作业二 实验报告

一、 实验目的

对 20 Newsgroups dataset 文档集使用 Naïve Bayes 方法进行分类。

二、 实验环境

Windows 10 + Python3.6.5

三、 实验步骤

预处理方法同作业一。然后划分训练集与测试集，每个类别取 80%的文档作为训练集，20%为测试集，并放入相应的文件夹。

对于一个文档 d 和一个分类 c

$$P(c|d)=\frac{P(d|c)P(c)}{P(d)}$$

MAP 是最大化后验概率，或者说：最有可能的类别。

根据贝叶斯规则，转化为

$$=\underset{c\in C}{\operatorname{argmax}}\frac{P(d|c)P(c)}{P(d)}$$

去掉共同的分母，转化为

$$=\underset{c\in C}{\operatorname{argmax}}P(d|c)P(c)$$

文档 d 表示为 特征 x1 .. xn

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(d \mid c)P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c)P(c)$$

多项朴素贝叶斯独立假设

词袋模型假设：假设位置并不重要

条件假设：假设特征概率 P(xi,cj)是独立的，在类别 c 给出的情况下。

由此可以推出以下等式：

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c)P(c)$$

$$c_{NB} = \underset{c \in C}{\operatorname{argmax}} P(c_j) \prod_{x \in X} P(x \mid c)$$

其中 $P(c_j)$为第 j 类文档的先验概率，P(x|c)为第 c 类文档中单词 x 出现的概率，以上值均可由频率统计得到：

$$\hat{P}(c_j) = \frac{doccount(C = c_j)}{N_{doc}}$$

$$\hat{P}(w_i \mid c_j) = \frac{count(w_i, c_j)}{\sum_{w \in V} count(w, c_j)}$$

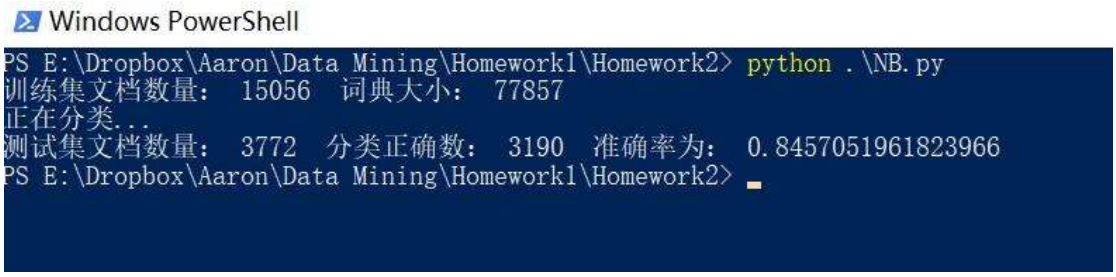
为避免某一项 P(x|c)为 0（测试集中包含训练集没有的单词）影响分类结果，为 P(x|c)采用平滑的操作：

$$\hat{P}(w_i \mid c) = \frac{count(w_i, c) + 1}{\sum_{w \in V} \left(count(w, c) + 1 \right)}$$

$$= \frac{count(w_i, c) + 1}{\left(\sum_{w \in V} count(w, c) \right) + |V|}$$

|V|为训练集词典的大小（不计重复）

四、 实验结果


Windows PowerShell
PS E:\Dropbox\Aaron\Data Mining\Homework1\Homework2> python .\NB.py
训练集文档数量： 15056 词典大小： 77857
正在分类...
测试集文档数量： 3772 分类正确数： 3190 准确率为： 0.8457051961823966
PS E:\Dropbox\Aaron\Data Mining\Homework1\Homework2>