深度学习与自然语言处理第一次作业

冯士轩

1 作业题目

学习英文文献中信息熵计算原理， 根据所学知识对文本中的中文字词的信息熵计算。

2 简介

信息熵是信息论的基本概念，描述信息源各可能事件发生的不确定性。20 世纪 40 年代，香农借鉴了热力学的概念，把信息中排除了冗余后的平均信息量称为“信息熵”，并给出了计算信息熵的数学表达式。信息熵的提出解决了对信息的量化度量问题。信息是个很抽象的概念，人们常常说信息很多，或者信息较少，但却很难说清楚信息到底有多少信息论之父香农第一次用数学语言阐明了概率与信息冗余度的关系。

汉字信息熵的作用在于提取文本的特征，使汉字的信息得到准确真接的表达。因此，汉字信息熵在中文文本分析有重要的意义。它可以用于文本检索和自动分类，从而提高文本的搜索效率和分类精度。

其中信息熵具有三个基本性质：

1、单调性，对于一个汉字或词组他发生的概率越高，所代表的信息越少，其信息熵的值越低；

2、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；

3、累加性，即多随机词组同时发生存在的总不确定性的量度是可以表示为各词组不确定性的量度的和，这也是广度量的一种体现

3 原理

汉字信息熵是指中文字符所包含的信息量，它是一种测量文字或字符拥有的信息熵的度量。计算汉字信息熵的方法是首先量化中文文本，然后通过计算每个字符出现的概率，经过概率和熵值的统计，最终得出汉字信息熵的大小。

N-Gram 语言模型就是用来计算一个句子的概率的模型。给定一个句子 $S=W_1,W_2$，….W_n。那么这个句子出现的概率为

$$P(S) = P(W_1, W_2, \dots, W_K) = p(W_1)P(W_2 \mid W_1) \dots P(W_K \mid W_1, W_2, \dots W_{k-1})$$

假设一个文本$X = \{\dots X_{-2}, X_{-1}, X_0, X_1, X_2 \dots\}$，用 P 代表 X 的概率分布，$E_P$ 代表关于 P 的期望。那么关于 X 的信息熵为：

$$H(X) \equiv H(P) \equiv -E_P \log\ P(X_0 \mid X_{-1}, X_{-2}, \dots)$$

根据大数定理，当统计量足够大的时候，词、二元词组、三元词组出现的概率大致等于其出现的频率。

故一元模型的信息熵计算公式为

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

其中 P(x)可近似于每个字在语料库中的出现频率。

二元语言模型的信息熵计算公式为：

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

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率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。

三元语言模型的信息熵计算公式为：

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

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率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 实验操作

4.1 数据处理

数据库中的语料是 txt 格式的，其中包含无用的符号和出现频率很高但对研究无意义的词语。在信息检索时，为节省存储空间和提高搜索效率，在处理自然语言文本之前过滤掉停词表内的字词。

4.2 计算一元语言模型信息熵

本文的统计语言模型是基于字词的，因此需要对中文句子进行分词。选用 python 中文分词系统 jieba 对句子进行分词。通过 jieba.lcut 给定中文字符串，分解后返回一个列表，需要用 for 循环访问。

通过对文本中所分词得到的列表进行查询可以得到，经过停词处理后的文本词库总字数位 4293481，其中不同的字数为 173010，出现频率最高的前五个字分别为('的', 115600)、('了', 104515)、('他', 64708)、('是', 64458)、 ('道', 58623)。根据一元信息熵公式可以求出一元信息熵为 12.179278029070783。

```
词库总词数： 4293481   不同词的个数： 173010
出现频率前5的1-gram词语： [('的', 115600), ('了', 104515), ('他', 64708), ('是', 64458), ('道', 58623)]
1gram: 12.179278029070783
```

4.3 计算二元语言模型信息熵

因为二元信息熵需要考虑上文产生的影响，所以不能使用停词表处理后的数据。首先只需使用正则表达式对其中的特殊符号和数字进行删除，然后对选择的词组进行中文检测，来选取中文字符所组成的词组，从而来计算信息熵。

通过运算可以得到，文本词库总字数位 4253331，其中不同的字数为 1950264，出现频率最高的前五个字分别为（'叫道', 5009），（'道我', 4953），（'笑道', 4271），（'听得', 4202），（'都是', 3905）。根据二元信息熵公式可以求出二元信息熵为 6.946045347794212。

```
词库总词数： 4253331   词的种类数目： 1950264
出现频率前5的2-gram词语： [['叫道', 5009], ['道我', 4953], ['笑道', 4271], ['听得', 4202], ['都是', 3905]]
2gram: 6.946045347794212
```

4.4 计算三元语言模型信息熵

三元信息熵计算与二元信息熵数据处理方式相似，只是计算公式不同。

通过运算可以得到，文本词库总字数位 4194262，其中不同的字数为 3481425，出现频率最高的前五个字分别为（'忽听得', 1137），（'站起身来', 733），（'哼了一声', 573），（'笑道你', 566），（'吃了一惊', 534）。根据三元信息熵公式可以求出三元信息熵为 2.350668388879347。

```
词库总词数： 4194262   词的种类数目： 3481425
出现频率前5的3-gram词语： [['忽听得', 1137], ['站起身来', 733], ['哼了一声', 573], ['笑道你', 566], ['吃了一惊', 534]]
3gram: 2.350668388879347
```

5 实验结果分析

对比三种语言模型计算得到的结果可以看出随着 N 取值变大，文本中信息熵不断减少，而词的种类增多。分析出现此现象的原理，可知当 N 越大时，词长的增大导致了字排列组合所形成的词种类增多。而文本中的字词一般为有意义的固定词组，固定的的词组使得由

字组成词和组成句的不确定性减少，文章变得更加有序，从而文本的信息熵降低。