**程序报告**

学号：2013536 姓名：汤清云

1. **问题重述**

（简单描述对问题的理解，从问题中抓住主干，必填）

====================================================================

1. 使用基于python的pandas，numpy，sklearn等库进行训练数据的相关特征处理。
2. 从训练数据中设计出基础模型，将模型应用在测试数据上，查看其分类功能。
3. 使用停用词包以去除停用词，将每句话拆分为短语；转化为词频向量，构建TF-IDF权重矩阵，提取特征构建模型。并使用label标签其内容类别。
4. **设计思想**

（所采用的方法，有无对方法加以改进，该方法有哪些优化方向（参数调整，框架调整，或者指出方法的局限性和常见问题），伪代码，理论结果验证等… **思考题，非必填**）

====================================================================

采用方法：归一化、修改文本向量化函数，调节参数。

1. 选择**TfidfVectorizer**算法，优先考虑在文档中出现频率高的词语以体现同类文本的特点。
2. 调整ngram\_range=(1, 2)进行字符串的前后组合，选取1-2个词为组合方式，构造出新的词袋标签。
3. 使用('aaaaaaaaaaa', MaxAbsScaler())进行归一化处理，提升模型准确度。
4. 修改test\_size=0.3，提高训练数据占比，优化模型。
5. **代码内容**

（能体现解题思路的主要代码，有多个文件或模块可用多个"===="隔开，必填）

====================================================================

# 将sms的正负比例调整为1：1

sms = pd.read\_csv(data\_path, encoding='utf-8')

sms\_pos=sms[(sms['label']==1)].sample(frac=1.0)

sms\_neg=sms[(sms['label']==0)].sample(frac=1.0)[:len(sms\_pos)]

sms=pd.concat([sms\_pos,sms\_neg],axis=0).sample(frac=1.0)

#===================================================================

('tfidf', TfidfVectorizer(token\_pattern=r"(?u)\b\w+\b",stop\_words=stopwords,ngram\_range=(1,2))),

('aaaaaaaaaaa', MaxAbsScaler()),

('classifier', MultinomialNB())

======================================================================

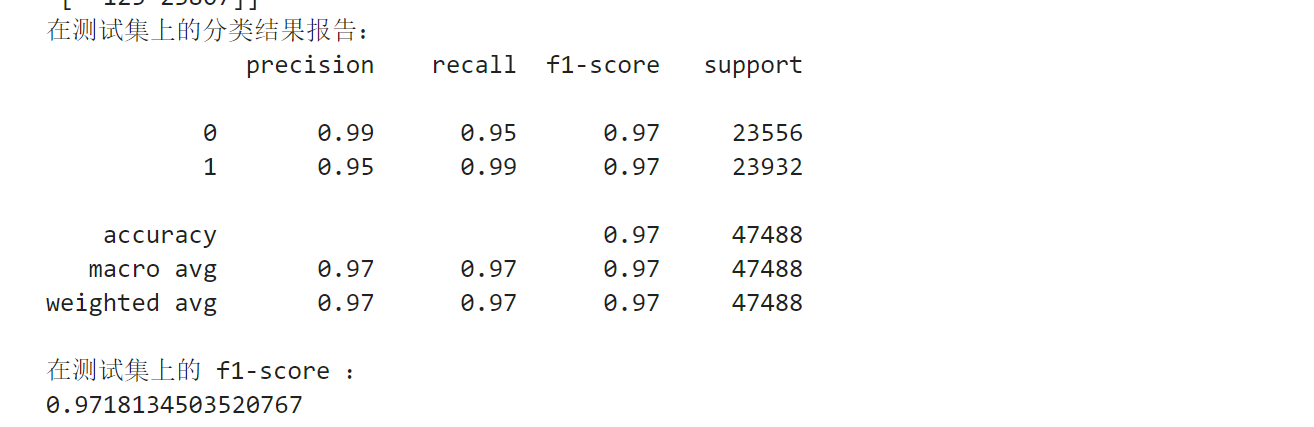
1. **实验结果**

（实验结果，必填）

====================================================================

判别结果如下：





1. **总结**

（自评分析（是否达到目标预期，可能改进的方向，实现过程中遇到的困难，从哪些方面可以提升性能，模型的超参数和框架搜索是否合理等），**思考题，非必填**）

====================================================================

1. 达到目标预期，提升了模型分类准确率至9/10
2. 实验过程困难：对分词函数的构造极为生疏，以及对朴素贝叶斯思想不熟悉。
3. 模型超参数只修改了ngram\_range，是目前准确度最高的赋值。