

中国科学技术大学

学士学位论文



基于卷积神经网络的电影评分预测

作者姓名：姚凯

学科专业：计算机科学与技术专业

导师姓名：曹涌 主管研究员，金培权 副教授

完成时间：二〇一七年五月

University of Science and Technology of China
A dissertation for bachelor's degree



**Convolutional Neural networks
based movie rates prediction**

Author's Name: Kai Yao
Speciality: Computer Science and Technology
Supervisor: Yohn Cao, Peiquan Jin
Finished Time: May, 2017

致 谢

在完成此论文期间，我有幸得到了微软亚洲研究院的曹涌老师和科大的金培权老师的指导。两位老师的悉心指导，让我受益匪浅。在此对两位老师的帮助表示感谢。曹老师在我大四实习时给予了很多关心和帮助；金老师在我本科学习的过程中给予了很多帮助。两位老师都在我留学申请过程帮助进行了推荐。感谢两位老师！

在本科学习的过程中，我得到了邢凯老师的很多指导和帮助，在邢老师的实验室学到了很多。邢老师也在我留学申请的过程中帮忙作了推荐，在此向邢老师表示感谢！

在本科学习生活的过程中，班主任张俊霞老师给予了我很多关心和帮助，在此深表谢意！

在科大的四年里，受到过很多老师和同学的关心帮助，由于篇幅所限不能一一尽述，在此一并表示感谢！

目 录

摘要	3
Abstract	4
第 1 章 绪论	5
1.1 课题背景与意义	5
1.2 相关研究现状	6
1.3 本课题主要内容	7
第 2 章 文本情感分析	9
2.1 基于词库的方法	9
2.2 基于机器学习的方法	9
2.2.1 朴素贝叶斯分类器	9
2.2.2 支撑向量机	11
2.3 小结	13
第 3 章 基于深度学习的文本情感分析	14
3.1 关于神经网络	14
3.2 词嵌入	15
3.3 卷积神经网络	15
3.3.1 卷积神经网络的结构	15
3.3.2 卷积神经网络应用于文本问题	16
3.3.3 具体算法	17
3.4 复发神经网络	18
3.5 小结	19
第 4 章 基于卷积神经网络的影评分类实验	20
4.1 数据集与预处理	20
4.1.1 数据的获取与概况	20
4.1.2 分词与训练词嵌入	20
4.1.3 实验设置	21

4.2 不同的卷积神经网络结构	22
4.2.1 单层单通道	22
4.2.2 单层双通道	22
4.2.3 双层卷积神经网络	22
4.3 Baseline 实验	22
4.4 小结	23
第 5 章 实验结果与分析	24
5.1 实验结果	24
5.2 结果分析	24
5.3 小结	27
第 6 章 总结	28
6.1 全文总结	28
6.2 未来工作展望	28
参考文献	30

摘 要

近年来,随着各种论坛、评论网站不断涌现,越来越多的用户文本出现在网络上。这些文本中很多具有情感倾向,如影评、商品评论、博客评论等。分析这种倾向,有利于获得大众对于电影或商品的评价、了解舆情。因此,对文本进行情感分析的研究有着很强的实际意义。

目前对文本进行情感分析的方法有很多,传统的方法有基于词库的方法,使用已知的情感词典;也有基于机器学习的方法,如朴素贝叶斯和支撑向量机。近年来,深度学习的方法也被用在文本情感分析的问题上。卷积神经网络(Convolutional Neural Network, 简称为 CNN)是一种发源于计算机视觉领域的深度学习算法,因其模型非常适合图像数据的特征而在计算机视觉领域取得了巨大成功。通过使用“词嵌入”(word embedding)的方法,卷积神经网络也被应用于文本领域,在文本分类等问题上取得了成功。

本文以来自豆瓣的中文短评为语料,使用卷积神经网络对其进行情感分析,判断其情感倾向,并由此预测每一条短评对应的评分(星数)。本文采用 Word2vec 的算法,从分词后的 12 万句短评和 10 万篇长评的语料训练出一个中文词语的 word embedding。基于训练出的 word embedding 将每个词语转化为词向量,在此基础上使用 Tensorflow 搭建神经网络的模型,进行情感分析。本文重点实现了卷积神经网络的多种结构,同时实现了复发神经网络(Recurrent Neural Network, 简称 RNN),以及传统算法如朴素贝叶斯和支撑向量机在这一问题上的应用。分析和比较了不同算法的实验结果以及性能。

关键词: 情感分析, 深度学习, 卷积神经网络

Abstract

Many kinds of forums and comment sites appear on the internet in recent years, therefore, more and more user-generated text contents can be found on the internet. A big part of the text contents generated by user have sentiments, for example, movie reviews, product reviews, blog comments, etc. Analysis of the sentiments in user-generated text contents is helpful for getting feedback of certain movie or product from customers. It is also helpful for analysis of people's opinions or attitudes to certain event. Therefore, research concerning text sentiment analysis has practical meaning.

Currently, there is several methods to do sentiment analysis. For example, traditional lexicon based methods, using a sentiment dictionary; machine learning based methods, such as Naive Bayes classifier and Support Vector Machine (SVM). In recent years, deep learning methods are also used in the task of text sentiment analysis. Convolutional Neural Network (CNN) is deep learning algorithm which was originally used in computer vision field, it has gained significant success in computer vision since it is suitable for the data structure of picture. By using the technique of word embedding, we can also use CNN in the field of text. It has already succeeded in sentence classification.

This paper uses a corpus made up of Chinese movie short comments from Douban. We apply CNN models over this corpus to do text sentiment analysis, and predict the rates of each comments correspondingly. We use Word2vec algorithm to train our corpus of over 160 thousand short movie comments and 100 thousand movie review, and get a word embedding dictionary of Chinese words. We build CNN models over the embedded data using Tensorflow to do sentiment analysis and get the rate predictions. We focus on the implementation of different structure of CNN, we also implement a Recurrent Neural Network (RNN) model, and traditional methods such as Naive Bayes and SVM for comparison purposes. This paper compares different performance of each algorithms, and analyses the results.

Key Words: Sentiment Analysis, Deep Learning, Convolutional Neural Network

第 1 章 绪论

1.1 课题背景与意义

近年来，随着各种社交媒体和论坛的发展，互联网上的影评已经成为了观影者表达观点、进行交流的重要方式。互联网上出现了许多以影评为主要内容的网站、社区，如 IMDB，Rotten Tomatoes，以及中文的豆瓣等。在这些网站上每天都产生出大量的来自用户的文本。这些文本中往往包含了丰富的用户情感信息，如用户对某个电影、演员的评价等。这些用户生成的文本是网站了解用户观点，电影制作者进行市场调研的重要资源。

豆瓣、IMDB 上的影评都带有对应的评分（打星数），这样的“影评—评分”对可以看做有标注的数据，它们不需要进一步的人工标注，就可以被用作情感分析的语料。另一些博客或论坛性质的网站，则没有要求用户在给出评论的同时打分。图 1.1 为一些来自互联网的影评示例。



图 1.1 上、中：带有对应评分的影评，分别来自豆瓣和 IMDB；下：没有评分的影评，来自知乎

对影评进行情感分析，将文本形式的影评转化为对应的评分预测，对于网站了解用户对于电影的评价、获得用户反馈和针对用户进行推荐都有着重要意义。在网站不要求用户在评论的同时给出评分的情景下，文本是获得用户情感信息的唯一途径。分析用户文本远不如分析量化的评分容易，如果要人工进行统计分析，工作量会非常大。因此，使用算法对影评进行情感分析，从用户文本中预测该用户对电影的评分，可以大大帮助网站分析来自于用户的数据，收集用户对某一部电影的量化评价。量化后的影评数据方便进行统计，也可以帮助电影制作者进行市场调查。此外，一个用户对各个电影的评分的预测可以很好地体现该用户的特征，评分预测的结果可以用作推荐系统的输入数据，使网站针对每个用户投放的推荐更为准确。

对于既有影评，又要求用户同时对电影打分的网站，对影评进行情感分析仍然有很重要的意义。因为每个用户对于分数的理解有很大差别，有些用户认为三星是“不错”、“满意”的评分，而另一些则认为是“不满意”、“相当差”。故单纯基于用户打分来收集的对电影的评价往往是有一些偏差的。对文本影评进行情感分析的结果可以在原有评分的基础上起到一个平滑（smooth）的作用，使统计数据更加准确。

1.2 相关研究现状

文本情感分析一直是自然语言处理方向研究的热点，现代进行文本情感分析的方法一般有基于词库（或基于知识）的方法、基于机器学习的有监督学习和无监督学习的方法、以及基于深度学习的方法。

基于词库的文本情感分析方法会采用已知的情感词典（一般由人工筛选出常见的带有情感色彩的单词并进行标注），对于每一个句子，根据其包含的情感单词以及句子的语法结构计算该句的情感倾向^[1]。

现有的机器学习的方法被广泛应用于文本情感分析的问题。监督学习的方法如朴素贝叶斯、最大熵和支撑向量机等，都可以用于情感分析。Pang 等人使用这些方法在来自 IMDB 的英文影评数据上进行了二分类的实验，其结果证明使用单个单词（unigram）作为特征进行分类就可以取得很好效果，支撑向量机在 3 折交叉验证中取得了 82.9% 的准确率^[2]。

非监督学习的方法也被应用于情感分析，Peter D. Turney 等人使用一种基于情感极性（Sentiment Orientation）的非监督的方法^[3]。一个句子的分类由它包含

的所有短语的情感方向的平均值决定。每个短语的情感方向由它与单词 **excellent** 和 **poor** 的相互信息 (mutual information) 决定。该方法在来自 Epinions 的 410 条评论上进行了验证, 取得了 74% 的准确率。

深度学习的模型也被越来越多地应用于情感分析。卷积神经网络 (Convolutional Neural Network, 简称 CNN) 是一种最早发源于计算机图形学的深度学习模型。Yoon Kim 等人使用将单词 **embedding** 为词向量, 再用 **word vector** 组成的矩阵来模拟图形矩阵的方法, 将 CNN 应用在了文本分类的问题上^[4]。Yoon 等人使用一个简单的只有一层卷积层的 CNN 在 6 个数据集上进行了实验, 均取得了很好的效果。

1.3 本课题主要内容

本课题使用来自豆瓣的中文短评 (每条长度不超过 140 字, 带有评分, 可以看作有标注的数据) 作为语料, 使用卷积神经网络等方法进行文本情感分析, 由影评预测对应的评分。

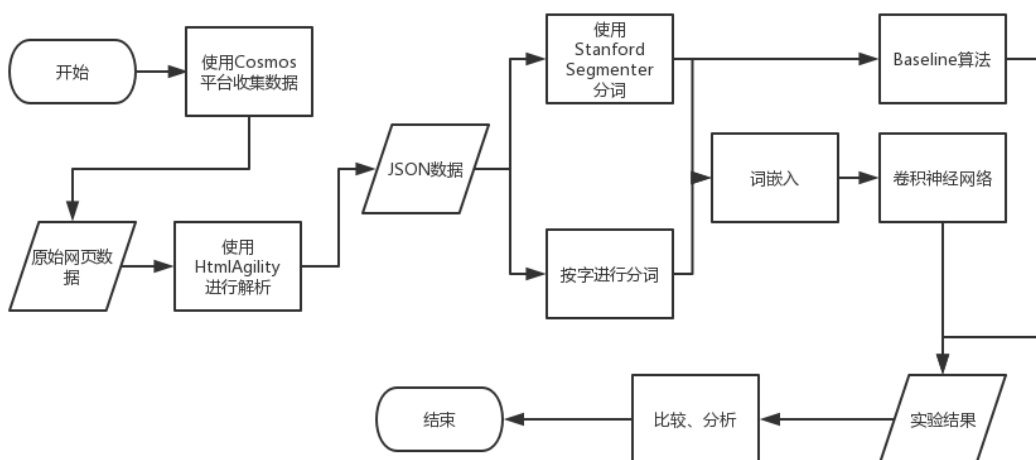


图 1.2 本课题实验流程图

1. 收集数据。使用微软必应搜索的 Cosmos 分布式处理平台获取了来自豆瓣的 16 万短评数据以及 10 万长评数据。

2. 数据预处理。进行中文分词, 使用 Word2vec 算法训练出中文单词的 word embedding

3. 建立深度学习模型, 调整网络结构, 找到适合中文影评数据的模型; 实现

朴素贝叶斯等传统算法，作为比较。

4. 比较和分析实验结果

本文的组织如下：第一章为绪论，介绍选题的背景和意义，简述当前相关领域研究状况和本课题主要工作；第二章介绍了文本情感分析的相关定义，以及基于词库的方法和基于机器学习的方法；第三章介绍了深度学习的方法在文本情感分析问题上的应用，详细介绍了用于文本分类的卷积神经网络，包括其模型细节和具体算法；第四章介绍了使用卷积神经网络进行影评分类实验的数据集和实验设置；第五章列出了实验结果，并对 **baseline** 实验以及多组卷积神经网络的实验结果进行了对比和分析；第六章进行了全文总结和未来工作的展望。

第2章 文本情感分析

文本情感分析 (Sentiment Analysis), 也被称为意见挖掘 (Opinion Mining), 是指对文本中的主观信息进行辨认、提取和量化。文本情感分析被广泛地应用在用户评论、调查的反馈、网络和社交媒体内容上。一般情况下, 文本情感分析的目标是判断文本的作者的 (对于某一个主题的) 情感倾向。进行情感分析的方法有很多, 常见的方法有基于词库的方法, 基于机器学习的方法如朴素贝叶斯和支撑向量机等等。

2.1 基于词库的方法

基于词库的文本情感分析方法 (Lexicon based method) 使用已知的情感字典, 这样的情感字典包含情感词以及对应的情感极性。在已有先验的情感词与对应的极性时, 可以计算这些情感词在测试句子中出现的次数。一个测试句情感倾向为正向的概率可以计算为 $P(+|D) = \frac{a}{a+b}$, 其中 a 和 b 分别是正向词语和负向词语在该测试句子中出现的次数。

基于词库的方法的优点在于不需要已经标注的数据, 因此适用的场景比较广。它的缺点在于, 必须存在合适的情感字典, 且方法的准确率很依赖于情感字典的质量。对于纯粹以情感词的数目来判断句子情感倾向的方法, 没有办法处理句子的复杂语义结构, 如“不是很好”这样的情况。

2.2 基于机器学习的方法

2.2.1 朴素贝叶斯分类器

朴素贝叶斯分类器 (Naïve Bayes Classifier) 是一种条件概率模型, 给定一个测试句子, 通过计算这个句子属于每一个类别的条件概率来对句子进行分类。贝叶斯分类器中句子的表示是“一袋子词” (Bag of word), 即将句子看作一个无序的词集合。只考虑每个词的出现次数, 不考虑词的相对位置。贝叶斯分类器假设给定一个类别的条件下, 每个单词的出现概率都是相互独立的, 即 $P(w_i|C)$ 相互独立。在这个假设下, 句子属于某一个类别的条件概率就可以通过计算它的

每个单词属于该类别的联合概率分布得到，如下：

$$P(C_j|D) = \frac{P(w_1, \dots, w_n|C_j) * P(C_j)}{P(w_1, \dots, w_n)} = \frac{P(C_j) * \prod_{i=1}^n P(w_i|C_j)}{\prod_{i=1}^n P(w_i)} \quad (2.1)$$

朴素贝叶斯分类器用于情感分析的算法如下：

```

1 // Training Process;
2 for j ← 1 to class_num do
3     for i ← 1 to vocab_size do
4         | count(wi, Cj) ← number of word wi in sentences of class Cj;
5     end
6     count(Cj) ← number of sentences of class Cj;
7     word_count(Cj) ← number of word in sentences of class Cj;
8     P(Cj) ←  $\frac{\text{count}(C_j)}{\sum_{j=1}^{\text{class\_num}} \text{count}(C_j)}$ ;
9 end

10 // Predicting Function;
11 Input: a segmented sentence;
12 Output: predicted class;
13 for j ← 1 to class_num do
14     score(j) ← lg P(Cj);
15     for i ← 1 to sent_length do
16         | score(j) ← score(j) + lg  $\frac{\text{count}(w_i, C_j) + 1}{\text{word\_count}(C_j) + \text{vocab\_size}}$ ;
17     end
18 end
19 Return augmaxj(score(j))

```

算法 2.1: 朴素贝叶斯分类器

朴素贝叶斯分类器用于情感分析的优点在于只需要比较少的训练数据就可以训练出合理的 $P(w_i|C)$ 。缺点在于它依赖的独立性假设是一个很强的假设，很多时候并不符合数据的实际状况。此外，朴素贝叶斯分类器只考虑每个词语出现的次数，而放弃了每个词语的位置的信息。

2.2.2 支撑向量机

支撑向量机 (Support Vector Machine, 简称 SVM) 是一种常用的有监督学习的算法。由于它的每个测试样例的输入必须是一个向量, 因此, 在使用支撑向量机进行情感分类之前, 必须对文本进行特征抽取, 将句子转化为向量。Ti.idf 是一种常见的特征抽取方法。

2.2.2.1 Tf.idf 向量化

Tf.idf 全称为 Term frequency-inverse document frequency。与朴素贝叶斯相同, tf.idf 也是将文本看作“一袋子词”, 只考虑每个词语出现的次数。使用 tf.idf 进行向量化首先需要对训练数据建立词典, 对每个词出现的次数进行统计, 选取出现次数多于一个阈值的 m 个词, 根据这 m 个词在某一个句子和全部句子的出现次数, 来将句子转化为一个 m 维的向量。Tf 指词频, 其值通过单词在句子中出现的次数除以句子的长度获得, 因为一个词语在句子中出现的次数越多, 很有可能代表它占了很大比重; Idf 可以通过含有该词语的文档占总文档的比率取倒数再取对数得到, 因为一个词语如果出现的次数非常多, 如汉语中的“的”, 则它实际上含有的信息很少, 不能作为句子的特征。而出现次数较少的词则含有更多的信息, 如果一个词只在很少的几个句子中出现, 这几个句子很有可能具有很高的相似度。Tf.idf 向量化的结果的每一位由对应单词的 tf 和 idf 相乘得到。

2.2.2.2 使用支撑向量机进行情感分析

支撑向量机的目的是找到一个超平面 (hyperplane, 从而将训练数据最优地分割为两类。为了使这个超平面是最优的, 对于每一个类别, 支撑向量机算法寻求让最多的该类别的数据点处于超平面的一侧, 且最大化超平面的宽度 (margin, 超平面每一边距离它最近的点到该超平面的距离)。

对于一个二分类的问题, 如果训练数据是线性可分的, 那么可以找到两个相互平行的超平面将这两类数据完全分隔开, 且这两个超平面之间的距离最大化。这样, 两个超平面之间的距离就是 margin, 而要求的最优的超平面就位于这两个平行的超平面的正中间。将输入数据表示为 $(\mathbf{x}_i, y_i), i = 1, \dots, n, y \in \{-1, +1\}$ 。在进行正规化之后, 两个平行的超平面可以表示为 $\mathbf{w} * \mathbf{x} + b = -1$ 和 $\mathbf{w} * \mathbf{x} + b = +1$ 。要使两个超平面之间的距离最大, 就需要最小化 $\|\mathbf{w}\|$, 故问题转化为:

$$\text{Minimize } \|\mathbf{w}\| \text{ subject to } y_i(\mathbf{w} * \mathbf{x}_i - b) \geq 1, \text{ for } i = 1, \dots, n \quad (2.2)$$

在处理实际问题时，训练数据常常是不线性可分的。通过引入 Soft Margin，允许一部分点在两个超平面之间，可以使支撑向量机能够处理这样的问题。Soft Margin 的支撑向量机要求在最大化 margin 宽度的同时，最小化处于两个超平面之间的点到对应超平面的距离（即最小化这些点的误差）。问题转化为最小化函数：

$$J(\mathbf{w}, b) = \left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(\mathbf{w} * \mathbf{x}_i - b)) \right] + \lambda \|\mathbf{w}\|^2 \quad (2.3)$$

由此得到线性二分类支撑向量机的算法：

```

1 // Training Process;
2 Use tf.idf to transform all text data to vectors.;
3 Random initialize  $w$  and  $b$ ;
4  $\eta_t \leftarrow$  learning rate at time  $t$ ;
5 for  $i \leftarrow 1$  to  $epoch\_num$  do
6      $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta_t \nabla_{\mathbf{w}_t} J(\mathbf{w}, b)$ ;
7      $b_{t+1} \leftarrow b_t - \eta_t \nabla_{b_t} J(\mathbf{w}, b)$ ;
8 end
9 // Predicting Function;
10 Input: a segmented sentence;
11 Output: predicted class;
12 Use tf.idf to transform input sentence to a vector  $x_i$ ;
13 Return  $sgn(w * x_i + b)$ 

```

算法 2.2: 支撑向量机

支撑向量机通过一个核函数（kernel function）将输入数据映射到一个高维的特征空间，通过使用非线性的核函数，如 RBF 函数，可以得到非线性的分类器。上述的支撑向量机算法只适用于二分类的问题，对于多分类问题，可以使用 one-against-one 的方法，对每两个类别组合都建立一个二分类的支撑向量机分类器，一共建立 $\frac{class_num * (class_num - 1)}{2}$ 个 SVM 分类器；或者使用 one-against-all，对每个类别，建立一个该类别与其他所有类别之间的分类器。

2.3 小结

本章对文本情感分析的定义进行了说明，并且对目前常见的几种文本情感分析的方法进行了介绍。首先对于传统方法中常用的基于情感字典的方法，本章进行了简介和优缺点分析；对于目前非常流行，本课题也会采用做 **baseline** 实验的机器学习方法朴素贝叶斯分类器和支撑向量机，本章较详细地介绍了其原理和算法。

第 3 章 基于深度学习的文本情感分析

3.1 关于神经网络

神经网络是一种模仿人类大脑运行模式设计的算法。神经网络由多个层(layer)组成,每层有多个神经元。运算在每个神经元中进行,通过数据在神经网络中流过模仿神经信号的传递。神经元中的运算一般会包括线性运算和一个非线性的激活函数,也有特殊的 pooling 层和 dropout 层。以一个全链接层(Full-connected layer)为例,隐藏层中的每一个神经元接受上一层的所有输出作为它的输入 \mathbf{x} , 在进行了线性运算 $\mathbf{w} * \mathbf{x} + b$ 之后, 经过一个非线性激活函数 f , 得到输出 $f(\mathbf{w} * \mathbf{x} + b)$ 作为输入传到下一层。

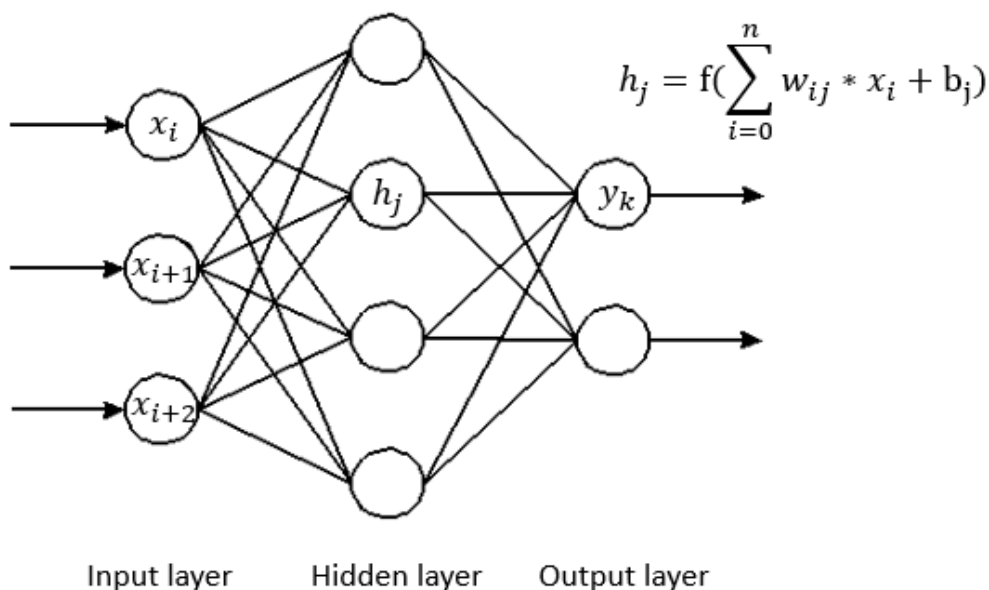


图 3.1 一个单全链接隐藏层的神经网络

神经网络具有非常强的表达能力,具有一个隐藏层的神经网络,在有足够多的神经元数目时可以模仿任意的函数(函数的值域必须在一个有限的区间内)。故近年来神经网络被广泛应用于模式识别、分类等领域,得到了广泛的研究,涌现了如卷积神经网络、复发神经网络等网络结构。通过词嵌入的方法,可以将单词表示为一个固定维数的向量,从而让文本数据可以作为神经网络的合适输入,

这样神经网络也可以用于像文本分类这样的问题中。

3.2 词嵌入

对于图像、声音相关的任务，一般都是处理稠密的、高维的输入数据，例如图像数据的像素级别的数据。但涉及到自然语言处理的问题时，传统的方法只能将每个单词表达为一个离散的原子的符号，如“猫”这个词表示为 Id_{234} ，“狗”这个词表示为 Id_{245} 。这样的表达只包含很少的信息，且忽视了不同单词间可能存在的关系，如“猫”和“狗”会被认为是没有关系的两个符号。这样的表达方法还会导致数据非常稀疏，进一步导致需要很多的数据量才能成功训练出统计模型。

词嵌入 (word embedding) 可以很好地解决上述的问题。词嵌入指将单个词语映射到高维特征空间 (一般 100 到 300 维) 的一个向量。单词之间的相互关系在词向量中可以得到体现，两个单词的相似度一般可以通过计算其词向量的 cosine 相似度得出。进行词嵌入的算法有两种：基于计数的方法在一个大的语料里统计出一些单词和它的邻居互相同时出现的次数然后将统计结果映射到一个小的、稠密的向量中；预测模型则是建立一个直接从一个词的邻居中预测出这个词的模型，在这个模型的过程中训练出一个最优的词向量。

Word2vec 是一种常用的高效率的词嵌入的算法，它是基于预测模型的算法，有 Skip Gram 和 Continuous Bag of Word 两种算法供选择，其区别在于预测模型的结构不同。本课题的词嵌入是使用 Word2Vec 训练出来的。

3.3 卷积神经网络

3.3.1 卷积神经网络的结构

卷积神经网络 (Convolutional Neural Network, 简称 CNN) 是一种特殊的神经网络结构，最初被应用在计算机视觉的领域。因为其模型结构非常适合图像数据的特征，卷积神经网络在计算机视觉领域取得了巨大成功，近年来，也被应用于自然语言处理领域的语义理解、语言模型、文本分类等问题上。

与全链接的神经网络层不同，卷积层的每个神经元并不以上一层的全部输出作为输入，而是只取局部的输出。卷积神经网络使用卷积核 (convolutional kernels, 又称为过滤器 filters)，每个卷积核只对应很小的感知域 (receptive fields, 大小可以为 2×2 , 3×3 等)，但这个感知域会贯穿输入数据的深度 (输入的第三维，

对应图像数据的 channel)。对于图像数据而言，第一层卷积隐藏层的每一个神经元，只对应对应图像输入的一个小的像素块，其大小等于卷积核的尺寸，包含全部的 RGB 三通道。这样的神经元只包含其感知域的输入数据的权重参数，只处理输入数据中的一小块局部数据并产生它的输出。

卷积层的另一个特点是参数共享。对于同一个卷积核的全部神经元，其参数是相同的，故卷积神经网络可以理解为一个卷积核在输入数据上进行“滑动”，不断进行卷积运算并产生一个二维的输出。由于每个卷积层一般会有多个卷积核（同一大小的卷积核可以有多个，随机初始化它们的参数以训练出不同的卷积核），所以每个卷积层都会接受一个三维的输入，并给出一个三维的输出（多个卷积核的输出结果叠加形成第三维）。这样，一个卷积层的输出在经过线性整流层（Rectified Linear Units layer, ReLU layer）作非线性化处理后，可以很方便地作为输入进入下一个卷积层。

卷积神经网络层相较于全链接层的一个优势在于不会有“维度诅咒”的问题。例如，对于一个 $100 \times 100 \times 3$ 的输入数据（图像数据或者经过词嵌入的文本数据都有这样的结构），在第一个隐藏全链接层中的每一个神经元都需要有 $100 \times 100 \times 3 = 30000$ 个权重参数，如果该层有 100 个神经元，则会产生 $100 \times 30000 = 3000000$ 个权重参数。而对于一个有 128 个 2×2 的卷积核的卷积层，只需要 $2 \times 2 \times 3 \times 128 = 1536$ 个权重参数。卷积神经网络的另一个优势在于可以提取输入数据的局部特征，每个卷积核只处理输入数据的一小块，多层卷积层后，图像数据的信息得到更好的归纳。

3.3.2 卷积神经网络应用于文本问题

卷积神经网络在计算机视觉领域的成功与图像数据的结构特点密切相关。与图像数据由像素点组成不同，文本数据由单词组成，因此，卷积神经网络不能直接应用于文本数据上。必须将文本数据映射成一个维度不高、稠密的矩阵数据形式。

通过对文本数据进行词嵌入，我们可以用一个句子中所有单词的词向量拼接出一个二维的矩阵。使用多个不同的词嵌入（要求词向量的维度相同），可以得到多个这样的二维数据，从而堆叠出一个三维的数据（第三维对应图像数据中的 channel）。这样，文本数据转化成为卷积神经网络可以接受的输入。

应用于文本数据输入的卷积神经网络与图像数据的卷积神经网络存在着一些差别，主要在于卷积核的大小。应用于图像数据的卷积神经网络卷积核大小可

以为 2×2 , 3×3 , 2×3 等, 长和宽没有严格的限制。但对于接受文本数据输入的卷积层, 其卷积核的宽度一般等于词向量的宽度, 即卷积核会占满词向量这一维, 只在句子长度这一维上滑动。这是因为每一行都是原输入数据中的一个单词, 如果在单词内部使用更小的卷积核进行卷积运算的话, 相当于“破开”了一个单词。在不了解词向量每一位的意义的情况下, “破开”词向量并不合理。此外, 卷积核的长度 (对应每次进行卷积运算时覆盖的单词数) 也不会太长, 一般很少会超过五。因为文本中每个单词相较于图像中的像素, 粒度要大一些, 如果卷积核的感知域覆盖了太多单词, 就不存在“提取局部特征”的优势了。

应用于文本领域的卷积神经网络与传统的 n -gram 方法有相似之处, 它们都会提取连续出现的单词作为句子或者文档的局部特征。相较于传统的 n -gram 方法, 卷积神经网络有如下几个优势: 首先卷积神经网络不用像 n -gram 那样建立非常大的词典 (在比较大的语料上, 3 -gram 已经需要建立庞大的词典才能正常工作); 此外, 卷积神经网络可以利用 GPU 的硬件优势, 进一步得到加速, 因为卷积运算作为一个计算机图形学中非常基础的运算, 通常在 GPU 的硬件层面上就会给予特别支持。

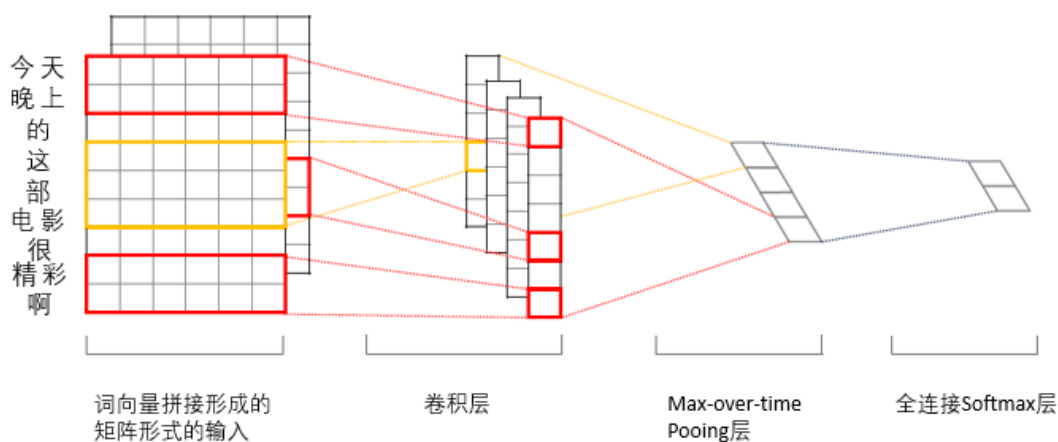


图 3.2 卷积神经网络应用于文本分类, 图像部分引用自^[4]

3.3.3 具体算法

用 $w_i \in \mathbf{R}^k$ 来表示句子中的第 i 个单词对应的长度为 k 的词向量, 一个长度为 n 的句子可以表示为:

$$x_{1:n} = [x_1; x_2; \dots; x_n] \quad (3.1)$$

使用 $x_{i:i+j}$ 来表示词向量 $x_i, x_{i+1}, \dots, x_{i+j}$ 连接的结果。一个卷积核 $w \in \mathbf{R}^{h \times k}$ ，将作用在一个包含 h 个单词的“窗口”上来产生一个特征。例如，一个特征 c_i 可以从单词 $w_{i:i+h-1}$ 中通过下式产生：

$$c_i = f(\mathbf{w} * \mathbf{x}_{i:i+h-1} + b) \quad (3.2)$$

这里的 b 为偏差值 (bias)， f 是一个非线性函数 (如 ReLU)。随着窗口在整个句子上滑动，这个卷积核可以作用在每一个可能的窗口上，产生了如下的特征序列：

$$\mathbf{c} = [c_1, c_2, \dots, c_{n-h+1}] \quad (3.3)$$

$\mathbf{c} \in \mathbf{R}^{n-h+1}$ 。在卷积层得到的特征序列上加以一个 **max-over-time pooling**，只保留特征序列中的最大值 $\hat{\mathbf{c}} = \max\{\mathbf{c}\}$ ，这是为了只保留每个特征序列中最重要的特征。在经过 **max-pooling** 层后，每个卷积核只保留了一个特征。多数个卷积核 (可以有不同窗口大小) 会产生一系列的特征，这些特征连接起来，作为最后的全连接 **Softmax** 层的输入。此外，在输入层和 **max-pooling** 层之后，我们的模型中加入两个 **dropout** 层。**Dropout** 层会随机地筛选出一部分输入，将其值设为 0，再进行输出。**Dropout** 层的目的是通过随机去掉一些神经网络中的信号传递以减少过拟合。

3.4 复发神经网络

复发神经网络 (Recurrent Neural Network, 简称 RNN) 是另一个常见的神经网络结构，复发神经网络中的信号不仅会向前传播，而且会传回给自身，因此被称为 **recurrent**。具体到文本问题上，复发神经网络单元每次在处理完一个单词后，除去产生一个输出外，还会记住一部分信息。在处理下一个单词时，会根据当前输入以及之前记住的信息来产生下一个输出，并更新记住的信息。

Long Short Term Memory，简称 **LSTM**，是一种特殊的复发神经网络。它使用一个“遗忘门”来选择性的忘记一部分记住的信息，同时使用一个“输入门”来记住一部分非线性话处理后的上一轮留下的输入。**LSTM** 的优势在于他可以处理长距离的依赖关系。

复发神经网络由于能够在文本数据流过网络时“记住”一些之前的信息，因此，可以很好利用到输入数据的全局特征和顺序特征。在文本分类这个问题上，他可以利用词语的位置特征，挖掘整个句子的信息。复发神经网络在文本相关的问题上一直有很好的表现。

由于本课题只使用 LSTM 作为一个用于比较的实验，重点仍然是卷积神经网络在情感分析问题上的应用，故只简单地介绍 LSTM 的原理。关于 RNN 和 LSTM 的具体内容，可以参考^[5]和^[6]。

3.5 小结

本章对深度学习的方法如何应用于情感分析问题进行了说明。首先简介了神经网络的基本原理，然后词嵌入技术进行了说明；之后对就卷积神经网络相关内容进行了详细介绍包括其原理、如何应用于文本问题以及具体的算法；最后简介了作为 baseline 实验的 LSTM 的基本原理。

第 4 章 基于卷积神经网络的影评分类实验

4.1 数据集与预处理

4.1.1 数据的获取与概况

本课题采用的数据集是来自豆瓣的中文影评数据。使用微软必应（Bing）搜索的 Cosmos 分布式处理平台获取。来自 Cosmos 的数据是原始的 HTML 文本，使用 HtmlAgility 库对其进行解析，提取出有用的影评、评分等信息，整理成 JSON 格式的文件。数据的基本信息如表格4.1所示。

整理生成的短评数据示例如表格4.2所示。

短评数据每条的长度不超过 140 个字符，每条都带有对应评分（1 到 5 星）；长评数据没有字数限制，同样每条都包括 1 到 5 星的评分。

选择来自豆瓣的中文影评作为语料的优势在于，短评和长评都可以看作有标注的数据，这就意味着我们不再需要再人工地对数据进行标注就可以在它们上面进行有监督的学习。此外，豆瓣的短评有长度的限制，易于转换为固定大小的输入矩阵，故我们使用短评作为主要的实验数据。对于获取到的长评数据，虽然它们也是有标注的数据，但由于它们之间长度相差很大，而且相对于情感信息比较密集的短评，长评中可能有很多谈论具体电影情节的，不带情感色彩的内容。所以本课题不直接使用长评数据进行实验，而是使用他们进行了词嵌入的训练。因为长评数据本身的数据量很大，其语境和用语习惯与短评非常接近，所以使用它们作为训练词嵌入的语料非常适合。

4.1.2 分词与训练词嵌入

获得了 JSON 格式的数据后，由于 JSON 中的 Text 域是原始的短评文本，所以还需要进行分词和训练词嵌入才能将这些数据转化为神经网络可以接受的形

表 4.1 短评数据数量表格

全部短评数量	全部长评数量	短评来源用户数量	短评对应电影数量
124591	100000	42517	4887

表 4.2 短评数据示例

Text	Username	Rate	Cid	Vote	MovieName	MovieId
这么高分我有点不懂	X.Lee	3	1005243594	12	拿起枪的简	10760385

注：整理后的短评 JSON 数据示例。长评数据格式与之类似，但多了 Title 域

表 4.3 实验数据划分表格

	训练集	验证集	测试集
评分预测实验	40000	10000	10000
二分类实验	28204	7059	7016

式。本课题采用的中文分词工具是斯坦福大学自然语言处理研究组的 Stanford Word Segmenter，该分词工具基于的算法是使用条件随机场（Conditional Random Field）去做 sequence model。

将短评数据和长评数据进行分词后，将其作为输入数据进行词嵌入的训练。训练词嵌入采用的算法是基于 Continuous Bag of Word 的 Word2vec，这是一种高效率的词嵌入训练算法。本课题使用 gensim 库中对 Word2vec 的实现，选取词向量长度为 100，从短评和长评数据中训练出了中文词语的词嵌入。

4.1.3 实验设置

本课题的实验分为两种，第一种是由中文电影短评预测该短评对应的评分（1 到 5 星），本课题以短文本五分类的方法来实现对影评的预测；第二种是将五分类的中的 3 星评论去除，以 1 星和二星评论作为负面评论，四星和五星评论作为正面评论，由短评的内容预测其评论的倾向（正面还是负面），将问题转化为个短文本二分类来做。

在使用短评数据进行实验之前，我对短评根据其作者发布的影评数量进行了筛选，只保留了发表影评数量超过 5 的较活跃用户的影评。符合要求的影评共 67255 篇。抽取其中 60000 篇，进行了 shuffle 后按表格 4.3 划分了训练集，验证集和测试集（由于第二类实验需要去除三星的评论，故第二类实验的数据规模与第一类有所区别）

4.2 不同的卷积神经网络结构

对于实验中的所有神经网络模型，均采用每个尺寸 128 个卷积核的设置。卷积核宽度固定等于词向量宽度，如卷积核长度为 1、2、3，则长度为 1、2、3 的卷积核各有 128 个，共 384 个不同的卷积核；在输入数据和 pooling 层后各设置一个 dropout 层，dropout 的比率在 0.0, 0.3, 0.5 这三个档位进行网格实验，选出最优的一组。

实验设置在卷积神经网络的输入数据如何断句、通道数以及卷积网络的层数等超参数的设置上有所分化，主要分为以下几类实验：

4.2.1 单层单通道

结构如第三章中所述，只有一层卷积层，卷积的输出经过 pooling 和 dropout 直接传给最后进行分类的全连接 Softmax 只使用一个词嵌入，故卷积神经网络的输入的第三维宽度只有一，对应图像数据只有一个通道。这里的词嵌入层也作为神经网络的一部分，词向量的值也会通过 backpropagation 更新。

我们使用单层单通道网络在词语层面（使用中文分词器断句）和字层面（在每个字断句）的输入上都进行了实验，其中词语层面的词嵌入使用 Word2vec 的训练结果初始化，字层面则随机初始化。

4.2.2 单层双通道

同样只有一层卷积层，但有两个词嵌入，故传给卷积层的输入第三维大小为 2，即有两个通道。这两个词嵌入都使用 Word2vec 的训练结果初始化，其差别在于只有其中一个在训练过程中保持更新，另一个则始终不变。单层双通道只在词语层面的输入上进行了实验。

4.2.3 双层卷积神经网络

使用两个卷积层，在字层面的输入上进行实验。旨在第一层完成字组合的特征提取，第二层提取更高层面的特征。

4.3 Baseline 实验

本课题进行了三个 baseline 实验用作对比和分析。分别为朴素贝叶斯分类器，基于 Tf.idf 特征抽取的支撑向量机，和 LSTM 神经网络。由于在本文的第二

章文本情感分析已经对 **baseline** 方法的基本原理进行了介绍，这里只说明具体的实验设置。

朴素贝叶斯分类器使用基于计数的方法计算 $P(w_i|C)$ 和 $P(C)$ 。对于单词的出现次数设置阈值 10，即一个单词只有在训练数据中出现的次数超过十次才能被用在预测中。具体的概率的计算方法和算法流程见算法2.1。

支撑向量机使用 Tf.idf 来进行特征抽取，将句子转化为一个一维的向量。对于 Tf.idf 的词典设置阈值 50，因为向量化的结果的长度等于词典大小，故需要选取一个比较高的阈值以压缩向量长度，选取 50 为阈值得到的向量长度约 2000。使用 RBF 的核函数，Soft Margin 的支撑向量机设置。对于评分预测的五分类的实验，使用 one-against-all 的方法，将支撑向量机由二分类拓展为 5 分类。具体的算法流程见算法2.2。

LSTM 神经网络使用 Tensorflow 实现。输入数据为经过词嵌入的得到的词向量序列，单层的 LSTM 之后，经过一个平均值池化 (mean-pooling)，对每个词向量的输出取平均值，得到一个长度为隐藏层宽度的一维向量。该一维向量再通过一个 Softmax，得到分类的预测结果。

4.4 小结

本章对本课题所做的工作进行了说明。首先说明了数据的来源，对数据的预处理，已经实验的总体设置；然后说明了实现的不同神经网络结构的区别；之后，说明了 **baseline** 实验的设置。

第 5 章 实验结果与分析

5.1 实验结果

Baseline 实验结果如表格5.1所示；卷积神经网络评分预测（五分类实验）实验结果如表格5.2所示、二分类预测实验结果如表格5.3 所示。

5.2 结果分析

可以看到使用中文分词器分词的效果要好于按字断句，说明对中文语言文本分词的确可以提取出文本中很多的特征。而基于字断句的话，在训练数据不非常大的情况下（40000 句仍然只能算比较少的数据量），或是模型复杂程度不够高的情况下，可能会导致一些信息或特征无法被提取出来，因而导致结果不如分词后的数据。但同时也可以注意到，在卷积核的设置包含了 3 或 4 这样比较大的窗口时，基于字断句的输入也会有很好的效果。这说明了说明卷积神经网络的确有提取局部特征（在这里体现为提取出字的组合的特征）的能力。

对于同一卷积神经网络结果，不同的卷积核设置也会明显地影响实验结果。对于输入使用分词器断句的实验，准确率在卷积核长度达到 2 时达到最大，加上长度为 3 的卷积核准确率反而下降；对于输入使用按字断句的实验，加上长度为 3 的卷积核后准确率得到了提高，但再加上长度为 4 的卷积核后（数据未列在上表上）准确率也下降了。给处理文本问题的卷积神经网络增加更大的卷积核可以看作一种 Tradeoff，一方面增加更大的卷积核有可能提取出更多的句子特征，另一方面，这样也会使整个模型更加容易过拟合。实验结果表明，在进行了分词的数据上，连续三个中文词语中可以提取的特征已经很少，由此带来的“更容易过拟合”的缺点对结果的影响甚至超过了提取三个连续词语的特征带来的影响。

表 5.1 Baseline 实验结果

	Naive Bayes	SVM	LSTM
评分预测（五分类）实验	41.54%	38.7%	43.62%
二分类实验	79.64%	78.31%	81.77%

表 5.2 评分预测实验准确率结果

Filters Sizes	CNN_char	CNN	CNN_2_channel	CNN_2_layer
1	40.41%	43.04%	-	-
1,2	42.11%	44.12%	43.29%	40.61%
1,2,3	42.49%	43.28%	-	-

注：实验设置见第四章第二、第三节相关内容，CNN_char 指使用按字断句的输入；双通道和双层只在单层的最优设置上进行了实验。

表 5.3 二分类实验结果

Filters Sizes	CNN_char	CNN	CNN_2_channel	CNN_2_layer
1	77.89%	81.42%	-	-
1,2	79.74%	82.05%	81.33%	77.29%
1,2,3	80.47%	81.50%	-	-

注：实验设置见第四章第二、第三节相关内容，CNN_char 指使用按字断句的输入；双通道和双层只在单层的最优设置上进行了实验。

表 5.4 Confusion Matrix

	1	2	3	4	5
1	0	306	166	86	61
2	229	0	303	79	22
3	280	687	0	992	340
1	59	127	647	0	587
1	43	53	153	368	0

注：横坐标为 Ground Truth，纵坐标为预测值。

单层双通道的模型也取得了很好的实验结果，但其最好的结果没有超过单层单通道的最好实验结果。这与我们开始时的期望略有出入，可能是由于双通道的模型更容易过拟合。双层卷积层模型的结果也超过了 **baseline** 算法，但落后于单层卷积层的两种模型。这与图像数据领域的结果有很大差别，可能是因为文本数据每个单词相较于图像数据的像素，包含的信息要多得多。因此，即使在单层卷积层的模型也可以提取出很多的信息，也会有很好的结果。而多层之后，反而由于抽象的层数太高、过拟合的机会增大，可能会导致效果反而不如单层的网络。

另外也可以注意到，五分类问题的结果准确率普遍比较低，不超过 45%。这是由于对影评进行五分类本身是一个比较困难的问题，与本课题数据集相近的 SST-1 数据集（同样是影评数据，分为五类，分别为非常负面、负面、中立、正面和非常正面），目前最前沿的成果也只能达到 48.7% 的准确率。这个问题准确率难以踢得很高是因为每个影评者自己的指标不同，同样一句影评，对于一些人来说会选择给四星，另一些人则会给五星、三星。此外，中间三档影评的区分度很低。本课题使用的数据集未经过人工筛选，只根据作者发过的影评数设置了一个阈值。故数据集中可能存在评论与打分不符，或是一些没有意义的垃圾信息。这些评论也会对结果产生影响。

表格5.4是五分类问题最后结果的 **confusion matrix**，即预测错误的分布。可以看到，大多数的错误分布在主对角线附近，即出现在相邻的两个分段。尤其是中间二、三、四三个阶段，出现的错误非常多。这印证了我们上一段的分析。

因为五分类的结果准确率较低，难以应用在实际问题中。我们又增加了二分类的试验，可以看到，二分类实验的准确率很高，达到了 82% 以上。同时可以

注意到，各种模型在二分类问题上的相对表现与五分类的相当，即五分类问题中表现好的模型，在二分类中仍有很好的表现。没有出现只在某一个问题上表现特别好或特别差的异常现象，实验结果与预期相符。

5.3 小结

本章对实验结果进行了总结和分析。比较了卷积神经网络的不同卷积核选取带来的结果差异；比较和分析了单层单通道和更复杂网络结构的实验结果；具体分析了什么评分预测的准确率较低。

第 6 章 总结

6.1 全文总结

本文研究了使用卷积神经网络对中文影评数据进行评分预测的问题。使用收集自豆瓣的中文影评进行了多组实验，比较和分析了卷积神经网络的不同结构对结果的影响。

本文首先对研究问题的背景和意义进行了简述；然后，本文对于情感分析的常用方法进行了介绍和分析，并对本文用作 **baseline** 实验的朴素贝叶斯分类器算法、支撑向量机算法的原理进行了详细描述，并整理出了它们的算法；之后，本文对神经网络模型在情感分析问题上的应用进行了说明，主要说明了神经网络的一般结构，词嵌入的相关概念和算法，重点对卷积神经网络及其在文本方面的应用进行了说明；本文详细描述了进行实验所使用的数据，以及各组实验的设置；最后，本文对实验的结果进行了汇报，并对结果进行了比较和分析。

从实验结果可以看出，基于卷积神经网络的方法在情感分析问题上有着不错的表现，使用宽度为 1, 2 的卷积核的基于词语断句单层卷积神经网络模型超过了全部的 **baseline** 算法；其他结构的卷积神经网络也取得了很好的实验结果。

6.2 未来工作展望

在以后的工作中。首先可以考虑进一步对卷积神经网络的结构进行优化，以达到更好的实验效果，本文的实验中只对少数几个模型进行了实验。还有很大的调节空间；从实验结果中可以看出，LSTM 模型在中文影评情感分类的问题上也有着很好的效果（在 **baseline** 算法中是最高的，也超过很多组非最优的卷积神经网络模型），而在文献^[7]中，Tang 等人提出了一种将卷积神经网络与 LSTM 相结合的方法。未来的工作可以考虑将这个方法应用于我们的中文电影评论的评分预测。

第五章中分析了数据中噪音较多是影响实验结果的一个重要因素，未来可能考虑人工对数据进行一次筛选，去掉一些噪音。这样可以提供实验的准确度。

此外，评分预测并不一定要当作一个分类问题来解决。实际上，考虑到影评评分的五个分段并不是完全独立的，这个问题使用回归（**regression**）的方式来解

决也有一定的道理。Drake 等人在 2008 年提出过一种“情感回归”的方法^[8]。未来的工作可以考虑先从支撑向量机回归模型开始，尝试用回归的方式解决这个问题。

参考文献

- [1] Taboada M, Brooke J, Tofiloski M, et al. Lexicon-based methods for sentiment analysis[J]. *Computational linguistics*. 2011, 37 (2): 267–307.
- [2] Pang B, Lee L, Vaithyanathan S. Thumbs up? Sentiment Classification using Machine Learning Techniques[J]. *eprint arXiv:cs/0205070*. May 2002.
- [3] Turney P D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews[C]//Association for Computational Linguistics, Proceedings of the 40th annual meeting on association for computational linguistics. [S.l.]: Association for Computational Linguistics, 2002: 417–424.
- [4] Kim Y. Convolutional neural networks for sentence classification[J]. *arXiv preprint arXiv:1408.5882*. 2014.
- [5] Socher R, Perelygin A, Wu J Y, et al. Recursive deep models for semantic compositionality over a sentiment treebank[C]//Citeseer, Proceedings of the conference on empirical methods in natural language processing (EMNLP): volume 1631. [S.l.]: Citeseer, 2013: 1642.
- [6] Sundermeyer M, Schlüter R, Ney H. Lstm neural networks for language modeling.[C]//Interspeech. [S.l.], 2012: 194–197.
- [7] Tang D, Qin B, Liu T. Document modeling with gated recurrent neural network for sentiment classification.[C]//EMNLP. [S.l.], 2015: 1422–1432.
- [8] Drake A, Ringger E, Ventura D. Sentiment regression: Using real-valued scores to summarize overall document sentiment[C]//IEEE, Semantic Computing, 2008 IEEE International Conference on. [S.l.]: IEEE, 2008: 152–157.