**2.3 项目任务内容**

**任务10：研发AI代理应用**

**研究目标**

随着消费类视听终端（如电视、智能屏等）的普及，用户对个性化内容推荐、智能交互的需求日益增长。本项目面向传统视听终端操作系统存在数据处理能力和推理效率的限制，基于消费类视听终端操作系统的多模态输入输出和调用AI大模型等功能，通过任务分析，优化云端与本地协同推理机制，研究提出AI代理实现技术，实现轻量化的数据推理与动态推荐，提升智能交互体验。

**主要研究内容**

开展基于消费类视听终端操作系统的 AI 代理实现技术研究。基于操作系统的多模态输入输出能力，调用云端视听大模型等功能，深入研究 AI 代理实现技术，开发面向该操作系统的 AI 代理应用。通过构建任务监管与过滤、任务分析、任务拆解、轻量化智能推理、任务执行与反馈等多个功能模块，结合本地与云端协同推理架构，使各模块紧密协作，实现云端与本地协同执行任务，提升用户的智能交互体验。

**参加单位任务分工:**

|  |  |
| --- | --- |
| 任务 | 负责单位 |
| 研发AI代理应用 | 北邮 |

* 1. **项目详细技术方案**

**技术路线：研究AI代理应用**

1. 研究AI代理应用框架

该研究的关键在于如何基于消费类视听终端操作系统的应用场景构建AI代理框架，达成系统智能交互、高效处理任务和稳定运行的效果。针对消费类视听终端的多模态输入输出场景，AI代理应用框架如图1所示：

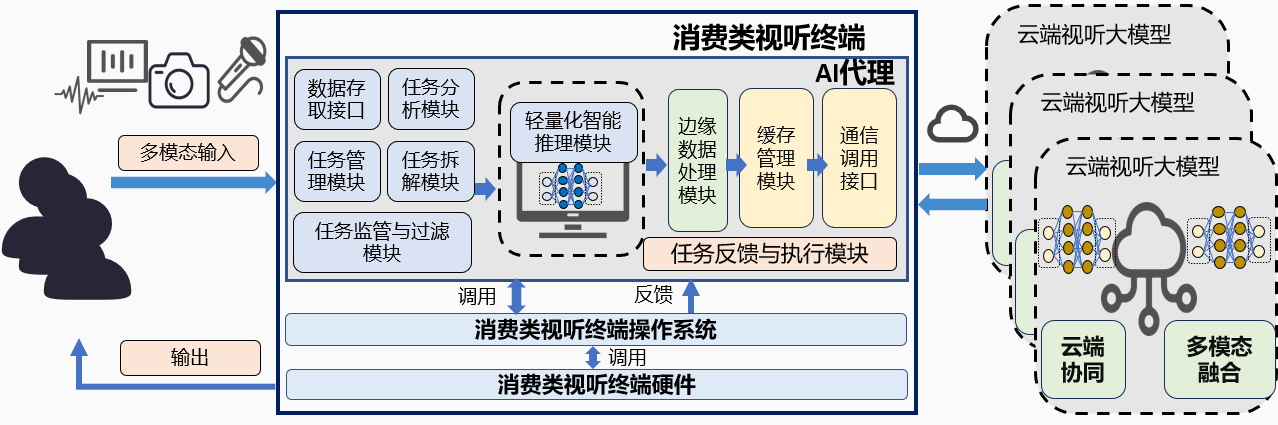
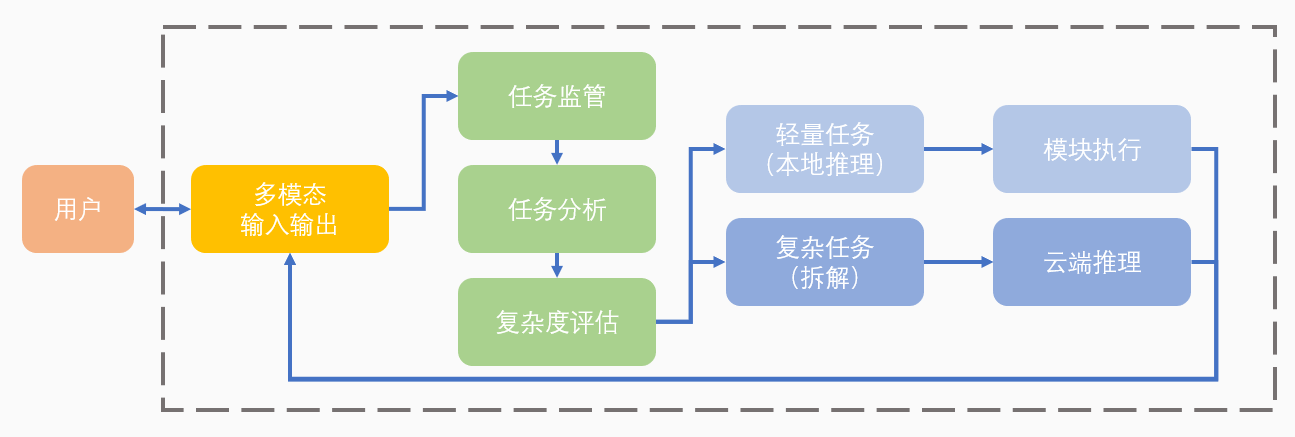


图1 基于消费类视听终端操作系统的AI代理应用框架

该框架采用采用模块化设计理念：多模态输入输出模块支持文本、语音、图像、视频等多种输入方式；任务监管与过滤模块负责针对用户发起的任务请求进行监管和筛选；任务分析模块针对经过监管与过滤的任务请求进行深度语义解析；轻量化智能推理模块在本地执行轻量化的智能推理任务；任务拆解模块针对复杂任务请求进行拆解；云端视听大模型交互模块针对高复杂度任务与云端的大模型进行交互；任务执行与反馈模块整合本地与云端的计算结果，执行任务操作并反馈给用户。各模块分工明确、协同工作，框架的技术路线如图2所示：



**图** **2 AI代理应用框架技术路线**

2）多模态输入输出模块

开发多模态输入模块功能，针对不同输入形式采用相应的技术方案。

文本输入：通过自然语言处理技术，系统能对用户输入的文本进行深度理解。通过扩充词库与上下文分析，系统能理解专业术语和口语化表达。

语音输入：集成语音识别技术，系统能适应不同环境的语音输入，捕捉语音信号，经过信号处理和模型运算，将语音精准转换为文本信息，为后续的任务处理提供基础。

图像/视频输入：通过计算机视觉技术，系统能够快速识别图像或视频中的物体、场景等元素，并分析物体的属性、位置、动作等信息，为后续任务处理提供基础。

开发多模态输出模块功能，针对不同输出形式采用相应的技术方案。

语音输出：通过语音合成技术，系统能够生成自然流畅的语音输出，并根据场景和用户需求调整语调、语速和音量。

文本输出：支持文本显示，将处理结果以自然语言形式呈现给用户。

3）任务监管与过滤模块、任务执行模块、任务分析模块与任务拆解模块

开发任务模块，针对任务流程进行解构，通过模块化设计实现对任务的监管与过滤、分析、拆解和执行功能。

任务监管与过滤模块：通过预设规则和基于历史数据的智能过滤机制，精确匹配和识别异常或无效任务。如检查请求格式是否正确、内容是否包含非法指令或敏感信息等：若请求合法合规，则将其传递至任务分析模块；否则系统提示用户重新输入。

任务分析模块：通过自然语言处理技术，系统能够对任务请求进行深度语义解析。通过分词、语法结构分析和语义关系理解，提取关键实体，并结合知识库进行任务分类和复杂度评估。若为纯文本任务，且复杂度较低，初步判定为轻量任务；若涉及多模态（图像、音频、视频），则判定为复杂任务。

任务拆解模块：针对复杂任务，依据预设的拆解规则或采用智能模式进行任务分解，部分简单任务在本地执行，部分复杂任务交由云端处理。规则引擎根据任务分类结果和任务请求中的关键信息，自动选择合适的拆解规则，将复杂任务拆解为一系列相对简单的子任务，确保每个子任务都具有明确的目标和可执行性。

任务执行模块：针对本地和云端的计算结果，基于统一的执行规范和接口标准进行操作执行。系统能够根据任务优先级和依赖关系，动态调度资源，确保任务高效执行。

4）轻量化智能推理模块、云端视听大模型交互模块、协同推理架构与反馈机制

开发推理模块，针对推理流程采用模块化设计方案，形成本地和云端两种推理方式，辅以协同推理架构和反馈机制有机结合。

轻量化智能推理模块：根据任务复杂度评估结果，动态选择本地推理或云端推理模式，针对本地简单任务基于轻量级模型和推理算法进行推理计算。并通过模型剪枝和知识蒸馏等优化技术，减少模型复杂度和计算量，提升本地推理效率。

云端视听大模型交互模块：针对复杂任务，基于智能任务拆解和高效通信机制，实现与云视听大模型的无缝对接和高效交互。系统能根据任务需求选择最合适的云试听大模型，并通过缓存机制优化推理效率。

协同推理架构：设计任务分发策略和数据交互策略，实现本地与云端资源的优化利用和协同工作。对于轻量级任务，优先在本地设备上执行，利用本地预先部署的轻量级模型和算法，在短时间内完成任务处理。 针对高复杂度任务，因本地设备的计算资源与存储能力限制，发送至云端进行处理。

反馈机制：对于需要本地和云端协同处理的任务，在任务完成后，将本地和云端的计算结果进行融合。根据任务的性质和需求，采用不同的结果整合方式，如图像识别任务中，本地设备先进行初步的图像特征提取，云端进行深入的分类与识别，将两者的结果进行合并与优化，为用户提供更准确的识别结果。

5）集成 AI 代理应用各模块

对AI代理应用各模块的集成主要分为以下三个步骤：

架构优化：针对系统架构进行整体审查和优化，统一各模块的数据交互标准，确保模块之间的数据传输格式一致，避免因数据格式不匹配导致的错误。如任务分析模块的输出数据需要与任务拆解模块的输入数据格式保持一致，确保任务信息能够准确传递。

模块整合：充分考虑模块间相互关系和协同工作方式，对各模块的功能边界进行清晰界定，确保模块之间功能不重叠、不缺失。如任务分析模块和任务拆解模块紧密协作，任务分析模块将用户请求解析后，准确地将任务信息传递给任务拆解模块进行处理。

系统调用优化：深入分析系统中各模块之间的调用关系，找出可能存在的冗余计算环节。如某些模块在不同的任务流程中会重复计算相同的数据，通过优化调用逻辑，将重复计算进行合并或缓存，减少系统资源的浪费。

各模块集成后进行模块功能测试和系统集成测试，确保AI代理应用的可用性与稳定性。

模块功能测试：针对每个模块的功能进行详细测试，验证其是否符合设计要求。设计全面的测试用例，涵盖各种可能的输入情况和边界条件。如测试多模态输入输出模块时，模拟不同的输入场景（如嘈杂环境下的语音输入、模糊图像输入等），确保模块能稳定处理各种输入形式。

系统集成测试：在模块集成完成后，进行集成测试，验证各模块之间的协同工作是否正常，数据交互是否准确无误。如测试任务监管与过滤模块与任务分析模块的协同工作时，模拟多种任务请求（包括合法任务和非法任务），确保系统能够正确识别和处理不同类型的任务。