南京工業大學
NANJING TECH UNIVERSITY

用户数据采集与关联分析

（结课作业）

吴志祥

18205185639

1030624832@qq.com

南京工業大學
NANJING TECH UNIVERSITY

01 文本数据分析

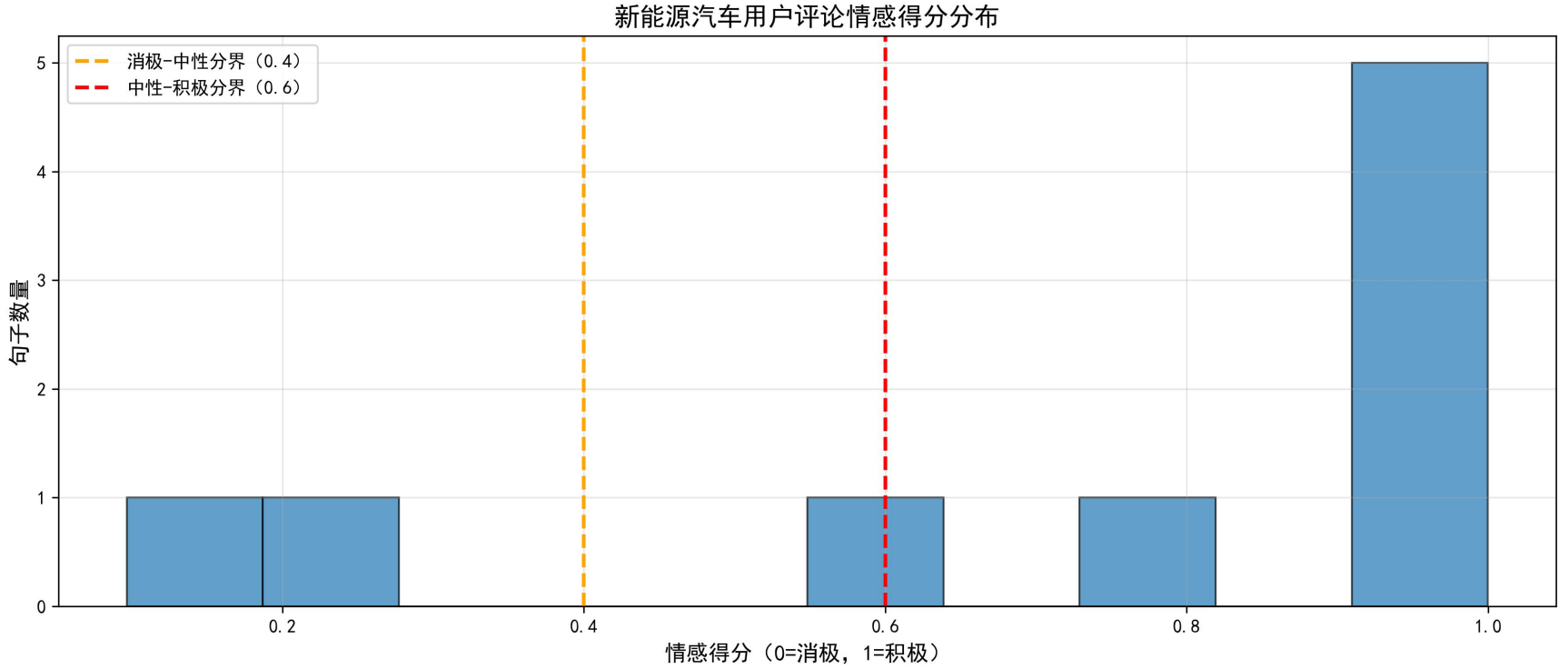
吴志祥

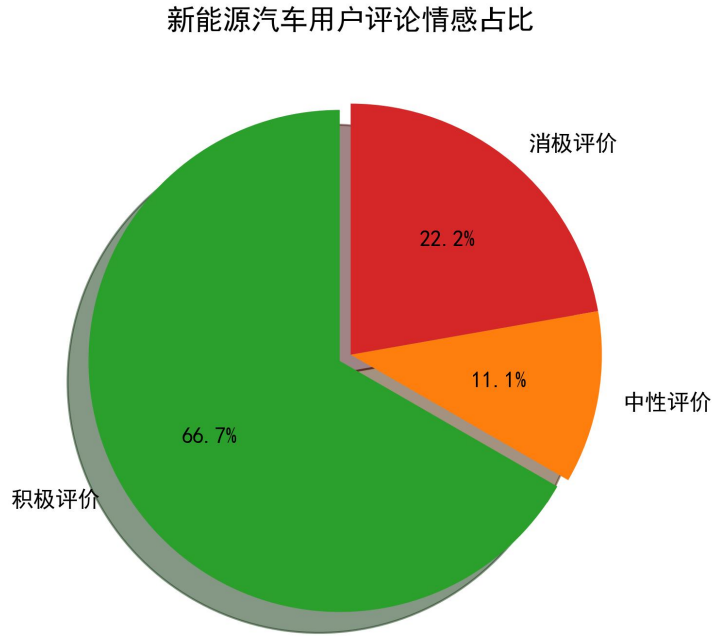
18205185639

cnwzx2012@njtech.edu.cn

第四

```python
from snownlp import SnowNLP
import matplotlib.pyplot as plt
import numpy as np

# ---------------------- 1. 准备待分析文本 ----------------------
# 示例：新能源汽车用户评论文本（包含积极、消极、中性评价）
text = """
这款新能源汽车的续航真的超出预期，充一次电跑500公里完全没问题，太满意了！
内饰做工很精致，智能驾驶功能也很好用，操作简单流畅，推荐大家购买。
价格有点贵，比同级别燃油车贵了3万多，性价比一般。
刹车有点软，试驾时感觉制动距离偏长，安全性让人担心。
空间表现中规中矩，后排坐三个成年人刚好，不算拥挤也不算宽敞。
售后服务态度很好，有问题响应很快，维修保养费用也合理。
续航虚标严重，实际只能跑350公里左右，和宣传的差距太大了。
外观设计很时尚，回头率很高，身边朋友都问是什么车型。
充电速度有点慢，快充需要1.5小时才能充满，不太方便长途出行。
整体来说还不错，虽然有小缺点，但在新能源车里算是表现优秀的了。
"""

# ---------------------- 2. 文本分割与情感计算 ----------------------
# 按句子分割文本（基于中文标点）
sentences = [s.strip() for s in text.replace( old: "\n",  new: "").split("。") if s.stri

# 逐句计算情感得分（0-1），并分类
emotion_scores = []
positive_sentences = []  # 积极（得分≥0.6）
neutral_sentences = []   # 中性（0.4 < 得分 < 0.6）
negative_sentences = []  # 消极（得分≤0.4）

for sent in sentences:
    score = SnowNLP(sent).sentiments
    emotion_scores.append(score)
    if score >= 0.6:
        positive_sentences.append(sent)
    elif score <= 0.4:
        negative_sentences.append(sent)
    else:
        neutral_sentences.append(sent)

# 统计各类情感占比
total = len(sentences)
pos_ratio = len(positive_sentences) / total * 100
neu_ratio = len(neutral_sentences) / total * 100
neg_ratio = len(negative_sentences) / total * 100

# ---------------------- 3. 生成可视化图表 ----------------------
plt.rcParams['font.sans-serif'] = ['SimHei']  # 解决中文显示问题
plt.rcParams['axes.unicode_minus'] = False    # 解决负号显示问题

# 创建2x1子图（情感分布直方图 + 情感占比饼图）
fig, (ax1, ax2) = plt.subplots( nrows: 2,  ncols: 1, figsize=(12, 10))

# 子图1：情感得分分布直方图
ax1.hist(emotion_scores, bins=10, color='#1f77b4', alpha=0.7, edgecolor='black')
ax1.axvline(x=0.4, color='orange', linestyle='--', linewidth=2, label='消极-中性分界（0.4）')
ax1.axvline(x=0.6, color='red', linestyle='--', linewidth=2, label='中性-积极分界（0.6）')
ax1.set_xlabel('情感得分（0=消极，1=积极）', fontsize=12)
ax1.set_ylabel('句子数量', fontsize=12)
ax1.set_title('新能源汽车用户评论情感得分分布', fontsize=14, fontweight='bold')
ax1.legend()
ax1.grid(alpha=0.3)

# 子图2：情感占比饼图
labels = ['积极评价', '中性评价', '消极评价']
sizes = [pos_ratio, neu_ratio, neg_ratio]
colors = ['#2ca02c', '#ff7f0e', '#d62728']
explode = (0.05, 0, 0)  # 突出积极评价

wedges, texts, autotexts = ax2.pie(
    sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%',
    shadow=True, startangle=90, textprops={'fontsize': 11}
)
ax2.set_title('新能源汽车用户评论情感占比', fontsize=14, fontweight='bold')

# 调整子图间距
plt.tight_layout()

# 保存图片（高清格式）
plt.savefig( *args: 'emotion_analysis_result.png', dpi=300, bbox_inches='tight')
plt.show()

# ---------------------- 4. 输出分析结果 ----------------------
print("=== 情感分析结果汇总 ===")
print(f"总评论句子数：{total} 句")
print(f"积极评价：{len(positive_sentences)} 句（{pos_ratio:.1f}%）")
print(f"中性评价：{len(neutral_sentences)} 句（{neu_ratio:.1f}%）")
print(f"消极评价：{len(negative_sentences)} 句（{neg_ratio:.1f}%）")
print("\n=== 典型句子示例 ===")
print("积极评价示例：", positive_sentences[0] if positive_sentences else "无")
print("消极评价示例：", negative_sentences[0] if negative_sentences else "无")
```





经济与

第四讲 情感分析

2. 完成sentiment_analysis_1，sentiment_analysis_2，sentiment_analysis_3，sentiment_analysis_4，4份代码。做截图，并简要做出分析。

```
!pip install snownlp
!pip install -U textblob
!python -m textblob.download_corpora
```

英文

```
# 英文调出TextBlob NLP处理包
```

Features

- Noun phrase extraction
- Part-of-speech tagging
- Sentiment analysis
- Classification (Naive Bayes, Decision Tree)
- Language translation and detection powered by Google Translate
- Tokenization (splitting text into words and sentences)
- Word and phrase frequencies
- Parsing
- n-grams
- Word inflection (pluralization and singularization) and lemmatization
- Spelling correction
- Add new models or languages through extensions
- WordNet integration

```
text = "I am happy today. I feel sad today."
```

```
from textblob import TextBlob
blob = TextBlob(text)
```

```
blob
```

```
# 居然不动的打印出来了？
# 实际上已经把文本分成了句子了，看一看
blob.sentences
```

```
blob.sentences[0].sentiment
```

```
# 上面的结果什么意思呢？
# 情感极性0.8，主观性1.0。说明一下，情感极性的变化范围是[-1, 1]，-1代表完全负面，1代表完全正面。
# 我表达的是我很高兴，那么这个结果是对的
```

```
blob.sentences[1].sentiment
```

```
# 整段文本的情感呢？
blob.sentiment
```

```
# 你可能会觉得没有道理，怎么一句“高兴”，一句“沮丧”，合并起来最后会得到正向结果呢？
# 首先不同极性的词，在数值上是有区别的，我们应该可以找到比“沮丧”更为负面的词汇，而且这也符合逻辑，谁会这么“天上一脚，地下一脚”手舞地描述自
```

中文

```
# 中文分析，用的是SnowNLP包
```

Features

- 中文分词（Character-Based Generative Model）
- 词性标注（TnT 3-gram 隐马）
- 情感分析（现在训练数据主要是买卖东西时的评价，所以对其他的一些可能效果不是很好，待解决）
- 文本分类（Naive Bayes）
- 转换成拼音（Trie树实现的最大匹配）
- 繁体转简体（Trie树实现的最大匹配）
- 提取文本关键词（TextRank算法）
- 提取文本摘要（TextRank算法）
- tf，idf
- Tokenization（分割成句子）
- 文本相似（BM25）
- 支持python3（感谢erning）

```
text_cn = u"我今天很快乐。我今天很愤怒。"
```

```
#注意在引号前面我们加了一个字母u，它很重要，因为它提示Python，“这一段我们输入的文本编码格式是Unicode，别搞错了哦”，至于文本编码格式的细节，有
```

```
from snownlp import SnowNLP
```

```
senti_cn = SnowNLP(text_cn)
```

```
# 看看snownlp包的分句能力
for sentence in senti_cn.sentences:
    print(sentence)
```

```
senti_cn_1 = SnowNLP(senti_cn.sentences[0])
```

```
# 一个细节上的问题，英文是s.sentiment，中文是s.sentiments，多了一个s
# 另外，在句法上和英文的也略有不同，比如直接用语句：senti_cn.sentences[0].sentiments是会报错的
senti_cn_1.sentiments
```

```
senti_cn_2 = SnowNLP(senti_cn.sentences[1])
```

```
senti_cn_2.sentiments
```

这里你肯定发现了问题——“愤怒”这个词表达了如此强烈的负面情感，为何得分依然是正的？

这是因为SnowNLP和textblob的计分方法不同。SnowNLP的情感分析取值，表达的是“这句话代表正面情感的概率”。也就是说，对“我今天很愤怒”一句，SnowNLP认为，它表达正面情感的概率很低很低。

这样解释就是ok了

```
senti_cn.sentiments
```

```
# 整个句子，貌似就有问题了
```

作业 ¶

注：设计课程的时候，可以把作业单独拿出来，放在另外一个文档中

第四讲 情感分析

2. 完成sentiment_analysis nt_analysis_4，4份代码。做截图，并简要做 结分析。

情感分析-高级-时间序列

工作步骤

- 1 能够知道如何处理excel文件
- 2 怎么用snownlp处理一句话的情感，生成情感值
- 3 如何批量的按行、快速生成每行文本的情感值 .apply
- 4 怎么把情感的趋势线画出来

```
准备工作，之前已经安装了snownlp整个包，还要安装ggplot这个包
! pip install ggplot
```

In [1]:
```
import pandas as pd # pandas用来处理表格数据的工具包
```

In [2]:
```
df = pd.read_excel("restaurant-comments.xlsx")
```

In [3]:
```
df.head()
```

Out[3]:

	comments	date
0	这辈子最爱吃的火锅，一星期必吃一次啊！最近才知道他家还有免费鸡蛋羹..................炒鸡好吃炒鸡嫩...	2017-05-14 16:00:00
1	第N次来了，还是喜欢?...... 从还没上A餐厅的楼梯开始，服务员已经在那迎宾了，然...	2017-05-10 16:00:00
2	大姨过生日，姐姐定的这家A餐厅的包间，服务真的是没得说，A餐厅的服务也是让我由衷的欣赏，很久...	2017-04-20 16:00:00
3	A餐厅的服务哪家店都一样，体贴入微。这家店是我吃过的排队最短的一家，当然也介于工作日且比较晚...	2017-04-25 16:00:00
4	因为下午要去天津站接人，然后我倩前几天就说想吃A餐厅，然后正好这有，就来这吃了。 来的...	2017-05-21 16:00:00

```
# 注意这里的时间列。如果你的Excel文件里的时间格式跟此处一样，包含了日期和时间，那么Pandas会非常智能地帮你把它识别为时间格式，接着往下做就可以了。

反之，如果你获取到的时间只精确到日期，例如"2017-04-20"这样，那么Pandas只会把它当做字符串，后面的时间序列分析无法使用字符串数据。解决办法是在这里加入以下两行代码：

from dateutil import parser
df["date"] = df.date.apply(parser.parse)

这样，你就获得了正确的时间数据了
```

In [4]:
```
text = df.comments.iloc[0]  # iloc是索引（用来定位到指定的位置），这里就是第一条评论文本
```

In [5]:
```
text
```

Out[5]: '这辈子最爱吃的火锅，一星期必吃一次啊！最近才知道他家还有免费鸡蛋羹··················炒鸡好吃炒鸡嫩啊！！新出的红皮土豆也好好吃，还有炸酥肉，秒杀任何火锅店啊！服务员太可爱，告诉我们半份豆花是4块儿，一份豆花是6块儿，点两个半份比较合适，太实在了哈哈哈，每次妈妈说开心果好吃服务员都给我们打包带走??希望A餐厅早日出咖喱锅，期待ing······'

In [6]:
```
from snownlp import SnowNLP
```

In [7]:
```
s = SnowNLP(text)
```

In [8]:
```
s.sentiments
```

Out[8]: 0.4244401030222834

情感分析数值可以正确计算。在此基础上，我们需要定义函数，以便批量处理所有的评论信息。

In [9]:
```
def get_sentiment_cn(text):
    s = SnowNLP(text)
    return s.sentiments  # snownlp是通用的，就没那么准确
```

然后，我们利用Python里面强大的apply语句，来一次性处理所有评论，并且将生成的情感数值在数据框里面单独存为一列，称为sentiment。

In [10]:
```
df["sentiments"] = df.comments.apply(get_sentiment_cn)  #这行代码很有意思！
```

In [11]:
```
df.head(100)
```

Out[11]:

	comments	date	sentiments
0	这辈子最爱吃的火锅，一星期必吃一次啊！最近才知道他家还有免费鸡蛋羹..................炒鸡好吃炒鸡嫩...	2017-05-14 16:00:00	4.244401e-01
1	第N次来了，还是喜欢?...... 从还没上A餐厅的楼梯开始，服务员已经在那迎宾了，然...	2017-05-10 16:00:00	4.506914e-01
2	大姨过生日，姐姐定的这家A餐厅的包间，服务真的是没得说，A餐厅的服务也是让我由衷的欣赏，很久...	2017-04-20 16:00:00	1.000000e+00
3	A餐厅的服务哪家店都一样，体贴入微。这家店是我吃过的排队最短的一家，当然也介于工作日且比较晚...	2017-04-25 16:00:00	1.182003e-01
4	因为下午要去天津站接人，然后我倩前几天就说想吃A餐厅，然后正好这有，就来这吃了。 来的...	2017-05-21 16:00:00	8.712261e-01
5	A餐厅是我们的定点单位，单位大聚小聚都来这，所以享受VIP待遇，来了不用等位。这的菜品新鲜，...	2017-05-05 16:00:00	9.980202e-01
6	这两天风特别大，家里也没什么可吃的，外卖也不知道吃什么！又懒得出去，发现好久没吃火锅啦！灵机...	2017-04-22 16:00:00	9.820584e-01
7	A餐厅真的是天津每一个店我都来过无数次了！ 今天觉得我拍的图很好看hhhh于是就来发条...	2017-05-20 16:00:00	6.392747e-03
8	今天又来到新开路的A餐厅，食材还是非常新鲜，羔羊肉味道鲜美，墨鱼滑味道新鲜！我们一家三口吃的...	2017-05-12 16:00:00	4.267672e-02
9	服务态度一如既往的好，看我东西多，主动帮忙提东西，所有的服务员都主动打招呼，非常热情。我点的...	2017-05-07 16:00:00	6.235047e-01
10	服务好赞！外送小哥实际A餐厅自己配的，送之前一个电话说别急已经在送的路上，送到后也一个电话说...	2017-04-10 16:00:00	1.965386e-04
11	服务一如既往的这么赞、论食材是火锅中的精粹，论服务也绝对是龙头老大！人多最爱来吃A餐厅了、每...	2017-04-23 16:00:00	9.999997e-01
12	一直特别特别特别爱A餐厅真的服务太贴心了细节决定一切呀用餐的时候接了个电话无意中跟朋友说今天...	2017-05-07 16:00:00	9.999998e-01
13	服务没的说啊，非常的周到啊，从始至终不用自己喊服务员，主动的倒饮料，帮我们下肉，吃完还给我们...	2017-05-14 16:00:00	9.999515e-01
14	口味还不错?点了一份蛋炒饭，加荷包蛋?溏心的。捞派捞面?还不错，菌汤番茄?的都好吃。炸馒头一...	2017-05-14 16:00:00	9.999862e-01
15	话说我真的很久很久没有吃A餐厅了，朋友想吃，我俩就过来了。她喜欢番茄锅，所以我们点了牛油麻辣...	2017-04-25 16:00:00	1.215408e-03
16	提前预订的因为孩子生日孩子们又喜欢那的口味服务我就不都说了一如既往的优质每位服务员态度都是那...	2017-04-27 16:00:00	1.000000e+00
17	A餐厅新开路店是在天津所有A餐厅门店当中，我最愿意光临的，因为这里停车方便，排队人也不像其他...	2017-03-17 16:00:00	9.827555e-01
18	再次来享受A餐厅的服务，真是好，等位的时候可以下棋等餐，而且有美甲做，需要排队很久，柠檬水不...	2017-05-06 16:00:00	8.931110e-01
19	服务和味道一如既往的好，主要最爱吃他家DIY的小料！这次还用手机扫了个公众平台用一楼的照片打...	2017-04-14 16:00:00	9.998846e-01
20	A餐厅的服务一直很好，光顾很多次了，这次去是小伙伴们帮忙预定，为我庆生，A餐厅的工作人员还提...	2017-05-21 16:00:00	1.000000e+00
21	今天下班，和朋友说好去吃A餐厅?，到那没有排队就可以直接进去吃??，点了它这新品，红皮土豆，...	2017-04-14 16:00:00	9.100772e-01
22	1.环境还好，算中等。 2.服务不错，越来越好，有生日布置，原来送小蛋糕，现在改果盘中...	2017-05-04 16:00:00	9.270364e-01
23	A餐厅火锅 上周五带对象去吃的，我本人对火锅不是特别爱吃，吃不了辣，对象喜欢吃辣的，就...	2017-03-16 16:00:00	9.999957e-01
24	这次是在情人节当天过去的，以前从来没在情人节正日子出来过，不是因为没有男朋友，而是感觉哪哪人...	2017-02-20 16:00:00	6.334066e-08
25	一直喜欢A餐厅，去过其他店，这次陪父母来他家体验。排队等待时服务员很贴心的送来小吃，很温馨。...	2017-05-22 16:00:00	9.999988e-01
26	家门口经常去服务没的说环境也挺好的	2017-05-24 01:50:00	6.348401e-01

新的列sentiment已经生成。我们之前介绍过，SnowNLP的结果取值范围在0到1之间，代表了情感分析结果为正面的可能性。通过观察前几条数据，我们发现点评网站上，顾客对这家分店评价总体上还是正面的，而且有的评论是非常积极的。

但是少量数据的观察，可能造成我们结论的偏颇。我们来把所有的情感分析结果数值做一下平均。使用mean()函数即可。

In [12]:
```
df.sentiments.mean()
```

Out[12]: 0.6987503312852683

整体接近0.7，正面。我们再来看看中位数值，使用的函数为median()。

In [13]:
```
df.sentiments.median()
```

Out[13]: 0.9270364310550024

```
我们发现了有趣的现象——中位数值不仅比平均值高，而且几乎接近1（完全正面）。
这就意味着，大部分的评价一边倒表示非常满意。但是存在着少部分异常点，显著拉低了平均值。
下面我们用情感的时间序列可视化功能，直观查看这些异常点出现在什么时间，以及它们的数值究竟有多低。
```

我们需要使用ggplot绘图工具包。这个工具包原本只在R语言中提供，让其他数据分析工具的用户羡慕得流口水。幸好，后来它很快被移植到了Python平台。

我们从ggplot中引入绘图函数，并且让Jupyter Notebook可以直接显示图像。

In [14]:
```
%pylab inline
```

```
%pylab is deprecated, use %matplotlib inline and import the required libraries.
Populating the interactive namespace from numpy and matplotlib
```

```
D:\Anaconda\Lib\site-packages\IPython\core\magics\pylab.py:166: UserWarning: pylab import has clobbered these variables: ['text']
`%matplotlib` prevents importing * from pylab and numpy
  warn("pylab import has clobbered these variables: %s"  % clobbered +
```

In [15]:
```
import matplotlib.pyplot as plt
#plt.scatter(df["date"],df["sentiments"],color='blue',figsize=(10,12))
plt.figure(figsize=(10, 12))
plt.scatter(df["date"],df["sentiments"],color='blue')
df = df.sort_values("date")
plt.plot(df["date"],df["sentiments"],color='red',linestyle='--',linewidth =1)
plt.show()
```

第四讲 情感分析

2. 完成sentiment_analysis… …alysis_4，4份代码。做截图，并简要做… …析。

第四讲 情感分析

2. 完成sentiment_analysis_1，sentiment_analysis_2，sentiment_analysis_3，sentiment_analysis_4，4份代码。做截图，并简要做结分析。