

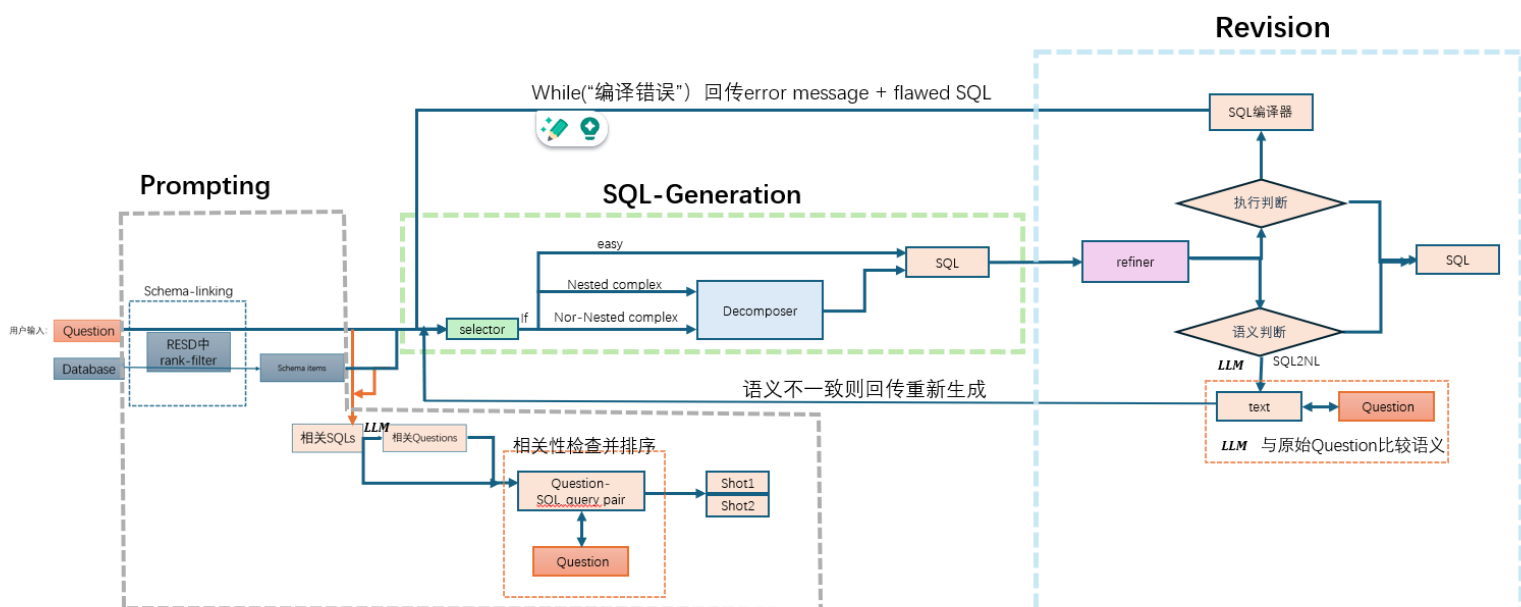
NL2SQL 方案 1

基本信息：

本系统主要分为 4 个部分：

1. Fine-tuning (微调大模型参数)
2. Prompt (为输入的问题提供相关的数据库内容和提供相关的例子 (few-shot))
3. SQL-Generation (接受了所有的 schema items, Question, few shots, 相关的修饰之后生成对应的 SQL)
4. refiner (对生成的 SQL 语句进行改进)

流程图（没有 fine-tuning 部分）



详细内容：

1. Fine-tuning: 这是在检测之前预训练的部分。(这一部分还没有搞明白是如何运作的)。通过阅读文章，我们有以下几个想法：
 - a) 首先是用跨领域(cross-domain)数据集去微调，这种方法可以提升模型在多个领域中的泛化能力。通过在多领域的数据上进行微调，可以使模型更好地适应不同领域的特征和模式，从而在各种任务中表现更佳。
 - b) 当前的训练集主要是training data $T = \{Q, S, T, C, D\}$ ，其中 Q 代表 question, S 代表 SQL, T 代表 table, C 代表 column, D 代表 database, 我们的想法是给 training data $T = (Q, S, T, C, N, D)$ ，其中 N 代表 natsql, 训练从 question 到 table, column, 再到 natsql, 接着 skeleton(看其骨架生成), 最后再是 SQL, 从

而使其训练效果达到最佳。

2. Prompt:

- a) Schema items: 输入整体的 Database content 与 输入问题进行一个初步的 schema-link, 在这里使用 RESDSQL 中的 rank-filter 筛选出于问题相关的表格信息 (我们通过相关实验看出这个 rank-filter 能筛选出与问题相关的问题 (不会将有用的问题筛选出去))
- b) Shot: 这里我们会给出与提的问题的例子 (包括问题和对应的 SQL)。例子是如何给出的所示如下: 我们使用大语言模型, 根据 SQL 的特定语法 (如 SELECT FROM, SELECT FROM WHERE, SELECET FROM WHERE GROUPBY HAVING) 和给出的数据库内容, 填充生成相关 SQL 语句, 然后将 SQL 语句转化成对应的自然语言问题, 一一对应 (Question-Query)。在输入要转化的问题之后, 将该问题与之前准备好的 Question-Query 表同时输入大模型, 进行相关性查找 (相关性主要依赖语法, 次要依赖内容) 找到最相关的两个问题, 这两个 Question-Query 作为 shot\例子, 也是 prompt 的一部分。
- c) 修饰: 一些提示词, 比如“接下来你要处理 NL2SQL 的相关内容”和一些给出内容的注释, 比如“database schema: ... Example1: ... Example2... Question:...”以及“优先考虑第一个例子, 次要考虑第二个例子”等。

3. SQL-generation: (这一部分我们主要使用现有的结构) 比如先使用一个 selector 检测需要生成的语句的复杂程度 (easy, nested complex, non-nested complex), 如果为 easy 则直接生成 SQL, 如果为之后的两个难易程度则使用 Decomposer, 先用 CoT 将问题分解成两个小问题 (如果小问题依旧很难就继续分解), 然后逐步生成对应的 SQL 语句并融合在一起生成整体的 SQL。(如图)

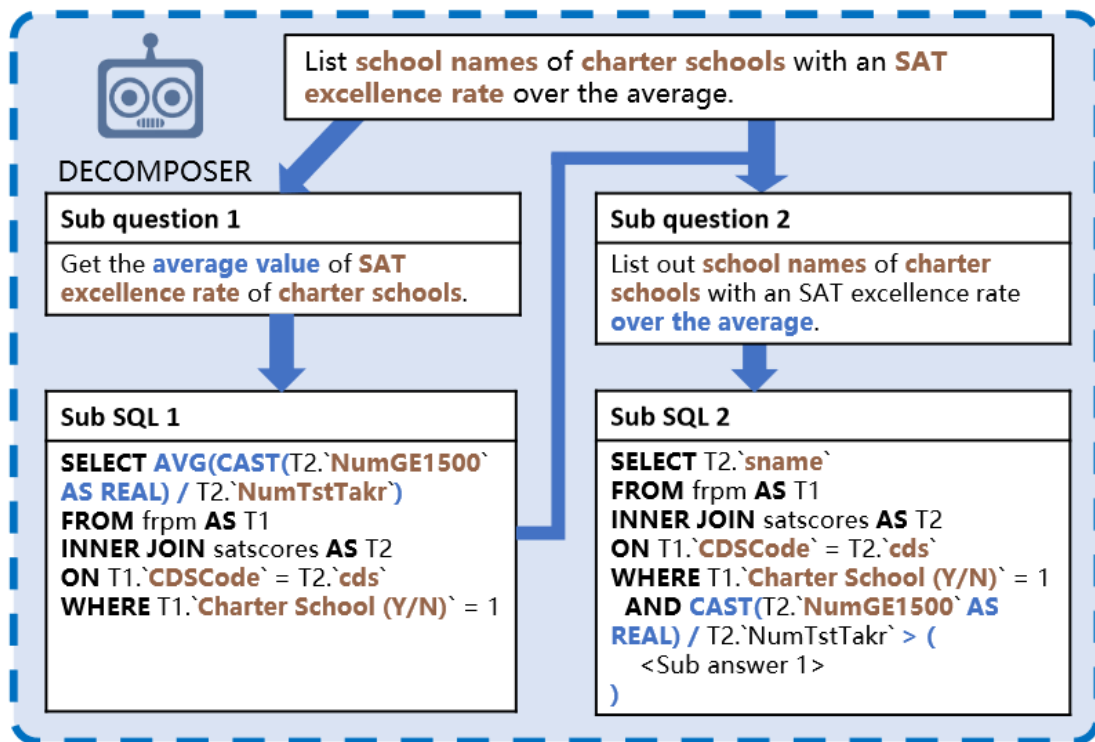


Figure 3: The Decomposer Agent Illustration.

4. Refiner: 我们改进了现有的改进机制，将 SQL-generation 生成的 SQL 同时输入到一个判断是否可执行的编译器和一个判断语义是否正确的大语言模型
 - a) 执行判断：输入到 SQL 编译器，如果成功则进行下一步，如果失败将报错的内容和该 SQL 同时输入回 SQL-generation 让他改进重新生成 SQL。
 - b) 语义判断：先将生成的 SQL 输入到大语言模型（与 SQL-generation 中不是同一个）里面转化为对应的问题，再将新生成的问题与原始的问题在大语言模型里面进行比较，如果成功则进行下一步，如果失败将失败的信息和该 SQL 同时输入回 SQL-generation 让他改进并重新生成 SQL

在 SQL 同时通过两个判断器后，我们就生成了最终的 SQL。