1、毕设主要内容：

对文学文本（路遥、陈忠实、贾平凹作品）进行计量分析与可视化

2、工具：

Python及相关库（numpy、matplotlib、pandas、scikit-learn等），语言工具（quita、MATTR.exe、mawatatarad2.exe、wordsmith等）

3、任务：

1. Python对文本进行分词、词性标注等自然语言处理，使用中文库jieba、ltp和英文库nltk。

2. 语言工具对分词或者词性标注后的文本进行处理，使用quita、mattr、mawatatarad得到各种指标的量和数据

3. Python及相关库对数据进行分析和可视化，包括语言工具无法进行计算的指标用Python进行计算，已经得到的数据进行数据拟合，利用聚类和分类模型挖掘文本的量化特征（作品与作品、作家与作家之间的相同或者不同的量上的特征）

4、题目：

基于Python的路遥文学作品量化分析与可视化 李震南

基于Python的陈忠实文学作品量化分析与可视化 王煜

基于Python的贾平凹文学作品量化分析与可视化 奥圣杰

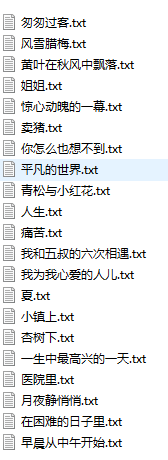
目的：使用python语言和相关库,对自然语言文学作品进行分析和可视化操作,从而得到作品与作品、作家与作家之间的相同或不同的量化特征。

## 具体工作流程

所有工作要形成过程文档，详细记录每一步中具体做法，以备后续使用。

第一项工作：整理作家作品的txt文档，保存格式必须为utf-8，目前已经有的包括：

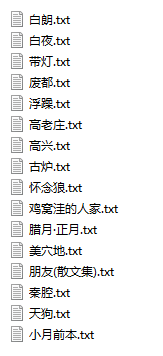
路遥



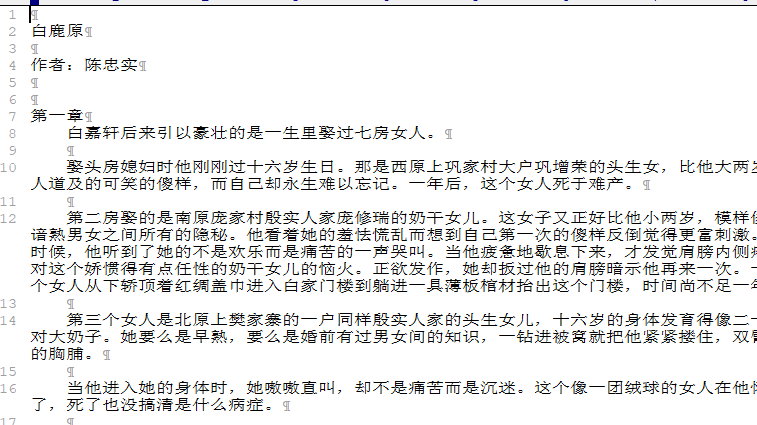
陈忠实



贾平凹



在此基础上，查缺补漏，找到每个作家缺少的一些作品补充进来；另外，把文本格式处理好，主要使用ultraedit或者editplus来做，具体替换使用正则表达式等请自行查找解决办法，并且记录到过程文档之中。例如《白鹿原》



这个里面要去掉篇名，去掉作者，去掉第几章这些内容，多余的空行也要去掉。最后只保留作品的具体内容，全部形成一段，即没有任何回车换行，所有内容从头到尾就只剩一段。最后，保存为utf-8格式的txt文件。

另外一项工作就是把作家作品整理名录，按照“陈忠实统计分析数据.xlsx”如下格式，信息尽可能的准确：



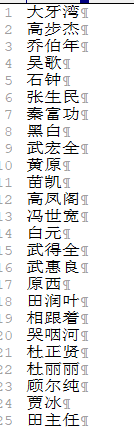
第二项工作：对于整理好的txt文档进行分词和词性标注，主要使用Python+jieba或者Python+哈工大pyltp，分词做一份，分词+词性标注做一份。具体性能上看jieba快一些，但是准确性pyltp高一些，jieba我已经实现分词，但词性标注还没做好，因此最好两个全部都调好以便做对比（jieba分词、jieba词性标注、pyltp分词、pyltp词性标注）。

jieba：<https://github.com/fxsjy/jieba>

pyltp：<https://pyltp.readthedocs.io/zh_CN/latest/>

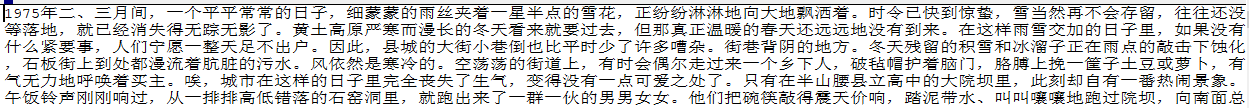
https://github.com/HIT-SCIR/pyltp

分词时需要对每个作品尤其是小说里面的人名地名等专有名词做字典（每部小说建一个字典），以便能够分词的更加准确。例如平凡的世界分词时我建的字典部分内容：

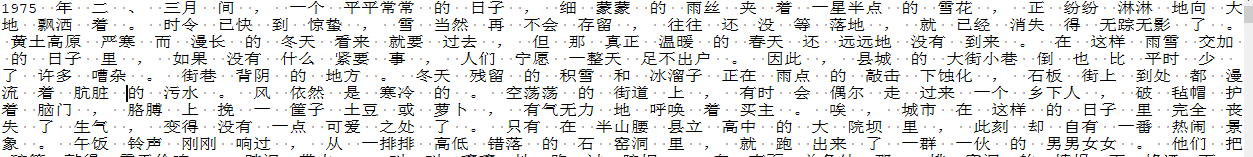


分词后效果如下：

分词前



分词后

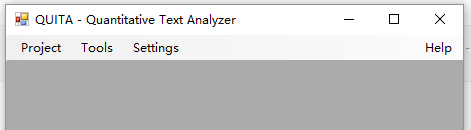


最后所有txt分门别类放到4个目录（jieba分词，jieba标注，pyltp分词，pyltp标注）

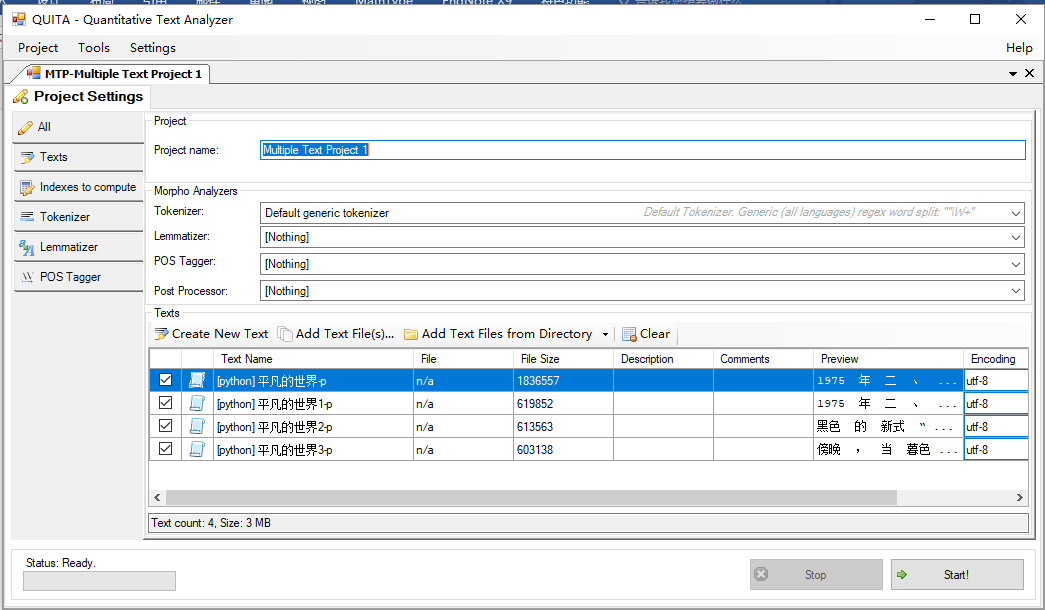
第三项工作，分词后的txt进行相关数据的计算并保存还有数据拟合和可视化等工作。主要涉及到几个工具quita，mattr，mawatatarad等。

第一个工具是quita，里面的有些指标需要阅读文档来理解计算方法，参见“2014\_QUITA MANUAL.pdf”第6项，

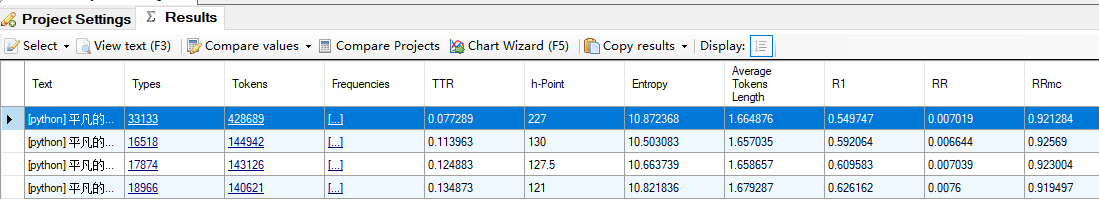
project—new project



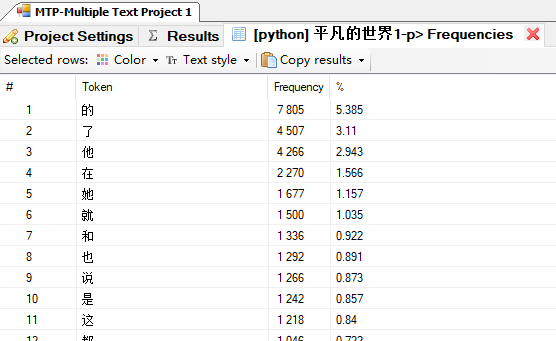
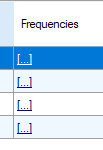
打开分词处理好的txt—add text files---start开始计算，注意后面encoding识别出来必须是utf-8



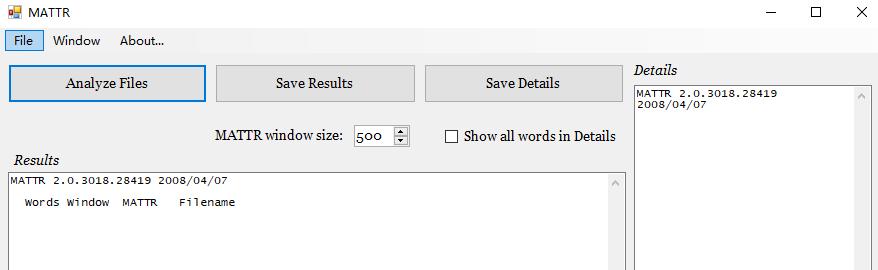
计算后copy results—选择第一个copy grid。。。，第二个会有乱码，保存到相应的Excel文档中以备后续分析处理。



另外还需保存其他几个内容，依次双击每一个[…]打开词频表进行保存，同样是copy results—第一项copy grid。。。保存到Excel文档中，如“平凡的世界.xlsx”中所示

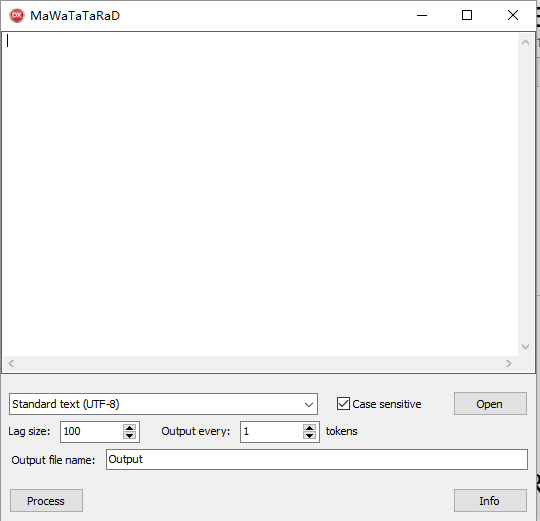


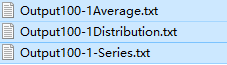
第二个工具是MATTR，使用并不复杂，主要计算TTR和分布，MATTR window size分别选择100和500进行计算两次（得到两批结果数据），同时勾选show all words in details，这时该工具会很慢，耐心等待，没有死机，最后要save results和save details两次。后续保存的数据都要进行分析、数据拟合和可视化工作。尤其是details文件里的数据，需要用到Python处理。





第三个工具是mawatatarad2，也是计算TTR和分布的，使用也比较简单，lag size跟上一个window size一样设置100和500计算两次。三个输出数据后两个也是得到后后续需要Python处理





第四项工作：使用Python来对得到的所有数据进行分析、拟合和可视化显示了，细节稍后再做。

一些有用的网站：

wordcloud：http://amueller.github.io/word\_cloud/index.html

imageio: https://imageio.readthedocs.io/en/latest/index.html

http://imageio.github.io/

scipy: https://scipy.org/

numpy: http://numpy.org/

https://www.numpy.org.cn/

https://github.com/wizardforcel/matplotlib-user-guide-zh/blob/master/3.1.md

matplotlib http://matplotlib.org/

pandas: http://pandas.pydata.org/

nltk：https://textminingonline.com/dive-into-nltk-part-i-getting-started-with-nltk

https://www.clarin.ac.uk/wmatrix

https://www.ram-verlag.eu/journals-e-journals/glottometrics/