

项目上下文说明书（AI 可读版）

政府违规信息智能核查系统 — 项目上下文说明书（AI 可读版）

一、项目简介（Project Overview）

本项目旨在构建一个**政府违规信息智能核查系统**，帮助政府领导和监管部门利用人工智能自动识别各类数据违规问题（如社保欺诈、税务异常、数据造假、市场监管违规等）。

核心愿景：

让政府人员无需懂数据库，只用自然语言就能“让AI帮他查出问题”。

系统通过自然语言理解（NLU）、数据库语义分析（Schema Understanding）、智能问题生成（Problem Bank Generation）和 Text-to-SQL 技术，形成“机器先查、人再核”的新型政府监管工作流。

二、目标用户与需求背景

目标用户：

- 某地政府领导及监管人员（有数据库访问权限，但无SQL编写能力）
- 政府数据中心技术人员（辅助AI核查）

核心痛点：





- 数据量大，人工排查成本高；
- 各部门数据库割裂，无法跨库核查；
- 领导不懂SQL，不知道如何提问；
- 无法快速发现数据矛盾与潜在违规。

系统解决方案：

- 建立一个能够自动学习数据库结构和业务逻辑的智能体；
- 自动生成“问题库”，提前发现潜在违规线索；

- 将自然语言需求转化为可执行的SQL查询；
- 输出证据链可追溯的结果报告。

三、系统目标（Project Objectives）

目标类别	说明
 功能目标	构建一套能自动理解数据库、生成问题库、执行SQL查询的AI系统
 用户目标	让非技术人员（政府领导）可直接用自然语言核查数据
 技术目标	实现“AI Schema 理解 → 提示词优化 → SQL生成 → 自我修正”完整流程
 效率目标	将核查效率从“按周”提升至“按小时”，并减少90%以上人工成本

四、系统总体架构（System Architecture）

系统分为四个阶段，每个阶段都有独立的AI模块：

阶段1：智能体创建（AI Schema Learner）

- 输入：多个政府部门的数据库（如社保、税务、市场监管、统计等）
- 过程：AI 自动解析数据库 Schema（表名、字段、类型、注释、外键）
- 输出：数据库“认知图谱”（Schema Understanding Graph）
- 特征：能识别字段语义（如“employment_status=在职”）、建立字段关联关系

阶段2：问题库生成（Problem Bank Generator）

- 输入：AI已理解的数据库 + 人类提供的规则样例
- 过程：
 - AI通过样例学习三类“冲突模式”：
 1. 状态矛盾（如在职+领取失业金）
 2. 资格冲突（资质与背景矛盾）
 3. 聚合异常（宏观数据异常）
 - 自动组合字段、生成潜在核查问题

- 输出：自动生成的问题库（预制问题模板 + 逻辑说明 + 涉及表字段）

阶段3：任务识别与数据聚类（Task-driven Data Clustering）

- 输入：用户自然语言需求（如“帮我对账”）
- 过程：AI识别任务类型，从数据库中聚类相关字段与表
- 输出：与任务高度相关的数据视图（Task-specific Data View）

阶段4：Text-to-SQL 转换模块

分为两个子模块：

(1) 提示词工程模块（Prompt Optimization）




- 接收自然语言需求
- 自动重写为“可被模型理解”的高质量提示词
- 示例：
“在职人员不得领取失业补贴”
→
“请生成SQL，查找employment_table中status='在职'且subsidy_table中type='失业'的人员”

(2) SQL生成与修正模块（Text-to-SQL + Auto Debugger）

- 使用微调后的模型生成SQL；
- 自动验证、捕获错误并修正；
- 循环迭代直至生成正确、可执行的SQL；
- 自动收集失败/成功样本用于模型持续优化。

五、输出结果与展现形式（Outputs & Visualization）

输出类型	说明
✅ 宽松查询结果	直接违规，含完整证据链（如A表与B表逻辑冲突）

输出类型	说明
 严格查询结果	疑似违规，需人工核查
 可视化看板	违规类型、地区、时间维度展示
 智能评分	基于机器学习的“违规风险评分”机制

六、技术实现要点（Technical Focus）

模块	技术方向
Schema 理解	自动解析数据库结构 + 语义标签生成
问题库生成	基于 Few-shot 学习与模式抽象的 Query Template 生成
Text-to-SQL	使用微调LLM（如Qwen2.5/Claude/Custom LLaMA）+ SQL验证循环
Prompt 优化	自适应提示词重写模型（Prompt Rewriter）
Debugger	错误捕获与自修正机制（SQL Error Feedback Loop）
持续学习	从历史修正中自动构建微调样本（Self-Refinement Dataset）

七、应用价值与未来扩展（Value & Future）

当前应用：

- 社保冒领检测；
- 税务异常核查；
- 统计造假识别；
- 企业空壳检测。



未来扩展方向：

- 跨地区数据互联互通；
- 智能风险预警模型；
- 对接政务协同平台，实现“发现→处置→反馈”闭环；
- 接入知识图谱，实现因果推理式稽查。

附录（适用于AI理解）

- **输入类型**：数据库结构（Schema JSON）+ 人类语言问题 + 历史违规样本；
 - **输出类型**：SQL语句、违规报告、可视化结果；
 - **关键依赖模块**：Schema Parser / Prompt Rewriter / Text2SQL / Auto Debugger / Problem Bank Generator；
 - **系统目标函数**：最大化 SQL 生成准确率 & 问题发现召回率。
-

如果你愿意，我可以现在帮你把这份说明书直接输出成一份：

-  Word 文档（团队共享版）
-  或者 Markdown 文档（AI友好版，适合放进项目仓库）

你希望我输出哪一种格式？（我建议 Markdown，因为AI读取最方便）