An MLOps Framework for Lifecycle Management of Intelligent in NPPs[1]

Seokjoo Choi[a*], Hoon Jeong [a], YeonJin Ryu [a], KiJeong Kil[a], YoungSoo Min[a], YoonJoon Lee [b],YoungHo Chae[c]
[a] *Suredatalab Co. 38, Expo-ro 446 Beon-gil, Yuseong-gu, Daejon, 34050*
[b] *KAIST, School of Computing, 291, Daehak-ro Yuseong-go, Daejon 3414*
[c] *KAERI, 111, Daedeok-Daero 989 Beon-gil, Yusoeng-gu, Daejon, 34057*
[*] *Corresponding author: asbawi@suredatalab.com*

***Keywords:** NPP, AI model, IDSS, MLOps Framework*

1. Introduction

The integration of Artificial Intelligence (AI) into nuclear power plant (NPP) operations has emerged as a transformative force in enhancing operational safety and efficiency. Intelligent Operator Decision Support Systems (IDSS) are increasingly being deployed to assist operators in navigating complex, mission-critical scenarios by processing vast amounts of sensor data and identifying potential anomalies in real-time.

Currently, many AI-based training and support systems in the nuclear industry rely on static development approaches, where models are trained once and rarely updated. This lack of continuous management fails to address the dynamic nature of operational environments and complicates adherence to strict regulatory requirements. Consequently, there is an urgent need for a systematic MLOps framework that can automate the lifecycle management of AI models to guarantee compliance with the high-reliability demands of the nuclear industry.

To address this, this study proposes a systematic MLOps framework designed to automate the lifecycle management of AI models, featuring an optimized pipeline and a case study demonstrating its effectiveness in transitioning static models into adaptive, reliable systems

Section 2 reviews related work on AI-driven training systems in the nuclear industry and current MLOps methodologies. Section 3 details the proposed MLOps framework. Section 4 describes the implementation of the system and the technical stack used for development. Section 5 presents a case study evaluating the framework's performance and its impact on operational reliability. Section 6 discusses the implications of our findings regarding human-AI oversight, and Section 7 concludes the paper with a summary of contributions and directions for future research.

2. Related Work

2.1 AI-Driven Training Systems in the Nuclear Industry

Modern NPP training has transitioned from traditional physics-based simulation to AI-integrated frameworks. Earlier studies established the foundation for high-fidelity simulators [1]. However, as operational complexity increases, recent research has pivoted toward AI-augmented training environments to enhance operator situational awareness [2].

In the domain of Operator Support Systems (OSS), machine learning models are increasingly utilized to identify anomalies in real-time sensor data, offering proactive diagnostic guidance [3]. They often struggle with the dynamic nature of NPP operational environments. Furthermore, a critical limitation in existing literature is the lack of a sustainable approach to maintain these models; current AI-based training systems are typically developed as static implementations, failing to account for the continuous evolution of plant operational data [4].

2.2 Recent MLOps Methodologies

MLOps has become the standard paradigm for deploying machine learning systems in high-stakes industries. Recent methodologies have focused on several key components:

- CI/CD/CT Pipelines: Continuous Integration, Deployment, and Training (CI/CD/CT) are foundational to ensuring that models are reliably tested and updated [5]. In safety-critical sectors, these pipelines are augmented with automated verification gates to ensure system stability.
- Data and Model Versioning: Modern frameworks prioritize data lineage—the ability to trace the origin of training data and its impact on model performance [6].
- Continuous Monitoring and Drift Detection: Advanced frameworks employ automated monitoring to detect Data Drift and Concept Drift [7]. By identifying performance degradation in real-time, these systems trigger automated re-training cycles, transforming static models into adaptive systems.

Despite these advancements in general MLOps, there is a distinct research gap in applying these methodologies specifically to the nuclear industry, where strict regulatory compliance and safety standards demand a

[1] This work are partially supported by National Research Foundation of Korea (NRF) under Grant RS-2022-00144150

more rigorous, verifiable approach to lifecycle management than standard commercial applications [8].

3. Proposed MLOps Framework

The proposed MLOps framework is designed to integrate the development and service platforms, specifically addressing the stringent safety and reliability. By adopting a decoupled yet integrated design, the framework ensures that AI development remains agile while maintaining strict operational stability.

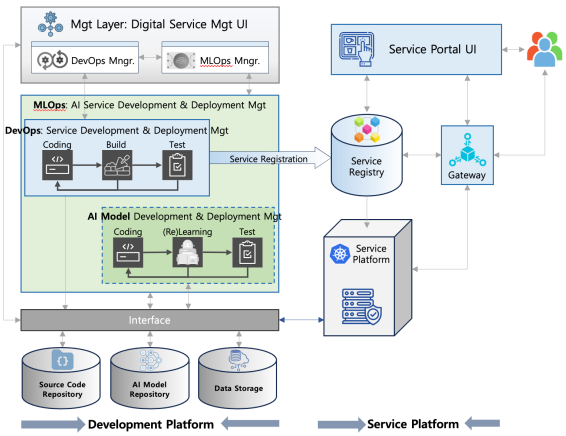

Fig. 1 MLOps Framework Architecture

3.1 Development Platform: AI Service Lifecycle Management

The development platform automates the end-to-end lifecycle, ensuring that AI-based support systems are not static but evolve in alignment with plant operational data.

- DevOps: Orchestrates the integration of service coding, building, and testing. This ensures software stability and maintains a comprehensive history of changes, providing the traceability required for high-reliability systems.
- AI Model: Manages the entire ML lifecycle, including coding, (re)learning, and testing. The re-learning pipeline is a critical component, enabling the system to adapt to data drift and maintain model accuracy in response to evolving NPP operational parameters.
- Repositories: Function as a robust version control system. By documenting data lineage alongside models and code, the framework enhances reproducibility and ensures full compliance with regulatory audit standards.

3.2 Service Platform: Orchestration and Delivery

The service platform provides a secure and scalable environment for deploying and running validated AI services.

- Service Registry: It enforces system consistency by ensuring that only authorized, validated service versions are invoked during operation.
- Service Platform (Orchestrator): Utilizing container-based orchestration, it ensures system availability, scalability, and precise resource allocation, which are vital for real-time decision support.
- Gateway & Service Portal UI: The gateway secures communications, while the UI is designed to communicate AI-generated insights with transparency, reinforcing operator trust in the decision support process.

3.3 Management Layer

DevOps/MLOps Manager: Serves as the control tower for the entire pipeline. It provides real-time visibility into pipeline performance and supports informed operational decision-making. By detecting performance degradation or drifts, it triggers automated re-learning processes or initiates deployment approvals.

3.4 System Workflow and Operational Flow

The service platform provides a secure and scalable environment for deploying and running validated AI services. To illustrate how the proposed MLOps framework operates in a cohesive manner, the end-to-end workflow is structured into a continuous, closed-loop process spanning the development and service platforms.

The workflow proceeds through the following sequential phases:

1. Data Ingestion and Pipeline Triggering: Collected operational data are continuously fed into the Data Storage.
2. Model Development and Validation: Within the Development Platform, the AI Model (Re)Learning component updates the model using version-controlled data and source code retrieved from the respective Repositories
3. Service Registration and Deployment: Once validated, the model is packaged and registered into the Service Registry via the DevOps build and test pipeline. The Service Platform Orchestrator then deploys the new service version into the production environment using container-based orchestration.
4. Operational Support and Feedback: Finally, deployed AI services are exposed through the Gateway and Service Portal UI, delivering real-time decision-support insights to NPP operators. Simultaneously, ongoing operational metrics are fed back into the system, sustaining the continuous lifecycle management loop.

4. Implementation and System Design

4.1 Development Platform: AI Service Lifecycle Management

The technical stack is strategically divided into the Development Platform and the Service Platform to

maintain a clear boundary between experimental modeling and production execution.

Development Platform Infrastructure:

- Version Control and Repositories: Source code management and data lineage tracking are implemented using robust versioning systems (e.g., Git-based repositories) to ensure full auditability.
- ML Development and Training Engine: The environment utilizes containerized Python environments equipped with standard machine learning libraries to execute the coding, testing, and automated (re)learning pipelines.

Service Platform Infrastructure:

- Container Orchestration: The service platform relies on a containerization and orchestration engine (e.g., Kubernetes) to manage deployed AI services, guaranteeing high availability, fault tolerance, and precise resource allocation.
- Service Registry and Gateway: A centralized registry maintains metadata and version control for all production services, while a secure API gateway manages incoming and outgoing traffic to the Service Portal UI.

Infrastructure Integration Layer:

- Communication and data synchronization between the development repositories and the production service platform are handled via secure, standardized interfaces, ensuring that automated deployment gates and re-learning triggers operate seamlessly without compromising system security.

4.2 Pipeline

The end-to-end pipeline automation is the core mechanism that bridges the development platform and the service platform, ensuring that AI models for the NPP IDSS remain accurate, robust, and compliant. The pipeline consists of four sequential stages:

- Data Collection: Operational data streams, including plant sensor readings and historical logs from the data storage, are systematically gathered. This stage ensures continuous ingestion of real-time plant parameters required to detect operational shifts.
- Data Verification: This stage validates data integrity, handles missing values, and screens for anomalies to prevent corrupted data from influencing the learning process.
- Automated Training: Triggered by performance degradation or identified data drift from the management layer, the automated training engine executes the (re)learning workflow. It utilizes version-controlled datasets and source code to update the model weights while adhering to predefined computational constraints.
- Model Verification: Following training, the updated model undergoes automated testing. This verification gate ensures that the model meets strict performance before it is packaged and registered into the service platform for deployment.

4.3 Interface and UX

The interface and UX layer bridges the gap between complex MLOps automation and the operational workflows of nuclear power plant (NPP) operation. The layer is divided into two primary portals:

Service Portal UI for Operators:

- Designed as the primary operational touchpoint, the Service Portal UI delivers real-time AI-driven decision-support insights, anomaly detections, and diagnostic recommendations during plant operations.
- To maintain operator trust in high-stakes environments, the UI incorporates explainable AI (XAI) features that visualize the rationale behind model predictions, ensuring that operators can critically evaluate the system's guidance before taking action.

Digital Service Management UI:

- This administrative dashboard provides total visibility over the underlying MLOps lifecycle, tracking pipeline execution states, model version histories, and system performance metrics.
- Model developers and system administrators can monitor data drift indicators, review automated verification results, and manually authorize or override deployment gates, maintaining strict human-in-the-loop governance over the AI lifecycle.

5. Case Study / Application Results

Due to the security constraints and operational safety requirements of live NPPs, obtaining direct real-time production data for live experimentation is challenging. Therefore, this study implements a simulation-based validation environment [9].

5.1 Experimental Setup

The experimental environment is designed to emulate a nuclear power plant's operational lifecycle, integrating both the development and service platforms:

- Simulation Environment: High-fidelity simulation tools are utilized to generate synthetic operational data streams, replicating normal steady-state operations as well as transient and anomalous accident scenarios.
- Data and Pipeline Configuration: The MLOps pipeline is configured to automatically ingest these logs, perform data verification, trigger automated re-learning upon simulated drift detection, and execute model verification gates.
- Evaluation Metrics: The performance of the framework is assessed based on pipeline execution latency, model adaptation accuracy against simulated anomalies, and the overall administrative overhead required for lifecycle management.

5.2 Performance Evaluation of the MLOps Pipeline

The automated MLOps pipeline was subjected to continuous simulation streams to evaluate its end-to-end responsiveness and reliability. When simulated operational shifts and data drifts were introduced, the management layer successfully detected the anomaly thresholds and automatically initiated the re-learning pipeline without manual intervention. Container-based orchestration ensured seamless service updates through the Gateway, demonstrating that the framework can maintain continuous model availability and system stability under simulated operational workloads.

5.3 Impact on Training Efficiency and Model Quality

The automated MLOps pipeline was subjected to continuous simulation streams to evaluate its end-to-end responsiveness and reliability. When simulated operational shifts and data drifts were introduced, the management layer successfully detected the anomaly thresholds and automatically initiated the re-learning pipeline without manual intervention. Container-based orchestration ensured seamless service updates through the Gateway, demonstrating that the framework can maintain continuous model availability and system stability under simulated operational workloads.

6. Discussion

While the proposed MLOps framework automates the end-to-end lifecycle, safety-critical nuclear environments require a cautious approach to full automation due to high uncertainty in unpredictable scenarios. To balance automated efficiency with human safety, the framework embeds human-in-the-loop (HITL) governance into the management layer. While routine tasks are automated, final deployment approvals and critical overrides remain under human authority via the Digital Service Management UI, ensuring that operational agility does not compromise safety.

Deploying MLOps in the nuclear sector presents unique challenges regarding scalability and strict regulatory compliance, as nuclear engineering demands verifiable transparency and rigorous documentation to satisfy regulatory bodies like KINS. The proposed framework ensures scalability through container-based orchestration and microservice architectures to handle expanding operational data without bottlenecks. Simultaneously, regulatory compliance is achieved by enforcing strict data lineage tracking, version control, and automated verification gates. By maintaining an immutable audit trail, the framework bridges the gap between modern AI lifecycle management and the rigorous accountability required for safety-critical infrastructure.

7. Conclusion

This study proposed an MLOps framework tailored for IDSS in nuclear power plants, bridging the gap between agile AI development and strict regulatory standards. By integrating automated pipelines and decoupling platforms, the framework ensures continuous adaptability while guaranteeing auditability through data lineage tracking and verification gates. Furthermore, simulation-based validation demonstrated that the framework reduces lifecycle overhead, enhances model robustness against data drift, and maintains a balance between automation and human oversight.

Several promising directions remain:

- Future work will focus on scaling the framework to handle concurrent multi-model deployments across diverse plant subsystems and multiple reactor units.
- Enhancing the interpretability of automated re-learning triggers and model predictions will further strengthen operator trust and situational awareness in high-stakes control room settings.
- Rigorous verification should be successfully satisfied and documentation standards are also required for safety-critical nuclear engineering environments.
- As simulation-based results mature, future research will explore phased integration with hardware-in-the-loop (HIL) simulators and actual operational pilot environments, subject to regulatory approval and safety validation.

REFERENCES

[1] A Scholz, Nuclear power plant simulators for operator training, inis.iaea.org, 1988
[2] H Mendoza, V Yadav, SB Alam, D Eskins, and R Iyengar, Advances in digital twins and AI/ML for condition monitoring in nuclear applications, Frontiers in Energy Research, 2026
[3] A Chaudhary, J Han, S Kim, A Kim, and S Choi, Anomaly detection and analysis in nuclear power plants, Electronics, 2024
[4] S Sinha and YM Lee, Challenges with developing and deploying AI models and applications in industrial systems, Discover Artificial Intelligence, 2024
[5] D Kreuzberger, N Kühl, and S Hirschl, Machine learning operations (MLOps): Overview, definition, and architecture, EEE access, 2023
[6] Renan Souza et al, Provenance data in the machine learning lifecycle in computational science and engineering, 2019 IEEE/ACM Workflows in Su pport of Large-Scale Science, 2019
[7] J Zenisek, F Holzinger, and M Affenzeller, Machine learning based concept drift detection for predictive maintenance, Computers & Industrial Engineering, 2019
[8] J Abella et al, SAFEXPLAIN: a Complete Approach Towards Trustworthy AI-Based Safety-Critical Systems, Euromicro Conference on Digital System Design (DSD), 2025
[9] Annual Report: Research Project on the Development of AI-Based Decision Support Technology to Prevent Human Error in Nuclear Power Plant, KAERI, 2025