

示例指南

示例的用途

用途是作为提示词的一部分去提示大模型不同的查询条件对应怎么样的SQL语句. 一组示例由一对Query和SQL组成, Query会和用户的Query进行匹配, 匹配中后会作为提示词的一部分.

技巧

下面是两个相同结果的查询, 左边是正面案例, 右边是反面案例

```
1 SELECT
2 name 名字,
3 attendance_date 日期,
4 attendance_time 时间
5 FROM HWOA.dbo.V_ATN_Record
6 WHERE name = '{员工名字}'
7 AND attendance_date = '{日期}' --格式 %Y-%m-%d %H
```

```
1 DECLARE @employeeName VARCHAR(50);
2 SET @employeeName = '{员工名字}';
3
4 SELECT
5     v.name as 名字
6     a.attendance_date as 日期,
7     attendance_time as 时间,
8 FROM HWOA.dbo.atn_record a
9 JOIN V_Employee v ON a.employee_id =
    v.employee_ID
10 WHERE v.name = @employeeName
11 AND CONVERT(DATE, a.attendance_date) =
    '{日期}'
```

我们分析下为什么左边的好于右边的:

1. SQL越短越好, 因为给模型参考的SQL越短, 生成的SQL也会越短, 即模型生成SQL所输出的token数少, 出错的机会也越少, 同时所消耗的时间也会更少。
2. 没有JOIN操作. 因为一旦涉及到联合查询, 在select后面的字段名上就需要带上表名, 例如 a.attendance_date, 大模型生成的SQL中, 可能出现v.attendance_date, 但是如果是单表查询, 就不需要表名前缀。建议通过创建视图来消除JOIN操作。
3. 没使用数据库特性相关的语法, 例如数据库变量、数据库内部函数。因为大模型在训练阶段所使用的SQL训练语料只占非常小的一部分, 要细致地区分不同数据库的语法会增大难度。
4. 没使用变量. 不是所有数据库都支持变量, 因此带变量的SQL在大模型的训练语料中出现的次数更少, 这会使得生成带变量的SQL更容易出错, 并且使用变量会让SQL变得更长. 如果需要提示大模型变量的类型, 如"VARCHAR(50)", 建议放在数据源描述页面, 例如:

- 1 字段名, 数据类型, 描述
- 2 employee_ID, int, 员工ID, 是V_Employee的主键
- 3 name, nvarchar, 姓名
- 4 name, sysname, 姓名
- 5 need_attendance, tinyint, 是否需要打卡, 1-是、2-否
- 6 entry_date, date, 入职日期
- 7 organ_name, varchar, 部门名称

5. 输出字段的表示. "name as 名字"中, 语法as表示输出时, 将字段name显示为"名字", 这个技巧能提高输出内容的可读性.

其他技巧:

1. 示例SQL中添加的注释内容也有一定概率会生效, 因此可以通过添加注释的方式提醒模型去满足一些特殊的需求. 注意: 当添加注释的时候, 模型生成的SQL也会带有注释, 这会增加Token的输出量, 增加时间消耗.