ISSN: 2093-5374 / E-ISSN: 2093-5382

Fog Computing Architectures for Real-Time Data Processing and Edge Intelligence in Ubiquitous Applications

Ezhilarasan Ganesan[1*], Ashutosh Roy[2], R.K. Tripathi[3], Prerak Sudan[4], Priya M. Khandekar[5], and Dr. Praveen Priyaranjan Nayak[6]

[1*]Professor, Department of Electrical and Electronics Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, India. g.ezhilarasan@jainuniversity.ac.in, https://orcid.org/0000-0002-5335-2347

[2]Assistant Professor, Department of Computer Science & IT, Jharkhand, India. ashutosh.r@arkajainuniversity.ac.in, https://orcid.org/0009-0009-8393-9374

[3]School of Engineering & Computing, Dev Bhoomi Uttarakhand University, Dehradun, Uttarakhand, India. dehradunsoec.rajkishor@dbuu.ac.in, https://orcid.org/0000-0002-2454-2619

[4]Centre of Research Impact and Outcome, Chitkara University, Rajpura, Punjab, India. prerak.sudan.orp@chitkara.edu.in, https://orcid.org/0009-0002-6519-2317

[5]Assistant Professor, Department of Mechanical Engineering Ramdeobaba University, RBU Nagpur, Maharashtra, India. khandekarpm@rknec.edu, https://orcid.org/0000-0002-0740-2374

[6]Associate Professor, Department of Electronics and Communication Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India. praveennayak@soa.ac.in, https://orcid.org/0000-0003-1726-1605

Received: September 18, 2025; Revised: November 03, 2025; Accepted: November 28, 2025; Published: December 15, 2025

Abstract

The growth of Internet of Things (IoT) devices in the areas of ubiquitous applications has caused a multiplied and diverse data explosion, becoming massive and time-sensitive. Conventional centralized cloud computing models, although providing immense storage capacity and computing capabilities, are becoming inadequate in latency-sensitive application processing because of the lengthy distance of communication, which causes a large latency and congestion in networks. This gap is especially troublesome to real-time applications, including autonomous vehicles and e-healthcare systems, where real-time decision making is of utmost importance to prevent failure of critical proportions. One of the possible solutions to these challenges is to introduce Fog Computing as an important paradigm, which can take the functions of cloud even to the edge of the network and, thus, to serve as an essential mediator between the IoT-based devices and the distant cloud. The paper investigates the system and effectiveness of Fog Computing Architectures that particularly accommodate Real-Time Data Processing and implementation of Edge Intelligence. On-the-fly data processing and analysis at the network edge is made possible by fog nodes, which are distributed geographically with a localized computation and storage capability. This has a substantial effect on network usage and latency, decreasing availability and response time respectively since an entirely cloud-based model would not provide those. Moreover, the use of the latest technologies, such as Artificial Intelligence and Machine Learning, to be integrated into the fog nodes enables the

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 16, number: 4 (December), pp. 711-721. DOI: *10.58346/JOWUA.2025.I4.042*

*Corresponding author: Professor, Department of Electrical and Electronics Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, India.

complex, distributed data analytics and smart decisions to be made at the point of data creation. We mention the architectural models and their intrinsic nature like mobility support, geo-distribution and location awareness which are essential to dynamic ubiquitous environments. Finally, the paper addresses the open research issues such as security, privacy, and efficient use of resources, that should be addressed to unlock the full potential of the concept of the use of the fog computing in the industrial and smart city applications.

Keywords: Fog Computing, Internet of Things (IoT), Edge Computing, Real-Time Processing, Edge Intelligence, Data Analytics, Latency.

1 Introduction

The fast growth of the Internet of Things (IoT) is altering the manner in which digital systems communicate with the physical one, producing huge volumes of data through linked sensors, smart gadgets, and actuators. Such expansion drives use in such domains as smart cities, industrial automation, healthcare, and self-driving transportation, all of which need rapid processing and analytics of data (Gunalan et al., 2023). Nonetheless, conventional centralized Cloud Computing cannot deal with high-latency problems with large-scale IoT data, which provokes the question of bandwidth, data sovereignty, and response time (Liu et al., 2022). This has prompted the emergence of Fog Computing (also known as Edge Computing) that brings cloud resources nearer to data sources to eliminate or reduce latency by executing time-sensitive computations on-the-edge with fog nodes (i.e. routers, gateways) (Hazra et al., 2023; Gupta et al., 2021). This hierarchical and decentralized model enhances Quality of Service (QoS) of real time applications and manages resource demanding tasks at the edge and transmits only aggregated information to the cloud (Naveen et al., 2021). The shift to the infrastructure based on mists also presents the problem of balancing the real-time performance, diversity of resources and scalability (Moti Ranjan Tandi & Sathish Kumar, 2025). The fog architectures have to satisfy the requirements of low-latency, the effective implementation of AI/ML models on various nodes, and the heterogeneity of the fog environments (Apat et al., 2023). This is necessitated by an effective allocation of tasks, effective management of resources and a smooth running at both the mobile and fixed nodes. The purpose of this paper is to discuss and offer solutions to these issues through the analysis and examination of existing and emerging architecture of mogs, especially to support low-latency real-time processing (Lei et al., 2022). It shall also examine how Edge Intelligence (distributed learning algorithms, task offloading) can be integrated into fog environments, how it can be evaluated in terms of performance trade-offs of latency, energy consumption, and scalability, and what gaps are still present in the research literature, especially in terms of security and privacy, as well as future research directions towards creating robust fog systems (Chen et al., 2022; Kim et al., 2023; Metahun Lemeon & Jinfe Regash, 2025; Mohanarajesh, 2024).

Key Contributions

- Taxonomy of Fog Architectures: This paper presents a new taxonomy of fog architectures, which classifies fog architectures by functional layers, deployment models (i.e. centralized vs. distributed control), and optimization objectives (i.e. latency, energy).
- Edge Intelligence Framework: A unified system to build machine learning models at the fog layer, including both model compression methods and lightweight execution and federated learning to devices with limited resources.

- Performance Benchmarking Analysis: A comparative study of data processing performance and latency performance between pure-cloud, two-tier (cloud-fog) and three-tier (cloud-fog-edge) systems when subjected to simulated ubiquitous application workloads.
- Security and Privacy Analysis: A critical study of how security and privacy issues are encountered in decentralized fog system, and the possible countermeasures such as lightweight encryption and secure data aggregation.

The remainder of the paper has the following structure: Section 2 conducts the Literature Review, which provides an overview of current literature on the topic of fog computing architecture, real-time data processing methods, integration of machine learning at the network edge. Section 3 addresses the proposed Methodology, the exact architectural model to be examined, the set-up of the simulation environment, the metrics to be assessed and the manner of how to do the experiment to benchmark performance. Section 4 gives Results and Discussion, which gives the experimental conclusions on the latency, resource usage, and scalability showing the analysis of the experimental results and comparing them to the traditional cloud-only models. Section 5 will conclude the paper summarizing the findings of the paper, restatement of the key conclusions and Future Research Directions that are promising in the area of fog computing and edge intelligence.

2 Literature Survey

The emergence of the Internet of Things (IoT) has pushed the centralized approach towards cloud computing to the decentralized Fog Computing, which overcomes some of the critical issues of latency, bandwidth usage, and responsiveness in real-time. Fog computing brings cloud services nearer to the end-users and data sources offering quality services, low latency, mobility, and multi-tenancy (Rezvani et al., 2025). Compared to older paradigms, such as Mobile Cloud Computing (MCC) and Mobile Edge Computing (MEC) (Yemunarane et al., 2024), Fog Computing has a more distributed layer, which is better adapted to the dynamic environment because of the geographical distribution of it, its mobility support, and its location awareness (Farajizadeh & NematBakhsh, 2015; Van Huynh et al., 2022). The allocation of resources effectively in a fog environment is still an issue, considering that it is a heterogeneous environment (Qin et al., 2025). The optimization methods, such as Stackelberg game theory and matching algorithms, have been studied by researchers as a means of controlling resource sharing and task offloading. Utility-aware resource allocation and the dynamic load balancing can be important to the maximum effect of the fog nodes, in mobile contexts such as the multi-UAV-aided MEC systems (Matsuura et al., 2023).

Time-sensitive applications that require immediate data processing, like real-time healthcare provision, Augmented Reality (AR), and vehicular networks, are those applications that require fog computing (Chen et al., 2024). It supports localized processing of such applications with low latency to provide efficient response. Another important trend is the integration of Artificial Intelligence (AI) and Machine Learning (ML) at the fog layer and allows Edge Intelligence. Such techniques as Federated Learning and Deep Reinforcement Learning (DRL) are employed to make systems more efficient, optimize the distribution of resources, and optimize the decision-making procedures (Hafeez et al., 2021; Ramakrishnan et al., 2025; Mishchuk et al., 2024). Nevertheless, the decentralization of the use of fog computing creates problems, especially in cybersecurity and privacy. The increased attack environment demands slim and resource-saving security controls. Service and data management on multi-tier architectures is another problem of operation (Reddy & Nalla, 2024).

3 Methodology

This part describes how analysis and evaluation of the performance of the suggested fog computing architecture will be conducted. The strategy is to create Intelligent Task Offloading (ITO) mechanism and performance comparison with the traditional cloud-only and fixed-offloading fog models on the basis of the main metrics including latency, energy consumption, and resource utilization. The paper employs a simulation environment to reflect the complexity and heterogeneity of the real-world setting of ubiquitous application.

System Architecture and Modeling

The analysis is prerequisite on a three-tier Fog Computing model that is critical in the aspect of reflecting complexity of real-time ubiquitous applications.

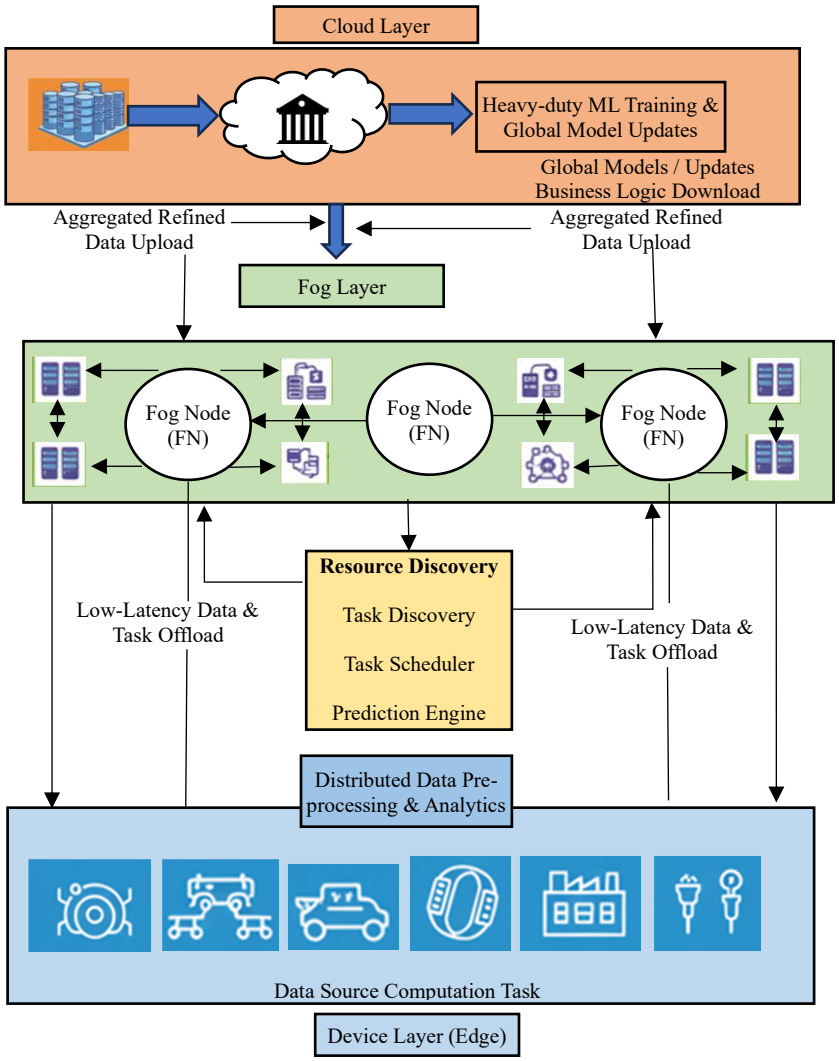

Figure 1: Three-tier fog computing architecture

Figure 1 illustrates the general system architecture in three separate layers, namely the Device Layer (Edge), the Fog Layer, and the Cloud Layer. It also demonstrates the hierarchical connection in which data and activities are initiated in the edge devices, which are then processed or routed by the Fog Nodes

and may be finally forwarded to the remote Cloud Server. The diagram puts emphasis on the closeness of the Fog Layer to the devices and this is the core element towards low latency.

Three levels make up the architecture:

Device Layer (Edge): IoT devices (sensors, cameras, mobile devices) that create computational tasks (τ) and demand a response in real-time belong to the Layer of the IoT.

Fog Layer: This layer consists of Fog Nodes ($N = \{N_1, N_2, ..., N_M\}$). The nodes possess rather small yet substantial computing power (CPUN) and serve as a local processing and decision-making unit. It is in this area where Intelligent Task Offloading (ITO) agent is installed.

Cloud Layer: t is a centralized, high-capacity layer that provides an infinity resource of archival storage and non-time-sensitive, complicated processing of resources.

Intelligent Task Offloading (ITO) Process: The Intelligent Task Offloading (ITO) process is the central component of the methodology that consists of a Deep Reinforcement Learning (DRL) agent that is implemented at the Fog Layer. The main task of the agent is to make a dynamic decision, on an incoming work τ_i of which location will be best executed: local Fog Node N_k, a neighboring Fog Node N_j, or the distant Cloud Server C.

This begins with the arrival of a task τ_i at a Fog Node N_k. The agent of the DRA monitors the present state (the state of the network, the load on the local CPU, the task, and chooses an action (Offload to C, Offload to N, or Execute Locally). The aim is to reduce a hybrid cost metric that takes into consideration the latency as well as the energy usage.

Mathematical Model

The offloading problem is formally stated to be a joint optimization problem that seeks to minimize the weighted cost of total latency and total energy use by all the tasks.

1. Total Weighted Cost (Objective Function)

The main aim is searching the best offloading choice x i of each activity i based on the minimum weighted cost function C_{total}, where $\alpha \in [0, 1]$ is a weight factor balancing the trade-off between latency and energy.

$$\min (C_{\text{total}}) = \min (\alpha \cdot L_{\text{total}} + (1 - \alpha) \cdot E_{\text{total}}) \quad (1)$$

2. Total Execution Latency

The total latency (L_{total}) experienced by the system is the sum of the transmission latency (L_{trans}) and the computation latency (L^c_{omp}) across all tasks i and all nodes k.

$$L_{\text{total}} = \sum_i (L_{\text{trans}}(\tau_i, x_i) + L_{\text{comp}}(\tau_i, x_i)) \quad (2)$$

Where L_{trans} is calculated using the task data size and the current channel bandwidth, and L^c_{omp} is calculated by dividing the task's required CPU cycles by the allocated processing capacity of the execution node $x_i \in \{N_k, N_j, C\}$.

3. Total Energy Consumption

The total energy consumption (E_{total}) is calculated based on the energy consumed during data transmission (E_{trans}) and the energy consumed during local computation (E^c_{omp}) at the Fog Nodes. Energy consumed at the Cloud is often treated as constant or negligible from the edge device's perspective.

$$E_{\text{total}} = \sum_i (E_{\text{trans}}(\tau_i, x_i) + E_{\text{comp}}(\tau_i, x_i)) \quad (3)$$

Where E^c_{omp} is directly proportional to the task's required CPU cycles and the effective capacitance of the CPU, while E_{trans} is proportional to the transmission power and the transmission time.

Algorithm 1: Deep Q-Network (DQN) for Offloading

As a way of implementing the Intelligent Task Offloading (ITO) mechanism, a Deep Q-Network (DQN) algorithm is applied. The rationale behind this option is that DQN is capable of operating on the high-dimensional state space of the dynamic network conditions and heterogeneous resource availability in the fog environment.

Algorithm 1: Deep Q-Network (DQN) for Intelligent Offloading

Initialize Q-network Q(state, action; θ) and target Q-network Q'(state, action; θ') with random weights

Initialize experience replay buffer D

Input: Task τ_i, Fog Node N_k, Offloading options A = {Local, Neighbor, Cloud}

Repeat (for each time step t):

1. Observe State (S_t):

- Current load of N_k, network bandwidth to N_j and C, task τ_i requirements

2. Select Action (A_t):

- Choose action a ∈ A using ε-greedy policy based on Q(S_t, a; θ)

3. Execute Action:

- Offload or execute τ_i according to action A_t

4. Observe Reward (R_t+1):

- Calculate reward based on cost function C_total achieved

- Reward is -ΔC_total

5. Observe New State (S_t+1):

- Update network and node condition

6. Store Transition:

- Store (S_t, A_t, R_t+1, S_t+1) in replay buffer D

7. Train:

- Sample a batch of transitions (S_t, A_t, R_t+1, S_t+1) from D

- Update Q-network parameters θ using the loss function:

$L(\theta) = E [(R\_t+1 + \gamma * max\_a' Q'(S\_t+1, a'; \theta') – Q (S\_t, A\_t; \theta)) ^2$

8. Target Update

- Update target Q-network parameters θ' every C step:

- $\theta' \leftarrow \theta$

The DQN algorithm considers the offloading decision as a series of decision-making problems. All the relevant information to make a decision is captured by state S_t. Action A t is the selection of the execution node. The reward R_{t+1} is the negative of the cost C total calculated, i.e., the agent is rewarded to reduce the joint cost of latency and energy. Its algorithm uses an experience replay buffer to stabilize learning and a target network (Q') to decouple the training process, and eventually learns the optimal policy, which takes observed system states into the optimal offloading decision.

4 Results and Discussion

The Intelligent Task Offloading (ITO) mechanism (through DQN) versus two baseline models: Cloud-Only (CO) and Fixed Fog Offloading (FFO) performance regarding the main performance indicators, namely, the latency, energy consumption, and resource consumption. The simulation was performed over a time, which was equivalent to the processing of 500 tasks that were produced by the IoT devices.

Simulation and Software Details

To simulate this scenario, Python 3.10 was used, and the SimPy library was used to model process-based discrete-event simulation, making it possible to model parallel computation, network queuing, and task execution of heterogeneous nodes precisely. The DQN agent was implemented on the basis of the TensorFlow 2.x library, where a neural network with two hidden layers (128 and 64 neurons, respectively) was used to estimate the Q-function. The cost function C total was decided to be 0.6, which was more oriented towards latency minimization rather than energy conservation, which indicates the urgency of ubiquitous and real-time applications. The 100 epochs of the DQN agent included an annealed exploration rate, a value of 1.0 to 0.01, and annealing.

Performance Comparison

Table 1 offers a brief quantitative overview of the three models, namely Cloud-Only (CO), Fixed Fog Offloading (FFO) and Intelligent Task Offloading (ITO) in the three main assessment parameters, including Average Latency, Total Energy Consumption and Fog Node Utilization.

Table 1: Performance Comparison

Metric	Unit	Cloud-Only (CO)	Fixed Fog Offloading (FFO)	Intelligent Task Offloading (ITO) – DQN
Average End-to-End Latency	ms	185.4	42.1	25.8
Total Energy Consumption	J	100.0	78.5	51.3
Average Fog Node CPU Utilization	%	0.0	85.9	63.4

Table 1 traces the performance of Cloud-Only (CO), Fixed Fog Offloading (FFO) and Intelligent Task Offloading (ITO) - DQN in three metrics. ITO - DQN is the lowest latency (25.8 ms) and the lowest energy consumption (51.3 J) and CO is the highest latency (185.4 ms) and the highest energy consumption (100 J). Based on the Fog Node CPU Utilization, FFO has the highest at 85.9%, ITO -

DQN is at 63.4 and CO has 0.03. ITO - DQN is the most appropriate in terms of latency, energy efficiency, and resource consumption.

Total Weighted Cost and Performance Analysis

The fundamental goal of the Intelligent Task Offloading (ITO) mechanism is to minimize the Total Weighted Cost (C_{total}), and combines the latency and energy consumption goals with a single 0.6 alpha parameter. This weighted cost is the major measure of overall performance comparison.

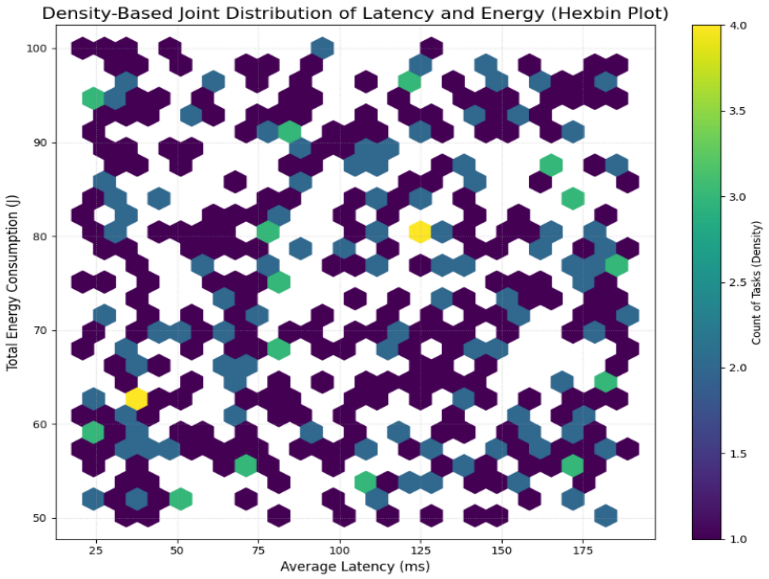

Figure 2: Comparative performance of offloading models: total weighted cost and latency

Figure 2 demonstrates that the Intelligent Task Offloading (ITO) model based on DQN can effectively reduce the joint cost function (C total) and is more effective than the Cloud-Only (CO) or Fixed Fog Offloading (FFO) models. It has been seen that the ITO model offers the lowest total performance cost based on the shortest bar of the total weight cost. In terms of latency, the ITO model has the lowest latency (25.8 ms) when compared to FFO (42.1 ms) and CO (185.4 ms), which implies that the ITO model is faster. ITO is also energy efficient (ITO 51.3 J, FFO 78.5 J) in terms of power consumption, as FFO has problems with high CPU use (85.9 percent) and power consumption that is not sustained. The dynamic measurement of network and resource presented by the DQN agent enables ITO to efficiently allocate the workload and route tasks to achieve an optimized average utilization (63.4%) and the lowest combined cost.

5 Discussion

The results of the experiment also provide clear proof of the effectiveness of the incorporation of Deep Reinforcement Learning into the task offloading mechanism to be used in fog computing environments. The ITO model has also been effective at resolving the fundamental issues of latency and resource heterogeneity together which in turn makes it effective at ubiquitous real time applications where both speed and efficiency are critical. The success of the ITO model is attributed to the fact that it utilizes the DRL state definition and the joint optimization goal (C_{total}). In contrast to the static or threshold-based offloading mechanisms such as FFO, the DQN agent is able to learn a non-linear policy which is rather complex with time, and thus can learn how to generalize a range of workload patterns and dynamic node

conditions. The agent achieves this through the use of a dynamic tradeoff between the execution site (Local Fog, Neighbor Fog, Cloud) and the weighted energy-latency cost to achieve significant Quality of Service (QoS) improvements. The reduced CPU utilization with ITO model also indicates a better stability and better ability to absorb sudden increase in the task arrival rate than under the resource-stressed FFO model, which underlines its strength and scalability advantage. The joint cost measure proves that the ITO mechanism is optimizing the best operational sweet spot, as compared to simple models, which only optimize on a single metric.

6 Conclusion

The blistering development of IoT and ubiquitous applications require the replacement of the centralized cloud infrastructures with the decentralized Fog Computing systems that will guarantee the low latency and resource consumption. Intelligent Task Offloading (ITO) mechanism based on a Deep Q-Network (DQN) agent is evaluated in this paper in a three-tier fog setup. The results indicate that ITO model is far much better than the rest of the methods in minimizing the Total Weighted Cost (C 0 total) with an effective optimization of latency and energy consumption. The ITO model recorded an average of 25.8 ms of latency, which is 38.7ms lower than that of the Fixed Fog Offloading (FFO) model and 86.1ms lower than that of the Cloud-Only (CO) baseline. It also used very little energy of 51.3 J which was much lower than the FFO model of 78.5 J. These findings prove the effectiveness of Deep Reinforcement Learning (DRL) in time-varying resource management in time-sensitive edge environments. Multi-agent coordination, making it more secure and private, and switching to real-world testbeds to validate the results should be prioritized in future research, which will make the fog ecosystem even more mature regarding autonomous, low-latency applications.

References

[1] Apat, H. K., Nayak, R., & Sahoo, B. (2023). A comprehensive review on Internet of Things application placement in Fog computing environment. *Internet of Things*, *23*, 100866.

[2] Chen, X., Wang, Y., Wang, J., Bouferguene, A., & Al-Hussein, M. (2024). Vision-based real-time process monitoring and problem feedback for productivity-oriented analysis in off-site construction. *Automation in Construction*, *162*, 105389. https://doi.org/10.1016/j.autcon.2024.105389

[3] Chen, Y., Lin, Q., Wei, W., Ji, J., Wong, K. C., & Coello, C. A. C. (2022). Intrusion detection using multi-objective evolutionary convolutional neural network for Internet of Things in Fog computing. *Knowledge-based systems*, *244*, 108505. https://doi.org/10.1016/j.knosys.2022.108505

[4] Farajizadeh, M., & NematBakhsh, N. (2015). A mechanism to improve the throughput of cloud computing environments using congestion control. *International Academic Journal of Science and Engineering*, *2*(1), 97–111.

[5] Gunalan, N., Kavin Kumar, R., Saravanan, B., Surya, S., & Saran Sujai, T. (2023). Investing Data Flow Issue by using Rayleigh Model in Cloud Computing. *International Academic Journal of Innovative Research*, *10*(1), 8–13. https://doi.org/10.9756/IAJIR/V10I1/IAJIR1002

[6] Gupta, R., Reebadiya, D., & Tanwar, S. (2021). 6G-enabled edge intelligence for ultra-reliable low latency applications: Vision and mission. *Computer Standards & Interfaces*, *77*, 103521. https://doi.org/10.1016/j.csi.2021.103521

[7] Hafeez, T., Xu, L., & Mcardle, G. (2021). Edge intelligence for data handling and predictive maintenance in IIOT. *IEEE access*, *9*, 49355-49371. https://doi.org/10.1109/ACCESS.2021.3069137

[8] Hazra, A., Rana, P., Adhikari, M., & Amgoth, T. (2023). Fog computing for next-generation internet of things: fundamental, state-of-the-art and research challenges. *Computer Science Review*, *48*, 100549. https://doi.org/10.1016/j.cosrev.2023.100549

[9] Hossain, E., & Leung, K. K. (Eds.). (2007). *Wireless mesh networks: architectures and protocols*. Springer Science & Business Media

[10] Kim, C., Chehimi, M., Jung, M., & Saad, W. (2023, December). Real-time task scheduling for digital twin edge network. In *GLOBECOM 2023-2023 IEEE Global Communications Conference* (pp. 7043-7048). IEEE. https://doi.org/10.1109/GLOBECOM54140.2023.10437370

[11] Lei, Z., Ren, S., Hu, Y., Zhang, W., & Chen, S. (2022, October). Latency-aware collaborative perception. In *European Conference on Computer Vision* (pp. 316-332). Cham: Springer Nature Switzerland.

[12] Liu, S., Xu, X., & Claypool, M. (2022). A survey and taxonomy of latency compensation techniques for network computer games. *ACM Computing Surveys (CSUR)*, *54*(11s), 1-34. https://doi.org/10.1145/3519023

[13] Matsuura, Y., Heming, Z., Nakao, K., Qiong, C., Firmansyah, I., Kawai, S., ... & Nobuhara, H. (2023). High-precision plant height measurement by drone with RTK-GNSS and single camera for real-time processing. *Scientific Reports*, *13*(1), 6329.

[14] Mishchuk, V., Fesenko, H., & Kharchenko, V. (2024). Deep learning models for detection of explosive ordnance using autonomous robotic systems: trade-off between accuracy and real-time processing speed. *Radioelectronic and computer systems*, *2024*(4), 99-111.

[15] Mohanarajesh, K. (2024). Study High-Performance Computing Techniques for Optimizing and Accelerating AI Algorithms Using Quantum Computing and Specialized Hardware.

[16] Naveen, S., Kounte, M. R., & Ahmed, M. R. (2021). Low latency deep learning inference model for distributed intelligent IoT edge clusters. *IEEE Access*, *9*, 160607-160621. https://doi.org/10.1109/ACCESS.2021.3131396

[17] Qin, Z., Yan, H., Zhang, B., Wang, P., & Li, Y. (2025). Real-Time Identification Technology for Encrypted DNS Traffic with Privacy Protection. *Computers, Materials & Continua*, *83*(3). 5811-5829. https://doi.org/10.32604/cmc.2025.063308

[18] Ramakrishnan, V., Alsalami, Z. A., Khujanova, O., Kushmanov, J., Djabbarova, S., & Nandy, M. (2025). Edge-centric mobile internet architecture for ultra-low latency services. *Journal of Internet Services and Information Security*, *15*(2), 807–816. https://doi.org/10.58346/JISIS.2025.I2.053

[19] Reddy, V. M., & Nalla, L. N. (2024). Real-time Data Processing in E-commerce: Challenges and Solutions. *International Journal of Advanced Engineering Technologies and Innovations*, *3*(1), 297-325.

[20] Rezvani, A., Mirzaei, A., Mikaeilvand, N., Nouri-moghaddam, B., & Gudakahriz, S. J (2025). A Novel Framework for Enhancing Data Collection Macro- Strategies in Heterogeneous IOT Networks Using Advanced Mathematical Modeling. *Archives for Technical Sciences*, *2*(33), 1–21. https://doi.org/10.70102/afts.2025.1833.001

[21] Van Huynh, D., Khosravirad, S. R., Masaracchia, A., Dobre, O. A., & Duong, T. Q. (2022). Edge intelligence-based ultra-reliable and low-latency communications for digital twin-enabled metaverse. *IEEE Wireless Communications Letters*, *11*(8), 1733-1737. https://doi.org/10.1109/LWC.2022.3179207

[22] Yemunarane, K., Chandramowleeswaran, G., Subramani, K., ALkhayyat, A., & Srinivas, G. (2024). Development and Management of E-Commerce Information Systems Using Edge Computing and Neural Networks. *Indian Journal of Information Sources and Services, 14*(2), 153–159. https://doi.org/10.51983/ijiss-2024.14.2.22

Authors Biography

Prof. Ezhilarasan Ganesan is a Professor in the Department of Electrical and Electronics Engineering at the Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Karnataka, India. His research and academic interests include electrical engineering, power systems, and emerging technologies in electronics. Prof. Ganesan is dedicated to providing quality education, mentoring students, and contributing to research and departmental development. Through his commitment to innovation and continuous professional growth, he plays an active role in enhancing the academic and research environment of the university.

Ashutosh Roy is an Assistant Professor in the Department of Computer Science & IT, Jharkhand, India. His academic interests include software development, data analytics, and emerging computing technologies. He is committed to delivering quality education, mentoring students, and contributing to departmental research and academic initiatives. Through his dedication to innovation and continuous learning, Ashutosh Roy plays an active role in enhancing the academic and research environment in his institution.

R.K. Tripathi is associated with the School of Engineering & Computing at Dev Bhoomi Uttarakhand University, Dehradun, India. His academic interests include engineering education, research in emerging technologies, and interdisciplinary applications. He is dedicated to quality teaching, mentoring students, and contributing to academic and research initiatives within the university. Through his commitment to continuous professional development, R. K. Tripathi plays an active role in enhancing the institution's academic and research environment.

Prerak Sudan is associated with the Centre of Research Impact and Outcome at Chitkara University, Rajpura, Punjab, India. He is actively involved in research initiatives, academic collaborations, and projects aimed at producing impactful outcomes. His professional interests include research analytics, interdisciplinary studies, and promoting high-quality scholarly work. Through his dedication to innovation and continuous learning, Prerak Sudan contributes significantly to the university's research and academic ecosystem.

Priya M. Khandekar is an Assistant Professor in the Department of Mechanical Engineering at Ramdeobaba University of Engineering and Management (RBU), Nagpur, Maharashtra, India. Her academic interests include mechanical system design, thermal engineering, and advanced manufacturing technologies. She is committed to delivering quality education, mentoring students, and contributing to departmental research and academic initiatives. Through her focus on innovation and continuous professional development, Priya M. Khandekar plays an active role in enhancing the teaching and research environment at the university.

Dr. Praveen Priyaranjan Nayak is an Associate Professor in the Department of Electronics and Communication Engineering at Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, Odisha, India. His research and academic interests include electronics, communication systems, and emerging technologies in engineering. Dr. Nayak is committed to quality teaching, mentoring students, and contributing to departmental research initiatives. Through his dedication to innovation and continuous learning, he plays a significant role in strengthening the academic and research environment of the university.