Available Online: 26 March 2026 DOI: 10.54254/2977-3903/2026.32426

Application research on intelligent probing based on large-scale AI models in internet fault handling

Hongli Gao[1*], Xielina Abdurékeman[1], Lingjuan Che[1], Mierxiati Maimaiti Rouzi[1], Peng Li[1], Mierkaiti Lazike[1], Xiao Lyu[1], Xin Zhao[2], Chuanjiang Zhang[3], Jiahui Chen[4]

[1]China Mobile Communications Group Xinjiang Co., Ltd., Urumqi, China
[2]Ningbo Putian Communication Technology Co., Ltd., Ningbo, China
[3]Feisida Technology (Beijing) Co., Ltd., Beijing, China
[4]Chengdu Ruinuode Technology Co., Ltd., Chengdu, China

*Corresponding Author. Email: gaohongli@xj.chinamobile.com

Abstract. With the exponential growth in the complexity of Internet architectures and the widespread adoption of cloud-native service technologies, traditional operation and maintenance (Artificial Intelligence for IT Operations, AIOps) models—largely reliant on the paradigm of "expert rules + fixed scripts"—have become increasingly passive and inefficient when confronted with unknown faults and massive volumes of alerts. This study focuses on the application of large-scale AI model–based intelligent agents across the full lifecycle of Internet fault handling, aiming to construct autonomous O&M agents endowed with capabilities of perception, decision-making, and execution. The paper first analyzes the core challenges in current fault management: alert storms leading to missed and false incident reports, cross-system data silos hindering root cause localization, and heavy reliance on expert experience in manual troubleshooting, resulting in delayed response times. On this basis, a hierarchical solution architecture based on large-model agents is proposed, comprising a multi-source data perception layer, a fault reasoning and decision-making layer, and an automated execution layer [1]. By integrating Retrieval-Augmented Generation (RAG) techniques with an O&M knowledge base, the proposed approach equips intelligent agents with the ability to interpret topology metrics, log semantics, and change events. Furthermore, the introduction of chain-of-thought reasoning and reflection mechanisms enables the agents to simulate expert diagnostic pathways, thereby achieving millisecond-level anomaly detection and minute-level root cause identification.

Keywords: large-scale AI models, intelligent agents, Internet fault handling, root cause analysis, self-healing operations, AIOps

1. Introduction

With the explosive growth of Internet service scale and the accelerated iteration of technology stacks, modern network infrastructures have evolved into ultra-large-scale, highly dynamic complex systems. While microservice architectures decouple functional modules, they introduce mesh-like invocation dependencies; containerization and Serverless technologies enhance resource elasticity, yet blur fault boundaries; and the

frequency of changes has been compressed from weekly cycles to hourly intervals, where any configuration deployment or version update may trigger cascading failures. Against this backdrop, faults have become the norm rather than the exception, and the traditional operations and maintenance model centered on "manual troubleshooting" is approaching its efficiency ceiling. In response, this paper focuses on the application of large-scale AI model–based intelligent agents in Internet fault handling, addressing three core research questions. First, how to construct domain-specific intelligent agent architectures for operations and maintenance that can effectively integrate multi-source data with domain knowledge. Second, how to design fault reasoning mechanisms that leverage the chain-of-thought capabilities of large models to approximate expert-level diagnosis. Third, how to achieve controllable and interpretable automated remediation, thereby reducing fault recovery time while ensuring operational safety. This study aims to validate the feasibility of transitioning large-model intelligent agents from the role of "auxiliary advisors" to that of "frontline executors," providing both theoretical grounding and practical reference for next-generation self-healing Internet systems [2].

2. Issues in traditional fault handling systems

Traditional Internet fault handling predominantly relies on a paradigm of "expert rules + manual troubleshooting." Although AIOps has been partially implemented in anomaly detection, significant structural challenges persist when examined from a full-lifecycle perspective. These challenges can be summarized into five aspects:

(1) Perception Layer: Alert Overload and Information Silos: Monitoring systems commonly rely on static threshold configurations, resulting in situations where "false positives obscure the true root cause." More critically, logs, metrics, call traces, and change events are distributed across disparate toolchains, leading to fragmented data. During fault incidents, operators must manually retrieve and correlate information across systems. Prior to the introduction of large models, there is a lack of a unified semantic layer capable of interpreting such heterogeneous data.

(2) Diagnosis Layer: Tacit Knowledge and Broken Causal Chains: The troubleshooting expertise of senior engineers is largely embedded as tacit knowledge, making it difficult to systematically codify and replicate. Meanwhile, existing algorithms are primarily correlation-based rather than grounded in causal inference. When encountering novel fault patterns not previously encoded in rule bases, model confidence drops sharply, ultimately necessitating human intervention for validation and decision-making.

(3) Decision Layer: Static Playbooks and Scenario Mismatch: Automation capabilities typically remain limited to "script execution" rather than "task planning." Operational playbooks exist as fixed scripts; however, real-world faults often involve multiple concurrent triggers. Pipeline-style playbooks are therefore inadequate for handling complex, interwoven scenarios. Operators must still manually determine which scripts to invoke and how to adjust parameters, rendering such processes only semi-automated and requiring repeated confirmation during critical execution windows.

(4) Collaboration Layer: Linear Workflows and Delayed Response: Fault response processes generally follow a linear sequence of "detection → group formation → personnel escalation → log sharing," where technical execution, communication, and decision-making roles are tightly coupled. Each additional step of information alignment significantly prolongs the Mean Time to Recovery (MTTR). This issue is particularly pronounced in cross-team collaboration scenarios, where ambiguous responsibility boundaries often lead to delays and missed optimal remediation windows.

(5) Evolution Layer: Discontinuity in Data Accumulation and Capability Reuse: Each fault incident should ideally contribute to improving system resilience; however, in practice, resolution often marks the end of the process. Root causes are not structurally labeled, and remediation actions are not abstracted into reusable capabilities. Similar faults tend to recur, especially with personnel turnover, as knowledge assets dissipate with staff mobility. Consequently, the system fails to continuously evolve from historical incidents.

In summary, a significant gap has emerged between the exponential growth in Internet system complexity and the linear progression of fault handling capabilities. This gap represents the critical entry point for large-model intelligent agents: leveraging cognitive capabilities to overcome the rigidity of rule-based scripts, using generalized knowledge to break down data silos, and transitioning from passive response to proactive planning.

3. Overview of AI large-model–based intelligent agent technologies

3.1. Technical principles of AI large models

AI large models are fundamentally built upon deep neural network architectures, leveraging hierarchical feature extraction and complex pattern learning capabilities. Through training on massive datasets, these models learn the intrinsic structures and patterns embedded within data. The training process generally consists of several key stages: data preprocessing, model construction, model training, and model evaluation. Data preprocessing involves cleaning, organizing, and annotating raw data to provide suitable inputs for the model. Model construction entails designing neural network architectures tailored to specific task requirements. During model training, parameters such as weights and biases are iteratively updated through mechanisms including forward propagation, activation functions, loss functions, and optimization algorithms. Model evaluation, typically conducted on validation datasets, assesses the model's generalization capability. In the field of Natural Language Processing (NLP), large AI models such as the GPT series and BERT have achieved remarkable success. These models are capable of understanding semantic meaning and contextual relationships in natural language, enabling complex tasks such as text generation and semantic comprehension. In the context of Internet fault handling, the natural language understanding capabilities of large models can be leveraged to transform operators' natural language queries into structured query parameters. The underlying design principles are illustrated in Figure 1.

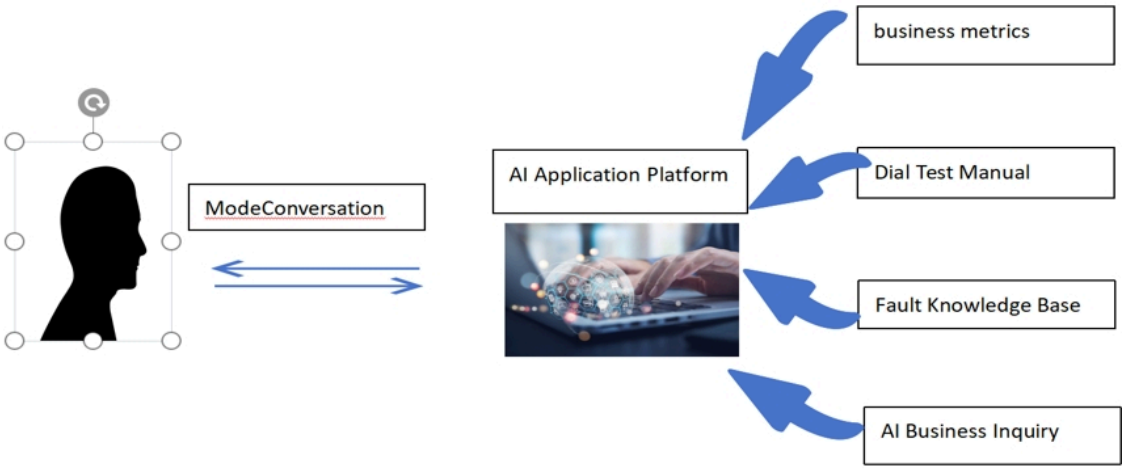

Figure 1. Schematic diagram

3.2. Principles of AI intelligent agents for probe-based fault handling

When mapped to fault handling scenarios, intelligent agent technology can be understood as encapsulating the cognitive troubleshooting model of senior engineers into a continuously operating digital system. Through multimodal alignment, event extraction, and change perception, the agent establishes an integrated perception pipeline, decomposing abstract objectives—such as "resolving a database fault"—into executable atomic steps. Based on contextual information, the model autonomously selects parameters and invokes appropriate operations, while continuously recording the success rates of different remediation strategies across various fault scenarios. Over time, this enables the identification of "high-success-rate troubleshooting paths" and the elimination of inefficient playbooks. Importantly, the application of intelligent agents in fault handling is not merely an extension of AIOps augmented with large models. Rather, it represents the first realization of a cognitive troubleshooting system integrating four core capabilities: semantic perception, expert-level reasoning, autonomous execution, and experiential asset accumulation. This paradigm enables operations teams to break free from the vicious cycle of "increasing system complexity → frequent faults → escalating manpower investment," and instead move toward a new model characterized by "self-healing systems and persistent knowledge accumulation." The conceptual framework of agent-based fault handling is illustrated in Figure 2, and the corresponding process flow is shown in Figure 3.

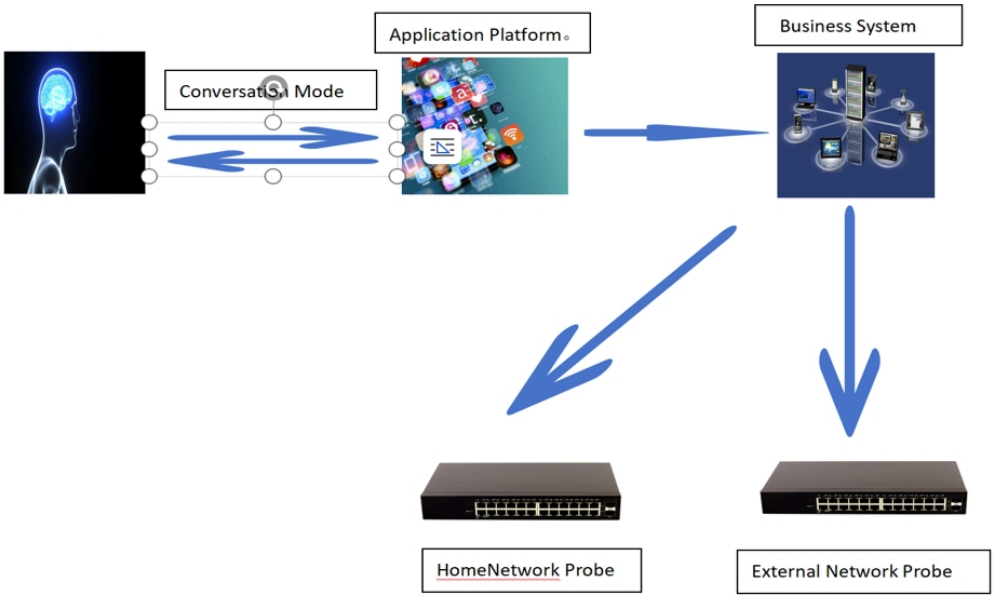

Figure 2. Schematic diagram of agent fault handling

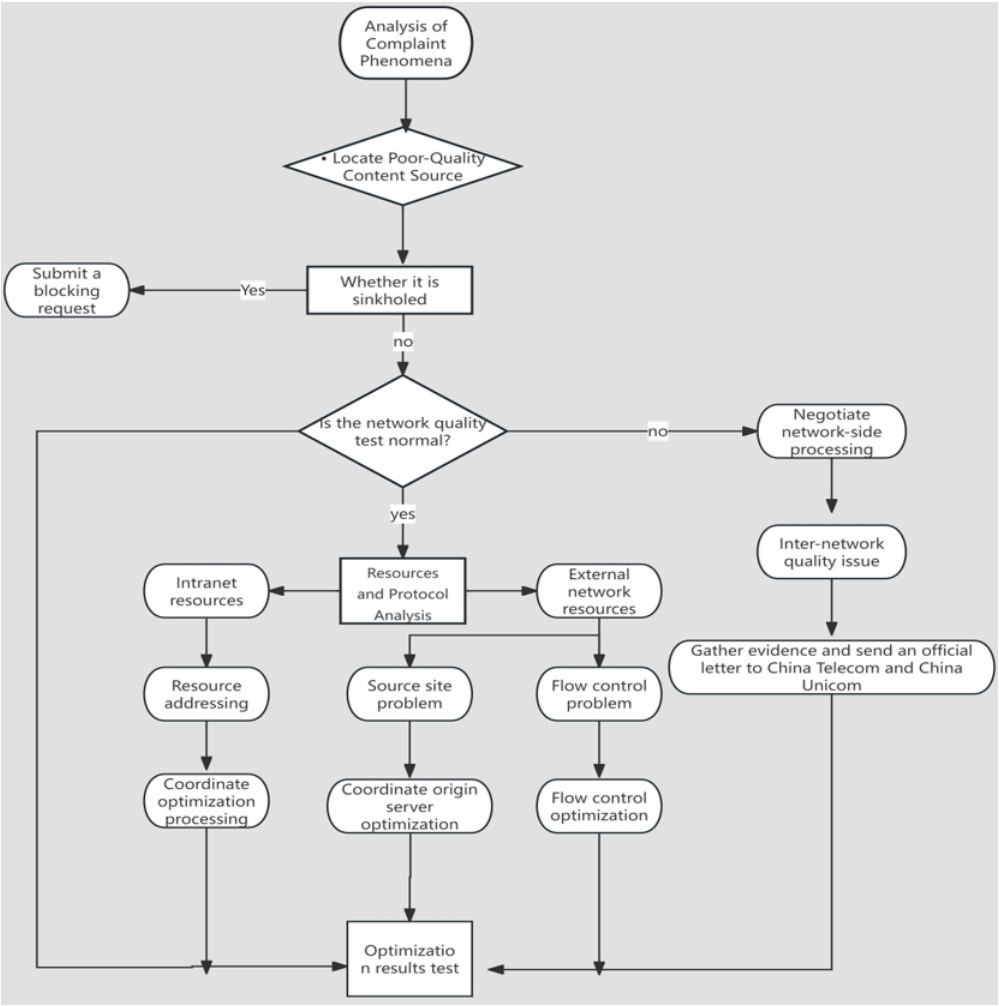

Figure 3. Flowchart of agent fault handling

4. System architecture for internet fault handling based on AI large-model intelligent agents

With the evolution of intelligent agents from "tool users" to "intelligent decision-makers," not only are the accuracy and efficiency of problem-solving significantly improved, but systems also acquire genuine capabilities for learning and continuous evolution. This transformation enables the delivery of more intelligent, reliable, and efficient service experiences, marking a critical step toward higher levels of system intelligence. The concept of "multi-agent systems" represents a higher-level abstraction. Such systems do not directly execute specific tools; instead, they manage and orchestrate one or more subordinate agents. These subordinate agents may be basic agents or composite agents themselves, forming a tree-structured, recursively extensible collaborative network. Within this architecture, the intelligent assistant functions as an advanced AI-driven decision-support system, equipped with capabilities such as learning and memory, intelligent planning, multi-task coordination, and autonomous execution. It operates analogously to an experienced senior assistant, supporting the resolution of complex operational scenarios. In addition, an Internet resource troubleshooting case repository serves as a foundational knowledge base for fault resolution, particularly in complaint-driven troubleshooting contexts. It provides standardized guidance, reference cases, and structured

planning support to ensure accurate and efficient handling of user requirements. The overall system architecture for Internet fault handling based on AI large-model intelligent agents is illustrated in Figure 4.

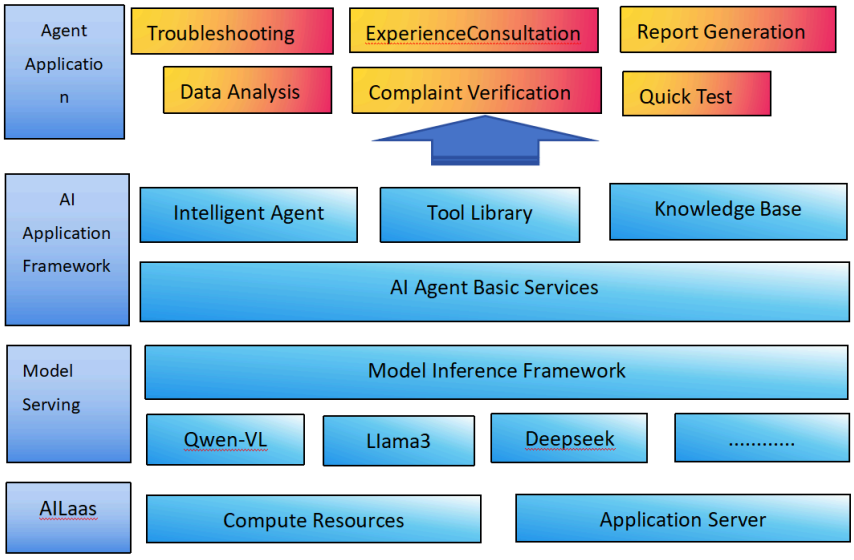

Figure 4. System architecture

5. Collaborative fault diagnosis process based on AI large models

The proposed fault diagnosis framework adopts a multi-stage processing pipeline: Stage 1: Multi-Source Perception: This stage enables comprehensive data understanding. When operations personnel input abnormal indicators, the intelligent agent processes heterogeneous data sources and outputs structured raw fault events, including temporal information, affected entities, and descriptions of observed phenomena. Stage 2: Event Compression: Multiple raw fault events are consolidated and compressed into a set of core events, effectively reducing redundancy and mitigating the impact of alert storms. Stage 3: Root Cause Reasoning: Root cause analysis is conducted through a combination of memory retrieval mechanisms and historical fault case matching, alongside chain-of-thought decomposition. This process ultimately produces a root cause diagnosis with expert-level reasoning fidelity. Stage 4: Remediation Planning: The system translates diagnostic outputs into actionable remediation strategies, progressing from diagnostic reports to operational playbooks, and further into automated execution. Simultaneously, the experience gained is captured and incorporated into the knowledge base, forming a closed-loop learning system. In summary, the AI large-model–based intelligent agent framework for Internet fault handling integrates multi-source perception to unify data views, employs event compression to address alert overload, leverages root cause reasoning for expert-level diagnosis, and utilizes remediation planning for task decomposition. Automated execution bridges the final operational gap, while experience accumulation ensures that each fault contributes to system evolution. Fundamentally, this process externalizes, formalizes, and scales the tacit cognitive models of operations engineers, thereby constructing a self-healing digital nervous system. The corresponding workflow is illustrated in Figure 5.

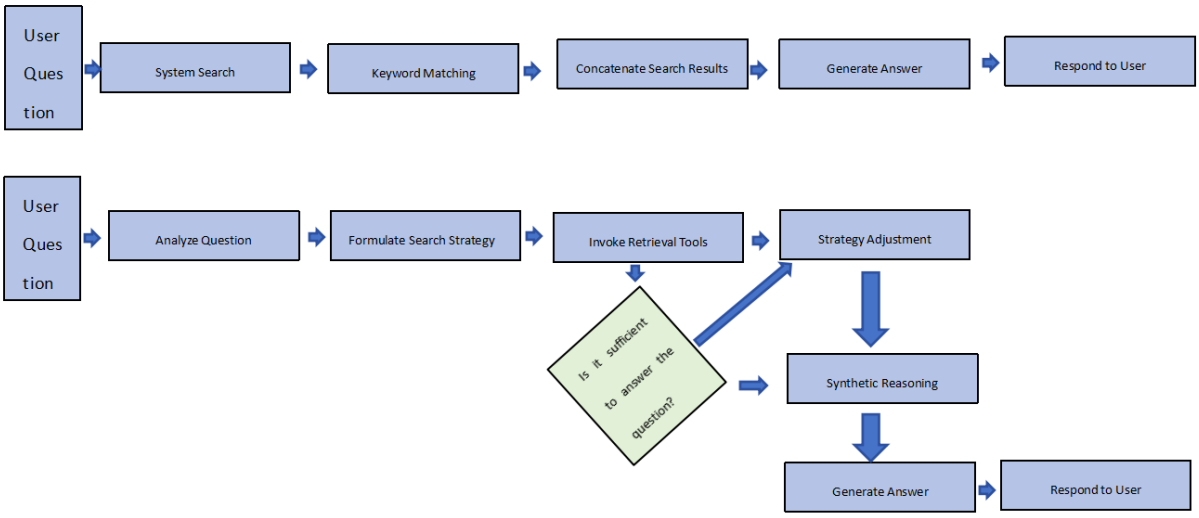

Figure 5. Conceptual flowchart

5.1. Large language models and their deployment

Functionality development based on large language models enables adaptive learning within knowledge bases and efficient processing of massive Internet probing datasets. AI assistant agents automatically archive and record communication messages and experiential documents, as well as store files shared within group environments. Knowledge assets can also be imported into the experience repository via web-based interfaces, where they are subsequently vectorized and indexed to support semantic retrieval. In parallel, the AI assistant systematically organizes business data and resource files, preserving knowledge artifacts while constructing vectorized indices to enhance retrieval efficiency and contextual understanding [3].

5.2. AI-driven intelligent business diagnosis

Through user intent recognition, the system accurately captures user queries and operational requirements, and rapidly identifies potential issues by leveraging the knowledge base. Fault identification agents can quickly determine fault categories based on predefined fault-handling workflows, significantly reducing the time required for manual analysis. By integrating real-time probe data with historical business testing data, the system provides robust data support for troubleshooting. Seamless data invocation ensures consistency and integrity across datasets, thereby improving the precision and efficiency of fault resolution. Furthermore, by analyzing historical business data, the system can predict potential failure points in advance, enabling proactive network resource allocation and optimization. This predictive capability reduces the operational impact of faults and enhances overall system stability and reliability [4].

5.3. Multimodal data fusion and analytical techniques

In automated business testing scenarios, the system captures DNS, TCP, and HTTP network interaction data, along with service-level perception indicators, enabling multidimensional fault localization and analysis. This facilitates the identification of root causes underlying service quality degradation and supports accurate fault diagnosis. Operational scenarios extend beyond traditional network-layer troubleshooting to include application-layer diagnostics. Through in-depth analysis of HTTP and SSL/TLS protocols, combined with

probing data and packet capture analysis, the system extracts and classifies protocol-level resource elements associated with test targets. This includes the ability to analyze HTTPS resources that are typically inaccessible through conventional methods. The parsed HTTPS resource information includes source URL, request method, response code, latency, resource size, start time, end time, cache-control policies, content type, resource ownership, and associated network operators. The analytical workflow is illustrated in Figure 6.

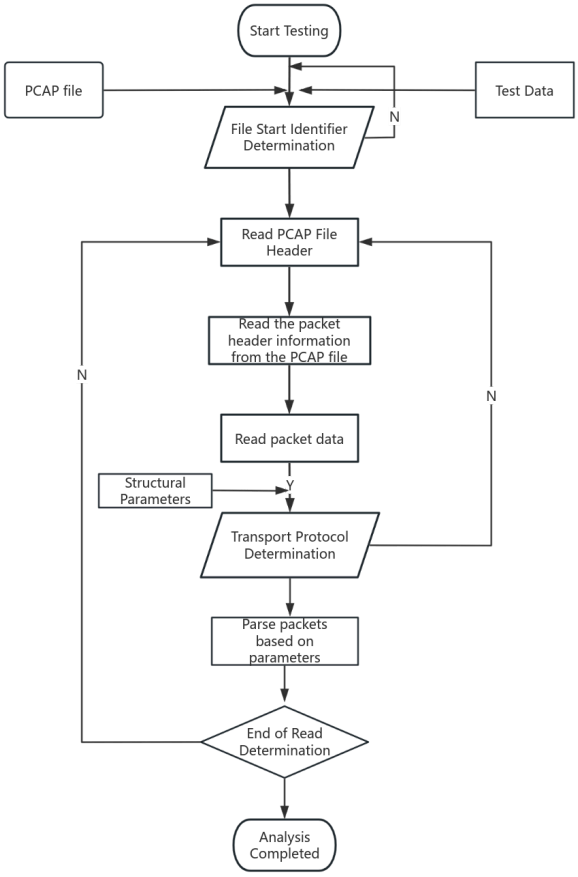

Figure 6. Analysis flowchart

5.4. AI-based report generation

Within the AI reporting agent, integration with external systems is achieved through APIs, while supporting data import in formats such as CSV and Excel. Using Apache POI, the system programmatically manipulates Word documents, populating templates with precision while automatically managing font styles and paragraph formatting. For presentation generation, Aspose.Slides is utilized to produce PowerPoint files, transforming data into visual representations such as bar charts and line graphs, complemented by explanatory text and optimized layout design. Additionally, Large Language Models (LLMs) are employed to interpret the semantic meaning of data and automatically generate analytical narratives, thereby enhancing the intelligence and readability of generated reports.

6. Application value and benefit analysis

6.1. Improvement in operational efficiency

The proposed system standardizes management processes, enables intelligent fault handling, and automates business data report generation, thereby reducing repetitive administrative tasks and communication overhead. Automated operations and data generation significantly alleviate personnel workload and enhance overall O&M efficiency. Automation replaces manual operations: traditional probing tasks require manual configuration, execution, and analysis by O&M personnel, with an average duration of approximately two hours per task. The proposed system achieves full-process automation, reducing task completion time to under five minutes—an efficiency improvement of approximately 24 times. Root cause localization is significantly accelerated: by leveraging AI for real-time packet (pcap) analysis and multidimensional data correlation, the time required for root cause identification is reduced from four hours to approximately 15 minutes, improving response speed by a factor of 16. Intelligent report generation further enhances productivity: daily and weekly reports, which previously required two person-days to complete, can now be generated within 10 minutes with an accuracy rate exceeding 95%, thereby freeing human resources for higher-value tasks.

6.2. Economic benefits

The development of an intelligent agent system for Internet fault troubleshooting addresses the lack of specialized multi-agent collaborative capabilities in scenarios involving faults, user complaints, and service quality degradation. In typical customer complaint handling scenarios, the time from initial complaint reception to root cause identification can range from two hours to an entire day. For example, when a user reports that a webpage is inaccessible, the traditional workflow involves multiple steps: customer service acknowledges the complaint and reassures the user, escalates the issue to technical departments, performs troubleshooting (including simulated access, DNS resolution checks, packet capture analysis, coordination with other teams for competitive network comparisons, and verification of potential access restrictions), identifies the root cause, and finally provides feedback to the user. This process is both complex and time-consuming. With the introduction of "AI-powered intelligent troubleshooting," fault diagnosis can be initiated with a single command. By integrating provincial-level access restriction query interfaces, the system can automatically verify whether a domain name or its resolved IP address is blocked. This allows customer service representatives to conduct root cause analysis and provide feedback to users in real time while maintaining communication, thereby improving operational efficiency and reducing labor costs associated with manual troubleshooting [5].

6.3. Social benefits

The implementation of an "AI+" intelligent Internet fault troubleshooting agent significantly enhances the ability of frontline customer service and O&M personnel to complete service quality issue resolution in a closed-loop manner, improving work efficiency by more than five times and supporting digital transformation initiatives. Moreover, the system substantially improves the timeliness of customer complaint resolution, effectively reduces the impact scope of service quality issues, and enhances user experience. Increased customer satisfaction, in turn, contributes positively to corporate image and brand reputation.

7. Conclusion

The core innovation of this study lies in the introduction of AI large-model–based intelligent agents as the "cognitive core" of the system. The findings demonstrate that such agents can overcome the limitations of traditional rule-based alert systems by leveraging advanced natural language understanding and reasoning capabilities. The collaborative mechanism between intelligent agents and probing systems has been shown to be highly effective. This dual-driven paradigm—combining "proactive probing" with "intelligent analysis"—establishes a closed-loop fault handling process: the probing system detects anomalies and triggers the intelligent agent, which then analyzes the issue and orchestrates targeted probing tasks for validation. This significantly enhances fault response speed, improves processing efficiency, and reduces Mean Time to Repair (MTTR). Overall, the application of AI large-model intelligent agents in conjunction with probing systems opens a new pathway for Internet fault management. It not only elevates the level of operational intelligence but also drives a paradigm shift from "reactive firefighting" to "proactive prevention" and ultimately toward "autonomous operations." As large-model technologies continue to mature, future Internet systems are expected to achieve stronger self-healing capabilities, providing more robust guarantees for service continuity and stability.

References

[1] Yan, L. (2022). *Research on large-scale data production systems based on automated probing of mobile applications* [Master's thesis, Hunan University].

[2] Liu, G. (2021). *Deep mining of QoE probing data and applications of artificial intelligence*. China Telecom Guangdong Branch.

[3] Bu, X. (2020). Design of an intelligent probing system for IMS interconnection and terminal device status monitoring. *Electric Power Information and Communication Technology*, *18*(11), 36–43.

[4] Zhang, C. (2025). Application of AI large-model agents in core network fault handling. *Guangxi Communication Technology, 81*(4), 20–24.

[5] Liu, W. (2026). Boundary benefit analysis of intelligent agent applications in distribution network diagnosis scenarios. *Guangxi Communication Technology*, *81*(1), 68–73.