NLP 中文信息熵实验报告

贾云飞 ZY2303207

摘要

本文介绍了 Zipf's Law 以及中文信息熵的概念和计算方法，用多部中文武侠小说语料库，验证了 Zipf's Law，进行了基于字和词的中文信息熵计算，并分析了使用不同模型对中文语料信息熵值的差异。

1. 简介

Zipf's Law 是由美国语言学家 George Kingsley Zipf 于 1935 年提出，是描述自然语言中词汇分布非均匀性的一种经验定律。Zipf's Law 主要表述了这样一个现象：在大多数自然语言的语料库中，某一特定词的出现频率与其在词频排名上的倒数成反比关系。该存在揭示了语言使用的经济原则和人类认知的局限性，它在不同语言、不同文本类型中展现出惊人的普适性，是语言统计学的重要基石。

信息熵是信息论中一个重要的概念，用于衡量信息的不确定性和复杂度。中文语言的复杂性和多样性使得计算中文信息熵变得复杂。因此，本文介绍了中文信息熵的概念和计算方法，并分析了使用不同模型对中文语料信息熵值的差异。本文使用多部中文武侠小说语料库，验证了 Zipf's Law。采用信息熵的公式，进行了中文基于字的信息熵计算以及基于词的一元、二元、三元模型的信息熵计算，并通过停用词消除、jieba 库分字或词等操作估计信息熵的计算方法。

2. 实验方法

2.1 验证 Zipf's Law

齐夫定律（Zipf's Law）是一个实验定律，而非理论定律。齐夫定律很容易用点阵图观察，坐标为 log（row）和 log（frequency）。如果所有的点接近一条直线，那么它就遵循齐夫定律。给出一组齐夫分布的频率，按照从最常见到非常见排列，第二常见的频率是最常见频率的出现次数的 1/2，第三常见的频率是最常见的频率的 1/3，第 n 常见的频率是最常见频率出现次数的 1/n。为验证 Zipf's Law，选取十六本中文武侠小说，分别去除其特殊字符，并利用 jieba 库进行分词操作，最后计算词频和排名，绘制对数图观察是否符合实验定律。

2.2 信息熵

信息熵是信息论中用来度量信息的不确定性和复杂度的一种方法。对于一个离散随机变量 X，其信息熵可以通过以下公式计算：

$$\mathrm{H(X)} = \sum_{x \in X} P(x) \log(\frac{1}{P(x)}) = -\sum_{x \in X} P(x) \log(P(x))$$

其中，P(x) 是变量 X 取值为 x 的概率。式中对数一般取 2 为底，单位为比特。

2.3 语言处理

对于中文文本，其信息熵可以通过以下步骤计算。

S 表示某一个有意义的句子，由一连串特定顺序排列的字或字或词 x1, x2, ...xn 组成，n 为句子的长度。利用条件概率的公式，S 这个序列出现的概率等于每一个字或词出现的条件概率相乘，于是 P(x1, x2, ...xn)可展开为：

$$P(x1, x2, \dots xn) = P(x1)P(x2|x1)\dots P(xn|x1, x2, \dots xn - 1)$$

其中 P(x1)表示第一个字或词 x1 出现的概率；P(x2|x1)是在已知第一个字或词的前提下，第二个字或词出现的概率；以此类推可得到。

$$P(S) = P(x1)P(x2) \dots P(xn)$$

其对应的统计语言模型就是一元模型。

若我们假设任意一个字或词 xi 出现的概率只同它前面的字或词 xi-1 有关，S 的概率变为：

$$P(S) = P(x1)P(x2|x1)\dots P(xn|xn - 1)$$

其对应的统计语言模型就是二元模型。也可以假设一个字或词由前面 N-1 个字或词决定，即 N 元模型。

当 N=3 时，每个字或词出现的概率与其前两个字或词相关，为三元模型，对应 S 的概率变为：

$$P(S) = P(x1)P(x2|x1)\dots P(xn|xn - 2, xn - 1)$$

根据以上不同的模型，可以根据其联合分布的随机变量，将信息熵改编成可用于二元、三元模型计算的联合信息熵：

$$H(X|Y) = -\sum_{y \in Y} P(y) \log\big(P(x|y)\big)$$

$$= -\sum_{y \in Y} P(y) \sum_{x \in X} P(x) \log\big(P(x|y)\big)$$

$$= -\sum_{y \in Y} \sum_{x \in X} P(x, y) \log\big(P(x|y)\big)$$

3. 实验过程

3.1 数据预处理

数据预处理包括验证 Zipf's Law 的数据预处理和计算信息熵时的数据预处理。

验证 Zipf's Law 时，为真实反映中文语料库信息，对其中包含的大量乱码与无用或重复的中英文符号，进行预处理，处理之后绘出对数图。

在计算信息熵时，基于分词的中文信息熵计算需要停词表。停词表包括文章中的标点以及常见语气助词等无实意的字或词。

本文使用了 python 的 jieba 库来进行中文词汇的分词，该库的主要任务是将读取的字符串，按照数据库中的中文词汇，将中文字符串分成多个词组，便于后

面进行词组信息熵的计算。数据集为十六本武侠小说。

首先删除无意义的字符比如空格、回车、制表符、段落符；其次根据已经获取的停词表，将相应的停词，如标点、英文字母、阿拉伯数字、无实意语气词等进行删除；最后可以选择直接使用汉字，或者使用 jieba 库将长字符串进行分词，得到多个词组。

3.2 计算信息熵

如果统计量足够，根据大数定理，词或二元词组或三元词组出现的概率大致等于其出现的频率。

一元模型的信息熵计算公式为

$$\mathrm{H(X)} = -\sum_{x \in X} P(x) \log(P(x))$$

其中 P(x)可近似等于每个词在语料库中出现的频率。

二元模型的信息熵计算公式为

$$\mathrm{H(X|Y)} = -\sum_{y \in Y} \sum_{x \in X} P(x, y) \log\big(P(x|y)\big)$$

其中联合概率 P(x,y)可近似等于每个二元词组在语料库中出现的频率，条件概率 P(x|y)可近似等于每个二元词组在语料库中出现的频数与以该二元词组的第一个词为词首的二元词组的频数的比值。

三元模型的信息熵计算公式为

$$\mathrm{H(X|Y,Z)} = -\sum_{y \in Y, x \in X, z \in X} P(x, y, z) \log\big(P(x|y, z)\big)$$

其中联合概率 P(x,y,z)可近似等于每个三元词组在语料库中出现的频率，条件概率 P(x|y,z)可近似等于每个三元词组在语料库中出现的频数与以该三元词组的前两个词为词首的三元词组的频数的比值。

4. 实验结果

4.1 验证 Zipf's Law

在本实验中，我们采用了基于 Python 的分字或词工具 jieba 进行验证，将十六本武侠小说分别计算词频和排名，绘制对数图如下。

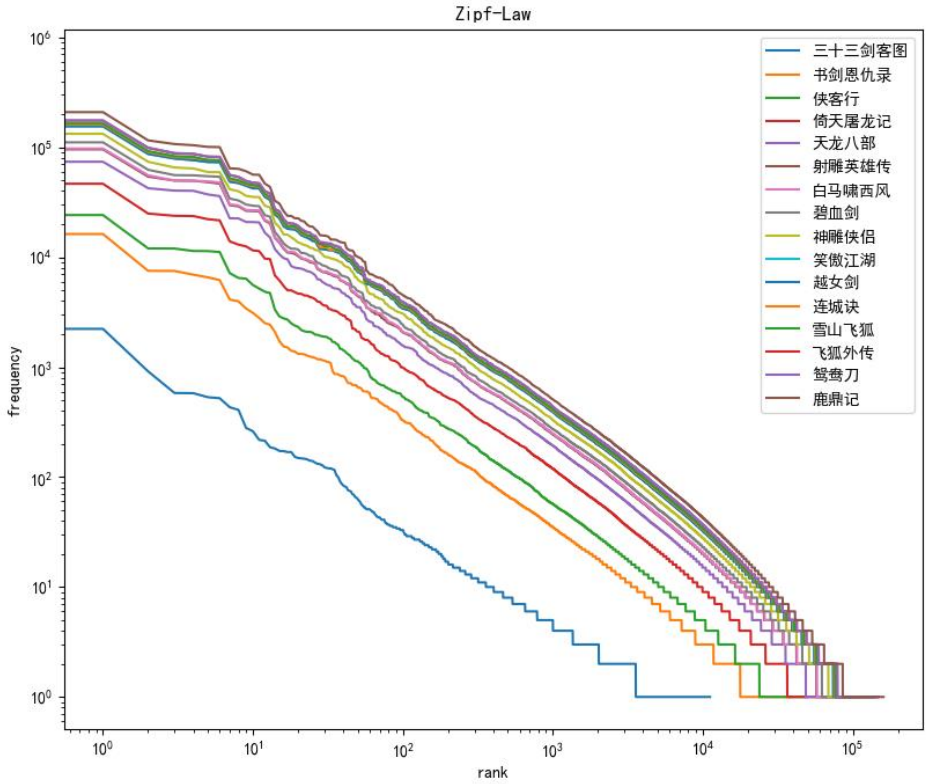

图 1 十六本武侠小说词频和排名对数图

4.2 计算信息熵

首先不使用分词，将语料库中的每个文字都当作相互独立的分布，搭建一元模型，计算得到基于字的一元模型的中文平均信息熵。

之后采用了基于 Python 的分词工具 jieba 库进行分词。并构建一元模型、二元模型和三元模型分别使用三种模型对十六本武侠小说单独每部都进行分词的中文平均信息熵的计算。得到的结果如下表所示。

表 1 十六本武侠小说每部分字/词中文平均信息熵表

书名	字单位一元模型信息熵/bit	词单位一元模型信息熵/bit	词单位二元模型信息熵/bit	词单位三元模型信息熵/bit
白马啸西风	9.224685	11.205605	2.714367	0.263181
碧血剑	9.763351	12.932855	3.729197	0.379314
飞狐外传	9.627404	12.715544	3.776883	0.384518
连城诀	9.521108	12.244399	3.349697	0.308295
鹿鼎记	9.662416	12.877021	4.705981	0.662226
三十三剑客图	10.015207	12.444231	1.650065	0.068375
射雕英雄传	9.754463	13.13673	4.336633	0.46126
神雕侠侣	9.660061	12.891323	4.36283	0.537993
书剑恩仇录	9.760196	12.783513	3.918619	0.427123
天龙八部	9.789651	13.182218	4.533592	0.56323
侠客行	9.438565	12.347765	3.741167	0.453531
笑傲江湖	9.515424	12.617519	4.571825	0.724734

雪山飞狐	9.503196	12.12658	2.783892	0.231849
倚天屠龙记	9.707602	13.009854	4.419257	0.56293
鸳鸯刀	9.210499	10.998277	2.113501	0.181148
越女剑	8.784704	10.270746	1.735488	0.225989

结论

本实验首先基于十六本中文武侠小说验证了 Zipf's Law 同时研究了如何测量中文的信息熵。首先绘制基于中文语料库的 Zipf's Law 验证的对数图，之后应用了分字、分词以及一元、二元、三元的模型计算语料库的中文信息熵。研究过程中能较为充分的考虑非实意中文的情况，应用停词表进行筛选，并对比不同语料库的结果，进行总结和归纳，最后得到基于字的、基于词的一元、二元、三元模型中文平均信息熵的取值。

由对数图可以看出十六本中文小说的词排名和词频均基本上成反比关系，用实验验证了 Zipf's Law。同时得到了基于不同模型的字/词的中文平均信息熵，可以看出随着模型元数 N 的增大，文中所含的信息熵在减小，中文信息熵与分词数有较为明显的关系，同时可以看出样本越大其检测的结果越接近真实。

不过本文研究过程中仍发现了一些问题，检测的结果可能也与文本的类型有关，停词表中有“过”和“去”这种类型的停词，可能单一的汉字确实没有实意，不过有些组合的词汇比如“过来”、“过去”等是有实际意义的，使用停词删除可能会造成信息的丢失，最终无法正确统计信息熵。