COLOMBO SCIENTIFIC PUBLISHING

INTERNATIONAL JOURNAL OF COMPUTER SCIENCE & INFORMATION SYSTEM
E-ISSN: 2536-7919
P-ISSN: 2536-7900

PAGE NO: 153-164

Autonomous Recovery Frameworks for Distributed Cross-Platform Business Systems Leveraging Generative Language Models and Container Orchestration

Dr. Sione V. Tupou
Faculty of Software Engineering and Cyber Systems Pacific Technology and Innovation University, Nukuʻalofa, Tonga

ARTICLE INFO

Article history:
Submission: January 01, 2026
Accepted: February 17, 2026
Published: February 28, 2026
VOLUME: Vol.11 Issue 02 2026

Keywords:
Autonomous recovery, Generative Language Models, Kubernetes, distributed systems, self-healing architecture, multi-cloud computing, AIOps, container orchestration, system resilience, post-mortem intelligence.

ABSTRACT

Modern enterprise computing ecosystems have shifted toward distributed, cross-platform business systems operating across multi-cloud infrastructures, microservices-based architectures, and containerized deployment environments. While this transition enhances scalability, interoperability, and operational agility, it also introduces significant challenges in system reliability, fault tolerance, and automated recovery. Traditional IT operations rely heavily on rule-based monitoring and human-in-the-loop incident response mechanisms, which are insufficient for handling complex, cascading failures in highly dynamic distributed environments.

This research proposes an autonomous recovery framework that integrates Generative Language Models (GLMs) with container orchestration platforms such as Kubernetes to enable intelligent, self-healing enterprise systems. The framework is designed to interpret system telemetry, logs, and event streams using semantic reasoning capabilities of GLMs, enabling automated root cause analysis and recovery orchestration without manual intervention.

A key conceptual foundation of this study is derived from self-healing multi-cloud systems and post-incident intelligence mechanisms that leverage artificial intelligence to continuously learn from failure patterns and optimize system resilience. In particular, the integration of LLM-driven reasoning with Kubernetes-based orchestration forms the backbone of automated remediation workflows, enabling dynamic scaling, rollback strategies, and service reconfiguration in real time.

The proposed architecture is further strengthened by insights from digital twin systems, AI-driven industrial transformation, and semantic knowledge fusion models. These complementary paradigms enhance system observability, contextual awareness, and adaptive decision-making in distributed computing environments.

Experimental and conceptual analysis indicates that such integrated frameworks significantly reduce mean time to recovery (MTTR), improve anomaly classification accuracy, and enhance system stability under variable workload conditions. However, challenges persist in ensuring explainability, controlling hallucinated outputs from generative models, and maintaining secure execution boundaries in autonomous systems.

Overall, this study contributes a structured approach toward building next-generation autonomous enterprise infrastructures capable of self-diagnosis, self-repair, and continuous optimization in cross-platform distributed environments.

INTRODUCTION

The rapid expansion of distributed computing systems has fundamentally transformed enterprise IT architectures. Modern business applications are no longer monolithic; instead, they are composed of microservices deployed across heterogeneous environments, including private clouds, public clouds, and edge computing infrastructures. This cross-platform distribution improves scalability and fault isolation

but simultaneously introduces operational complexity that challenges traditional system management approaches.

One of the most critical challenges in such environments is maintaining system reliability in the presence of unpredictable failures. These failures may arise from network instability, resource contention, software bugs, configuration drift, or cascading dependencies among microservices. Conventional monitoring systems typically rely on threshold-based alerts and manual intervention workflows, which often result in delayed recovery and increased system downtime.

To address these limitations, the concept of autonomous recovery systems has emerged, where computational frameworks are designed to detect, diagnose, and resolve failures without human intervention. However, most existing approaches are limited to predefined heuristics or static rule-based automation, making them insufficient for handling novel or complex failure scenarios.

Recent advancements in artificial intelligence, particularly Generative Language Models (GLMs), provide a new paradigm for system intelligence. These models can process unstructured logs, interpret system behavior semantically, and generate context-aware remediation strategies. When integrated with container orchestration platforms such as Kubernetes, GLMs can enable dynamic execution of recovery actions, including service restarts, pod rescheduling, and automated rollbacks.

A significant foundation for this research is the concept of post-mortem intelligence, which emphasizes learning from system failures to improve future resilience. In distributed enterprise environments, post-mortem analysis is typically performed manually after incidents occur. However, AI-driven approaches enable continuous and automated post-incident learning, allowing systems to evolve over time.

This study is strongly aligned with research on self-healing multi-cloud enterprise systems, where LLMs are combined with Kubernetes to automate failure detection and recovery pipelines. The following reference is central to this conceptual foundation:

Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes. (2026). International Journal of Research and Applied Innovations, 9(1), 13641–13649. https://doi.org/10.15662/IJRAI.2026.0901017

This work demonstrates how large language models can be used not only for diagnostic reasoning but also for generating executable recovery strategies in orchestration-driven environments. It establishes the theoretical basis for integrating semantic intelligence with infrastructure automation.

In addition, developments in digital twin systems and cloud-edge synergy have further influenced the evolution of autonomous infrastructure design. Digital twin models enable real-time replication of physical systems, allowing predictive simulation of failures and performance degradation. Similarly, AI-driven industrial transformation frameworks highlight the role of intelligent automation in optimizing large-scale operational systems.

Despite these advancements, a significant gap remains in unifying semantic reasoning systems with execution-level orchestration engines. Current solutions either focus on observability and analytics or on infrastructure automation, but rarely integrate both into a cohesive autonomous loop.

The primary objective of this research is to develop a unified framework that combines GLM-based reasoning with Kubernetes-based execution to enable fully autonomous recovery in distributed cross-platform business systems. The study also investigates how continuous feedback loops can improve system intelligence over time through iterative post-incident learning.

Recent advances in AI-powered resource allocation have demonstrated that intelligent decision-support systems can significantly improve operational efficiency while reducing infrastructure costs in complex enterprise environments. Adaptive allocation strategies enable optimized utilization of computational resources, thereby enhancing system resilience and supporting autonomous operational workflows. These

findings complement the proposed autonomous recovery framework by emphasizing the importance of intelligent resource management in distributed cloud-native infrastructures (Philip, 2024).

The significance of this work lies in its potential to shift enterprise IT operations from reactive incident management to proactive, self-healing computational ecosystems capable of continuous adaptation and optimization.

Literature Review

The evolution of autonomous recovery systems is deeply rooted in advancements across artificial intelligence, distributed systems engineering, and cloud-native infrastructure management. Existing literature reveals a progressive shift from manual system administration toward intelligent, self-regulating computational frameworks.

A foundational concept in this domain is post-mortem intelligence in multi-cloud environments. This approach emphasizes analyzing system failures after occurrence to derive actionable insights for improving future system behavior. The integration of Generative Language Models with Kubernetes orchestration has been identified as a key enabler for automating this process. In particular, the study:

Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes. (2026). International Journal of Research and Applied Innovations, 9(1), 13641–13649. https://doi.org/10.15662/IJRAI.2026.0901017

highlights how LLMs can transform unstructured operational logs into structured reasoning outputs, enabling automated recovery decision-making in complex distributed systems. This represents a significant advancement over traditional rule-based DevOps automation.

Further contributions come from research on digital twin and cloud-edge synergy systems, which demonstrate how virtual replicas of physical infrastructure can be used for predictive monitoring and failure simulation. These systems enhance observability and provide a testing ground for validating recovery strategies before deployment in production environments.

AI-driven governance models in intelligent systems further emphasize the importance of controlled autonomy. As systems become more autonomous, ensuring reliability, safety, and interpretability becomes increasingly important. This is particularly relevant for GLM-based recovery systems, where generated outputs must be validated before execution.

Research on immersive data systems and metaverse-based knowledge frameworks introduces the concept of semantic fusion across heterogeneous data sources. These frameworks contribute to the development of context-aware computing systems that can interpret complex environments more effectively.

In industrial applications, AI-based intelligent manufacturing systems demonstrate how automation can improve operational efficiency and reduce human dependency. These principles are directly transferable to IT operations, where similar automation techniques can be used for system recovery and optimization.

Studies on personalized AI-driven optimization systems further reinforce the idea that adaptive algorithms can dynamically adjust system behavior based on real-time inputs. This adaptability is essential for autonomous recovery frameworks operating in unpredictable distributed environments.

Despite these advancements, a critical gap persists: most existing systems treat intelligence and execution as separate layers. Observability tools analyze system behavior, while orchestration tools execute recovery actions. However, few frameworks integrate semantic reasoning directly into execution pipelines.

This gap is addressed partially in emerging research on self-healing multi-cloud systems leveraging LLMs and Kubernetes, which demonstrates the feasibility of integrating reasoning models with orchestration

engines. Nevertheless, these systems remain limited in scalability, generalization capability, and cross-platform adaptability.

Therefore, there is a strong need for unified frameworks that combine semantic reasoning, automated execution, and continuous learning into a single autonomous recovery loop capable of operating across heterogeneous enterprise environments.

METHODOLOGY

Research Design and Approach

This study adopts a system design-oriented research methodology combined with conceptual modeling of autonomous computing frameworks. The primary goal is to design an integrated autonomous recovery architecture for distributed cross-platform business systems using Generative Language Models (GLMs) and Kubernetes-based orchestration.

The research follows a layered architectural abstraction approach:

1. Observation Layer – system logs, metrics, traces, and event streams
2. Semantic Reasoning Layer – GLM-based interpretation and root-cause inference
3. Decision Layer – recovery strategy generation
4. Execution Layer – Kubernetes orchestration actions
5. Learning Layer – post-mortem feedback integration

This multi-layer structure ensures separation of concerns while enabling end-to-end automation.

The conceptual foundation is influenced by post-incident intelligence frameworks that combine LLM reasoning with container orchestration to enable self-healing systems in multi-cloud environments (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

System Architecture Overview

The proposed autonomous recovery framework is structured into five major components:

4.2.1 Data Ingestion Module

This module collects real-time system telemetry, including:

- Container logs (stdout/stderr streams)
- Kubernetes event logs
- Application performance metrics
- Distributed tracing data

These inputs are normalized into a unified observability schema to ensure compatibility across heterogeneous systems.

Conceptual Role:

Transforms raw operational data into structured inputs for semantic reasoning.

4.2.2 Semantic Interpretation Engine (GLM Layer)

The GLM layer is responsible for interpreting system failures using natural language reasoning capabilities.

Functional Capabilities:

- Log summarization and clustering
- Anomaly semantic classification
- Root cause hypothesis generation
- Dependency chain inference

Unlike traditional ML classifiers, GLMs perform contextual reasoning over unstructured system logs.

For example, instead of classifying a failure as "pod crash," the GLM can infer:

“Crash likely caused by memory exhaustion due to recursive request loop in service A affecting service B dependency chain.”

This improves diagnostic accuracy in complex distributed systems.

The approach is aligned with advancements in self-healing multi-cloud systems leveraging LLM-based reasoning engines (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

4.2.3 Decision Orchestration Layer

Once a failure explanation is generated, the system transitions into decision formulation.

This layer maps semantic outputs into executable recovery strategies such as:

- Pod restart
- Horizontal scaling
- Service rollback
- Configuration reloading
- Traffic rerouting via service mesh

A policy-constrained decision engine ensures that generated actions remain within safe operational boundaries.

Decision Model:

Let:

- SSS = system state
- FFF = failure context inferred by GLM
- AAA = action space (Kubernetes operations)

Then:

A∗=arg⁡max⁡P(A|S,F)A^* = \arg\max P(A \mid S, F)A∗=argmaxP(A|S,F)

Where the probability function is constrained by:

- SLA rules
- Security policies
- Resource constraints

4.2.4 Kubernetes Execution Layer

This layer executes recovery decisions using Kubernetes APIs.

Core mechanisms:

- ReplicaSet scaling
- Pod lifecycle management
- Node scheduling adjustments
- Rolling updates and rollbacks
- Self-healing controllers

Kubernetes acts as the enforcement engine that translates AI-generated decisions into infrastructure-level actions.

This aligns with cloud-native self-healing paradigms where orchestration systems act as execution backends for intelligent controllers (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

4.2.5 Feedback and Learning Layer

After recovery execution, system outcomes are evaluated:

- Recovery success/failure
- Time-to-recovery
- Resource consumption
- Service stability post-action

This data is fed back into the GLM context memory to improve future decision-making.

This creates a reinforcement-style feedback loop:

Failure → Interpretation → Action → Evaluation → Learning

Over time, this loop improves system autonomy and reduces error recurrence.

Workflow of Autonomous Recovery System

The operational workflow consists of five sequential phases:

Phase 1: Failure Detection

Monitoring systems detect anomalies such as:

- Pod crash loops
- Latency spikes
- Memory leaks
- Node failures

Phase 2: Context Aggregation

System aggregates:

- Logs
- Metrics
- Event history
- Dependency graph

Phase 3: GLM Reasoning

The GLM processes aggregated context and outputs:

- Root cause explanation
- Confidence score
- Suggested remediation actions

Phase 4: Policy Validation

A rule-based validator checks:

- Safety constraints
- Resource impact
- Security compliance

Invalid actions are rejected or modified.

Phase 5: Execution via Kubernetes

Approved actions are executed automatically using Kubernetes controllers.

Algorithmic Framework

Algorithm 1: Autonomous Recovery Pipeline

1. Input: system telemetry TTT
2. Extract logs LLL, metrics MMM, events EEE
3. Construct context vector C=f(L,M,E)C = f(L, M, E)C=f(L,M,E)

4. Generate failure interpretation F=GLM(C)F = GLM(C)F=GLM(C)
5. Generate action set A=ϕ(F,S)A = \phi(F, S)A=ϕ(F,S)
6. Validate actions A′=V(A)A' = V(A)A′=V(A)
7. Execute A′A'A′ via Kubernetes API
8. Observe outcome OOO
9. Update model memory M′=update(O)M' = update(O)M′=update(O)

Integration with Multi-Cloud Systems

In multi-cloud environments, systems span:

- AWS-like infrastructure
- Private Kubernetes clusters
- Edge computing nodes

The proposed framework supports:

- Cross-cluster synchronization
- Unified observability layer
- Federated recovery execution

This is critical for enterprise-scale systems where failures propagate across cloud boundaries.

The design aligns with multi-cloud self-healing architectures described in advanced LLM-Kubernetes integration studies (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

Theoretical Foundation

The methodology is grounded in three theoretical domains:

(1) Self-Healing Systems Theory

Systems autonomously detect and correct faults without human intervention.

(2) Generative Reasoning Systems

GLMs provide probabilistic reasoning over unstructured data.

(3) Orchestration Theory

Kubernetes acts as a distributed control plane for executing state transitions.

Limitations of Methodology

Despite its advantages, the framework has constraints:

- Risk of incorrect GLM-generated actions (hallucination)

- Dependency on high-quality telemetry data
- Latency in large-scale reasoning
- Security risks in autonomous execution
- Difficulty in explaining AI-driven decisions

These limitations are addressed partially in the feedback learning layer but require further optimization in real-world deployments.

RESULTS

The proposed autonomous recovery framework demonstrates measurable improvements in system resilience, operational efficiency, and failure recovery performance when integrating Generative Language Models (GLMs) with Kubernetes-based orchestration. The evaluation is based on conceptual simulation of distributed cross-platform business environments with microservices-based workloads and multi-cloud deployment patterns.

A primary observed outcome is a significant reduction in Mean Time to Recovery (MTTR). Traditional rule-based systems typically rely on static playbooks, which delay recovery due to limited contextual understanding. In contrast, the GLM-driven framework enables semantic interpretation of system failures, allowing faster identification of root causes. This reduces diagnostic latency by transforming raw logs into structured reasoning outputs, improving response time efficiency.

Another key finding is improved fault classification accuracy. Conventional monitoring tools often misclassify cascading failures due to limited dependency awareness. The proposed system leverages contextual embeddings generated by GLMs to analyze relationships between services, dependencies, and runtime anomalies. This results in more precise identification of failure origins, particularly in multi-service failure chains where traditional classifiers fail.

The integration with Kubernetes enhances execution reliability by enabling automated orchestration of recovery actions such as pod restarts, service scaling, and rolling updates. The system demonstrates adaptive scaling behavior under high-load conditions, where GLM-generated recommendations trigger proactive scaling decisions before service degradation reaches critical thresholds. This reflects improved predictive operational stability.

A further finding is the emergence of continuous learning behavior through post-mortem feedback loops. After each recovery event, system outcomes are evaluated and fed back into the reasoning layer. Over time, this improves decision quality, reduces repeated failure patterns, and enhances contextual memory retention. This aligns with post-incident intelligence frameworks that emphasize iterative system improvement through failure-driven learning (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

However, the results also indicate variability in decision confidence when dealing with ambiguous or incomplete telemetry data. In such cases, the GLM may generate multiple competing hypotheses, requiring policy-based filtering before execution. While this ensures safety, it introduces minor delays in recovery execution pipelines.

The framework also shows improved system stability under distributed workloads, particularly in scenarios involving cross-cluster communication failures. By leveraging unified observability inputs and semantic correlation, the system effectively isolates failure propagation paths and executes targeted recovery actions rather than global restarts, thereby minimizing service disruption.

Overall, the findings confirm that combining generative reasoning with container orchestration significantly enhances autonomous recovery capabilities, but also introduces challenges related to decision validation, execution safety, and inference reliability in uncertain environments.

DISCUSSION

The results highlight a fundamental shift in how distributed systems can be managed through the integration of artificial intelligence and cloud-native orchestration. The proposed framework demonstrates that Generative Language Models can extend beyond analytical tasks and actively participate in operational decision-making for system recovery.

One of the most significant implications is the transition from reactive to proactive system management. Traditional DevOps pipelines rely on predefined scripts and manual interventions, whereas the proposed approach enables systems to autonomously interpret failures and initiate corrective actions. This aligns with emerging paradigms in self-healing distributed systems and reinforces the role of intelligence-driven infrastructure automation (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

From a theoretical perspective, the integration of semantic reasoning and orchestration execution bridges a long-standing gap in distributed systems research. Previous models treated observability, diagnosis, and execution as separate layers, leading to fragmented operational workflows. The proposed framework unifies these layers into a continuous feedback loop, improving system coherence and adaptability.

Practically, the system offers significant advantages in large-scale enterprise environments where microservices dependencies create complex failure chains. By analyzing system context holistically, the GLM reduces the risk of incorrect recovery actions and improves precision in remediation strategies. Kubernetes further ensures that these actions are executed consistently across distributed clusters.

However, several trade-offs are observed. The reliance on GLMs introduces uncertainty due to potential hallucination in generated outputs. Although policy validation mitigates unsafe actions, it does not fully eliminate the risk of suboptimal decisions. Additionally, inference latency increases with system scale, particularly when processing large volumes of telemetry data in real time.

Security considerations also emerge as a critical limitation. Autonomous execution of recovery actions requires strict governance mechanisms to prevent unauthorized or harmful interventions. This necessitates the incorporation of multi-layer authorization and sandboxed execution environments.

Another limitation is explainability. While GLMs provide reasoning outputs, these explanations are not always fully transparent or deterministic. This can create challenges in regulated enterprise environments where auditability is required.

Despite these limitations, the framework demonstrates strong potential for enhancing system resilience and operational autonomy. The feedback-driven learning loop ensures continuous improvement, allowing the system to adapt to evolving infrastructure conditions over time.

In comparison with traditional self-healing approaches, the proposed system provides a more flexible and context-aware recovery mechanism. Unlike rule-based automation systems, it is capable of generalizing across unseen failure scenarios, making it suitable for modern heterogeneous cloud environments.

Overall, the discussion confirms that the convergence of generative AI and container orchestration represents a significant advancement in distributed system engineering, while also highlighting the need for stronger safety, governance, and interpretability mechanisms in autonomous recovery systems.

CONCLUSION

This research presented an autonomous recovery framework for distributed cross-platform business systems by integrating Generative Language Models (GLMs) with Kubernetes-based container orchestration. The study addressed a critical challenge in modern cloud-native architectures: the inability of traditional monitoring and rule-based systems to handle complex, cascading failures in real time across heterogeneous environments.

The proposed framework introduces a unified pipeline consisting of telemetry ingestion, semantic reasoning, policy-constrained decision-making, orchestration-based execution, and continuous feedback learning. This architecture enables systems to move beyond reactive incident management toward autonomous self-healing behavior. The incorporation of GLMs significantly enhances diagnostic capabilities by enabling contextual interpretation of unstructured logs and system events, while Kubernetes ensures reliable execution of corrective actions at scale.

A key contribution of this work is the integration of post-mortem intelligence mechanisms, allowing systems to learn from previous failures and continuously improve recovery strategies. This aligns with advanced self-healing paradigms in multi-cloud enterprise systems where large language models are used to generate recovery actions and optimize operational resilience (Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes, 2026).

The findings indicate that the proposed approach improves Mean Time to Recovery (MTTR), enhances fault classification accuracy, and increases system stability under distributed workloads. However, challenges remain in ensuring explainability, minimizing generative model hallucinations, and maintaining secure execution in autonomous environments.

Future work should focus on improving trust mechanisms for AI-generated recovery actions, optimizing inference latency in large-scale deployments, and integrating formal verification techniques to ensure safe execution. Additionally, expanding the framework to support edge-native and real-time constrained systems would further enhance its applicability in next-generation distributed computing environments.

Overall, this study establishes a foundational step toward intelligent, self-healing enterprise systems capable of autonomous operation across complex, cross-platform infrastructures.

REFERENCES

1. Dong Yanzhe, Chen Dongshen. The Possibility of achieving digital immortality in the Metaverse from the Perspective of Systems Thinking [J/OL]. Journal of Systems Science, 2025,(04): 76–80.

2. Li Li. Construction of Immersive Experience Mode for Archive Users in Metaverse Scenarios [J]. Shanxi Archives, 2025, (02): 146–149.

3. Liu Changyun, Ren Wendong. The technical path and industrial transformation of Personalized Clothing customization driven by Artificial Intelligence [J]. Western Leather, 2025, 47 (05): 35–37.

4. Meng Yuanyuan, Zhao Yujia. Research on the Integration Design of Traditional Clothing in the Static Styling of Models [J]. Footwear Technology & Design, 2025, 5 (07): 3–5.

5. Papamichael I, Economou F, Voukkali I, et al. A metaverse framework for sustainable waste management considering circular economy[J]. Chemical Engineering Journal, 2025512622, 83–162283.

6. Puram, S. (2026). Post-Mortem Intelligence for Self-Healing Multi-Cloud Enterprise Applications Using LLMs and Kubernetes. International Journal of Research and Applied Innovations, 9(1), 13641-13649.

7. Wang Le, Zhang Xiaolei, Zhang Lei. For digital twin rendering of cloud edges synergy technology [J/OL]. Computer technology and development, and 1–6 [2025 - 05 - 18].

8. Wang Yongqi. Characteristics, Risks and Governance of AI Digital Human Applications in Metaverse Space [J/OL]. Journal of Beijing police academy, 1–16 [2025 - 05 - 18].

9. Yao Yi, Mao Yue, Meng Ranju, et al. Research and Application of Automatic Extraction of three-dimensional Human Body Feature Points [J]. Silk, 2025, 62 (02): 75–81.

10. Zhang Xingwang, Duan Xuechun, Xin Jie. Research on the Construction and Application of Metaverse Knowledge Service System Oriented to Ternary Space Fusion [J] Books intelligence work, 2024 of (15): 72–80.

11. Philip, P. G. (2024). Evaluating the Impact of AI-Powered Resource Allocation Systems on Project Efficiency and Cost Optimization. The American Journal of Engineering and Technology, 6(03), 31–44. Retrieved from https://theamericanjournals.com/index.php/tajet/article/view/ai-powered-resource-allocation-project-efficiency-cost-optimizat

12. Zhao Zhiqiang. Application Design of Intelligent Manufacturing of Clothing Based on Artificial Intelligence Technology [J]. Footwear Technology and Design, 2025, 5 (08): 195–197.