**Lab 8 Naïve Bayes**

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | **17363011** | 学院 | 专业 |
| 姓名 | **陈政培** | **智能工程学院** | **智能科学与技术** |

1. 实验目的： **利用python的文本处理能力将文档切分成词，通过集合元素的唯一性生成词列表（不包含重复词汇），进而构建词向量（词集向量或词袋向量），从词向量计算概率，然后构建分类器对邮件文档进行垃圾邮件分类**
2. 实验环境：python3.6.8、vs code
3. 实验步骤：
4. 切分文本词
5. 生成词汇表

提供的bayes.py有少量的语法错误和格式错误，需要修正后正常使用

1. 生成词向量

在bayes.py文件中，setOfWords2Vec函数中

returnVec[vocabList.index(word)] = 1

仅实现了统计这个词语是否出现，为了让词向量能够统计量我们将代码更改为

returnVec[vocabList.index(word)] += 1

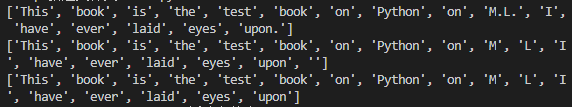
1. 训练算法，计算概率

计算出概率P(c1)、P(w|c1)、P(w|c0)

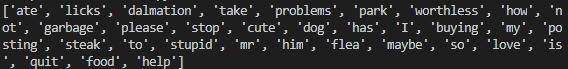
1. 利用50封邮件进行测试和训练

因为每一次训练集和测试集都是随机生成的，所以每一次运行结果可能不太相同

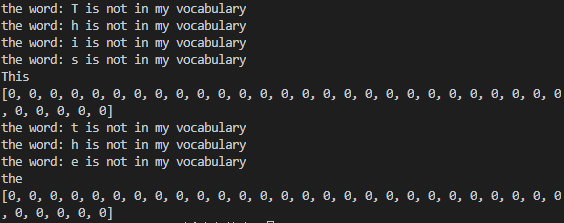
1. 实验结果与分析：
2. 切分词——textParse()函数内容实现



1. 生成词汇表

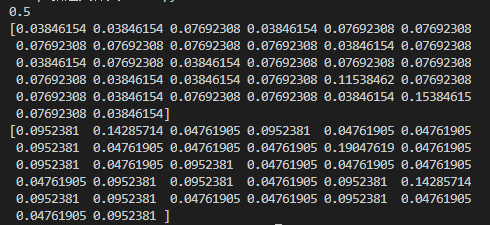


1. 生成词向量

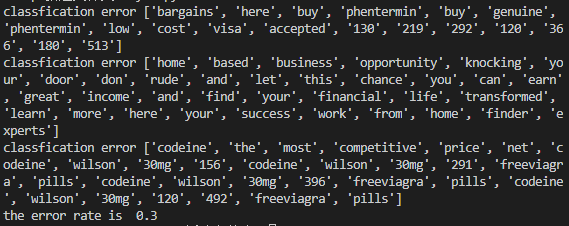


由于单词This和单词the并没有在词汇表中出现所以向量全部为0

1. 计算概率



1. 测试与训练



本次运行error rate是0.3，也出现过0.1或者0.5

1. 作业：
2. 利用sklearn中BernoulliNB分类该邮件数据集

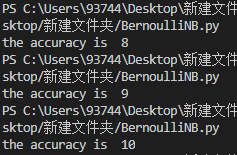
from sklearn.naive\_bayes import BernoulliNB

from sklearn.metrics import accuracy\_score

主要使用的就是sklearn库中的这两个，一个是BernoulliNB模型，一个是用于测试正确率的accuracy\_score

其他数据预处理，训练集测试集随机分配都是沿用的bayes.py中的函数

测试结果：



训练正确率相对比naïve bayes稳定高一些，稳定一些

1. Bayes.py中语句更换

详见代码文件bayes.py

1. 将向量集用TF-IDF词向量替代，测试分析结果

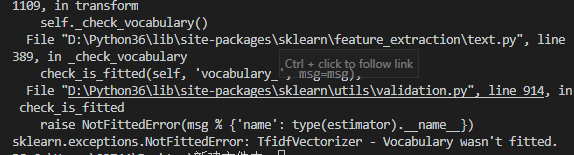
from sklearn.feature\_extraction.text import TfidfTransformer

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.feature\_extraction.text import TfidfVectorizer

采用了sklearn中的几个转换器，其中CountVectorizer进行词频统计，TfidfTransformer将矩阵转化为TF-IDF矩阵

但是无论如何调试数据类型，生成的TF矩阵总是无法和调用的分类器相互匹配，出现数据类型的报错



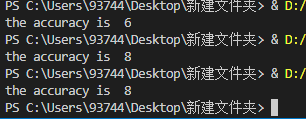
1. 编程实现
2. 基于tensorflow实现朴素贝叶斯分类器

由于是基于tensorflow1的代码基础进行更改的，所以在tensorflow2中无法正常运行

tf.distributions.Normal(loc=mean, scale=tf.sqrt(var))

此函数distributions属性只在1.0版本有定义，2.0版本取消了这个属性并且没有制作新的同类型的参数

在1.0环境下运行accuracy结果



相比BernoulliNB性能会差一些

1. 实验总结：

* TF-IDF和普通的词袋模型还是有一定差异的，尤其是在sklearn库中提供的TF-IDF转换生成的是一个根据输入数据变化维度的矩阵
* 先用词袋模型筛选出一些高热度词汇，再用TF-IDF计算其权值，得到词袋模型中词汇的TF-IDF值，值越高说明该词区分每条语句的效果越好。两者联合使用的效果会更好
* Sklearn.naive bayes库中除了BernoulliNB还有GaussianNB等不同的分类器，使用方法基本类似，都是先生成对应xxxNB()的类，然后调用相关函数方法

1. 参考文献
2. <https://yq.aliyun.com/articles/408869>
3. <https://www.jianshu.com/p/0422853b57a8>
4. <https://stackoverflow.com/questions/55010025/tensorflow-probability-error-attributeerror-module-tensorflow-probability-ha>
5. <https://www.cnblogs.com/to-creat/p/6888295.html>
6. <https://scikit-learn.org/dev/modules/generated/sklearn.naive_bayes.BernoulliNB.html#sklearn.naive_bayes.BernoulliNB>