

<b>1 引言 .....</b>	<b>2</b>
1.1 编写目的.....	2
1.2 背景.....	2
1.3 定义.....	2
1.4 参考资料.....	2
<b>2 用途 .....</b>	<b>3</b>
2.1 功能.....	3
2.2 性能.....	3
2.2.1 精度.....	3
2.2.2 时间特性.....	3
2.2.3 灵活性.....	4
2.3 安全保密.....	4
<b>3 运行环境 .....</b>	<b>4</b>
3.1 硬设备.....	4
3.2 支持软件.....	4
3.3 数据结构.....	5
<b>4 使用过程 .....</b>	<b>6</b>
4.1 安装与初始化.....	6
4.2 输入.....	7
4.2.1 输入数据的现实背景.....	7
4.2.2 输入格式.....	8
4.2.3 输入举例.....	8
4.3 输出对每项输出作出说明.....	8
4.3.1 输出数据的现实背景.....	8
4.3.2 输出格式.....	9
4.3.3 输出举例.....	9
4.4 文卷查询.....	10
4.5 出错处理和恢复.....	11
4.6 终端操作.....	11

# 用户手册（GB8567——88）

## 1 引言

### 1.1 编写目的

本用户手册的编写是为了帮助用户更好地了解和使用本小组开发的《Clean MainBox》邮箱系统，提高用户与软件的亲和度。用户手册将讲述怎样安装、配置和使用本软件，以及该软件在使用过程中应该注意的一些问题。

本说明书的预期读者为：本邮箱系统的最终用户

### 1.2 背景

本项目全称为《Clean Mailbox》，由张森、严灏共同提出设计，并由张森、严灏协同开发的一款基于朴素贝叶斯算法的垃圾邮件过滤系统。本软件需要在 windows XP 及以上系统，连接互联网的环境下运行，并可以完成在互联网环境下收发邮件、标记删除垃圾邮件等功能。

本软件需要通过邮箱账户获取来自电子邮件服务商的账户数据，并通过对服务商发送请求完成电子邮件的相关操作。

### 1.3 定义

- **系统：**若未特别指明，统指本邮箱软件；
- **QT Creator：**Qt Creator 是跨平台的 Qt IDE，Qt Creator 是 Qt 被 Nokia 收购后推出的一款新的轻量级集成开发环境（IDE）。此 IDE 能够跨平台运行，支持的系统包括 Linux（32 位及 64 位）、Mac OS X 以及 Windows；
- **Anaconda：**一个开源的 Python 发行版本，包含了 conda、Python 等 180 多个科学包及其依赖项；
- **分类器：**在已有数据的基础上学习的一个分类函数或构造出的一个分类模型；
- **SQA：**软件质量保证（SQA-Software Quality Assurance）是建立一套有计划，有系统的方法，来向管理层保证拟定出的标准、步骤、实践和方法能够正确地被所有项目所采用。

### 1.4 参考资料

- a. 《项目开发计划 2.0》见数据仓库 <https://github.com/airzs/EmailBox/>
- b. 《软件需求规格说明书 2.0》见数据仓库 <https://github.com/airzs/EmailBox/>

- c. 《软件设计说明书》见数据仓库 <https://github.com/airzs/EmailBox/>
- d. 《软件测试计划》见数据仓库 <https://github.com/airzs/EmailBox/>
- e. 《软件工程——实践者的研究方法》 Roger S. Pressman, Bruce R. Maxim 编著，机械工业出版社，2015
- f. 《数据库系统概念》 Abraham Silberschatz, Henry F. Korth, S. Sudarshan 编著，机械工业出版社，2012
- g. 《统计学习方法》 李航编著，清华大学出版社，2012
- h. 《机器学习》 周志华编著，清华大学出版社，2016

## 2 用途

### 2.1 功能

功能介绍：

1. 登录：用户输入正确的账号的独立密码可以登录到相应的邮箱账号，获取账户信息；
2. 保存账户账号密码：用户在登录时通过点击保存按钮可以保存账号的账号与密码；
3. 写信：用户通过自己的账号向其他用户发送邮件；
4. 草稿箱：用户写的邮件存为草稿；
5. 发件箱：用户写的邮件发送成功的邮件自动存到发件箱
6. 查看邮件：用户可以查看自己账户的各类邮件信息；
7. 删除邮件：用户可以删除自己账户中的邮件；
8. 垃圾邮件过滤：系统对接收到的邮件进行分类
9. 退出登录：在主界面退出登录并保存所有用户信息；

### 2.2 性能

#### 2.2.1 精度

- 输入数据精度的要求：输入数据为文本，无精度要求
- 输出数据精度的要求：小数点后保留 2 位有效数字
- 传输过程中的精度：小数点后保留 2 位有效数字

### 2.2.2 时间特性

- 响应时间：不超过 3 秒；
- 更新处理时间：软件界面更新响应为即时刷新；
- 数据的转换和传送时间：数据传输处理为即时处理
- 解题时间：分类器训练时间不超过 1mins，分类器分类时间不超过 3 秒；

### 2.2.3 灵活性

- 操作方式上的变化：该系统适用于 Windows XP 及以上的操作系统
- 运行环境的变化：在 Windows PC 上通用
- 精度和有效时限的变化：无
- 计划的变化或改进：软件升级后不影响用户原有数据

## 2.3 安全保密

暂未涉及。

## 3 运行环境

### 3.1 硬设备

- Windows PC 一台；
- 酷睿 i3 以上的处理芯片一枚及以上；
- 1GB 显存及以上的兼容显卡；
- 500M 的存储空间及以上
- Internet 互联网接入
- 联机 VGA 彩显一台

### 3.2 支持软件

- 操作系统：Windows XP 及以上操作系统

- 编程开发软件：Visual Studio 2015，QT Creator 5.7.0，Anaconda 3.4.0
- 数据库管理系统：Qt 自带数据库 QSQLite3
- 支持软件：Python 3（Jieba）

### 3.3 数据结构

本软件在运行过程中用到了 SQLite3 数据库，数据表定义如下所示

```
user(id, name, pswd, last);
email(id, sender, receiver, time, title, body, variety, valid);
draft(id, sender, receiver, title, body);
```

数据表结构如下所示：

用户信息数据字典：

属性名	存储代码	类型	备注
标识符	Id	char	用户编号
邮箱账号	Name	char	用户账号
邮箱密码	Pswd	char	用户密码

邮件信息数据字典：

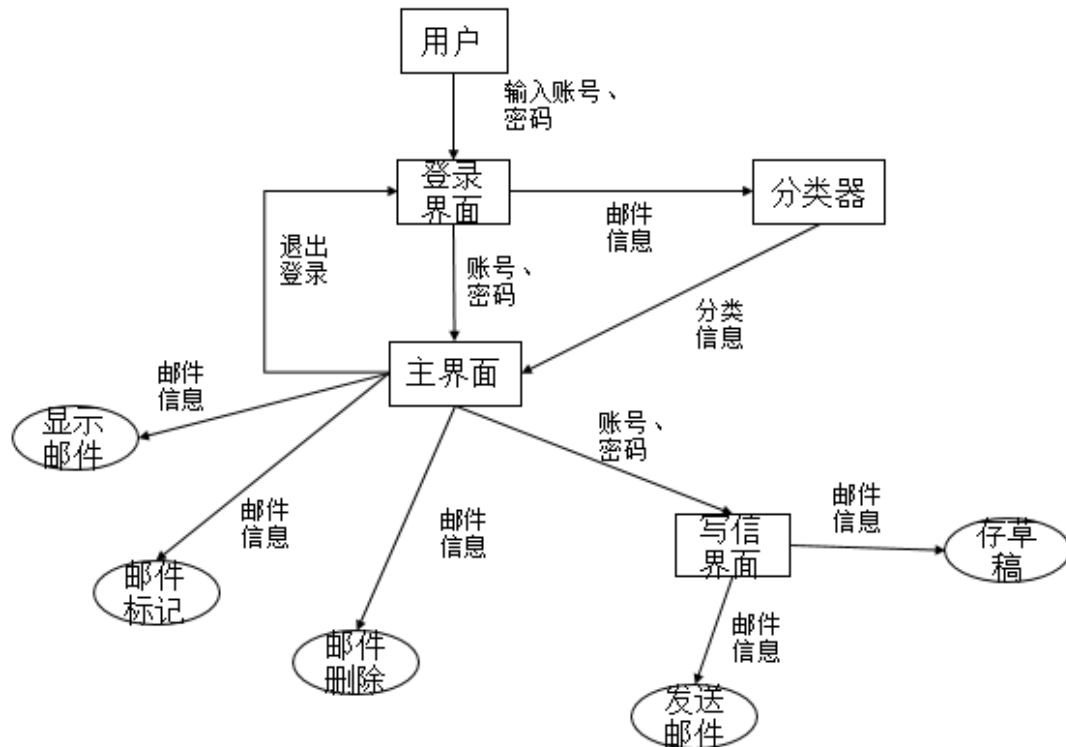
属性名	存储代码	类型	备注
标识符	ID	int	邮件唯一标识符
发送者	Sender	char	邮件发送者
接受者	Receiver	char	邮件接收者
日期	Time	char	发送时间
标题	Title	Char	邮件标题
正文	Body	Char	邮件正文
类别	Variety	Int	邮件类别
有效位	Valid	Int	邮件有效位

草稿信息数据字典：

属性名	存储代码	类型	备注
标识符	ID	int	草稿唯一标识符
发送者	Sender	char	邮件发送者
接受者	Receiver	char	邮件接收者
标题	Title	Char	邮件标题
正文	Body	Char	邮件正文

## 4 使用过程 \*\*\*

使用过程如下图所示：



### 4.1 安装与初始化

1.在浏览器中开启邮箱的 POP 协议，设置客户端授权密码。

在网页上打开 163 邮箱，进入主界面，依次点击设置->POP3/SMTP/IMAP



接着点击设置 POP 协议并保存：



再点击客户端授权密码，开启并设置独立授权密码



## 2. 安装软件

下载解压安装包 Clean MailBox，进入文件夹点击 Clean\_MailBox.exe 即可打开本软件。用户账户信息存储于数据库 SQLite3 中，即文件 database.db 中，安装过程中无需命令对其进行操作（若该文件丢失账户信息会丢失，但系统不会受影响）。若点击 Clean\_MailBox.exe 弹出登录界面，输入正确的 163 账户及独立密码后能正确登入主界面，并在几秒后提示分类成功，则安装成功，程序可以正确运行。

## 4.2 输入

本软件输入主要来自用户在登录和写信时的键盘输入以及用户对在操作过程中对用户界面的点击事件。

### 4.2.1 输入数据的现实背景

输入数据的现实背景主要是

- 情况——用户登录和写信时；
- 情况出现的频度——频度与用户操作习惯有关；

- c. 情况来源——软件的最终用户；
- d. 输入媒体——键盘；
- e. 限制——用户账户密码必须正确，否则；
- f. 质量管理——输入数据出错时软件会提示用户输入有问题；
- g. 支配——若用户登录成功，账户数据将会被分配给数据库和主界面，以使用户使用和查询，写信完成同样会将邮件分配至数据库。

## 4.2.2 输入格式

登录时的账号输入：

	类型	长度	格式基准	词汇表
账号	Char	不能为空	以左面的边沿为基准	数字、字母、@、.
密码	Char	不能为空	以左面的边沿为基准	数字、字母、字符

写信时的输入：

	类型	长度	格式基准	词汇表
账号	Char	不能为空	以左面的边沿为基准	数字、字母、@、.
主题	Char	无具体限制	以左面的边沿为基准	数字、字母、字符等
正文	Char	无具体限制	以左面的边沿为基准	数字、字母、字符等

## 4.2.3 输入举例

输入举例如下：

- a. 账号：13890458375@163.com（非空）  
密码：\*\*\*\*\*（非空）
- b. 账号：1846182474@qq.com（非空）  
主题：邮件发送测试（可省略）  
正文：这是一封正式邮件，请注意查收（可省略）

## 4.3 输出

### 4.3.1 输出数据的现实背景

说明输出数据的现实背景，主要是：

- a. 使用——给用户，用以提示用户或输出一些用户想要的信息；



- b. 使用频度——没有具体的规则，与用户的使用习惯有关；
- c. 媒体——显示屏显示；

## 4.3.2 输出格式

邮件信息的输出：（列表显示）

	类型	长度	格式基准	词汇表
发送者	Char	不能为空	以左面的边沿为基准	数字、字母、 @、.
主题	Char	无具体限制	以左面的边沿为基准	数字、字母、字 符等
日期	Char	无具体限制	以左面的边沿为基准	数字、字母、字 符等

草稿信息的输出：（列表显示）

	类型	长度	格式基准	词汇表
接受者	Char	不能为空	以左面的边沿为基准	数字、字母、 @、.
主题	Char	无具体限制	以左面的边沿为基准	数字、字母、字 符等
日期	Char	无具体限制	以左面的边沿为基准	数字、字母

提示信息的输出：（弹框输出）

	类型	长度	格式基准
提示信息	Char	不能为空	以左面的边沿为基准

## 4.3.3 输出举例

为每种输出类型提供例子。对例子中的每一项，说明：

- a. 定义——每项输出信息的意义和用途；
- b. 来源——是从特定的输入中抽出、从数据库文卷中取出、或从软件的计算过程中得到；
- c. 特性——输出的值域、计量单位、在什么情况下可缺省等。

输入举例如下：

	类型	定义	主体	来源	特性
邮件信息输出	发送者	邮件发送者 账号	189456222@qq.com	从邮件服务器中提取解析获得	字符串类型，不能为空
	主题	邮件的主题	邮件发送测试	从邮件服务器中提取解析获得	字符串类型，邮件主题无内容时

					可以为空
	日期	邮件发送的日期	这是一封正式邮件， 请注意查收	从邮件服务器中提取解析获得	字符串类型，邮件正文无内容时可以为空
草稿信息输出	接收者	邮件接收者账号	1846182474@qq.com	数据库中获取	字符串类型，不能为空
	主题	邮件的主题	草稿箱测试	数据库中获取	字符串类型，草稿主题无内容时可以为空
	日期	邮件发送的日期	这是一封草稿邮件， 请注意是否正确保存	数据库中获取	字符串类型，草稿正文无内容时可以为空
提示信息输出	提示	系统提示信息	“提示：发送成功！”	系统直接打印	事先已经写入代码

## 4.4 文卷查询

本软件具有对邮件信息等的查询能力，对数据库的查询主要是对各类邮件信息的查询以及对用户信息的查询，数据库表的定义以及具体结构在本文 3.3 数据结构一节已经给出，此处不再赘述。

本软件中对数据库相关的初始化主要是关系模式的建立，具体的用户信息都来自用户的输入和从邮件服务器上获取的内容，关系模式的创建过程如下所示：

```
QSqlQuery query;
query.exec("create table user (
    \"id      varchar(40) not null, \"
    \"name    varchar(20), \"
    \"pswd    varchar(40) not null, \"
    \"last    int      not null,\"
    \"primary key(id)\"");
query.exec("create table email (\"
    \"id      integer      primary key,\"
    \"sender   varchar(40)  not null, \"
    \"receiver varchar(40)  not null, \"
    \"time     varchar(40)  not null, \"
    \"title    varchar(200) not null, \"
    \"body     varchar(1000) not null, \"
    \"variety  int          not null, \"//0-norm;1-spam;2-work;3-friend;4-un
    \"valid    int          not null)");
query.exec("create table draft (\"
    \"id      integer      primary key,\"
    \"sender   varchar(40)  not null, \"
    \"receiver varchar(40)  not null, \"
    \"title    varchar(200) not null, \"
    \"body     varchar(1000) not null)");
return true;
```

用户的查询都是通过图形界面上的点击完成，系统再根据点击事件执行相关查询操作。查询方式相对单一，查询时只能查询特定的邮件信息，查询的能力也有限，但在本软件的应用环境下是更为简单的方式，而且足够满足查询要求。查询事件中具体的命令对用户来说都

是不可见的，因此没有针对用户的控制规定

## 4.5 出错处理和恢复

为了确保本软件能够进行分类操作，用户需要保证 `data` 文件的完整性，若 `data` 文件夹中部分文件丢失，软件中邮件解析分类模块可能会失效，但其他模块能正常工作。出现这种情况，用户需要重新拷贝一份完整的 `data` 文件夹到该目录下。

为了确保软件能够长久地存储用户信息，用户需要保证 `databas.db` 文件的完整性，若该文件丢失，用户会丢失所有以前的操作数据，但程序仍能正常运行。若 `database.db` 文件丢失无法找回则程序无法恢复用户信息。

因为程序在传输协议上的问题，长期操作可能会出现邮件信息的异常，此时可以手动改变程序信息或删除 `database.db` 重置用户信息，此问题将在本软件以后的更新版本中得到解决。

若软件还有其他问题导致程序不能正常使用，请尝试重新登录或重置用户信息，甚至重新覆盖程序，如出现此情况也可以联系开发团队 [1846182474@qq.com](mailto:1846182474@qq.com)。

## 4.6 终端操作

本软件是单机程序，不涉及多终端问题。