**北 京 邮 电 大 学**

**本科毕业设计（论文）中期进展情况检查表**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 学院 | | | 计算机学院 | 专业 | | 计算机科学与技术 | | 班级 | 2014211307 |
| 学生姓名 | | | 王超 | 学号 | | 2014211310 | | 班内序号 | 07 |
| 指导教师姓名 | | | 刘晓鸿 | 所在单位 | | 计算机学院 | | 职称 | 副教授 |
| 设计（论文）题目 | | | （中文）基于循环神经网络的影评情感分析系统的设计与实现 | | | | | | |
| （英文）Design and implementation of the movie-review sentiment classification system based on RNN | | | | | | |
|  | 主要内容: (毕业设计（论文）进展情况，字数一般不少于1000字)  一、豆瓣短评爬虫  使用python Requests库对豆瓣电影进行爬虫，分析豆瓣网页请求，发现豆瓣分类网页中，后台发来json response，前端从而进行解析显示。通过对该url发送请求来获取豆瓣电影的唯一标记id以及url。  之后，根据电影id访问对应电影的短评首页，通过BeautifulSoup分析网页抓取短评所在元素以及该短评对应评分，从而根据评分分类positive和negative标签，将其按评分保存到各个txt中。  二、预处理  训练word2vec模型需要一个较好的提取语料特征的语料库，考虑到语料库的要求，使用了搜狗实验室提供的2012全网新闻数据(1.54G)。  由于该语料库为XML格式，并且其中文为GBK格式，而非通用的UTF-8格式，需要先进行转码及抓取指定元素内容的步骤，本步骤主要使用iconv及grep命令实现。  之后执行常规分词所要执行的预处理操作，首先过滤常见符号、数字、英文，之后使用jieba默认模型进行分词，之后使用该分词后语料库训练Word2Vec模型，并将其vocabulary 及其对应关系模型保存为numpy array，便于后续处理。得到的Word2Vec模型形状为572297\*300。  其次，对训练集进行预处理操作。对训练集过滤非中文字符，使用jieba模型进行分词，再根据网上开放的一份较大的停用词表进行去除停用词。  三、LSTM模型对照实验  RNN模型能够将信息持久化，能够处理序列数据。但对于长序列数据，会出现梯度下降、梯度爆炸等问题，导致无法得到优良的结果。而LSTM的提出解决了RNN在长期依赖方面的不足。本次实验主要设置了LSTM与传统RNN进行比照。 | | | | | | | | |
| 目前已完成任务 | 在设置循环神经网络模型参数前，为了确定神经网络步长，读取训练集每篇语料词长，以及频次，使用matplotlib可视化得知语料内容长度满足正态分布，绝大部分语料长度在50以内，因而确定步长为50。超过50的则丢弃，不足则以0补齐。  实验中设置的LSTM主要为4层，输入层、隐藏层、dropout层和输出层。输入层，即Embedding层，设定输入参数句长为50、向量维度（400）等。第二层，隐藏层，设定LSTM单元数为64。第三层为dropout层，减少过拟合。第四层为输出层，即softmax层。  输入，通过word2vec模型提取特征矩阵idx，将语料数值化，每条语料得到一个50\*400的矩阵。  训练集经过shuffle打乱，防止模型不稳定，损失函数跳跃严重。  输入向量经过embedding，馈送到dynamic\_rnn。之后主要采用tensorflow框架的lstm单元来完成循环神经网络模型的训练。  传统RNN模型和GRU模型参数设置与上面LSTM模型基本相同，将使用的tensorflow的lstm单元更换成了rnn单元，稍稍根据输出矩阵形状调整矩阵，主体与LSTM相同，仅设置了一层隐藏层。      四、结果分析  使用matplotlib可视化测试集的accuracy指标后可以看到，上左图是LSTM模型，右图是传统RNN模型，下左图是GRU模型。在设置了均衡训练集的情况下，三者训练时间相近，在经过20轮训练后已经趋于稳定。单层RNN的准确率在81%，2层的RNN准确率反而下降到了76%，而单层LSTM的准确度有96%，单层GRU的准确度高达97%。可以看到，LSTM和GRU相对传统RNN具有极大优势。根据文献得知，GRU和LSTM性能在很多任务上不分伯仲。因此，在长序列文本的情感分析的问题上，将主要基于LSTM或GRU模型进行优化训练。  五、应用  搭建了一个根据输入评论内容可以返回情感分析结果的网页。前端提交输入的内容后，后端调用神经网络模型得出预测值，从而返回情感分析结果。技术上主要使用python Flask Web框架构建一个wsgi服务器，渲染引擎主要使用Jinja2后端渲染。不过由于模型较大，加载模型时间约5分钟，单次提交评论测试，响应时间略久，达到了近10秒之多。 | | | | | | | | |
| 是否符合任务书要求进度 | | | | | | | | |
| 尚需完成的任务 | 添加ROC、F1等评估标准对不平衡数据测试进行评估。  增加对照模型如GRU、IndRNN，从而进行优化  完成毕业论文，并准备答辩 | | | | | | | | |
| 能否按期完成设计（论文） | | | | | | | | |
| 存在问题和解决办法 | 存  在  问  题 | 损失函数过拟合问题 | | | | | | | |
| 拟  采  取  的  办  法 | 1. 调整模型参数 2. 增加训练数据 3. 添加dropout层 | | | | | | | |
| 指导教师签字 | |  | | | 日期 | | 年 月 日 | | |
| 检查小组意见 | | 负责人签字： 年 月 日 | | | | | | | |

注：可根据长度加页。