步的处理。

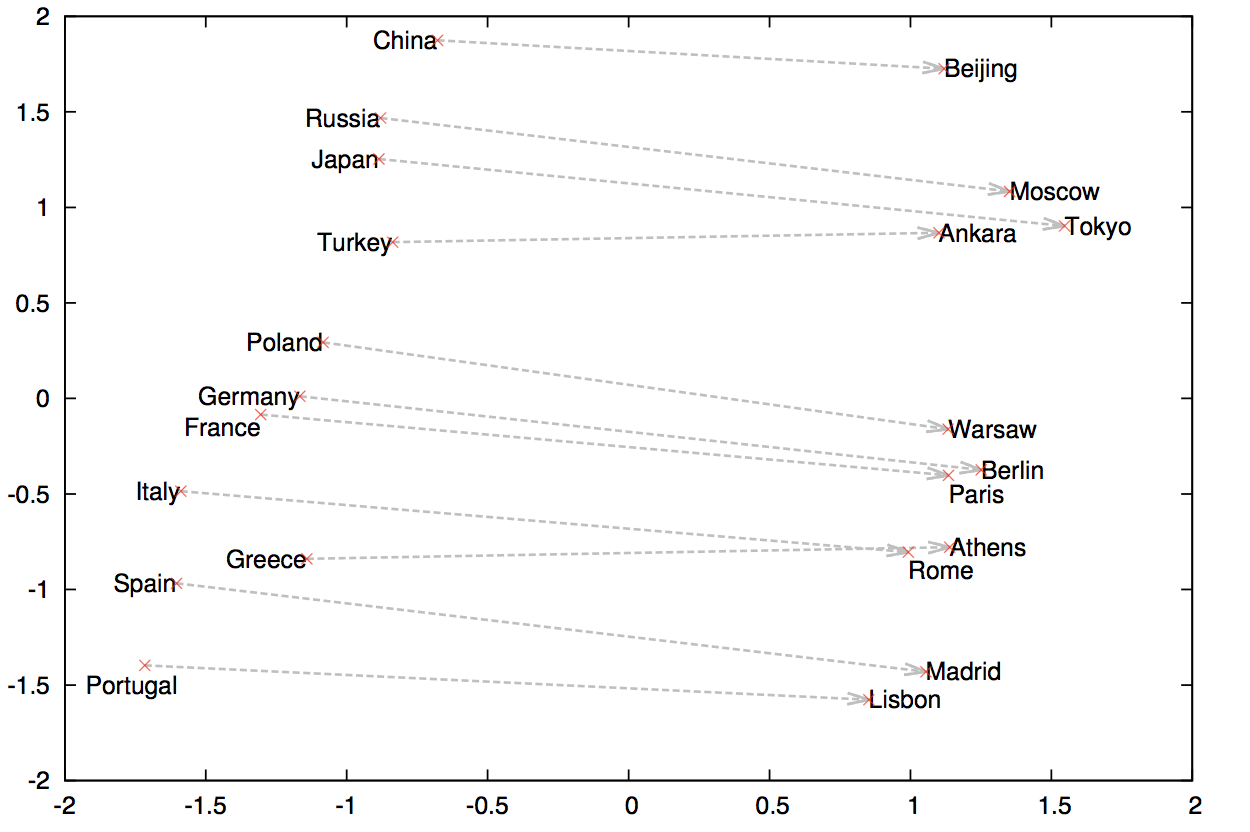


图4.1 利用PCA降维观测国家与首都的词向量关系

Fig.4.1 Country and Capital Vectors Projected by PCA

4.2.2 两种语言网络模型结构

谷歌的自然语言处理团队在对比了反馈语言网络模型(Feedforward Neural Net Language Model, NNLM )、循环神经网络语言模型(Recurrent Neural Net Language Model, RNNLM)和并行神经网络等模型的基础上，提出了两种对数线性模型，连续词袋(Continuous Bag-of-Words Model, CBOW)模型和语法转移(Continuous Skip0gram Model)模型。

两种模型的不同之处在于CBOW模型利用同一时间戳内第个词语作为网络的输出，的前后上下文词语作为网络的输入，通过上下文预测中间的词语。而Skip-gram模型通过把第个词语作为输入，该词语的前后上下文作为网络的输出，通过某词语预测其上下文词语，完成整个网络的训练。

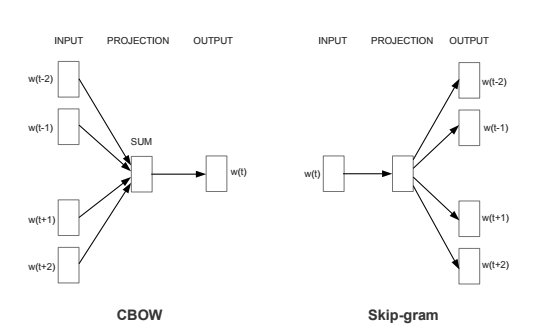


图4.2 两种word2vev模型结构

Fig.4.2 Two word2vec model architectures

利用谷歌提供的语法语义文本集，本文同样也对这些语言模型结构进行了性能测试，该文本集全名Semantic-Syntactic Word Relationship。由谷歌提供，维护了对两万余条语句的语义和语法的标注，本文对不同的模型仍采用论文中的固定维变量，但是本文没有采用原文中的640维，而是100维，是因为大维度更适合于英语、法语、德语和西班牙语等拉丁语系，而中文与这些拉丁语系等语言还有很大的不同。

表4.2 不同语言模型网络的语法语义准确度

Table 4.2 The results of different model architecture

|  |  |  |
| --- | --- | --- |
| Model Architecture | Semantic Accuracy | Syntactic Accuracy |
| RNNLM | 8.3 | 31.2 |
| NNLM | 21.1 | 56 |
| CBOW | 23.8 | 62 |
| Skip-gram | 51.8 | 59.6 |

经过上述的分析，可以很容易得出虽然CBOW模型在语法类的任务上有着最好的表现，但是综合表现则是Skip-gram模型取得最好的结果。因此在本文设计的方案中将采用Skip-gram模型作为最终的的词向量训练模型网络。

## 4.3 双向长短期记忆神经网络

在上一章内容的最后，提出了介绍了一种LSTM的改进方案，双向长短期记忆神经网络(Bi-LSTM)，并给出了它的结果，但是并没有介绍它的具体结构，在这一小节将对这一神经网络进行进一步的分析。

本文把经过Skip-gram模型训练之后的词向量作为Bi-LSTM的输入，表示维语言向量中的第个词。每一个词向量代表一个需要进行情感分析的句子。而模型结构的主体是由两个常规的LSTM组成的，每个神经单元都是常规的输入门、输出门和遗忘门组成的。本文采用的模型结构中的两个LSTM网络分别对序列数据进行正序处理和倒序处理，在给定的时间戳时刻把这两个前向LSTM和后向LSTM连接起来。

在模型确定之后，并且所有的门结构都采用Sigmoid激活函数，针对神经单元采用不同激活函数的表现有怎样的差异进行对比，本文选择三种常规的主流激活函数(Sigmoid，Tanh，ReLU)，实验得到的迭代曲线如下：

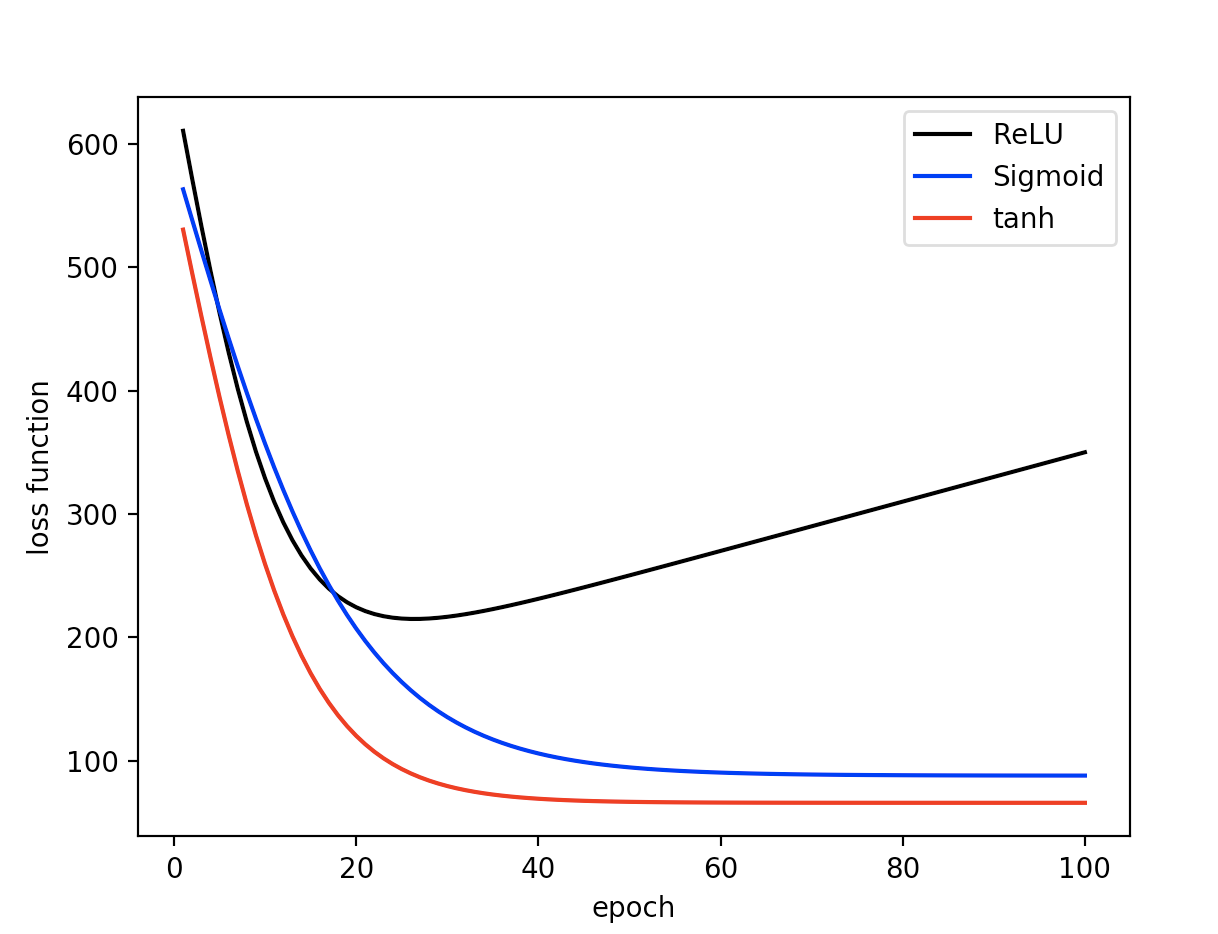


图4.3 不同激活函数获得代价函数变化曲线

Fig.4.3 The loss function of different activation functions

从图中可以看出，线性校正单元作为激活函数，在LSTM中并不能取得很好的效果，甚至代价函数最后是发散的，而采用正切激活函数tanh的效果要好于S型激活函数Sigmoid。

对于分类器的选择，因为本文的最终结果是一个二分类问题，即正向类和负向类。常规的分类器的选择有逻辑回归(Logistic Regression, LR)和支持向量机(Support Vector Machine, SVM) 的方法，这两种分类方法都已经被提出很久并且取得了很多成功的应用。其中逻辑回归是基于统计学的最大熵模型，而SVM则是基于空间距离的最大间隔的方式，SVM采用核函数的方式，有着严格的数学推导过程，保证所有的点均匀的分布在分割超平面的两侧。然而逻辑回归通常不采用核函数的方式，其表达形式可以求导，十分易于反向传递。本文针对这两种不同的分类起，使用测试集的数据进行验证。



图4.4 不同分类器的准确度

Fig.4.4 The accuracy of different classifiers

从图像上可以知道，这两种分类器的表现其实殊途同归，没有本质上的差异，但是由于逻辑回归的结构简单，易于反向传递，而且在小样本上的表现要优于支持向量机，因此本文的分类器将采用逻辑回归做最终的分类器函数。

## 4.4 深度学习框架选择

在确定了本系统的所有步骤和工作流程，并对每一个方案的不同实施方案进行了比较之后，接下来要选择的是搭载深度学习的模型，现在比较流行的模型有TensorFlow、Pytorch、Caffe、以及MXNeT，以及在这些API基础上封装的Keras、TensorLayer和TFLearn等,下面本文对这些框架进行简介。

1. TensorFlow：TensorFlow是由谷歌公司谷歌大脑团队（录属于Google机器智能研究机构）提出的，采用数据流图，用于数值计算的开源软件库，广泛的应用于机器学习和深度神经网络方面的研究。该框架 具有高度的灵活性，可以有开发者自由的定义图的结构以及驱动计算的内部循环。同时支持CPU和GPU的加速运算，具有很高的移植性。并且具有自动求微分的能力，可以加速梯度下降的运算过程。另外支持多种语言，可以利用C++和Python语言直接调用。

2. Pytorch：Pytorch拥有着比TensorFlow更好的加速运算，该框架由Facebook公司人工智能实验室Yann LeCun教授所带领团队提出，结合了Torch7高效灵活的GPU加速后端和直观的Python前段。具有便携方便，代码易读和广泛支持深度学习诸多模型的特点。在推出以后迅速得到了诸多开发者和研究人员的青睐。但是缺点是由于其推出时间晚于TensorFlow，文档和参考内容均少于Tensor-Flow，而近一段时间Pytorch的应用和文档也逐步增多，其资源匮乏的问题得到了一定程度上的缓解。

3. Caffe：Caffe是人工智能届著名的科学家贾扬清就读于加州大学伯克利分校时的作品，在他加入谷歌以后得到了进一步的完善。它的问世时间最早，可以说是最早商业化的主流工业级深度学习工具包。Caffe框架基于C++编程语言编写，支持Matlab、Python和C++接口。目前被广泛的应用于计算机视觉领域，并且取得了很多的成功应用以及商业部署。但是由于其一流框架结构中的一些问题，Caffe对于RNN和LSTM等循环网络和语言模型的支持比较薄弱。

4. Caffe2：Caffe2则是贾扬清博士离开谷歌加入FaceBook之后对Caffe提出的进一步完善。Caffe2相比于Caffe1而言，更加注重模块化，在移动端吧、大规模部署上表现卓越。同前面两种框架一样，采用C++ Eigen库，支持ARM架构。在延续了Caffe对卷积神经网络的良好支持的基础上，增加了对循环伸神经网络诸如RNN、LSTM的支持。延续了处理视觉类型问题上的一贯优秀表现，增加了对自然语言处理、手写识别和时序预测的支持。但是总体来讲，Caffe2问世时间太短，并没有很多的应用案例和实践文档，目前Caffe向Caffe2的迁移还很缓慢。

关于几个框架的优点和缺点由于评价十分主观，无法妄下结论。但是学者Mitch De Felice在2017年5约公布的一份数据似乎可以说明。

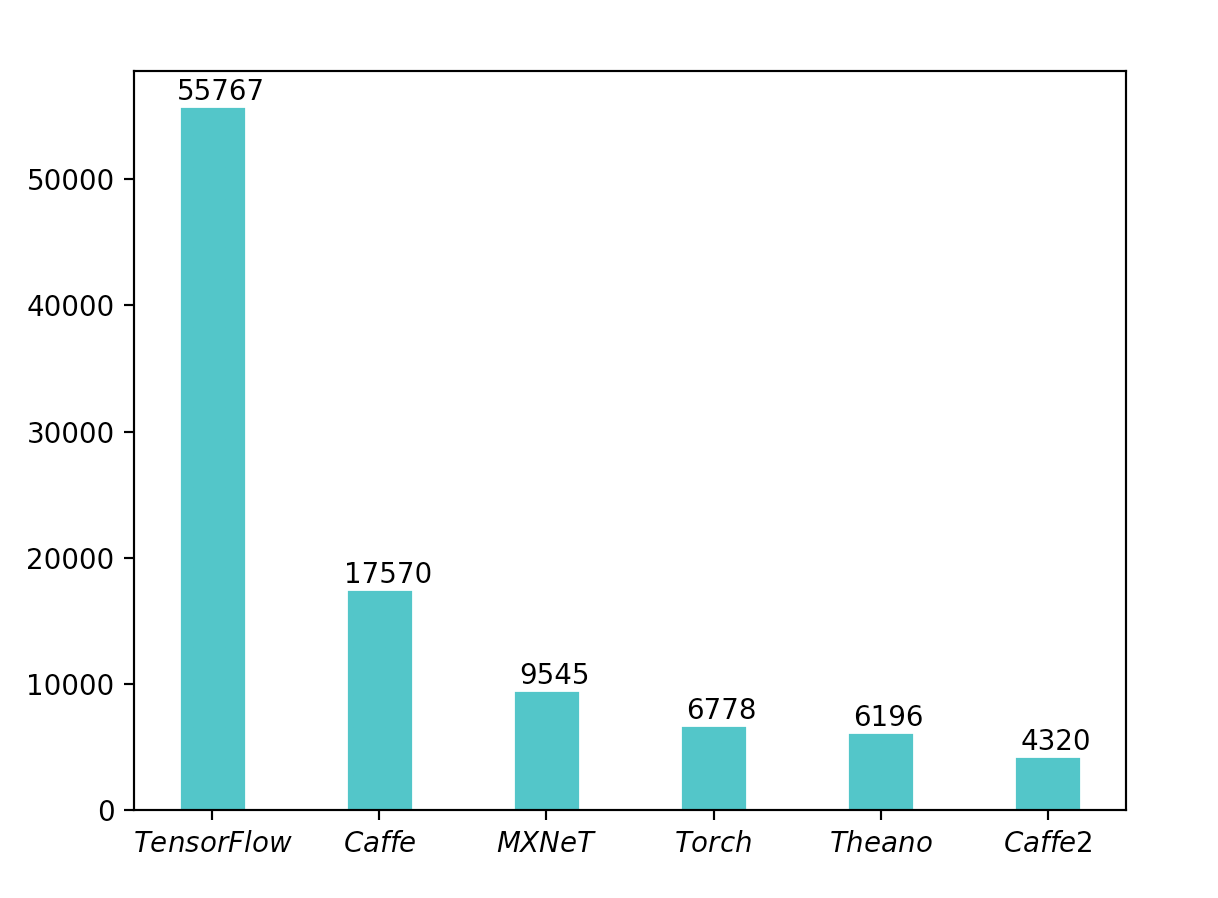


图4.5 不同机器学习框架的热度

Fig.4.5 The popularity of different frameworks of deep learning

该数据是机遇对Github上所有仓库的检索，分析几种主流框架的使用人数，并用此来体现热度。另外，值得说明的是，途中的Theano已经被官方宣布不再维护，因此本文没有对该框架进行介绍。在上文中提到的Pytorch由于问世较晚，在Mitch De Felice统计其他框架热度的时候，并未奖Pytorch纳入对比范围。

综上可以知道，TensorFlow是目前最受欢迎的深度学习框架，因此本文也最终确定将TensorFlow作为本文代码的框架选择。

## 4.5小结

本章完整的介绍了本文涉及的情感分析方案的组成流程：第一部分词器，第二部分词向量化网络，第三部分双向长短期记忆神经网络。在分词器的部分本文对比了各种分词器的优劣性，分析并采用了基于云上的ElasticSearch与IK分词器结合的方式。在第二部分，本文介绍了Word2Vec技术的提出、发展与意义，在复现了原文的集中网络模型后又对其优点与缺点进行了分析，并最终选择了Skip-gram模型。在第三部分，本文更为详细的介绍了双向长短期记忆神经网络，并且针对于网络结构中的激活函数的输入维度、网络工作流程、激活函数的选取、分类器的选取都做了系统的比较，并作出了最终的选择以及选择原因。最后一部分则是详细介绍了主流深度学习框架并分析了优劣点，做出了最终的选择是TensorFlow作为代码的编程框架。

# 第5章 参数优化与训练过程

在完整的了解了本文的方案设计模型与工作流程之后，接下来就要介绍在训练过程中的一些参数的选择，以及系统的调优，还有系统的训练方式。

首先，本文第一个要讨论的就是双向长短期记忆神经网络(Bi-LSTM)当中的初始化参数的状态设定。

第二，本文将对Skip-gram模型的结构与参数选择进行探究。

然后，本文将对Bi-LSTM模型的结构参数选择进行对比试验，分析最种结果。

随后，本文将对训练过程中的包括梯度下降法和学习率之类的参数做出定量对照实验，分析不同的选值。

最后，本文将对神经网络训练过程中的过拟合现象进行进一步分析，并介绍了Dropout算法，并对Dropout算法和其他算法做对比试验，分析不同正则方式等效果。

## 5.1 初始化状态

首先要进行的是对网络模型初始化方案等探讨，所谓网络初始化，就是指在训练开始之前，神经网络结构模型、权重等参数的赋值，常规的做法有零向量初始化，向量初始化以及噪声初始化和一些改进的方案。

零向量初始化的做法是在为开始训练模型的时候，默认所有的参数的初始化状态都是零。然后在训练的过程之中参数的随着网络模型的变化而逐步发生改变。向量初始化则是将初始化状态设置为一个固定的向量，该方法一般作为二阶系统的第二个神经网络模型的初始化参数设置方案，因为已经大概知道神经网络最终的参数，此法可以加速网络收敛速度。但是向量初始化方案很容易造成神经网络的过拟合，一般只有在足够的序列数据或有较少的状态重置的情况下使用。同理，如果网络参数变化量较小，最终收敛于零向量附近，那么零向量初始化的方案也有很大的可能会过拟合。一个简单的解决方案是使初始化的状态噪声化，也就是使用随机噪声数据作为初始化的向量，常用的方案有高斯噪声和，甚至还有一些改进的方案可以根据反向传播的代价函数的大小来初始化状态噪声的大小。

那么本文就对这几种不同的初始化状态进行测试，用来判断几种初始化方案的优劣。不同于前面章节的验证方案，前面章节的验证方案是为了验证某种结构模型下不同变化的差异，本文采用的是第三方的数据集，因为第三方数据集都是被精心维护并做过数据清洗工作的，因此其结果具有一般意义上的普适性。而在本章节本文将全部采用在第二章介绍的方法采集到的数据集，只是用于本方案，并不具有一般的普适性。

本文改写了TensorFlow中的tf.zeros函数，可以产生初始化的零向量。本文分别比对了几种不同初始化状态方式的平均代价函数，其结果如下：

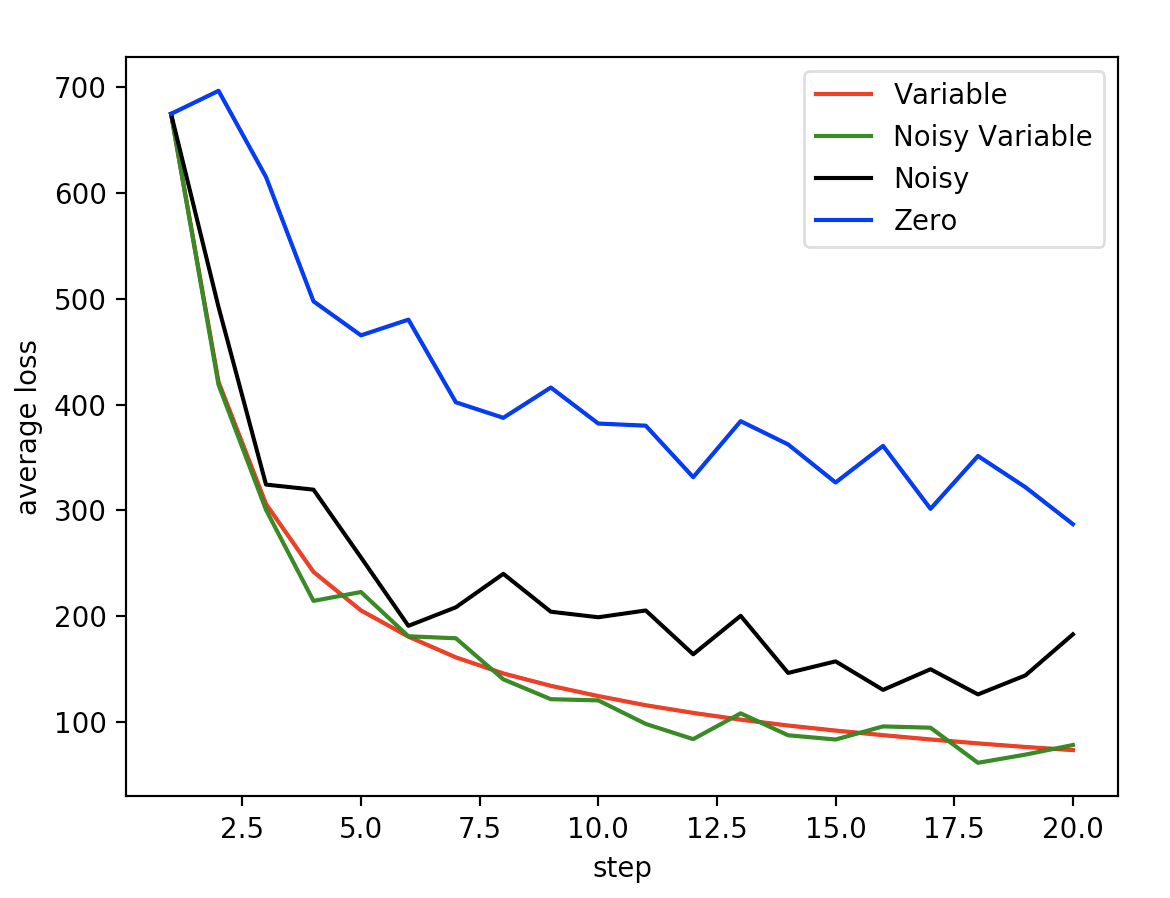


图5.1 不同初始化方案下训练集平均代价函数随着时间的变化

Fig.5.1 Average loss function from different kinds of initial states on training data

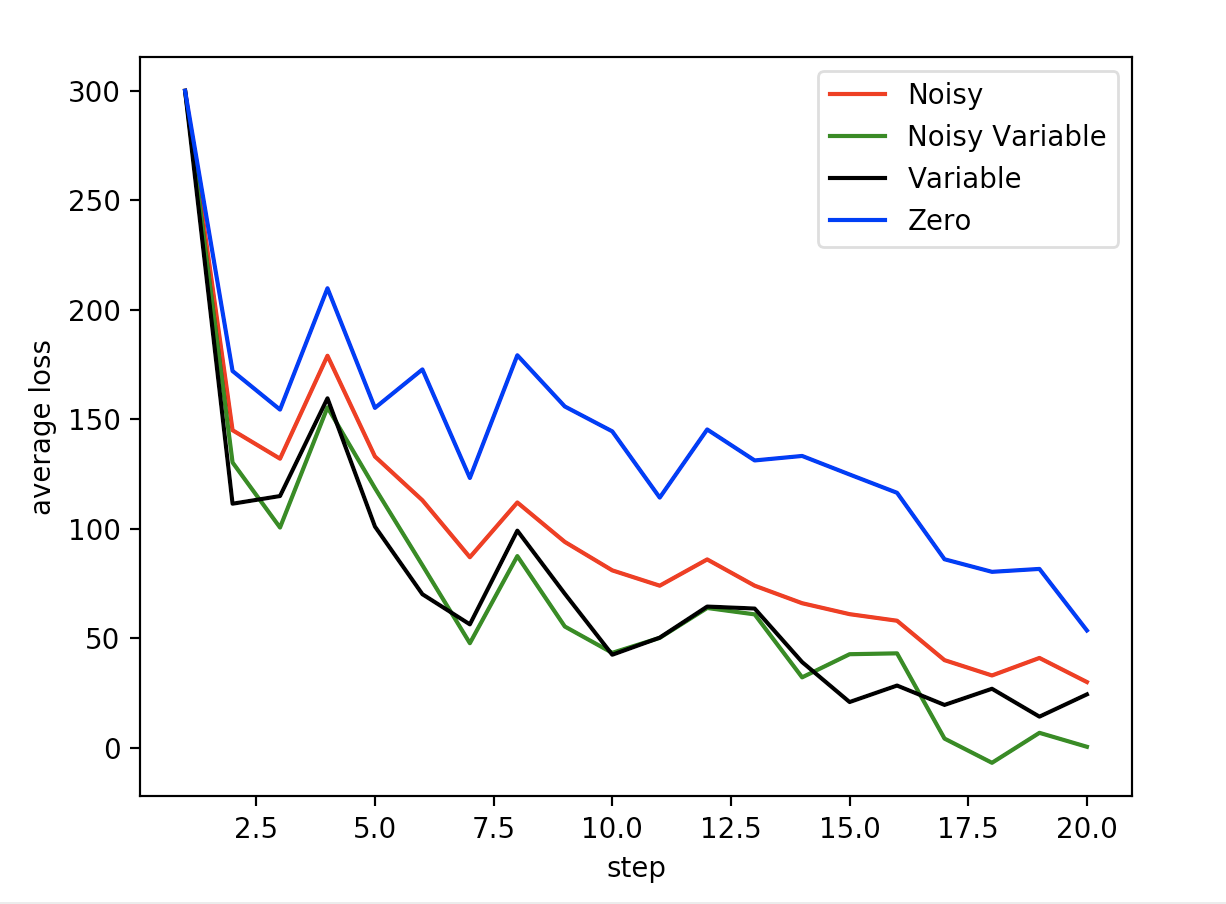


图5.2 不同初始化方案下训交叉验证集平均代价函数随着时间的变化

Fig.5.2 Average loss function from different kinds of initial states on validating data

上图为几种不同的初始化方式在训练集和交叉验证集上的表现，本文根据随时间不断迭代的参数的平均代价函数曲线可以得到以下观察：

1. 所有非零初始化状态逗可以加快训练过程，并且可以改善泛化性能。
2. 变量初始化比随机噪声初始化的方案更加有效，训练速度更快且性能更好。
3. 在变量初始化的基础上增加噪声的方案对于性能收效甚微，其表现对于神经网络性能接近于变量初始化方案。

综上考虑，本文采用向量初始化状态的方案来初始化神经网络参数。

## 5.2 Skip-gram模型参数选择

在前面的介绍内容之中，本文通过对比几种不同的词向量方法的结果，最终确定了Skip-model作为最终的词向量模型。确定了Skip-gram模型之后还有几个重要的参数需要进行探讨，例如词向量的维度，负采样率的设置，以及对高频词的采样率。这些参数的设置对于网络的效果，以及词向量的质量有着很大的影响，接下来本文一个一个的比较，来判断这些参数的选择。

5.2.1 负采样率

负采样方法最初的雏形是由Gutmann 和Hyvarinen提出的噪声对抗算法，后来被Mnih和The成功的应用在自然语言处理建模领域[46-47]。负采样算法通过采用随机负采样的方式，代替原有的霍夫曼树结构，目的就是加速训练过程，提高词向量准确度。由于Skip-gram模型是根据某个词预测其上下文的词，那么这些词就是正样本，其他的词就是负样本。

但是，在一个词典中，不同的词出现在语料中的次数有多有少，也就是出现概率有高有低，负采样算法就是通过建立一种映射关系，使得高频词有着更高的被采样频率的算法。映射的双方是各个词语出现的频率和在[0, 1]区间内被等分成段空间，通过对等距离分割区间上随机产生的随机数对应着非等距离分割的样本词。此时的优化目标就是：

(5.1)

此时，对于等距离分割区间数量k的选择就显得十分重要，在原始文献中，作者给出的值范围是应用在大型样本数据集上是2-5，小型样本文本数据集上是5-20，由于本文只是针对某几种场景的小型样本数据集，本文的样本集的数量和质量都称不上大，因此对5-20对的不同取值进行试验。

另外对于可能存在的对于词，它的负样本中包含它自己本身的情况，原文没有做出讨论，本文在设计的方案中采用重新执行产生[0, 1]随机数的方法，重新执行整个流程。

还有一种比较常用的做法是从CBOW模型中引用过来的，在CBOW模型中，往往选择窗口大小作为负采样数量，但是本文认为通过对比的做法可以更好的提高综合性能，可以做出选择，而不是固定负采样数量。

表5.1 不同负采样数量的结果

Table 4.1 The results of different negative sampling selections

|  |  |  |
| --- | --- | --- |
| K | Time [min] | accuracy [%] |
| 5 | 38 | 33.6 |
| 6 | 44 | 43.1 |
| 7 | 51 | 51 |
| 8 | 58 | 59.5 |
| 9 | 62 | 66.3 |
| 10 | 67 | 68.2 |
| 11 | 72 | 68.9 |
| 12 | 89 | 66.3 |
| 13 | 107 | 69.2 |
| 14 | 144 | 69.1 |
| 15 | 189 | 68.9 |

从表中可以得知值在不断的增大过程中，准去率和训练实践都有所提升，在10附近算法的准确率达到较高水准，随着的继续增大，训练时间需要更多，但是对于算法准确率的提升却没有达到相应的高度。故本文认为在本方案中取。

5.2.2 降采样阈值选择

关于采样还有一点值得注意的就是对于采样词语的抛弃。我们知道在一段话中，起到关键作用，也就是可以揭示这段话主题的词往往是一些出现频率比较小的词，在一整段关于美食的描述中，“美味”的词只会出现一次，而类似于“的”、“他”、“是”，“和”之类的词却会出现很多次，其中“的”、“他”、“她”、“它”等词语叫常见词，也就是高频词。“是”、“和”、“中”、“的”、“得”、“地”等几十个词被称为停止词。而对于这种常见词语过多的采样却忽略了对于关键词的采样会十分不利于对于文本信息的挖掘的。

因此要提出一种算法，目的是消除高频词对罕见词对影响，这个算法必须满足两个条件：

（1）对于主题内容揭示比较准确的词语获得的权重较大、或更不容易被过滤掉。而高频常见词权重很低或容易被过滤掉。

（2）停止词不应该被采样，或以极小的概率被采样。

一种常见的消除常见词语罕见词之间的不平衡现象的做法是对于每一个被采样的单词，在采样过程之中都有一定的概率被抛弃，这就意味着出现更高的常见词被抛弃的概率也更大，这样就使得那些罕见词有更大的概率被采样进来，进而可以更好的揭示文章主题。词语被抛弃的概率遵循着公式：

(5.2)

其中)是词出现的频率，是设定的阈值，这种方法也被称为降采样法。下面本文对于的不同选择作出判断，其中通过前面的分析结果，负采样率最终固定在10。

表5.2 不同降采样阈值的结果

Table 5.2 The results of different subsampling thresholds

|  |  |  |
| --- | --- | --- |
| t | Time [min] | Accuracy [%] |
| 0.1 | 67 | 68.1 |
| 0.01 | 58 | 68.3 |
| 0.001 | 49 | 67.7 |
| 0.0001 | 37 | 67.8 |
| 0.00001 | 28 | 66.9 |
| 0.000001 | 14 | 55.4 |
| 0.0000001 | 6 | 38.1 |

从表格中可以看出，由于降采样算法的存在，可以加速整个算法的训练过程，并且随着阈值的逐步减小，加速效果也逐步明显。但是当阈值进一步减小的时候，可以发现算法的准确度出现了下降，这是由于随着阈值的降低，某些包含关键信息的并不是那么罕见的词语也被舍弃掉了，在这里本文采用0.00001的阈值，也就是的取值。

5.2.3 词向量维度

词向量可以说是深度学习在自然语言处理领域最成功的一次应用了。不同的词向量维度适用于不同的自然语言处理任务，在下面本文对不同的维度进行探究，其他的参数均按照上面的讨论所得结果使用。

表5.3 不同词向量维度的结果

Table 5.3 The results of different dimensions of word embedding

|  |  |  |
| --- | --- | --- |
| Dimension | Time [min] | Accuracy [%] |
| 50 | 24 | 49.1 |
| 100 | 38 | 53.3 |
| 200 | 58 | 62.7 |
| 500 | 92 | 72.8 |
| 1000 | 433 | 75.1 |
| 1500 | 2577 | 79.2 |
| 3000 | 6237 | 82.1 |

从表格之中可以看出来，随着词向量维度的增大，对于文本的情感分析的准确度确实可以得到提升，但是这是牺牲了训练时间为代价的。也许小维度的词向量在诸如实体命名、词性标注等语法任务上表现更加出色，但是在情绪分析领域，好无疑问维度较大的词向量效果更好。

结合试验所采取的硬件条件，以及计算能力，本文最终选择500维的词向量维度，这是在有限的计算条件下，能够采取的准确率较高的方案。

## 5.3 Bi-LSTM模型参数选择

由于结构上的不同，LSTM模型与其他的深度学习神经网络模型在应用上也有着很多的不同。这体现在神经网络层数、正则化、参数优化等各个方方面面。

5.3.1 神经网络层数和神经单元

LSTM与最传统的深度学习CNN最大的不同之处在于对于神经网络层数的选择上。传统的神经网络，ANN或者CNN很容易就采用了一个几百层甚至上千层的神经网络，但是一般的LSTM都是仅有几层的机结构，谷歌搭建的大规模NMT自然语言翻译系统，包含了八个编码器层和八层解码器层的复杂网络结构中，层数也没有超过二十层。

下面本文对不用层数的Bi-LSTM网络的表现，本文分别在训练及和交叉验证集上验证网络稳定时的代价函数。另外，在本文中如五特殊指出，LSTM相关网络本文均采用交叉熵的代价函数。

表5.4 不同层数的网络的结果

Table 5.4 The results of different layers of neural network models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dimension | Time [h] | Accuracy [%] | Training Loss | Validating Loss |
| 1 | 0.72 | 74.2 | 622.421 | 763.872 |
| 2 | 1.35 | 78.1 | 681.919 | 599.451 |
| 3 | 4.12 | 77.3 | 691.331 | 698.543 |
| 4 | 18.3 | 78.2 | 591.542 | 711.948 |
| 5 | 27.1 | 72.9 | 645.286 | 1079.232 |

从表格中可以看出，层数的叠加对于网络模型的整体表现并没有太大提升，因此深层次的LSTM并不适用于情感分析任务，而且随着神经网络层数的加深，神经网络模型结构甚至出现了过拟合的现象，交叉验证集的代价函数变得突然巨大了起来，因此不宜选择过于深层的Bi-LSTM神经网络模型。另外随着神经网络层数的增加，算法达到收敛状态的训练时间层指数增长。因此，不宜选择过深的神经网络模型，综合考虑，本文最终选择两层的Bi-LSTM神经网络模型。

在神经网络的层数确定以后，下一步就是选择隐含层每层神经网络的神经单元数。输入层的神经单元数，由于本层承接Skip-gram模型的输出，也就是维度维500的向量。对于隐含层，不同的神经单元数个数的模型表现可以有很大的差异，关于Bi-LSTM的论文数量十分稀少，也并没有成熟的开源Bi-LSTM方案公布，因此本文对不同的神经单元个数进行了尝试，由于本文之前确定了诸如神经网络隐含层层数、激活函数等参数，下面只对本文确定了的参数进行控制变量法对比试验，以其判断在现有条件下，不同隐含层神经网络没层单元数对算法准确性的影响。另外其他的神经网络单元数可能与其他的激活函数或网络深度组合会达到更好的效果，但是本文并没有取得这样的组合模型。

表5.5 不同层数的网络的结果

Table 5.5 The results of different layers of neural network models

|  |  |  |
| --- | --- | --- |
| Units | Time [h] | Accuracy [%] |
| 128 | 0.96 | 58.4 |
| 256 | 1.21 | 54.9 |
| 512 | 1.43 | 63.7 |
| 1024 | 1.62 | 55.9 |
| 2048 | 2.23 | 86.9 |
| 4096 | 3.27 | 74.1 |

由于网络结构在每个隐含层拥有2000个左右神经单元可以取得最好的结果，因此本文择隐含层每层有2048个神经单元的网络结构模型。

## 5.4 训练过程参数选择

5.4.1 大规模数据的梯度下降

在数据量十分巨大的情况下，传统的梯度下降，也就是每一次梯度更新都需要计算所有样本的代价函数的反向传递，这种方法被称为批量梯度下降法(Batch Gradient Descent, BGD)，批量梯度下降法由于每次都计算了所有的样本数据，因此每次都是朝着最小化梯度的方向前进，可以实现全剧最优解，该算法的做法就是用所有的训练数据更新参数，缺点因此也十分明显，那就是要占用大量的计算资源，算法的训练时间特别长。

另外一种常用的做法是每次更新的时候只考虑一个样本点，下一次更新则选择另外一个样本点。这种做法叫做随机梯度下降法(Stochastic Gradient Des-cent, SGD)。另外由于样本中的噪声存在可以参数的更新可能不会朝着极小值的方向更新，但是由于更新很多轮，总体的方向还是会超值极小值的方向前进。另外，如果代价函数存在不止一个极小值，算法极有可能无法取得全剧最优解，而在局部极小值点附近徘徊。这种做法的好处是，可以节省大量的计算资源和计算时间，缺点则是可能陷入局部最优解或在全剧最优解附近徘徊。

除此之外还有另外一种做法，是综合上述两种方法的折中方法，这种做法是把训练集分为若干个不同的批次，每一次更新其中一个批次，然后下一次利用另外批次更新参数，直至更新完所有的参数。这种方法被叫做小批量梯度下降法(Mini-Batch Stochastic Gradient Descent, MSGD)。这种方法综合了批量梯度下降法和随机梯度下降法的特点，算法的好坏十分取决于批量的大小，。这些批量也被称为Batch，影响算法的关键因素就是Batch的大小，如果选择合适，改算法拥有着超越批量梯度下降法和随机梯度下降法的表现，但是如果Batch的大小不理想，这种算法的表现就会十分差劲了。在本文中本文针对几种不同的梯度下降方式进行尝试比对，得到如下的代价函数变化图：

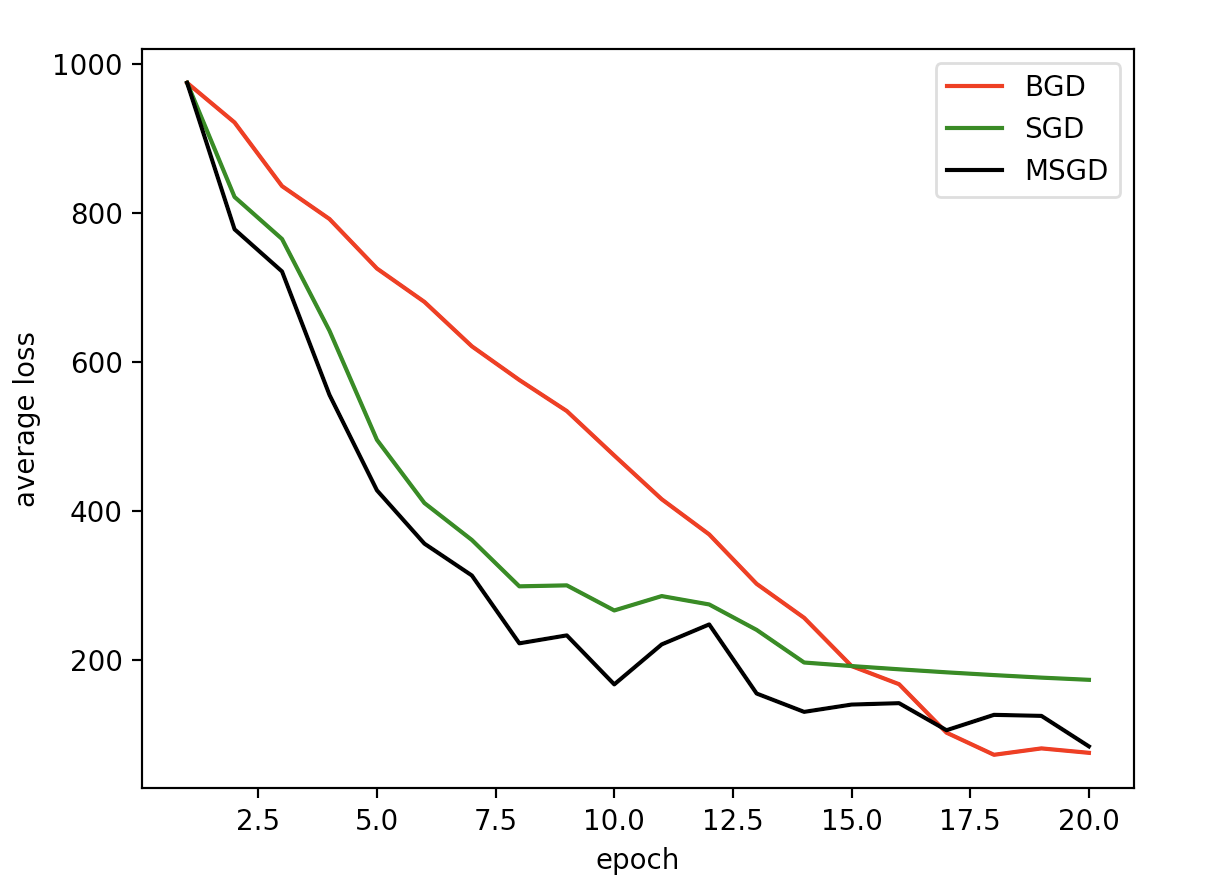


图5.3 不同梯度下降方式下训练集代价函数随着迭代的变化

Fig.5.3 Loss function from different kinds of gradient descent ways on training data

其中由于SGD的下降速度十分缓慢，即使迭代一百轮后，代价函数还在以极小的梯度下降，但是此时对于准确度已经没有实质上的提升了。另外本文也对不同的Batch大小进行了尝试，由于Batch不同大小对方案性能影响很大，的结果并没有展示，这里只展示表现较好结果。



图5.4 不同批量大小方式下训练集代价函数随着迭代的变化

Fig.5.4 Loss function from different batch sizes on training data

最终本文确定了Batch大小为100的小批量梯度下降法。

5.4.2 学习率的选择

还有一个关于网络模型收敛速度的至关重要的参数就是学习率。不同学习率的设置在很大程度上会影响网络模型甚至整个方案的性能表现。学习率可以影响两个关键的性能：第一个是收敛速度，也就是训练时间；第二个是代价函数的收敛值，也就是算法的准确度。

学习率如果取一个较大的值，神经网络代价函数可以以很快的速度下降，大大减少神经网络的训练时间，但是缺点在于神经网络可能无法取得最优解，而是在最优解附近徘徊，这一点又一点像批量梯度下降法。

但如果学习率取值过小，神经网络的代价函数则会以很慢的速度下降，者带来的问题就是神经网络的训练过程会消耗十分漫长的时间，但是优点同样明显，该算法一定会取得代价函数的极小值。所以学习率的选择变成了速度和性能的博弈选择问题。

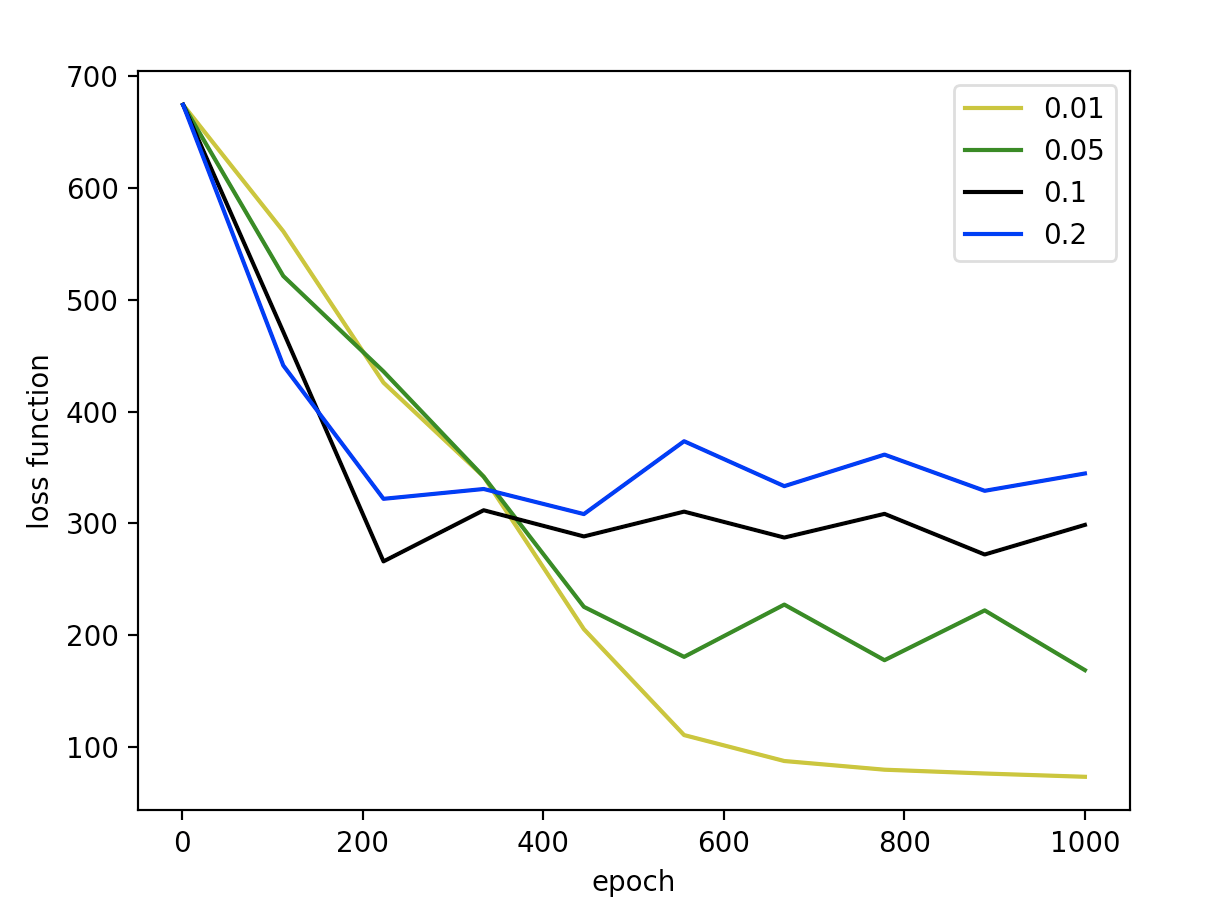


图5.5 不同学习率下训练集代价函数随着迭代的变化

Fig.5.5 Loss function from different learning rates on training data

从图上可以看到随着学习率的增大，代价函数的下降趋势确实变得更加明显，但是随着训练次数的进一步加深，代价函数很难出现进一步的下降，这就是因为学习率选择过大，使得代价函数始终在极小值附近徘徊，却无法进步一下降到极小值点。那么就需要一种可以在训练前中期可以保持一个较大的学习率，在训练过程中逐步见效，最终在训练尾声，可以衰减到一个很小值的动态学习率方案。

本文采用的方法是学习率指数衰减法，初始化学习率为0.2，然后每迭代100次，学习率就会乘以0.92，这样经过几次迭代循环，学习率就会收敛于一个无穷小的数，可以看它的具体表现：

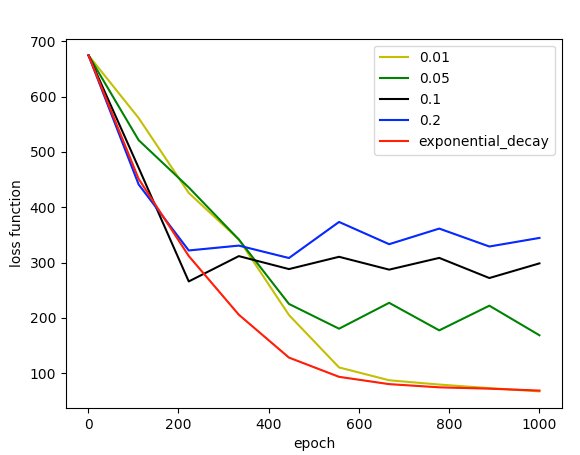


图5.6 指数衰减学习率在代价函数上的表现

Fig.5.6 Loss function on exponential decay learning rate

从图上也可以看出采用指数衰减学习率的方式可以得到一个较好的结果，既可以实现代价函数的快速下降，更不会影响准确度。

## 5.5 正则化

在神经网络训练过程之中，一个不可能避免的问题就是过拟合问题。所谓过拟合是指对于样本集的数据训练过度，对于非样本集中的数据反而无法预估，这是由于训练样本的同时也训练了样本集中的噪声，而现实世界的噪声无处不在，无法预估，因此无法进行很好的处理。与过拟合成对出现的叫做泛化性，过拟合就意味着泛化性能差。过拟合产生的原因有很多种，包括有样本集太小了、模型太复杂以及训练的时间过长。

下面用一个简单的线性回归的例子介绍一下过拟合现象。图中的蓝色线是一个正常的拟合曲线，点均匀地分布在曲线两侧。而红色的线则是过拟合的代表，它过渡的拟合了样本集中的点，对于新的未标注的点可能拟合的就不是那么准确了。

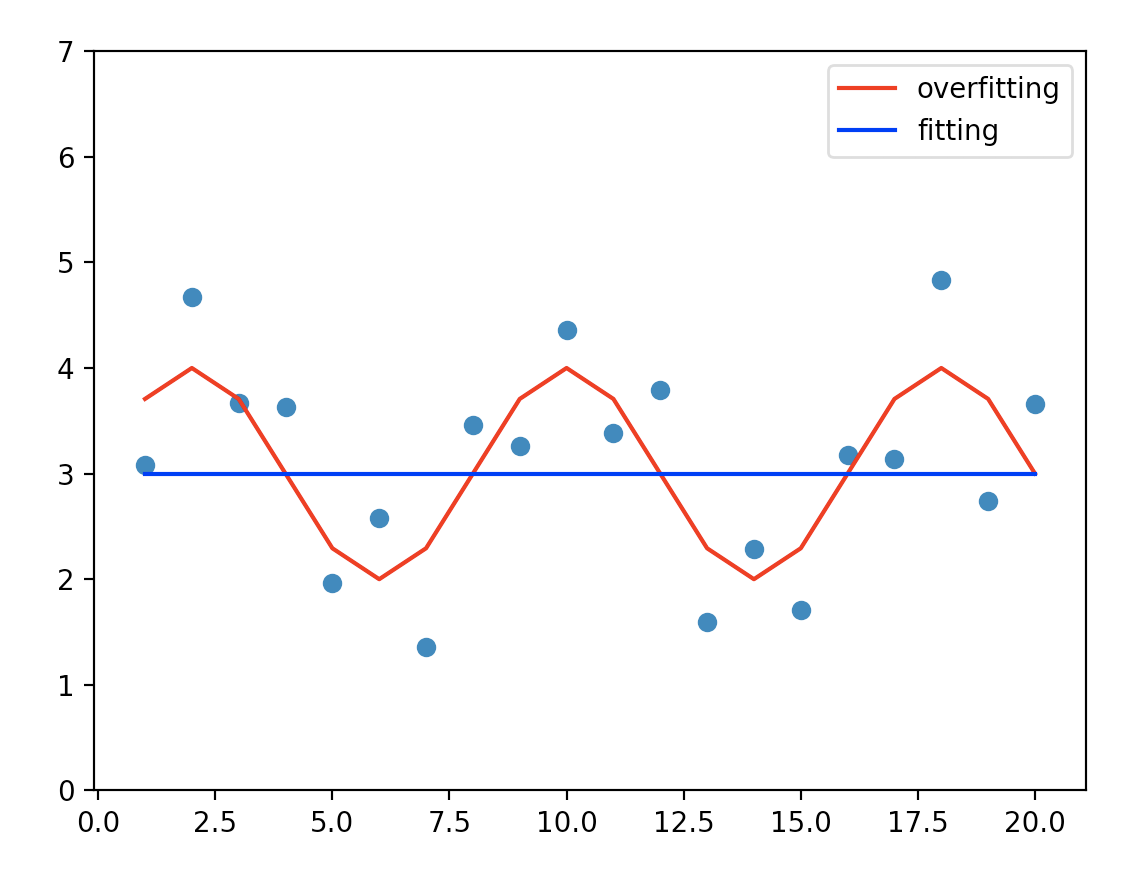


图5.7 过拟合现象

Fig.5.7 An example of over-fitting

解决过拟合问题可以从三个角度出发，第一个就是增大训练集数据规模，第二个是采用正则化的方法，正则化就是在目标函数中引入约束项的方法，可以使得目标函数可以约束模型中的复杂部分，常用的有L1范数正则和L2范数正则等。第三就是采用Dropout算法。

5.5.1 Dropout算法

关于Dropout算法，就是在神经网络的训练过程中，按照事先规定好的概率随机地忽略掉一些神经元，忽略掉的含义就是包括这些神经单元本身、激活函数、门结构以及传递过程统统无视掉，这些神经单元处于不工作状态。由于每一次都随机的忽略掉一些神经元，所以对于随机梯度下降来说，每一部分批量都在训练不同的网络。

Dropout算法的提出很好的解决了这个问题，在层与层之间加上这种算法之后，每一次迭代都会从原始的网络模型中发现一个与之前不同，而且规模较小的模型，如下图所示：

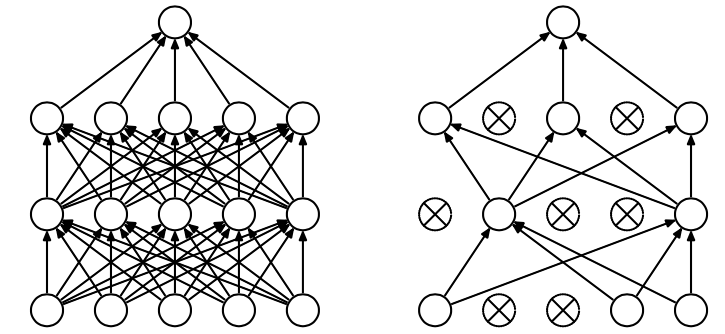


图5.8 Dropout示意图

Fig.5.8 The diagram of dropout

这样一来，对于一个有n个节点的神经网络，在加入Dropout算法之后，就可以看作是2n个模型的集合了，这也起到了组合的效果，但此时要训练的参数数目却是不变的，这在很大成分上减少了时间过长的问题，而且可以有效减少过拟合现象。下图为加上Dropout算法的神经网络与原始网络的对比图。



(a)标准网络 (b)Dropout网络

图5.9 神经元的变化

Fig.5.9 The changes of neurons

自从问世以来，Dropout算法已经在诸多领域取得很好的成果。但是在计算机视觉领域的大多数应用中，批归一化(batch normalization)算法已经成为了归一化的不二选择，但是在自然语言处理领域则不尽然，在这一领域Dropout算法仍然是关键的正则化方法。近年来一些比较成功的改进Dropout 算法被提了出来，包括自适应舍弃元和进化舍弃元等等，但是这些改进Dropout算法都不能很好的应用于NLP领域，这是因为在时间的影响下，由于Dropout舍弃的单元不参与循环神经网络的循环链接，在RNN和LSTM的聚合作用下，很快词向量就被清零了。

5.5.2 循环Dropout算法

在前文中的分析可以知道Dropout算法应用在NLP领域上存在的问题，下面本文介绍一种解决方案就是循环Dropout(Recurrent Dropout)算法，该算法的核心思想就是在每个时间间隔神经网络模型均采用相同的神经单元Dropout抛弃。这样可以有效避免噪声沿着时间序列逐步增大的问题，可以有效对序列化模型的结果进行正则化约束，同时减少训练时间。

本文对比了该算法和其他正则化的方法进行比较，得到如下结果：

表5.6 不同正则化方法的结果

Table 5.6 The results of different regularizations

|  |  |  |
| --- | --- | --- |
| Regularization | Time [h] | Accuracy [%] |
| L1 | 27.2 | 73.2 |
| L2 | 31.1 | 77.8 |
| Dropout | 22.1 | 81.1 |
| Recurrent Dropout | 23.2 | 86.3 |
| L1+L2 | 33.1 | 79.1 |

其中值得说明的是L1范数正则化的方式极易产生稀疏矩阵，也就是传递矩阵中很多参数收敛到0最后导致网络参数全部为0。同样还有常规的随机Dropout算法，也是极易把词向量衰减为0，导致神经网络无法继续进行下一步的迭代。

从上面的表格之中可以知道正如之前分析的一样，循环Dropout率可以取得最佳的效率以及最快的训练速度，因此将作为本文的正则化方案选择，另外还有一个需要讨论的是Dropout率。

5.5.3 循环Dropout率的设定

循环Dropout算法的唯一参数就是Dropout率，也就意味着每一层的神经单元有多少是被舍弃掉的。本文对于不同的Dropout率进行比较，应用于循环Dropout算法上，得到如下结果：

表5.7 循环舍弃元方法不同舍弃率的结果

Table 5.7 The results of different dropout rate

|  |  |  |
| --- | --- | --- |
| Dropout rate | Time [h] | Accuracy [%] |
| 0.1 | 23.3 | 73.2 |
| 0.3 | 16.7 | 77.8 |
| 0.5 | 8.3 | 89.7 |
| 0.7 | 4.2 | 56.3 |
| 0.9 | 3.1 | 21.3 |

可以得出结论：0.5的舍弃率的循环Dropout算法十分适用于本流程，这种方法可以有效减少过拟合现象，并且可以起到加速运算，提升算法准确率的作用。

## 5.6 实验结果与分析

本文采用查全率(Recall)、查准率(Precision)和F1分数(F1 Score)三个指标对方案本身进行衡量。这三个衡量标准的衡量指标是由四个值组成，分别是正确肯定(True Positive, TP)、正确否定(True Negative, TN)、错误肯定(False Positive, FP)和错误否定(False Negative, FN)。其含义如下：

正确肯定(TP)：那些真实值为真，预测值也为真的样本。

正确否定(TN)：那些真实值为假，但预测值也为假的样本。

错误肯定(FP)：那些真实值为假，但预测值为真的样本。

错误否定(FN)：那些真实值为真，但预测值为假的样本。

其中查准率、查全率和F1分数的含义是：

查准率(Precision, P)：所有预测为真的样本中真实值为真的样本占比，其公式为：

(5.3)

查全率(Recall, R)：所有真实值为真的样本中被正确预测出来了的样本数量占比，查全率也被翻译成为召回率，其公式为：

(5.4)

F1分数(F1 Score)：统计学中关于查准率和查全率综合性能到表现，其计算方法为：

(5.5)

经过利用训练集和交叉验证集确定了的方案的全部模型和参数，可以得到最后的结果，对于测试集中的正负各五千条评论，本文的方案给出的最终的最佳预测结果如下

表5.7 本方案最佳预测结果

Table 5.7 The best performance of the program

|  |  |  |
| --- | --- | --- |
|  | Positive | Negative |
| True Predict | 4393 | 4557 |
| False Predict | 607 | 443 |

该最佳结果的查准率为90.84%，查全率为87.8%，其F1分数为89.29%。另外为了消除特殊性对系统的影响，本文采用了多次重复性实验，本文利用测试集对已经训练好了的模型进行了十次验证，得到了十次重复试验的平均结果，其表现如下

表5.8 本方案平均预测结果

Table 5.8 The average performance of the program

|  |  |  |
| --- | --- | --- |
|  | Positive | Negative |
| True Predict | 4109.2 | 4325.7 |
| False Predict | 890.8 | 674.3 |

该方案的平均预测结果的查准率为85.9%，查全率为82.18%，其F1分数为84%。

作为对比试验，本文测试了某公司提供的情感分析API，该API提供可以进行情感分析的接口，同样也是通过JSON文件 传输文本数据。由于该API是收费模式，因此本文只对测试机进行了对比测试，得到的准去率如下：

表5.9 目标API预测结果

Table 5.9 The performance of target API

|  |  |  |
| --- | --- | --- |
|  | Positive | Negative |
| True Predict | 3902 | 4021 |
| False Predict | 1098 | 989 |

该API的平均预测结果的查准率为79.78%，查全率为78.04%，其F1分数为78.86%。

为了得到更好的比对效果，本文分别把目标API的三个标准，以及本方案的平均测试标准通过柱状图对比出来，可以得到无论是对正样本的预测还是负样本的预测，本系统均远超过该API，但是由于该API是基于云的解决方案，速度比本文的方案要快很多。

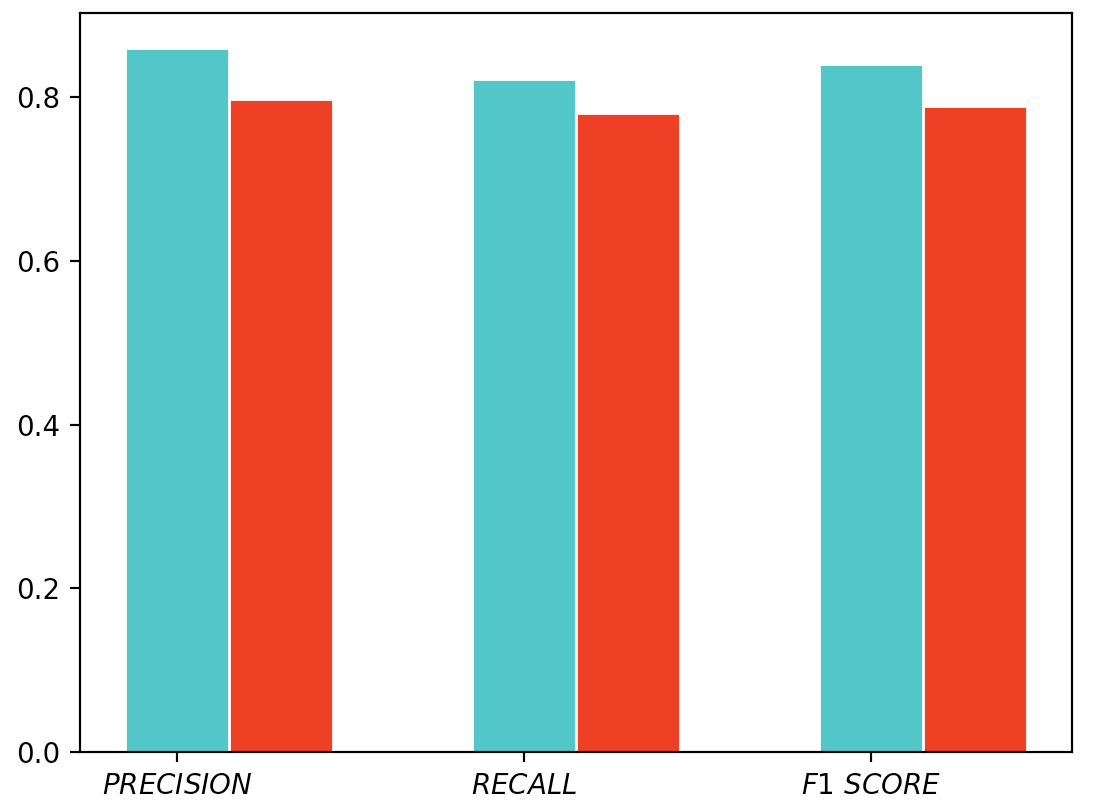


图5.10 不同方案的对比结果

Fig.5.10 The results of different programs

其中左侧蓝色部分是本方案的表现，红色的是API的结果。可以看出在性能上还是有一定的差距的，之所以有这么大的差距，主要原因是在由于本文对样本集大多数都是由影评组成，而影评中往往存在大量的修辞手法，而该API是采用原始的词典方法进行情感分析，就无法识别出影评中的真正情感倾向。而本文采用的双向长短期记忆神经网络则可以有效的改善这一点。另外本方案目前还只适用于对影评进行分析，如果涉及到其他领域的文本样本，则并没有这么好的效果。

## 5.7小结

本章节对于前面章节完整设计方案的进行了具体的实施，并且对于整个流程中的各个模型及参数的选择进行了比对实验，分别包括有：1.Skip-gram算法中的参数选择以及神经网络设置。2.Bi-LSTM的神经网络模型的参数设置、初始化方式、神经具体结构。3.各种常见的梯度下降方法以及对比，还有最终选择随机梯度下降后的参数设置，学习率的选择以及不同学习率的比对。4.由过拟合现象引出几种常见正则化方案，并进行比对分析，最终采用了循环Dropout算法，以及Dropout 率的设置。5.最终方案的最佳表现和多次实验的平均表现。并利用三种评价方案对算法的准去度进行判断。

# 第6章 总结与展望

本章是全文的最后一章，本章将分为两个部分，一是对前面的工作的总结，包括训练的细节，还有值得注意的地方。第二部分则是指明本文仍存在的问题，以及下一阶段的工作。

## 6.1 工作总结

对于文本设计的中文情感分析方案在当今的互联网时代有着十分重要的意义，无论在提升产品质量、提升用户体验还是在舆论监督方面都有着巨大的帮助。尽管应用深度学习已经在情感分析等自然语言处理领域取得了很大的成功，但是对于中文的成功应用还是很少，很多公司还在进行尝试阶段，现有的较为成熟的中文情感分析方案更多的还是依赖于基于词典的统计学方法。本文提出了一整套关于深度学习的中文情感分析方案，并且取得了不错的效果。虽然对比英语等拉丁语系的应用，本方案的准确度还有很大的提升空间，但是在科学发展的道路上这种尝试是必不可少的，本文基于深度学习中的长短期记忆神经网络(LSTM)，提出了一种双向LSTM神经网络结构，利用词向量的方法对文本数据进行情感分析。

首先第一个面对的问题就是文本样本严重不足，而且没有标注。本文采用的方法是利用爬虫技术抓取电影评论文本数据并根据其评分系统自动分类。

接着需要把整段的文本数据进行分词，这也是中文自然语言处理领域必不可少的工作。本文分析了中文分词方案中的两种分词方式词典法和统计学方法，并且分析了不同的开源分词方案，最终选择了适合的分词技术。

接下来面对的问题是把词转化为电脑到中易于处理的数字化存储形式，本文采用的是谷歌提出的Word2Vec方法，采用神经网络模型的结构数字化词信息，经过本文对两种W2V模型，CBOW和Skip-gram的分析，最终选择了效果较好的模型。然后对于模型训练过程中的参数调节，网络结构和优化方案本文都进行了选择和调整。

接下来本文对论文的主体方案，Bi-LSTM网络进行了数学分析和实际检验。该网络模型可以有效的减少常规RNN和LSTM中的梯度消失问题，并可以从一段文本的两端验证文章的情感态度，避免了评论前后态度不一致或是中立观点对于结果的影响。随后本文又进一步确立了网络训练过程之中的参数调节、梯度下降算法、正则化方式、学习率的统计以及初始化方式等。

最后本文通过实验分析可以得到，本文的方法是针对于中文文本情感分析等有效工具，本文分别通过三个方面：查准率、查全率和F1分数验证了本文提出的方案等性能和结论，并与其他常用的中文情感分析方案作出对比。

## 6.2 问题与展望

包括深度学习在内的机器学习领域现在正是处于风口浪尖的时刻，我们作为高等院校的科研人员在积极投身研究的同时，也要警惕泡沫现象，等到风口浪尖过去，泡沫挤出，行业回归平静的时候仍然要保有一颗初心，科研的道路没有一帆风顺，要平心静气，不骄不躁，稳扎稳打才能走通科研的道路。

在的训练过程中，我发现一个有趣的现象，简单的模型配合海量的大数据样本的效果要远远超过数据量很小的复杂模型。因此数据样本的数量和质量决定了算法能够达到的高度，而算法、模型、优化只能无限的逼近这个高度。在本文的整套方案和完整的训练过程之中，文本数据无论是数量还是质量都是远远不够的，数量上是可以得到的数据太少了，很多公司都设置了严格的反爬虫技术。质量方面，由于本文是大量采用抓取的电影评论数据，少量的商品评论数据，因此该文本数据集有着很大的重复性，样本集不同评论之间的主题重复率很高，并不具有普遍的适用性。希望以后可以有更多的大公司以公开自己的评论文本数据。

在分词技术领域，有一种新型的基于机器学习的方式正在逐渐兴起，但是目前并没有成功的应用存在，希望随着科学的逐渐发展，可以有更多更好的分词程序开源出来，这样可以进一步提升分类效果的准确性。

由于受到硬件计算能力的限制，Skip-gram模型并不一定是最适合的中文词向量化方式，有学者提出了大规模的英文词向量化的CBOW模型，但是并没有中文词向量工具取得类似的成果，而且该方案的实现硬件上消耗极大，因此在这里并没有对其进行尝试。

有学者在LSTM的基础上，提出了注意(Attention)机制，本文在选择了Bi-LSTM方案之后并没有对结合注意机制的改进LSTM网络进行尝试，不知道类似的改进结构是否有着更好的表现。

# 参考文献

[1] Turing, A.M. Computing machinery and intelligence[J]. Mind. 1960, 59: 433-460.

[2] F Rosenblatt. The perceptron, a perceiving and recognizing automaton Project Para[D]. Cornell Aeronautical Laboratory, 1957.

[3] N Chomsky & G Miller. Introduction to the formal analysis of natural languages[M]. New York: Wiley, 1963.

[4] N Chomsky. Formal properties of grammar[M]. New York: Wiley, 1963.

[5] N Chomsky. Knowledge of language: Its nature, origin, and use[M]. New York: Wiley, 1986.

[6] Shannon. A mathematical theory Of Communication[J]. Bell System Technical Journal, 1948, (27):379-423.

[7] 冯志伟.自然语言的计算机处理[M] .上海:上海外语教育出版社, 1996.

[8] 邵泽国. 语言科学发展的新分支—自然语言处理[J]. 电子科技, 2013 ,5(26): 167-171.

[9] B Pang, L Lee. Opinion mining and sentiment analysis[M]. Seattle: Foundations and Trends in Information Retrieval, 2008.

[10] F Jelinek. statistical method for speech recognition [D]. MIT press, 1998.

[11] Linnainmaa S. The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors[D]. Master’s thesis, Univ. Helsinki. 1970.

[12] Tang J, Deng C, Huang G B. Extreme learning machine for multilayer perceptron[J]. IEEE transactions on neural networks and learning systems, 2016, 27(4): 809-821.

[13] Dubossarsky E, Friedman J H, Ormerod J T, et al. Wavelet-based gradient boosting[J]. Statistics and Computing, 2016, 26(1-2): 93-105.

[14] Thomas M Cover, Joy A Thomas. Elements of Information Theory[M]. New York: Wiley, 1991

[15] Moraes R, Valiati J O F, Neto W P G O. Document-level sentiment classification: An empirical comparison between SVM and ANN[J]. Expert Systems with Applications, 2013, 40(2): 621-633.

[16] Ackley D H, G Hinton. A learning algorithm for boltzmann machines[J]. Cognitive science, 1985, 9(1), 147-169.

[17] Krizhevsky A, Sutskever I, Hinton. Imagenet classification with deep convolutional neural networks [C]. Advances in neural information processing systems. 2012: 1097-1105.

[18] Kalchbrenner N, Grefenstette E, Blunsom P. A Convolutional Neural Network for Modelling Sentences [C]. ACL. 2014:655-665.

[19] Johnson R, Zhang T. Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding. [C]. NIPS. 2015: 49–58.

[20] Johnson R, Zhang T. Effective Use of Word Order for Text Categorization with Convolutional Neural Networks. [J]. Nature，2016，529(7587): 484-489.

[21] Kim Y. Convolutional Neural Networks for Sentence Classification[C]. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), 1746–1751.

[22] Wang P, Xu J, Xu B, Liu C, Zhang H, Wang F, Hao H. Semantic Clustering and Convolutional Neural Network for Short Text Categorization[C]. Proceedings ACL .2015: 352–357.

[23] A Waibel, T Hanazawa, G Hinton，Phoneme recognition using time-delay neural networks [J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1989, 37(3):328-339

[24] Yoshua Bengio. A Connectionist Approach to Speech Recognition[J]. International Journal of Pattern Recognition and Artificial Intelligence, 1993, 7(4): 647-657.

[25] Yoshua Bengio, Y LeCun. Convolutional networks for images, speech, and time series[C]. The handbook of brain theory and neural networks, 1998: 255-258.

[26] Alex Graves. Supervised Sequence Labelling with Recurrent Neural Networks [M]. New York: Springer，2012.

[27] Yoshua Bengio. Learning long-term dependencies with gradient descent is difficult [J]. IEEE Transactions on Neural Networks, 1994, 5(2):157-166.

[28] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions [J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based, 1998, 6(2):107-116.

[29] 吴军.数学之美[M] .北京:北京邮电出版社, 2014.

[30] 邵帅, 刘学军, 李斌. 基于关键句分析的微博情感倾向性研究[J]. 计算机应用研究, 2018, 35(4). http://www.arocmag.com/article/02-2018-04-046.html

[31] 黄昭婷, 刘媛, 丁鑫. 一种基于影评文本情感分析的电影特征模型[J]. 国家新闻出版广电总局电影数字节目管理中心，2017，1(9)：26-30.

[32] 乐国安, 董颖红, 陈浩, 赖凯声. 在线文本情感分析技术及应用[J]. 心理科学进展，2013，21(10):1711-1719.

[33] 王素格，李德玉，魏英杰. 基于同义词的词汇情感倾向判别方法[J]. 中文信息学报，2009，23(5) : 68-74.

[34] 王伟, 王宏伟. 特征观点对购买意愿的影响：在线评论的情感分析方法[J]. 系统工程理论与实践，2016，36(1) : 63-76.

[35] 冯兴杰, 张志伟, 史金钏. 基于卷积神经网络和注意力模型的文本情感分析[J]. 计算机应用研究, 2018, 35(5). <http://www.arocmag.com/article/02-2018-05-057.html>

[36] 梁军, 柴玉梅, 原慧斌, 等. 基于极性转移和LSTM递归网络的情感分析[J]. 中文信息学报，2016，29(5) : 152-159.

[37] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, Christian Jauvin. A Neural Probabilistic Language Model [J]. Journal of Machine Learning Research, 2003, 2(3):1137-1155.

[38] Bo Pang, Lillian Lee. Thumbs up?: sentiment classification using machine learning techniques [J]. Neural Information Processing Systems (NIPS) 2002, 2(10):79-86.

[39] I Goodfellow, Jean Pouget, Yoshua Bengio, Mehdi Mirza, e.t. Generative Adversarial Nets [C]. Journal of Machine Learning Research, 201, 27:1-9.

[40] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech Recognition with Deep Recurrent Neural Networks. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), (3):6645–6649.

[41] Sebastian Ruder, Parsa Ghaffari, John G. Breslin. A Hierarchical Model of Reviews for Aspect-based Sentiment Analysis[Z], arXiv, 2016, https://arxiv.org/abs-/1609.02745

[42] 梁南元. 书面汉语自动分词系统—CDWS [J]. 中文信息学报，1987, 1(2)：44-52.

[43] 郭进. 统计语言模型及汉语音字转换的一些新结果[J]. 中文信息学报，1993, 7(1): 18-27.

[44] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. Dis-tributed Representations of Words and Phrases and their Compositionality [Z], arXiv, 2013, https://arxiv.org/abs/1310.4546

[45] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space [Z], 2013, https://arxiv.org/abs/1301.3781

[46] Michael U Gutmann and Aapo Hyvarinen. Noise-contrastive estimation of un-normalized statistical models, with applications to natural image statistics[J]. The Journal of Machine Learning Research, 13:307–361, 2012.

[47] Andriy Mnih and Yee Whye Teh. A fast and simple algorithm for training neural probabilistic language models[Z]. arXiv, 2012, https://arxiv.org/abs/1206.6426

# 致谢

时间匆匆而过，转眼间，在东北大学的研究生生活就要告一段落了。

如今，即将离开校园的我，将要开启人生的新篇章，内心充满着对未来生活的向往。回顾我的研究生生涯，我感慨万千。

首先，我要感谢我的恩师薛定宇教授。薛老师为人治学研究，认真又不失风度，他在我读研伊始就鼓励我选择我自己的研究兴趣，同时充分的鼓励我进行我的研究，激励着我前进。薛教授在我的研究生生涯充分的给予我的充分的自由，让我能够自主的学习，锻炼自主学习能力。另外我在薛老师身上学到的不止有严谨的教学态度，低调朴实的科研精神，更重要的是做人。希望我有朝一日可以像薛老师一样成为一个有态度的学者，一位品格高尚的长者，一位温暖可爱的亲人。

我要感谢同一个实验室的潘峰老师，陈大力老师和王浩然老师，我从你们的身上学习到了很多，无论是学术研究还是生活上，你们都给我提供了非常大的帮助。特别是潘峰老师，在百忙之中还对我的论文编写提出宝贵的意见，指导我完成论文的编写。感谢你们对我的宽容，让我能够自由的科研，希望我们实验室可以永远保持团结，像一家人一样。

感谢东北大学信息学院人机所的每一位老师，是你们打开了我求知的大门，在各位老师的精彩授课上，我才能走向机器学习的大门。特别是陈东岳老师，在您的课上我学到了很多，特别是您的数学思维，让我获益匪浅，叹为观止。

感谢在大疆实习期间遇到的阮士杰，胡强和苗非繁等工程师，是你们用亲身经历和朝夕相处教导我如何成为一名合格的工程师，以及一名合格的工程师应该具有的态度。

感谢我的同学们，感谢岳晓阳同学、于嘉辉同学、郑夏同学等等。因为有你们的帮助，我的学习生涯才会丰富多彩，和你们携手一起走过的岁月是我最美好、最温暖的记忆，在你们身上我学到了很多。

感谢我的父母，我的父母不仅对我有养育之恩，而且他们所给予我的自由成长对于我的性格形成有着很大的帮助，感谢他们一直支持着我读书，一直支持着我前进。

最后，感谢百忙之中抽出时间评阅本论文的专家和教授，诚恳地期待各位专家的帮助和指点。