**程序报告**

学号：2111454 姓名：李潇逸

1. **问题重述**

通过机器学习进行垃圾短信的识别。

1. **设计思想**

利用机器学习进行模型构建和训练

1. **代码内容**

import os

os.environ["HDF5\_USE\_FILE\_LOCKING"] = "FALSE"

# ---------- 停用词库路径，若有变化请修改 -------------

stopwords\_path = r'scu\_stopwords.txt'

# ---------------------------------------------------

def read\_stopwords(stopwords\_path):

"""

读取停用词库

:param stopwords\_path: 停用词库的路径

:return: 停用词列表，如 ['嘿', '很', '乎', '会', '或']

"""

stopwords = []

# ----------- 请完成读取停用词的代码 ------------

#----------------------------------------------

data = open(stopwords\_path,"r")

while True:

line = data.readline()

line = line.replace("\n","")

if not line:

break

else:

stopwords.append(line)

return stopwords

# 读取停用词

stopwords = read\_stopwords(stopwords\_path)

# ----------------- 导入相关的库 -----------------

from sklearn.pipeline import Pipeline

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.naive\_bayes import BernoulliNB

from sklearn.naive\_bayes import MultinomialNB

from sklearn.naive\_bayes import ComplementNB

# pipline\_list用于传给Pipline作为参数

pipeline\_list = [

# --------------------------- 需要完成的代码 ------------------------------

# ========================== 以下代码仅供参考 =============================

('cv', CountVectorizer(token\_pattern=r"(?u)\b\w+\b", stop\_words=stopwords)),

('classifier', MultinomialNB())

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

# ------------------------------------------------------------------------

]

# 加载训练好的模型

from sklearn.externals import joblib

# ------- pipeline 保存的路径，若有变化请修改 --------

pipeline\_path = 'results/pipeline.model'

# --------------------------------------------------

pipeline = joblib.load(pipeline\_path)

def predict(message):

"""

预测短信短信的类别和每个类别的概率

param: message: 经过jieba分词的短信，如"医生 拿 着 我 的 报告单 说 ： 幸亏 你 来 的 早 啊"

return: label: 整数类型，短信的类别，0 代表正常，1 代表恶意

proba: 列表类型，短信属于每个类别的概率，如[0.3, 0.7]，认为短信属于 0 的概率为 0.3，属于 1 的概率为 0.7

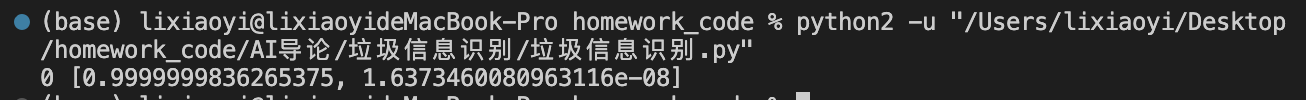
"""

label = pipeline.predict([message])[0]

proba = list(pipeline.predict\_proba([message])[0])

return label, proba

1. **实验结果**





1. **总结**

达到了与其设计目的。

改进方向有如下几点：

* 选择更好的模型，使准确率提升。
* 考虑使用深度学习。