南华大学计算机学院

**实 验 报 告**

**课程名称 机器学习**

**实验名称 朴素贝叶斯实现垃圾信息分类**

**姓 名 贺凌云**

**学 号 20204352203**

**专 业 人工智能**

**班 级 软智2002班**

**日 期 2022年6月16日**

南华大学

**1.实验目的**

掌握使用机器学习的sk-learn实现朴素贝叶斯分类。

基于python语言实现朴素贝叶斯分类，将SMS数据集进行分类。

**2.实验环境**

Windows 10

Pycharm 2022.12

**3.实验要求**

(1)划分训练集和测试集（测试集占20%）

(2)对测试集的预测类别标签和真实标签进行对比

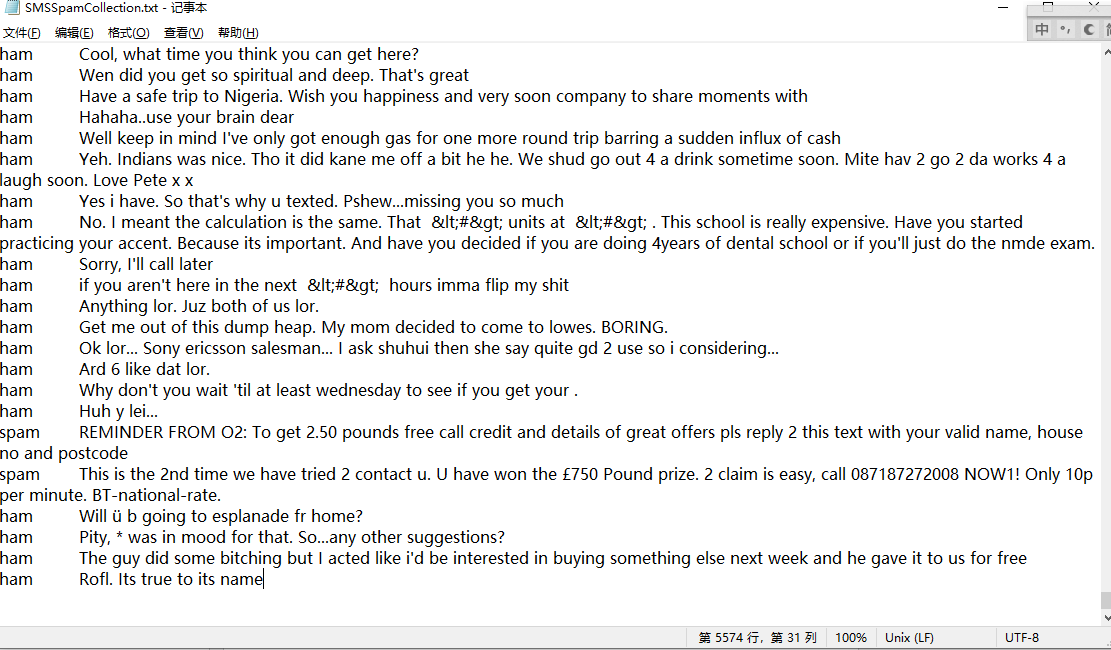
(3)掌握特征提取方法

(4)输出分类的准确率

4.实验设计

（1）数据说明

数据整理自著名的SMS短信数据集，数据集涵盖了大量来自SMS社交网络的各类信息，包含了许多垃圾信打开下载的.txt⽂件，可以看到数据集长这样，标签（ham和spam,spam就是指垃圾短信）与⽂本之间的分隔符是⼀个tab键,也就是‘\t’



（2）字段说明

SMS数据集中所有的信息都带有标签，HAM代表普通信息，而SPAM代表垃圾短信，在测试时会把百分之20的数据标签抹除作为测试集，而80％作为模型的训练集。

朴素贝叶斯算法分类垃圾短信，就是要找出哪些单词最常出现在垃圾短信中，将这些最常出现的单词，作为特殊单词，⽤来过滤短信。接下来就是⽤词袋⽅法处理⽂本信息，也就是统计⼀⼤段话⾥的不同单词的出现次数，最后得到⼀个频率矩阵，矩阵的⾏就是数据集⾥的每⼀⾏短信，矩阵的列就是短信⾥每个单词，元素值就是该单词的出现频率。有了频率，那么特殊单词也就能找到了。可以⽤sklearn库提供的CountVectorizer()⽅法实现词袋处理。

5.程序和结果

from sklearn.naive\_bayes import BernoulliNB

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.model\_selection import train\_test\_split

import pandas as pd

path = 'SMSSpamCollection.txt'

names = ['labels', 'messages']

data = pd.read\_csv(path, sep='\t', header=None, names=names)

data = data.replace({'ham': 0, 'spam': 1}) #

print('数据集展⽰：')

print(data)

print('\n----------------------------------\n')

X = data['messages']

y = data['labels']

x\_train, x\_test, y\_train, y\_test = train\_test\_split(X, y, train\_size=0.8, random\_state=123)

# 实现词袋模型

vector\_nominal = CountVectorizer()

vector\_burnout = CountVectorizer()

# 伯努利模型分类垃圾短信

train\_matrix = vector\_burnout.fit\_transform(x\_train)

test\_matrix = vector\_burnout.transform(x\_test)

Bernoulli = BernoulliNB()

clm\_bernoulli = Bernoulli.fit(train\_matrix, y\_train)

result\_burnout = clm\_bernoulli.predict(test\_matrix)

print('伯努利模型的预测结果,类型，长度：')

print(result\_burnout, type(result\_burnout), result\_burnout.shape)

print('伯努利模型的前⼀百个预测结果：')

print(result\_burnout[0:100])

print('伯努利模型准确率评分：' + str(clm\_bernoulli.score(test\_matrix, y\_test)))

文本

描述已自动生成

文本

描述已自动生成

经过训练和测试得出：算法的预测准确率是0.9802690582959641

**6.实验总结**

通过学习和使用sk-learn算法实现朴素贝叶斯分类，我对其有了进一步的认识：

朴素贝叶斯在 scikit-learn 上有伯努利朴素贝叶斯（就是咱们这个例子使用到的模型），多项式朴素贝叶斯、高斯朴素贝叶斯，这些在刘建平的文章《scikit-learn 朴素贝叶斯类库使用小结》中有介绍。本题具体用法可说：

1、打开下载的.txt⽂件，标签（ham和spam,spam就是指垃圾短信）与⽂本之间的分隔符是⼀个tab键,也就是‘\t’

2、⾸先⽤pd.read\_csv函数读取该数据集时要注意设置分隔符sep=’\t’，然后⽤replace⽅法把“ham”标签⽤0替代，“spam”⽤1替代，⽅便看预测结果。

3、朴素贝叶斯算法分类垃圾短信，就是要找出哪些单词最常出现在垃圾短信中，将这些最常出现的单词，作为特殊单词，⽤来过滤短信。接下来就是⽤词袋⽅法处理⽂本信息，也就是统计⼀⼤段话⾥的不同单词的出现次数，最后得到⼀个频率矩阵，矩阵的⾏就是数据集⾥的每⼀⾏短信，矩阵的列就是短信⾥每个单词，元素值就是该单词的出现频率。有了频率，那么特殊单词也就能找到了。可以⽤sklearn库提供的CountVectorizer()⽅法实现词袋处理。

4、将训练数据和测试数据输⼊到词袋模型⾥，就可以得到对应的频率矩阵。最后运⽤sklearn提供的伯努利模型和对垃圾短信进⾏分类。

5、两个模型返回的分类结果都是长度为1115（我设置的训练集占⽐为80%），类型为ndarray的列表。最终，伯努利模型的准确率为0.980。