课程作业 1-A

尧祥临 202518023406038 前沿交叉科学学院

1 任务说明

本报告选择的是课程作业 1-A，其主要任务包括：

- 收集中英文语料
- 计算在收集样本上英语字母和单词或汉字的概率和熵
- 利用收集的英文文本验证齐夫定律（Zipf's law）
- 在不同样本量下探究结果差异

2 语料获取与清洗

2.1 现代与历史语料获取

2.1.1 维基百科随机词条爬取

语料获取上，本报告为了统一在同一语言环境获取中英文语料，从而选择了维基百科（Wikipedia）作为爬取的目标，主要思路是利用了维基百科提供的随机词条功能，每次进入随机的百科词条页面之后，获取相应的词条正文内容。分别对中文维基和英文维基使用随机词条功能即可爬取到大量的语料，这样的好处是能确保爬取到的语料可以认为是正式规范的，一定程度上能反应现代英语和现代汉语的特征；但是同样也有一定的缺点，例如受百科这样的体例限制，内容主要是对人物事迹、重要事件经过以及概念的叙述，在主题上有所匮乏。

完整的爬取代码如附录 7.1 所示。

对于中文词条，爬虫脚本中访问的 URL 需要设置为 zh.wikipedia 确保随机出来的词条是中文词条，同时在最后添加 variant=zh-cn 确保词条文字是大陆简体；对于英文词条，只需要设置为 en.wikipedia 即可。经过对多个词条网页源代码的分析，发现维基百科词条的正文内容总出现 #mw-content-text > div.mw-content-ltr.mw-parser-output 的路径下，所以这里我选择直接锁定这一路径，在这一路径下再获取所有位于 <p> 内的段落内容。这样每次爬取到的语料是来自正文而不是词条网页内其余文字内容，从而保证语料是成段落成规模的句子而不是一些杂乱的短语或者重复性的网页引导文字。

```
if lang == 'zh':
        self.base_url = "https://zh.wikipedia.org/wiki/Special:Random?variant=
            zh-cn"    #加上zh-cn确保爬下来的是大陆简体中文
elif lang == 'en':
        self.base_url = "https://en.wikipedia.org/wiki/Special:Random"
```

```
content_div = soup.select_one('#mw-content-text > div.mw-content-ltr.mw-parser-
    output') # 直接定位到该路径
paragraphs = content_div.find_all('p')
```

对于每一个词条，脚本会分别记录下三个字段，title、content 和 url。其中 title 用来记录词条的名称，例如“巴鲣 - 维基百科，自由的百科全书”；url 用来记录下该词条的链接，并通过这个来保证记录下来的词条内容不会因为两次随机到了同一个词条而产生重复；最关键的 content 用来记录正文文本内容，在爬取过程中会在删除文本段落前后可能的空格之后将所有段落直接拼接起来，同时分别对中英文进行不同的处理：中文直接去掉文本中存在的任何空白字符，而英文则将长空格保留为一个从而至少为不同单词之间留下空格做分隔。

```
content = ''.join([p.get_text().strip() for p in paragraphs])
content = ' '.join(content.split()) if self.lang == 'en' else re.sub(r'\s+', '', content)  # 分情况处理，英文需要用空格来分隔单词，中文直接可以把空白字符给去掉
```

在保存的时候，考虑到这里爬取的语料未来可能另有其他用处，所以并没有直接存储为纯文本的 txt 格式，而是将 title、content 和 url 作为三个字段，把每个词条写成一个 json 字典，最终全部爬取的文本保存为一个大的 jsonl 文件。对于中文和英文的语料，均爬取 20000 条不重复的词条，以保证文本足够充足用来进行统计分析。

2.1.2 历史语料

为了进一步去分析中文和英文在统计上的各种特征，本报告额外获取了一些历史语料，来探究现代汉语与现代英语同古代汉语（文言文）和古英语之间在统计上存在的差异。对于中文的历史语料本报告选择了《史记》，从某种意义上说史记的纪传体同百科的词条有一定的相似之处；对于英文的历史语料，本报告选择了莎士比亚最长的剧本《哈姆雷特》，因莎翁对于英语词汇的极大丰富与拓展。

2.2 清洗思路与方法

针对爬取到的中英文各 20000 条词条，需要对文本内容进行基本的清洗。

3 中文语料分析结果

4 英文语料分析结果

4.1 字母

4.2 单词

4.3 Zipf 定律

5 语料对比

6 总结

7 附录

7.1 爬虫代码

```
import numpy
```

7.2 语料清洗代码

```
import re
```