# **AI医疗助手 - GraphRAG技术概念验证计划 (PoC)**

### **1. 项目背景与目标**

#### **1.1 问题陈述**

当前，医院医生在诊疗罕见病时，常需花费大量时间查阅医学教科书或最新研究论文。此过程不仅效率低下，还可能因信息检索不全面而增加误诊风险。

#### **1.2 项目愿景**

我们旨在开发一款基于大语言模型（LLM）的AI医疗助手应用，为医生提供一个能够即时响应、信息准确的智能问答工具，帮助他们快速获取罕见病的诊疗方案与前沿研究成果。

#### **1.3 本次PoC目标**

本阶段的核心任务是进行技术概念验证（PoC）。我们不开发完整应用，而是聚焦于验证核心技术路径的可行性，为后续开发提供决策依据。具体目标包括：

* 验证GraphRAG技术处理医疗文本与PDF论文的有效性。
* 对比LlamaIndex与LangChain两大主流框架在构建GraphRAG流程中的优劣。
* 积累团队在GraphRAG项目上的实践经验。

### **2. 技术方向与验证策略**

#### **2.1 核心技术**

我们将采用检索增强生成（RAG）作为基础方法。考虑到医疗知识中实体与关系的复杂性，我们将重点探索**GraphRAG**技术，它通过构建知识图谱来捕捉数据间的深层联系，有望提供更具推理能力的答案。

#### **2.2 验证方案**

为全面评估技术可行性，我们将采取以下并行策略：

* **双轨技术验证**：分别使用LlamaIndex和LangChain两个框架，处理Text和PDF两种数据源，共计四个技术探索任务（T1-T4）。
* **建立性能基线**：不使用RAG，直接向基础大语言模型提问（B1），以此作为参照，衡量GraphRAG带来的实际增益。

### **3. 任务分配与时间表**

**当前焦点：** 团队所有精力应集中在 **3.1** 和 **3.2** 部分的任务上，确保它们在周五结束前完成。

#### **3.1 技术探索与测试任务池 (3人认领)**

| **任务编号** | **任务名称** | **任务描述** | **交付成果** | **截止日期** |
| --- | --- | --- | --- | --- |
| **T1** | **探索 LlamaIndex + Text 数据** | 目标:  - 用 LlamaIndex 库+一份 txt 格式的文本，实现GraphRAG的基本流程，打通“文本→图谱→问答”全链路。  - 初步体验知识图谱检索带来的推理能力提升。  要点:  - 选用结构清晰、有实体和隐含关系的 txt 文本（如百科条目、科普讲义等）。  - 用 LlamaIndex 支持的文本 Loader 加载数据。  - 实现实体抽取、关系提取、知识图谱构建、基于图谱的检索与问答。  难点:  - 如果文本内容较简单，抽取出的实体/关系较少，图谱结构会较为稀疏，难以体现多跳推理优势。  - LlamaIndex 的实体关系抽取能力受限于底层 LLM和配置，部分复杂关系可能无法自动识别。  - Graph Index 的构建和用法需要理解 LlamaIndex 的Graph模块。  建议:  - 初期选用2段以上内容有交叉的文本，提升实体连通性。  - 如实体抽取效果不好，可人工补充部分实体/关系。  - 测试时建议设计多跳（非直接事实）型问题。 | 一个可独立运行的Python脚本 | **周五, 6月20日, 结束前** |
| **T2** | **探索 LangChain + Text 数据** | 目标:  - 用 LangChain 库+一份 txt 格式文本，搭建GraphRAG流程，体验 LangChain 在知识图谱与检索方面的能力。  要点:  - 用 LangChain 的 Document Loader 加载 txt 数据。  - 实现实体识别、关系构建、图数据库（如 Neo4j/networkx）存储，基于图的多跳问答。  难点:  - LangChain 组件丰富但配置繁琐，需要理清工作流。  - 图数据库的选择与对接存在技术门槛。  - Prompt设计对多跳推理效果影响大，需多次调整测试。  建议:  - 先用 networkx（纯Python）快速实现原型，后续再考虑 Neo4j。  - 充分利用 LangChain 社区示例和官方模板。  - 多设计测试用例，覆盖单跳与多跳关系检索。 | 一个可独立运行的Python脚本 | **周五, 6月20日, 结束前** |
| **T3** | **探索 LlamaIndex + PDF 数据** | 目标:  - 以1-3份 PDF 文档为数据源，用 LlamaIndex 实现 PDF 解析、知识图谱构建与GraphRAG问答。  要点:  - 用 LlamaIndex PDF Loader 或外部 PDF 解析库（如PyMuPDF）提取文本。  - 完整实现实体抽取、关系构建、图谱存储、基于图谱的检索问答。  - 适当整理PDF内容，保证格式不会导致文本提取混乱。  难点:  - PDF文本解析准确性有限，格式错乱、表格图片丢失等问题常见。  - 医学PDF实体/关系多、命名复杂，抽取质量难保证。  - 图谱结构可能异常复杂，检索和推理难度增大。  建议:  - 开始阶段可用科普或技术类PDF代替，降低难度。  - 重点关注实体和关系抽取的准确率，可人工校对部分结果。  - 对于结构复杂PDF，先人工提取1-2页作为样本测试。 | 一个可独立运行的Python脚本 | **周五, 6月20日, 结束前** |
| **T4** | **探索 LangChain + PDF 数据** | 目标:  - 以1-3份 PDF 文档为数据源，用 LangChain 实现 PDF内容提取、知识图谱搭建和基于图谱的问答。  要点:  - 利用 LangChain PDF Loader 或集成 PyPDF2/pdfminer.six 等解析PDF文本。  - 结合实体识别、关系抽取、图数据库实现知识图谱构建与问答。  - 处理多文档情况下的实体归一和跨文档关系。  难点:  - PDF分段、格式错乱等影响下游抽取质量。  - 多文档、跨文档实体重名/指代消解，关系跨界问题突出。  - 图谱过大时，本地图结构易超内存。  建议:  - 先从单文档做起，调通流程后再扩展多文档。  - 如遇实体抽取瓶颈，可用ChatGPT/LLM手动辅助提取。  - 明确文档间实体和关系归一规则。 | 一个可独立运行的Python脚本 | **周五, 6月20日, 结束前** |

#### **3.2 数据准备任务 (Becky负责)**

| **任务编号** | **任务名称** | **负责人** | **任务描述** | **交付成果** | **截止日期** |
| --- | --- | --- | --- | --- | --- |
| **C1** | **最终数据源与核心问题确认** | **Becky** | 提供最终用于本次PoC的权威数据和问题。 | 1. 最终的数据文件。  2. 一份明确的问题列表。 | **周五, 6月20日, 结束前** |

#### **3.3 基线模型测试 (暂不设时间表)**

| **任务编号** | **任务名称** | **任务描述** | **交付成果** | **截止日期** |
| --- | --- | --- | --- | --- |
| **B1** | **Baseline LLM Performance** | 1. 确定2-3个要测试的基础LLM。  2. 准备好将用于测试的核心问题。  3. 不使用RAG，直接向这些模型提问，并记录它们的原始回答。 | 一份包含问题、所用模型、所用Prompt和模型原始回答的文档。 | **暂不设置** |

#### **3.4 团队成果汇总 (暂不设时间表)**

| **任务编号** | **任务名称** | **负责人** | **任务描述** | **交付成果** | **截止日期** |
| --- | --- | --- | --- | --- | --- |
| **D1** | **异步分享成果与发现** | **全体技术成员** | 1. 拿到C1提供的最终问题后，运行自己的脚本。  2. 对关键的问答结果进行截图。  3. 将截图和简要发现分享到项目微信群。 | 在微信群中分享的**一系列结果截图和讨论**。 | **暂不设置** |