# Amazon 商品监控与邮件通知 MCP 服务器

一个功能强大的 Model Context Protocol (MCP) 服务器，用于自动化 Amazon 商品监控和分析。该系统能够定时获取 Amazon 平台上表现最佳的商品信息，通过多维度分析筛选最优商品，并生成详细的分析报告通过邮件发送给用户。

## 🎆 核心功能

### 🔍 智能商品搜索

* 支持关键词和类别搜索
* 自动抓取商品信息（价格、评分、评论数、图片等）
* 支持多页面批量搜索
* 对 Amazon 反爬虫机制的优化处理
* **🎯 Amazon 联盟支持**: 自动为所有商品链接添加联盟 ID (joweaipmclub-20)

### 📊 多维度商品分析

* **最佳评分分析**: 综合考虑评分和评论数量
* **最高折扣分析**: 基于价格分布的智能折扣识别
* **最佳销量分析**: 基于评论数量的销量排名
* 自动生成分析摘要和推荐理由

### 📧 专业邮件报告

* 精美的 HTML 邮件模板
* 支持 Markdown 格式报告生成
* 自动包含商品链接和详细信息
* 支持 Gmail 和其他 SMTP 服务器

### ⏰ 定时监控系统

* 灵活的监控频率设置（日、周、月）
* 自动化监控任务管理
* 完整的监控历史记录
* 支持多个监控任务并行运行

## 🚀 快速开始

### 前置要求

* Python 3.10+
* [uv](https://github.com/astral-sh/uv) 包管理器
* Chrome/Chromium 浏览器（用于网页抓取）

### 安装说明

1. **克隆项目**

git clone <repository-url>  
cd amazon-product-monitor-mcp

1. **创建虚拟环境**

uv venv  
source .venv/bin/activate # Linux/Mac  
# 或  
.venv\Scripts\activate # Windows

1. **安装依赖**

uv sync

1. **运行服务器**

# STDIO 模式（适用于 MCP 客户端集成）  
sh run.sh  
  
# 或直接运行  
python server.py

## 🛠️ 可用工具

### 核心搜索和分析工具

#### search\_amazon\_products

搜索 Amazon 商品并获取详细信息。

**参数**: - keyword: 搜索关键词 - category: 商品类别（All, Electronics, Books, Clothing, Home, Sports, Toys） - max\_pages: 最大搜索页数（1-5）

#### analyze\_products

分析商品数据，识别最佳商品。

**参数**: - products\_data: JSON 格式的商品数据

#### run\_complete\_analysis

一键运行完整的搜索和分析流程。

**参数**: - keyword: 搜索关键词 - category: 商品类别 - max\_pages: 最大搜索页数

### 邮件报告工具

#### send\_email\_report

发送 HTML 格式的分析报告。

**参数**: - analysis\_result: 分析结果 JSON 数据 - recipient\_email: 收件人邮箱 - sender\_email: 发送者邮箱 - sender\_password: 邮箱密码或应用密码 - keyword: 搜索关键词

#### generate\_markdown\_report

生成 Markdown 格式的分析报告。

### 监控管理工具

#### create\_product\_monitor

创建新的商品监控任务。

#### run\_product\_monitor

执行指定的监控任务。

#### list\_product\_monitors

列出所有监控任务。

#### get\_monitor\_history

获取监控历史记录。

#### remove\_product\_monitor

删除指定的监控任务。

## 📝 使用示例

### 基本商品搜索和分析

# 1. 搜索商品  
search\_result = search\_amazon\_products(  
 keyword="gaming laptop",  
 category="Electronics",  
 max\_pages=2  
)  
  
# 2. 分析结果  
analysis = analyze\_products(search\_result)  
  
# 3. 生成报告  
report = generate\_markdown\_report(analysis, "gaming laptop")

### 一键完整分析

# 运行完整的分析流程  
result = run\_complete\_analysis(  
 keyword="智能手机",  
 category="Electronics",  
 max\_pages=3  
)  
  
print(result['markdown\_report'])

### 发送邮件报告

# 发送分析报告  
send\_result = send\_email\_report(  
 analysis\_result=analysis,  
 recipient\_email="user@example.com",  
 sender\_email="your-gmail@gmail.com",  
 sender\_password="your-app-password",  
 keyword="gaming laptop"  
)

### 设置定时监控

# 创建监控任务  
monitor = create\_product\_monitor(  
 keyword="graphics card",  
 category="Electronics",  
 email="notify@example.com",  
 frequency="daily"  
)  
  
# 执行监控  
result = run\_product\_monitor(  
 monitor\_id=monitor['monitor\_id'],  
 sender\_email="your-gmail@gmail.com",  
 sender\_password="your-app-password"  
)

## ⚙️ 配置说明

### Amazon 联盟功能

🎯 **内置联盟支持**: 本系统已预配置 Amazon 联盟 ID joweaipmclub-20，所有生成的商品链接都会自动包含联盟标签。这意味着：

* **Markdown 报告**: 所有商品链接自动添加 ?tag=joweaipmclub-20
* **HTML 邮件报告**: 邮件中的商品链接自动包含联盟标签
* **API 返回数据**: 所有 product\_url 字段自动包含联盟 ID
* **追踪支持**: 通过报告链接产生的购买行为可被 Amazon 联盟系统追踪

### 环境变量

在 MCP 客户端中配置以下环境变量以启用邮件功能：

* SENDER\_EMAIL: 发送邮件的邮箱地址（推荐使用 Gmail）
* SENDER\_PASSWORD: 邮箱密码或应用专用密码

### Gmail 配置指南

1. **启用两步验证**: 在 Google 账户设置中启用两步验证
2. **生成应用密码**: 在 Google 账户安全设置中生成应用专用密码
3. **使用应用密码**: 在 SENDER\_PASSWORD 中使用生成的应用密码，而不是普通密码

### MCP 客户端集成

在 Claude Desktop 中配置本 MCP 服务器：

{  
 "mcpServers": {  
 "amazon-product-monitor": {  
 "command": "sh",  
 "args": ["/path/to/amazon-product-monitor-mcp/run.sh"],  
 "env": {  
 "SENDER\_EMAIL": "your-gmail@gmail.com",  
 "SENDER\_PASSWORD": "your-app-password"  
 }  
 }  
 }  
}

## 📄 数据存储

本系统使用 JSON 文件存储监控数据：

* product\_monitor\_data.json: 监控任务和历史记录
* 数据自动备份和恢复
* 支持数据导入导出

## 🔒 安全考虑

* **邮箱凭据安全**: 使用环境变量存储敏感信息
* **率限控制**: 对 Amazon 请求进行限速保护
* **错误处理**: 全面的异常处理和日志记录
* **数据验证**: 严格的输入数据验证

## 🐛 故障排除

### 常见问题

1. **Chrome WebDriver 问题**
   * 确保安装了 Chrome 浏览器
   * 如果 WebDriver 无法启动，系统会自动切换到 requests 模式
2. **邮件发送失败**
   * 检查 Gmail 应用密码是否正确
   * 确保启用了两步验证
   * 检查网络连接
3. **商品搜索空结果**
   * 尝试使用不同的关键词
   * 检查网络连接和 Amazon 访问权限
   * 调整搜索参数

### 日志记录

系统提供详细的日志记录，可以帮助诊断问题：

# 查看实时日志  
python server.py --log-level DEBUG

## 📚 技术架构

* **框架**: FastMCP (Model Context Protocol)
* **网页抓取**: Selenium WebDriver + BeautifulSoup
* **数据处理**: Python 标准库 + 自定义算法
* **邮件服务**: SMTP + HTML 模板
* **数据存储**: JSON 文件系统

## 🌐 支持的 Amazon 站点

目前主要支持 Amazon.com，未来计划支持更多国际站点。

## 🎯 Amazon 联盟功能总结

本系统已完全集成 Amazon 联盟功能，为您提供以下价值：

### ✅ 自动化联盟链接

* **全局支持**: 所有生成的商品链接自动包含您的联盟 ID joweaipmclub-20
* **多格式兼容**: Markdown 报告、HTML 邮件、API 数据统一支持
* **一键部署**: 无需额外配置，开箱即用

### 📊 收益追踪

* **完整覆盖**: 所有通过报告产生的购买行为都能被 Amazon 联盟系统识别
* **数据透明**: 报告中明确标识联盟链接，增强用户信任
* **收益优化**: 为您的联盟营销提供技术支持

### 🚀 业务价值

* **流量转化**: 将商品监控服务转化为收益渠道
* **用户体验**: 提供有价值的商品推荐和分析服务
* **可扩展性**: 支持大规模部署和多用户服务

## 🔮 未来计划

* ☐ 支持更多 Amazon 国际站点
* ☐ 增加价格历史跟踪功能
* ☐ 支持更多电商平台
* ☐ Web 管理界面
* ☐ 更高级的商品分析算法
* ☐ 自定义报告模板

## 🤝 贡献

欢迎提交 Issue 和 Pull Request！请遵循以下原则：

1. 保持代码简洁和可读
2. 添加必要的测试用例
3. 更新相关文档
4. 遵循现有的代码风格

## 📜 许可证

MIT License - 详情请参阅 <LICENSE> 文件。

## ℹ️ 免责声明

本工具仅供学习和研究使用。请遵守 Amazon 的使用条款和相关法律法规。使用者需要对其使用行为负责。

作者: MiniMax Agent  
日期: 2025-07-13