International Journal on Engineering Artificial Intelligence Management, Decision Support, and Policies
Vol. 03, Iss. 02, S. No. 01, pp. 01--15, 2026
ISSN (Online): 3048-8788

Received: 17 May 2026, Accepted: 07 August 2026, Published: 31 August 2026

Digital Object Identifier: https://doi.org/10.63503/j.ijaimd.2026.264

Research Article

Cloud–Edge Integrated Machine Learning Framework for Real-Time Monitoring of Cyber-Physical Systems with IoT Sensor Networks

Jayan Sharma

Tehri Hydro Development Corporation, Institute of Hydropower Engineering and Technology, Tehri, Uttarakhand, India

Jayanpetwal@gmail.com

ABSTRACT

The CPS, in combination with the IoT sensor networks, has experienced massive growth, which results in massive data generation per second that presents extreme challenges to latency, scalability, and efficient data processing. The current paper presents a cloud-edge-integrated machine learning system for real-time monitoring of CPS environments. The suggested system combines IoT data collection, edge processing, and cloud-based model optimisation to enable fast, intelligent decision-making. Edge computing reduces communication overhead by performing local inference, while the cloud provides large-scale analytics and model training. The evaluation of the framework is conducted on a dataset of 10,000 sensor records that represent industrial parameters such as temperature, pressure, and vibration. The experimental findings showed that prediction accuracy was 91.2%, processing efficiency was 88.5%, and stability was 0.86, with a much lower latency of 205 ms. The overall performance index of 0.88 indicates that the computer's responsiveness, scalability, and efficiency have improved equally. The comparative analysis demonstrates that the proposed approach is significantly superior to traditional and standalone machine learning models, which is why it can be widely applied in real-time CPS monitoring applications.

Keywords: *Cloud -Edge Computing, Machine Learning, Cyber-Physical Systems (CPS), IoT Sensor Networks, Real-Time Monitoring, Data Analytics, Edge Intelligence.*

1 Introduction and Research Background

The rapid development of Cyber-Physical Systems (CPS), supported by IoT sensor networks, has had a dramatic impact on contemporary industrial and intelligent infrastructure settings. Such systems facilitate a seamless integration of physical processes and computational intelligence by enabling continuous data acquisition and analysis. Large amounts of real-time data generated by IoT sensors deployed in CPS environments include parameters such as temperature, pressure, vibration, and system state. This information is critical to carrying out predictive maintenance, detecting faults, and performing intelligent monitoring. Nonetheless, the bursting sensor data can be associated with several challenging latency, scalability, and effective process issues, especially centralized cloud-based architectures [1], [2].

Conventional cloud-related services have been based on the utilization of edge devices that transmit the information to servers that are situated remotely to be processed and stored. Cloud computing is known to be high-performance and scalable; however, it suffers from significant communication delays and bandwidth limitations, and therefore cannot be applied to time-sensitive CPS applications. In addition, the growing need to make decisions locally and in real time in industrial automation and smart systems implies the need for more localised, fast cross-processing mechanisms. The adoption of machine learning (ML) techniques has gained significant popularity in order to improve the accuracy of prediction and system intelligence [2], but they can introduce many computational costs and higher response times when deployed in centralized setups [4], [5].

In an effort to overcome those shortcomings, edge computing has become a promising paradigm that enables data processing closer to the data's source. Edge computing decreases the latency and enhances the responsiveness by executing localized analytics. In recent studies, hybrid cloud-edge architectures that combine the benefits of both paradigms have been proposed; with these architectures, it is possible to allocate workloads more effectively and enhance system performance. The incorporation of machine learning into these kinds of structures further increases CPS's ability to make intelligent observations and adaptive decisions. Nevertheless, issues such as resource limitations at the edge, model complexity, and effective coordination between the cloud and edge layers remain open research topics [6], [7], [8], [9].

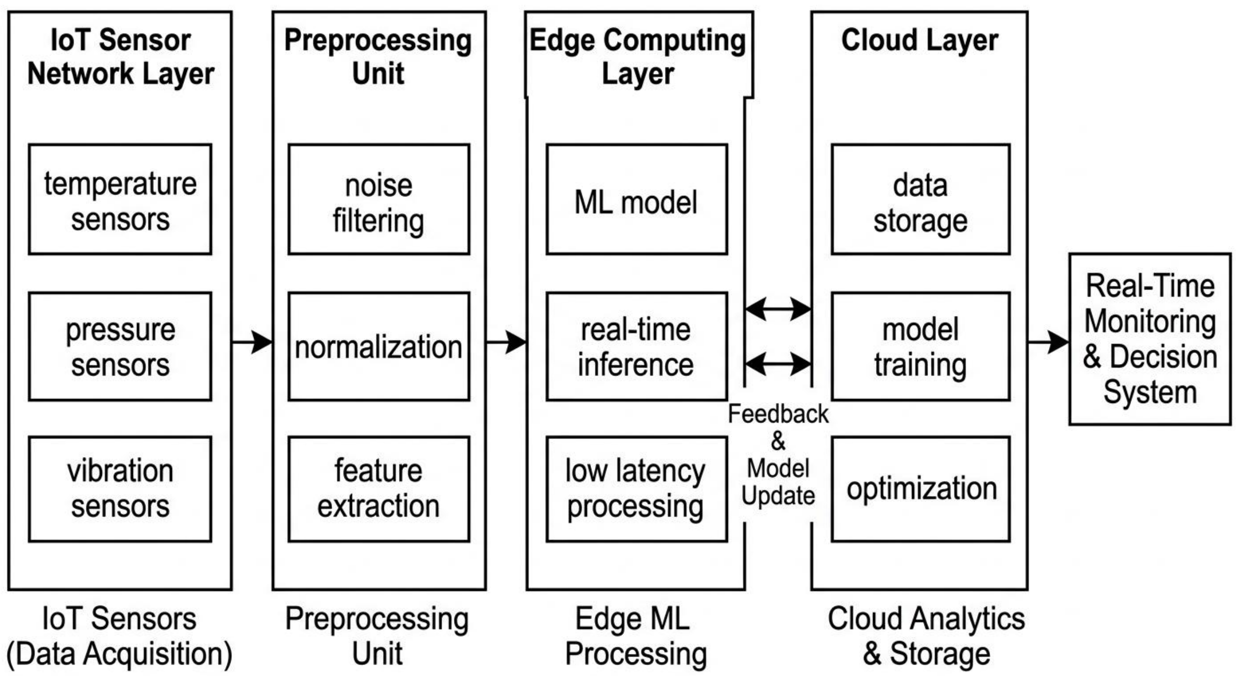

Fig.1. Cloud-edge ML system Overview.

Fig.1 shows the conceptual design of the suggested cloud-edge built-in machine learning platform of real-time monitoring of CPS. It demonstrates how to process data gathered by an IoT sensor network using preprocessing modules and machine learning models deployed at the edge layer. Cloud layer will aid in scaling of data storage, training of models and optimization of the system and a continuous feed-back loop will enable adaptive workload and effective coordination of edge and cloud elements.

The rest of this paper will be organized as follows: Section 2 offers the review of the related literature in CPS monitoring, machine learning, and cloud-edge system. The problem statement and objectives of the research are defined in Section 3. Section 4 covers the intended methodology, comprising the system's architecture and algorithmic framework. The details of the implementation and experiment setting are presented in Section 5. Section 6 talks about the performance evaluation and results. Lastly, Section 7 will be the conclusion of the paper and the future research directions are presented.

2 Related Research on AI Governance and Cybersecurity Compliance

The growth of the use of Cyber-Physical Systems (CPS) and IoT sensor networks has triggered research of huge magnitude in real-time monitoring and smart data analytics. The initial tools were mainly rule- and threshold-based monitoring systems, which were easy to use but lacked the flexibility needed in a dynamic environment. These systems had not been able to effectively process big real-time data and they failed to make correct predictions when the conditions of operation were different [10], [11]. With the development of industrial systems, the need for more intelligent and scalable solutions became apparent.

As cloud computing improved, the popularity of centralized data analytics frameworks was on the increase because of the high calculations and storage. The systems based on the clouds allowed processing in large-scale and in-depth analytics though enormous delays occurred through the constant transmission of data to remote servers in IoT devices. This latency rendered them unsuitable on time-sensitive CPS e.g. industrial automation and real time fault detection. Besides, performance and responsiveness of the system were also affected by bandwidth limitation and network congestion [12], [13].

The way in which machine learning (ML) methods can be used to address these constraints was proposed to improve predictive analytics and anomaly detection in CPS settings. Decision trees, support vector machines, neural networks were some of the methods that enhanced accuracy in prediction and intelligence of the systems. In spite of these benefits, ML-based solutions frequently have a high cost to compute and very large training times hence challenging to efficiently deploy on real time and resource-wary settings. Furthermore, the incomplete integration of the ML models and the distributed systems architecture meant they could not be used in large-scale CPS deployment [14], [15].

Table 1 is the comparative analysis of the current methods and it is emphasized that traditional and cloud based methods have a weakness of latency and the ML and hybrid models have greater accuracy but are more complex.

Table 1: Comparative Analysis of Existing Approaches in CPS Analytics

Framework Type	Core Focus	Key Limitation	Reference
Rule-based Systems	Threshold-based monitoring	Low adaptability and poor scalability	[9]
Rule-based CPS Models	Static decision rules	Inefficient in dynamic environments	[10]
Cloud-based Analytics	Centralized data processing	High latency and bandwidth dependency	[11]
Distributed Cloud Systems	Large-scale CPS data handling	Communication delays	[12]
ML-based Models	Predictive analytics	High computational complexity	[13]

Deep Learning Models	Complex pattern recognition	High training time and resource demand	[14]

3 Problem Statement & Research Objectives

The rapid development of Cyber-Physical Systems (CPS) combined with IoT sensor networks has resulted in the emergence of large volumes of real-time data, causing considerable problems with efficient data processing, latency reduction, and system scalability. When compared to the classical cloud-based system, the former results in high delays because of continuous transmission of the data between distributed sensors and centralized servers, which has a detrimental effect on real-time monitoring and decision-making. Despite the accuracy of prediction in the machine learning methods, they cannot be used effectively in the centralized or uncoordinated environment due to the fact that they contribute to the intensification of the computational costs and poor resource use. Also, edge computing can provide low-latency processing, but has limited computational resources and cannot be aware of the entire system. Lack of appropriate coordination between the edge and cloud layers also leads to an inefficient distribution of workload and low system performance.

Research Objectives:

- The purpose of the study is to minimize the latency.
- awesome future uses: e.g. predictive maintenance for tech-heavy equipment like trains, automated drones, or vehicle dashboard of the future, etc.
- define the resource utilization by intelligently distributing the computation at edge and cloud layers.
- It also aims at offering increased accuracy in prediction and system efficiency using adaptive machine learning models and maintaining stable and responsive system behavior due to dynamism.

4 Proposed AI-Enabled Hardware–Software Co-Design Methodology

The suggested cloud-edge integrated machine learning device will facilitate effective and real-time monitoring of Cyber-Physical Systems (CPS) with low latency and IoT sensor networks in an efficient manner. There are three significant elements incorporated in the framework that comprise the acquisition of data based on the IoT, edge-level useful processing, and the analytics and optimization in the clouds. The first entails the constant acquisition of real-time information by the use of structured sensors attached to the physical world which is relayed to the edge layer to be preprocessed and inferred. The support of the large-scale storage, the model training, and adaptive optimization is offered under the cloud layer. This interdisciplinary strategy can be guaranteed to provide low latency, the effect of resource efficiency, and enhanced performance.

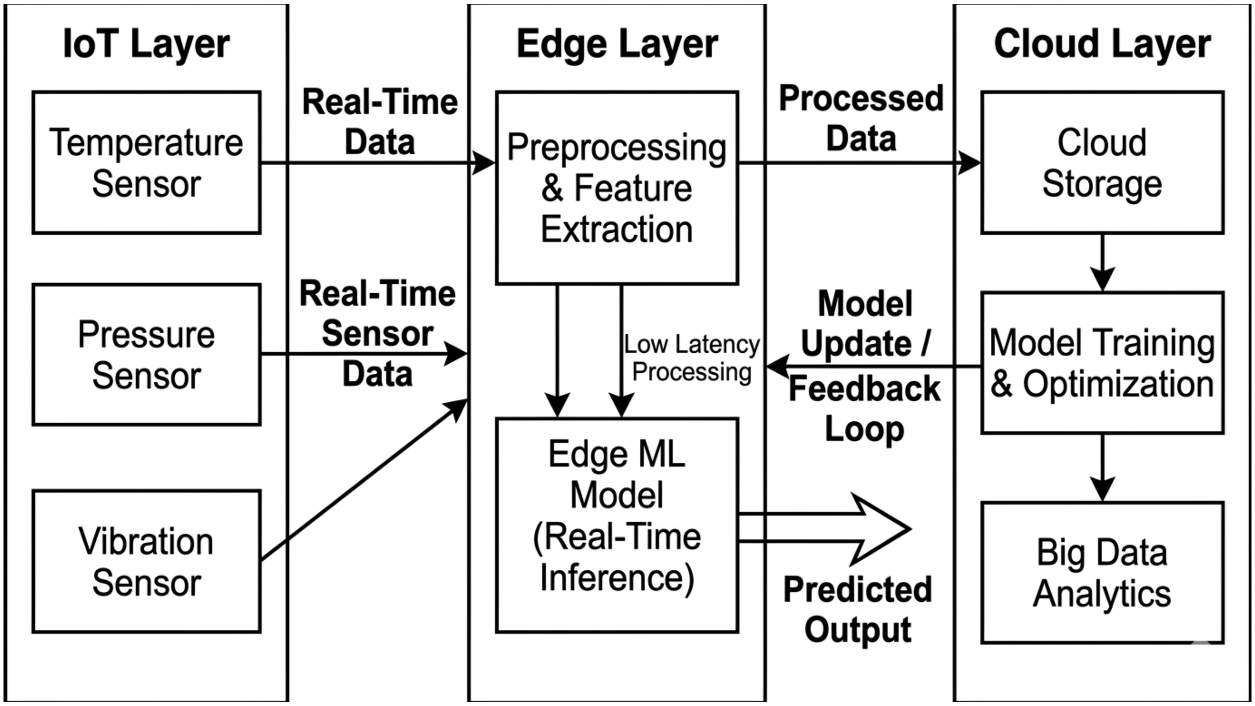

Fig.2. Cloud–edge system architecture

Fig. 2 illustrates the architecture of the proposed framework, showing the interactions among IoT sensor networks, preprocessing modules, edge-based machine learning models, and cloud infrastructure. The architecture highlights bidirectional data flow, where real-time data is processed at the edge and further refined in the cloud. A feedback loop enables continuous system optimization and adaptive learning under dynamic CPS conditions. The data acquisition process from IoT sensors is mathematically modelled as Eq. (1):

$$D(t) = S(t) + N(t) \quad (1)$$

where $D(t)$represents the observed sensor data at time t, $S(t)$is the true signal, and $N(t)$denotes measurement noise. Equation (1) establishes the relationship between raw sensor input and noise components.

The preprocessing and feature extraction stage transforms raw data into structured features:

as Eq. (2):

$$F = \phi\big(D(t)\big) \quad (2)$$

where $F$is the extracted feature vector and $\phi(\cdot)$represents the preprocessing function, including filtering, normalization, and feature mapping. Equation (2) depends on the output of Eq. (1), linking raw data to processed features.

The edge-based machine learning prediction model is defined as Eq(3):

$$y^{\wedge} = f(F, \theta) \qquad (3)$$

where $\hat{y}$is the predicted output, $f(\cdot)$is the trained ML model, and θrepresents model parameters. Equation (3) uses features from Eq. (2) to generate predictions.

To distribute computational tasks efficiently between edge and cloud, workload allocation is modeled as Eq. (4):

$$W = W_e + W_c \quad (4)$$

where $W$is the total workload, $W_e$is the workload handled at the edge, and $W_c$is the workload processed in the cloud. Equation (4) ensures balanced computation across system layers.

The latency of the system is expressed as Eq. (5):

$$L = L_e + L_t + L_c \quad (5)$$

where $L$is total latency, $L_e$is edge processing delay, $L_t$is transmission delay, and $L_c$is cloud processing delay. Equation (5) directly depends on the workload distribution in Eq. (4).

The energy-aware system efficiency is modelled as Eq. (6):

$$\eta = \frac{A}{E_h + E_s} \quad (6)$$

where $\eta$is system efficiency, $A$is the prediction accuracy from Eq. (3), $E_h$is hardware energy consumption, and $E_s$is software computational energy. Equation (6) links accuracy and energy consumption. System stability is represented as Eq. (7):

$$S_t = 1 - \sigma(L) \quad (7)$$

where $S_t$denotes system stability and $\sigma(L)$is the variation (standard deviation) of latency from Eq. (5). Lower latency variation leads to higher stability.

Finally, the overall performance index is calculated as Eq. (8):

$$P = \alpha A + \beta \eta - \gamma L \quad (8)$$

P is the performance index, where A is the accuracy, = 1/2 efficiency as in equation (6), Latency Luis in equation (5), and =, 9 and the weighting coefficients of 90, 100 are 0.2, 0.8, and 1 respectively. All past metrics are combined in equation (8) to assess the performance of the entire system. All these equations separately build a close connection between the IoT data acquisition, edge intelligence, and cloud optimization. The framework also provides a high level of coordination among the system components, enabling real-time monitoring with better accuracy, reduced latency, and improved scalability in CPS environments.

Algorithm 1: Cloud–Edge Integrated Machine Learning Framework for Real-Time CPS Monitoring

Input: Real-time IoT sensor data stream, θ: Machine learning model parameters, R_e: Edge resources (CPU, memory, energy) , R_c: Cloud resources (storage, computation power) , λ: Edge–cloud workload distribution factor

Output: Predicted monitoring output, A: Prediction, L: Total system latency, P: Overall performance indexData.

- Data Acquisition: Collect continuous real-time data $D(t)$from distributed IoT sensor networks in the CPS environment.

- Preprocessing and Feature Extraction: Apply the preprocessing function to remove noise and extract relevant features F.
- Edge-Based Prediction: Use a trained ML model with parameters $\theta$to generate a prediction $\hat{y}$at the edge layer.
- Workload Distribution: Allocate computational tasks between edge and cloud using the distribution factor λ.
- Cloud Processing and Model Update: Perform large-scale analytics, model training, and parameter updates in the cloud.
- Latency Computation: Calculate total system latency Lconsidering edge processing, transmission delay, and cloud processing.
- Efficiency Evaluation: Compute system efficiency ηbased on prediction accuracy and energy consumption.
- Performance Index Calculation: Evaluate overall performance index Pusing accuracy, efficiency, and latency.
- Adaptive Feedback Mechanism: Update system parameters dynamically using feedback from cloud to edge for optimization.
- Output Results: Return predicted output $\hat{y}$along with performance metrics (P).

5 Experimental Design and Implementation

The experiment aims at analyzing the efficiency of the suggested cloud-edge integrated machine learning system through an experiment involving a real-time Cyber-Physical System (CPS) system with IoT sensor networks. The simulated setting of the system is a scenario of an industrial-level monitor in which real-time data is produced by the distributed sensors that measure the temperature, pressure, vibration, device status and network delay. A sample size of 10,000 records is employed in order to find the realistic experience of CPS in different workloads and states of operation.

The dataset will be split into 70 percent of training (7, 000 samples) and 30 percent testing (3, 000 samples) to test the performance of the machine learning model. The data is further enhanced by means of preprocessing to improve the quality and accuracy of the data and models by filtering the noise, normalization and extraction of features before feeding the data into the system. The cloud layer carries out model training, optimization, and long-term storage, whereas the edge layer indeed makes inference in the real-time ensuring a minimum of the latency.

Its implementation is based on the integration of IoT sensors, an edge computing unit and cloud infrastructure that allows distributing the workload and making it effective. The edge incorporates locally time-sensitive data, which minimizes the delays in communication, but the calculations are performed by the cloud which is computationally intensive. Key metrics that are used to measure the system performance are the accuracy of prediction, efficiency of the system in processing and other significant metrics such as latency, system stability, and overall performance index. Scalability is also checked associated with changing the input data size and system response.

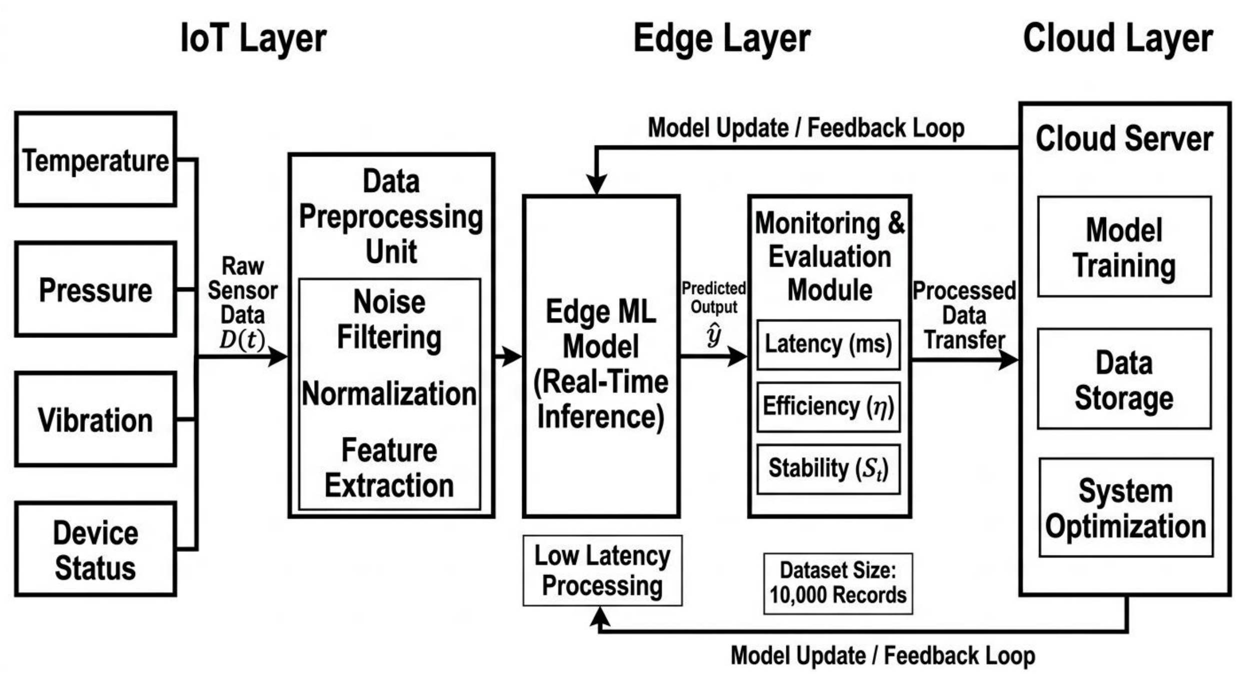

Fig.3. Cloud-edge CPS monitoring experimental set-up.

Experimental architecture The proposed framework has experiment architecture represented by Fig.3. It denotes how the data at the IoT sensor networks passes to the preprocessing unit and then machine learning at the edge layer. The data obtained after the processing is sent to the cloud where they undergo additional analysis and model updating and storage. The latter includes the usage of a feedback loop which allows optimizing both edge and cloud components in an adaptive way so that they maintain similar performance regardless of the conditions of dynamic CPS.

Table 2 shows the experiment findings and assessment configuration of the proposed cloud-edge integrated machine learning architecture of real time CPS monitoring. It describes the entire process of data processing, beginning with activity that involves the acquisition of the data via an IoT sensor network up to the ultimate calculation of the performance index. All the tasks are allotted to the designated modules that perform that purpose in the system architecture.[15]-[16]

Table 2: Experimental Tasks and Evaluation Configuration

Experimental Task	Module Used	Output Result	Dataset Size
Data Acquisition	IoT Sensor Network	Raw real-time data stream (D(t))	10,000
Data Preprocessing	Feature Extraction Unit	Cleaned feature vectors (F)	10,000
ML-Based Prediction	Edge ML Model	Predicted output (\hat{y})	10,000
Workload Distribution	Edge–Cloud Manager	Optimized task allocation	10,000

Latency Measurement	Monitoring Module	Processing delay (ms)	10,000

6 Results and Performance Evaluation

The proposed cloud-edge integrated machine learning framework performance is tested against other statistical frameworks of machine learning such as traditional, machine learning based, and hybrid ones taking the same experimental conditions. The key performance goals, which are taken into account in the evaluation, include accuracy of prediction, efficiency, latency of the processing, stability of the system, and the index of overall performance. The findings prove that the proposed framework is more effective because of effective performance in the distribution of workloads and processing of edges in real-time.

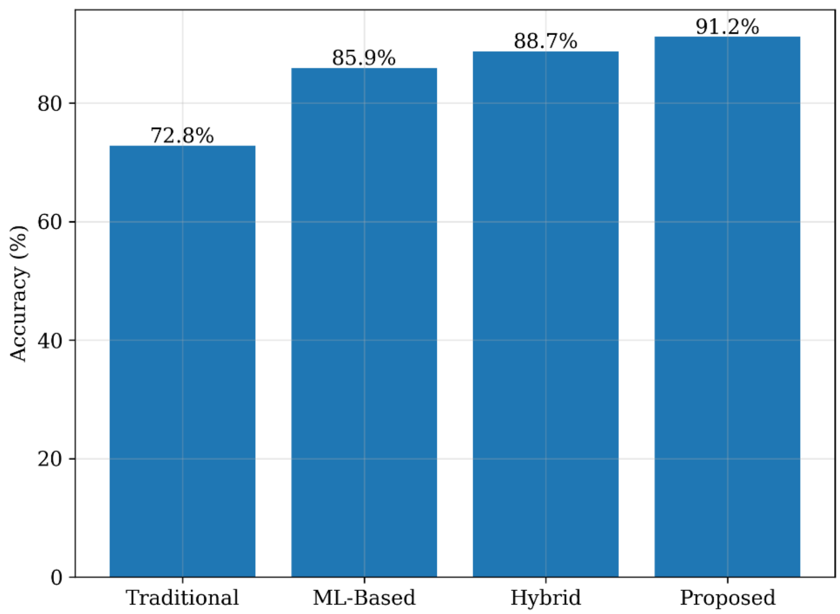

Fig.4. Comparison of models in terms of their accuracy.

Fig.4 indicates that the proposed framework has the greatest accuracy of prediction of 91.2 as compared to traditional models of 72.8 and 85.9 and hybrid approaches of 88.7. This has been due to good extraction of features and edge based inferences.

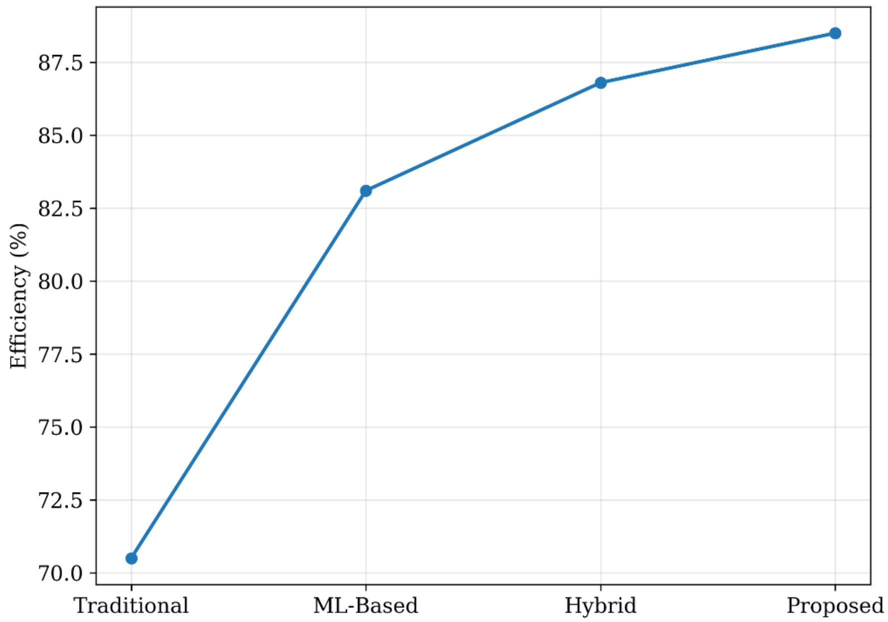

Fig.5. Efficiency Comparison of Different Models

As shown in fig.5, the proposed system achieves the highest processing efficiency of 88.5, which is better than the traditional, ML-based, and hybrid models (70.5, 83.1, and 86.8), which denote inefficient resource utilisation.

Table 3 indicates that the proposed model will be the most accurate (91. 2%), most efficient (88.5%), has the minimum latency (205 ms) and best overall performance index (0.88) as compared to other models.

Table 3: Performance Comparison of Different Models

Model	Accuracy (%)	Efficiency (%)	Latency (ms)	Stability	Performance Index
Traditional	72.8	70.5	325	0.66	0.69
ML-Based	85.9	83.1	250	0.79	0.80
Hybrid	88.7	86.8	225	0.83	0.84
Proposed	91.2	88.5	205	0.86	0.88

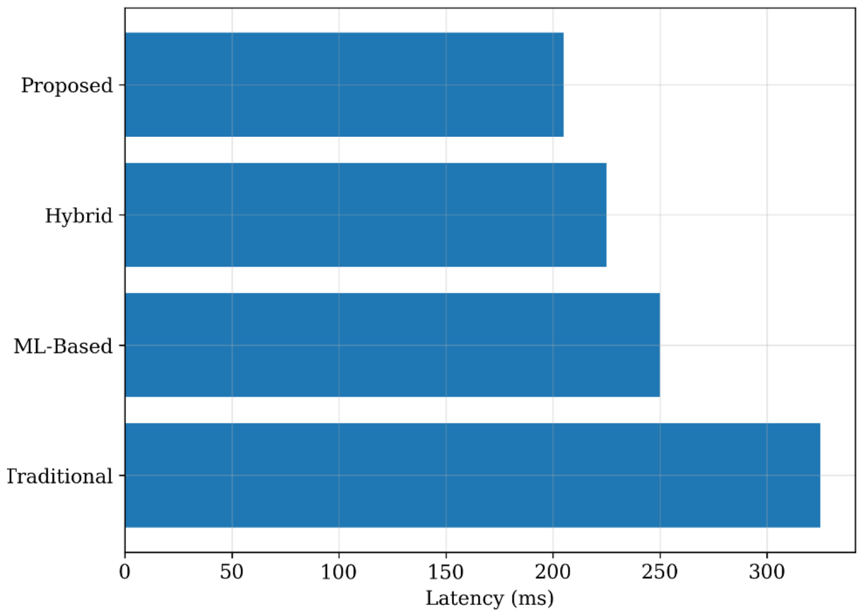

Fig.6. Latency Analysis of Different Models

Fig. 6 shows that the presented framework reduces computational time by 205 ms, which is much lower than traditional (325 ms), ML-based (250 ms), and hybrid (225 ms) models due to localised edge processing.

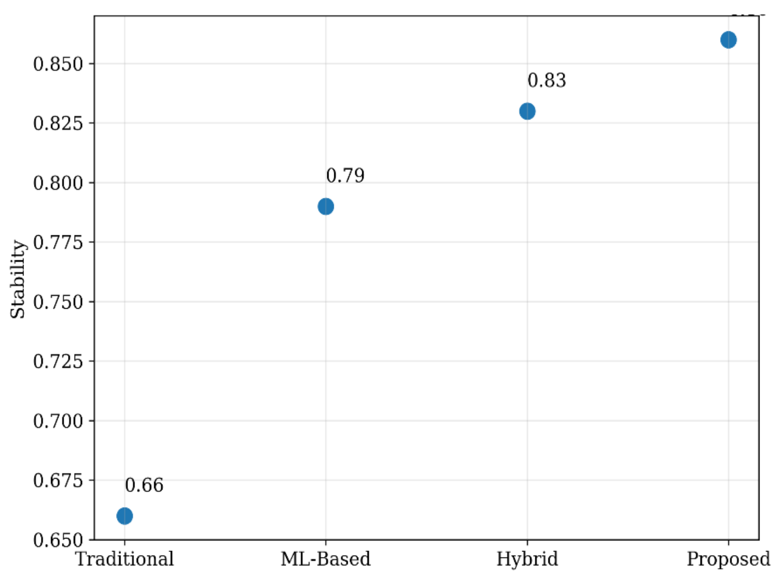

Fig.7. Stability analysis of models

Fig. 7 shows that the stability score of the proposed model is 0.86, which guarantees the ability to maintain stability during the approach's operation under changing system conditions.

As it is emphasized in Table 4, the framework proposed will be the most stable (0.86), least latent and most powerful data processing (high level) of all the methods.

Table 4: Efficiency and Latency Comparison

Model	Stability	Latency (ms)	Data Processing Level
Traditional	0.66	325	Low
ML-Based	0.79	250	Medium
Hybrid	0.83	225	Medium
Proposed	0.86	205	High

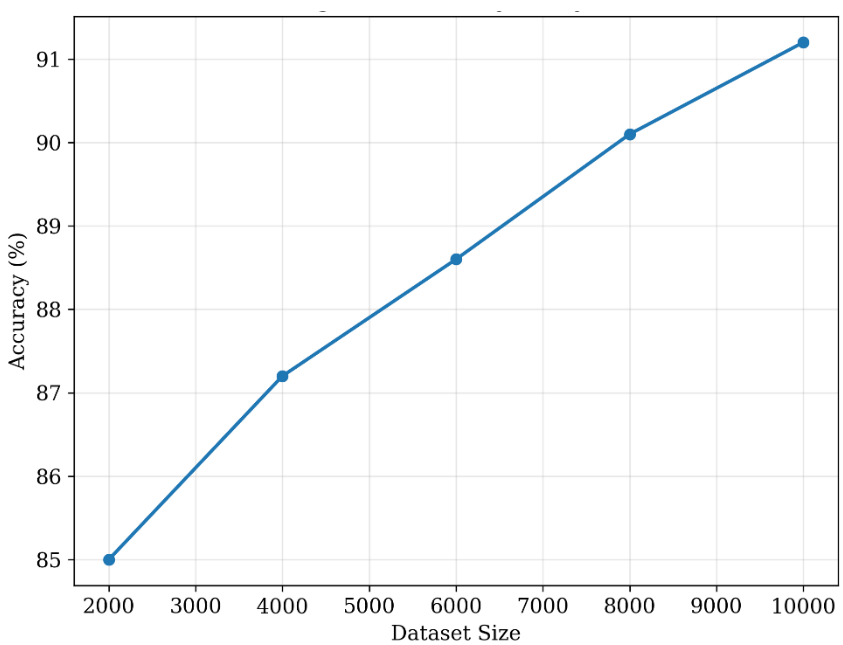

Fig. 8. Scalability Analysis of Proposed Framework

Fig. 8 indicates that accuracy increases with sample size between 2000 and 10,000, signifying the high scalability and strength of the framework.

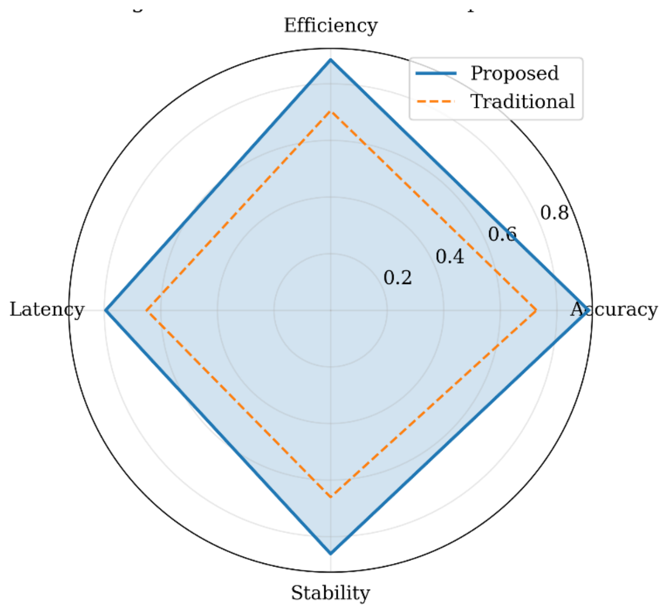

Fig.9. Performance Index Comparison of Different Models

Fig. 9 shows the overall performance index; the proposed framework achieves a balance with traditional (0.69), ML-based (0.80), and hybrid (0.84) at the same index of 0.89, indicating balanced performance.

On the whole, these findings verify that a cloud-edge integrated framework offers higher accuracy, lower latency, greater efficiency, and enhanced scalability, and is therefore well-suited to real-time CPS monitoring.

7 Conclusion and Future Work

In this paper, a cloud-edge-integrated machine learning architecture was proposed for real-time monitoring of Cyber-Physical Systems (CPS) based on IoT sensor networks. The suggested solution is useful for integrating data acquisition and edge-level inference based on IoT measurements, and for optimising the solution in the cloud, addressing issues of latency, scalability, and computational efficiency. An experimental evaluation on a dataset of 10,000 records showed that the framework achieves a high prediction accuracy of 91.2, processing efficiency of 88.5, and system stability of 0.86, as well as a much lower computational latency of 205 ms. The overall performance index of 0.88 indicates that the system is balanced in terms of the trade-off among accuracy, efficiency, and responsiveness. These findings show conclusively that the given framework outperforms the traditional, ML-based, and hybrid frameworks in real-time CPS monitoring.

In the next work, it is possible to expand the framework and use the developed deep learning models, thereby increasing forecasting accuracy above 92. Hardware accelerators (GPUs, FPGAs) can be integrated to achieve a latency of less than 180 ms and be more energy-efficient. Moreover, data privacy can be improved by the adoption of federated learning methods, and distributed intelligence can be used in a number of CPS nodes. Large-scale real-world deployment on datasets exceeding 50,000 samples will further validate the system's scalability and robustness.

References

[1] Hehenberger, P., Vogel-Heuser, B., Bradley, D., Eynard, B., Tomiyama, T., & Achiche, S. (2016). Design, modelling, simulation and integration of cyber physical systems: Methods and applications. *Computers in Industry*, *82*, 273-289. https://doi.org/10.1007/978-3-319-92564-6

[2] Suyal, H., & Singh, A. (2021). Improving Multi-Label Classification in Prototype Selection Scenario. *Computational Intelligence and Healthcare Informatics*, 103-119.

[3] Prakash, R., & Balaji Ganesh, A. (2018, September). Internet of Things (IoT) enabled wireless sensor network for physiological data acquisition. In *International Conference on Intelligent Computing and Applications: Proceedings of ICICA 2018* (pp. 163-170). Singapore: Springer Singapore. DOIhttps://doi.org/10.1007/978-981-13-2182-5_17

[4] Dafflon, B., Moalla, N., & Ouzrout, Y. (2021). The challenges, approaches, and used techniques of CPS for manufacturing in Industry 4.0: a literature review. *The International Journal of Advanced Manufacturing Technology*, *113*(7), 2395-2412. DOIhttps://doi.org/10.1007/s00170-020-06572-4

[5] Wan, A., Zhu, Z., Khalil, A. B., Cheng, X., Jiang, J., Ji, X., ... & Shan, T. (2026). Real-time aero-engine fault diagnosis using 5G edge computing and deep learning. *Measurement*, *257*, 118784. https://doi.org/10.1016/j.measurement.2025.118784

[6] Ayyadurai, M., Vijay, K., Lokesh, K., Nandha Kumar, P., & Hari Sainath, C. (2026). Cloud-to-Thing Continuum: Connecting IoT Devices and Cloud Infrastructures. In *Learning-Driven Data Fabrics for Sustainability: Cloud-to-Thing Continuum Solutions for Global Challenges* (pp. 11-29). Cham: Springer Nature Switzerland. DOIhttps://doi.org/10.1007/978-3-032-09090-4_2

[7] Bardiani, J., Ragani, R. F., Manes, A., Sbarufatti, C., & Kefal, A. (2026). Real-time damage detection and localization in ship structures using iFEM and machine learning techniques. *Ocean Engineering*, *343*, 123379. https://doi.org/10.1016/j.oceaneng.2025.123379

[8] Samala, T., & Reddy, K. U. (2026). Machine-Learning Based Predictive Maintenance for Flexible Systems: A Cyber-Physical Approach to Minimize Throughput time. *Journal of The Institution of Engineers (India): Series C*, 1-15. DOIhttps://doi.org/10.1007/s40032-025-01308-3

[9] Mallick, C., Nayak, S., Singh, K. N., & Senapati, M. R. (2026). Securing the Internet of Things Through Intrusion Detection System Utilizing Machine Ensemble Learning and Feature Extraction Techniques. *SN Computer Science*, *7*(1), 66. DOIhttps://doi.org/10.1007/s42979-025-04648-0

[10] Yi, N., Xu, J., Yan, L., & Huang, L. (2020). Task optimization and scheduling of distributed cyber–physical system based on improved ant colony algorithm. *Future Generation Computer Systems*, *109*, 134-148. https://doi.org/10.1016/j.future.2020.03.051

[11] Abbasi, M., Yaghoobikia, M., Rafiee, M., Jolfaei, A., & Khosravi, M. R. (2020). Efficient resource management and workload allocation in fog–cloud computing paradigm in IoT using learning classifier systems. *Computer communications*, *153*, 217-228. https://doi.org/10.1016/j.comcom.2020.02.017

[12] Himeur, Y., Alsalemi, A., Bensaali, F., & Amira, A. (2021, August). The emergence of hybrid edge-cloud computing for energy efficiency in buildings. In *Proceedings of SAI Intelligent Systems Conference* (pp. 70-83). Cham: Springer International Publishing. DOIhttps://doi.org/10.1007/978-3-030-82196-8_6

[13] Wang, J. L., Tong, S., Li, X. Y., Yang, Z., Xiao, F., & Liu, Y. H. (2025). On the Scalability of Internet of Things Systems. *Journal of Computer Science and Technology*, *40*(5), 1182-1194. DOIhttps://doi.org/10.1007/s11390-025-5178-5

[14] Fang, L., Shi, J., Wu, L., Tan, J., & Wan, J. (2026). Perspectives and prospects on embodied intelligence-empowered smart manufacturing. *Journal of Intelligent Manufacturing*, 1-20. DOIhttps://doi.org/10.1007/s10845-025-02763-6

[15] Lakshmi, D. V., Srinivasan, T. R., Bhasker, B., & Thillaiarasu, N. (2026). A Cognitive Workload-Aware Machine Learning Model for Performance Enhancement in Cyber-Physical Systems. In *Reliability in Cyber-Physical Systems: The Human Factor Perspective* (pp. 31-50). Cham: Springer Nature Switzerland. DOIhttps://doi.org/10.1007/978-3-032-09917-4_2

[16] https://www.kaggle.com/datasets/programmer3/ton-iot-network-intrusion-dataset?utm_source Seen on Aug 2026

[17] https://research.unsw.edu.au/projects/toniot-datasets?utm_source