

# 擎智运维大模型

擎创科技出品

2023.12

## Make Data Think

以AI激活运维数据智慧，助力客户数字化转型

# 01 大模型和智能运维

02 运维大模型的探索

03 擎智运维大模型实践

## 目录

## Contents



# AIGC 大模型成为超级热点

- 01 2022年末以来，大语言模型纷纷问世
- 02 国外，OpenAI、Meta、Google等分别推出了ChatGPT、Llama、Gemini产品
- 03 国内以百度、清华、阿里、华为等为代表相继发布了文心一言、ChatGLM、通义千问、盘古等模型

OpenAI

Bard

LLaMA  
by Meta

Meet  
Claude

Falcon LLM  
Another Open Source LLM. Use  
HuggingFace to load the model.

Baidu 百度

文心一言



通义大模型

腾讯混元  
大模型

智谱·AI

GLM-130B  
An Open Bilingual Pre-Trained Model

MOSS

01

## 问答/对话

大模型的应用场景有哪些？

正在搜索: 大模型的应用场景

正在为你生成答案...

大模型的应用场景非常广泛，例如在医药研发、卫星遥感、灾害评估、自然生态监测等场景有巨大价值<sup>1</sup>。云厂商使用大模型主要有2个层面的应用考虑：一是提高生产力，包括普适性 AI 应用，即使用大模型完成小模型的快速定制，另外就是快速的将大模型使用到现有产品中去，包括智能问答、服装设计等场景，在不断的使用大模型探索应用的边界<sup>2</sup>。

希望这些信息对你有所帮助。

了解详细信息: [1. zhuanlan.zhihu.com](#) [2. zhuanlan.zhihu.com](#) [3. zhihu.com](#) +3 更多 1 共 20

新主题

有问题尽管问我...

0/2000

02

## 代码生成

```
sentiments.ts write_sql.go parse_expenses.py addresses.rb
1 import datetime
2
3 def parse_expenses(expenses_string):
4     """Parse the list of expenses and return the list of triples (date, value, currency).
5     Ignore lines starting with #.
6     Parse the date using datetime.
7     Example expenses_string:
8     2016-01-02 -34.01 USD
9     2016-01-03 2.59 DKK
10    2016-01-03 -2.72 EUR
11    """
12    expenses = []
13    for line in expenses_string.splitlines():
14        if line.startswith("#"):
15            continue
16        date, value, currency = line.split(" ")
17        expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),
18                        float(value),
19                        currency))
20    return expenses
```

Copilot

Replay

03

## 内容生成

<p><b>Create original content that ranks for SEO</b></p> <p>Generate educational blog articles that are keyword-rich and plagiarism-free.</p>	<p><b>Boost ad conversions with better copy</b></p> <p>Easily write and test more copy variations to increase sales and improve ROAS.</p>	<p><b>Finish your first draft 10X faster</b></p> <p>Tell Jasper about what you want and then watch the AI write paragraphs in seconds.</p>
<p><b>End writer's block with ideas from a robot</b></p> <p>Stuck staring at a blank page? Relax and let Jasper write creative copy for you.</p>	<p><b>Breakthrough the language barrier</b></p> <p>No matter your native tongue, write creatively and clearly in 25+ languages.</p>	<p><b>Scale up your content marketing fast</b></p> <p>Repurpose existing content and generate new content without hiring junior writers.</p>

04

## 多模态生成



CR

请解释下面这条告警：Nov 8 09:39:24 host1 kernel: Error:Mpx:Path Bus 1 Tgt 0 Lun 0 to CK200083800438 is dead



这条告警是来自Linux内核的一条错误消息，它描述了一个与存储设备之间通信的问题。让我解释其中的一些关键部分：

1. `Nov 8 09:39:24`：这是告警消息的时间戳，表示它发生的日期和时间。在这个例子中，告警发生在11月8日的09:39:24。
2. `host1`：这是告警消息中标识主机或服务器的名称，表示告警来自名为"host1"的计算机。
3. `kernel`：这表明告警消息来自Linux内核，通常是与操作系统的核心功能相关的问题。
4. `Error:Mpx:Path Bus 1 Tgt 0 Lun 0`：这部分描述了存储路径的关键信息。在这里，它指出了存储路径（可能是一种连接到主机的存储设备），其中包括以下信息：
  - `Bus 1`：总线编号，通常用于标识不同的存储控制器或总线。
  - `Tgt 0`：目标 (Target) 编号，表示在总线上的目标设备。
  - `Lun 0`：逻辑单元编号，表示目标设备上的不同逻辑单元（通常是硬盘或分区）。
5. `to CK200083800438`：这部分可能是关于存储设备本身的标识信息，它指出了与存储路径相关联的设备，可能是设备的序列号或其他标识符。
6. `is dead`：这是告警的主要内容，它指示存储路径或设备与主机之间的通信出现了问题，可能意味着该路径或设备已经无法正常工作或无法访问。

## 大语言模型与运维相关的能力

### 自然语言处理能力

大语言模型具备强大的自然语言处理能力，可以理解和分析文本数据。日志和告警通常以自然语言的形式呈现，大语言模型能够有效地识别和理解其中的实体和语义

### 运维领域知识

大语言模型具备基础的运维领域知识，能正确理解运维对象、故障机理等概念

### 持续学习和改进能力

大语言模型可通过提示词工程、参数微调、外挂运维知识库/知识图谱实现持续学习和改进

### 推理能力

大语言模型能够理解语法、语义和语境，并基于运维领域知识和上下文依赖完成逻辑推理和推断

### 自然语言生成能力

大语言模型可以生成高质量的自然语言文本，支持多轮对话，优化运维人机交互模式

### 代码生成能力

大语言模型可以生成自动化任务和脚本的代码，以执行各种运维任务，如配置管理、日志分析、性能监控等

大模型具备基本的运维知识，针对告警数据能够给出较为合理的分析



- » 具备基本的运维知识，针对大部分告警能够给出合理的分析
- » 具有多轮对话能力



- » 缺乏特定的告警知识（尤其是私域知识）
- » 分析的结果有时较为表面，无法挖掘告警之间深入的关联性
- » 问答过程有长度限制
- » 模型的回答不稳定
- » 无法在企业内网部署

尴尬之处：GPT和开源大模型存在较大差距



基于开源大模型，通过训练/微调、检索增强、提示词等方式，构建运维大模型

## 01 本地化部署

1. 私域数据的安全
2. 开源大模型+训练或微调

## 02 集成现有工具

1. LLM+现有的算法/工具/知识库

## 03 不能为了LLM而LLM

1. 提升运维效率
2. 弥补现有运维方法的不足
3. 解决现有运维过程中的痛点

## 04 充分发挥LLM的长处

1. 语言生成能力
2. 对话能力
3. 一定的推理能力

01 大模型和智能运维

02 运维大模型的探索

03 擎智运维大模型实践

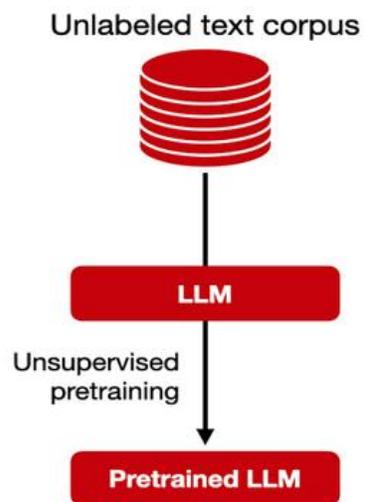
目录

Contents



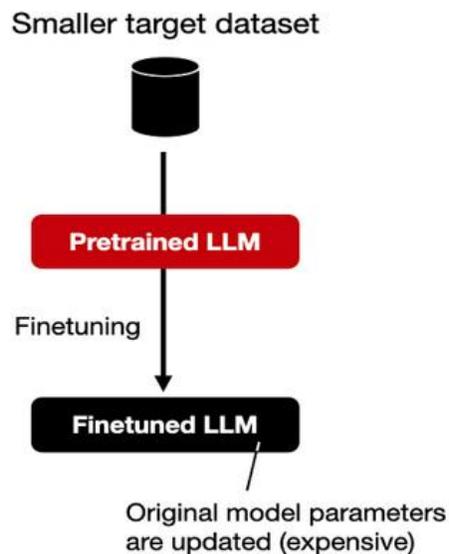
# 大模型应用方式1：修改大模型的参数

## Step 1: Pretraining



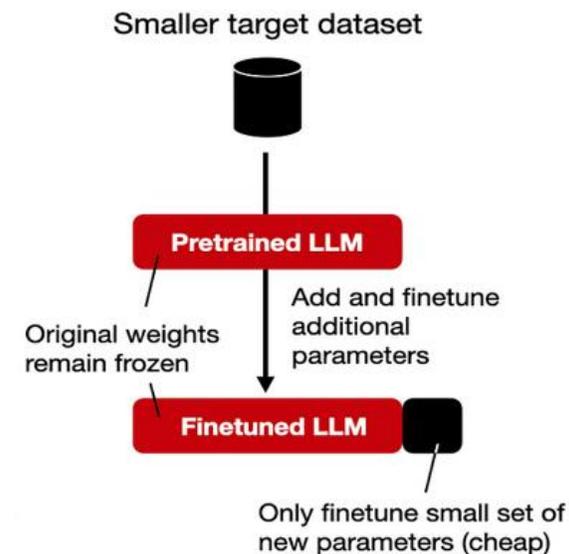
预训练

## Step 2a: Conventional finetuning



全参数微调

## Step 2b: Parameter-efficient finetuning



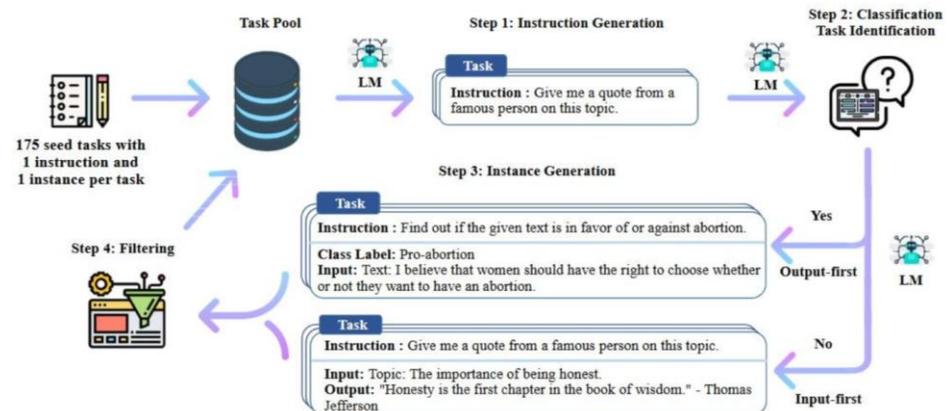
部分参数微调

# 大模型应用方式2：不修改大模型的参数

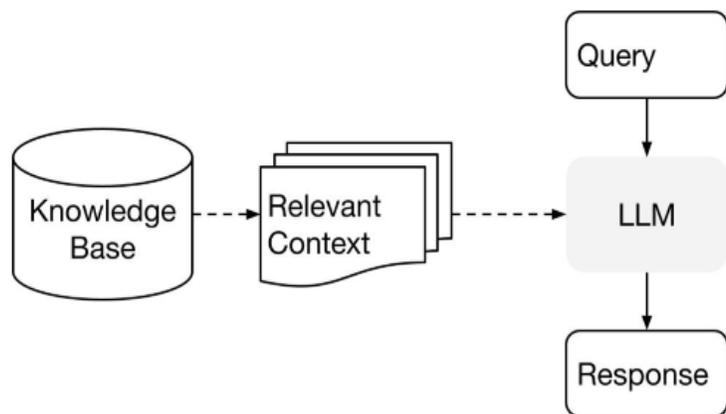
## 01 提示词 (Prompt)

1 Translate English to French: ← *task description*  
2 sea otter => loutre de mer ← *examples*  
3 peppermint => menthe poivrée ← *examples*  
4 plush girafe => girafe peluche ← *examples*  
5 cheese => ..... ← *prompt*

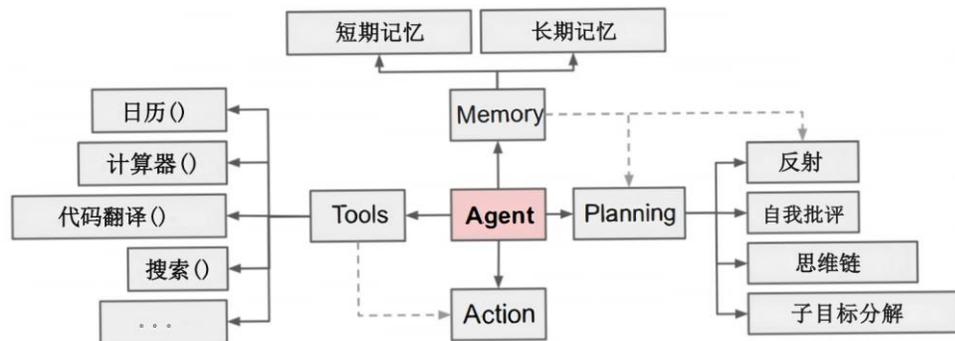
## 02 指令 (Instruction)



## 03 检索增强 (RAG)



## 04 Agent & Tools



## 模型微调

针对自然语言转查询、告警根因分析等任务，采用全参数微调、部分参数微调等方式，训练面向特定任务的大模型

## 运维知识

基于公域和私域运维知识库，通过检索增强等方式，丰富大模型的运维知识，结合大模型的语言生成能力，使得大模型理解日志/告警/事件等

## 推理能力

通过提示词工程、思维链、微调等方式，增强大模型在运维领域的推理能力，从而赋予大模型告警根因分析等能力

## 运维工具

通过模型训练/微调、Agent、提示词等方式，使得大模型能够和智能运维算法、数据库、Python代码等进行自动结合，从而提升大模型的能力

## 日志/告警分析

痛点：需人工干预、新日志无法处理  
目标：无需设置参数、解析单条日志

## 故障报告自动生成

痛点：编辑过程冗长、综合能力弱  
目标：过程自动化、智能化



## 告警理解

痛点：日志/告警内容难以理解  
目标：利用检索增强及时，提升日志/告警的可读性，自动生成处理方案

## 日志/告警/事件探查

痛点：查询手段固化，探索能力弱  
目的：提供灵活的多类型数据探查能力

## 告警分析

痛点：告警风暴，根因定位难  
目标：实现告警数据探查和根因定位

01 大模型和智能运维

02 运维大模型的探索

03 擎智运维大模型实践

目录

Contents



# 场景一：基于知识增强的告警理解

## 请分析以下告警的故障根因

路径：/alert\_NETCOOL.log, ORA-00060. More info in file  
/.../trace/NETCOOL\_ora\_29530.trc.temp

## 已知信息

ORA-00060: deadlock detected while waiting for resource  
Cause: Transactions deadlocked one another while waiting for resources.  
Action: Look at the trace file to see the transactions and resources involved. Retry if necessary.

## 开源大模型

缺乏相应的知识

故障根因：数据库中出现内存泄漏。

告警中提到了在/alert\_NETCOOL.log 文件中看到了 **ORA-00060** 的告警，同时还有文件路径信息。这说明在 alert\_NETCOOL.log 文件中记录的是数据库中某个表在内存中的使用情况，而该表所在的 tablespace TEMP 表中的 temp segment 容量不足，造成了内存泄漏。内存泄漏会导致数据库性能下降，甚至可能出现死锁等故障。因此，这很可能是导致告警的根本原因。

## 开源大模型

故障根因： Transactions deadlocked

解释：在该日志中，检测到有一些 Transactions 在等待资源时已经 deadlocked。这意味着某些 Transaction 正在等待其他 Transaction 提供资源，导致系统出现了 deadlock 故障。

解决方案：查看 trace 文件，确定 Transactions 和资源的具体信息，并尝试重新运行 Transactions。如果有必要，可以重复执行 deadlock 故障排除步骤。

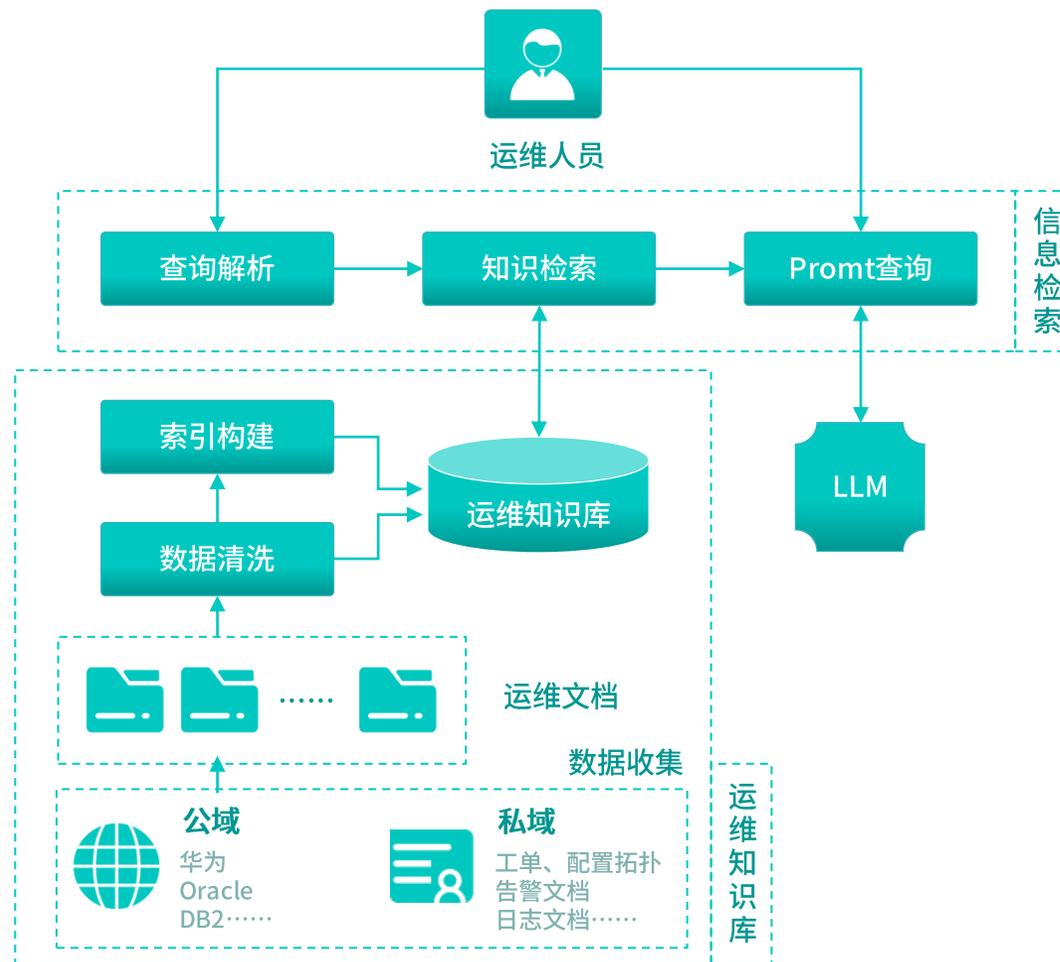
# 场景一：基于知识增强的告警理解

## 运维知识库

- 包含服务器、存储、数据库、中间件、操作系统、虚拟化与容器、应用等公域知识
- 支持字符串和向量检索的融合
- 支持长文档的内容理解和关联

## 构建过程

1. 收集各类运维数据
2. 将运维数据清洗、整理成文档形式
3. 将数据文档以向量的形式进行存储并构建索引



# 场景二：日志/告警解析

日志1: IP: 10.142.212.10 , Port: 80 - Connection open  
日志2: IP: 10.142.212.11 , Port: 22 - Connection open



模板: IP: \$(IP地址) , Port: \$(端口) - Connection open

告警1: 通讯节点1(node1):在2019-01-10 05:26:51  
时出现交易异常:超过367秒无交易上送

告警2: 通讯节点2(host2):在2019-01-10 05:49:41  
时出现交易异常:超过361秒无交易上送



模板: 通讯节点\$NUM(主机名):在\$datetime  
时出现交易异常:超过\$NUM秒无交易上送

	优点	缺点
大模型解析	<ul style="list-style-type: none"><li>• 强大自然语言理解能力,能更加准确地理解文本结构(正确分词,并抽取命名实体信息)</li><li>• 基于运维常识,从语义层面判断变量和常量,对少量告警也可正确解析</li><li>• 通过提示词工程或参数微调可以能够应对更多场景,具有高度可扩展性</li><li>• 需要较少的人工干预和参数设置</li></ul>	<ul style="list-style-type: none"><li>• 对计算资源和硬件有要求(需GPU实现推理、参数微调)</li><li>• 实时聚类性能不足风险</li></ul>
传统解析算法(Drain、Spell、IPLOM)	<ul style="list-style-type: none"><li>• 基于统计特征,解释性强,效果有保障</li><li>• 多种成熟方案,支持离线、在线、混合多种模式</li><li>• 传统计算资源(无需GPU),性能有保障</li></ul>	<ul style="list-style-type: none"><li>• 需手工引入运维领域知识。如:定义正则识别实体</li><li>• 仅基于统计特征,数据量不足时,无法正确识别变量和常量,模版不稳定</li><li>• 较多人工干预和参数设置。如:设置相似度阈值</li></ul>

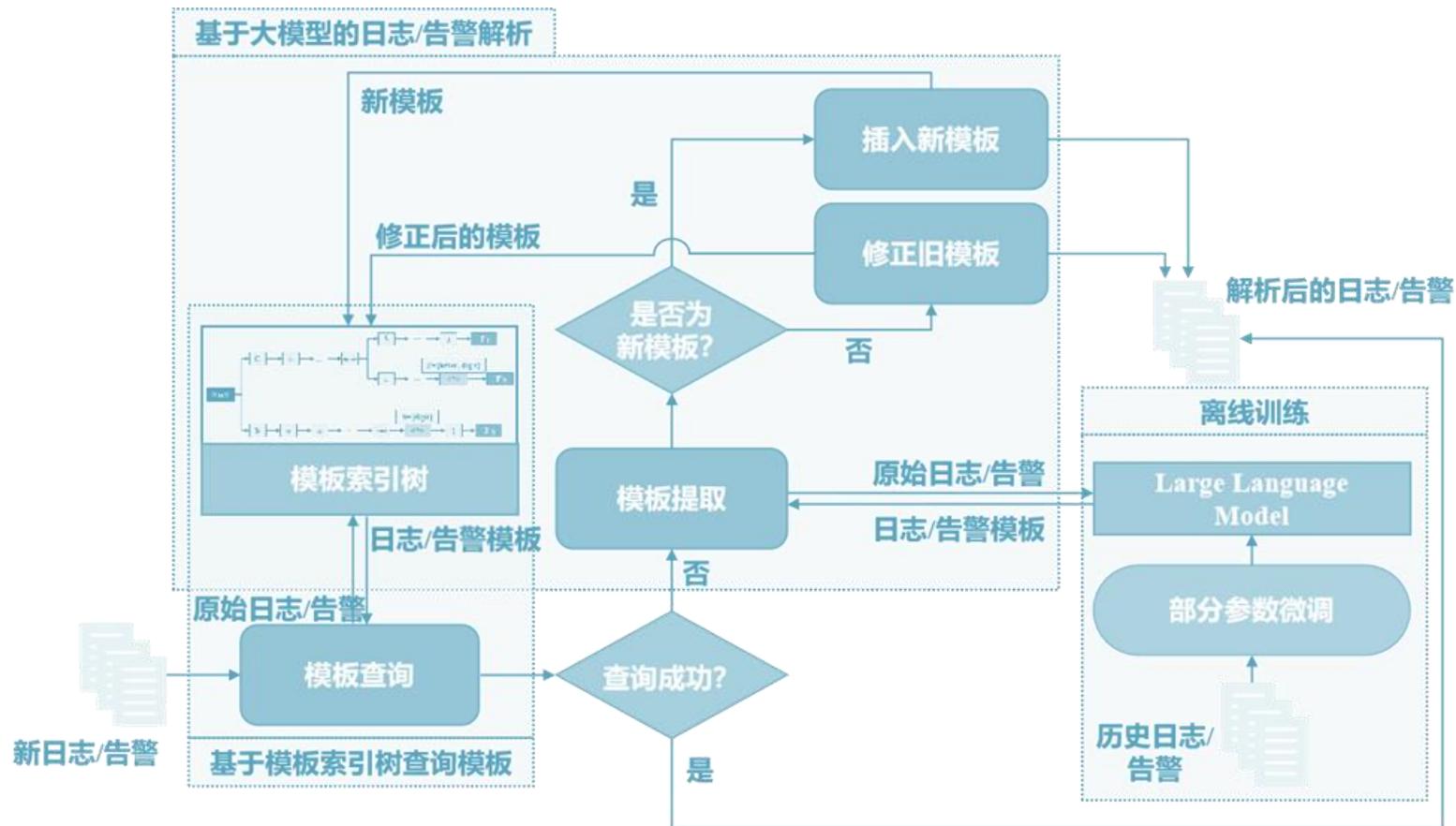
# 场景二：日志/告警解析

## 通过微调技术，让大模型能够快速、准确的对日志/告警提取模板

- 能够处理单条日志（新日志的解析）

## 通过索引结构，提升大模型的解析效率

- 只需使用大模型解析新出现的日志
- 重复日志基于索引自动进行解析





自然语言

告警数据



日志数据



CMDB数据



事件数据



指标数据



全参数微调得到**查询大模型**

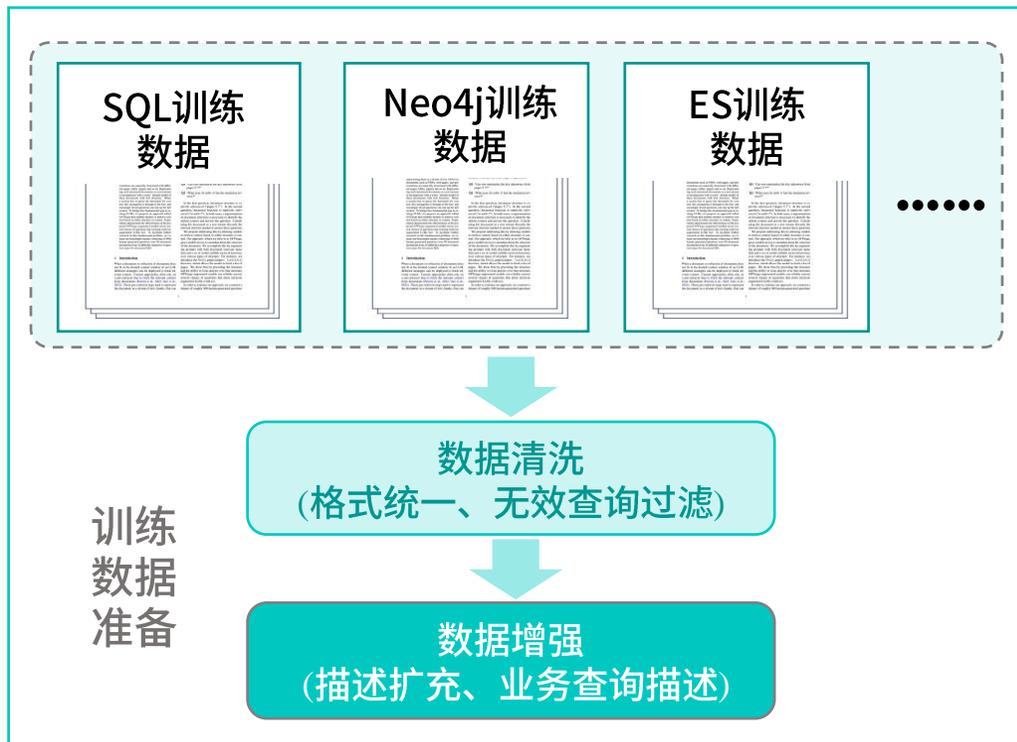
目前支持的查询语言：

- SQL
- Elasticsearch
- Cypher

仍在不断的添加新的查询语言

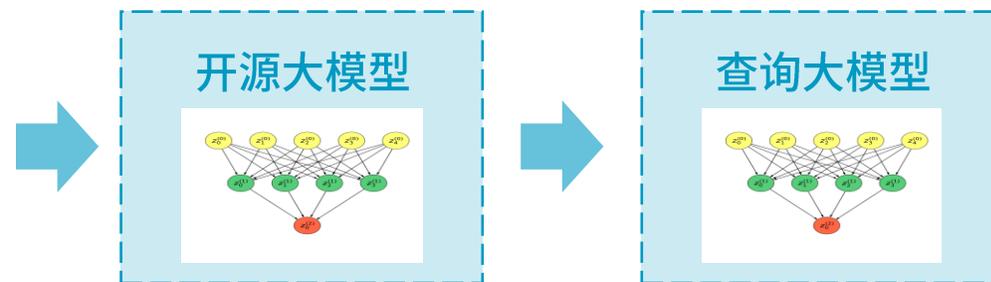
## 核心是高质量的大规模训练数据

- 融合领域和业务相关的查询
- 融合中文+英文查询描述



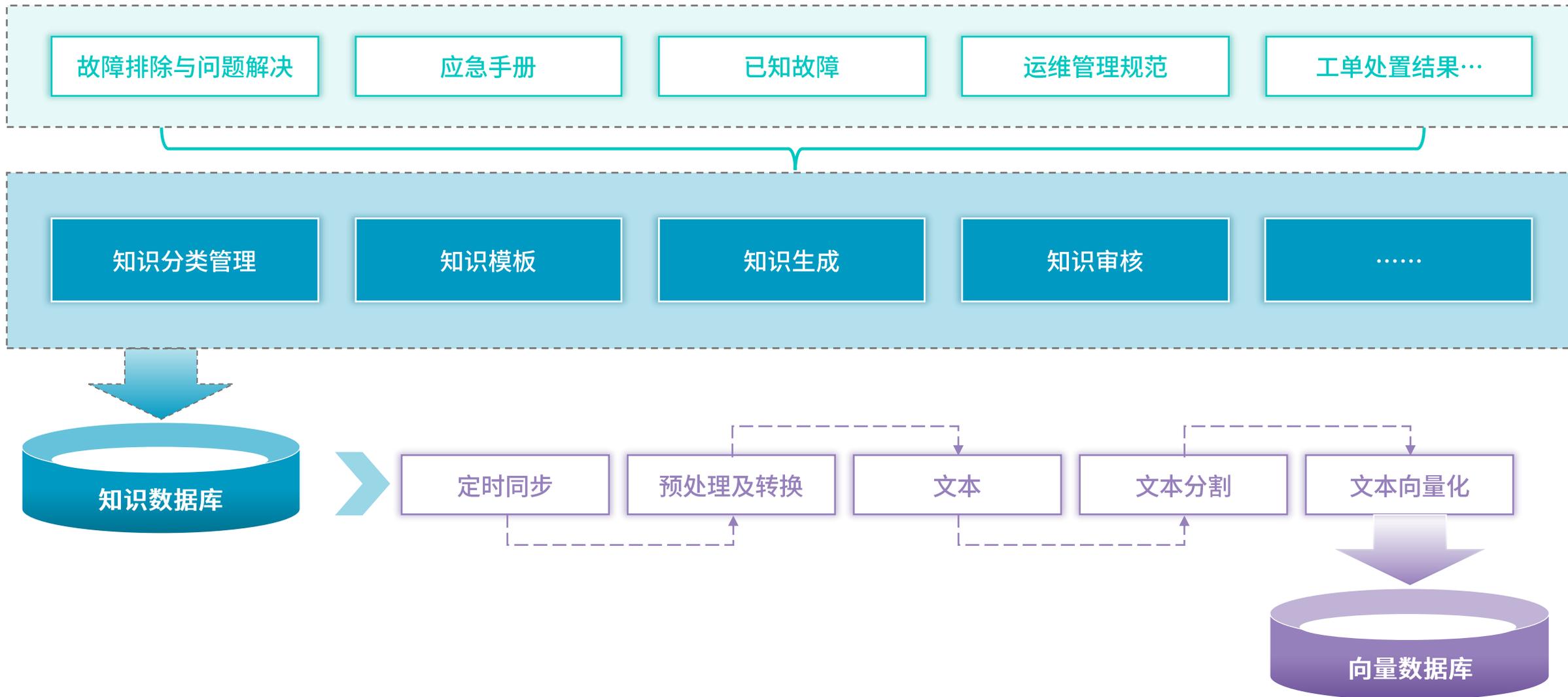
## 一些有趣的发现:

- ✓ 大模型具有中文描述的泛化能力
- ✓ 业务相关训练数据的必要性
- ✓ 大模型具有良好的自然语言到数据库Schema的映射能力



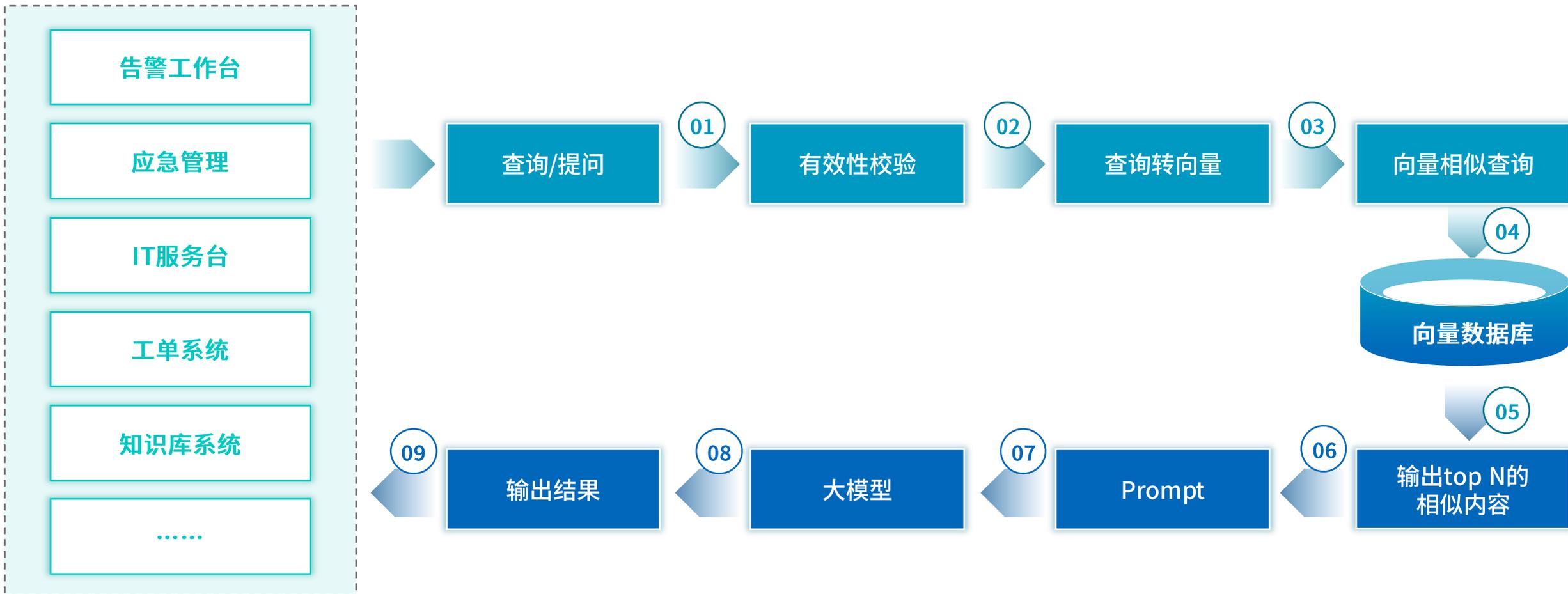
# 场景四：告警分析及根因定位

## 运维知识体系构建



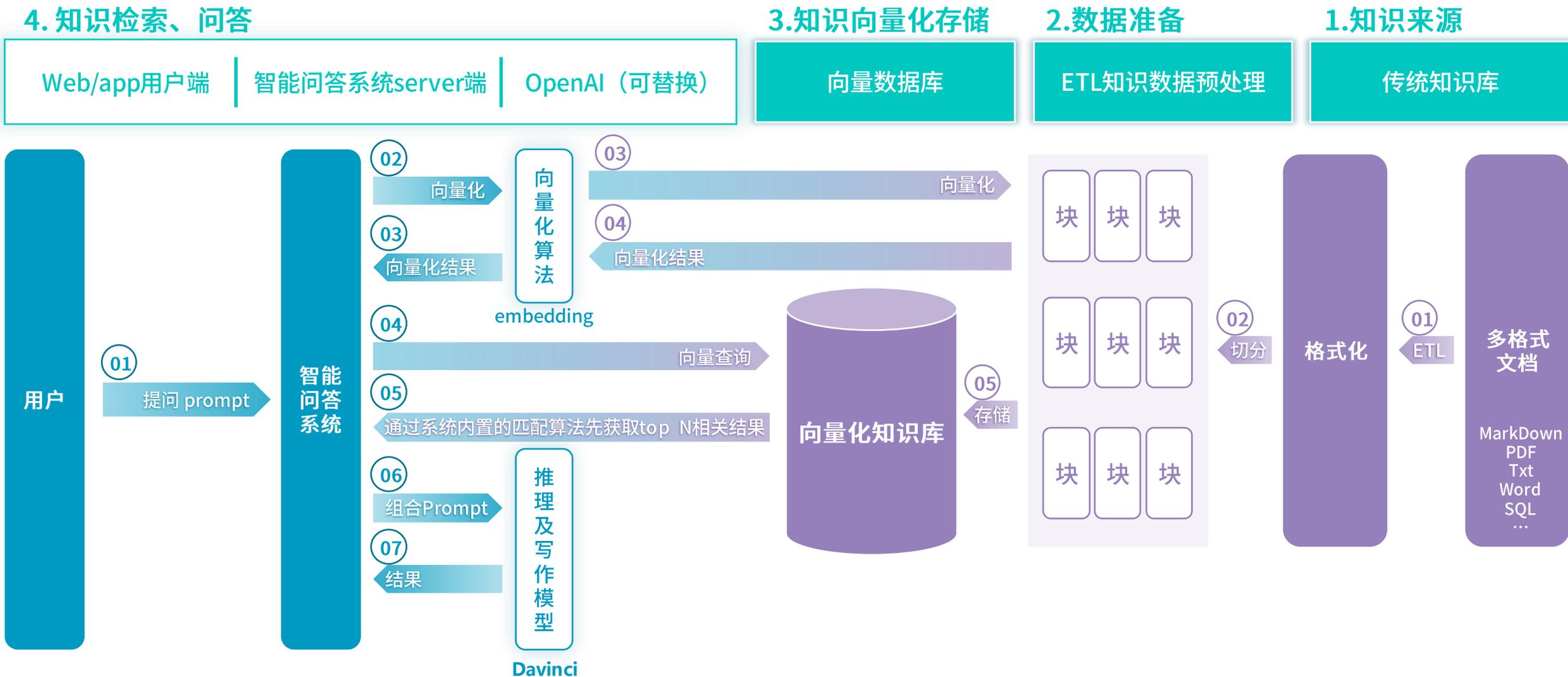
# 场景四：告警分析及根因定位

## 运维知识体系应用



# 场景四：告警分析及根因定位

## 系统集成边界



## 01 • 自然语言转Neo4j查询

基于自然语言进行告警源的关联分析

## 02 • 基于大模型的根因定位

CMDB的告警关联能力  
融合排障思路的思维链 (COT)



运维大模型仍然需要高质量的数据：告警、CMDB、。。。

基于大模型能力，认真思考如何利用大模型提升运维效率

Q1：哪些任务适合大模型，哪些任务适合现有算法和工具（效果、效率、性价比）

Q2：大模型带来哪些新的运维场景

Q3：动态的思维方法来思考这些问题（大模型的演进方向）

新一代的运维体系架构

# THANKS



[www.eoitek.com](http://www.eoitek.com)



[info@eoitek.com](mailto:info@eoitek.com)



4008 215 724

## Make Data Think

以AI激活运维数据智慧，助力客户数字化转型