

2025年秋云计算期末大作业

《智能体云原生开发》期末大作业需求说明书

1. 作业目标

本大作业要求大家综合运用云原生技术（Cloud Native）与大模型智能体（LLM Agents），构建一个具备实际应用价值的学习相关系统。重点考察大家对工程的理解与工程实现能力。

2. 交付物

- 代码仓库（GitHub/GitLab）：包含 Dockerfile、依赖配置文件、完整源码及环境配置指南。
- 演示视频（3-5分钟）：录屏展示核心功能，并包含 1 分钟左右的架构讲解（需说明数据流向及云服务调用逻辑）。
- 技术文档（PDF/Word/MD都可以）：
 - 架构设计：系统架构图及使用到的云原生组件（如 Docker、K8S、Redis, Serverless、微服务等），LLM Agent的工具链等。
 - 分工说明：明确每位成员的贡献百分比及具体负责模块。
 - 智能体策略：展示系统中关键的 Prompt 模板及其LLM Agent设计过程和工具链。

给同学们的开发建议：

- 架构优先：所有的命题体现“云原生”特征，避免单一脚本运行，使用容器化部署。
- 注重容错：大模型存在幻觉，你的智能体设计要考虑校验环节（Check layer）。
- 分工明确：建议组员分为“算法/Agent 组”和“架构/工程组”，在分工介绍中详细说明。

3. 命题详细说明

命题一：PPT 内容扩展智能体

- 痛点场景：考前复习只有干巴巴的 PPT 标题，缺乏背景细节和深度解释，导致自学效率低下。
- 核心目标：开发一个能“读懂” PPT 逻辑并自动查漏补缺的智能助手。
- 输入要求：PPT 文件（本地上传或云端 URL）。
- 至少包含下面功能清单：
 - 语义解析：识别 PPT 的层级结构（如目录、主标题、子标题、正文及图片描述）。

- **知识扩充：** 针对每一页知识点，自动调用 LLM 或搜索工具补充原理说明、公式推导或代码示例。
 - **多维搜索：** 联动外部权威资源（如 Wikipedia, Arxiv, 学术 API）获取延伸阅读材料。
 - **云原生技术（仅参考）：**
 - 使用 Unstructured 或 PyMuPDF 进行文档解析。
 - 使用 Vector Database（如 Milvus, Pinecone）存储 PPT 切片，实现基于语义的相关性检索。
 - **考核指标：** 笔记的逻辑结构是否清晰；联想内容与原 PPT 内容的语义相关度。
-

命题二：学习效果评估和巩固智能体

- **痛点场景：** 学完新知识后缺乏客观的评估手段，无法发现自己的知识盲点。
 - **核心目标：** 构建一个能基于学习资料自动出题、判卷并进行个性化纠偏的闭环系统。
 - **输入要求：** 特定领域的教材、PDF 笔记或录音转写文本。
 - **至少包含下面功能清单：**
 - **动态出题：** 根据资料自动生成选择题、简答题，并确保题目覆盖核心考点。
 - **智能判卷：** 分析用户输入的答案，不仅给出对错，更要给出详尽的解析。
 - **弱点记忆：** 自动收集用户高频错误，生成“错题本”。
 - **云原生技术（仅参考）：**
 - 使用 Redis 或 MongoDB 存储用户的学习状态和历史错题（持久化记忆）。
 - 利用 LLM 评估模式（如 Prometheus 提示词法）以及 Agent 的一些策略提高判卷的公平性。
 - **考核指标：** 出题的难度梯度是否合理；针对错题的“小灶”建议是否有针对性。
-

命题三：跨学科知识图谱智能体

- **痛点场景：** 知识碎片化严重，难以看透不同学科间（如神经科学与深度学习）的内在关联。例如学习“神经网络”时，不知道它和“生物学”或“高等数学”到底有什么深层联系。
- **核心目标：** 利用智能体挖掘跨领域概念的桥梁，并构建可视化的知识图谱。
- **输入要求：** 一个核心概念词（例如“熵”、“最小二乘法”）。
- **至少包含下面功能清单：**
 - **关联挖掘：** 强制 Agent 在不同学科领域（数学、物理、社会学等）寻找相关概念。
 - **图谱构建：** 提取实体及其关系，生成标准的节点/边数据结构（JSON）。
 - **动态可视化：** 在 Web 端渲染可交互的跨学科知识网。

- **云原生技术（仅参考）：**
 - 后端使用 Neo4j 图数据库或轻量级图形数据结构存储关系。
 - 前端使用 D3.js 或 Echarts 进行关系拓扑展示。
 - **考核指标：**发现“远亲概念”的逻辑合理性；图谱展示的直观性。
-

命题四：行研雷达智能体——增量追踪与更新

- **痛点场景：**行业报告时效性极差，传统报告往往“生成即过时”，无法应对瞬息万变的市场。
 - **核心目标：**开发一个具备定时巡检、增量比对和冲突报警功能的动态监控智能体。
 - **输入要求：**初始行研报告或行业核心关键词。
 - **至少包含下面功能清单：**
 - **自动巡检：**利用云端定时器实现 24 小时自动全网资讯抓取。
 - **增量对比：**比对“新发现”与“旧结论”，识别数据变化（如预测增长率从 5% 调至 2%）。
 - **冲突仲裁：**当信息源冲突时，根据来源优先级（官方 > 媒体 > 传闻）自动判定可信度。
 - **云原生技术（仅参考）：**
 - 使用 Serverless Functions（如 AWS Lambda, 阿里云 FC）配合 Cron Triggers。
 - 使用 Object Storage (S3/OSS) 存储报告版本历史。
 - **考核指标：**能否准确识别并高亮显示关键数据的变动；定时任务的稳定性。
-

命题五：长文本“事实卫士”智能体

- **痛点场景：**长文档（如毕业论文、可行性报告）多人协同或分章节生成时，极易出现逻辑自相矛盾。
- **核心目标：**构建一个作为“中间件”的校验智能体，确保长文档事实的一致性。
- **输入要求：**5000 字以上的长文档内容。
- **至少包含下面功能清单：**
 - **事实提取：**自动提取文中的关键事实（数据、日期、结论、人名）。
 - **冲突检测：**扫描全文，发现并高亮前后不统一的描述（如第一章和第五章对同一数据的引用冲突）。
 - **溯源校验：**自动联网核实冲突事实的真实来源，并给出修正建议。
- **云原生技术（仅参考）：**
 - 使用 Redis 作为“事实黑板”实现并发状态下的统一事实注册。

- 实现结果的可视化分析仪表盘（Dashboard）。
 - **考核指标：**冲突查杀的准确率；对重复内容和逻辑矛盾的识别广度。
-

4. 评分标准（总分 100）

不卷代码行数，也不卷文档长度，希望大家真正从下面几个点做出有价值的东西：

- **技术架构（30%）：**是否合理使用了云原生组件，系统是否考虑了稳定性和扩展性。
- **智能逻辑（30%）：**工程架构，Prompt 工程、推理链路（CoT）、事实准确性等，提示词是否严谨，Agent 是否能处理异常输入或模型幻觉。
- **工程质量（20%）：**代码规范、README、Dockerfile、分工文档，是否能让老师看得懂。
- **演示效果（20%）：**视频讲解清晰度、场景解决的痛点深度Demo 演示是否流畅。

扣分项与警戒线

- **硬伤扣分 (-10~20分)：**代码无法在 Docker 或标准环境中运行成功。
- **严重违规 (取消成绩)：**
 - 抄袭他人已有开源项目且未注明引用。
 - 分工文档造假。