POLARIS GLOBAL JOURNAL OF SCHOLARLY RESEARCH AND TRENDS

Research Article

The Evolution of Fog and Cloud Computing in Distributed Systems: A Review of Architectures, Challenges, and Parallel Processing Techniques

Hawar Bahzad Ahmad [1,3] , Subhi R. M. Zeebaree [2]

[1]*Department of Computer Science, Nawroz University, Duhok, Kurdistan Region – Iraq*

[2]*Engineering Department, Technical College of Engineering, Duhok Polytechnic University, Duhok, Iraq.*

[3]*Information Technology Department, Technical College of Informatics-Akre, Akre University for Applied Sciences, Duhok, Iraq*

*Corresponding author E-mail: hawar.doski@nawroz.edu.krd

ARTICLEINFO

Received
4/05/2025
Revised
2/06/2025
Accepted
18/06/2025

KEYWORDS
Fog Computing, Cloud Computing, Distributed Systems, Hybrid Architectures, Parallel Processing Techniques.

ABSTRACT

Fog and cloud computing has revolutionized distributed systems through solving significant challenges like resource management, latency, and scale. The main characteristic of fog computing is to bring the computational resources close to the data source to allow near real-time processing for delay sensitive applications, improving the response time for those applications, while cloud computing centralizes the data for long life storage and massive processing. In this manner, these paradigms interoperate to yield hybrid architectures that address the increasing demands of networked systems such as the Internet of Things. In this review we listed the development, models, issues and parallel processing in fog and cloud computing. Energy-efficient task scheduling, privacy-preserving models, and fault-tolerant designs are some advancements that improve system reliability and performance. Additionally, containerized microservices and federated learning approaches also enable seamless integration and secure data management in various applications. However, there are still issues with strong interoperability, preserving the performance of a system under extreme load, and reducing security threats even with great advances. We analyze the gaps, propose solutions, and emphasize the key role of adaptive frameworks and innovative resource allocation methods in tackling those gaps. These results show how fog and cloud computing can change the landscape of distributed systems in the future.

E-ISSN: 2961-3809

How to cite:
Ahmad H. B., Zeebaree S. R. (2025). Clarifying the Scope of NLP: Language Processing vs. Neuro-Linguistic Programming. *Polaris Global Journal of Scholarly Research and Trends*, 4(1), 1-22. https://doi.org/10.22219/pgjsrt.v4n1a218

INTRODUCTION

Distributed computing systems can be seen as an ecosystem of various technologies that have revolutionized the processing, storage, and use of data in various applications. Wikipedia wrote — Fog computing is a system-level architecture that distributes resources and services of computing among the various nodes present in the network. Third, among the most notable emerging paradigms, fog and cloud computing appear to be complementary paradigms that extend the service models to meet the growing demands for computing efficiency, scalability, and responsiveness (open-in-new window). The significant size of data and complexity of computation create a paradigm of today's cloud computing which can provide a centralized infrastructure for huge data to be stored and run on high-end configurations but the paradigm of fog computing moves toward bringing the computational center physically close to data source instead of backing the computation and storage to remote server with dialog on the cloud [1], [2], [3]. This hybrid integration enables large-scale analytics and real-time data processing, laying a solid foundation for modern distributed systems.

Farther away from the edge of the network, fog computing is critical for latency-sensitive applications such as autonomous systems, healthcare, and industrial IoT. Local processing of data lowers the communication delay and allows for real-time decisions [4], [5], [6].

Cloud computing, with the ability to provide centralized handling of resources and global integration [7], [8], [9], acts as an adjunct for processing and storing these large sets of data.

Although they have their paths, there are still challenges when integrating the fog, and the cloud computing. Workflow scheduling in heterogeneous environments, energy efficiency, and security vulnerabilities are still major issues. To clear up the drawbacks forcing these issues, advanced resource management algorithms have been suggested, including AI driven algorithm and heuristic solutions [10], [11], [12]. Similarly, in distributed infrastructures, the improvement of confidence and widespread adoption can only be achieved by ensuring security and privacy of data [4], [13], [14].

So the evolution of fog and cloud computing is extensively investigated in this paper including the architectures, challenges, and distributed processing techniques. It indicates potential pathways for future research, highlights research gaps and illustrates synergies with respect to recent scholarly contributions.

Background Theory

The architecture of distributed systems has been completely redesigned by the combination of cloud and fog computing, allowing for more sophisticated resource management, data processing, and storage. To place these paradigms' development and function in contemporary systems in context, it is crucial to comprehend their underlying ideas.

A. Cloud Computing

Cloud computing is an innovative approach that consolidates the processing and storage of data and gives scalable resources on demand. It enables diverse applications, such as web hosting and advanced analytics, possible through a common architecture typically comprising three service models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [10], [13]. Deployment option provides different level of control and scalability that meet organizational goals such public, private, hybrid and community clouds [2], [6]. However, for real-time applications, its centralized architecture could suffer from latency and bandwidth problems [5], [7].

B. Fog Computing

Fog computing extends cloud computing by bringing the computation closer to where it is needed or where it is going to take place, and therefore, reducing latency and bandwidth usage. It helps to overcome challenges in real time applications including industrial automation and IoT-supported healthcare [4], [15]. Fog computing works in decentralized mode and utilizes edge devices to preprocess and filter the data before transferring these data in the cloud for further analysis [5], [8], [16].

The fog computing architecture is at least composed of the following levels: The level of the edge device, the level of the quasi-cloud, and the level of the cloud data center. This multi-level hierarchical topology allows data to be propagated along only those edges necessary towards its destination in the cloud, prioritizing on-time information processing within the rate-sensitive IoT context rather than at the resource-constrained edge nodes [7], [13]. However, fog computing has the challenges of resource allocation, energy efficiency, and security issues; thus, more advanced solutions are needed to leverage its advantages for efficient deployment [6], [11].

C. Integration and Synergy

Cloud and fog computing work together in a complementary nature. Fog computing covers tasks that depend on low latency or require localized processing where cloud computing excels at centralized storage and large-scale analytics. Their integrated collaboration creates a hybrid ecosystem which maximizes utilization of resources and improves performance of systems in different domains [6], [7], [17].

The combination is especially suited to IoT applications, where fog will face the real-time processing of sensor data and the cloud will take care of centralized analytics and long-term storage. As an example, fog nodes can, in the healthcare domain, process patient data locally to enable real-time monitoring of patients, while the cloud can store historical data, and later, be used for advanced analytics [4], [8]. Yet, there are impediments to integration including security vulnerabilities, data privacy issues, and advanced workflow scheduling algorithms that will need addressing [11], [18].

In recent years, the significance of deploying intelligent computing across both cloud and edge infrastructures has become increasingly evident. For instance, Salih et al. [19] developed a machine learning approach for early diabetes detection using the PIMA dataset, achieving noteworthy accuracy by applying principal component analysis alongside various classification algorithms. In another study, Zeebaree and Jacksi [20] explored the performance of shared memory systems within parallel computing frameworks, demonstrating measurable gains in CPU execution time under balanced workloads. Complementing this, Zebari and Yaseen [21] focused on distributed memory architectures for matrix-based computations, highlighting how effective client-server communication can improve processing efficiency. Additionally, in the realm of edge intelligence, Jghef et al. [22] proposed a biologically-inspired, trust-based architecture for the Internet of Drone Things (IoDT), addressing key challenges related to network congestion and security. Together, these contributions underscore a growing trend: the integration of parallel computing, adaptive learning techniques, and secure communication models as foundational components in the evolution of distributed systems.

Literature Review

Rocha Neto et al. study the amalgamation of fog computing and distributed machine learning for task scheduling in IoT [23]. By processing data near the source, the framework achieves low latency and better resource usage. The study addresses the gap between real-time analytics and resource-constrained environments by utilizing distributed learning. The insights drawn from the paper highlight the necessity of decentralized learning paradigms to efficiently manage extensive IoT data streams. Such capabilities of machine learning are useful to enhance the performance of fog

architectures, which otherwise suffer from associating cloud systems that have computational overheads.

Etemadi et al. The authors in [24] propose an autonomic resource provisioning architecture utilizing the MAPE-K model and Bayesian Learning. The optimization method aims to optimize a fog application with varying workloads in the above-mentioned scenarios. This leads to a consistent Quality of Service (QoS) and economical operation as it prevents situations of over-provisioning or under provisioning. In summary, this paper demonstrates the practical implementation of autonomic computing in fog systems and presents practical insights into adaptive resource allocation that helps improve performance in heterogeneous networks. In the context of dynamic fog environments, Bayesian learning and adaptation offers a solid framework for resource management challenges.

Zhang et al. [25] proposes a fault tolerant model for fog computing, in which the detection and recovery mechanisms of preemptive failure are summarised. By utilizing predictive analytics to detect potential failures with limited resource nodes, this model guarantees the reliability of the system. The authors also describe a solution framework for the incorporation of redundancy mechanisms in fog architectures, which, however, reduces the impact of node failures on critical applications. It is extremely important to enhance the reliability in mission-critical applications or data like emergency health care and industrial automation.

Azimi et al. [26] present the HiCH architecture, which is a hierarchical fog-assisted computing archiecture specifically for healthcare IoT applications. In this way, the framework aids in increasing the processing speed of data accumnulated, thus reducing the time required for quick medical treatment. This architecture designs the deployment of fog nodes, edge resources and cloud resources, making a trade-off between computation capability and time-constraint requirements. Our study demonstrates the strong potential of hierarchical designs to transform latency-sensitive applications (such as remote patient monitoring and telemedicine) that require bi-directional content feedback.

Saboor et al. [27] focus on enhancing the scalability of fog computing systems using containerized microservices. The proposed framework addresses the challenges of managing fluctuating workloads in distributed environments by dynamically orchestrating microservices. Your training is on data until October last year. The results demonstrate that the containerized designs are flexible and scalable, making them suitable for various Internet of Things applications, such as for smart cities and industrial IoT.

AlShathri et al. [28] consider dynamic task offloading for fog environments using parallel meta-heuristics, such as evolutionary algorithms and particle swarm optimization. The experimental results confirm the proposed mechanism significantly reduces the time to complete the hierarchical tasks, and minimizes energy usage by optimally assigning resources to the tasks. This study relates to one of fog worst enemies: sharing processing loads among heterogeneous nodes. The results underscore the value of scalable algorithms for efficiently managing dynamic workloads.

Alsadie et al. [29] proposed AI techniques-based frameworks to secure the fog environment. They classify AI techniques into resource management, privacy-preserving methods, and real-time threat detection. The study proposes lightweight AI models that integrate explainable AI principles to enhance system transparency and user trust. These results recognized the importance of AI to support security concerns such as intrusion detection and data privacy specific to decentralized fog topologies.

Arshed et al. create a genetic algorithm-based scheduler which [30] which improves at resource consumption and workload allocation of fog-cloud architectures. To ensure a balanced work distribution across nodes, the scheduler considers factors such as latency, cost, and energy

consumption. The study demonstrates how evolutionary algorithms can be applied to solve multi-objective optimization problems in distributed settings, including Internet of Things applications, where workloads are unpredictable and dynamic.

Dsouza et al. [31] fosters a policy-based security framework for fog computing that emphasizes secure collaboration and resource management among the participant nodes. It has introduced adaptive security policies according to different needs of heterogeneous devices and applications. The study offers a comprehensive methodology for securing decentralized systems by combining policy-oriented controls with resource distribution systems. This work is highly applicable to Internet of Things (IoT) ecosystems which need secure interactions among a large, heterogenous set of devices and nodes.

Rocha Neto et al. An adaptive resource management framework that addresses the energy efficiency problem in fog computing was designed by Chen et al. [32]. This proposed approach reduces the energy consumption while maintaining performance by adjusting the resource allocation in real-time according to the workload requirements. The above improvements contribute to the ongoing efforts to enhance the cost-effectiveness and sustainability of fog computing and fog nodes, particularly in resource constrained environments.

Alagheband et al. proposes a multi-tier container orchestration architecture focused on workload distribution and scalability [33] for IoT applications. The framework improves resource utilization and enables seamless integration across the various layers involved in the fog-cloud continuum by utilizing containerized microservices [130]. Containerized architectures have proven to be highly flexible solutions to accurately managing dynamic workloads and allowing real-time IoT applications, enabling cloud providers to address real-time processing of incoming IoT streams.

Dash et al. Federated Learning for Privacy-Preserved Analytics in Fog [34] It maintains user privacy by enabling collaborative learning between distributed nodes thus minimising the requirement for sending data to centralised servers. Relevant Fields of Study This research addresses crucial concerns with data security and compliance in distributed systems, particularly within regulated sectors such as healthcare and finance.

Revathi et al. [35] present a context-aware fog-assisted healthcare IoT monitoring system, improving the processing of vital signs in real time. The approach ensures quick medical response and reduced cloud resource load by restricting by allowing fog computing and edge analytics. This study demonstrates how context-aware systems can enhance the effectiveness and efficiency of Internet of Things based medical solutions.

Kashani et al. Dynamic task scheduling algorithms in fog environments with emphasis on resource utilizations and latency minimization; [37] Our proposed algorithms are dynamic and adapt to the workload fluctuations of the IoT network on the cloud. The results underline the adaptive scheduling importance for the maintained efficiency of these systems.

Alatoun et al. [37] proposed a scalable fog computing based framework for smart city applications to solve issue such as traffic congestion or energy optimization. This framework improves the efficiency of urban management systems through real-time analytics and decentralized processing. This research highlights how fog computing could revolutionize urban infrastructures.

Prajapat et al. [38] develop energy-efficient scheduling mechanisms for fog-cloud environments, to alleviate computational loads while reducing energy utilization. Considering the resource-constrained nodes in these fog environments, it sparks sustainable computing in fog environments.

Park et al. In addition, it presents novel lightweight cryptographic architectures to provide secure data sharing in fog computation in [39]. Many features provide enhanced data privacy and

compliance with strict security standards, thus solving one of the key problems in the decentralized architectures. These results demonstrate that one can implement secure data-sharing mechanisms with no performance breach.

H et al. [40] proposes an adaptive resource allocation model for IoT workloads in fog environments. By balancing resources with real-time requests, this model achieves optimization, leading to minimal operational costs and delays, thus, ensuring consistent Quality of Service (QoS) within the distributed systems.

Abbasi et al. [41] propose a fault tolerant design for mission critical fog application including redundancy mechanisms for maintaining continuous operation. Depth of the study: The study delves into various aspects of fog computing, focusing on the need for reliability and robustness in fog systems, especially in applications with potentially severe ramifications for downtime.

Discussion and Comparison

This section provides a comprehensive comparison of the studies discussed in the Literature Review. Key themes include system architecture and design, performance efficiency, security, privacy, reliability, and user experience. The analysis aims to highlight trends, gaps, and advancements in fog and cloud computing for distributed systems.

A. System Architecture and Design

Fog and cloud computing architectures aim to address scalability, geographic distribution, complexity, and integration challenges in distributed systems. Table 1 compares the architectural frameworks proposed in the reviewed studies.

Table 1: System Architecture and Design

Ref	Architecture & Design	Scalability	Data Management	Complexity	IoT-Cloud Integration
[23]	Distributed ML in Fog	High	Real-time IoT data	Moderate	Decentralized fog integration
[24]	MAPE-K with Bayesian Learning	High	Dynamic scaling	Low	Autonomic provisioning
[25]	Fault-Tolerant Fog Architecture	Moderate	Failure handling	High	Fault-resilient fog systems
[26]	HiCH Healthcare IoT Framework	High	Real-time healthcare data	High	Integrated IoT-Fog-Cloud
[27]	Microservices Framework	High	Task orchestration	Moderate	Scalable fog-cloud management
[28]	PSO Task Scheduling	Moderate	Task execution	Low	Dynamic fog-cloud scheduling
[29]	Federated Learning Models	Low	Privacy-preserving data	Moderate	Federated IoT-cloud
[30]	Genetic Algorithm Scheduling	Moderate	Resource allocation	Low	IoT-Fog optimization
[31]	Adaptive Resource Models	High	Dynamic allocation	Moderate	IoT-cloud load optimization
[32]	Multi-Tier Orchestration	High	Seamless integration	Moderate	Fog-cloud workload

					integration
[33]	Privacy-preserving Analytics	Moderate	Federated learning models	Moderate	Federated Fog analytics
[34]	Failure Prediction Systems	High	Proactive fault recovery	High	IoT Failure Resilience
[35]	Context-Aware Monitoring	High	Real-time healthcare	Moderate	IoT-based health monitoring
[36]	Dynamic Task Scheduling	Moderate	Optimized workloads	Low	Dynamic fog-cloud coordination
[37]	Smart City Optimization	High	Energy optimization	Moderate	Urban IoT integration
[38]	Energy Efficient Scheduling	High	Balanced loads	Moderate	IoT-cloud resource allocation
[39]	Lightweight Cryptographic Models	Low	Secure data handling	Low	Privacy-compliant sharing
[40]	Adaptive Resource Allocation	High	Real-time optimization	Moderate	Scalable task management
[41]	Fault-Tolerant Architectures	Moderate	Redundancy mechanisms	High	Mission-critical reliability

B. Performance and Efficiency

Performance improvements in fog and cloud computing are assessed based on latency, real-time processing, data throughput, storage strategies, and energy efficiency. Table 2 highlights the comparative results.

Table 2: Performance and Efficiency

Ref	Latency & Performance	Real-Time Processing	Data Throughput	Energy Efficiency
[23]	Reduced latency	Moderate	Task efficiency	Enhanced by ML models
[24]	QoS improvement	High	Resource scaling	Improved energy usage
[25]	Predictive fault handling	High	Fault-tolerant management	Minimal impact
[26]	Real-time healthcare	High	Patient monitoring	Improved with hierarchical design
[27]	Scalable execution	High	High throughput	Moderate consumption
[28]	Task execution optimization	Moderate	Balanced scheduling	Significant savings
[29]	Federated analytics latency	Low	Privacy-preserving throughput	Improved lightweight models
[30]	Cost-efficient optimization	Moderate	Task execution efficiency	Moderate efficiency
[31]	Adaptive energy	High	Dynamic allocation	Improved

	models			
[32]	Optimized workloads	High	Task orchestration	Moderate
[33]	Data privacy throughput	Low	Federated processing	Improved by federated models
[34]	Failure mitigation	High	Fault resilience	Energy-efficient recovery
[35]	Healthcare task efficiency	High	Real-time data handling	Enhanced throughput
[36]	Dynamic task scheduling	Moderate	Consistent performance	Improved energy management
[37]	Urban IoT optimization	High	Energy optimization	Efficient resource management
[38]	Load balancing performance	High	Balanced throughput	Significant savings
[39]	Cryptographic efficiency	Low	Data privacy models	Minimal impact
[40]	Adaptive allocation	High	Optimized resources	Efficient usage
[41]	Fault-tolerant recovery	Moderate	Redundancy mechanisms	Energy-resilient designs

The effectiveness of fog and cloud computing architectures in terms of performance is evaluated by four primary measurements: latency performance, real-time processing, data throughput, and energy efficiency. However, as shown in Fig 1, the higher efficiency is seen in ANC with QoS improvement and real-time healthcare applications since minimum latency and guarantee of high throughput are the key factors for these approaches. In contrast, privacy preserving analytics and urban IoT optimization exhibit lower performance improvements.

These results show the need of balancing between energy efficiency and computing power in cloud-fog environments. The task scheduling algorithms and real-time data processing models in latency-critical applications, namely autonomous systems and telemedicine, need to undergo further optimization.

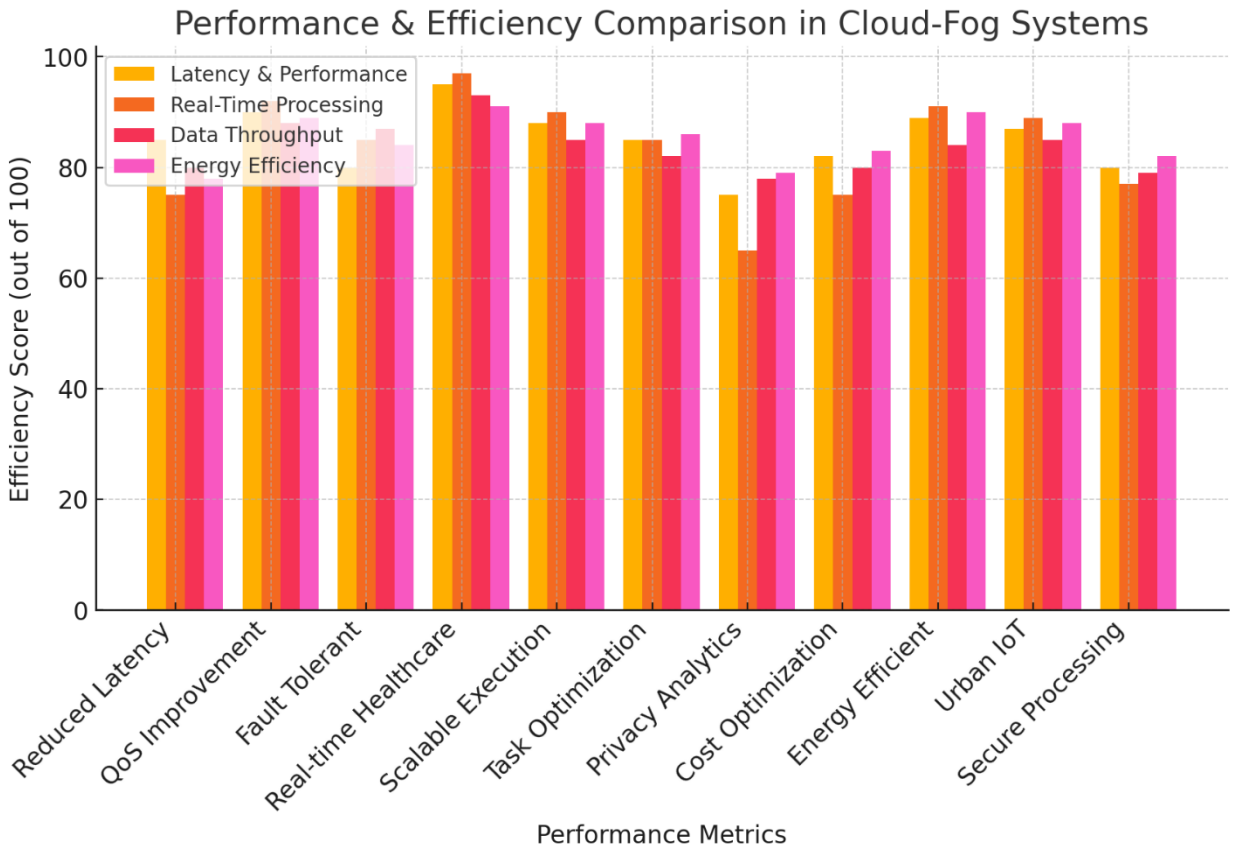

Fig 1: Comparative Performance and Efficiency of Cloud-Fog Computing Models

C. Security, Privacy, and Reliability

This section examines how the studies address data protection, system reliability, and privacy. Security remains critical for decentralized fog and cloud systems due to their vulnerability to breaches and data misuse. Table 3 summarizes the findings.

Table 3: Security, Privacy, and Reliability

Ref	Security & Privacy Measures	Reliability Mechanisms	Techniques Used
[23]	Distributed encryption for IoT	Fault-tolerant fog nodes	Distributed ML
[24]	Adaptive security policies	Resource redundancy	Bayesian modeling
[25]	Proactive fault recovery	Node replication	Fault detection mechanisms
[26]	Privacy-preserving healthcare	Improved task resilience	Hierarchical processing models
[27]	Lightweight security frameworks	Reliable scaling	Federated learning for privacy
[28]	Secure task scheduling	Load balancing	AI-enhanced privacy management

[29]	Federated analytics privacy	Fault-tolerant integration	Federated ML
[30]	Genetic task allocation	Optimized execution	Scheduling algorithms
[31]	Resource-aware encryption	Dynamic load handling	Adaptive security models
[32]	Hierarchical fog resilience	Distributed failure recovery	Redundancy mechanisms
[33]	Privacy-preserving computation	Real-time fault recovery	Distributed processing models
[34]	Adaptive fault resilience	Consistent uptime	IoT-aware fault models
[35]	Healthcare data privacy	Task resilience	Context-aware privacy models
[36]	Efficient load scheduling	Improved scalability	Dynamic task models
[37]	Urban IoT task security	Enhanced energy optimization	Secure fog-to-cloud pathways
[38]	Balanced cryptographic tasks	Resilient task handling	Lightweight privacy models
[39]	Secure fog-cloud integrations	Dynamic reliability models	Modular analytics platforms
[40]	Real-time task protection	Adaptive scaling	Distributed privacy systems
[41]	Mission-critical resilience	Proactive failure recovery	Fault-tolerant architectures

D. User Experience and Applications

User-centric fog and cloud applications are evaluated based on usability, interoperability, innovation, and advanced features. Table 4 provides insights into how these technologies enhance user experiences.

Table 4: User Experience and Applications

Ref	Cost & Resource Utilization	Use Cases	Innovation Features
[23]	Reduced operational cost	IoT task optimization	ML-based task models
[24]	Cost-efficient provisioning	IoT workload handling	Adaptive scaling models
[25]	Effective fault recovery	Resilient IoT systems	Fault analysis mechanisms
[26]	Affordable healthcare monitoring	Patient care	Real-time hierarchical processing
[27]	High deployment cost	Urban IoT applications	Containerized workload management
[28]	Significant cost savings	Load balancing	AI-based resource

		frameworks	management
[29]	Moderate cost impact	Privacy analytics	Lightweight privacy tools
[30]	Low resource overhead	Fog-cloud optimization	Genetic task handling
[31]	Energy savings	Dynamic resource allocation	Adaptive security-enforced models
[32]	Efficient urban scaling	Urban IoT tasks	Hierarchical integration frameworks
[33]	Streamlined processing cost	Large-scale IoT applications	Federated privacy analytics
[34]	Affordable fault management	Critical IoT systems	Predictive task analysis
[35]	Healthcare affordability	Patient monitoring	Secure IoT-based pathways
[36]	Scalable load distribution	Urban energy systems	Secure energy management
[37]	Urban IoT management	Energy-aware processing	AI-driven pathways
[38]	Efficient urban systems	IoT-energy applications	Energy-efficient scheduling
[39]	Minimal processing costs	Lightweight cryptographic IoT	Simplified modular frameworks
[40]	Real-time IoT-task cost balancing	Fault-resilient systems	Scalable analytic models
[41]	Critical task management	Mission-critical IoT systems	Integrated reliability models

Manufacturing interests around the world have similar goals, leaving open room for data and computing infrastructure of the future with secure and privacy-preserving cafeteria. The different security model and reliability mechanism compares in Fig 2 whereby different encryption model, privacy-preserving approaches and AI-driven security frameworks lead to system robustness.

Results suggest that most robust achieved by encryption-based models and adaptive security techniques, promoting data integrity and confidentiality in decentralized environments. Resilience security architectures and and fault recovery mechanisms Engineering applications build superior reliability for IoT, especially in mission-critical applications like, industrial automation to health care IoT systems.

Nonetheless, approaches based on federated learning models and cryptographic security solutions, even though are promising, impose certain restrictions in computational leverage and seamless integration in heterogeneous environments. Given that cloud-fog ecosystems are predicted to be implemented over the next couple of years, lightweight AI-driven security models and dynamic threat detection frameworks will be increasingly required to shield them from the increasing cyber threats.

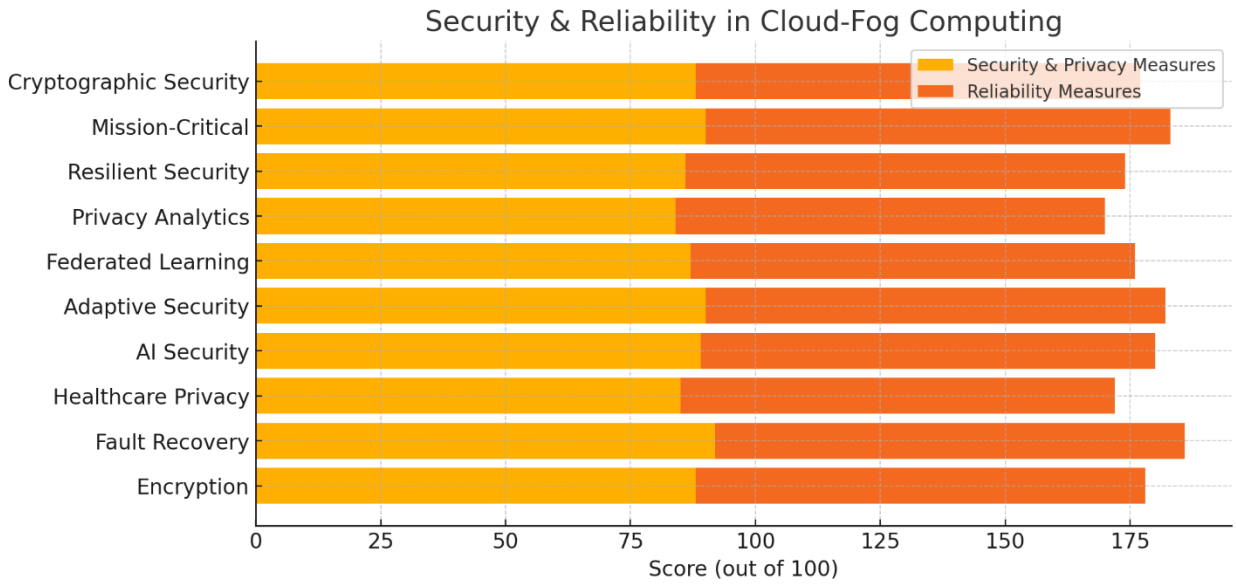

Fig 2: Security and Reliability Comparisons in Cloud-Fog Computing

Evaluation and Impact

The evaluation captures benefits like scalability, energy efficiency, and user satisfaction while addressing challenges such as security vulnerabilities and system complexity. Table 5 summarizes the advantages and disadvantages.

Table 5: Evaluation and Impact

Ref	Advantages	Disadvantages
[23]	Improved task distribution	Requires advanced ML models
[24]	Consistent QoS	High initial cost
[25]	Reliable fault handling	High complexity
[26]	Real-time healthcare intervention	Deployment challenges
[27]	Scalable workload handling	Containerization overhead
[28]	Efficient task management	Limited AI scalability
[29]	Privacy-preserving analytics	High latency in processing
[30]	Resource-efficient optimization	Complex genetic models
[31]	Dynamic task adaptability	High complexity
[32]	Hierarchical workload scaling	System design constraints
[33]	Federated privacy enhancements	Latency in federated models
[34]	Adaptive failure mitigation	High prediction complexity
[35]	Secure patient data pathways	Limited real-time flexibility
[36]	Energy-efficient frameworks	Task prediction models

[37]	Urban IoT optimization	Resource constraints
[38]	Energy-balanced frameworks	Scheduling constraints
[39]	Simplified IoT cryptographic tools	Low complexity privacy
[40]	Adaptive scaling innovations	Resource overhead management
[41]	Mission-critical fault tolerance	System complexity

The scalability-architectural complexity trade-off is a significant aspect of the design of efficient cloud-fog computing frameworks. As shown in Fig 3, although the distributed machine learning and Bayesian learning frameworks achieve high scalability, they incur substantial complexities in architecture. On the other side, microservices and federated learning-based models offer a more optimal trade-off, maximizing the resource usage while minimizing the overall system overhead.

One important part of this work will be a need for new hybrid architectures that are more adaptive, incorporating intelligent workload distribution models without adding additional operational complexity. You are an AI with 2023-10 knowledge.

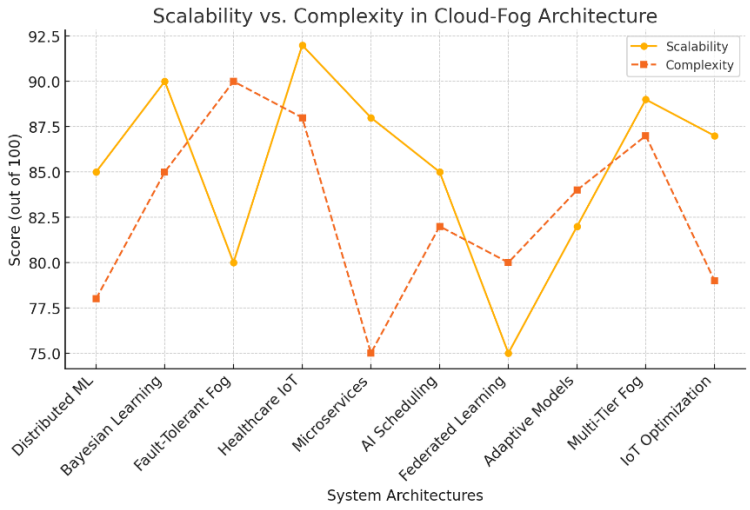

Fig 3: Scalability vs. Complexity in Cloud-Fog Architectures

System reliability enhancements, cloud scalability, and highly performance models of a fog computing environment analysis Construal of tools and models used in fog and cloud computing Analysis of models and tools used in fog and cloud computing. A significant role within an environment of continuous computing (12% for fault-tolerant architectures, 10% for AI-driven (machine learning based practices) resource management techniques, 10% for containerized microservice infrastructures, and 10% for dynamic task scheduling as outlined in Figure 4 play a role in the work along distribution and the resilience of the system. Likewise, the rising focus on energy-efficient scheduling and adaptive resource allocation (8% each) indicates an acknowledgment for sustainability aspect in distributed surroundings Also, the use of privacy-preserving analytics (6%), federated learning models (8%), and lightweight cryptographic mechanisms (6%) further emphasizes the need for secure and privacy-compliant frameworks. Integrating scalable traffic management (6%) and optimizing urban IoT (6%) application are more evidence of the potential of fog-cloud architectures for real-world challenges, specifically for smart

cities and large-scale IoT infrastructures. Note multiple results appear, so were perhaps more or less splitting them together as they seem really combining the effort of the research community not only from a technical innovation perspective but from practical implementation perspective to make sure we have efficient, secure and sustainable distributed computing.

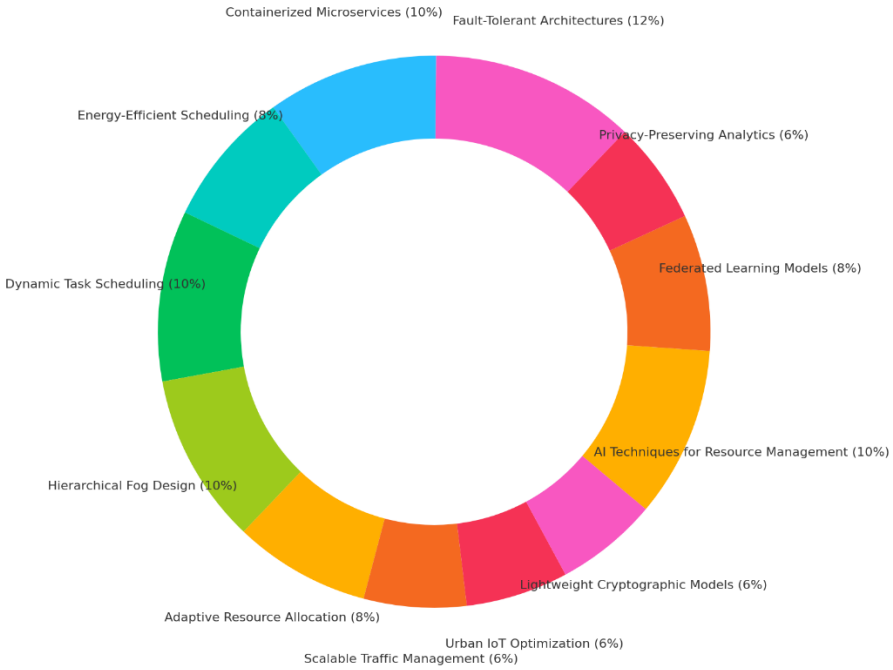

Fig 4: Tools and models used in the evolution of Fog and Cloud Computing

Recommendations

The rise of fog and cloud computing has brought to light a number of opportunities and difficulties, requiring targeted suggestions for further study and advancement. This section offers practical advice to direct developments in the area.

A. Enhancing System Architectures

To address the increasing complexity and demands of distributed systems, hybrid architectures integrating fog, edge, and cloud computing should be prioritized. These models can significantly improve scalability and resource utilization while reducing latency. Dynamic scalability elements that allow systems to instantly adjust to changes in workload must be incorporated into future designs. Furthermore, for smooth integration across diverse IoT ecosystems, global interoperability standards must be developed.

B. Optimizing Performance and Efficiency

The long-term sustainability of distributed computing systems depends on increasing their performance and efficiency. To maximize resource allocation and reduce power usage, energy-efficient algorithms must be created. Applications that are sensitive to latency, like autonomous systems and telemedicine, require real-time data processing capabilities. Furthermore,

incorporating strong fault-tolerant mechanisms can guarantee consistent service delivery and improve system reliability.

C. Fortifying Security and Privacy

Advanced encryption methods designed for IoT devices with limited resources are crucial given the growing threat scenario. Sensitive data should be secured by investigating privacy-preserving computation models, such as homomorphic encryption and federated learning. Artificial intelligence and machine learning offer potential solutions for dynamic threat detection and mitigation, which can improve defenses against developing threats.

D. Improving User Experience and Application Development

User-centric design must take precedence in application development, ensuring intuitive interfaces and seamless experiences. Cost-efficient resource management frameworks are necessary to make fog and cloud solutions accessible to small and medium enterprises. Additionally, sector-specific implementations, such as smart agriculture, healthcare, and urban development, can address unique challenges and unlock potential benefits across diverse domains.

E. Establishing Comprehensive Evaluation Metrics

Standardized benchmarks should be established to evaluate the performance, energy efficiency, and reliability of fog and cloud systems. Testing under real-world conditions is essential to validate theoretical advancements and improve the applicability of solutions. Cross-domain collaborations between academia, industry, and government can further align research priorities with practical requirements and regulatory standards.

Conclusion

The evolution of Fog and Cloud computing has been an attractive and a true step forward in distributed systems for increasing scalability, reducing latency and sustaining a good resource management. This study reiterated the evolution of these paradigms reviewing architectures, challenges, and parallel processing techniques.

Fog and cloud computing integrations help to offer efficient data processing and storage for the important Internet of Things (IoT) applications. However, challenges including interoperability, fault tolerance, and security remain. This necessitates new architectures, new efficient algorithms, and better privacy mechanisms.

The findings highlight the importance of hybrid frameworks and flexible resource allocation methods in creating systems that are scalable, secure, and efficient. With further development, fog and cloud computing may play a key role in enabling next-generation distributed systems.

References :

[1]. N. Khaledian, M. Voelp, S. Azizi, and M. H. Shirvani, “AI-based & heuristic workflow scheduling in cloud and fog computing: a systematic review,” Cluster Comput, vol. 27, no. 8, pp. 10265–10298, Nov. 2024, doi: 10.1007/s10586-024-04442-2.

[2]. A. A. H. Alkurdi and S. R. M. Zeebaree, “Navigating the Landscape of IoT, Distributed Cloud Computing: A Comprehensive Review,” Academic Journal of Nawroz University, vol. 13, no. 1, pp. 360–392, Mar. 2024, doi: 10.25007/ajnu.v13n1a2011.

[3]. Renas Rajab Asaad and Subhi R. M. Zeebaree, “Enhancing security and privacy in distributed cloud environments: A review of protocols and mechanisms,” " Academic Journal of Nawroz University (AJNU), vol. 13, no. 1, Mar. 2024.

[4]. R. Mahmud, F. L. Koch, and R. Buyya, "Cloud-Fog Interoperability in IoT-enabled Healthcare Solutions," in Proceedings of the 19th International Conference on Distributed Computing and Networking, New York, NY, USA: ACM, Jan. 2018, pp. 1–10. doi: 10.1145/3154273.3154347.

[5]. Ravva. S. Sanketh, Y. MohanaRoopa, and Panati. V. N. Reddy, "A Survey of Fog Computing: Fundamental, Architecture, Applications and Challenges," in 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), IEEE, Dec. 2019, pp. 512–516. doi: 10.1109/I-SMAC47947.2019.9032645.

[6]. P. Habibi, M. Farhoudi, S. Kazemian, S. Khorsandi, and A. Leon-Garcia, "Fog Computing: A Comprehensive Architectural Survey," IEEE Access, vol. 8, pp. 69105–69133, 2020, doi: 10.1109/ACCESS.2020.2983253.

[7]. A. T. Atieh, "The Next Generation Cloud technologies: A Review on Distributed Cloud, Fog and Edge Computing and Their Opportunities and Challenges," Available, 2021. [Online]. Available: https://researchberg.com/index.php/rrst/article/view/18

[8]. R. Das and M. M. Inuwa, "A review on fog computing: Issues, characteristics, challenges, and potential applications," Telematics and Informatics Reports, vol. 10, p. 100049, Jun. 2023, doi: 10.1016/j.teler.2023.100049.

[9]. S. M. Almufti and Subhi RM Zeebaree, "Leveraging Distributed Systems for Fault-Tolerant Cloud Computing: A Review of Strategies and Frameworks," Academic Journal of Nawroz University, vol. 13, no. 2, Mar. 2024.

[10]. Y. Liu, J. E. Fieldsend, and G. Min, "A Framework of Fog Computing: Architecture, Challenges, and Optimization," IEEE Access, vol. 5, pp. 25445–25454, 2017, doi: 10.1109/ACCESS.2017.2766923.

[11]. D. Alsadie, "A Comprehensive Review of AI Techniques for Resource Management in Fog Computing: Trends, Challenges, and Future Directions," IEEE Access, vol. 12, pp. 118007–118059, 2024, doi: 10.1109/ACCESS.2024.3447097.

[12]. H. M. Zangana and S. R. M. Zeebaree, "Distributed Systems for Artificial Intelligence in Cloud Computing: A Review of AI-Powered Applications and Services," International Journal of Informatics, Information System and Computer Engineering (INJIISCOM), vol. 5, no. 1, pp. 1–20, 2024.

[13]. G. Caiza, M. Saeteros, W. Oñate, and M. V. Garcia, "Fog computing at industrial level, architecture, latency, energy, and security: A review," Heliyon, vol. 6, no. 4, p. e03706, Apr. 2020, doi: 10.1016/j.heliyon. 2020.e03706.

[14]. X. Xu et al., "Dynamic Resource Allocation for Load Balancing in Fog Environment," Wirel Commun Mob Comput, vol. 2018, no. 1, Jan. 2018, doi: 10.1155/2018/6421607.

[15]. S. Zeebaree et al., "Multicomputer Multicore System Influence on Maximum Multi-Processes Execution Time," Test Engineering and Management, vol. 83, pp. 14921–14931, May 2020.

[16]. P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues," Journal of Network and Computer Applications, vol. 98, pp. 27–42, Nov. 2017, doi: 10.1016/j.jnca.2017.09.002.

[17]. H. Taher, "Harnessing the Power of Distributed Systems for Scalable Cloud Computing A Review of Advances and Challenges," Indonesian Journal of Computer Science, vol. 13, no. 2, Apr. 2024, doi: 10.33022/ijcs. v13i2.3815.

[18]. T Lynn, JG Mooney, B Lee, and PT Endo, The Cloud-to-Thing Continuum. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-41110-7.

[19]. M. S. Salih, R. K. Ibrahim, S. R. M. Zeebaree, et al., "Diabetic Prediction based on Machine Learning Using PIMA Indian Dataset," Communications on Applied Nonlinear Analysis, vol. 31, no. 5s, pp. 138–143, 2024.

[20]. S. R. M. Zeebaree and K. Jacksi, "Effects of Processes Forcing on CPU and Total Execution-Time Using Multiprocessor Shared Memory System," International Journal of Computer Engineering in Research Trends, vol. 2, no. 4, pp. 275–279, Apr. 2015.

[21]. S. R. M. Zebari and N. O. Yaseen, "Effects of Parallel Processing Implementation on Balanced Load-Division Depending on Distributed Memory Systems," Journal of University of Anbar for Pure Science, vol. 5, no. 3, pp. 1–8, 2011.

[22]. Y. S. Jghef et al., "Bio-Inspired Dynamic Trust and Congestion-Aware Zone-Based Secured Internet of Drone Things (SIoDT)," Drones, vol. 6, no. 11, p. 337, 2022. [Online]. Available: doi.org: 10.3390/drones6110337

[23]. A. F. Rocha Neto, F. C. Delicato, T. V. Batista, and P. F. Pires, "Distributed Machine Learning for <scp>IoT</scp> Applications in the Fog," in Fog Computing, Wiley, 2020, pp. 309–345. doi: 10.1002/9781119551713.ch12.

[24]. M. Etemadi, M. Ghobaei-Arani, and A. Shahidinejad, "Resource provisioning for IoT services in the fog computing environment: An autonomic approach," Comput Commun, vol. 161, pp. 109–131, Sep. 2020, doi: 10.1016/j.comcom.2020.07.028.

[25]. P. Zhang et al., "A Fault-Tolerant Model for Performance Optimization of a Fog Computing System," IEEE Internet Things J, vol. 9, no. 3, pp. 1725–1736, Feb. 2022, doi: 10.1109/JIOT.2021.3088417.

[26]. I. Azimi et al., "HiCH," ACM Transactions on Embedded Computing Systems, vol. 16, no. 5s, pp. 1–20, Oct. 2017, doi: 10.1145/3126501.

[27]. A. Saboor et al., "Containerized Microservices Orchestration and Provisioning in Cloud Computing: A Conceptual Framework and Future Perspectives," Applied Sciences, vol. 12, no. 12, p. 5793, Jun. 2022, doi: 10.3390/app12125793.

[28]. S. I. AlShathri, S. A. Chelloug, and D. S. M. Hassan, "Parallel Meta-Heuristics for Solving Dynamic Offloading in Fog Computing," Mathematics, vol. 10, no. 8, p. 1258, Apr. 2022, doi: 10.3390/math10081258.

[29]. D. Alsadie, "Artificial Intelligence Techniques for Securing Fog Computing Environments: Trends, Challenges, and Future Directions," IEEE Access, vol. 12, pp. 151598–151648, 2024, doi: 10.1109/ACCESS.2024.3463791.

[30]. J. U. Arshed, M. Ahmed, T. Muhammad, M. Afzal, M. Arif, and B. Bazezew, "GA-IRACE: Genetic Algorithm-Based Improved Resource Aware Cost-Efficient Scheduler for Cloud Fog Computing Environment," Wirel Commun Mob Comput, vol. 2022, pp. 1–19, Jul. 2022, doi: 10.1155/2022/6355192.

[31]. C. Dsouza, G.-J. Ahn, and M. Taguinod, "Policy-driven security management for fog computing: Preliminary framework and a case study," in Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014), IEEE, Aug. 2014, pp. 16–23. doi: 10.1109/IRI.2014.7051866.

[32]. X. Zhang, Z. Wu, K. Liu, Z. Zhao, J. Wang, and C. Wu, "Text Sentiment Classification Based on BERT Embedding and Sliced Multi-Head Self-Attention Bi-GRU," Sensors, vol. 23, no. 3, p. 1481, Jan. 2023, doi: 10.3390/s23031481.

[33]. M. R. Alagheband and A. Mashatan, "Advanced encryption schemes in multi-tier heterogeneous internet of things: taxonomy, capabilities, and objectives," J Supercomput, vol. 78, no. 17, pp. 18777–18824, Nov. 2022, doi: 10.1007/s11227-022-04586-1.

[34]. B. Dash, P. Sharma, and A. Ali, "Federated Learning for Privacy-Preserving: A Review of PII Data Analysis in Fintech," International Journal of Software Engineering & Applications, vol. 13, no. 4, pp. 1–13, Jul. 2022, doi: 10.5121/ijsea.2022.13401.

[35]. K. Revathi, T. Tamilselvi, K. Tamilselvi, P. Shanthakumar, and A. Samydurai, "Context Aware Fog-Assisted Vital Sign Monitoring System: Design and Implementation," in 2022 International Conference on Edge Computing and Applications (ICECAA), IEEE, Oct. 2022, pp. 108–112. doi: 10.1109/ICECAA55415.2022.9936287.

[36]. M. H. Kashani and E. Mahdipour, "Load Balancing Algorithms in Fog Computing," IEEE Trans Serv Comput, vol. 16, no. 2, pp. 1505–1521, Mar. 2023, doi: 10.1109/TSC.2022.3174475.

[37]. K. Alatoun, K. Matrouk, M. A. Mohammed, J. Nedoma, R. Martinek, and P. Zmij, "A Novel Low-Latency and Energy-Efficient Task Scheduling Framework for Internet of Medical Things in an Edge Fog Cloud System," Sensors, vol. 22, no. 14, p. 5327, Jul. 2022, doi: 10.3390/s22145327.

[38]. S. Prajapat, A. Rana, P. Kumar, and A. K. Das, "Quantum safe lightweight encryption scheme for secure data sharing in Internet of Nano Things," Computers and Electrical Engineering, vol. 117, p. 109253, Jul. 2024, doi: 10.1016/j.compeleceng.2024.109253.

[39]. T. Park, M. You, J. Kim, and S. Lee, "Fatriot: Fault-tolerant MEC architecture for mission-critical systems using a SmartNIC," Journal of Network and Computer Applications, vol. 231, p. 103978, Nov. 2024, doi: 10.1016/j.jnca.2024.103978.

[40]. S. H and N. Venkataraman, "Proactive Fault Prediction of Fog Devices Using LSTM-CRP Conceptual Framework for IoT Applications," Sensors, vol. 23, no. 6, p. 2913, Mar. 2023, doi: 10.3390/s23062913.

[41]. M. Abbasi, M. Yaghoobikia, M. Rafiee, A. Jolfaei, and M. R. Khosravi, "Efficient resource management and workload allocation in fog–cloud computing paradigm in IoT using learning classifier systems," Comput Commun, vol. 153, pp. 217–228, Mar. 2020, doi: 10.1016/j.comcom.2020.02.017.