《Python程序设计》课程项目报告

学号	姓名	成绩
2018302100035	梁皓然	
2020301021123	陈滨琪	
2020302191545	王昭翔	
2020301181049	徐纬	

指导老师	黄文斌
完成日期	2021年11月30日

*PART1:*梁皓然

1. 选题

- 程序名称：基于朴素贝叶斯的情感分析
- 选题原因：同时在上另一门课《机器学习》，和python语言结合起来，应该可以做出点有趣的成果：）

2. 需求分析

- 背景：互联网外卖以服务、快捷为卖点，用户的评论与反馈对商家、平台都很重要。
- 功能：本文根据数据中的评论，采用朴素贝叶斯算法来分析用户情感，将用户评论划分为“好评”，“差评”。

3. 设计

程序的主要模块

- 标准库/扩展库的使用
- 分词
- 构造词语列表
- 糅合好评与差评
- 构造词表

  获取训练集中所有不重复的词语构成列表
- 计算单词是否出现并创建数据矩阵
- 训练数据
- 测试数据

标准库/扩展库的使用

```
from bayes import Bayes
import jieba
import pandas as pd
from numpy import *
```

1. 朴素贝叶斯模块并没有使用IPython官方库的bayes模块，而是使用GitHub用户stevewang0提供的替换库。下载地址：MLInAction。

   使用该库的原因是它比官方提供的库更为简洁，代码以及其能实现的功能非常明确，适合像我这种机器学习入门新手。

   （对该py文件我会在另一个Jupyter Notebook文档中详细说明，这一部分很关键，它是本文实现贝叶斯分类器的核心。）

2. jieba库的使用在本小组创建的《文本挖掘基础知识》中文分词模块有详细说明，这里不再赘述。

3. pandas的运用详情见一篇CSDN博客：《import pandas as pd什么意思》。

4. NumPy包的核心是ndarray对象。它封装了n维同类数组。很多运算是由编译过的代码来执行的，以此来提高效率。

4. 关键代码

- 构造词表

  获取训练集中所有不重复的词语构成列表

```
myVocabList = Bayes.createVocabList(listOPosts)
```

- 计算单词是否出现并创建数据矩阵

```
trainMat = []
for postinDoc in listOPosts:
    trainMat.append(Bayes.setOfWords2Vec(myVocabList, postinDoc))
```

- 训练数据

```
p0V, p1V, pAb = Bayes.trainNB0(array(trainMat), array(listClasses))
```

- 测试数据

```
while True:
    inputS = input(u'请输入您对本商品的评价：')

    testEntry = wordCut(inputS)
    thisDoc = array(Bayes.setOfWords2Vec(myVocabList, testEntry))
    print('评价: ', Bayes.classifyNB(thisDoc, p0V, p1V, pAb))
```

5. 运行效果

```
请输入您对本商品的评价：服务挺不错的
评价：  1
请输入您对本商品的评价：个人觉得性价比很高
评价：  1
请输入您对本商品的评价：服务员态度非常恶劣
评价：  0
请输入您对本商品的评价：吃完就拉肚子，差评
评价：  0
```

1 表示好评 0 表示差评

图1-1 测试结果

PART2:陈滨琪

1. 选题

- 程序名称：基于textblob的文本情感分析与数据可视化
- 选题原因：个人希望走多维度发展道路，兼修多领域和跨领域的学科。又因为专修英语，采集了一些国际知名大学关于计算社会科学的论文进行文本分析。

2. 需求分析

- 背景：随文本分析逐渐成为热门研究方向，预处理文本与情感分析、词频筛选逐渐成为开启研究的基本功。
- 功能：程序大概分为三个模块，首先是对文本(txt)文件进行分词、移除文本停用词、正则表达式清理文本、词干提取与词形还原五个操作，同时附加了命名实体处理板块（因为会对词频分析产生干扰，在词频部分设置新的停用词库又遇到了困难，因而没有添加到总程序里而是另外附件了）第二个模块是基于textblob实现对预处理后文本的情感分析，第三个模块是基于TF-IDF算法将预处理后文本按算法筛选出500个词，排列成词云。

3. 设计

程序的主要模块

- 分词
- 移除文本常见停用词
- 正则表达式清理文本
- 词干提取与词形还原 （附：命名实体处理）
- 将预处理好的文本形成一个新的txt文件
- 使用textblob对文本进行情感分析
- 使用textblob对文本进行情感分析
- 将关键词排列为词云

标准库/扩展库的使用

```
import nltk from nltk.corpus
import stopwords
from nltk import stem
from nltk.stem.wordnet import WordNetLemmatizer
import re
from textblob import TextBlob
import sys
import jieba
import jieba.analyse
from optparse import OptionParser
from wordcloud import WordCloud,ImageColorGenerator
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt
```

4. 关键代码

- 定义停用词函数

  循环遍历句子中的每一个单词并检查是否有停用词，最后将结果组合。

```
def remove_stop_words(splitresult,stop_words):
    return" ".join([word for word in splitresult if word not in stop_words])
```

- 利用正则表达式去除掉文本中除数字、字母和空格外的字符

```
def clean_text(x):
    temp=re.sub(r'([^\s\w]|_)+','',x).split()
```

- 对整个文件的情感进行分析并打印

```
blob=TextBlob(data)
print(blob.sentiment)
```

- 设置词云参数，使词云根据背景颜色变化

```
wordcloud = WordCloud(mask=mask,font_path =
'OLDENGL.TTF',max_words=500,mode='RGBA',background_color=None).generate(result)
```

5. 运行效果

图2-1 词云

```
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Administrator\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Sentiment(polarity=0.09255807581621198, subjectivity=0.3941443331674252)
```

图2-2 情感分析

*PART3*王昭翔

1. 选题

- 程序名称：基于NRC词典的文本情感分析
- 选题原因：NRC词典拥有比其他文本分析工具更加丰富的情感种类

2. 需求分析

- 背景：《你好，李焕英》是由贾玲执导的电影，在全球上映。符合中国观众口味的电影是否能够博得国际审美的认可？于是打算用NRC词典情感分析揭示这一结果
- 功能：通过国际影评平台的相关评论数据，用NRC词典对其进行情感分析，最后显示文本中八种情感成分的权重并且将数据可视化

3. 设计

程序的主要模块

一、 数据预处理——合成向量

- 在指定目录下筛选所有的一维情感数据文件
- 将不同情感的数据合并并且向量化，将单词单个情感的一维数据合并成为单词的八维情感向量数据

二、 情感分析，数据可视化

- 将目标文本运用nltk进行分词
- 对词语列表进行遍历

- 对每个词语的情感向量叠加，得出文章情感向量
- 运用pandas与matplotlib实现数据呈现与可视化

标准库/扩展库的使用

```
import xlrd
import xlsxwriter
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

1. 由于词语的情感向量储存在excel文件中，程序要多次打开excel文件
2. 同上，写入excel文件
3. os模块提供了与操作系统即电脑系统之间进行交互功能，实现自动化提取目录下文件
4. 利用numpy库中的array数组，实现情感向量的数学叠加，提高机器运行效率
5. 利用pandas库实现数据读取
6. 利用matplotlib绘制彩色饼图，实现数据可视化

4. 关键代码

- 寻找目录下的xls文件，路径合成列表

```
for file in file_data:
    file_type=os.path.splitext(file)[1]#提取拓展名
    if file_type ==".xls":
        all_file_list.append(root_path+"\\"+file)
```

- 合成词语情感向量

  词语在文件中存在则找到相应的值并赋值，不存在则赋值为0

```
if data[m][0] in word_in_file:
    weight=file[word_in_file.index(data[m][0])][1]
    data[m].append(weight)
else :
    data[m].append(0)
```

- 通过建立的索引目录找到word对应的情感向量

```
for v in all_emotion:
        all_word.append(v[0])
for word in wordlist:
        if word in all_word:
            weight=all_emotion[all_word.index(word)][1:]
```

- 向量叠加

  利用numpy库中的array

```
v1=np.array(weight)
v2=np.array(emotion_count[1])
emotion_count[1]=list(v1+v2)
```

- 合成饼图，实现数据可视化

```
df.plot(kind='pie',y='weight',legend='False',colors=colors,labels=labels)
plt.show()
```

5. 运行效果

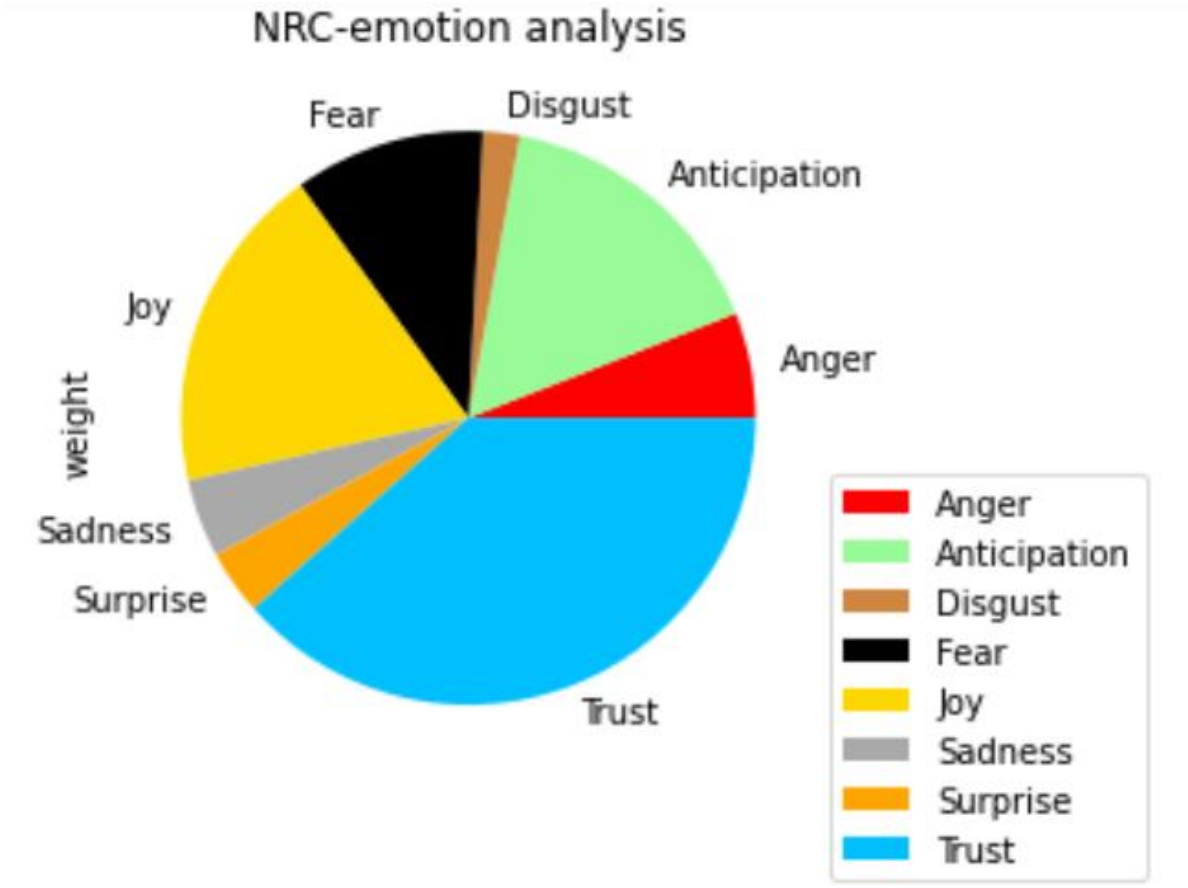

图3-1 NRC-emotion analysis

	emotion	weight
0	Anger	0.059361
1	Anticipation	0.161952
2	Disgust	0.020664
3	Fear	0.107528
4	Joy	0.185577
5	Sadness	0.044314
6	Surprise	0.036271
7	Trust	0.384334

图3-2 情感分布

- 从上文结果可以看出，前三位分别是trust、joy、aticipation。相对正面的情感成分居多，说明国外友人对《你好，李焕英》持肯定态度。
- trust代表信任的情感，我认为其中应该包涵了国外友人对中国电影制作的赞赏与信心，以及对母爱亲情的美好歌颂——这是人类无国界，共通的情感。
- joy代表喜悦，这揭示了《你好，李焕英》作为喜剧的成功，能够感染大众。
- anticipation代表期待，这是国际友人对于中国的下一部更加优秀的作品的期待。

*PART4*徐纬

1. 选题

- 随着新媒体技术的不断更新，网络论坛（BBS）这种缺少即时性、便利性的社交媒体似乎走出大众视野，逐渐被边缘化。
- 但诸如杭州西湖网、武汉东湖社区等地方网络论坛发展已久，其受众依然广泛，且已经成为民众参与城市治理的重要途径。

2. 需求分析

- 通过对武汉市网民在东湖社区发帖的情感分析，将武汉市民的舆情监测大数据化，了解武汉市基层社区治理中的成效与不足。

3. 设计

爬取数据并导出

通过网络爬虫爬取“荆楚网-东湖社区”中“大武汉”板块发帖主题。

共爬取37973条数据，由于数据过大，分为两份.xlsx文件。

详见压缩包内“数据”文件夹。

统计数据

制作折线统计图，统计每年的发帖量变化。由于该板块在2011年创立，故从2012年开始计算。

结果显示，除2011年外，每年发帖数量均在1000条以上，且波动较大，最大为2013年，可能与中央领导班子更换有关；最小为2021年，可能是网民舆情发表阵地由传统网络论坛逐渐转向门户APP。

数据清理

只提取标题中的名词、形容词，因为这两类词有明显的情感倾向。

进行词性标注；n 代表名词，adj 代表形容词。

先选出名词形容词所在的行，再选择索引，再根据索引从上面合并的结果中选出此条评论的所有词语

正负情感标注

我们使用2007年HOWNET发布的“情感分析用词语集”作为情感词表。HOWNET的构建秉承还原论思想，即所有词语的含义可以由更小的语义单位构成，而这种语义单位被称为“义原”（Sememe），即最基本的、不宜再分割的最小语义单位。

HOWNET构建了包含2000多个义原的精细的语义描述体系，并为十几万个汉语和英语词所代表的概念标注了义原。其中包括“中文正面评价”，“中文正面情感”，“中文负面情感”，“中文负面评价”等词表。

将“中文正面评价”，“中文正面情感”分别合并，并给每个词语赋予初始权重为1，作为正面评论情感词表；
“中文负面情感”，“中文负面评价”合并，并个每个词语赋予初始权重为-1，作为负面评论情感词表。

在提供的词表基础上进行优化，添加部分词语，以匹配情感词代码。

由于汉语中存在多重否定现象，即当否定词出 现奇数次时，表示否定；偶数表示肯定。按照汉语习惯，搜索每个情感词前两个词语，若为奇数，则调整为相反的情感。

绘制正负情感词云图

从图看出评论数据预处理后，分词效果较为符合预期。

从正面情感词云看出“支持”“喜欢”“赞”“感谢”等词出现频率较高，没有掺杂负面情感词语。

从负面情感词云看出“奇葩”“垃圾”“黑心”“坑”等词出现较多，说明负面情感很好的抽取了出来。

值得注意的是，“湖北”“社区”在两图中都出现了，说明不同网民在不同情境下可能对同一事物产生不同看法。

LDA主题模型

如果一篇文档有多个主题，则一些特定的可代表不同主题的词语就会反复出现。

此时，运用主题模型，能够发现文本中使用词语的规律，并且把规律显示的文本联系到一起，以寻求非结构化的文本集中的有用信息。

通过 LDA 主题模型，能够挖掘数据集中的潜在主题，进而分析数据集的集中关注点及其相关特征词，代码回复关键词获取查看。

首先建立词典及语料库，再主题数寻优，确定最适合的主题数，查看主题间平均余弦相似度，在此项目中，主题数为2时达到了最低。

获取主题

最后得到主题，一个列表代表一个主题，里面是一个主题中最可能出现的10个词语。

4. 关键代码

- 制作发帖数量统计折线图

```
time_year = data['时间'].value_counts()
year = list(time_year.index)
count = (time_year.values)
time_year = list(zip(year,count))
time_year = sorted(time_year,key=lambda x: x[0])
print(time_year)

# 绘制折线图
x = [time[0] for time in time_year]
y = [time[1] for time in time_year]
```

- 构造LDA主题寻优函数

```
# 构造主题数寻优函数
def cos(vector1, vector2):  # 余弦相似度函数
    dot_product = 0.0;
    normA = 0.0;
    normB = 0.0;
    for a, b in zip(vector1, vector2):
        dot_product += a * b
        normA += a ** 2
        normB += b ** 2
    if normA == 0.0 or normB == 0.0:
        return (None)
    else:
        return (dot_product / ((normA * normB) ** 0.5))

    # 主题数寻优
```

5. 运行效果

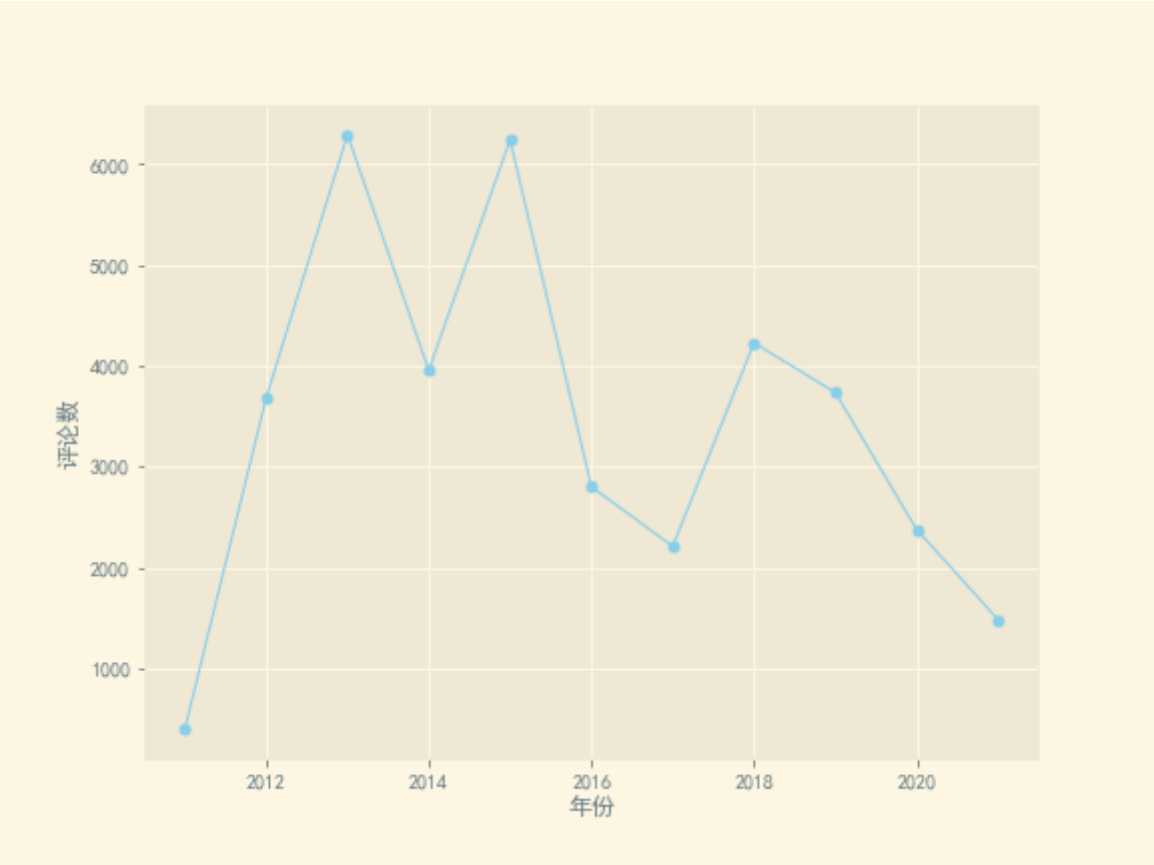

图4-1 评论数走势

6. 总结

*PART1:*梁皓然

选题动机

最开始接触情感分析这个概念是在某论坛上看到了一篇爬取环球时报主编推特推文做文本分析的帖子，用R语言实现了词云，情感分析，主题模型(LDA)。同时我每天都会手动打开前女友微博浏览一番，算是当乐子：）前女友发博的频率很高，而且她写的内容会让我觉得这个人非常的自大，而且喜怒无常，精神状况堪忧。但是“让人觉得”是非常主观的论断，这个时候我想到了不久前看到的那篇帖子。我想也许我能用更为直观的数据和图表得到更具说服力的结论。正好这学期有Python程序设计这门课，实践起来对我而言是一举两得的。

那个时候我没有任何Python语法基础，但是我仍然义无反顾地去做了。大概用一天就成功调用微博爬虫爬取了她的所有微博文本，第二天就把词云图，情感分析，主题模型做了出来。词云图里最显眼的字就是“我”和“自己”，初步印证我觉得她非常自大的判断，当然这种分析非常不靠谱。另外根据画出来的情感分析柱状图，“喜”“怒”“哀”“乐”“悲”“惊”“惧”的分布是非常均匀的，一个正常人怎么可能在短时间内情绪波动地这么严重，给人直观感受这是个精神病：)另外根据主题模型显示所有文本经过聚类后最高频的几个主题都是与自我高度相关的琐事，初步印证了我认为她是个十分无趣而且乏味的人并且十分匮乏与他人同理心的看法。

我所建立的都是很简陋的模型，很多都是网上各种copy的，也未经调试，因此精度肯定是很低的。但这对一个刚入门的新手而言已经能产生很大的满足感和成就感了，不得不说兴趣是最好的老师，让我主动地克服很多困难强迫自己学习之前从未接触过的算法和第三方库，如果是自己不感兴趣的专业课，大概率是拖到考前一两周找两套卷子做完应付了事，绝对不会像那些天带着被兴趣点燃的激情探索自己未知的领域。那两天写出来的代码和生成的结果其实已经可以打包当作最后的课程项目，但是这样以来我在整个课程中的参与度就会变得非常低，所以就有了后面的基于Python的文本挖掘四人小组，进而才让我明白要把四个人凝聚起来共同做出一个成果是多么的不容易。

管理中遇到的交流问题

- 我们这个四人小组应该是周五下午班最早形成的一个小组，成组这么早是为了提前沟通，把这个项目做的更完善。我之前从未如此认真对待过任何一门课的课程项目，可能是因为我在最开始接触这门语言时得到的满足感有关。我想把这个课程项目做的非常完善，但事实证明我这样的想法是非常naive的。原因如下：
    - 如果最后做出来是个非常完整的项目，那么它至少得满足两个条件其中之一：我对整个项目的流程十分了解；经过大家的充分讨论得到一个周全的执行方案。前者是不满足的，因为我也没有经验，其实我当时是寄希望于后者的。而后者为什么也无法满足呢，后文会谈；
    - 如果这个项目的完整性是可能的，那么我得先了解每个人的长处，做事方式和价值观，并且这些条件得是匹配的才行。然而现实是我在群里大概说了下会做文本挖掘相关的课题，有人感兴趣，然后就因为这个组了队。其实这只满足了最基本的前提条件。当然，对于有些人而言这只是个3学分课程中占比20%的一小部分而已，所以也有了“为什么要花这么多精力呢?我还有很多其他事情，这一部分应付一下就得了。”的想法。这种想法我是非常理解的，但是在实践过程中我发现一旦一个组织里有人有这种想法，那么结局就是根本做不出来有价值的成果，也很难在这个过程中获益。这里让我不得不想到硅谷CEO们的教父格鲁夫的那本书《只有偏执狂才能生存》，想把一件事做得完美，就必须“偏执”，而“偏执”就意味着你得花费大量的时间和精力纠结如何把整个项目做的更好，这不是每一个人都愿意做的，所以导致最后的成果上限非常低；
    - 组织应该是建立在信任的基础上，那么第一步就是彼此了解。我在尝试对这个项目的相关的人际关系负责。但效果并不好，我的单方面的尝试并没有得到良性的正反馈，大多数人的表现是冷漠的，不负责任的。而且同时伴随着拖延，效率低下的问题。

基于以上的考量，而且我是非常不愿意催促一个消极怠工的人按我的要求完成工作的，因为我能料想到我会收到一大堆搪塞的借口。最后我把课程项目切分为完全独立的四个部分，~~因为我不想对别人的低效率和消极态度继续负责了。~~

有关学习文档

在最开始搜索各种关于文本挖掘的材料的时候，就一直苦恼于没有一个专门写给新手的学习文档，那为什么我们自己不做一个呢？而且我自然而然地认为大家都会积极地参与到学习文档的构建和讨论中来，自学的同时可以锻炼使用Jupyter Notebook和撰写技术文档的能力，这在我看来是非常赚的一件事，并且从中获得的乐趣可以抵消很大一部分探索陌生领域的不适感。我在接触有关文本挖掘的基础概念时发现了很多有趣的知识和学习资料（文档、视频、博客等），我觉得把他们汇集到一个文档中不管是对自己梳理整个学习文本挖掘的逻辑还是对分享给其他初学者而言都是十分有意义的事情。但是好像只有我一个人觉得有意义，这个时候我已经对别人低参与度的原因没有兴趣了，对此我表示很可惜：（

小结

学习一门新的编程语言最合适学习方法应该是自学。只有具备一定的编程能力的时候才有可能通过分工合作完成一个更立体、更完整的项目，而且这个时候也需要更多的技巧，比如沟通能力，阅读文档文献的能力，管理分工合作的能力。通过这大半个学期的实践我发现大多数人（包括我自己）并不具备这些能力，故没有做出一个高质量项目的可能。这个时候需要更多专业人士的分阶段指导，但是我们课程的师生比和老师同学的状态，我觉得我们并不具备这样的条件：（

最后引用巴菲特的一句名言作为给其他将要做课程项目并且准备招募组员的同学的忠告：

> "当你雇用某人，要看他是否具备三种品质：正直诚实、聪明能干和精力充沛。如果缺少第一种品质，那后二种品质会要你的命。"

*PART4:*徐纬

一次跨学科的Python之旅

机缘巧合之下，四位不同学部、不同学科的同学在一堂计算机编程课上组了队。跨学科的背景使我们从不同角度分析Python能实现的种种可能——数据可视化、机器学习、文本挖掘、爬虫……最终我们敲定了一个共同主题——情感分析。将看似枯燥且冗长的文本提取出来，判断感情倾向，无疑是自动化的一大步。我们决定结合本专业专长以及兴趣选定不同的主题。

作为马院的学生，社区舆情一直是我感兴趣的一个方向，并且已经申报了大创课题“画出最大同心圆：制度优势与治理效能良性互动的实践逻辑——武汉市社区党领共治的基层治理格局研究”。但之前所做的研究多为使用社会科学中的问卷及量表进行量化分析，或是对特定人群进行采访，进行质性分析。如若能够使用大数据手段，爬取武汉网民的舆情数据，一定会为这份大创项目添光增彩不少。

说干就干，我选取了武汉市最有代表性的网络论坛——东湖社区作为爬取对象，在“大武汉”板块爬取了三万多条贴子，作为数据来源。事实证明，该数据很有代表性，词汇具有鲜明的“汉味”，能够鲜明地反应武汉居民在城市生活中的喜怒哀乐。

当然，在项目过程中也出现了一定困难，首先是由于对Python库使用不太熟练，在调试、读取、修改源csv文件的时候花了很长时间。其次，在如何运用停用词表、使用四份不同类型的情感词表方面，我学习了很长时间。

当然本项目也存在一定的不足，首先是对情感分析还不够透彻，提取的部分主题词并不能很好地反映居民情感。其次，本项目采用的HOWNET词库较老，时效性不强，对于最新的网络词汇可能存在错判现象。

7. 评语

教师评语	