## 基于Bayes 的垃圾邮件分类器的设计与实现

1. **开发环境：**
   1. Windows 10 64 位系统 / macOS High Sierra 10.13.2
   2. Python 3.6.4 (64 位)，需要安装下列依赖库

可以直接 pip install xxx，附上开发时用的版本

* + 1. jieba – 0.39
    2. numpy – 1.14.0
    3. Django – 2.0.1
  1. PyCharm 2017.3.2，可直接创建 Django 项目，免去许多配置

为保证正确运行，请尽量使用相同的开发环境。 涉及到python 的路径设置，请在 Pycharm中调试时，选择正确的运行路径。

注意：涉及到中文文本的处理，请使用 UTF-8 编码！

1. **项目结构：（基于 Django 开发的Web界面）**
2. polls，管理整个 Web 的前后端交互、请求、渲染等
3. spam，后端的处理。也可以分别单独运行该项目下的 clean\_data.py、preprocess\_data.py 、bayes.py 和 bayes\_uci.py；这些也作为网页后端的数据处理的接口和辅助函数等。子文件夹如下：
   * data 文件夹，经过清洗过的 trec06c 中的数据，因为源数据中有的含有乱码，程序无法处理，所以去掉了大约10000个无效文件
   * model 文件夹，存储训练好的模型参数，在单独测试时，可以直接读取，不必重复训练
   * myham、myspam 文件夹，用户自定义添加的正常邮件和垃圾邮件
   * UCI 文件夹，是 UCI 数据集的文件
   * index 文件，来自trec06c 数据集中的 data/full/index，放在这里是为了方便操作，如果用户请求修改数据集中邮件的标签，会随之更新
4. SpamBayesWeb 文件夹，Django 项目的设置
5. Static 文件夹，网页端使用到的 css 样式和一些 js 代码，用于渲染网页
6. Templates 文件夹，html 网页代码，通过 Django 注入数据，作为前端 UI
7. **主要代码文件说明**

本项目是 Django 框架开发的，也可以抛开网页 UI 直接运行代码，命令行输出。

* 1. Bayes 垃圾邮件分类器的实现，主要在 spam 文件夹下
* bayes.py，基于 trec06c 数据集的分类，相关运行方式在代码注释中给出
* bayes.py，基于 UCI 数据集的分类器实现，相关运行方式在代码注释中给出
* clean\_data.py，对 trec06c 数据集的数据进行清理，将清洗过的数据放在 spam/data 文件夹下。注意：只需要运行一次，在运行时，请将 trec06c 数据集的压缩包也解压在 spam 文件夹中（所给代码中已经删去，但保留了 data 数据）
* preprocess\_data.py，对 trec06c 数据进行预处理，封装一些 bayes 分类时的一些方法
  1. Django 框架，网页 UI
* 主要是 SpamBayesWeb/polls 下的 views.py，里面提供了各种请求的处理和前端渲染

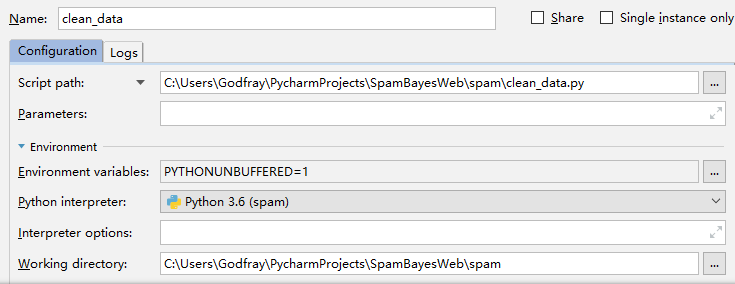
1. **程序运行方式**

可以通过 Pycharm 导入打开，支持命令行运行和 Web UI 运行。为了提高演示的效率，对实际演示的数据集进行了缩减，完整结果请根据注释中 main 函数的地方或者是 views.py 的 index 函数中的对应地方修改。

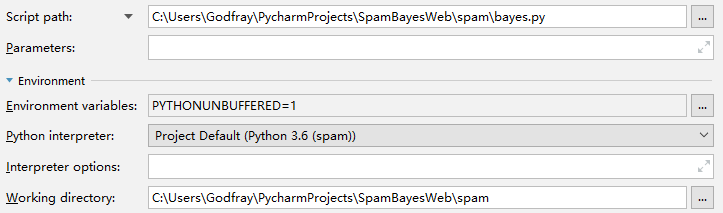
*注意：由于 python 环境和项目设置的路径不同，需要根据实际情况修改。直接运行可能会出错。有的时候需要重新选择一下 Python interpreter 就没有问题了。*

1. 只使用命令行的分类，在 Pycharm 中运行比较方便，也可以在命令行，切换到 spam 目录中，然后输入 “python clean\_data.py” （或者其他的 .py 文件）运行，等待结果输出。 Pycharm 中的设置主要是 Script path 和 Working directory

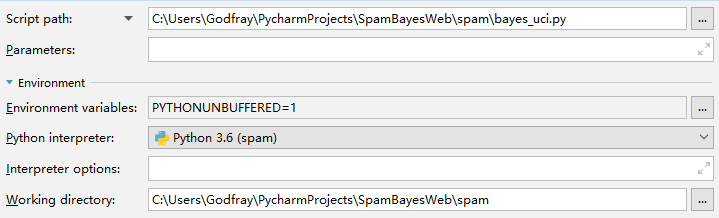
* 首先对数据进行清洗，选择合法有效的数据，运行 spam 中的 clean\_data.py，设置如下：



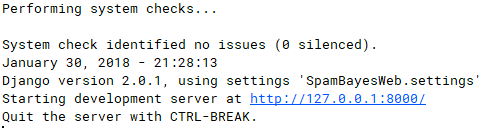
* Trec06c 数据集测试，运行 spam 中的 bayes.py



* UCI 数据集测试，运行 spam 中的 bayes\_uci.py



B) Web UI 演示运行，命令行中，在第一层 SpamBayesWeb 文件夹下，使用 “python manage.py runserver 8000” 运行，当有如下的提示后，可以访问 <http://127.0.0.1:8000> ，看到项目的 UI 界面，下面的操作请直接在网页上操作即可。



推荐在 Pycharm 中调试与运行，保持默认配置即可：

