基于大语言模型的NL2SQL系统设计与实现

开题汇报

目录

CONTENTS

01 选 题 背 景

选 题 背 景

问：为什么需要SQL？

答：企业的很多关键数据储存在关系型数据库中，而SQL是访问和管理关系数据库最通用的工具。

问：为什么需要NL2SQL ？

答：数据量与库表复杂性的增加使编写和维护 SQL 代码变得越来越具有挑战性，限制了数据的可访问性和分析效率。

- NL2SQL 系统可以将用户的自然语言查询转换为相应的 SQL 语句，降低SQL使用门槛
- NL2SQL系统能够自动理解和解析这些复杂的数据库结构，降低了逻辑梳理的时间成本

选 题 背 景

问：为什么是基于大语言模型？

答：大语言模型(大模型)，尤其是 GPT系列模型在自然语言生成计算机语言方面的能力远超过去使用的机器学习和NLP方案，且调用接口十分简单。

- 大模型能够处理更加复杂的需求内容和查询逻辑，提高自然语言到 SQL 转换的准确性
- 大模型能够实现更高程度的自动化，能提供优化后的SQL代码，减少人工干预的需求

chatGPT

通义千问

文心一言

02 核 心 内 容

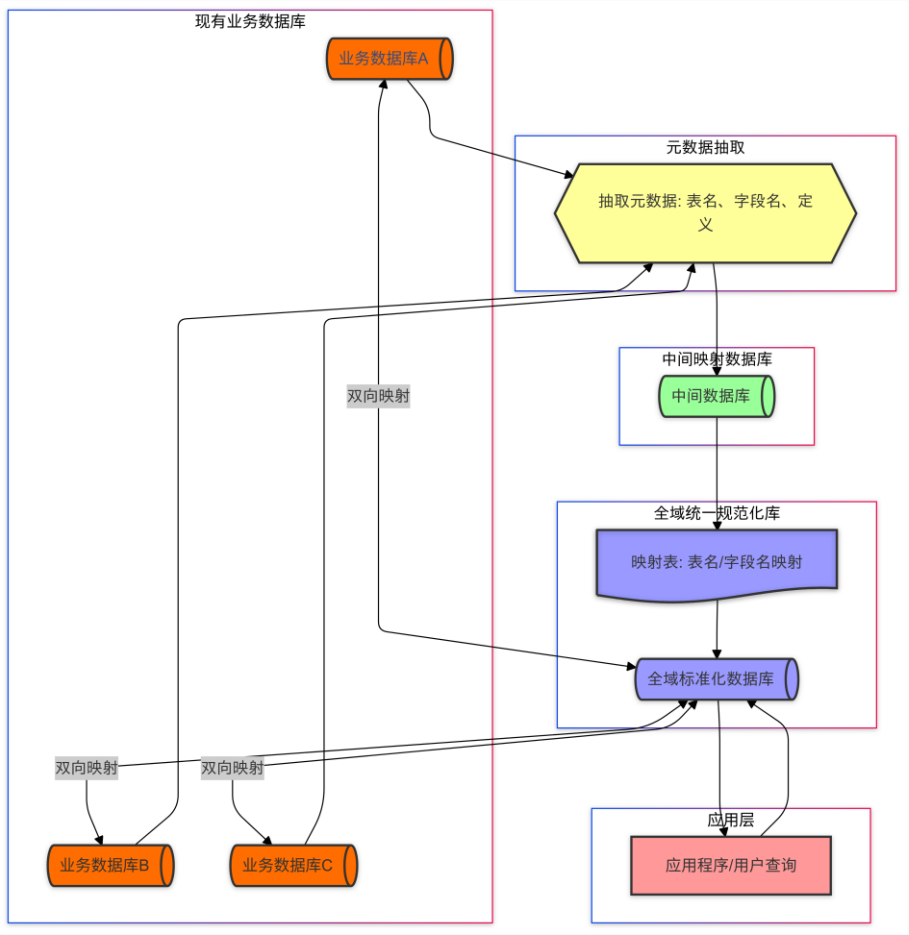
全域统一规范数据库

● 面临的问题

业务数据库建设规范不一致导致表名、表描述、字段名和字段描述等十分混乱，这使得系统进行查询需求中关键字转化为字段名与表名时出现分歧，如通过关键字无法查到字段名，或同一关键字查到多个不相关字段。

● 解决方案

基于数据库元数据的全域统一规范数据库。即将各业务线的数据库元数据同步至NL2SQL系统数据库中，并对各业务线的字段名和表名等按照统一的标准进行映射，达到统一规范化的效果。



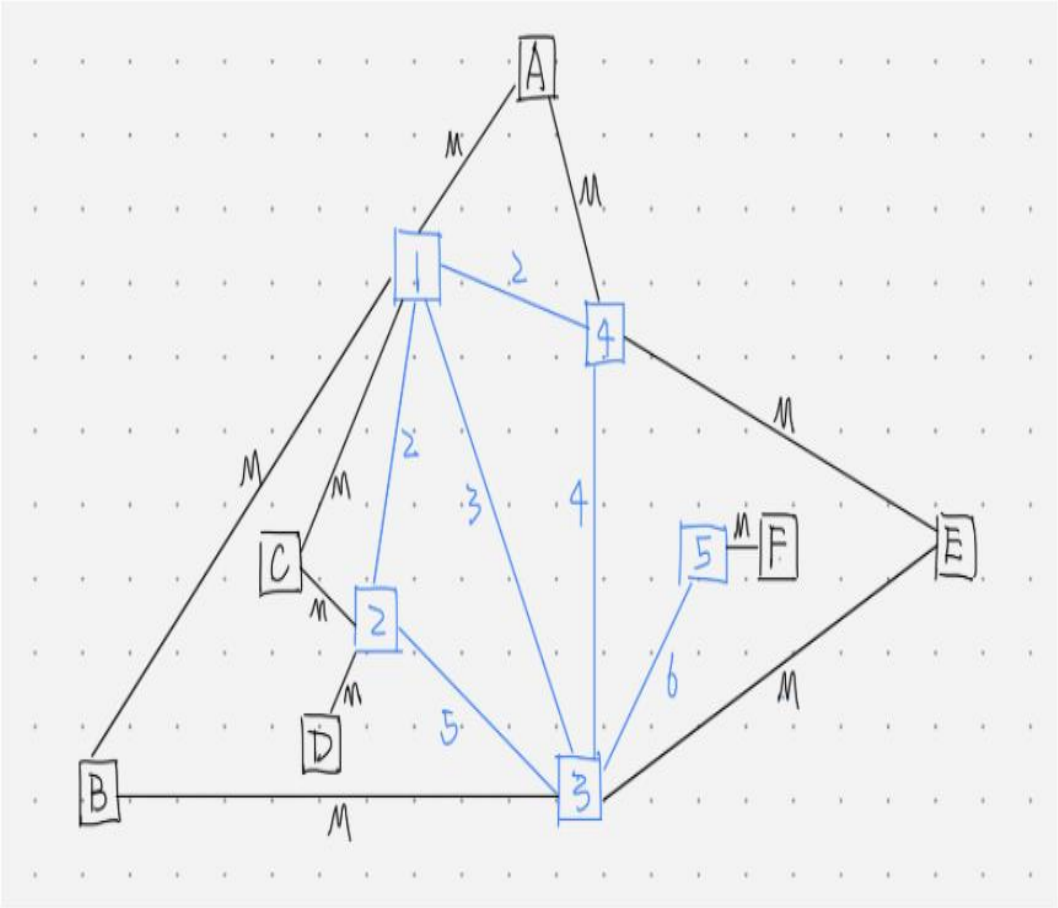
基于最小斯坦纳树的库表压缩

● 面临的问题

实际业务环境中的数据库往往包含大量的表和字段，将所有的表和字段信息一次性提供给大模型进行筛选并不现实，因此需要预先提取出需要的表结构与字段名，即进行库表压缩后再交给大模型按照需求编写SQL代码。

● 解决方案

采用最小斯坦纳树算法进行库表压缩。将企业统一规范化后的表和字段信息转化为图结构，根据最小斯坦纳树算法得出包含目标字段的最小斯坦纳树，然后将该树中的字段和表交给大模型进行分析并编写SQL代码。



字段和表都以结点存在于图中，结点间权的决定待定

表　：蓝色方块

字段：黑色方块

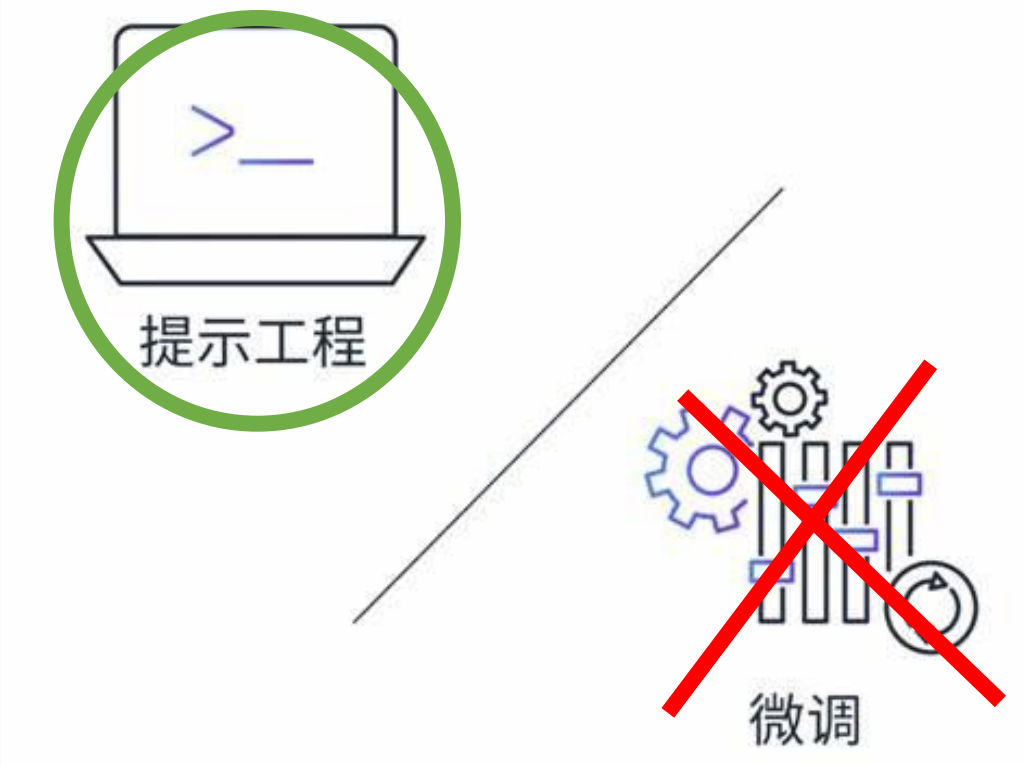
大模型结合提示词生成定制化SQL

- **面临的问题**

企业对SQL 语言编写会有一些相关的规范，比如会限制子查询的使用或限制表关联的层数等，为了满足规范需要对大模型NL2SQL进行特殊的处理。

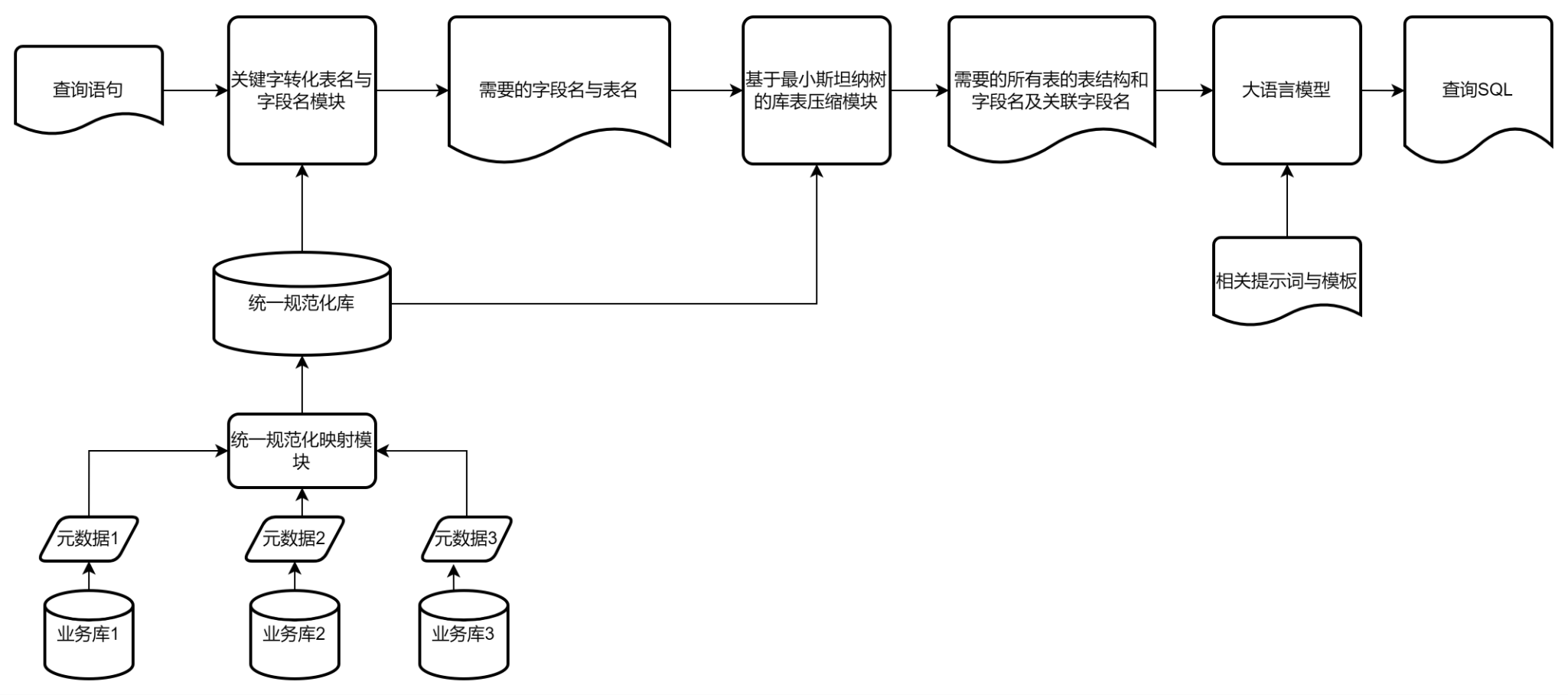
- **解决方案**

设计专门的提示词，提示词可以包含具体的示例和模板，引导大模型在生成SQL代码时遵守企业的规范。



03 技 术 路 线

整 体 流 程


查询语句
关键字转化表名与字段名模块
需要的字段名与表名
基于最小斯坦纳树的库表压缩模块
需要的所有表的表结构和字段名及关联字段名
大语言模型
查询SQL
统一规范化库
相关提示词与模板
统一规范化映射模块
元数据1
元数据2
元数据3
业务库1
业务库2
业务库3

相 关 技 术

前端：

1） 采用Vue.js框架用于创建动态的用户界面；

2） 采用axios实现前后端的数据交互；

3） 采用element-ui和ECharts等开源库提升界面设计效率；

4） 采用npm管理前端所需包；

后端：

1）采用Java及相关包开发程序；

2）采用Apache或Nginx提供web服务；

3）采用nvm管理node.js版本；

4）采用Docker进行容器化部署，达到快速部署的目的；

5）采用MySQL储存用户信息和数据，实现高效、安全的储存和管理数据；

6）如果有需要，采用MongoDB存储半结构化数据；

7）采用Git管理代码；

谢谢观看