☎ 智能文件监控程序 + OBS自动化 - 增强版

Python 3.6+ License MIT OBS 28.0+

一个基于Python的智能文件监控系统,集成OBS WebSocket自动化功能,能够实时监控用户发言文件,智能识 别特定模式并自动切换OBS场景。

淼 核心特性

□ 文件监控

• 🖸 实时监控: 0.5秒检测间隔, 实时响应文件变化

• **窗 智能解析**: 自动提取用户发言内容, 去除时间戳和标识符

• **② 多格式支持**: 兼容多种日志格式和编码

• 🖸 自动切换: 文件删除时自动切换到最新文件

■ OBS自动化

• WebSocket集成: 通过WebSocket协议控制OBS Studio

• ② 智能识别: 检测"看X"模式, X为场景编号

• 🔯 时间控制: 自动切换保持120秒, 之后返回默认场景

• ② 冷却保护:切换期间拒绝新的切换请求 • // 场景管理: 自动获取和映射所有OBS场景

② 智能功能

• [3] 模式识别: 检测用户发言中"看"字和数字的组合

• 📊 状态显示: 实时显示连接状态、场景信息和切换结果

• ⑩ 自动配置: 首次运行自动生成配置文件

• ② **超时选择**: 启动时提供默认选项, 无需等待用户输入

❷ 快速开始

1. 🖉 环境准备

系统要求

- Python 3.6 或更高版本
- OBS Studio 28.0 或更高版本
- Windows/Linux/macOS

安装依赖

自动安装所有依赖 python install_dependencies.py

```
# 或手动安装
pip install watchdog obsws-python
```

2. Mar OBS Studio 配置

- 1. 启动 OBS Studio
- 2. 点击 工具 → WebSocket 服务器设置
- 3. ✓ 勾选 启用 WebSocket 服务器
- 4. 设置服务器端口 (默认: 4455)
- 5. 设置认证密码 (可选)
- 6. 点击 确定 保存

♀ 提示: 首次连接时程序会自动扫描并配置所有场景

3. ② 程序配置

程序会自动创建 obs_config.json 配置文件:

```
{
    "obs_connection": {
        "host": "192.168.10.110",
        "port": 4455,
        "password": "123456"
    },
    "scene_settings": {
        "default_scene": "默认-0",
        "switch_duration": 120
    }
}
```

4. Ø 启动程序

推荐方式 (自动化启动)

```
python start.py
```

直接启动

```
python fileMonitor.py
```


程序启动后会自动提示, 无需等待用户输入:

- **OBS功能**: 10秒后默认启用 (选择 y)
- 监控模式: 3秒后默认选择实际模式 (选择 1)

刻 说明: 超时功能确保程序能够自动化启动,无需手动干预

打包和分发

₽ 自动化打包

本项目提供了完整的自动化打包方案,可以将Python源码打包成独立的Windows可执行文件(.exe)。

```
# 一键打包
python build_exe.py
```

□ 发布包内容

打包完成后,会生成包含以下文件的发布包:

- 冒智能文件监控程序.exe 主程序 (约7MB)
- ⑩ obs_config.json OBS配置文件
- FREADME.md 详细说明文档
- 📝 使用说明.txt 快速使用指南

♀ 使用方法

- 1. 下载发布包并解压
- 2. 双击运行智能文件监控程序.exe
- 3. 无需安装Python环境, 独立运行

⑥ 监控参数配置

☑ 监控频率设置

- 内容变化检测: 0.5秒间隔 (实时响应)
- 新文件检查: 每10分钟自动检查
- 自动文件切换: 发现更新文件时立即切换

♀ 输出优化

- **图标标识**: 使用emoji图标区分不同类型的消息
- 时间戳: 精确到秒的时间显示
- 分层信息: 清晰的信息层级结构
- 颜色区分: 不同类型事件使用不同的视觉标识

Ⅲ 输出界面说明

❷ 事件类型标识

- 🔳 [时间] 普通信息
- ☑ [时间] 成功操作
- 🗙 [时间] 错误信息
- [时间] 文件内容变化
- 🖸 [时间] 文件切换

[信息层级

```
    □ 14:30:15] 文件内容变化
    □ 文件: 用户发言记录_2025_01_03.txt
    ☑ 原始内容: 2025-08-29 22:53:09[用户发言]t: 看123号房间
    ※ 用户发言: 看123号房间
    □ 检测结果: 发现"看"字和数字 -> 123
    ☑ [14:30:15] 场景已切换到编号 123
```

❷ OBS自动场景切换功能

❷ 核心功能

- 自动连接: WebSocket自动连接OBS Studio
- 场景映射: 自动获取所有场景并生成编号映射
- 智能切换:根据用户发言中的数字自动切换对应场景
- 时间控制: 120秒后自动返回默认场景
- 冷却机制: 在切换期间内拒绝新的切换请求

然 配置文件结构

程序会自动生成 obs_config.json 配置文件:

```
{
    "obs_connection": {
        "host": "192.168.10.110",
        "port": 4455,
        "password": "123456",
        "connect timeout": 5,
        "reconnect_interval": 10
    "scene_settings": {
        "default_scene": "默认-0",
        "switch duration": 120,
        "scenes": {
            "1": {
                "name": "录制",
                "number": 1,
                "enabled": true,
                "description": "场景1: 录制"
```

```
}
}
}

monitoring": {
    "enabled": true,
    "auto_switch": true,
    "debug_mode": false
}
```

② 使用流程

1. 用户发言: 看123号房间

2. 内容提取: 提取纯净内容 看123号房间

3. 数字检测: 检测到看字和数字123

4. 场景匹配: 查找编号123对应的场景

5. **自动切换**: OBS切换到目标场景

6. 定时返回: 120秒后自动返回默认场景

17 时间控制机制

• 切换保持时间: 120秒 (可配置)

冷却期防止:在120秒内无法再次触发切换自动返回:超时后自动返回到默认场景

13 智能数字检测功能

🕝 功能说明

程序能够自动检测用户发言中是否同时包含"看"字和数字:

• 满足条件: 内容包含"看"字 且 包含数字

• 返回结果: 提取第一个出现的数字 (支持整数和小数)

• 无匹配:返回None并显示未检测到的提示

② 支持的数字格式

• 整数: 0, 1, 123, 456

• 小数: 7.5, 12.34, 100.0

多数字:返回第一个数字

Ⅲ 检测示例

☑ 成功检测的例子

输入: "看123号房间" -> 输出: 123 输入: "看房间456" -> 输出: 456 输入: "我想看7.5个小时" -> 输出: 7.5

输入: "看房间88和99" -> 输出: 88 (第一个数字)

输入: "看0号" -> 输出: 0 输入: "看第1集" -> 输出: 1

X 无法检测的例子

输入: "看看这个" -> 无数字 输入: "房间123很好" -> 无"看"字 输入: "想看看电影" -> 无数字 输入: "看" -> 无数字 输入: "看" -> 无数字

龄 智能内容提取功能

Q 支持的格式

程序能够自动识别并提取以下格式的用户发言内容:

1. 完整时间戳格式:

2025-08-29 22:53:09[用户发言]t: 有黄水吗

□ 提取结果:有黄水吗

2. 简化标签格式:

3. 时间用户名格式:

14:30:15 李四: 大家好 ☐ 提取结果: 大家好

智 智能处理

- 自动去除: 时间戳、用户标识、用户名等前缀信息
- 空内容检测: 自动检测并标识空内容
- 原始内容保留: 无法识别的格式保留原始内容
- 多格式支持: 兼容多种日志格式

今 性能优化

□ 监控效率

- 快速响应: 0.5秒内容检测间隔确保实时性
- 智能检查: 10分钟新文件检查间隔平衡性能和及时性
- 事件防抖: 避免重复触发机制

圖 资源消耗

• 低CPU占用: 高效的文件监控算法

• 最小内存: 仅加载必要的文件内容

• 自动清理: 智能的资源管理

高级功能

回 自动文件切换

- 监控文件被删除时自动切换到最新文件
- 定期检查是否有更新的用户发言记录文件
- 智能文件优先级排序

1 自定义配置

- 可修改监控频率参数
- 可自定义OBS场景映射
- 可配置默认场景和切换时长

// 故障排除

常见问题

◎ OBS连接失败

- × 连接失败! 请检查:
- 1. OBS Studio 是否已启动
- 2. WebSocket 服务器是否已启用
- 3. 连接信息是否正确 (host, port, password)

文件监控无响应

- △ 文件监控可能的问题:
- 1. 文件路径是否正确
- 2. 文件是否有写入权限
- 3. 文件编码是否为UTF-8

◎ 场景切换失败

- × 场景切换失败可能原因:
- 1. 场景编号不存在
- 2. OBS连接已断开
- 3. 正在冷却期内

夏 更新日志

v2.0.0 (最新版)

- 添加超时默认选择功能
- 才 清理测试和调试文件
- 罗斯项目文档结构
- 【 优化程序启动流程

v1.9.0

- ✓ 修复"看看10的啊"检测问题
- 增强错误处理机制

v1.8.0

- 集成OBS WebSocket自动化
- ✓ 实现120秒自动返回机制
- 🗸 添加场景冷却保护

| 许可证

本项目采用 MIT 许可证。详情请参阅 LICENSE 文件。

♡ 贡献

欢迎提交 Issue 和 Pull Request!

& 技术支持

如果您在使用过程中遇到问题,请:

- 1. 查看本README文档
- 2. 运行 python install_dependencies.py 确保依赖完整
- 3. 使用项目内置的调试工具
- 4. 提交详细的Issue报告