iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IX **Month of publication:** September 2026

DOI: https://doi.org/10.22214/ijraset.2026.84759

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

Adaptive Transfer Learning Based Anomaly Detection for Industrial IoT System

Abinya S[1], Dr. S. Sasirekha[2]
[1]PG Student, Department of Computer Science [Artificial Intelligence and Machine Learning], NITTTR Chennai, Chennai, India
[2]Associate Professor, Department of Computer Science & Engineering, NITTTR Chennai, Chennai, India

Abstract: This paper proposes a transfer learning-based anomaly detection framework coupled with adaptive learning for predictive maintenance in Industrial Internet of Things (IIoT) systems. The proposed framework is designed and validated using a small-scale prototype to demonstrate its feasibility and effectiveness. The proposed architecture addresses the computational limitations of edge devices by employing a cloud-assisted/remote processing framework, where sensor data are streamed through Apache Kafka for real-time analysis. A lightweight Conditional Autoencoder is used to detect anomalies from streaming sensor data, making the framework suitable for resource-constrained industrial environments. To overcome the scarcity of machine-specific training data, a global model is trained using data collected from machines belonging to the same machine category. The trained global model is subsequently adapted to individual machines using transfer learning whenever machine-specific data are available. The framework continuously monitors incoming sensor data for concept drift and performs adaptive retraining when persistent drift is detected, enabling the model to learn changes in machine behavior caused by aging and wear. The system processes incoming data through parallel anomaly detection and drift detection pipelines while storing labeled operational data for future model adaptation and retraining. The proposed framework provides an adaptive and scalable approach for real-time anomaly detection in Industrial IoT predictive maintenance systems. The experimental implementation is carried out using a remote processing framework. The proposed architecture is cloud-ready and is intended to support cloud-assisted deployment for large-scale Industrial IoT environments in future work.

I. INTRODUCTION

Real-time IIoT applications often face computational constraints at edge devices, making the deployment of complex deep learning models challenging. To address this limitation, the proposed framework adopts a cloud-assisted architecture in which sensor data generated by industrial machines are published to an Apache Kafka message queue, while the cloud application continuously consumes the streaming data for real-time anomaly detection. Industrial machines exhibit concept drift over time due to factors such as wear, aging, and changing operating conditions. Conventional anomaly detection techniques generally assume a stationary data distribution and therefore fail to adapt to these gradual changes, leading to degraded detection performance. To overcome this challenge, the proposed framework continuously monitors the incoming sensor stream for concept drift. Drift detection is performed using an online drift detection mechanism, where anomalous observations are excluded from the drift analysis to prevent false drift detection. When persistent drift is observed for a predefined number of consecutive detections, the framework concludes that the machine behavior has changed and initiates adaptive retraining using the newly collected normal operating data.

To achieve efficient real-time inference, the proposed system employs a lightweight yet effective Conditional Autoencoder for anomaly detection. Since collecting sufficient machine-specific training data is difficult in industrial environments, the framework first trains a global model using data from a particular machine category (for example, generators or motors) rather than relying on data from a single machine. The trained global model is stored as a model artifact, while its associated metadata, including model information and storage location, are maintained in a PostgreSQL-based metadata repository.

When a new machine is introduced, the corresponding global model is adapted using transfer learning with the available machine-specific data, allowing the model to capture machine-specific characteristics while preserving the generalized knowledge learned from the global model. If sufficient adaptation data are unavailable, the global model can be used directly for anomaly detection without mandatory local retraining. Incoming sensor data are processed through two parallel pipelines: an anomaly detection pipeline and a concept drift detection pipeline. The anomaly detection module labels incoming observations and stores them in the database, creating a continuously growing repository of operational data that can be utilized for future adaptive retraining and training of additional machine models. Consequently, the proposed framework also contributes toward mitigating the data scarcity problem commonly encountered in industrial predictive maintenance applications.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

II. PROPOSED METHODOLOGY

A. *Overview*

The proposed framework presents an adaptive transfer learning-based anomaly detection system for predictive maintenance in Industrial Internet of Things (IIoT) environments. The framework is designed to support real-time anomaly detection of streaming data, while addressing the computational limitations of edge devices, and also adapts to the changing behavior of industrial machines over time. Instead of performing computationally intensive anomaly detection at the edge, the proposed system adopts a cloud-assisted/remote processing architecture, which separates the data processing layer from the data collection layer, and in which sensor data are continuously streamed to the cloud/remote server for processing.

The framework consists of two major layers: the Data Collection Layer and the Data Processing Layer. The data processing layer comprises anomaly detection, concept drift detection, and adaptive model retraining. These components work together to detect anomalies and report them in real time, while adapting to changes in the operating conditions of industrial machines.

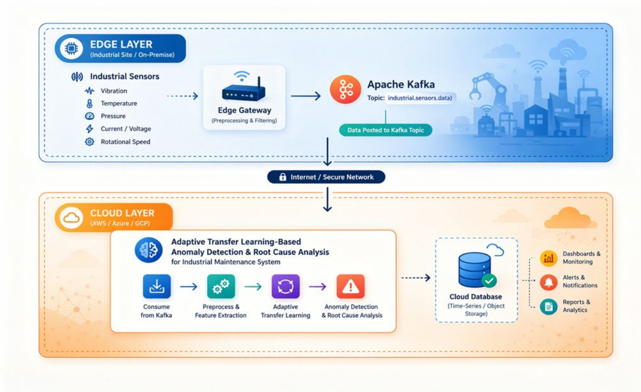


Fig. 1. Overview of the proposed framework.

B. *Data Collection Layer*

In industries, a single machine can have multiple sensors, which continuously generate sensor measurements such as vibration, acoustic, temperature, current, and decomposed signal features (IMF-1, IMF-2, and IMF-3). These measurements have to be collected from the monitored machines and posted to the Kafka message queue in JSON format, along with the machine ID from which the data were collected. The proposed framework employs Apache Kafka as the messaging layer to ensure reliable and scalable transmission of sensor data. Each incoming sensor record is published to a Kafka topic, enabling asynchronous communication between the data producer and the processing modules. This architecture decouples data acquisition from data processing and supports continuous real-time monitoring without interrupting machine operation.

C. *Data Processing Layer*

The proposed data processing layer listens to the message queue topic for messages from the data collection layer. The data processing layer can run somewhere in a remote server or cloud. The JSON string received from the message queue is passed through the anomaly detection and adaptive retraining modules.

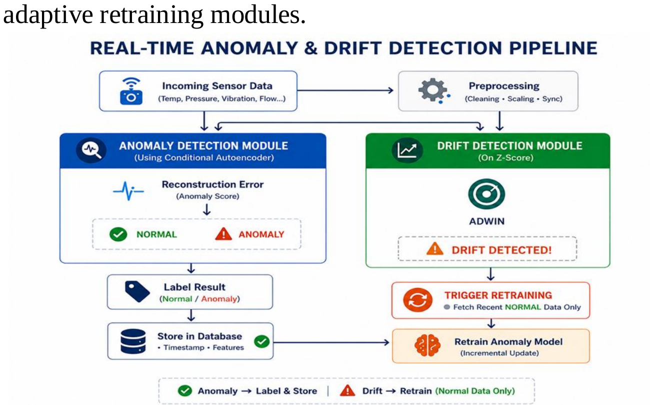


Fig. 2. Architecture of the data processing layer.

1) *Anomaly Detection Module*

a) Global Conditional Autoencoder Model: The anomaly detection module employs a lightweight Conditional Autoencoder to learn the normal operating characteristics of industrial machines based on machine type. Owing to the limited availability of machine-specific training data, a global model is initially trained using historical data collected from machines belonging to the same machine category, such as generators or motors. This enables the model to capture generalized operational characteristics without requiring extensive data from every individual machine.

The trained global model is stored as a model artifact, while the associated model information, including machine category and model location, is stored separately in a PostgreSQL metadata repository. This separation enables efficient management and retrieval of models during deployment.

b) Transfer Learning for Machine-Specific Adaptation: When the anomaly detection system receives data from a specific machine, the corresponding global model is retrieved from the model repository. If sufficient machine-specific operational data are available, transfer learning is performed to fine-tune the global model, allowing it to learn the unique operating characteristics of the target machine while preserving the generalized knowledge acquired during global training.

In situations where sufficient local data are unavailable, the global model can be directly employed for anomaly detection without mandatory machine-specific retraining. This approach reduces the dependency on large machine-specific datasets and enables rapid deployment across multiple industrial assets.

c) Real-Time Anomaly Detection: The incoming sensor stream is continuously consumed from the Kafka topic and processed by the anomaly detection module. Before inference, the sensor measurements are normalized using the corresponding feature scaler associated with the trained model. The normalized features, together with the machine category information, are supplied to the Conditional Autoencoder to reconstruct the input data.

The anomaly detection decision is based on the reconstruction error. Sensor observations with reconstruction errors exceeding the predefined threshold are classified as anomalous, whereas the remaining observations are considered normal operating conditions. Each processed observation, along with its anomaly label, is stored in the database for future analysis and adaptive model learning.

2) *Drift Detection Module*

Industrial machines experience gradual changes in operating behavior due to wear, aging, environmental conditions, and operational variations. These changes may alter the statistical characteristics of sensor data over time, resulting in concept drift.

To address this challenge, the proposed framework continuously monitors the incoming sensor stream through an independent drift detection pipeline operating in parallel with the anomaly detection process. Only normal operating observations are considered for drift analysis, ensuring that abnormal events do not influence the detection of genuine behavioral changes.

When the drift detection module identifies persistent changes in the normal operating behavior for a predefined number of consecutive observations, the framework concludes that the machine characteristics have evolved and initiates adaptive model retraining.

a) Adaptive Model Retraining: To confirm whether it is actually concept drift or a small fluctuation, the proposed framework receives the sensor data and uses ADWIN to detect data drift. The number of times the data drift must persist before it is concluded to be concept drift is configurable. If the data drift exists for the predefined number of times, the framework confirms it as concept drift. The existing machine model is then retrained to adapt to the updated operating conditions using the data stored in the database. The retraining is performed only using the normal machine data from the database. This enables subsequent anomaly detection to utilize the adapted model.

Adaptive model retraining in the proposed framework supports continuous improvement by retraining the model using labeled sensor observations generated during anomaly detection. Consequently, the framework not only enables real-time anomaly detection but also progressively improves its understanding of machine behavior throughout the operational lifecycle.

III. IMPLEMENTATION

The proposed framework is implemented using a modular architecture consisting of independent processing components for database management, message streaming, anomaly detection, and web service integration. The implementation follows a producer–consumer architecture where real-time sensor data are continuously processed for anomaly detection.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

A. Initialization

This part of the implementation trains the global model (Autoencoder) using training data from a CSV file, stores the model as an .h5 file, and stores the metadata. The model metadata table stores information related to the trained global models, including the machine type, model scope, model name, model location, threshold location, and model status. The entire framework is implemented using multiple threads: one thread handles the database connection pool, one thread handles anomaly detection, and another handles the adaptive training pipeline.

IV. EXPERIMENTAL SETUP

In this phase of implementation, the system is not integrated with any direct IoT sensor. Instead, a data simulation Python script is used to generate data as a JSON string and post it to Kafka, as it would be from real-time sensor data. The generated data follows the format of the “Smart Manufacturing IoT-Cloud Monitoring Dataset” downloaded from Kaggle. This data is posted to the Kafka topic “machine-data.”

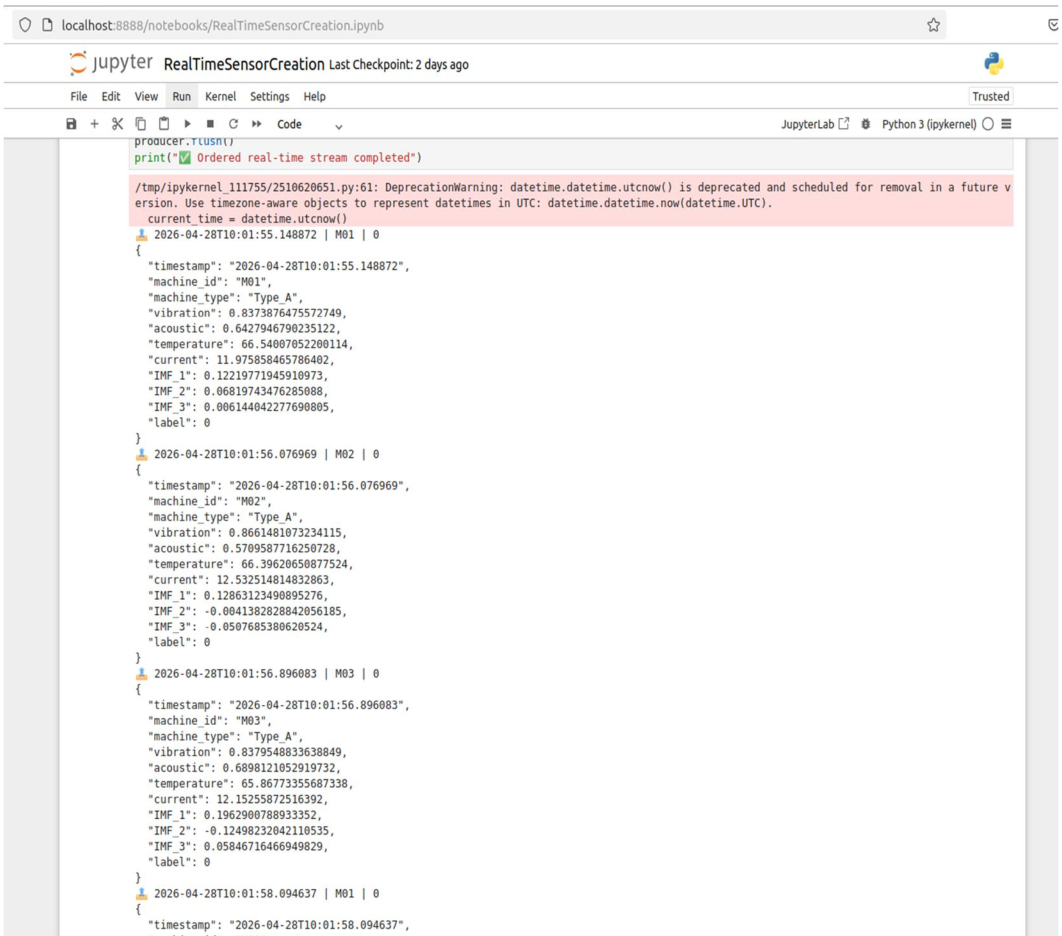


Fig. 3. Data simulation and streaming setup.

A. Database Management

The dataset consists of three machine instances: M01, M02, and M03. To identify machines exhibiting similar operational characteristics for transfer learning, clustering was performed using overlapping sensor data. The clustering results indicated that M01 and M03 share similar behavior. Based on this observation, M01 was selected to train the global model, representing the generalized behavior of the machine category. Subsequently, transfer learning was applied to develop a machine-specific model for M03, allowing the model to inherit the generalized knowledge learned from M01 while adapting to the unique characteristics of M03. The effectiveness of transfer learning was evaluated using four experimental scenarios: (i) before transfer learning using only M03 data, (ii) before transfer learning using the combined M01 and M03 data, (iii) after transfer learning using only M03 data, and (iv) after transfer learning using the combined M01 and M03 data.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

V. RESULTS AND ANALYSIS

A. Before Transfer Learning

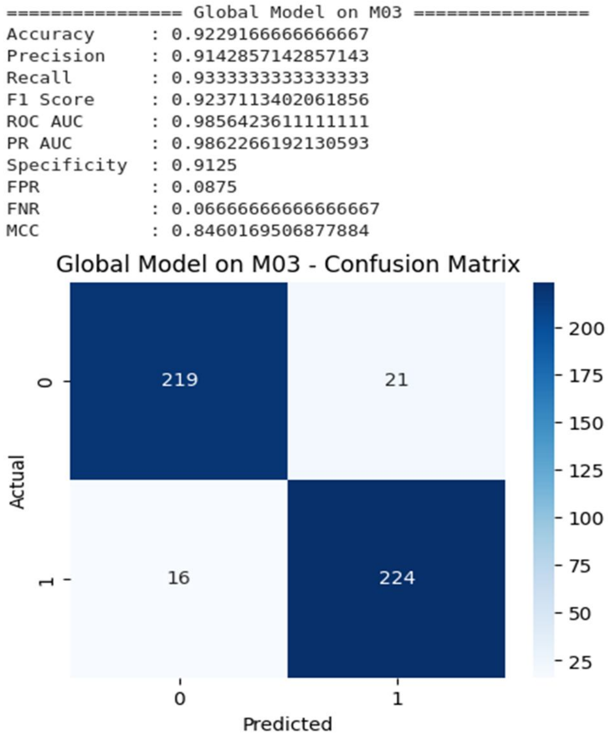


Fig. 4. Performance before transfer learning.

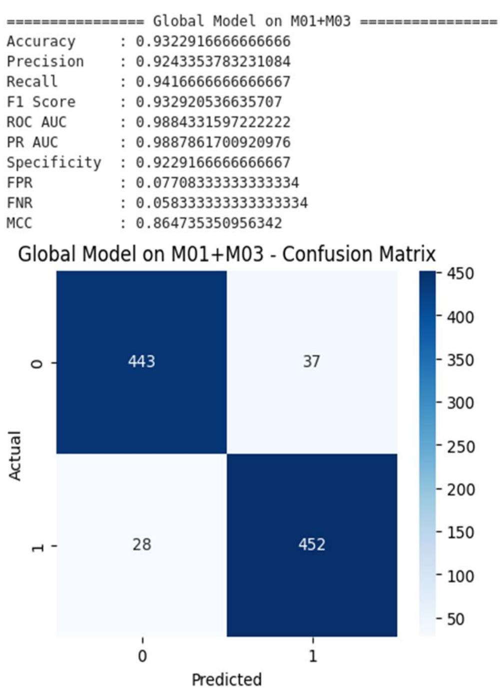


Fig. 5. Performance before transfer learning (continued).

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

B. After Transfer Learning

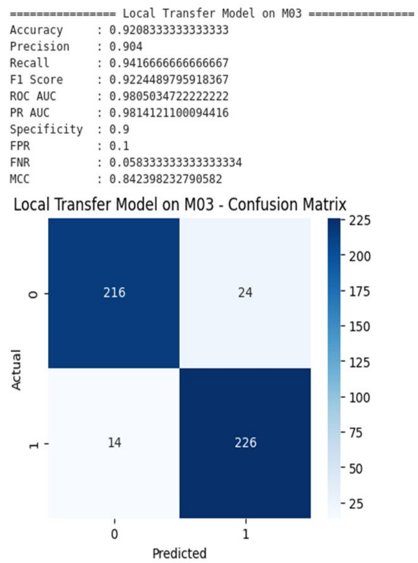


Fig. 6. Performance after transfer learning.

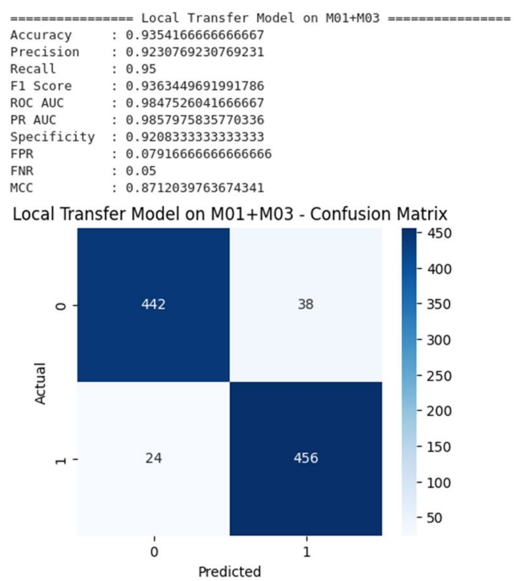


Fig. 7. Performance after transfer learning (continued).

It uses a connection pool of reusable database connections to the PostgreSQL database. These connections are used to store model metadata and streaming sensor data after labeling it using the anomaly detection module.

The implementation maintains two primary database tables. The model metadata table stores information related to the trained global models, including the machine type, model scope, model name, model location, threshold location, and model status. During anomaly detection, the application retrieves the corresponding model information from this metadata repository and loads the appropriate model dynamically. The sensor_stream table stores the results of the anomaly detection module.

Real-Time Data Streaming: The implementation uses Apache Kafka as the streaming middleware. A Kafka consumer [content to be completed].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

C. Comparison of Before and After Transfer Learning:

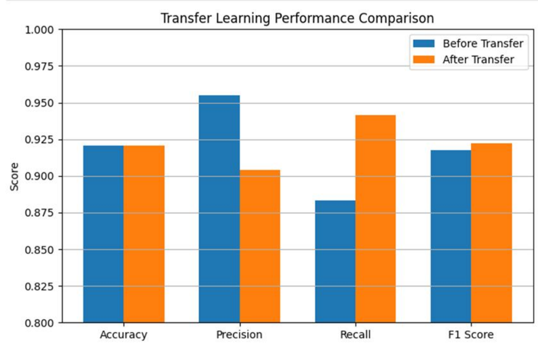


Fig. 8. Comparison of results before and after transfer learning.

VI. CONCLUSION

Adaptive transfer-learning-based anomaly detection has been implemented, and various test cases were performed to evaluate the generalization of the model. The model was better able to identify machine-specific anomalies after adaptive transfer learning. The effectiveness of the proposed transfer learning approach was evaluated using machine-specific data. After applying transfer learning, the model achieved an accuracy of 92.08%, precision of 90.40%, recall of 94.17%, and an F1-score of 92.24%. Among these metrics, recall showed the most significant improvement, increasing from 88.3% to 94.1%. This demonstrates that the adaptive transfer-learned model became more effective in identifying anomalies specific to a particular machine, while requiring only a limited amount of machine-specific training data. Since predictive maintenance prioritizes detecting actual faults over minimizing false alarms, the improved recall indicates that the proposed adaptive transfer learning approach enhances machine-specific anomaly detection and reduces the likelihood of missing critical anomalies.

VII. FUTURE WORK

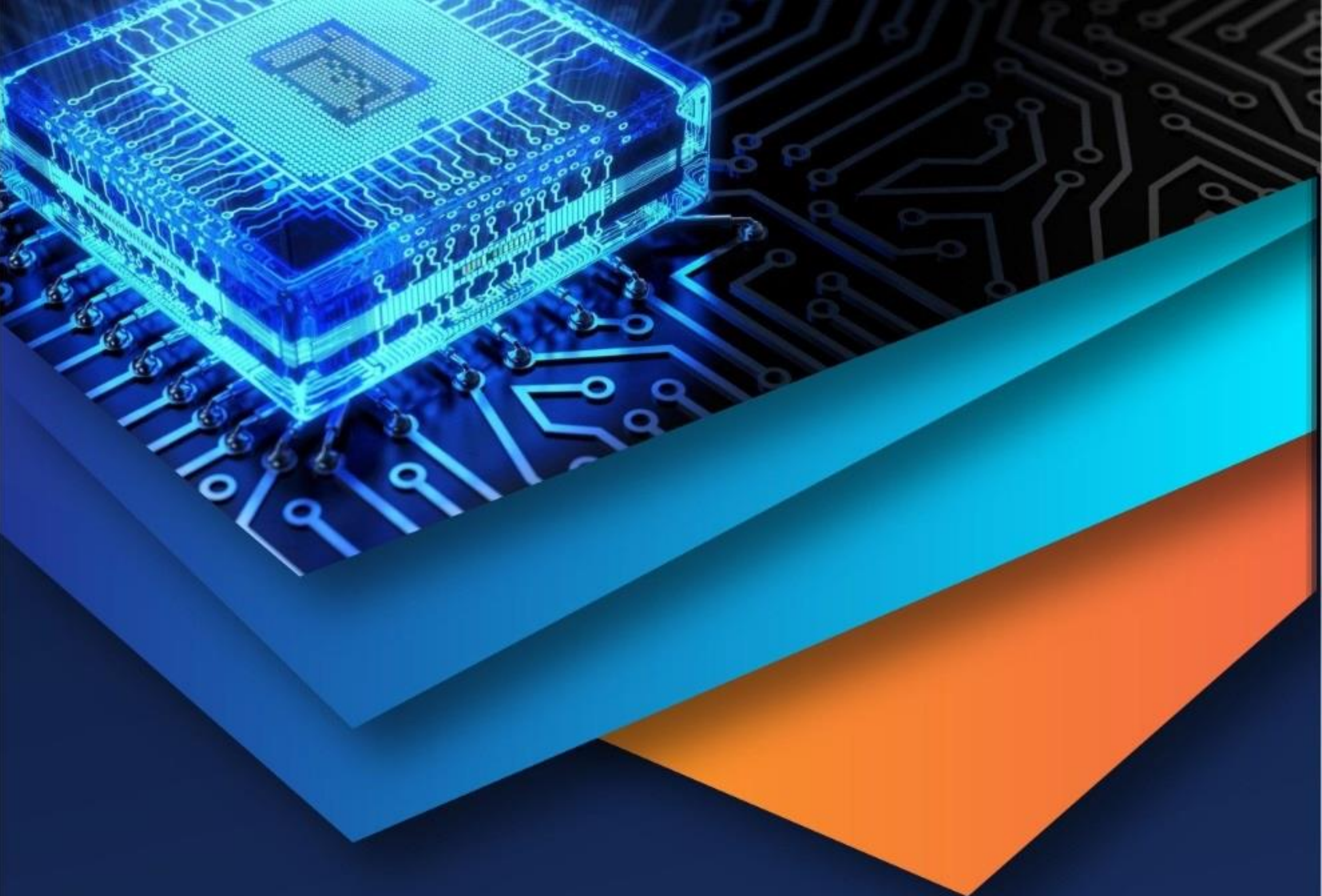
The proposed framework has certain limitations that can be addressed in future work. Although transfer learning has been successfully implemented using the Conditional Autoencoder in the current framework, its applicability to more advanced deep learning models has not been investigated. Future research can explore the integration of transfer learning with advanced anomaly detection models and evaluate its effectiveness in comparison with the proposed approach.

Furthermore, the current implementation has been validated using an offline dataset in a remote processing environment. The framework has not yet been evaluated using real-time streaming data from industrial machines. Therefore, future work will focus on deploying the proposed framework in a real-time Industrial IoT environment and validating its performance under practical operating conditions. In addition, the proposed architecture is designed to support cloud-assisted deployment, which will be explored and evaluated in future implementations.

REFERENCES

[1] R. Saleh, K. P. Dinh, B. Villányi, and T.-S. Hy, “Self-Evolving Multi-Agent Network for Industrial IoT Predictive Maintenance,” arXiv preprint arXiv:2602.16738, 2026.
[2] H. Choi, D. Kim, J. Kim, and P. Kang, “Explainable anomaly detection framework for predictive maintenance in manufacturing systems,” Applied Soft Computing, vol. 125, p. 109147, 2022.
[3] L. Bowden, Q. Chu, B. Cena, K. Ohno, B. Parney, D. Sharma, and M. Takeori, “Machine Failure Detection Based on Projected Quantum Models,” arXiv preprint arXiv:2601.15641, 2026.
[4] A. Giannoulidis and A. Gounaris, “A context-aware unsupervised predictive maintenance solution for fleet management,” Journal of Intelligent Information Systems, vol. 60, pp. 521–547, 2023.
[5] S. O. Alhuqayl, A. T. Alenazi, H. A. Alabduljabbar, and M. A. Haq, “Adaptive machine learning models for predictive maintenance in industrial internet of things systems,” Scientific Reports, 2026.
[6] S. F. Chevtchenko, M. C. M. dos Santos, D. M. Vieira, R. L. Mota, E. Rocha, B. V. Cruz, D. Araújo, and E. Andrade, “Predictive Maintenance Model Based on Anomaly Detection in Induction Motors: A Machine Learning Approach Using Real-Time IoT Data,” arXiv preprint arXiv:2310.14949, 2023.
[7] H. Zheng, A. R. Paiva, and C. S. Gurciullo, “Advancing from Predictive Maintenance to Intelligent Maintenance with AI and IIoT,” arXiv preprint arXiv:2009.00351, 2020.
[8] W. Wang, H. Zhang, D. Yang, and Y. Xiang, “Deep-learning-enabled predictive maintenance in industrial internet of things: Methods, applications, and challenges,” IEEE Systems Journal, vol. 17, no. 2, pp. 2602–2615, 2022.

[9] A. L. Alfeo, S. Cimino, and G. Vaglini, “Using an autoencoder in the design of an anomaly detector for smart manufacturing,” Pattern Recognition Letters, vol. 136, pp. 271–278, 2020.
[10] B. Steenwinckel, J. De Paepe, S. De Turck, and F. Ongenae, “FLAGS: A methodology for adaptive anomaly detection and root cause analysis on sensor data streams by fusing expert knowledge with machine learning,” Future Generation Computer Systems, vol. 123, pp. 30–45, 2021.
[11] M. Canizo, I. Triguero, A. Conde, and E. Onieva, “Multi-head CNN-RNN for multivariate time series anomaly detection: An industrial case study,” Neurocomputing, vol. 363, pp. 246–260, 2019.
[12] Z. Ding and M. Fei, “An anomaly detection approach based on isolation forest algorithm for streaming data using sliding window,” IFAC Proceedings Volumes, vol. 46, no. 20, pp. 12–17, 2013.
[13] U. W. Hore and D. G. Wakde, “Intelligent Predictive Maintenance for Industrial Internet of Things Using Machine Learning Approach,” in Proc. Int. Conf. Intelligent Cyber Physical Systems and IoT, Springer, 2022.
[14] R. Anandan, S. Gopalakrishnan, S. Pal, and N. Zaman, Industrial Internet of Things (IIoT): Intelligent Analytics for Predictive Maintenance. Wiley, 2022.
[15] D. Fernández-Francos, B. Martínez-Rego, O. Fontenla-Romero, and A. Alonso-Betanzos, “Automatic bearing fault diagnosis based on one-class classification using vibration signals,” Computers in Industry, vol. 125, 2021.
[16] D. Galatro, M. Shroff, and C. H. Amon, “Adaptive Transfer Learning Strategy for Predicting Battery Aging in Electric Vehicles,” MDPI, Jan. 2025.
[17] Kaggle, “Industrial IoT Dataset (Synthetic).” [Online]. Available: https://www.kaggle.com/datasets/canozensoy/industrial-iot-dataset-synthetic



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY