

social-auto-uploadPublic

Watch60Fork1.4kStar8k

main3 Branches4 Tags

Go to fileAdd fileCode

dreammis1. 抖音我这边遇到必须上传封面的情况，自动设置第一个推荐封面（大概半分钟）ab2dbb0 · last month185 Commits

db	fix: 将 sau_backend 目录的文件，再挪回根目录，以防止大...	8 months ago
examples	feat: 增加 xiaohongshu 的登录和发布相关脚本	8 months ago
media	fix: 修改 readme 文件，更换 VX 二维码	7 months ago
myUtils	Merge pull request #156 from gzxy0102/fix-docker	2 months ago
sau_backend	fix: 将 flask 脚本挪到根目录	8 months ago
sau_frontend	feat(account-management): make abnormal status clicka...	2 months ago
uploader	1. 抖音我这边遇到必须上传封面的情况，自动设置第一个推...	last month
utils	feat: 增加 xiaohongshu 的登录和发布相关脚本	8 months ago
videos	feat: 增加 xiaohongshu 的登录和发布相关脚本	8 months ago
.dockerignore	feat: 增加Docker容器环境	2 months ago
.gitignore	fix: 去掉 conf.py 文件，改为 example 示例	8 months ago
CLAUDE.md	docs: add project documentation in CLAUDE.md	2 months ago
Dockerfile	fix: 修复Docker中打包的前端请求地址为本地	2 months ago
README.md	feat: 增加Docker容器环境	2 months ago
__init__.py	fix: 将 sau_backend 目录的文件，再挪回根目录，以防止大...	8 months ago
cli_main.py	fix: 将 sau_backend 目录的文件，再挪回根目录，以防止大...	8 months ago
conf.example.py	1. 抖音我这边遇到必须上传封面的情况，自动设置第一个推...	last month
requirements.txt	fix: 更新 requirements 依赖包	8 months ago
sau_backend.py	Merge pull request #156 from gzxy0102/fix-docker	2 months ago
start-win.bat	chore(windows): add Windows batch script for one-click ...	2 months ago

README

social-auto-upload

social-auto-upload 是一个强大的自动化工具，旨在帮助内容创作者和运营者高效地将**视频内容一键发布到多个国内外主流社交媒体平台**。项目实现了对 抖音、Bilibili、小红书、快手、视频号、百家号 以及 Tiktok 等平台的视频上传、定时发布等功能。结合各平台 uploader 模块，您可以轻松配置和扩展支持的平台，并通过示例脚本快速上手。



目录

- 🔗 功能特性
- 📱 支持的平台
- 📖 安装指南
- 🚀 快速开始
- 📄 项目背景
- 📖 详细文档
- 🗣️ 交流与支持
- 👉 贡献指南
- 📄 许可证
- 🌟 Star History

🔗 功能特性

已支持平台

- 国内平台:
 - 抖音
 - 视频号
 - Bilibili
 - 小红书
 - 快手
 - 百家号
- 国外平台:
 - TikTok

核心功能

- 定时上传 (Cron Job / Scheduled Upload)
- Cookie 管理 (部分实现, 持续优化中)
- 国外平台 Proxy 设置 (部分实现)

计划支持与开发中

- 平台扩展:
 - YouTube
- 功能增强:
 - 更易用的版本 (GUI / CLI 交互优化)
 - API 封装
 - Docker 部署
 - 自动化上传 (更智能的调度策略)
 - 多线程/异步上传优化
 - Slack/消息推送通知

2025.10.30目前现状

该项目本人很长一段时间没维护了，有比较大的问题也是能简单快速修复就修复掉

因为我自己也在创业，每天时间都用不完

目前问题主要集中在

- 小红书部分，这部分是直接适用xhs这个库来实现的
- web 端（vue版本），这个版本是群友LeeDebug他帮忙做的（再次感谢他）

因为我日常也在用，我用的不是web端，而是最初 uploader 文件夹里的版本，也就是文档里提到的部分<https://sap-doc.nasdadaddy.com/> 所以这里一般遇到的问题，我都会尝试去解决，一并推送到该仓库

目前能遇到的问题，基本上都比较小，可能是元素变化导致的 在初期设计的时候，其实我已经参考了某些不可变元素去选择，极大的避免了后期因为平台页面修改导致的元素变化

该项目不仅仅只是技术人员，有不少是非技术的从业人员，他们是没能力修复一个简单弱小的bug 为了能帮助更多的人，所以呼吁技术小伙伴

如果大家

- 修复了一些bug
- 增加一些对大家有帮助的功能

请积极的提出pr，我会想尽可能的确认后合并的，在此感谢大家对于开源项目的支持，帮助更多的人

我自己也会尽100%的力量，在自己项目稳定后，修bug，加更多的平台，开发出gradin版本（更易部署），大家谅解

📱支持的平台

本项目通过各平台对应的 uploader 模块实现视频上传功能。您可以在 examples 目录下找到各个平台的使用示例脚本。

每个示例脚本展示了如何配置和调用相应的 uploader。

📖安装指南

- 克隆项目:

```
git clone https://github.com/dreammis/social-auto-upload.git
cd social-auto-upload
```
- 安装依赖: 建议在虚拟环境中安装依赖。

```
conda create -n social-auto-upload python=3.10
conda activate social-auto-upload
# 挂载清华镜像 or 命令行代理
pip install -r requirements.txt
```
- 安装 Playwright 浏览器驱动:

```
playwright install chromium firefox
```

根据您的需求，至少需要安装 chromium, firefox 主要用于 TikTok 上传（旧版）。
- 修改配置文件: 复制 conf.example.py 并重命名为 conf.py。在 conf.py 中，您需要配置以下内容：
 - LOCAL_CHROME_PATH: 本地 Chrome 浏览器的路径。比如 c:\Program Files\Google\Chrome\Application\chrome.exe 保存。

临时解决方案

需要在根目录创建 cookiesFile 和 videoFile 两个文件夹，分别是 存储cookie文件和 存储上传文件的文件夹
- 配置数据库: 如果 db/database.db 文件不存在，您可以运行以下命令来初始化数据库：

```
cd db
python createTable.py
```

此命令将初始化 SQLite 数据库。
- 启动后端项目:

```
python sau_backend.py
```

后端项目将在 http://localhost:5409 启动。
- 启动前端项目:

```
cd sau_frontend
npm install
npm run dev
```

前端项目将在 http://localhost:5173 启动，在浏览器中打开此链接即可访问。

非程序员用户可以参考：[新手级教程](#)

🚀快速开始

- 准备 Cookie: 大多数平台需要登录后 Cookie 信息才能进行操作。请参阅 examples 目录下各 get_xxx_cookie.py 脚本（例如 get_douyin_cookie.py, get_kks_cookie.py）的说明，运行脚本以生成并保存 Cookie 文件（通常在 cookies/[PLATFORM]/uploader/account.json）。
- 准备视频文件: 将需要上传的视频文件（通常为 .mp4 格式）放置在 videos 目录下。部分平台支持视频封面，可以将封面图片（例如 .png 格式，与视频同名）也放在此目录。如果需要上传标题及标签，请在视频文件旁创建一个同名的 .txt 文件，内容为标题和标签，以执行分隔。
- 修改并运行示例脚本: 打开 examples 目录中您想使用的平台的上传脚本（例如 upload_video_to_douyin.py）。
 - 根据脚本内的注释和说明，确认 Cookie 文件路径、视频文件路径等配置是否正确。
 - 您可以修改脚本以适应您的具体要求，例如批量上传、自定义标题、标签等。
- 执行上传: 运行修改后的示例脚本，例如：

```
python examples/upload_video_to_douyin.py
```

Docker 环境

自己构建镜像

- 构建Docker镜像:

```
docker build -t social-auto-upload:latest .
```
- 运行Docker容器:

```
docker run -d -it -p 5409:5409 social-auto-upload:latest
```

使用预构建镜像

- 拉取镜像:

```
docker pull gzxy/social-auto-upload:latest
```
- 运行Docker容器:

```
docker run -d -it -p 5409:5409 gzxy/social-auto-upload:latest
```

启动容器后访问: <http://localhost:5409>

📄项目背景

该项目最初是我个人用于自动化管理社交媒体视频发布的工具。我的主要发布策略是提前一天设置定时发布，因此项目中很多定时发布相关的逻辑是基于“第二天”的时间进行计算的。

如果您需要立即发布或其他定制化的发布策略，欢迎研究源码或在社区提问。

📖详细文档

更详细的文档和说明，请查看：[social-auto-upload 官方文档](#)

🗣️交流与支持

👉 Donate as u like - 如果您觉得这个项目对您有帮助，可以考虑赞助。

如果您也是独立开发者、技术爱好者，对 #技术变现 #AI创业 #跨境电商 #自动化工具 #视频创作 等话题感兴趣，欢迎加入社群交流。

Creator



微信公众账号

关注公众号，后台回复“上传”获取加群方式



交流群 (通过公众账号)

如果您觉得项目有用，可以考虑打赏支持一下

Active Core Team



Telegram

封装了 api 接口和 web 前端管理界面 (请注明来意: 进群、学习、企业咨询等)

👉贡献指南

欢迎各种形式的贡献，包括但不限于：

- 提交 Bug 报告和 Feature 请求。
- 改进代码、文档。
- 分享使用经验和教程。

如果您希望贡献代码，请遵循以下步骤：

- Fork 本仓库。
- 创建一个新的分支 (git checkout -b feature/YourFeature 或 bugfix/YourBugfix)。
- 提交您的更改 (git commit -m "Add some feature")。
- Push到您的分支 (git push origin feature/YourFeature)。
- 创建一个 Pull Request。

📄 许可证

本项目暂时采用 [MIT License](#) 开源许可证。

🌟Star-History

如果这个项目对您有帮助，请给一个 🌟 Star 以表示支持！

Star History



dreammis/social-auto-upload

© 2026 GitHub, Inc.

TermsPrivacySecurityStatusCommunityDocsContactManage cookiesDo not share my personal information