



# 2024 AI+研发数字峰会

AI+ Development Digital summit

AI驱动研发变革 促进企业降本增效

北京站 08/16-17

## 质量大模型及其在接口测试场景下的实践

李庆泉 蚂蚁集团

# 科技生态圈峰会 + 深度研习



—1000+ 技术团队的选择



K+全球软件研发行业创新峰会

时间: 2024.06.21-22



K+思考周®研习社

时间: 2024.10.17-19



K+思考周®研习社

时间: 2024.11.10-12



K+峰会详情



AI+研发数字峰会

时间: 2024.05.17-18



AI+研发数字峰会

时间: 2024.08.16-17



AI+研发数字峰会

时间: 2024.11.08-09



AIDD峰会详情



# 2024 AI+研发数字峰会

AI+ Development Digital summit

深圳站 11/08-09

## AI 驱动研发变革 促进企业降本增效

### 2024深圳站-议题设置

AI+产品线	LLM驱动产品创新	LLM驱动需求与业务分析	AI驱动设计与用户体验
AI+开发线	AI 原生应用开发框架与技术	AI Agents在研发落地实践	LLM驱动编程与单测
AI+测试线	LLM驱动测试分析与设计	基于LLM生成测试脚本与数据	LLM和AI应用的评测
AI+工程线	AI+DevOps 与工具 (LLM 时代的平台工程)	大模型对齐与安全	端侧大模型与云端协同
AI+领域线	领域大模型 SFT 与优化	知识增强与数据智能	大厂专场

扫描右侧二维码  
查看更多会议详情



早鸟票限时抢购中 (截止到9月30日)

¥ 3680

早鸟票

¥ 2800

学生票

# 目录

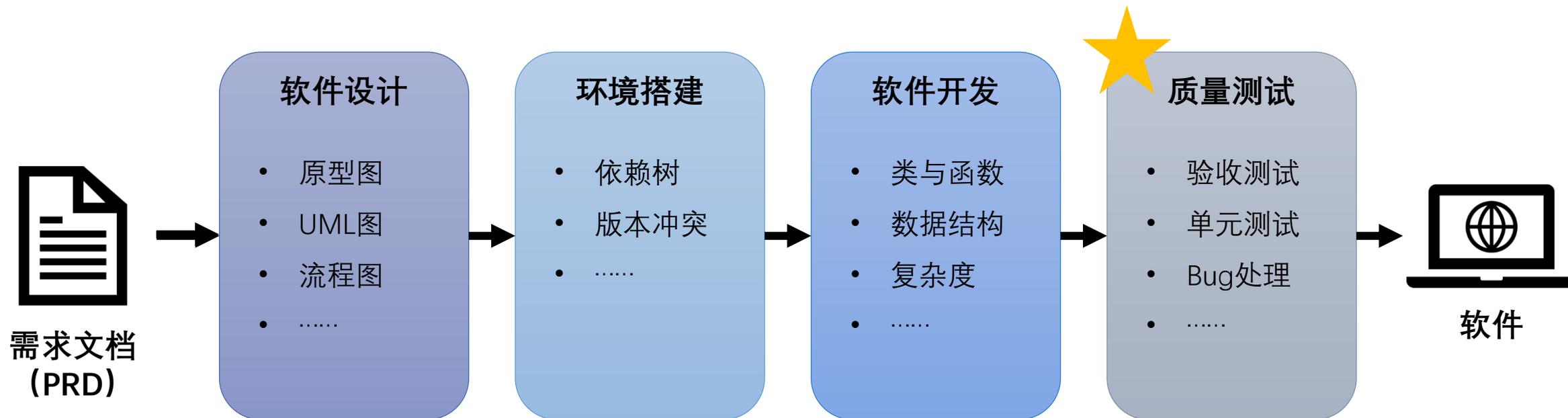
## CONTENTS

1. 大模型如何解决质量域问题
2. 质量大模型的构建路径
3. 质量大模型在接口测试中的实践
4. 未来展望

# PART 01

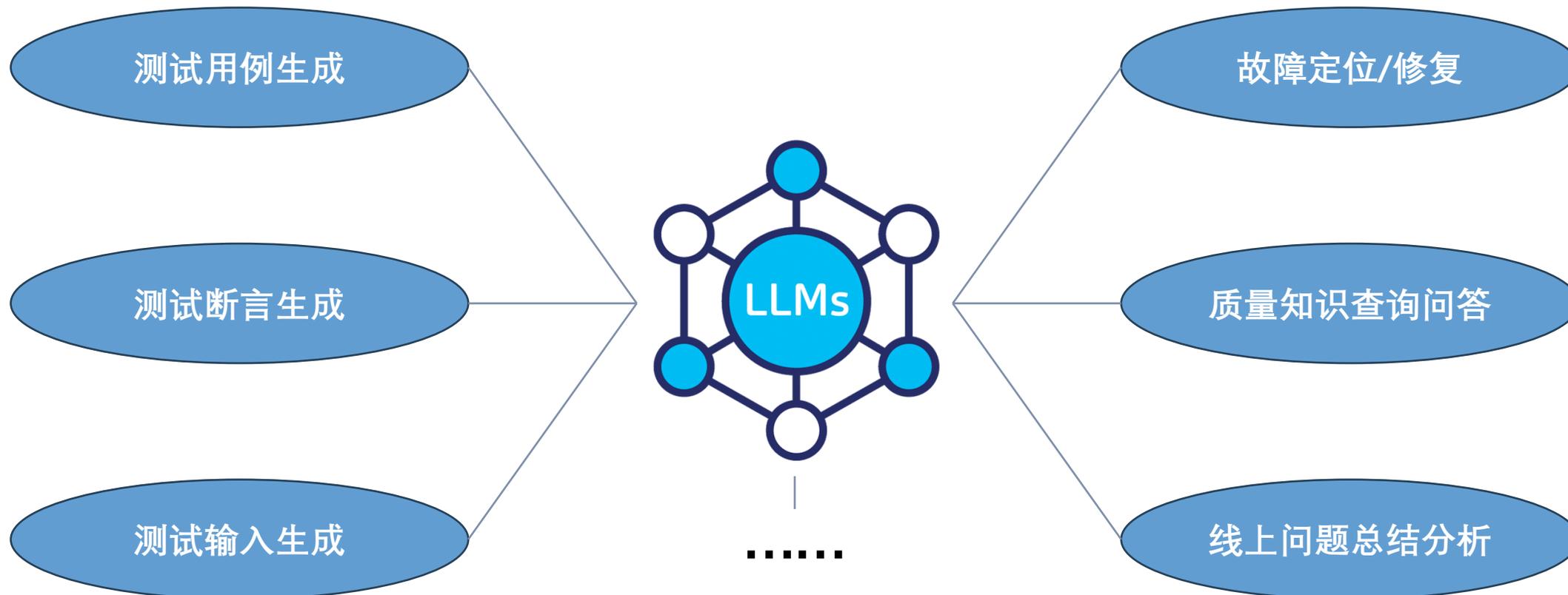
# 大模型如何解决质量域问题

# ▶ 大模型具有赋能软件研发全流程的能力



Li B, Wu W, Tang Z, et al. Devbench: A comprehensive benchmark for software development[J]. arXiv preprint arXiv:2403.08604, 2024.

# ▶ 大模型在质量域下的应用场景



Wang J, Huang Y, Chen C, et al. Software testing with large language models: Survey, landscape, and vision[J]. IEEE Transactions on Software Engineering, 2024.

# ▶ 大模型能够解决质量测试领域的哪些问题？

## 质量测试

### 覆盖率要求高

全面、准确地发现各种特殊情况与边界问题

### 自动化难度大

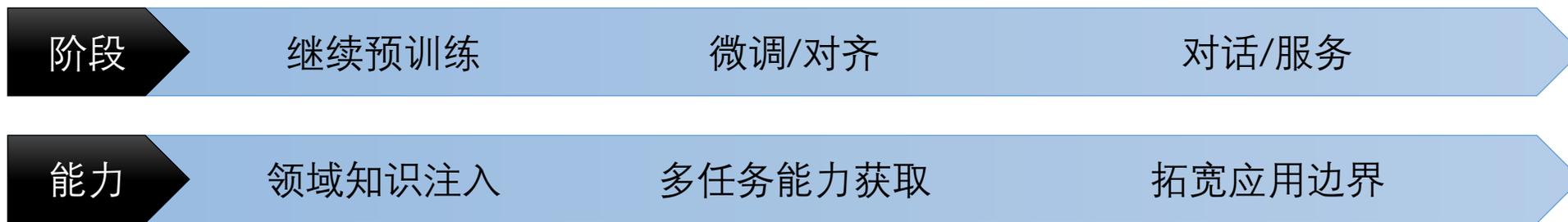
经验密度大、业务特异性高，难以自动化或手段难复用

### 人工成本高

测试用例撰写、质量问题分析等工作需要大量人工

1 严谨、全面

2 知识、经验



# ▶ 质量测试领域呼唤质量大模型



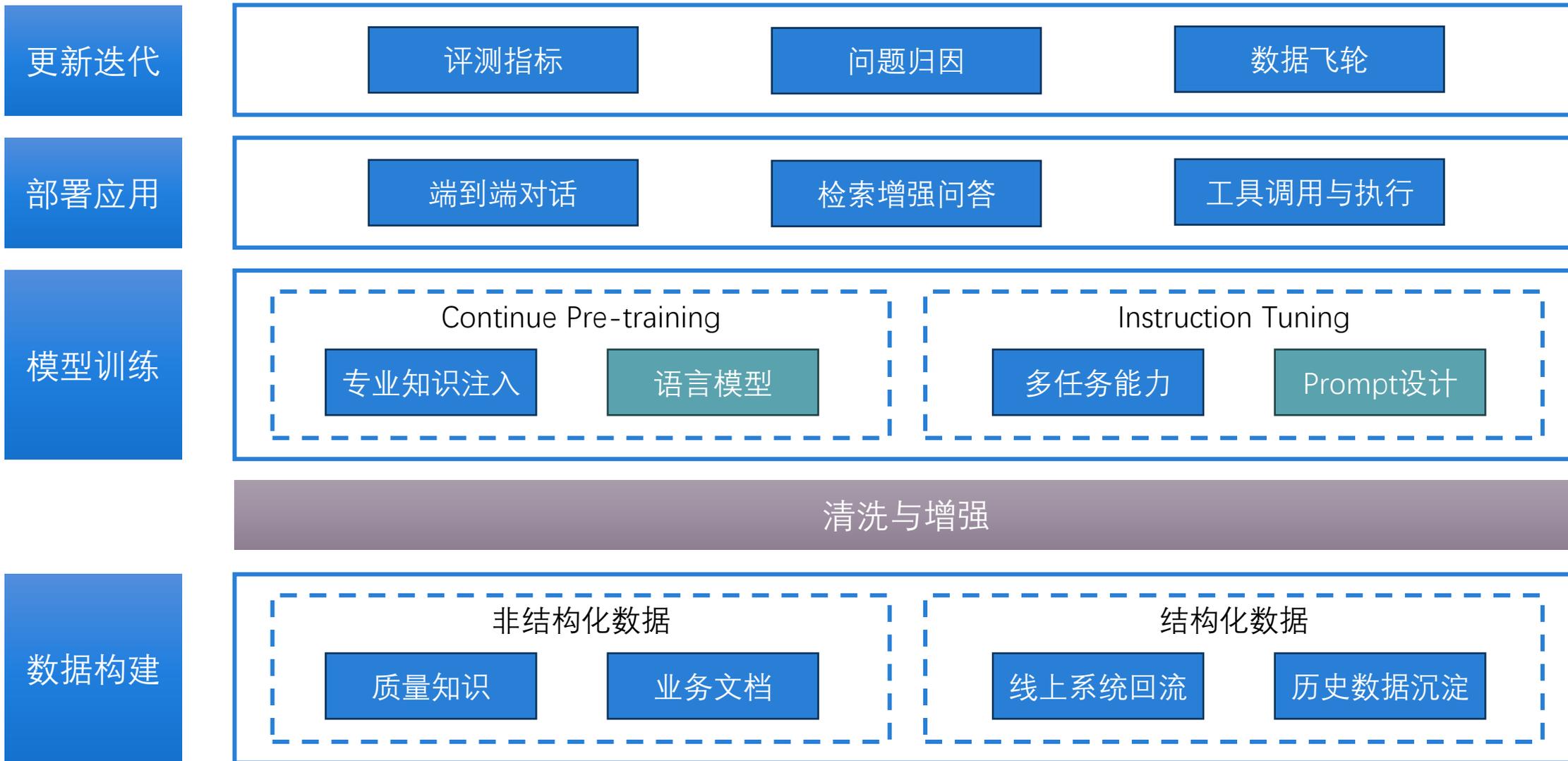
LLMs as Brains



## **PART 02**

# **质量大模型的构建路径**

# 质量大模型的构建路径



# ▶ 继续预训练——知识注入

训练材料：知识构建

知识来源

质量域

业务域

其他背景

知识处理

切片

聚类

数据增强

知识质量

筛选

清洗

数据分析

训练任务：语言模型

训练对象：开源模型底座



\* 不完全列举

# 指令微调——技能培训

**关键问题**：质量域中的许多任务都是以结构化数据为基础，如何用基于非结构化对话形式的大模型完成结构化数据任务？——解法在于**如何针对特定任务设计指令并进行微调**

流程

任务选取

数据获取

指令构建

微调

两大类任务选取：

质量/业务  
知识问答

向模型提供基础的  
质量问答能力

特定质量  
任务执行

以端到端形式执行的  
质量任务，如测  
试用例生成等

两大类数据来源：

基础知识

包括**外部**基础知识  
与**内部**的文档积累

任务数据

来源于真实场景的  
特定任务数据

指令构建技巧：

明确任务需求

清晰描述任务的推  
理逻辑、执行步骤

格式规约

对格式进行严格规  
约，降低工程难度

指令泛化

增强鲁棒性

微调注意事项：

数据分布

平衡各任务的数据  
分布，避免过拟合

训练成本

多卡训练、LoRA  
微调等，节省时间  
与显存成本

# 应用方式——作为端到端Bot

以端到端对话形式执行质量任务

上下文学习

**Input** Code Demonstrations

```
### Fix ESLint error in the following JavaScript code:
### Buggy JavaScript
return date.getFullYear();
break;
case 'month':

"rule_id" : no-unreachable
"evidence": Unreachable code.
"warning_line": break

### Fixed JavaScript
return date.getFullYear();
case 'month':
```

END\_OF\_DEMO Demonstration 1

Demonstration 2

...

Demonstration N

Query

**Instruction**

```
### Fix ESLint error in the following JavaScript code:
### Buggy JavaScript
return 'local';
break;
case 'mapbox:':

"rule_id" : no-unreachable
"evidence": Unreachable code.
"warning_line": break;

### Fixed JavaScript
```

*Error Snippet* *Error Context* *Context*

**Output**

```
return 'local';
case 'mapbox:':
```

逻辑构建

<b>Bug ID:</b>	JacksonDatabind-110
<b>Issue Title:</b>	Inconsistent handling of <code>Collections\$UnmodifiableList</code> VS <code>Collections\$UnmodifiableRandomAccessList</code>
<b>Issue Description:</b>	I'm sorry to bring that one up again, but I'm under the impression that the issue about unmodifiable collections <code>{..}</code> is still not solved completely. <code>{..}</code> Currently <code>java.util.Collections.unmodifiableList()</code> can only return these 2 types of unmodifiable lists, so I believe it is safe for now to just hardcode yet another special case for this class. This can currently be solved on user side by adding a mixin, bug since <code>Collections\$UnmodifiableRandomAccessList</code> is supported, I would find it natural to also support non-random access variant.

**Prompt 1**

Issue title + description	<buggy function>	Identify which line(s) are buggy in the code
---------------------------	------------------	--

The buggy line in the code is:

```
if ( type.hasRawClass ( CLASS_UNMODIFIABLE_LIST ) ) {
    conv = converter ( TYPE_UNMODIFIABLE_LIST , type ,
        list . class );
}
```

**Response 1**

**Prompt 2**

Previous context + response	Explain why these lines are buggy
-----------------------------	-----------------------------------

These lines are buggy because they only handle `Collections$UnmodifiableRandomAccessList`, which is not only possible type of unmodifiable list that can be returned by `Collections.unmodifiableList()`, as there is also `Collections$UnmodifiableList {...}`. This results in a failure to deserialize non-random access unmodifiable lists, such as those returned by

```
Collections.unmodifiableList(new LinkedList<>(Arrays.asList("f", "s")))
```

**Response 2**

**Prompt 3**

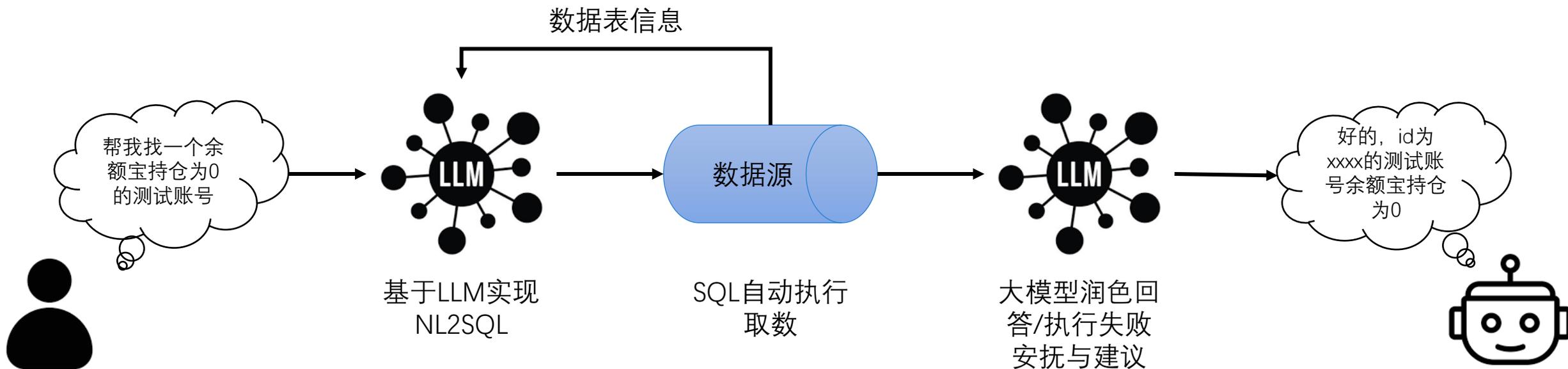
Previous context + response	Please provide a fixed version of the code with minimal changes.
-----------------------------	--

[1] Kang S, Yoon J, Yoo S. Large language models are few-shot testers: Exploring llm-based general bug reproduction[C]//2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE). IEEE, 2023: 2312-2323.

[2] Fakhoury S, Chakraborty S, Musuvathi M, et al. Towards generating functionally correct code edits from natural language issue descriptions[J]. arXiv preprint arXiv:2304.03816, 2023.

# ▶ 应用方式——作为 workflow 组件

大模型作为质量 workflow 当中的组件，为质量工具提供了丰富的泛化性和强大的生成能力，**提升了质量工具的能力上限**。以一个测试账号查询 workflow 为例：



大模型通过接受用户输入与数据表（表名、字段名称、字段描述、取值范围等）信息既可实现 NL2SQL，节省了小模型训练的成本

## **PART 03**

# **质量大模型在接口测试中的实践**

# ▶ 接口测试的特点与难题

专业性

对专家知识和经验要求高

场景化

需要结合特定业务逻辑

复杂性

人工成本和时间成本较高

# 质量大模型应用——校验点生成

输入

```
{
  "接口名": "interface_1",
  "入参": {
    "param_1": "xxx",
    "param_2": 1234,
    "param_3": true
  },
  "出参": {
    "status": 200,
    "result": {
      "success": true,
      "param_4": "xxx",
      "param_5": 1234,
    },
    "description": ""
  }
}
```



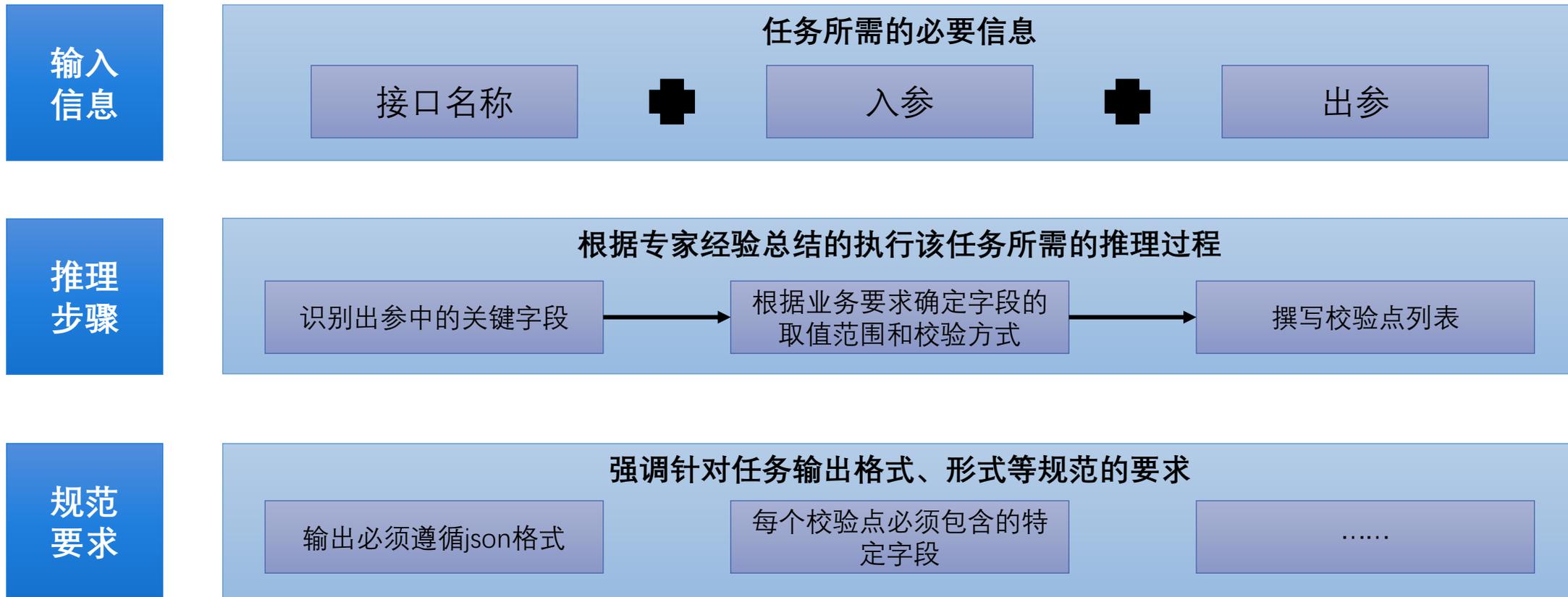
质量大模型

输出

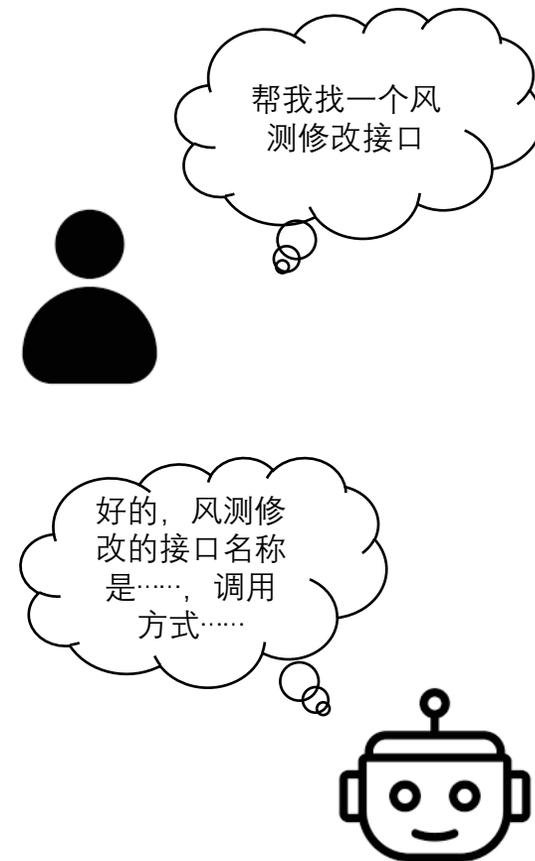
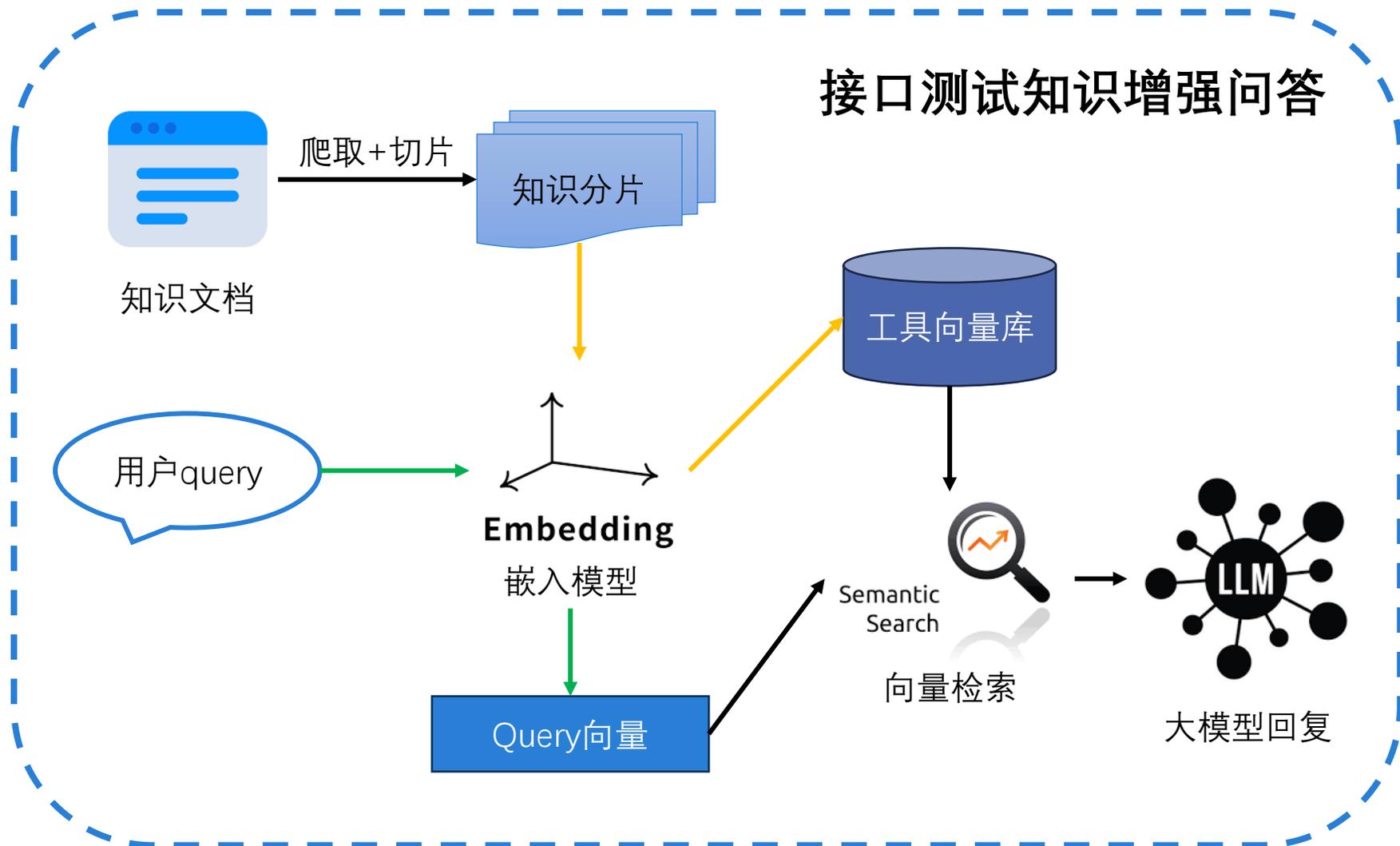
```
{
  "check_list": {
    {
      "key": "$.result.success",
      "value": true,
      "type": "boolean",
      "compareType": "equ",
      "desc": ""
    },
    {
      "key": "$.result.param_4",
      "value": "",
      "type": "string",
      "compareType": "notBlank",
      "desc": ""
    },
    {
      "key": "$.result.param_5",
      "value": 4321,
      "type": "int",
      "compareType": "equ",
      "desc": ""
    }
  }
}
```

# ▶ 质量大模型应用——校验点生成

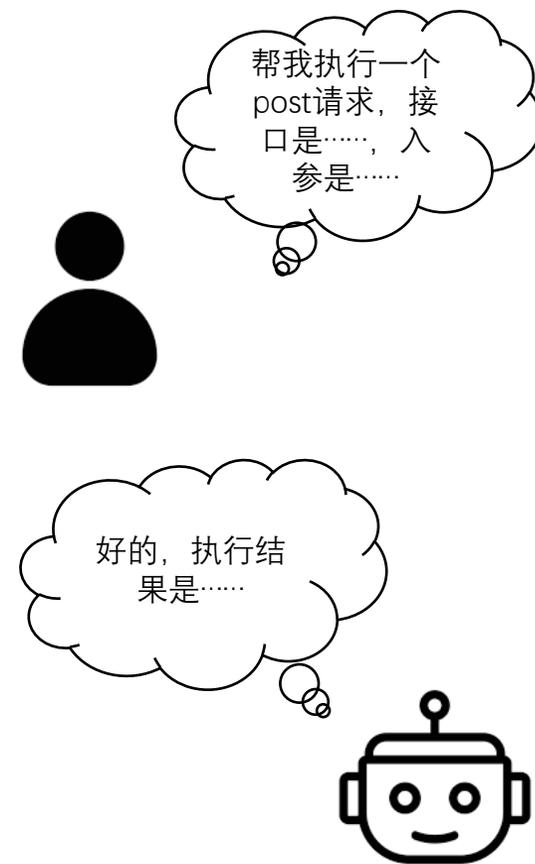
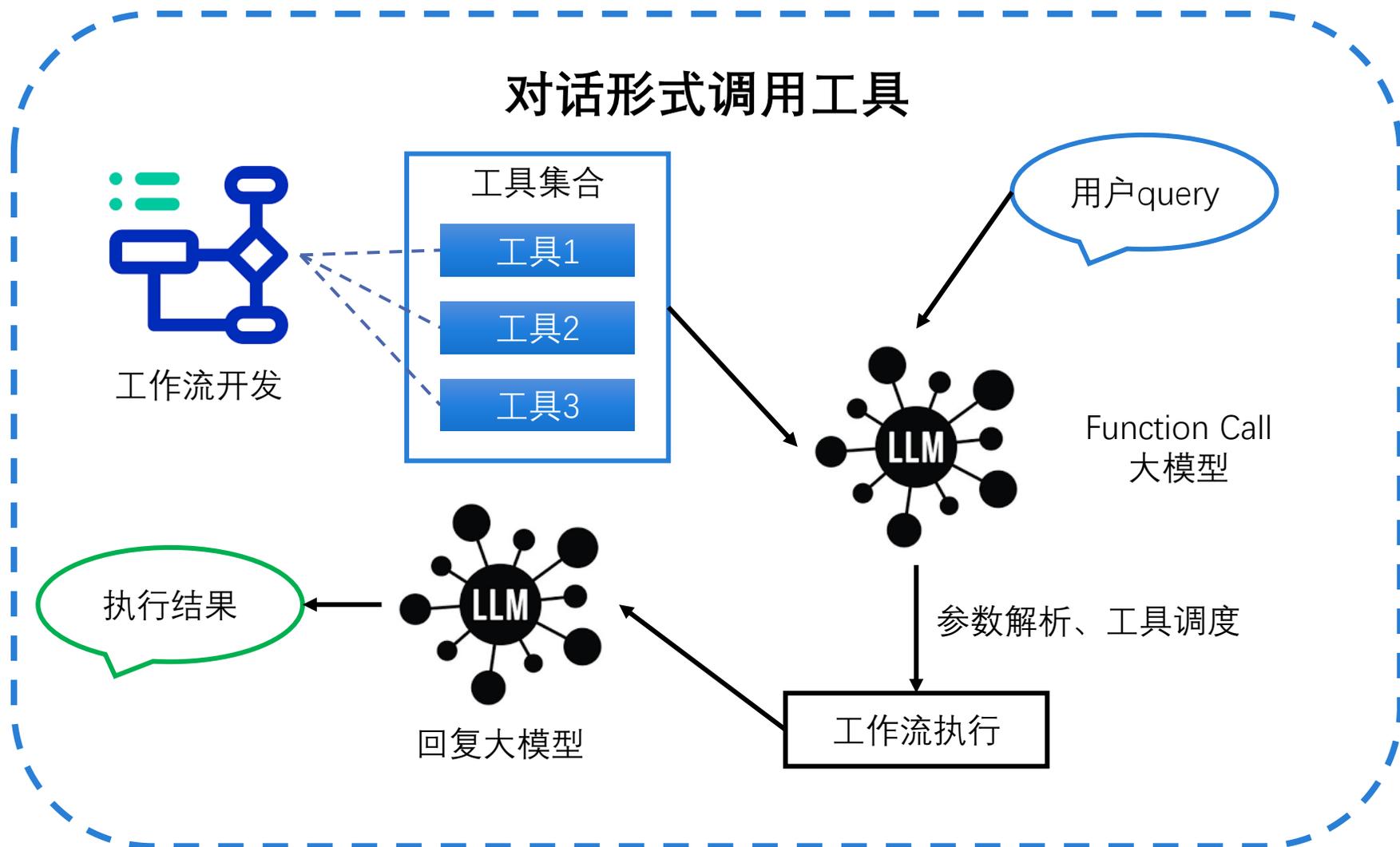
## Prompt组装



# 质量大模型应用——业务知识问答



# 质量大模型应用——工具执行



# PART 04

# 未来展望

# ▶ 总结

1

质量 + 大模型

产品内容上围绕**质量 + 大模型**的提效工作

2

垂类解决方案

产品设计上是一套**较通用的垂直领域的大模型解决方案**

3

大模型行业赋能框架

算法设计上是一套通用的**垂类领域大模型提效框架**

# ▶ 质量大模型技术趋势



- 浩如烟海的质量知识与业务经验
- 多样的质量任务
- 前置的问题发现
- 代码能力与问答能力的兼顾

# 科技生态圈峰会 + 深度研习



—1000+ 技术团队的选择



K+全球软件研发行业创新峰会

时间: 2024.06.21-22



K+思考周®研习社

时间: 2024.10.17-19



K+思考周®研习社

时间: 2024.11.10-12



K+峰会详情



AI+研发数字峰会

时间: 2024.05.17-18



AI+研发数字峰会

时间: 2024.08.16-17



AI+研发数字峰会

时间: 2024.11.08-09



AIDD峰会详情



# 2024 AI+研发数字峰会

AI+ Development Digital summit

深圳站 11/08-09

## AI 驱动研发变革 促进企业降本增效

### 2024深圳站-议题设置

AI+产品线	LLM驱动产品创新	LLM驱动需求与业务分析	AI驱动设计与用户体验
AI+开发线	AI 原生应用开发框架与技术	AI Agents在研发落地实践	LLM驱动编程与单测
AI+测试线	LLM驱动测试分析与设计	基于LLM生成测试脚本与数据	LLM和AI应用的评测
AI+工程线	AI+DevOps 与工具 (LLM 时代的平台工程)	大模型对齐与安全	端侧大模型与云端协同
AI+领域线	领域大模型 SFT 与优化	知识增强与数据智能	大厂专场

扫描右侧二维码  
查看更多会议详情



早鸟票限时抢购中 (截止到9月30日)

¥ 3680

早鸟票

¥ 2800

学生票

# THANKS

