# 大语言模型在信息系统智能运维中的应用

马宇航
盐城工学院，江苏 盐城 224051

[摘要] 信息系统规模扩大与架构复杂化增加了日志分析、告警处置和故障定位的难度。大语言模型具有语义理解、知识组织和任务生成能力，可用于日志解释、告警归并、根因分析、知识问答及脚本辅助。文章分析其技术基础与应用方式，构建数据、知识、模型和工具协同的运维架构，并讨论模型幻觉、数据安全、执行权限与推理成本等约束。研究认为，大语言模型适合作为智能运维的语义中枢和决策辅助工具，生产部署宜采用检索增强、规则校验、权限隔离和人工确认机制。

[关键词] 大语言模型；信息系统；智能运维；故障诊断；人机协同
DOI：10.64635/ja.2026.1436 中图分类号：TP18 文献标识码：A

# Application of Large Language Models in Intelligent Operation and Maintenance of Information Systems

Ma Yuhang
Yancheng Institute of Technology, Yancheng, Jiangsu 224051, China

**Abstract:** The expansion of information system scale and the increasing complexity of system architecture have made log analysis, alarm handling, and fault localization more difficult. Large language models have capabilities in semantic understanding, knowledge organization, and task generation, and can be applied to log interpretation, alarm aggregation, root cause analysis, knowledge-based question answering, and script assistance. This paper analyzes their technical basis and application modes, constructs an operation and maintenance architecture integrating data, knowledge, models, and tools, and discusses constraints such as model hallucination, data security, execution permissions, and inference costs. The study suggests that large language models are suitable as semantic hubs and decision-support tools for intelligent operation and maintenance. In production deployment, mechanisms such as retrieval augmentation, rule verification, permission isolation, and human confirmation should be adopted.

**Keywords:** large language model; information system; intelligent operation and maintenance; fault diagnosis; human-machine collaboration

## 引言

云计算、微服务、容器和分布式数据库改变了信息系统的运行形态。业务请求可能跨越多个服务节点，局部配置错误、资源争用或网络抖动均可能引发连锁异常。监控平台持续产生指标、日志、链路和告警数据，传统阈值规则擅长识别预设异常，却较难处理跨来源语义关联、未知故障解释和自然语言知识调用。深度学习提高了部分异常检测任务的自动化水平，但专用模型通常依赖稳定的数据分布和任务标签，模型输出也不易直接转化为运维人员可理解的处置依据。

大语言模型能够统一处理自然语言日志、故障工单、操作手册和脚本代码，为运维数据理解提供新的技术路径。公开研究已将其用于日志解析、异常检测、故障诊断和日志摘要。裴丹等提出，大语言模型时代的智能运维架构可由工具智能体、岗位智能体和运维人员共同构成人机协同系统，大模型负责语言交互、知识调用和工具编排[1]。因而，大语言模型在智能运维中的价值不宜仅以文本生成能力判断，还需分析其数据基础、工具调用方式、执行边界和评测条件。

## 1 大语言模型赋能智能运维的技术基础

### 1.1 运维数据的语义化处理

日志记录包含时间、组件、事件、参数和状态等信息，原始文本中还混有标识符、路径、端口及动态变量。传统日志分析通常先提取模板，再完成聚类或分类。大语言模型可以利用预训练形成的语言表示识别事件含义，并结合前后事件解释异常序列。Loghub 收录 19 个来自分布式系统、超级计算机、操作系统、移动系统和服务器应用等场景的真实日志数据集，为日志解析与异常检测研究提供了统一数据基础[3]。面对复杂故障，罗子秋等构建了结合运维知识库、工具调用和蒙特卡洛树搜索的根因分析系统，通过多轮“运维动作—环境反馈”逐步缩小故障范围[2]。这类方法

表明，大模型的作用已经从单次文本判断扩展到受工具和环境反馈约束的连续诊断。

运维数据并不限于日志。指标能够描述资源变化，调用链能够还原请求路径，配置文件能够反映系统预期状态，工单与操作手册保留历史经验。大语言模型可把这些异构内容组织为面向任务的证据集合，进而回答“异常发生在哪个组件”“哪些变更与故障时间接近”等问题。其优势集中于语义关联和解释生成，精确数值计算、实时流式检测仍宜交由监控引擎和专用算法完成。

### 1.2 模型能力与传统智能运维的互补

传统智能运维模型在固定任务上具有计算速度快、输出稳定和成本可控等特点。大语言模型具备少样本学习、自然语言交互与跨任务迁移能力，可降低新故障类型缺少标签时的分析门槛。二者适合形成“专用模型发现异常，大语言模型解释异常，规则系统约束操作”的组合。监控算法先筛选高风险对象，大语言模型调用知识库和历史案例形成诊断建议，规则引擎再检查命令参数、资源范围与审批条件。该组合能够减少大模型直接处理全部原始数据造成的时延和费用，也能保留原有监控系统的确定性能力。

## 2 大语言模型在信息系统运维中的主要应用

### 2.1 日志理解与异常识别

大语言模型可以完成日志模板抽取、事件摘要、异常分类和异常原因解释。面对格式变化或新版本日志，模型可依据语义识别同类事件，降低模板漂移对检测效果的影响。生产环境可先按照服务、会话或时间窗聚合日志，再由模型提取事件链和关键证据。异常结论需要同时保留原始日志位置、时间范围和推理依据，便于运维人员复核。对于高频日志，轻量模型或规则库可承担在线筛选，大模型负责疑难样本复核，避免逐条推理造成资源浪费。

### 2.2 告警归并与故障诊断

同一故障可能触发主机、网络、数据库和应用层多条告警。大语言模型可结合组件依赖、告警时间和变更记录生成事件摘要，识别重复告警与派生告警，并给出候选根因。公开的 AIOpsLab 研究构建 48 个问题，对多类智能体开展故障检测、定位、根因分析和缓解测试[4]。部分结果见表 1。

表 1 AIOpsLab 部分智能体任务准确率

| 智能体 | 故障检测/% | 故障定位 Acc.@1/% | 根因分析/% | 故障缓解/% |
|---|---|---|---|---|
| GPT-4-W-SHELL | 69.23 | 61.54 | 40.90 | 27.27 |
| REACT | 76.92 | 53.85 | 45.45 | 36.36 |
| FLASH | 100.00 | 46.15 | 36.36 | 54.55 |

表 1 显示，检测准确率较高并不代表根因分析和故障缓解同样可靠。运维任务越接近生产操作，涉及的状态判断、工具调用和执行约束越多。部署方案需要按检测、定位、分析和处置分别设置评价标准，不能用单项测试结果替代全流程验证。

### 2.3 运维知识问答与操作辅助

大语言模型可将制度文档、系统说明、历史工单和专家经验接入检索系统，为值班人员提供自然语言查询入口。当用户提出故障现象时，系统先检索与组件版本、错误码和业务场景相符的材料，再生成带来源的回答。模型还可根据操作模板生成巡检命令、查询语句和脚本草案。涉及重启、扩缩容、配置修改和数据删除的操作，需要经过参数校验、影响分析与人工批准。较稳妥的定位是把模型生成内容作为候选方案，而非直接作为生产指令。

## 3 大语言模型智能运维架构

### 3.1 数据、知识与模型协同

智能运维架构可划分为数据接入、知识组织、模型推理和安全执行四层。数据接入层汇集日志、指标、链路、告警、配置及工单，并完成脱敏、时间对齐和对象标识统一。知识组织层保存架构关系、运维手册、故障案例、标准脚本与变更记录，通过向量检索和结构化查询向模型提供证据。模型推理层承担意图识别、证据归纳、原因分析和方案生成。安全执行层连接监控平台、工单系统和自动化工具，通过白名单、最小权限、审批流程及回滚机制限制操作范围。

该架构的核心是让模型基于可追溯材料回答问题。检索结果需要包含文档版本、适用系统和更新时间，配置状态与监控数据则需标明采集时点。模型输出同步展示证据来源、置信信息和未确认事项，能够降低过期知识或信息缺失造成的错误判断。

### 3.2 运维任务闭环

一次完整任务包括异常触发、数据取证、候选诊断、方案校验、人工确认、工具执行和结果回采。模型根据告警调用日志与指标查询工具，形成根因候选及证据排序。规则引擎检查操作对象、命令格式和风险等级，高风险任务转交专业人员确认。执行平台记录模型版本、输入证据、生成建议、审批人员、实际

命令和执行结果。处置结束后，结果回写案例库，用于修订提示模板、知识条目和检测规则。闭环记录既服务于模型改进，也为责任追踪和故障复盘提供依据。

## 4 应用风险与现实约束

### 4.1 模型幻觉与诊断偏差

运维日志包含大量领域缩写和隐含规则，模型可能生成不存在的组件、命令或故障原因。生产应用可采用规则提取、样本验证、证据引用、结构化输出和拒答机制。当证据不足或多种根因接近时，系统应输出候选范围及待补充数据，不宜强行给出唯一结论。规则库还需记录适用系统和版本，避免旧规则被错误迁移到新环境。

### 4.2 数据安全与权限风险

日志、配置和工单可能包含账号、地址、业务标识与敏感操作记录。模型接收数据前需要完成分类分级、字段脱敏和访问控制。外部模型服务还涉及数据传输、留存范围与调用审计。高敏感系统可采用本地部署或受控推理环境，并将知识库权限与用户身份绑定。工具调用采用临时凭证和最小权限，模型只能访问当前任务所需资源。任何改变生产状态的动作均需留下完整审计记录。

### 4.3 成本、时效与系统适配

长日志和多轮工具调用会增加推理时延与计算成本。模型输入过长还可能淹没关键事件，降低诊断稳定性。工程实现可利用日志模板化、事件聚合、分层检索和缓存减少无效信息，并根据任务复杂度选择不同规模模型。实时检测由流式计算和专用模型承担，复杂诊断再调用大语言模型。系统升级、接口变化和知识过期也会影响工具调用质量，因而需要持续维护接口规范、知识索引与回归测试集。

## 5 大语言模型智能运维的实施路径

### 5.1 分层部署与人机协同

落地过程可从低风险、高频任务起步。第一阶段用于日志摘要、知识检索和工单生成，模型不接触生产执行权限。第二阶段增加告警关联、根因候选和脚本草案，由运维人员审核。第三阶段在成熟场景中开放受限工具调用，自动执行只覆盖可回滚、影响范围明确且规则稳定的操作。核心数据库、身份系统和关键业务变更继续保留人工批准。分层部署能够通过真实反馈逐步确认能力边界，减少一次性改造带来的运行风险。

### 5.2 评测体系与持续治理

评测体系需要覆盖正确性、效率、安全性和可追溯性。异常检测关注准确率、召回率和误报率，根因分析关注首位命中率及前三位命中率，处置任务关注平均恢复时间、操作成功率、回滚率和人工接管率。知识问答还需评价来源命中率与回答一致性。测试集应包含正常状态、已知故障、未知故障、配置变化和对抗性输入，并按照系统版本持续更新。模型、提示模板、知识库和工具接口的变更均需经过离线回放、仿真验证和小范围试运行。治理责任由模型技术人员、平台运维人员、安全管理人员和业务负责人共同承担，形成明确的权限边界与问题追踪机制。

## 6 结语

大语言模型为信息系统智能运维提供了语义理解、知识调用和任务编排能力，能够改善日志解释、告警归并、故障诊断及运维知识服务。公开基准同时表明，模型在检测、定位、根因分析和故障缓解阶段的表现存在明显差异，较强的语言生成能力不能直接等同于可靠的生产操作能力。实际建设宜保留传统监控与专用算法的实时检测优势，以检索增强和规则校验提高模型输出的证据性，通过权限隔离、人工确认和结果回采建立安全闭环。大语言模型的合理角色是连接数据、知识、人员与工具的运维语义中枢，其应用深度由任务风险、证据质量和系统治理能力共同限定。

**[ 参考文献 ]**

[1] 裴丹，张圣林，孙永谦，等. 大语言模型时代的智能运维[J]. 中兴通讯技术，2024,30(2):56-62.

[2] 罗子秋，苗元凯，李丹. 基于大语言模型蒙特卡洛树搜索的智算网络故障根因分析系统[J]. 中兴通讯技术，2025,31(2):21-30.

[3] ZHU J, HE S, HE P, et al. Loghub: A Large Collection of System Log Datasets for AI-driven Log Analytics[C]//2023 IEEE 34th International Symposium on Software Reliability Engineering. Piscataway: IEEE, 2023: 355-366.

[4] CHEN Y, SHETTY M, SOMASHEKAR G, et al. AIOpsLab: A Holistic Framework to Evaluate AI Agents for Enabling Autonomous Clouds[EB/OL]. arXiv:2501.06706, 2025.