

文章编号:2096 - 5389(2022)04 - 0096 - 04

基于 UI Automation 的桌面版微信自动化应用

杨恩波^{1,2}, 张 祝^{1,2}, 付瑞滢¹, 邹海东³

(1. 贵州省铜仁市气象局, 贵州 铜仁 554300; 2. 贵州省铜仁市气象防灾减灾中心, 贵州 铜仁 554300; 3. 贵州省松桃苗族自治县气象局, 贵州 松桃 554100)

摘 要:近年来即时通信软件的应用有效提高了政府部门间协作效率,同时在扩大公共气象服务覆盖面上发挥了重要作用。以微信为例,省、市、县三级气象部门通过加入其他职能部门工作交流群,可及时将服务信息发送到相关部门,对常规服务手段形成了有利补充。该文介绍了基于 UI Automation 的桌面版微信自动化应用,实现了相关功能,有效减轻气象部门业务人员的工作量,提升工作效率,同时降低漏发及误发风险。

关键词:微信;自动化;Python;气象服务

中图分类号:TP311.50 **文献标识码:**B

Application of Automated PC WeChat Based on UI Automation

YANG Enbo^{1,2}, ZHANG Zhu^{1,2}, FU Ruiying¹, ZOU Haidong³

(1. Tongren Meteorological Bureau of Guizhou Province, Tongren 554300, China;
2. Center for Meteorological Disaster Prevention and Mitigation of Tongren, Tongren 554300, China;
3. Songtao Meteorological Bureau of Guizhou Province, Songtao 554100, China)

Abstract: In recent years, the application of IM tools improves the cooperation efficiency between government apartments apparently, and plays a very import role in improving the coverage in public meteorological service. For example, after joining in different government apartments' WeChat groups, meteorological service information can be sent in a very efficient way, which is also a supplement of traditional ways. In this paper, we introduced an application of automated PC WeChat based on UI Automation, which can lower the workload of weather forecasters, improve their work efficiency and lower the relative risks at the same time.

Key words: WeChat; automation; Python; meteorological service

0 引言

近年来新媒体技术的应用有效提高了政府部门间协作效率,同时在扩大公共气象服务覆盖面上发挥了重要作用。以微信为例,省、市、县三级气象部门通过加入其他职能部门工作交流群,可及时将服务信息发送到相关部门,对常规服务手段形成了有利补充^[1];通过关注微信公众号,公众亦可及时有效地获取气象部门的权威预报及服务信息。贵

州省气象服务中心在 2019 年投入运行的微信机器人已实现将预警信号靶向发布至受影响地区的目标群,而常规预报、决策气象服务材料等内容的发送仍依赖于各级气象部门值班服务人员手动操作,面对服务对象较多的现状,这一做法费时费力,且存在漏发的可能,如何提高服务效率是急需解决的问题。

目前针对微信公众号的信息推送技术拥有微信官方的技术支持,但针对普通微信信号的自动化技

收稿日期:2021 - 08 - 25

第一作者简介:杨恩波(1990—),男,硕士,工程师,主要从事气象数据分析、软件开发等工作,E-mail:enbooyang1990@outlook.com。

资助项目:贵州省气象局科研业务项目(黔气科登[2020]03-21号);桌面版微信自动化技术在公共气象服务中的应用;铜仁市气象局科研业务项目(铜气科合[2019]04号);县级业务及应急值班值守显示软件开发。

术受限制较多。有开发者利用网页版微信开放协议^[2]开发了相关微信自动化工具,但该方法不具备普适性,部分微信账号特别是新申请的账号无法使用网页版微信开放协议;部分开发者使用其他相关通信协议实现了微信自动化功能,但使用方法未开源,且未获得官方认可,存在一定违规风险。

本文介绍基于 UI Automation 技术^[3-4](以下简称 UIA)的桌面版微信软件自动化的应用,实现对桌面版微信程序的控制,从而达到自动转发文字类、文件类(含图片)等信息的目的,减轻服务人员的工作压力并提升工作效率,同时以 12379 网站为例介绍预警信息的获取及入库等关键步骤执行过程,以 FTP 方式为例介绍文件类信息的获取及入库过程。

1 关键技术介绍

本文所介绍的关键技术为通过 UIA 实现对桌面版微信程序的控制,从而达到使其自动执行指定操作的目的。UIA 是由微软公司开发的界面自动化测试框架,为技术人员提供通过编程控制 Windows 操作系统绝大多数程序界面元素的方法,从而实现模拟相关操作。该框架主要用于桌面程序编写阶

段对界面稳定性进行测试,支持 Win32 应用程序、.NET Windows 窗体应用程序和 WPF 应用程序。

为减小开发难度,利用 Python 语言开发,UIA 采用封装后的 UI Automation 模块^[3],界面使用 PyQt5 构建,使用 SQLite 轻型数据库实现任务记录及日志记录等功能,数据库读写使用 Python 自带的 sqlite3 库。使用 Request 库实现 12379 网站预警信息获取^[5],使用 FTPlib 模块实现文件类信息的获取^[6];消息发送过程中为提高执行速度借助 Pyperclip 模块实现对剪贴板的控制^[7]。

2 应用设计介绍

2.1 主要运行流程

如图 1 所示,应用的主要运行流程为主进程依据时钟定时连接 SQLite 数据库,检索是否存在待发消息,如果有则依据消息类型分别通过 UIA 操控桌面版微信执行不同的信息发送操作,并及时在数据库中将该发送完的消息标记为已执行。SQLite 数据库中的消息记录由信息采集/入库程序负责,该功能可集成于主程序中,通过多线程的方式开启,也可使用独立程序执行。

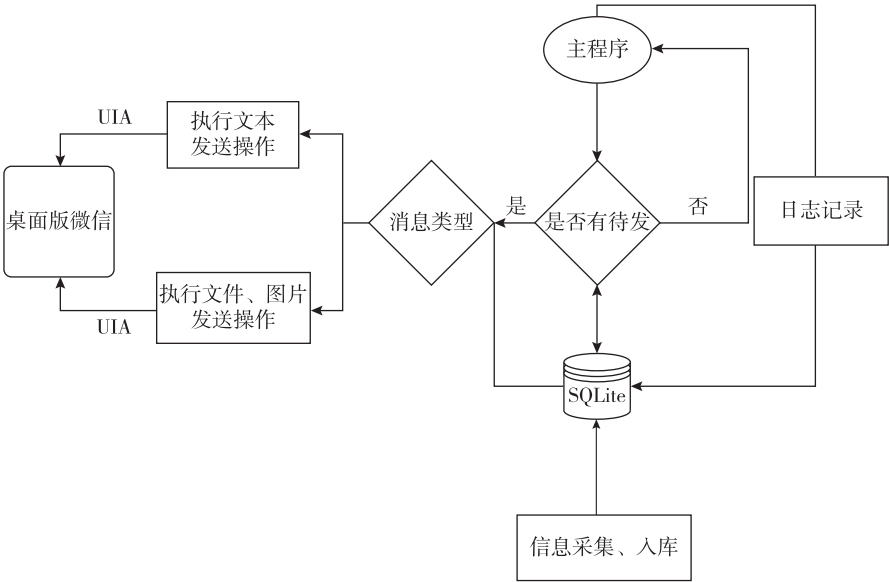


图 1 应用运行流程

Fig. 1 The logic diagram of the app

2.2 微信界面元素控制

桌面版微信界面中涉及自动控制的元素主要包含 5 个:主界面、搜索框、消息输入框、打开文件按钮、发送按钮。使用 UI Automation 对桌面版微信

软件进行解析,经梳理后关键元素相关数据如表 1 所示。获得相关元素数据后可通过指定程序根据控件的深度及控件名称来查找控件的位置,并进行操作。

表 1 微信软件关键元素解析表
Tab.1 The Analyzing result of WeChat's key elements

元素	类型	解析后名称	检索深度	快捷键
主界面	WindowControl	WeChatMainWindForPC	1	-
消息输入框	TextControl		4	Ctrl + C/V
搜索	EditControl	搜索	7	Ctrl + F
发送文件	ButtonControl	发送文件	14	-
发送按钮	ButtonControl	发送(S)	14	Enter

若要完成查找“铜仁气象微信工作群”，并进入消息编辑界面，方法如下：

```
importuiautomationasauto
wechatWindow=auto.WindowControl(searchDepth=1,ClassName='WeChatMainWndForPC')
TargetName='铜仁气象微信工作群'
wechatWindow.SendKeys('{Ctrl}f',waitTime=1.0)
search.SendKeys(TargetName,waitTime=1.0)
choice=wechatWindow.TextControl(searchDepth=12,Name='<em>'+TargetName+'</em>')
ifchoice.Exists(maxSearchSeconds=3):
    choice.Click(simulateMove=False)
    returnTrue
else:
    wechatWindow.SendKeys('{Esc}')
    returnFalse
```

若要在消息输入框中输入“天气预报”4 个字并发送，方法如下：

```
importuiautomationasauto
wechatWindow=auto.WindowControl(searchDepth=1,ClassName='WeChatMainWndForPC')
msg="天气预报"
edit=wechatWindow.TextControl(searchDepth=
```

4) # 查找消息输入框

```
edit.SendKeys(msg)
edit.SendKeys('{Enter}')
```

若要打开指定位置的雨量图并发送，方法如下：

```
importuiautomationasauto
wechatWindow=auto.WindowControl(searchDepth=1,ClassName='WeChatMainWndForPC')
file_path='D:\\rain.png'
sendfilebtn=wechatWindow.ButtonControl(searchDepth=14,Name='发送文件')
filepathinput=wechatWindow.EditControl(searchDepth=4,Name='文件名(N):')
fileopenbtn=wechatWindow.SplitButtonControl(Name='打开(O)')
```

```
sendfilebtn.Click(simulateMove=False)
filepathinput.SendKeys('%s'%file_path)
fileopenbtn.Click(simulateMove=False,waitTime=1.0)
edit.SendKeys('{Enter}')
```

在实际操作过程中可通过 Pyperclip 模块将消息内容存储在剪贴板中，同时利用发送快捷键 (Ctrl + C/V) 的方式提升执行速度及效率。

2.3 SQLite 数据库表设计

数据库包含 2 张表，表名分别为 Schedule (用于存储消息记录) 和 Log (用于存储日志)，表结构如表 2、表 3 所示：

表 2 Schedule 表结构

Tab.2 The structure of Schedule Table

字段名	数据类型	用途
UUID	Text	主键
logtime	Text	插入时间，用于存在多条待发消息时按顺序发送
content	Text	消息内容，对文件类消息存储其路径
target	Text	存储发送目标 csv 文件名
mark	int	标记段，0 代表未发送，1 代表已发送
type	int	标记消息类型，1 代表文本类，2 代表文档类，3 代表图片类

表 3 Log 表结构

Tab.3 The structure of Log Table

字段名	数据类型	用途
UUID	Text	主键
content	Text	日志内容
logtime	Text	日志插入时间

2.4 12379 网站预警信息读取

目前，各级气象台在国家突发事件预警信息发布系统中发布预警信号后，信息将逐级汇总到国家级服务器，同时在 12379 网站 (www.12379.cn) 上呈现，通过解析 12379 网站的 alarm_list_all.html 文件，将数据返回为 json 格式，然后逐一检索每一条预警信号，判断是否有指定地点气象台新发布的预警信号，方法如下：

```
alarmsite='http://www.12379.cn/data/alarm_
```

```
list_all.html'
alarmzone = '铜仁市气象台'
def readalarm():
    alarm = msgpara()['alarmmsg']
    alarmjson = requests.get(alarmsite)
    alarmjson.encoding = 'utf-8'
    alarmjson = json.loads(alarmjson.text)['
alertData'] # 旧的预警信号
    i = 0
    while i < len(alarmjson):
        if alarmzone in alarmjson[i]['descrip-
tion']:
            alarm = (alarmjson[i]['description'])
            break
        else:
            alarm = msgpara()['alarmmsg']
            i = i + 1
    return alarm
```

取得需要入库的预警信号文本内容后将其写入 SQLite 数据库 Schedule 表中即可,写入时需指定好需要发布的对象,即表 2 中 target 字段。

2.5 FTP 方式获取文件类消息

在应用开发过程中,曾尝试局域网共享及 FTP 服务器等 2 种方式获取文件类消息,二者均可实现设计目标。综合考虑网络安全及铜仁市气象台业务环境等因素后,采取了 FTP 服务器作为唯一途径。具体做法为:

①预报员将需要发布的文档或图片上传到 FTP 服务器指定目录;

②信息采集程序(主要使用 FTPLib 模块)定时扫描 FTP 服务器中指定目录,如果发现其中有文件,则将文件下载至本地,将本地路径(对应表 2 中 content 字段)等信息写入 SQLite 数据库 Schedule 表中,同时删除 FTP 服务器中的文件。

③主程序监测到文件类待发消息并执行完发送流程后,将本地文件删除。

3 应用成效

基于本文所介绍方法开发的微信机器人软件

已在铜仁市气象局正式投入运行。最新版本在完善文件类消息发送功能后于 2021 年 4 月下旬开始运行。截止目前共计向 30 余个党政工作交流群及公众气象服务群转发各类信息 505 条。其中文本类消息(含预警信号和 1060121 短信)209 条,文档类消息 160 条(主要是决策气象服务材料),图片类消息 136 条,在历次重大天气过程中第一时间为政府相关部门应对重大天气过程提供了服务信息,同时为部分公众提供日常气象预报作为出行的参考。

4 结束语

本文介绍了基于 UI Automation 技术的桌面版微信自动化方法,为解决相关自动化需求提供了新的开发思路。该方法可解决因微信账号权限无法使用网页版微信开放协议的难题,同时避免使用第三方微信自动化闭源协议,安全性较高。需要注意的是该方法无法实现消息发送的并发,即需要向目标群或好友逐一发送。尽管执行效率低于基于网页版 API 的相关工具,但相较于人工发送的方式已经有了很大提高,避免了漏发、误发等问题,同时对微信账号安全无影响。此外,该方法也可用于对值班人员的告警提醒,如在达到特定告警条件时,向值班人员拨打微信语音电话。

参考文献

- [1] 刘庆龙. 如何使公共气象服务效果更好[J]. 人民论坛, 2012(28):65.
- [2] YouFou. wxpy 使用文档[CP/OL]. (2017-02-21)[2017-06-28]. <https://wxpy.readthedocs.io/>.
- [3] Yin Kaisheng. UI Automation 使用文档[CP/OL]. (2016-11-15)[2021-12-17]. <https://github.com/yinkaisheng/Python-UIAutomation-for-Windows>.
- [4] Microsoft. Microsoft - UI - Automation[CP/OL]. (2019-08-24)[2021-09-15]. <https://docs.microsoft.com/en-us/dotnet/framework/ui-automation>.
- [5] Python Software Foundation. Requests 库文档[CP/OL]. (2021-07-13)[2022-01-03]. <https://requests.readthedocs.io>.
- [6] Python Software Foundation. ftplib 库文档[CP/OL]. (2019-10-14)[2020-10-05]. <https://docs.python.org/>.
- [7] Al Sweigart. Pyperclip 库文档[CP/OL]. (2013-12-29)[2021-02-21]. <https://pyperclip.readthedocs.io/>.