单位代码

学 号ZY2303803

分类号

****

**深度学习与自然语言处理（NLP）第一次课后作业**

|  |  |
| --- | --- |
| 院（系）名称 | 自动化科学与电气工程学院 |
| 专业名称 | 自动化 |
| 学生姓名 | 董晨辉 |

2024年04月

**深度学习与自然语言处理（NLP）第一次课后作业**

董晨辉

1127666815@qq.com

**Abstract**

通过中文语料库金庸小说大全的.txt文件验证了Zipf‘s Law(奇夫定律)，并在此基础上分别以词和字为单位计算了十余部小说全部文本的平均信息熵。

**Introduction**

1. 奇夫定律

齐夫定律（Zipf's law）由哈佛大学的语言学家乔治·金斯利·齐夫于1949年提出，是一种描述自然语言中单词出现频率的统计规律。

根据齐夫定律，在给定的语料库中，出现频率最高的单词出现的频率大约是出现频率第二位的单词的两倍，而出现频率第二位的单词则是出现频率第四位的单词的两倍。这种关系可以扩展到整个词汇表，其中每个单词的出现频率与其在频率表中的排名成反比。

1. 熵与信息熵

熵，泛指某些物质系统状态的一种量度，某些物质系统状态可能出现的程度。亦被社会科学用以借喻人类社会某些状态的程度。熵的概念是由德国物理学家克劳修斯于1865年所提出。最初是用来描述“能量退化”的物质状态参数之一，在热力学中有广泛的应用。它在控制论、概率论、数论、天体物理、生命科学等领域都有重要应用，在不同的学科中也有引申出的更为具体的定义，按照数理思维从本质上说，这些具体的引申定义都是相互统一的，熵在这些领域都是十分重要的参量。

信息熵这一概念由克劳德·香农于1948年提出。香农是美国著名的数学家、信息论创始人，他提出的“信息熵”的概念，为信息论和数字通信奠定了基础。信息熵是用于衡量不确定性的指标，也就是离散随机事件出现的概率，简单地说“情况越混乱，信息熵就越大，反之则越小”。 数学家香农给出了信息熵的计算公式，如下所示：

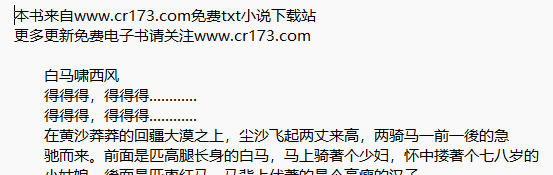
信息熵的三个性质为：单调性，即发生概率越高的事件，其携带的信息量越低。非负性，信息熵可以看作为一种广度量，非负性是一种合理的必然。累加性，即多随机事件同时发生存在的总不确定性的量度是可以表示为各事件不确定性的量度的和，这也是广度量的一种体现。香农从数学上严格证明了满足上述三个条件的随机变量不确定性度量函数具有唯一形式。

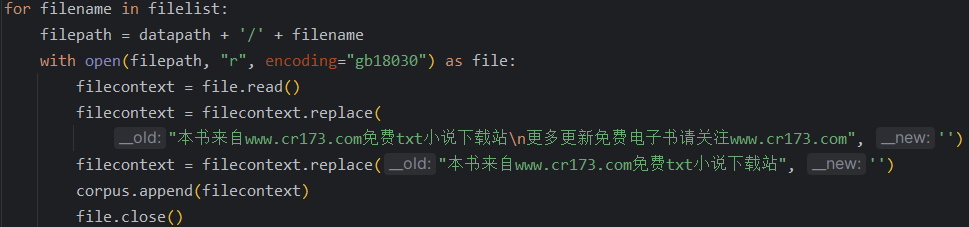
**Methodology**

在实验过程中，首先需要对.txt文本进行预处理，并对文本进行分词。在分词的基础上进行词频统计以及信息熵的计算。

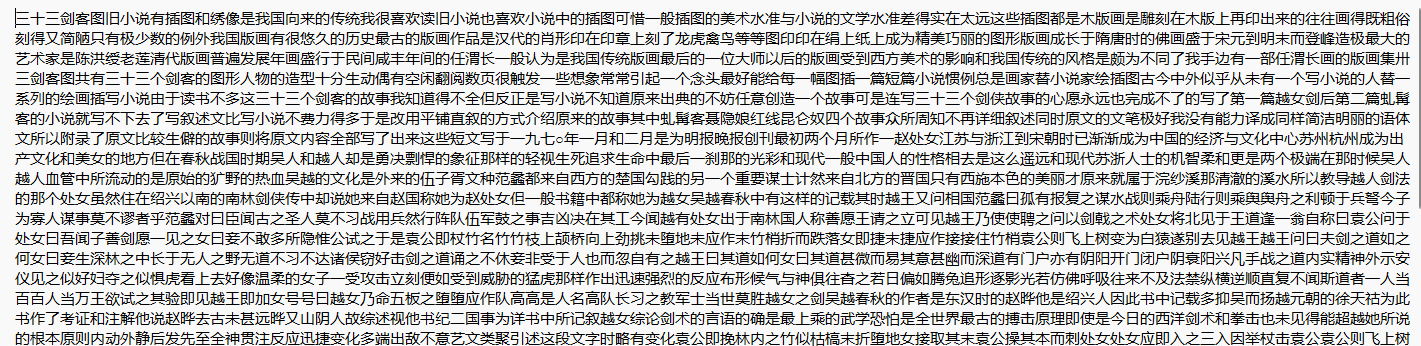
**M1: 文本预处理**

由于下载的文件里存在各种标点符号以及网页信息，去除了文件里面无关信息：





文本处理之后：



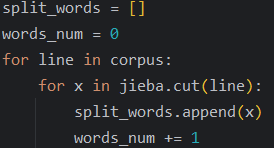
**M2: 中文分词**

中文分词，通俗来说，就是将一句(段)话按一定的规则(算法)拆分成词语、成语、单个文字。中文分词是很多应用技术的前置技术，如搜索引擎、机器翻译、词性标注、相似度分析等，都是先对文本信息分词处理，再用分词结果来搜索、翻译、对比等。在Python中，最好用的中文分词库是jieba。用“结巴”给一个中文分词库命名，非常生动形象，同时还带有一种程序员式的幽默感。

jieba分词支持四种分词模式，精确模式是最常用的分词模式，分词结果不存在冗余数据：

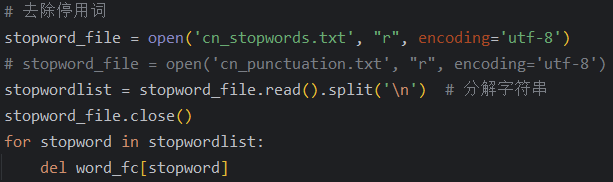
**jieba.cut（text, cut\_all=False）**

**如图所示使用jieba.cut()函数逐行进行分词并统计分词个数：**



**M3: 去除停用词**

在进行分词后应根据停用词表去除无关的停用词：



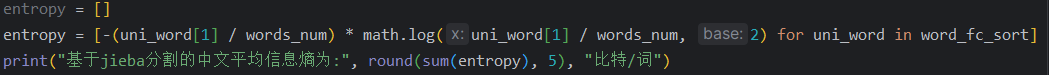
**M4: 词频统计**

在Python中，NLTK（Natural Language Toolkit）库是一个功能强大、广泛使用的自然语言处理库。词频统计使用NLTK库中的FreqDisk函数，能够统计数组zho给单词出现的频率。

如图所示，利用FreqDisk函数将split\_works[ ]列表中的分词进行词频统计，并按照词频数从大到小排列：



并按照公式计算信息熵：



**Experimental Studies**

**采用cn\_punctuation去除停词的实验结果如下所示：**

运行脚本后词频统计word\_fc\_sort[ ]如下图所示：

|  |  |
| --- | --- |
| Jieba分词 | 按字分词 |
|  |  |

表1分词结果

|  |  |  |
| --- | --- | --- |
|  | 按照jieba分词 | 按字分词 |
| 语料库字数 | 7273194 | 7269812 |
| 分词字数 | 4289223 | 7269812 |
| 平均字长 | 1.69569 | 1 |

表2信息熵

|  |  |  |
| --- | --- | --- |
|  | 按照jieba分词 | 按字分词 |
| 中文平均信息熵 | 12.15446比特 | 9.53684比特 |

绘制log-log图像：

|  |  |
| --- | --- |
|  |  |
|  |  |

图1只去除标点后的实验结果

**采用cn\_stopwords去除停词的实验结果如下所示：**

运行脚本后词频统计word\_fc\_sort[ ]如下图所示：

|  |  |
| --- | --- |
| Jieba分词 | 按字分词 |
|  |  |

表3去除停用词分词结果

|  |  |  |
| --- | --- | --- |
|  | 按照jieba分词 | 按字分词 |
| 语料库字数 | 5660926 | 4495115 |
| 分词字数 | 2872492 | 4495115 |
| 平均字长 | 1.85417 | 1 |

表4信息熵

|  |  |  |
| --- | --- | --- |
|  | 按照jieba分词 | 按字分词 |
| 中文平均信息熵 | 13.58133比特 | 9.95158比特 |

绘制log-log图像：

|  |  |
| --- | --- |
|  |  |
|  |  |

图2去除所有停用词后的实验结果

**Conclusions**

本文通过分词和词频统计，初步验证了奇夫定律，即在给定的语料库中，出现频率最高的单词出现的频率大约是出现频率第二位的单词的两倍，而出现频率第二位的单词则是出现频率第四位的单词的两倍，反映到log-log图中则是一条斜率约为-1的直线。这种关系可以扩展到整个词汇表，其中每个单词的出现频率与其在频率表中的排名成反比。

而通过对比去除停用词或标点可以看出对于一元模型，说明去除信息量较少的停词（例如“一些”、“大概”等）对于信息熵的影响较小。同时对比去除停用词前后可观察到去除停用词更加符合奇夫定律的规则。

通过对比使用jieba库进行分词和直接按字分词，表明了按照语意和词义进行分词所得到的平均信息熵要高于只按字分词的平均信息熵，并且jieba分词能够更好的验证奇夫定律。

**References**

[1] Peter F. Brown, Vincent J. Della Pietra, Robert L. Mercer, Stephen A. Della Pietra, and Jennifer C. Lai. 1992. An estimate of an upper bound for the entropy of English. Comput. Linguist. 18, 1 (March 1992), 31–40.

[2] https://blog.csdn.net/weixin\_42663984/article/details/115718241

[3] https://blog.csdn.net/GWH\_98/article/details/117001985