

项目现状与差距分析

该项目旨在提升大模型的主动提问能力和推理性能，代码库已包含完整的人机交互系统及强化学习训练框架。例如，**多数据集支持**（HotpotQA、AmbigQA、GSM8K 等）和**自动对话生成功能**已被设计集成（README中提及使用 Gemini API 生成多轮对话）¹。核心代码模块包括多轮交互系统（`multi_turn_system.py`）、Gemini API 集成脚本、数据扩展脚本、训练脚本（stage1/2）、及主入口文件等²¹。项目文档中也列出了 GPT-4 用户模拟、奖励系统、PPO 训练和评估体系等模块已完成³，并提供了快速安装、测试和运行示例（如 `quick_test.py`、`gemini_integration.py`）⁴²。

尽管基础架构较为齐全，当前运行结果暴露了若干**问题与差距**：

- **评估指标未达标**：在本地 Shadow 评估中，新旧系统的 Spearman 相关仅约 0.475（阈值需 ≥ 0.55 ），Top-10 重合度仅 ~0.292（阈值 ≥ 0.60 ），意味着当前模型对高质量样本的排序差异较大，需要调试评分机制和模型行为。
- **数据集配置错误**：生成样本时使用了错误的数据集标识（如尝试加载 `strategy_qa` 导致失败），需修正为正确的 HuggingFace 数据集名称（如 `strategyqa`）或采用替代数据源。
- **评估脚本逻辑需完善**：`shadow_run.py` 等脚本已应用初步补丁（如优先使用已物化 HF 数据），但仍须进一步修复参数处理和字段映射逻辑，确保不再覆盖真实 HF 数据并正确输出 manifest。
- **功能覆盖不足**：目前缺少完善的单元测试和 CI 支持，建议补充测试用例；数据预处理和生成模块（如数据混合策略、轨迹收集）需验证其准确性和效率⁵。
- **文档与同步**：仓库已有详细的 README 和开发指南（包含项目结构、脚本说明等）²；但为适应协作开发，应配置 GitHub 分支和同步方案，确保本地与 Colab 环境的代码一致。

开发任务清单

- **修复评估流程**：完善 `shadow_run.py` 和相关脚本，保证优先读取物化数据，不重复生成；优化评分系统（考虑减少因全部任务成功而导致的恒定输入警告）；调整任务分层采样和统计逻辑，提升 Spearman 和 Top-K 指标。
- **更新数据脚本**：检查 `rebuild_shadow_from_hf.py` 等脚本中数据集名称和拆分配置（如 HotpotQA's “distractor”、GSM8K's “main”），并根据最新 `datasets` 库修改，以确保成功加载并生成样本。调整数据格式映射，避免 manifest 字段缺失或错误。
- **完善审计与检测**：运行并强化 `audit_shadow_data.py` 和 `pre_run_check.py`，确保数据质量审计和预跑检查模块正常工作。根据需求新增必要的健康检测指标和阈值反馈机制。
- **功能验证与测试**：在本地快速运行多轮对话系统（如 `multi_turn_system.py`）和训练数据生成（`dataset_expansion.py`）等脚本，验证输出格式和合理性。编写示例场景，确保用户模型能处理合作/不合作/打断等多种行为模式。
- **单元测试与文档**：补充缺失的单元测试，验证模块功能及配置逻辑；更新文档（例如 `configs/default_config.yaml` 说明、使用示例等）。同时，可对比“项目完成总结”文档，检查其中提到的完成模块是否已完全落地³²。
- **持续集成设置**：配置 GitHub 仓库（建议设置分支保护、CI/CD 测试等），以便多人协作；可使用预先定义的测试脚本或引入 GitHub Actions 进行代码检查和自动化测试。

Colab Gemini 编程助手调研

Google Colab 最近集成了 **Gemini Code Assist** 编程助手，使开发者在笔记本中获得 AI 辅助。根据官方介绍，**Gemini** 在 Colab 中提供代码补全和生成能力：用户可以在 Colab 界面找到侧边栏的 Gemini 图标，通过对话或命令生成代码、调试或重构 Python 单元⁶⁷。重要的是，Gemini Code Assist 对个人用户提供**非常慷慨**

的免费额度：每日约 6000 次代码补全 和 240 次对话请求⁸，同时拥有高达 128K token 的上下文窗口⁹，适合处理较长代码或多文件项目。

对比来看，**ChatGPT (特别是GPT-4)** 在编码一致性和复杂任务上通常表现更佳¹⁰。现有评测显示，在开放式多语言开发任务中，ChatGPT 更擅长一次性生成完整可用的解决方案，而 Gemini 则倾向于分段生成并提供详细解释¹⁰¹¹。例如，面对一个创建计算器的开放式提示，ChatGPT 直接给出了功能齐全的网页应用实现，而 Gemini 初次生成的是简单的 Python 脚本（Gemini 需要切换到 Canvas 模式才能生成可用界面）¹¹¹²。总体而言，**ChatGPT 更可靠、语境感知更强、适合生产级应用**，而 Gemini 则在免费助手的角色中提供了不错的编码指导和教学式解释¹³¹¹。因此，在 Colab 上使用 Gemini 进行日常编码和调试是可行的（特别是对资源有限的情况），但对于复杂逻辑和架构设计可能仍需依赖 ChatGPT 的分析能力。

本地 – 云端协作开发计划

为充分利用本地资源与 Colab GPU，建议实行“本地-云端双助手”协作方案：

- **本地（Cursor）角色：**由 ChatGPT（即当前的“Cursor”工具）负责高层决策和复杂分析。主要任务包括：整体架构设计、算法调优、测试分析、日志审查、文档编写和 Git 操作管理等。当本地环境安装完成、基本功能验证后，Cursor 可编写/审核代码、调整配置，并将稳定代码推送至 GitHub。Cursor 特别适合处理多任务描述、生成复杂逻辑代码、以及逐步调试不易并行化的问题。

- **云端（Gemini）角色：**在 Colab 上运行带有 GPU 的 Notebook 环境，由 Gemini Code Assist 辅助实现具体编码和实验。主要任务包括：**重度计算任务**（如大规模模型训练、强化学习训练流程）、**数据处理脚本编写、辅助构建训练数据集**等。Gemini 可在 Notebook 中快速生成数据预处理、模型训练、评估可视化等代码片段，并即时协助调试（例如通过 Code Completion 或画布模式）。由于 Colab 提供 GPU/TPU 资源，训练大型模型和并行实验将在此环境完成，且生成的模型输出或中间数据可以保存至 Google Drive 或云端存储，再同步回本地仓库。

- **任务拆分与协同：**建议在完成本地模块开发（如脚本编写、单元测试后）后，即刻通过 GitHub 同步至远端，让 Colab 环境拉取最新代码。在 Colab 中，可以直接克隆仓库、编辑或运行 Notebook，然后将修改提交回 GitHub（可使用 Colab 的 Git 集成功能或命令行操作）。例如，一次训练迭代脚本由 Cursor 编写基础版本后，可在 Colab 上调用 Gemini 完善，并在完成后 push 到 `feature/xxx` 分支。Cursor 监控合并流程，并根据评估结果更新代码。如此循环，本地负责“**构思与验证**”，云端负责“**实现与扩展**”。

- **同步方案与工具：**利用 GitHub 作为唯一代码源保证同步一致；在 Colab 中可使用 `!git pull`/`!git push` 或 Colab 的 GUI 功能。定期由 Cursor 整合并审查 Colab 上完成的修改。此外，建立合理的分支策略（如 `main`/`develop`、特性分支）及合并流程，确保开发节奏可控。建议 Cursor 和 Gemini 的任务分配固定化：例如，Cursor 定期产出周计划、编写 Issue 列表，Gemini 在 Colab 日常提交实验报告和小bug修复。

项目文件与后续对接

本文件汇总了项目的现状、问题、Colab 与 Gemini 调研结果及协同开发方案。接下来可将此内容保存为项目摘要文档（例如 `PROJECT_PLAN.md`），并同步到仓库。在新对话或新开发者加入时，提供该文档可快速了解整个项目背景与后续任务。在后续工作中，建议定期更新项目文档和开发计划，以适应实验进展和资源变化。

参考文献：项目 README 和文档¹¹⁴²总结了系统功能与模块；Gemini Code Assist 官方说明和评测指出了其在 Colab 中的功能及与 ChatGPT 的比较⁶⁸¹⁰¹³。

¹ ⁴ ¹⁴ README.md

<https://github.com/Atomheart-Father/llm-active-questioning/blob/84eb040f9f1327768895970e26cfb6e68fd3e38e/README.md>

2 **GITHUB_READY.md**

https://github.com/Atomheart-Father/llm-active-questioning/blob/84eb040f9f1327768895970e26cfb6e68fd3e38e/GITHUB_READY.md

3 5 **PROJECT_SUMMARY.md**

https://github.com/Atomheart-Father/llm-active-questioning/blob/84eb040f9f1327768895970e26cfb6e68fd3e38e/PROJECT_SUMMARY.md

6 **Gemini Code Assist vs GitHub CoPilot and ChatGPT Github**

<https://www.frozenlight.ai/post/frozenlight/629/gemini-code-assist-vs-github-copilot-chatgpt/>

7 **Gemini in Colab Enterprise | Gemini for Google Cloud**

<https://cloud.google.com/gemini/docs/colab/overview>

8 9 **Gemini Code Assist Now Grants Generous Free-Usage Limits to Everyone - InfoQ**

<https://www.infoq.com/news/2025/03/gemini-code-assist-free-tier/>

10 11 12 13 **Gemini vs ChatGPT for Coding: Which AI Model Is Better?**

<https://www.index.dev/blog/gemini-vs-chatgpt-for-coding>