naiveBayes实验报告

2018年11月15日 22:49

实验过程:

一、数据预处理partition data(index,category,proportion=0.8)

直接使用knn算法中预处理之后的数据,将数据集按照80%训练集20%测试集的划分比例对数据集进行划分,共划分五次

二、统计训练集中每个类别下的单词以及单词出现次数countWords(trainFile)

建立cate_word(类别_单词)字典,记录每个类别中某个单词出现的次数,然后统计当前类中总的单词数目

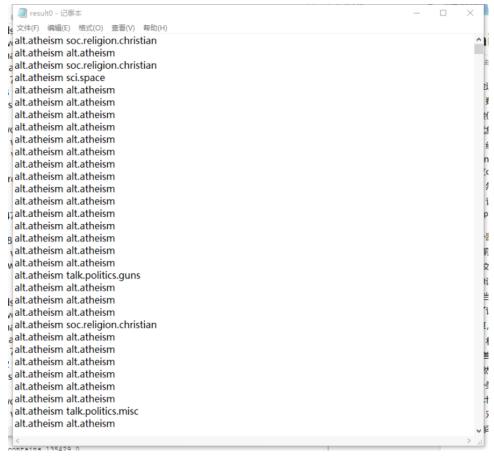
三、计算测试文档属于某个类的概率

computeProbability(cate,testWords,wordsNumOfCate,wordsNum,wordsOfCate):

某个测试文档属于某个类的概率=文档中词在当前类中出现的概率的乘积*当前类别出现的概率

- 1) 文档中词在当前类中出现的概率=当前类中该单词出现次数/(当前类的单词总数+训练样本中的特征词总数)
- 2) 当前类别出现的概率=当前类别中文档总数/训练样本中总的文档数目 为了计算方便,防止很小的数相乘造成溢出,对乘法运算取log变成加法运算,最后返回计算的概率值
- 四、朴素贝叶斯分类naiveBayes(trainFile,testFile,resultFile):
- 1) 首先通过countWord()方法获得训练文件的统计结果
- 2) 然后对测试文件中的每个文档进行朴素贝叶斯分类, 计算该文档属于各个类的概率大小, 保留概率最大的类作为当前文档的分类结果
- 3) 计算分类的准确率,并将分类的结果存入结果文件中
- 五、对五个不同的训练集-测试集 集合调用naiveBayes方法,对其分类准确率求均值,并且保存到文件中。

实验结果:



五次交叉验证的实验分类准确率如下

1	0.810235
2	0.861089
3	0.857446
4	0.869854
5	0.842777
average	0.848280

实验分析:

五次实验平均分类准确率0.848高于之前所做的knn分类器,在运行效率上每次实验均在一分钟内,明显高于knn分类器。