学　　期　2022 春季学期

北京航空航天大學
BEIHANG UNIVERSITY

NLP

第一次大作业

院（系）名称	自动化科学与电气工程学院
专业名称	控制科学与工程
学生姓名	陈真
学　　号	SY2103801

2022 年 4 月

- N-gram 定义

  N-gram 表示的是一个给定文本/音频样本中有 n 项（音素，音节，字，词）的一个连续序列。

- N-gram 数学表达

  N-gram 模型表示的是当前这个 word w_i 依赖于前面 N-1 个 word，所以可以表达为：

$$P\left(w_i|\ w_{i-n+1}^{i-1}\right)=P\left(w_i|\ w_{i-n+1}\cdots w_{i-1}\right)$$

$$\{MLE\}\approx\frac{c\left(w_{i-n+1}\cdots w_{i-1}w_i\right)}{c\left(w_{i-n+1}\cdots w_{i-1}\right)}$$

  其中为了书写方便，做出如下规定：

$$w_{i-n+1}^{i}=w_{i-n+1}\cdots w_{i-1}w_i$$

  最大似然估计 MLE 表示的是语料库中，在前 n-1 个 word 相同（都是 w_{i-n+1}^{i-1}）的情况下，下一个 word 是 w^i 的概率，一般来说，$c\left(w_{i-n+1}^{i}\right)=\sum_{w_i}c\left(w_{i-n+1}^{i-1}\right)$。

  对于 N-gram 模型来说，一般有如下几种简单的模型：

  1. Unigram $P\left(w_i\right)$:当前 word 出现的概率和之前的 word 没有关系，完全取决于自身统计概率；
  2. Bigram $P\left(w_i\left|w_{i-1}\right.\right)$：当前 word 出现的概率和前一个词有关；
  3. Trigram $P\left(w_i\left|w_{i-1}w_{i-2}\right.\right)$：当前 word 出现的概率和前两个词有关。

- 模型评价标准

  好的语言模型的应该具有能力：1.拟合：需要对训练集有一个比较好的匹配；2.泛化：对未出现 word 也要评估的比较好。

  对于一个 n-gram 模型，其概率为 $P\left(w_i|\ w_{i-n+1}^{i-1}\right)$，因此可以计算到某个长度为 m 句子 s 的概率 $P(s)=\prod_{i=1}^{m+1}P\left(w_i|\ w_{i-n+1}^{i-1}\right)$。假定某个语料 G 由 l 个句子组成，则整个预料的概率为 $P(G)=\prod_{i=1}^{l}P\left(s_i\right)$,可以计算得到模型 n-gram 模型 $P\left(w_i|\ w_{i-n+1}^{i-1}\right)$ 对于预料的交叉熵为:

$$H_P(G)=-\frac{1}{W_G}\log_2 P(G)$$

  其中，W_G 表示的是语料 G 中的 word 的数量。

- 实验

  按照上述模型评价标准中的计算方法，我们应用 3 种基本的 N-gram（N=1,2,3）

模型分别在给定的中文语料中计算按词与按字的交叉熵。这里我们分别在《神雕侠侣》、《天龙八部》与全部小说合并的文本中进行计算。

实验主要分为两步骤：

1.文本预处理：

(a) 按行读取文本，删除该行的空格、制表符、换行符号等无效字符。

(b) 只保留中文文本以及中文符号，删除数字、字母以及其他无效字符。

(c) 将预处理文本进行存储，形成合并后的待处理文本。

2.计算：

(a) 1-gram：直接将 word 出现的次数除以总的 word 数目得到 $P(w_i)$

(b) 2-gram：计算 $w_i w_{i-1}$ 的总数除以 w_i 的总数得到 $P\left(w_i \middle| w_{i-n+1}^{i-1}\right)$

(c) 3-gram：计算 $w_i w_{i-1} w_{i-2}$ 的总数除以 w_i 的总数得到 $P\left(w_i \middle| w_{i-n+1}^{i-1}\right)$

	按词			按字		
	1-gram	2-gram	3-gram	1-gram	2-gram	3-gram
总	10.7256	6.5669	3.1619	9.0785	6.2077	3.9023
《天龙八部》	10.3616	5.6747	2.2367	8.9571	5.7270	3.0778
《神雕侠侣》	10.4763	5.7123	2.1517	8.9646	5.7597	3.0227

图 1 结果

3.链接：

https://github.com/chengquan50/nlp-1