中文文本信息熵计算

潘熠暄
ZY2203112

一．摘要

本实验的理论部分基于 Peter Brown 的论文 Entropy of English，以金庸先生的 16 本小说（部分节选）为语料库，分别以词和以字为单位计算一元、二元和三元信息熵。

二．介绍

信息熵是一种用于衡量信息随机性和不确定性的概念。它最初是由克劳德·香农在 1948 年提出，之后被广泛应用于信息论、通信和统计物理等领域。

信息熵为一个非负数，用于表示信息源中的信息量的大小，其计算公式为：

$$H(X) = -\sum_{x \in X} p(x) \cdot \log_2 p(x)$$

其中，$p(x)$表示随机变量X取x的概率，在实际工作中，往往以统计数据的频率来代替。

本次实验选择计算三种分词模型下的信息熵，它们之间的区别在于：一元模型信息熵的计算公式就是前述公式，该模型的主要观点在于假定文本中的各个词据相互独立。另一种是 N 元模型，该模型的主要观点在于将自然语言句子视作 N-1 阶的马尔可夫模型，即规定句子中某词出现的概率只同它前面的 N-1 个词有关，本实验中计算二元模型和三元模型：

$$H(X|Y) = -\sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 p(x|y)$$

$$H(X|Y,Z) = -\sum_{x \in X} \sum_{y \in Y} \sum_{z \in Z} p(x,y,z) \log_2 p(x|y,z)$$

三．实验过程及结果

1.程序设计

选择使用 python 为编程语言，这主要是出于方便调用用于进行中文分词工作的 jieba 库；程序主要分为了数据预处理部分、词频统计部分和信息熵计算部分。

数据预处理部分的工作在于将原本的文本信息中的标点符号和换行符制表符和空白符等去除。

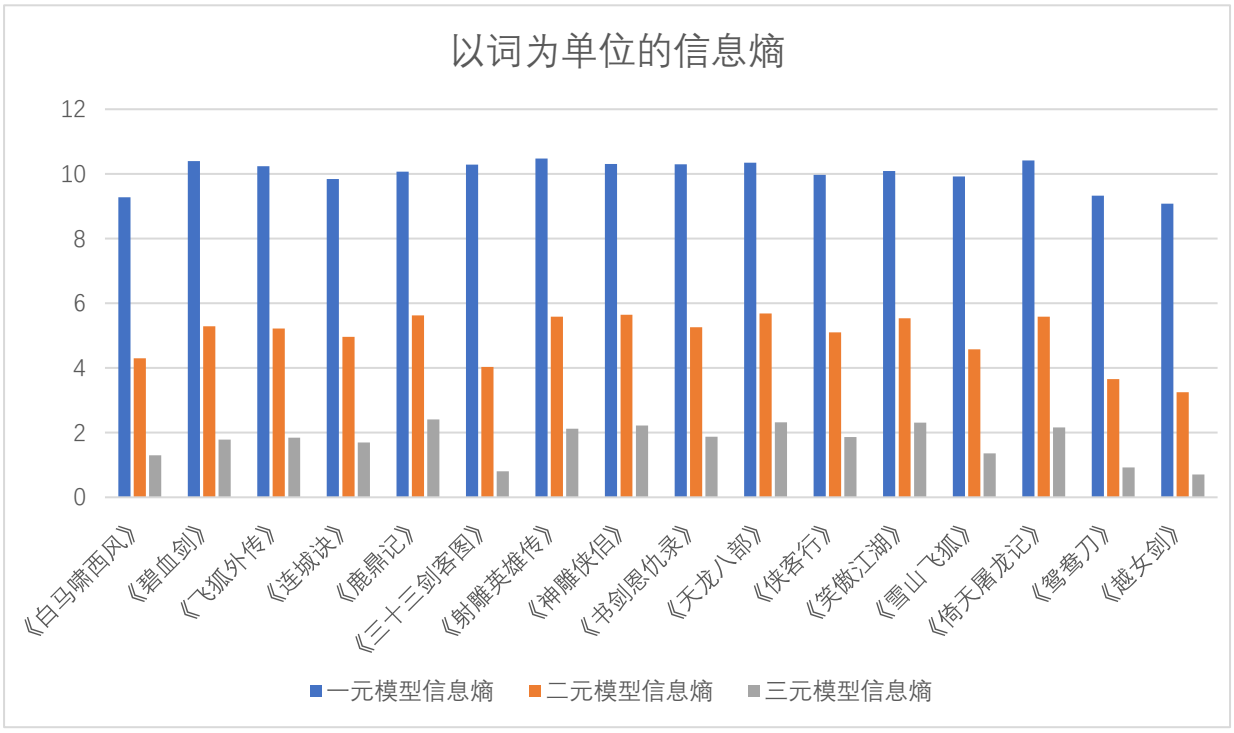
词频统计部分按照三个模型的需求分别以词和字为单位统计经过预处理后的文本中的词频。

信息熵计算部分根据信息熵的计算原理编写。

以下图表为使用编写程序计算语料库信息熵的具体结果。

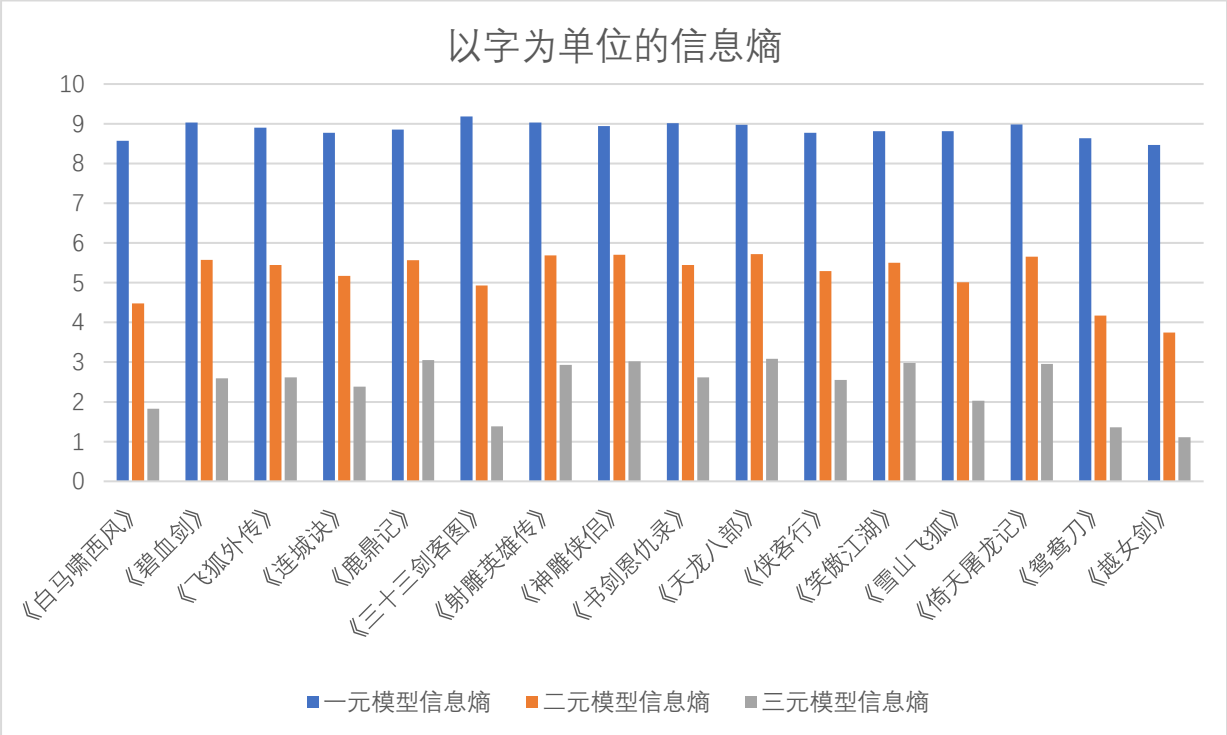
以词为单位计算信息熵:

	一元模型信息熵	二元模型信息熵	三元模型信息熵
《白马啸西风》	9.281337520591375	4.294850390240943	1.3032982153504078
《碧血剑》	10.394957889012563	5.289720783140237	1.7849562035389717
《飞狐外传》	10.234259392762072	5.2176010785394755	1.8431377820695345
《连城诀》	9.845921500406565	4.963615413405279	1.6967648145837089
《鹿鼎记》	10.07280437204348	5.625094897307343	2.404611453366006
《三十三剑客图》	10.289481477296896	4.0278715891787416	0.801259593885489
《射雕英雄传》	10.47375877852595	5.584517153084413	2.12270632658713
《神雕侠侣》	10.306379514495749	5.647383353698795	2.2154988454030664
《书剑恩仇录》	10.300669037312039	5.255251819420998	1.87510622160755
《天龙八部》	10.350375161674686	5.688636758281721	2.3193319272045323
《侠客行》	9.974383075081672	5.097103761135414	1.8647336092082136
《笑傲江湖》	10.087733851963135	5.534131617123746	2.312123423464072
《雪山飞狐》	9.91848027624308	4.5770689903856026	1.3598133451550096
《倚天屠龙记》	10.411855399752506	5.582692352316668	2.1631558730647438
《鸳鸯刀》	9.328360226538964	3.655060109659974	0.9205879341992785
《越女剑》	9.081187927807337	3.245313561190839	0.7005594028551748



以字为单位计算信息熵:

	一元模型信息熵	二元模型信息熵	三元模型信息熵
《白马啸西风》	8.56775297612155	4.480176265649609	1.8303828562740232
《碧血剑》	9.028339768339388	5.574221338255298	2.592989977221675
《飞狐外传》	8.905071132828937	5.446927140483657	2.6147577569849103
《连城诀》	8.769892679866324	5.170066688468085	2.3856548795521504
《鹿鼎记》	8.849787755705979	5.56217096128001	3.049557270725519
《三十三剑客图》	9.184447665384752	4.92528004083811	1.3869363290341359
《射雕英雄传》	9.029501541683516	5.6838725063136275	2.930822467699046
《神雕侠侣》	8.943399862261884	5.702183787871529	3.0161186468173367
《书剑恩仇录》	9.011307204963177	5.447826991700443	2.620320585219797
《天龙八部》	8.972083477650314	5.720984665151109	3.082553975569044
《侠客行》	8.774330192822088	5.2952808644517555	2.554547146120135
《笑傲江湖》	8.81224597362683	5.499704019266198	2.9779592563513066
《雪山飞狐》	8.80939305473917	5.006407577817559	2.030789418591874
《倚天屠龙记》	8.98530960168204	5.651883170097841	2.9562450635679167
《鸳鸯刀》	8.634645494613835	4.170874570835672	1.358983209310947
《越女剑》	8.468062421872615	3.7416608463881222	1.1131192786409791



三．结论

经分析计算，以词为单位的一元模型信息熵大小的平均值为 10.022，二元模型信息熵大小的平均值为 4.95537，三元模型信息熵大小的平均值为 1.730478；以字为单位的一元模型信息熵大小的平均值为 8.85909，二元模型信息熵大小的平均值为 5.19247，三元模型信息熵大小的平均值为 2.406359。

不难发现，不论是以词单位还是以字为单位进行计算，信息熵的值都随着模型元数增加而减小，这是因为 N 元模型认为词和前 N-1 个词相关，在信息熵计算时需要使用条件概率和联合概率，随着 N 的增加，需要参与计算的概率项数变多，信息熵结果值变小。

同时，以字为单位的信息熵仅仅在一元模型下比以词为单位的信息熵要小，这可能是因为在一元情形下，词的数目要多于以字为单位的词的数目；而多元时，由于不同词之间的固定搭配被模型捕捉，导致此时以词为单位的信息熵要小于以字为单位的信息熵。