

产品详细介绍

🔍 字段消歧助手

AI 产品经理 · 滴滴国际化数据分析团队实习项目 · 2026-07

Live Demo → field-kb-agent.streamlit.app

Python · BGE · ChromaDB · BM25 · DeepSeek API

问题与背景

痛点一：字段选错不报错，事后才发现

数梦同一张表存在大量名称高度相似、口径截然不同的字段组——SQL 能正常运行，但结论是错的，只有被业务方质疑才暴露，每次返工 2-3 小时。

真实案例：用 `is_td_finish` 做 CPO 分母，跑出数字比预期高 20%，排查后才发现应用 `is_td_grab`——因为客服工单在应答后任意环节均可产生。

痛点二：口径静默变更，无通知机制

数梦字段注释不跟踪变更历史，分析师常在不知情的情况下长期使用旧口径。

案例 A: `eyeball_pax_cnt` 新旧口径差异 10-30%，不知情时全部数据偏高。

案例 B: `commute_tag` 枚举合并后旧 GROUP BY 导致通勤人群被低估。

2-3h

字段选错单次返工

20-60h

团队月度返工损失

50-100h

月度字段确认耗时

5 张

高频坑表（导师圈定）

7 份

结构化知识库文档

产品架构



数梦 MCP

抓取字段元信息
批量获取注释



摄取 Agent

识别混淆字段组
生成结构化 MD



混合检索

BGE 向量 + BM25
RRF 融合 (k=60)



置信度分层

高/中/低三级
阈值 0.7 过滤



生成回答

DeepSeek API
表格对比 + 推荐

关键产品决策 (AIPM 核心工作)

分块策略

按二级标题整块切分

放弃：按段落/固定 token 数切分

知识库文档核心是字段对比表格，按段落切分会截断表格，检索到的 Chunk 是残缺的。按标题切分保留完整语义单元，每条踩坑案例独立成块，检索精度显著更高。

来源：算法同学 W 第一次技术评审 (D-11)

相关度阈值

阈值 0.7 (宁缺毋滥)

放弃：0.5 (更高召回率)

字段口径场景下，给出错误答案的代价远高于不回答——分析师宁可去问人，不能被错误引导出数据结论偏差。低阈值会让 LLM 拿不相关内容“凑答案”。

来源：算法同学 W 第三次技术评审 (D-13)

置信度分级输出

三级输出 + 强制免责

放弃：统一输出推荐结果

导师 L 评审时指出：Bot 回答“需向 Owner 确认”，分析师可能懒得确认直接用了。低置信字段需显式禁止给出推荐，强制引导用户找 Owner——这是用户体验倒逼的产品设计，不是技术约束。

来源：导师 L 第一次 Demo 评审 (D-18)

部署形态

双版本并行 (Coze + Streamlit)

不是妥协，是刻意设计

内部平台离职即失访问权限，不可携带演示。Coze / Streamlit 是公有云，离职后可继续演示 Portfolio。两个版本服务不同目标，共享同一套技术架构逻辑，不互相替代。

产品形态决策 · PRD §12.2

评估结果

87% 检索命中率 Top-3 含正确文档	8% 幻觉率 集中于注释为空字段	5min 字段确认耗时 原 15-30 分钟	20题 测试集规模 人工标注验证
------------------------------------	-------------------------------	-------------------------------------	-------------------------------

诚实说明：幻觉 8% 的根因是注释为空的字段，知识库内容本身不完整——这是数据质量问题，不是 RAG 问题。待 TODO-02 ~ 05 Owner 确认后，预计可降至 < 3%。

评审驱动的迭代（3 轮真实反馈）

来源	评审意见	我的判断	行动结果
算法 W	按段落分块会截断字段对比表格，单个 Chunk 语义残缺	采纳——分块是检索质量的基础，这是技术硬约束	改为按二级标题切分，命中率提升
导师 L	tick_cpo_cnt vs cpo_cnt 不只是去重差异，而是使用场景根本不同（事件 vs 人维度）	采纳——我靠注释推断了业务含义，但注释是错的。核心字段必须问懂业务的人	更新知识库文档，补充使用场景对比；System Prompt 新增场景对比输出规则
导师 L	低置信字段说“建议 Owner 确认”，分析师可能懒得确认直接用了	采纳——这不是技术问题，是产品逻辑问题。我对用户行为的预测不准	设计三级置信体系，低置信强制不输出推荐，显式阻断误用路径

三层产品形态（体现产品视野）

✅ 已实现 本地摄取 Agent - 调数梦 MCP 批量抓字段元信息 - 自动识别高相似度混淆字段组 - 按模板生成结构化 MD 知识库文档	✅ 已实现 Coze / Streamlit RAG Bot - 意图分类 → 路由检索 → 置信度分层输出 - 可对外演示，作为可携带 Portfolio - 检索命中率 87%，幻觉率 8%	📅 完整设计 · 待内部权限 内部理想方案 - 数梦字段变更 Webhook → 知识库自动迭代 - SQL 编辑器侧边栏实时混淆提示（零切换） - SQL 行为感知，自动识别隐性踩坑模式 - DC Bot 群消息触发，踩坑知识自动沉淀
---	---	--

用户旅程（字段确认场景）



技术栈

Python 3.9	Streamlit	BAAI/bge-base-zh-v1.5	ChromaDB (EphemeralClient)	BM25Okapi + jieba
RRF 融合 (k=60)	DeepSeek API	Coze 工作流	数梦 MCP	GitHub + Streamlit Cloud