

AI 网络爬虫工具

一、共同利益列表

利益相关者	关注点
内容创作者	希望工具能精准分析受众、优化内容发布策略
平台运营人员	关注内容合规、活跃度、用户增长
数据分析师	需要稳定、准确的内容数据和用户行为数据
开发团队	关注系统的可维护性、可扩展性
项目负责人	项目是否按时、高质量交付
法律顾问	用户数据是否符合法规，例如《个人信息保护法》
用户（UP 主）	关心视频分析以及观众反馈

二、需求调研

2.1 项目背景：

随着社交媒体的蓬勃发展，自媒体创作者们对于其内容的表现和受众反馈产生了日益增长的关注。为了更好地满足创作者们的需求，我们进行了深入的调研，旨在开发一款全面的自媒体分析工具。该工具旨在通过数据统计和分析，以及危机公关提醒，为创作者提供直观的信息，帮助他们更好地了解和优化其创作内容。

在调研过程中，需要掌握自媒体领域的市场状况的用户需求，明确自媒体创作者对于数据分析工具的期望。对此，我们设计了一款针对 up 主的自媒体分析工具，该功能能够进行数据爬虫，数据存储，数据分析功能，同时我们设计了前端交互界面，数据可视化，后端处理方面，我们设计了 up 主和视频的分析，包括基本数据可视化，情感分析，用户画像等。

本项目旨在开发一款自媒体分析工具，帮助内容创作者快速了解内容的传播情况，优化发布策略。

2.2 调研过程及方法

方法	内容描述
问卷调查	向 20 位 B 站 UP 主发放调研问卷，了解其对数据分析的需求与关注点。
半结构化访谈	与 5 位粉丝数超过 10 万的创作者进行了深度访谈。
竞品分析	分析了市面上已有的工具（如新榜、清博、飞瓜等），总结其优劣。
网络资料收集	参考知乎、小红书等平台上关于自媒体运营的数据分析经验贴。

2.3 用户画像

属性	内容
身份	以 B 站 UP 主为主，关注账号运营和粉丝增长的个人创作者或小团队
创作内容类型	动画、游戏、数码、生活、美妆等
用户目标	提升视频数据表现、提高用户互动、避免舆情风险、获取内容优化建议等
技术水平	一般具有基础的数据阅读能力，但缺乏专业的数据分析背景

2.4 用户需求分析

2.4.1 功能性需求

编号	功能名称	功能描述
FR-01	视频数据采集	系统支持通过爬虫自动获取 UP 主视频的各项数据（播放量、点赞、投币、弹幕等）。
FR-02	用户画像分析	系统能识别用户群体特征，如观看时长、地区、互动行为等。
FR-03	情感倾向分析	对评论内容进行情感分类（正向、中立、负向），评估舆情态势。
FR-04	数据可视化展示	以图表方式展示视频表现趋势、热度波动、用户分布等信息。
FR-05	危机提醒功能	当系统检测到负面评论比例上升或舆情波动异常时，进行预警。

编号	功能名称	功能描述
FR-06	报告导出	支持生成视频表现分析报告，导出为 PDF/图片格式。
FR-07	内容优化建议	根据历史数据推荐发布时间、关键词优化建议等。

2.4.2 非功能性需求

类别	描述
性能	系统应能支持每日分析至少 100 条视频内容，并在 10 秒内完成展示。
安全性	用户登录与数据传输需加密，用户隐私数据不得泄露。
可用性	系统界面简洁、响应流畅，支持移动端和 PC 端访问。
可维护性	后端模块化设计，便于后期功能扩展。
兼容性	前端适配主流浏览器及不同分辨率屏幕。

三、初步产品要求文档

3.1 产品目标

本项目旨在开发一款面向 UP 主的自媒体数据分析工具。该工具通过爬虫、情绪分析、用户画像、AI 助手等功能，帮助内容创作者：实时掌握视频表现数据；识别评论中的负面情绪风险；了解用户画像，优化内容方向；提高账号影响力与运营效率。

用户类型如下：

用户类型	典型特征	主要需求
B 站 UP 主	拥有数千至数十万粉丝，稳定内容输出	了解数据变化趋势、粉丝行为分析、及时获得负面评论提醒
新手创作者	粉丝量少，缺乏经验	获取发布建议、优化运营策略、提升互动量

3.2 产品范围说明

所涉及的平台主要包括：B 站，微博，知乎等网络社交平台。

本产品所支持的功能主要分为以下六大模块：

模块	特点及功能
数据爬取	用于爬取网络数据信息

模块	特点及功能
计算指标	实时收集 up 主点赞、收藏、评论、推荐、粉丝等数据并进行统计
情感分析	分析评论情感随时间的变化趋势，帮助用户了解评论的整体情感动向
用户画像	提取关键特征，如使用的关键词、表情等，通过数据关联将不同特征关联起来，形成用户画像的完整信息
用户界面	设计用户友好的界面，包括图表、表格等元素，然后通过数据呈现将计算结果以直观的方式在界面上展示
系统管理	实现用户认证与权限管理和日志记录

四、用户故事列表

US-01. 作为一名 B 站 UP 主，我希望能够查看视频的播放、评论、点赞等数据，以便了解内容传播效果。

US-02. 作为一名 B 站 UP 主，我希望获取粉丝的画像信息（性别、地区、年龄），以便制作更精准的内容。

US-03. 作为一名 B 站 UP 主，我希望分析评论的情绪倾向，以便及时发现负面反馈。

US-04. 作为一名 B 站 UP 主，我希望当评论中负面情绪激增时收到通知，以防止舆情风险。

US-05. 作为一名 B 站 UP 主，我希望可以导出我的分析数据为 PDF 报告，方便团队内部汇报或保存历史记录。

US-06. 作为一名 UP 主，我希望能收到评论情绪异常的提醒，以便及时调整或修改视频风格和内容。

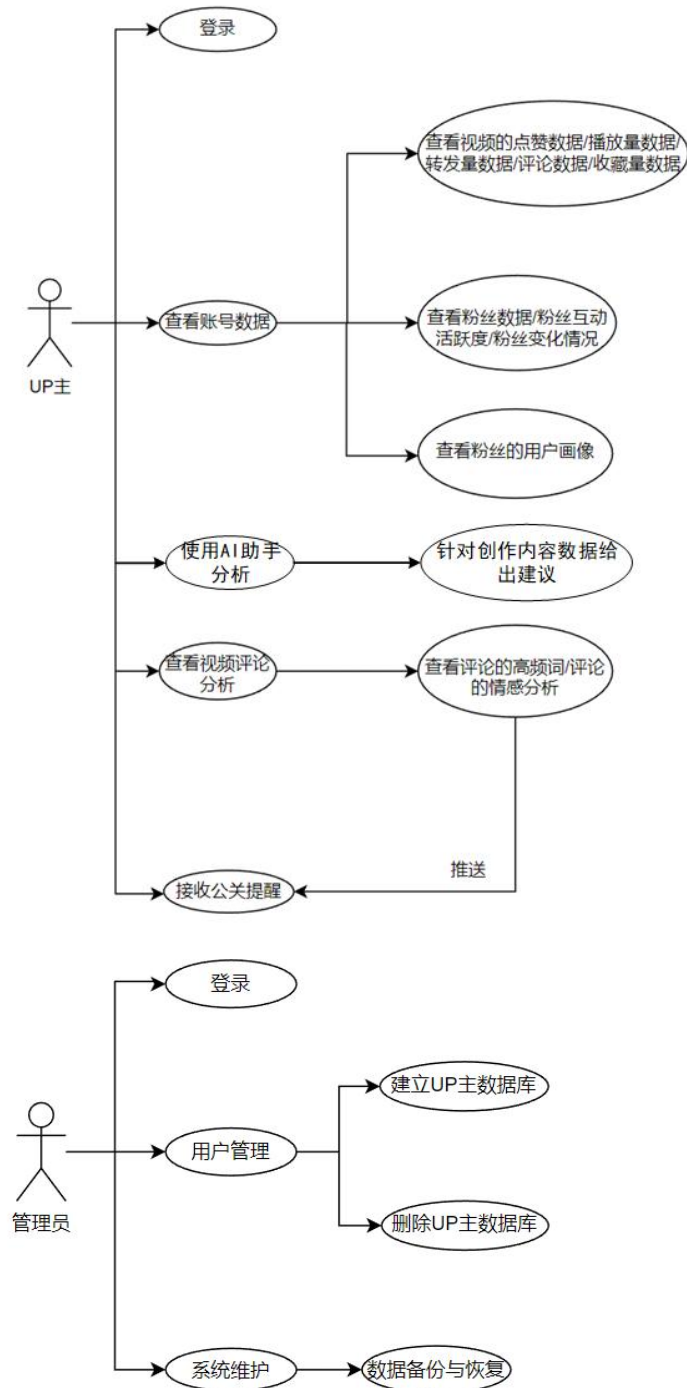
US-07. 作为一名 UP 主，我希望可以筛选某段时间内的视频数据，查看那段时间的创作效果。

注：按优先级从高至低排序。

五、需求分析模型

5.1 场景模型

5.1.1 系统用例图



5.1.2 用例模板

(1) 用例一：

用例名：登录。

用例描述：up主通过提供有效的身份信息进行登录。

参与者：up主。

前置条件：已经获得 up 主账号，并且具备登录系统的权限。

后置条件：获取信息更新进数据库。

基本流程：up 主点击登录，选择 up 主登录，输入账号信息和密码，系统验证提供的用户名和密码是否匹配。如果验证成功，系统进入主界面，可以开始使用各项功能；如果验证失败，系统显示相应的错误消息，需重新输入用户名和密码。

(2)用例二：

用例名：查看视频的点赞数据/播放量数据/转发量数据/评论数据/收藏量数据。

用例描述：up 主通过系统查看自己某段时间内视频数据变化情况。

参与者：up 主。

前置条件：up 主登录系统，系统爬取 up 主的视频数据并更新数据库。

后置条件：无。

基本流程：系统定时爬取 up 主视频数据并更新数据库，并可以按照指定时间范围进行筛选，比如选择查看过去七天的数据或者自定义时间段的数据，系统呈现选定时间范围内的视频数据变化情况，以图表或表格等形式清晰展示。

(3)用例三：

用例名：查看粉丝的用户画像。

用例描述：up 主通过系统查看自己的粉丝用户画像。

参与者：up 主。

前置条件：up 主登录系统，系统爬取 up 主粉丝数据并更新数据库，用户画像模型已定时更新，基于最新的粉丝数据生成用户画像。

后置条件：无。

基本流程：up 主登录系统，系统定时爬取 up 主粉丝数据并更新数据库，用户画像模型根据最新的粉丝数据生成粉丝的用户画像，up 主通过系统界面选择查看粉丝用户画像，包括但不限于年龄分布、地域分布、兴趣爱好等。up 主可以根据画像结果调整内容创作、互动策略等，以更好地满足粉丝的兴趣和需求。

(4)用例四：

用例名：查看评论的高频词/评论的情感分析

用例描述：up 主通过系统查看评论的高频词/评论的情感分析。

参与者：up 主。

前置条件：up 主登录系统，系统已定时爬取 Up 主视频的评论并更新数据库，针对最新的评论数据，系统已进行高频词提取和情感分析。

后置条件：无。

基本流程：up 主登录系统，系统定时爬取 up 主的视频评论并更新数据库，针对最新的评论数据，系统执行高频词提取和情感分析，up 主通过系统界面选择查看评论的高频词或情感分析。对于高频词，系统展示评论中出现频率最高的词汇，以词云、图表或表格等形式呈现。对于情感分析，系统呈现评论整体的情感倾向，包括积极、中性、消极等，并以可视化方式展示评论情感分布。

(5)用例五：

用例名：负面评论公关提醒

用例描述：当 up 主的视频负面评论过多时，提醒创作者进行公关处理。

参与者：up 主。

前置条件：up 主登录系统，系统已定时爬取 Up 主视频的评论并更新数据库，针对最新的评论数据，系统已进行高频词提取和情感分析。

后置条件：向 up 主推送提醒消息。

基本流程：up 主登录系统，系统定时爬取 up 主的视频评论并更新数据库，针对最新的评论数据，系统执行高频词提取和情感分析，系统检测是否存在负面评论，通过情感分析识别出情感倾向为消极的评论。如果负面评论数量超过预设的阈值，系统向 up 主推送提醒消息，提醒 up 主视频可能存在负面情感。

(6)用例六：

用例名：AI 数据分析

用例描述：up 主可以选择将自己目前创作内容数据上传给 AI 助手，获得 AI 助手对当前数据的分析和对创作内容改进的建议。

参与者：up 主

前置条件：up 主登录系统，系统已定时爬取 Up 主视频的评论并更新数据库，完成对创作内容数据分析。

后置条件：能够调用 AI 大模型接口，获得 AI 模型数据。

基本流程：up 主登录系统，系统定时爬取 up 主的视频评论并更新数据库，up 主通过系统页面选择 AI 助手建议，系统将 up 主创作内容数据上传给 AI 助手，AI 分析数据后返回。

(7)用例七：

用例名：管理员登录

用例描述：管理员登录系统。

参与者：管理员。

前置条件：已经获得系统管理员账号，并且具备登录系统的权限。

后置条件：无。

基本流程：管理员点击登录，选择管理员登录，输入账号信息和密码，系统验证提供的用户名和密码是否匹配。如果验证成功，系统进入主界面，可以开始使用管理员的各项功能；如果验证失败，系统显示相应的错误消息，需重新输入用户名和密码。

(8)用例八：

用例名：管理员对系统数据备份与恢复。

用例描述：管理员对自媒体分析工具进行数据备份与恢复。

参与者：管理员。

前置条件：已经登录系统并具备管理员权限。

后置条件：管理员成功完成数据备份与恢复，系统数据得到有效保护或成功还原。

基本流程：管理员登录系统，进入管理员控制台，管理员选择数据备份功能，选择备份的数据范围、存储位置等备份参数，系统执行数据备份，并生成备份文件。

(9)用例九：

用例名：管理员对 up 主建立数据库。

用例描述：管理员在自媒体分析工具中对 up 主建立数据库。

参与者：管理员。

前置条件：已经登录系统并具备管理员权限，并且获得 up 主授权。

后置条件：管理员成功完成对 up 主建立数据库，up 主数据被成功爬取并集成到数据库中。

基本流程：管理员登录系统，进入管理员控制台，管理员选择建立数据库功能，输入 up 主的基本信息和权限等，通过 B 站 api 爬取 up 主的数据，将数据存储在数据库中。

(10)用例十：

用例名：管理员删除 up 主数据库。

用例描述：管理员在自媒体分析工具中对 up 主数据库进行删除。

参与者：管理员。

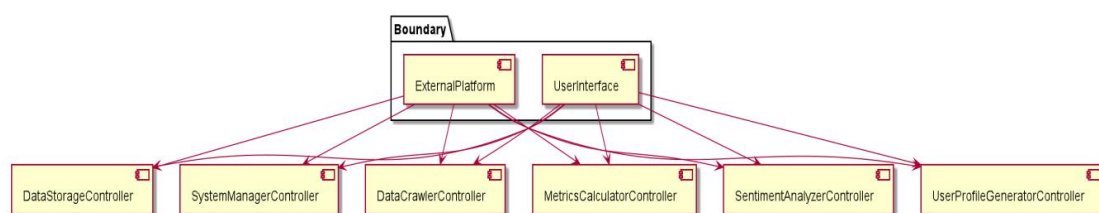
前置条件：已经登录系统并具备管理员权限。

后置条件：管理员成功完成删除 up 主数据库，目标 up 主数据被从系统中移除。

基本流程：管理员登录系统，进入管理员控制台，管理员选择删除数据库功能，输入 up 主的基本信息等，将目标 up 主数据从系统数据库中移除。

5.2 静态模型

5.2.1 边界类图



其中，UserInterface（用户界面）：表示系统与用户之间的交互界面，用户通过它使用自媒体平台分析工具。

ExternalPlatform（外部平台）：表示可能是其他系统或平台，通过它实现与自媒体平台分析工具的集成与交互。

DataCrawlerController（数据爬取控制器）：负责使用爬虫技术从网站获取视频发布者的相关信息。

MetricsCalculatorController（指标计算控制器）：负责计算并生成图表，如点赞、收藏、评论、推荐、粉丝随时间的变化图，展示账号数据的分析结果。

SentimentAnalyzerController（情感分析控制器）：利用情感分析工具或库对视频评论进行分析，判断评论的情感倾向，包括积极、消极或中性。

UserProfileGeneratorController（用户画像生成控制器）：负责提取用户关键特征，例如年龄、性别、地区、活跃度等，生成用户画像。

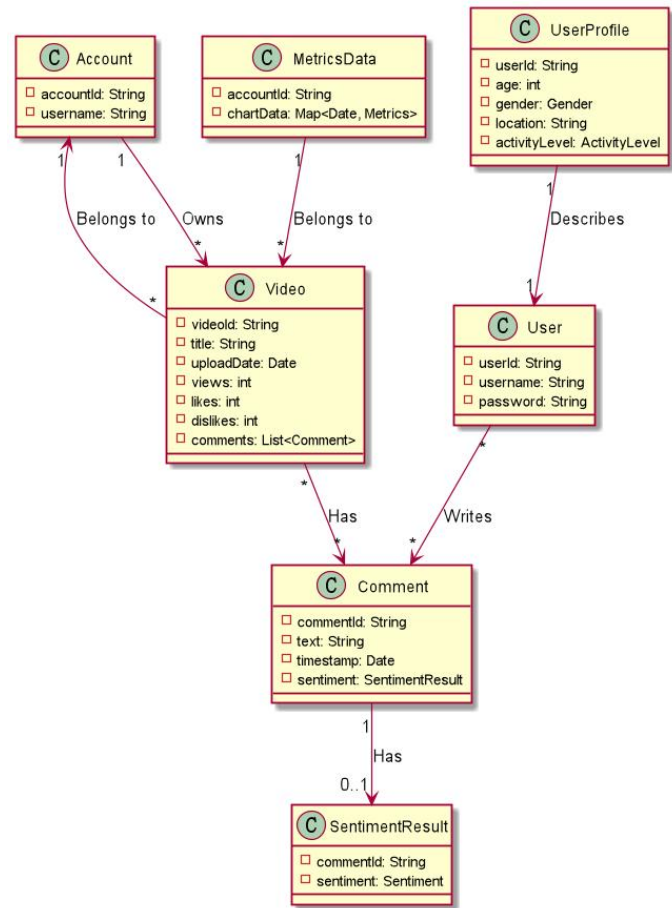
DataStorageController（数据存储控制器）：管理系统内数据的存储和检索，存储从网站爬取的信息以及计算生成的指标、情感分析结果和用户画像等数据。

SystemManagerController（系统管理控制器）：充当整个系统的中央控制器，负责协调各个控制器之间的交互和整体系统的管理。

关联线连接 UserInterface 和 ExternalPlatform 与各个控制器，表示用

户界面和外部平台通过这些控制器与系统交互，实现对数据爬取、指标计算、情感分析和用户画像等功能调用。

5.2.2 实体类图



在实体类图中，Account（账号）：表示自媒体平台上的用户账号，包含唯一的账号 ID（accountId）和用户名（username）。

Video（视频）：代表发布在自媒体平台上的视频，包含唯一的视频 ID（videoId）、标题（title）、上传日期（uploadDate）、观看次数（views）、点赞数（likes）、踩数（dislikes）和评论列表（comments）。

Comment（评论）：表示视频的评论，包含唯一的评论 ID（commentId）、评论文本（text）、时间戳（timestamp）和情感分析结果（sentiment）。

User（用户）：代表系统用户，包含唯一的用户 ID（userId）、用户名（username）和密码（password）。

MetricsData（指标数据）：包含特定账号的时间序列图数据，以 Map 形式存储日期和指标（Metrics）的关联。

SentimentResult（情感分析结果）：包含评论的情感分析结果，包括唯一的评论 ID（commentId）和情感（sentiment）。

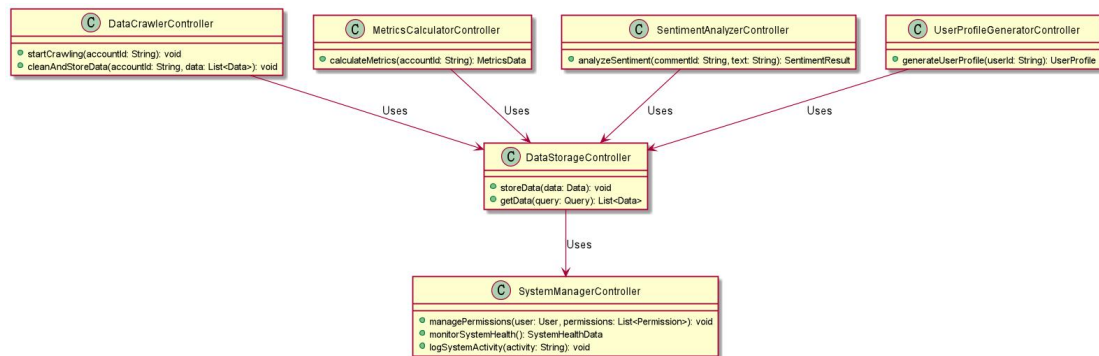
UserProfile（用户画像）：包含用户的关键特征，如年龄（age）、性别（gender）、地区（location）和活跃度（activityLevel）等，用于描述用户的整体特征。

Account 和 Video 之间的关系表示一个账号拥有多个视频（Owns），而一

个视频属于一个账号 (Belongs to)。Video 和 Comment 之间的关系表示一个视频可以有多个评论 (Has)，而一个评论属于一个视频。Comment 和 SentimentResult 之间的关系表示一个评论可以有零个或一个情感分析结果 (Has)。User 和 Comment 之间的关系表示一个用户可以写多个评论 (Writes)。MetricsData 和 Video 之间的关系表示一个 MetricsData 对象属于一个视频 (Belongs to)。UserProfile 和 User 之间的关系表示一个用户有一个用户画像，用于描述用户的特征 (Describes)。

本实体类图清晰地定义了系统中各个核心概念的属性和关系，为设计和实现自媒体平台分析工具提供了良好的基础。

5.2.3 控制类图



控制类图清晰地表示了自媒体平台分析工具中不同控制器之间的职责和关系，有助于系统的模块化设计和实现。其中 DataCrawlerController（数据爬取控制器）：负责通过爬虫技术爬取网站视频发布者的相关信息。

startCrawling(accountId: String): void: 启动爬取过程，传入账号 ID。

cleanAndStoreData(accountId: String, data: List<Data>): void: 清理并存储爬取得到的数据，传入账号 ID 和数据列表。

MetricsCalculatorController（指标计算控制器）：负责计算指标，生成数据分析图。

calculateMetrics(accountId: String): MetricsData: 根据账号 ID 计算指标，返回 MetricsData 对象。

SentimentAnalyzerController（情感分析控制器）：负责使用情感分析工具或库对每条评论进行情感分析。

analyzeSentiment(commentId: String, text: String): SentimentResult: 对评论进行情感分析，返回 SentimentResult 对象，传入评论 ID 和文本。

UserProfileGeneratorController（用户画像生成控制器）：负责提取用户关键特征，生成用户画像。

generateUserProfile(userId: String): UserProfile: 生成用户画像，返回 UserProfile 对象，传入用户 ID。

DataStorageController（数据存储控制器）：负责存储和检索系统内的各种数据。

storeData(data: Data): void: 存储数据，传入 Data 对象。

getData(query: Query): List<Data>: 根据查询条件获取数据列表，传

入 Query 对象。

SystemManagerController（系统管理控制器）：充当整个系统的中央控制器，负责系统级别的管理任务。

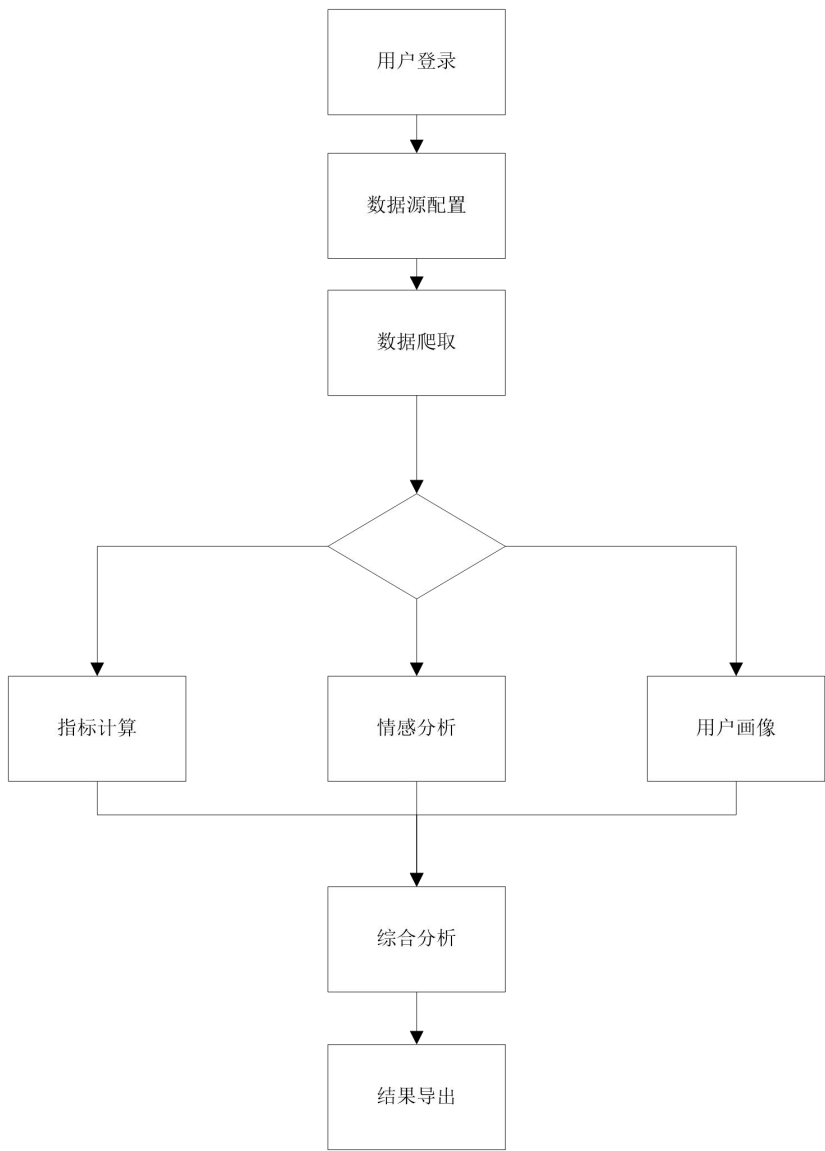
managePermissions(user: User, permissions: List<Permission>): void: 管理用户权限，传入用户和权限列表。

monitorSystemHealth(): SystemHealthData: 监控系统健康状况，返回 SystemHealthData 对象。

logSystemActivity(activity: String): void: 记录系统活动，传入活动描述。

图中，DataCrawlerController、MetricsCalculatorController、SentimentAnalyzerController、UserProfileGeneratorController 使用 DataStorageController 来存储和检索数据。DataStorageController 使用 SystemManagerController 来记录系统活动。

5.3 动态模型



系统流程图

用户通过输入用户名和密码到系统进行验证,验证通过后进入系统页面,系统提供给用户友好的配置界面,包括输入自媒体平台 URL、选择爬取规则等,用户可以选择添加新的数据源或编辑现有的数据源配置,并且系统提供数据源测试功能,确保配置正确。同时系统提供爬虫任务管理界面,用户可以查看数据爬取的进度和状态,并且允许用户中止、暂停或调整任务优先级。用户可以选择指标计算模块选择特定时间范围和指标进行分析,系统提供交互式图表,用户可以通过图表进行数据点的详细查看。用户可以选择情感分析功能对情感分析结果进行筛选,查看积极、消极或中性评论,并且提供词云等可视化工具,帮助用户更好地理解评论情感。用户可以通过用户画像模块根据不同特征维度筛选用户群体,提供用户画像的历史演变图,反映用户画像随时间的变化。同时用户可以通过交互式界面进行自定义分析,选择不同数据集和分析方法。提供深度学习模型的预测结果,帮助用户了解未来的趋势。

系统可以提供多样化的图表和图形展示,用户可以根据需要定制报表,导出分析结果为 PDF、Excel 等格式,以便与团队分享或进一步处理。

六、测试用例编写

6.1 单元测试

针对本次系统设计的每个类,分别测试类的属性和方法是否符合预期。

6.2 系统测试

本项目实现的功能主要有:数据爬取、数据库建立与更新、评论情感分析、建立用户画像、分析建立高频词词云、AI 助手。

数据爬取功能:爬取多个 B 站视频数据,包括播放率、评论、点赞收藏数等视频相关数据。

数据库建立与更新功能:在用户页面端测试数据库相关操作。

评论情感分析功能:对爬取到的随机 500 条评论数据进行评论情感分析,判断所设计的评论情感分析功能是否正常运行。

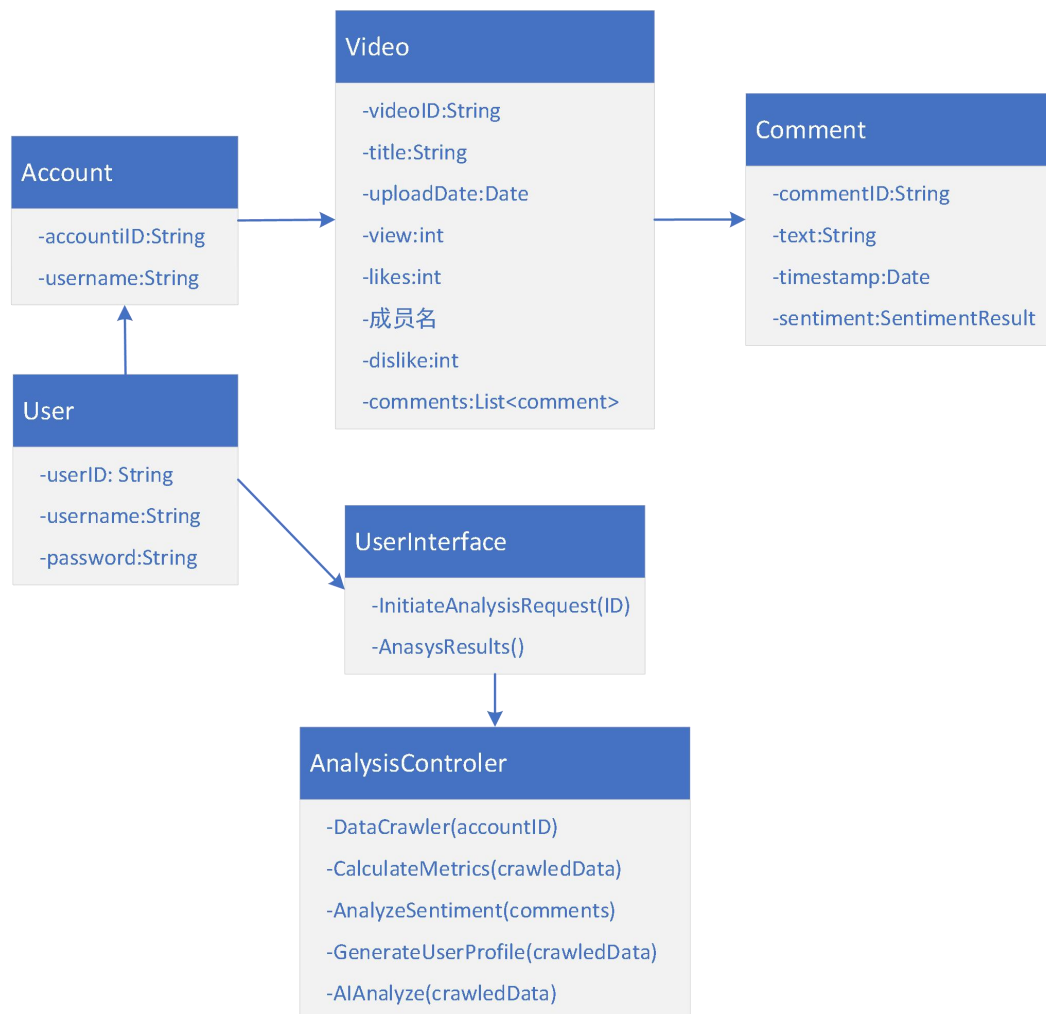
用户画像建立:随机选取 5 个创作者,并爬取他们的近 3 个月的作品,生成用户画像,观察用户画像结果。

分析建立高频词词云:针对随机选择的 5 个创作者的近 3 个月的作品,测试了高频词词云功能。

AI 助手:将某个创作者的数据输入到 AI 助手中,评估 AI 助手返回的数据分析结果和改进意见。

七、基于用例精华

7.1 精华后的类图



精华后的类图包括了 User（用户）类，Account（账户）类，Video（视频）类，Comment（评论）类，UserInterface（用户接口）类和 AnalysisControl（分析控制器）类。

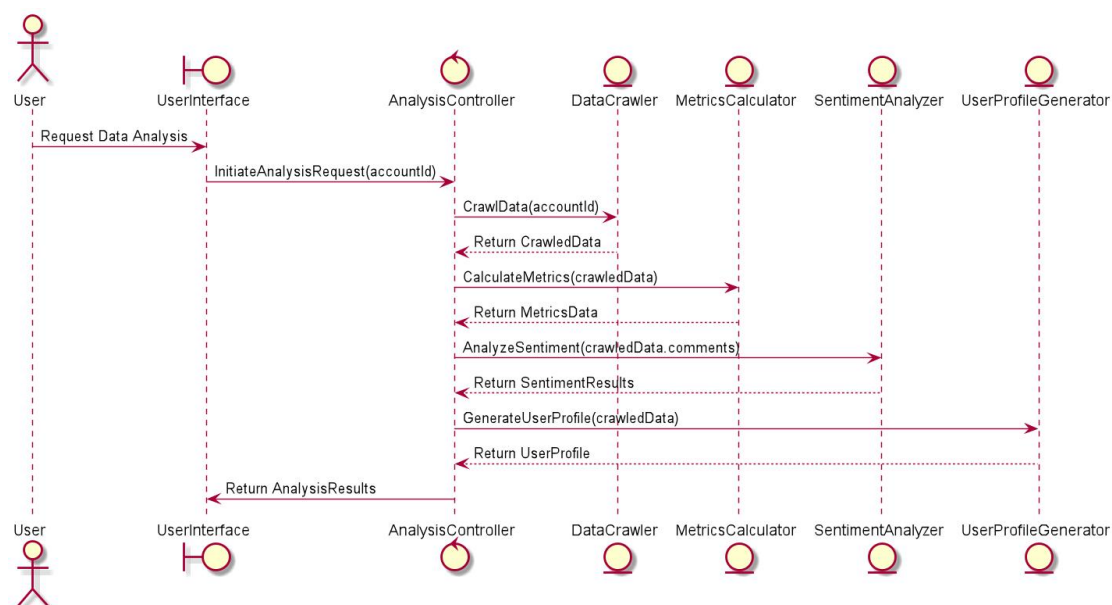
User(用户):代表系统用户,包含唯一的用户 ID(userID)、用户名(username)和密码(password)。

Account 和 Video 之间的关系表示一个账号拥有多个视频 (Owns), 而一个视频属于一个账号 (Belongs to)。Video 和 Comment 之间的关系表示一个视频可以有多个评论 (Has), 而一个评论属于一个视频。

UserInterface（用户接口）类作为边界控制器，将用户发起的请求传递给核心的 AnalysisController（分析控制器）类。

AnalysisController 负责协调各个核心模块的工作，并处理结果返回给用户。包括爬取网站视频相关信息；计算视频指标，产生数据分析图；对评论进行情感分析；根据爬取的信息提取用户的关键特征，生成用户画像；调用 AI 助手，对爬取的数据进行分析，给出优化建议。

7.2 精华中用到的顺序图



在这个顺序图中：User 是平台的用户，通过用户界面 UserInterface 发起数据分析请求。

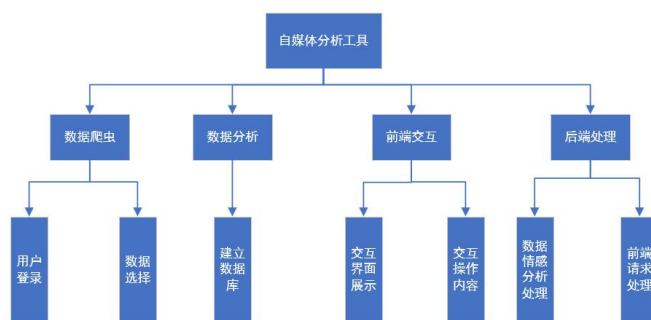
UserInterface 作为边界控制器，将请求传递给核心的 AnalysisController。AnalysisController 负责协调各个核心模块的工作，并处理结果返回给用户。DataCrawler 负责爬取网站视频发布者的相关信息。MetricsCalculator 负责计算指标，生成数据分析图。SentimentAnalyzer 利用情感分析工具对每条评论进行情感分析。UserProfileGenerator 根据爬取的信息提取用户的关键特征，生成用户画像。

八、体系结构设计

8.1 体系结构风格

我们的体系结构采用模块化和客户端-服务器风格体系结构，包含数据爬虫、数据分析、前端交互和后端处理四个功能模块。其中数据爬虫、数据分析、后端处理三个模块部署在服务器，前端交互模块部署在客户端。

8.2 系统整体结构图

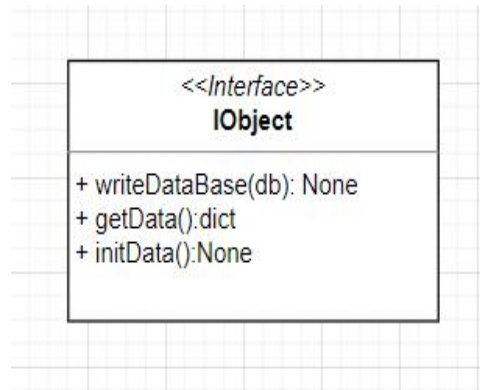


九、构件设计

9.1 接口设计

1. Interface IObject (接口名)

描述：接口，Up、Comment、Video 类应当实现此接口



方法：

(1) writeDataBase(db): None

描述：向数据库中写入自身的值，Up、Comment、Video 类应当实现此方法

返回：None

(2) getData():dict

描述：将自身的所有数据导出为字典类型

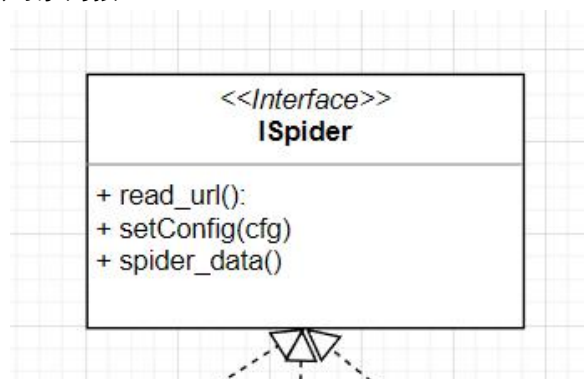
返回：数据字典(Key:Value)

(3) initData():None

描述：初始化自身数据，通过爬虫等方式，获取数据

返回：None

2. 各个爬虫对象的接口



成员函数

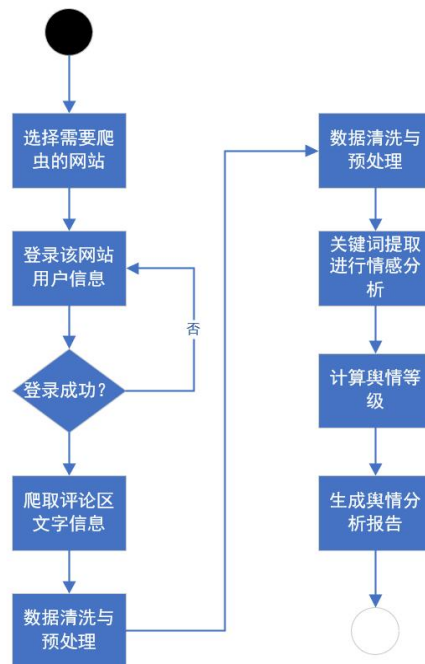
(1) + read_url(): 使用读取一个 url，返回读取后的 response

(2) + setConfig(cfg) 设置爬虫的配置，如 headers、cookies 等

(3) + spider_data() 爬取并保存数据

9.2 内部算法设计(活动图)

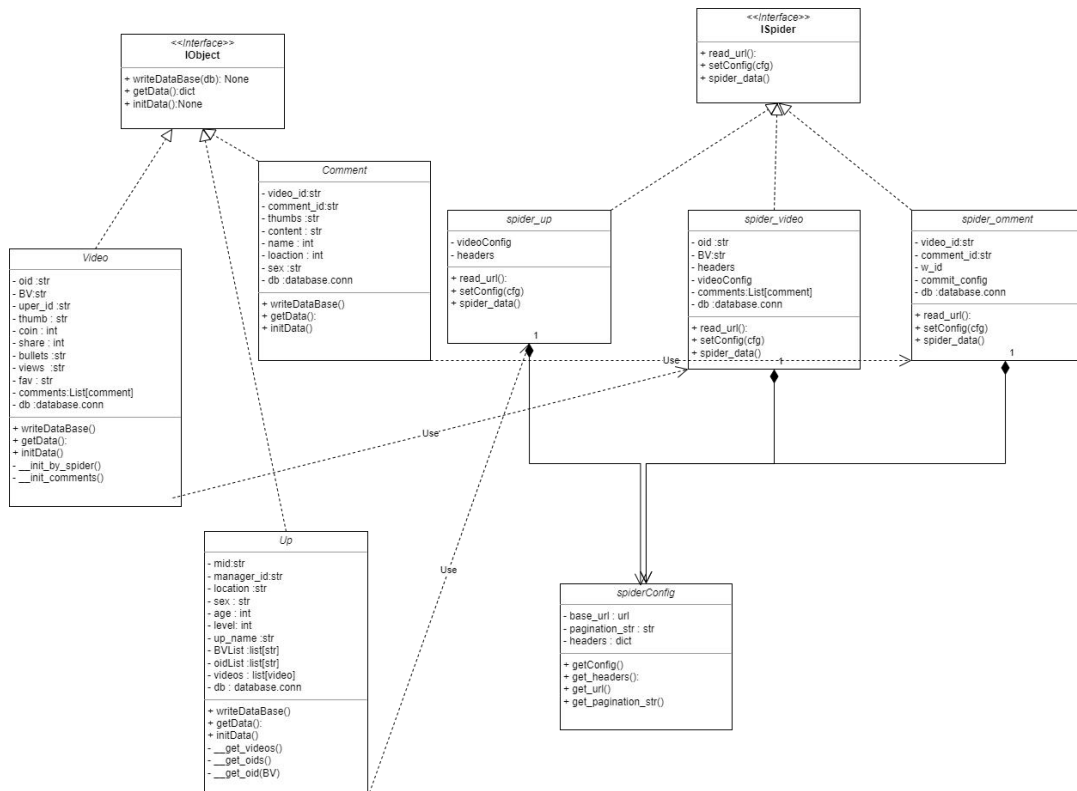
舆情分析（活动图）：



9.3 类图

基本信息：类图

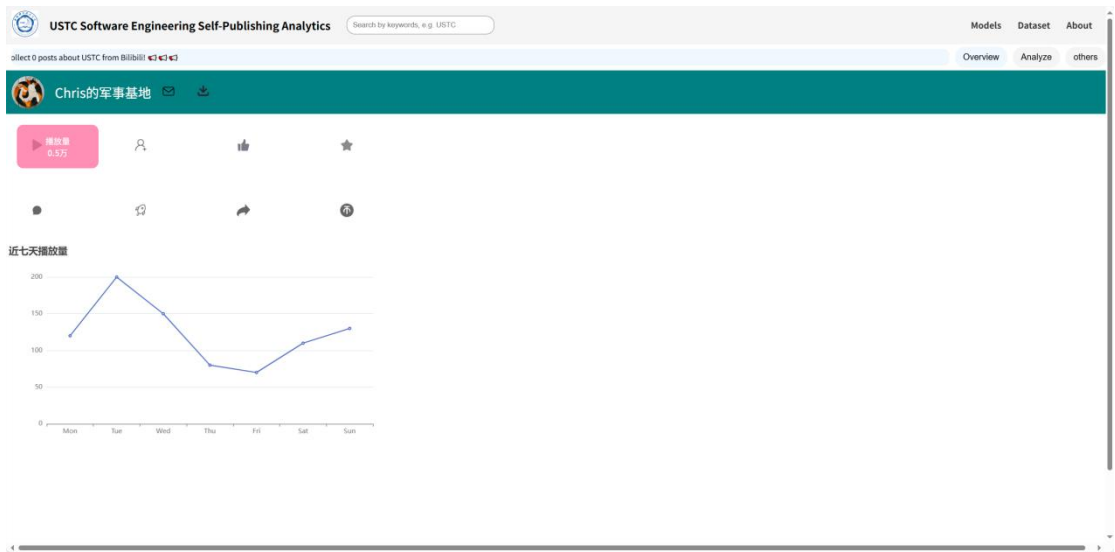
例如头部导航栏、侧边栏、表单组件等。每个组件应该有清晰的职责和接口



十、UI 和系统外部接口的设计

10.1 用户界面设计

10.1.1 页面原型图

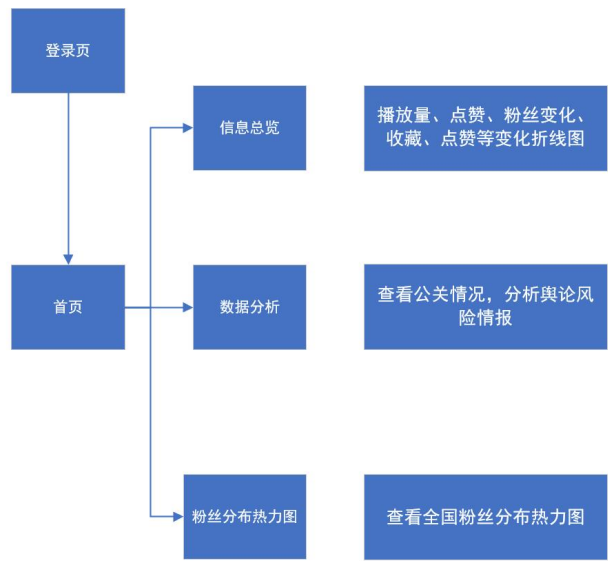


10.1.2 页面设计、

主要包含以下几个主要页面：

- (1) 登录/注册页面：包含登录/注册表单，有用户名和密码字段。
- (2) 主页/仪表盘：展示多种数据可视化图表，如折线图、柱状图、饼图等。
- (3) 通知页：展示用户的通知信息。

10.1.3 页面结构图



10.1.4 用户交互流程

- (1) 登录/注册操作：用户名和密码的输入，提交按钮的点击。
- (2) 数据图表的查看：滚动、放大、缩小、点选等交互操作。
- (3) 通知信息的查看和管理：查看详情、标记已读、删除等操作。

10.2 外部接口设计

技术选型

使用 Vue.js 作为前端框架，或 ECharts 进行数据可视化。

输入/输出

输入：后端接口提供的数据，用户的交互操作。

输出：可视化图表、用户界面。

接口描述

后端交互主要通过 RESTful API 进行，接口可能包括但不限于以下几种：

1. 用户登录接口

接口地址：/login

请求方式：POST

接口说明：用户登录，返回用户凭证

请求参数：

参数名	类型	是否必填	说明
username	string	是	用户名
password	string	是	密码

返回结果示例：

```
{
  "token": "string (认证令牌)"
  "userId": "string (用户 ID)"
  "role": "string (角色)"
}
```

2. 用户登出接口

接口地址：/logout

请求方式：POST

接口说明：登出当前用户，清除会话

请求参数：

无

返回结果示例：

```
{
  "message": "string (登出结果提示)"
}
```

状态码说明：200 成功

3. 获取不同类型数据接口

接口地址: /notification
请求方式: GET
接口说明: 获取用户系统通知列表
请求参数:
无
返回结果示例:

```
[
  {
    "title": "string",
    "time": "string",
    "content": "string"
  }
]
```

状态码说明: 200 成功

4. 获取评论信息接口

接口地址: /data/:type
请求方式: GET
接口说明: 根据类型获取数据
请求参数:
参数名 类型 是否必填 说明
type string 是 数据类型, 如 comment、video
返回结果示例:

```
{
  "data": "object (对应类型的数据)"
}
```

状态码说明: 200 成功, 404 类型未定义

5. 获取 json 格式视频信息接口

接口地址: /videoData
请求方式: GET
接口说明: 获取视频数据, 返回 JSON 格式
请求参数:
无
返回结果示例:

```
{
  "videos": "array (包含视频相关信息的 JSON)"
}
```

状态码说明: 200 成功

十一、数据的持久化设计

11.1 数据库选型

数据库类型：MySQL 8.x
存储方式：关系型表结构
说明：采用 MySQL 以支持结构化查询和事务控制，适合用户、评论、分析数据等持久化存储。

11.2 模块划分与持久化对象

模块	持久化对象
用户管理模块	用户信息（users）
舆情收集模块	视频数据（videos）、评论数据（comments）
舆情分析模块	情感分析结果（sentiments）、关键词提取（keywords）
系统管理模块	通知、日志等（notifications、logs）
用户管理模块	用户信息（users）

11.3 数据表设计

11.3.1 用户表

类字段名	Python 类型	映射表字段名	数据库字段类型	说明
id	int	id	INT PRIMARY KEY AUTO_INCREMENT	用户 ID
username	str	username	VARCHAR(50) UNIQUE	用户名
password	str	password	VARCHAR(100)	密码（加密）
role	str	role	VARCHAR(20)	角色
created_at	datetime	created_at	DATETIME	注册时间

11.3.2 视频表

类字段名	数据类型	映射表字段名	数据库字段类型	说明
id	int	id	INT PRIMARY KEY AUTO_INCREMENT	视频 ID
title	str	title	VARCHAR(200)	标题
description	str	description	TEXT	描述
upload_time	datetime	upload_time	DATETIME	上传时间
source	str	source	VARCHAR(50)	平台来源

11.3.3 评论表 (comments)

类字段名	数据类型	映射表字段名	数据库字段类型	说明
id	int	id	INT PRIMARY KEY AUTO_INCREMENT	评论 ID
video_id	int	video_id	INT	视频 ID
user	str	user	VARCHAR(100)	评论者
text	str	text	TEXT	评论内容
created_at	datetime	created_at	DATETIME	评论时间

11.3.4 情感分析表 (sentiments)

类字段名	数据类型	映射表字段名	数据库字段类型	说明
id	int	id	INT PRIMARY KEY AUTO_INCREMENT	分析 ID
comment_id	int	comment_id	INT	评论 ID
sentiment	str	sentiment	VARCHAR(20)	情感分类
confidence	float	confidence	FLOAT	置信度
analyzed_at	datetime	analyzed_at	DATETIME	分析时间

11.3.5 关键词表 (keywords)

类字段名	数据类型	映射表字段名	数据库字段类型	说明
id	int	id	INT PRIMARY KEY AUTO_INCREMENT	关键词 ID
video_id	int	video_id	INT	视频 ID
word	str	word	VARCHAR(50)	关键词
frequency	int	frequency	INT	出现频率
类字段名	数据类型	映射表字段名	数据库字段类型	说明

11.3.6 通知表 (notifications)

类字段名	数据类型	映射表字段名	数据库字段类型	说明
id	int	id	INT PRIMARY KEY AUTO_INCREMENT	通知 ID
user_id	int	user_id	INT	用户 ID
message	str	message	TEXT	通知内容
status	str	status	VARCHAR(20)	状态
created_at	datetime	created_at	DATETIME	创建时间

11.4 实体关系图 (ER 模型简述)

1. 一个用户 (users) 可能接收多个通知 (notifications)
2. 一个视频 (videos) 有多个评论 (comments)
3. 每条评论 (comments) 对应一个情感分析记录 (sentiments)
4. 每个视频 (videos) 提取多个关键词 (keywords)