基于 spark mllib 库的垃圾短信识别

[redacted]

一、spark 环境的配置过程

1. 安装 JDK，把 java 路径放进环境变量

去 oracle 官网下载最新版 JDK，下载好后双击 exe 文件运行，安装目录为 C:\Program Files\java

Win10 把 java 添加进环境变量的步骤：在搜索中输入【编辑系统环境变量】【高级】【环境变量】，在系统变量中找到“Path”变量，【编辑】，添加安装的 JDK 目录下的 bin 文件夹路径名：C:\Program Files\java\jdk1.8.0_101\bin。

在【环境变量】中添加 CLASSPATH，路径仍是上述路径。在【环境变量】中添加 JAVA_HOME，路径为 C:\Program Files\Java\jdk1.8.0_101

2. 安装 scala

去 scala 官网下载最新版 scala，下载好后双击 msi 文件运行，安装目录为 C:\Program Files\scala，默认会将 Scala 的 bin 目录添加到 PATH 系统变量。

3. 安装 spark

去 apache 官网下载预编译好的 spark，将 tgz 文件在 C:\SPARK 里解压，将 Spark 的 bin 目录（C:\SPARK\spark\bin）添加到系统变量 PATH 中。

4. 安装 hadoop

去 apache 官网下载预编译好的 hadoop，将 tar.gz 文件在 C:\Program Files\hadoop 里解压，添加了 HADOOP_HOME 系统变量：C:\Program Files\hadoop\bin

5. 点击 spark-shell

开启一个新的 cmd，然后直接输入 spark-shell 命令。

二、程序运行过程中遇到的问题

1. 在【txt2csv.py】中遇到编码问题

运行以下语句时，python 报错：UnicodeDecodeError: 'gbk' codec can't decode byte 0xbf。这是因为短信中带有各种各样的字符，超出了 gbk 编码范围。解决办法：直接使用 pandas 中的 read_table 函数，encoding=utf-8 即可把原始数据读取成 python 中的 dataframe，省去把数据转换成 csv 的过程。

```
lines = readfile.readlines()
```

2. 在【clean_cut.py】和【getfeatures.py】中遇到 byte 对象问题。运行以下语句时，python 报错：can't concat bytes to str。这是因为在 python3

中，str 对象 encode 后变成了 byte 对象，而 str 对象和 byte 对象不能直接相加，且 write 函数只接受 str 对象，因此直接把 encode('utf-8')去掉或在 encode('utf-8')后加上.decode()。

```
outfile.write((str(me_cate)+','+outstr).encode('utf-8')+'\n')
```

3. 【getfeatures.py】中 type(line)!=unicode 报错，导入 numpy 包，改为 type(line)!=np.unicode。

三、文本预处理

1. 把原始数据集从 txt 格式转成 csv 格式

将原始数据将 whole.txt 文件转为 CSV 格式的 whole.csv，这样 Python 中可以使用 pandas 工具包，来读取 csv 文件。

2. 分词

利用 jieba 包对短信文本进行中文分词，去除其中的标点、无用词，效果如下图所示。

	0	商业秘密 的 秘密性 那 是 维系 其 商业价值 和 垄断 地位 的 前提条件 之一
159992	1	t 娱乐 会 所 这么 多 哪里 有 保障 当然 首选 太阳城 您 入 多少 回
159993	1	唯雅 舞蹈 x 周年 庆典 钢管舞 爵士舞 瑜伽 肚皮舞 优惠 多多 报卡送 卡 再 送 专...
159994	1	浪漫 香榭 丽簇 桥千盛 携手 三八节 给 您 送 美丽 xx 元 抢购 xxx 元 祛斑 ...
159995	1	尊敬 的哥 比兔 新 老客户 您好 闹元宵 迎三 八哥 比兔 大 赠送 活动 开始 啦 为 ...
159996	1	您好 本人 在 上海 调车 多年 公司 每天 全国 各地 上 千部 回程 车 价格 优惠 有...

3. 下采样

由于垃圾短信和正常短信在原始数据集中的比例为 1:9，非常不平衡，因此需要对正常短信作下采样处理，即随机删除 8/9 的正常短信。分词和下采样之后的数据保存在 result.csv 里。

3. 文本表示

利用 sklearn 中 HashingVectorizer 方法把分词和下采样后的短信内容转化为 100 维的哈希向量，由于用哈希向量法转换生成的特征向量中每一项都是一个[-1,1]之间的数，而贝叶斯方法接收的特征矩阵是每一项都大于 0 的特征矩阵，因此在每一条内容转成特征向量后需要将其每一项都加 1。文本表示的结果保存在 feature.txt 里，向量如下图所示。

	0
0	0,1.0 1.0 1.0 1.0 1.0 1.3535533905932737 1.0 1...
1	1,0.757464374963667 1.0 1.0 1.0 1.0 1.0 1.2425...
2	0,1.0 1.0 1.0 1.0 1.3779644730092273 1.0 1.0 1...
3	0,1.5 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 ...
4	0,1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 0.5 1.0 1.0 ...

四、建立模型

1.朴素贝叶斯

按照 4:1 的比例随机划分训练集和测试集，在 spark 里建立朴素贝叶斯模型，平滑参数为 1，预测结果如下表所示：

表 1 朴素贝叶斯预测结果

预测\真实	正常短信	垃圾短信
正常短信	13817（TP）	4472（FP）
垃圾短信	2263（FN）	11467（TN）
precision = TP/(TP+FP) = 75.55%		
recall = TP/(TP+FN) = 85.93%		
F1 = 2*(Precision*Recall)/(Precision+Recall) = 80.40%		

可以看到，precision 为 75.55%，即预测为正常的短信中，有 75.55%的短信是正常的，24.45%被误认为是垃圾短信。Recall 为 85.93%，即所有正常短信中，有 85.93%被预测成正常的，14.07%被误认为是垃圾短信。F1 值为 precision 和 recall 的调和均值，F1 值为 80.40%，说明朴素贝叶斯分类模型较为稳健。

2.随机森林

按照 4:1 的比例随机划分训练集和测试集，在 spark 里建立随机森林模型，平滑参数为 1，预测结果如下表所示：

表 2 随机森林预测结果

预测\真实	正常短信	垃圾短信
正常短信	13835（TP）	4466（FP）
垃圾短信	2245（FN）	11473（TN）
precision = TP/(TP+FP) = 75.60%		
recall = TP/(TP+FN) = 86.03%		
F1 = 2*(Precision*Recall)/(Precision+Recall) = 80.48%		

可以看到，随机森林的预测效果与朴素贝叶斯相似， precision 比朴素贝叶斯高 0.05 个百分点，recall 高 0.1 个百分点，F1 值高 0.08 个百分点。预测效果整体比朴素贝叶斯稍好。