Volume 1, 2025, Pages 22-31 *Elaris Computing Nexus*
| Regular Article | Open Access

Efficient Data Transformation and Machine Learning Analytics in IoT Systems Using a Hybrid Fog Cloud Approach

Ali-Khusein
First Moscow State University, Russia.
alikhusein60@gmail.com

Article Info
Elaris Computing Nexus
https://elarispublications.com/journals/ecn/ecn_home.html

https://doi.org/10. XXXX/ECN/2025003

Corresponding author(s):
Ali-Khusein, First Moscow State University, Russia.
Email: alikhusein60@gmail.com

Received 05 January 2025
Revised 23 February 2025
Accepted 30 March 2025
Available online 10 April 2025
Published by Elaris Publications.

Abstract – This study analyzes the efficacy of three data processing architectures: fog computing, cloud computing, and a hybrid fog-cloud computing framework within an Internet of Things (IoT) context. We evaluate the performance of machine learning methods regarding computation time, communication overhead, precision, and efficiency on IoT-based sensor data. The proposed hardware utilizes a Raspberry Pi as a fog node for local data processing and a cloud infrastructure for executing computationally intensive operations. Each architecture processes both raw and processed real-world IoT sensor data. The transformation procedure at the fog node compresses data, reducing its size and minimizing overhead during transmission to the cloud. The findings demonstrate that the hybrid architecture offers the best balance of execution speed, communication efficiency, and accuracy when compared to cloud-only and fog-only methods. The hybrid system leverages cloud computing capabilities and minimizes transmission costs by collecting data locally.

Keywords – IoT, Fog Computing, Cloud Computing, Hybrid Architecture, Machine Learning, Data Aggregation, Network Efficiency, Execution Time, Communication Costs, Energy Consumption, Data Transformation.

I. INTRODUCTION

In the subsequent years of the Internet of Things (IoT), context-awareness facilitates the link between the virtual computer entities and physical realm, including data processing, network connectivity, and environmental sensing. Progress allows several sophisticated IoT applications, including smart clinical systems, smart energy models, smart transportation models, and smart construction. The consolidated architecture of IoT models encompasses intelligent IoT-centric application services and the foundational IoT sensor models. The Gartner projection predicts that the worldwide IoT marketplace will reach 5.8 billion IoT-centric applications by 2020, reflecting a 21% rise from 2019 [1]. The global expansion of the IoT industry is driven by wireless networking systems and the advent of innovative technologies, including cloud platforms. This development results in a significant surge in demand for interconnected application services and IoT devices.

An IoT device is designed to gather environmental data, collaborate with other devices using network technology, and be accessible over the Internet. Consequently, measurements including as throughput, latency, and energy consumption have gained importance in evaluating the performance of IoT devices due to their constrained hardware capabilities. Cloud computing solutions facilitate the processing, collection, and interpretation of data obtained via IoT networks within acceptable timeframes. While these two components and their attributes are technical matters that need study, application-layer protocols for communication and their implementation represent one of the most essential and formidable challenges for the IoT. IoT applications favor lightweight communications protocols that accommodate the constrained hardware capabilities of the used devices and may function well within these limitations. Numerous IoT application-layer protocols for communication have been created to address specific application challenges.

Previously, IoT devices communicated only via the cloud, where all data was solely stored. Currently, fog and edge computing are used with cloud computing to enhance computational performance and address intricate issues. In the future, direct communication among gadgets on the Internet will be feasible. IoT applications aggregate substantial user data and store it at several nodes before to and subsequent to processing. Maintaining the security, integrity, and availability of IoT data encounters several obstacles [2]. In the complex landscape of the IoT, data security stands as a critical issue at the core of this linked domain. The IoT produces an enormous amount of sensitive data, making its protection a formidable challenge. A multitude of concerns emerges, including the threat of illegal access, privacy violations, and data manipulation.

The Internet of Things (IoTs) is revolutionizing businesses via immediate data exchange, although the fast proliferation of networked devices poses considerable hurdles to effective resource distribution. The main aims of IoTs include attaining superior efficiency, diminishing latency, guaranteeing security, and lowering energy usage. Migabo, Djouani, and Kurien [3] examine the necessary strategies aimed at enhancing resourceful management in IoTs. Sophisticated optimization is required to be adaptable to the diverse extent of IoT tools with divergent power requirements and capacities in IoT networks. The main approaches discussed in a number of works involve heuristic approaches, hybrid approaches, game-theoretic approaches, and machine learning approaches. The heuristic algorithms like the ACO (ant colony optimization), PSO (particle swarm optimization), GAs (genetic algorithms) can offer convenient to near-optimal solutions and are specifically necessary in the dynamic environments of IoT.

Li et al. [4] introduced their study of a resource allocation scheme optimization in a complex computing environment consisting of the IoT, cloud and fog computing. In their study, they aimed at developing an efficient meta-heuristic algorithm of resource allocation management and guaranteeing a balance in the load between layers that are connected. The authors describe the issues related to the changing demands concerning the worker-load and the capabilities of managing the fog and the cloud facilities, since the nature and diversity of the IoT objects change. They introduce a meta-heuristic method, MHHO (Modified Harris-Hawks Optimization), that is supposed to dynamically assign resources depending on the workload features and availability with the goal of spreading workloads and mitigating the risk of using excessive resources. The algorithm aims at maximizing such performance parameters as reaction time, energy consumption and utilization efficiency of resources.

This paper aims to evaluate how efficiently fog, cloud and hybrid approaches of fog-cloud architectures process IoT sensor data. The area of concern is efficiency evaluation, the time of execution, cost of communication, and accuracy of the machine learning algorithms over the raw and modified data, and to identify the best data processing approach in resource-constrained environments.

The remainder of this study has been organized using the following structure: Section II reviews related works on fog computing in IoT systems, hybrid fog-cloud architectures for IoT, and data aggregation techniques in IoT networks. Section III describes our experimental setup describing hardware configuration, software and tools used, and data transformation process. In Section IV, results on fog (Raspberry PI), cloud, and hybrid have been provided; and observations concerning data consumption over the network, and accuracy recorded. Lastly, Section V concludes the study and highlights the performance of hybrid, cloud, and fog architectures in the processing of IoT sensor data, with major focus on accuracy, communication efficiency, and execution time of ML algorithms.

II. RELATED WORK

Fog Computing in IoT Systems

As described by Mouradian et al. [5], fog computational is a newly established architecture and paradigm that reallocates the computational capacity of a conventional cloud-centric structure from central data centers to localized end-user networks and devices. Fog computing primarily diversifies cloud computing infrastructure to the network's edge, enabling a novel array of seamless applications and services for end-users. From the inception of IoT-centric applications, the number of devices linked to the web has surged to billions, and this trend is expected to continue in the foreseeable future. Conventional cloud-centric centralized systems are incapable of responding to all interlinked devices in actual-time without compromising client experience. To address issues posed by IoT architecture.

A study by Angel et al. [6] on cloud, fog, and edge computing has thoroughly examined the interplay between IoT and these computing systems, elucidating how fog and edge computing may connect cloud computing and IoT by relocating computation nearer to end-user devices, thus mitigating challenges related to energy consumption, latency, and contextual awareness. They examined various aspects of cloud, fog, and edge layers, considered as distinct tiers. In the context of device-enhanced MEC, one must account for additional complexities arising from the dynamic characteristics of device-based resource providers, including decentralized ownership, limited energy, mobility, resource limitations, improved proximity to other device-centric resource providers, and D2D (device-to-device) communication capabilities.

The technological viability of the lowest layer of device-improved MEC, whereby devices like smartphones serve as peripheral service suppliers, has been shown in models such as Honeybee, Aura, MClouds, Hyrax, and MMPI. Nevertheless, the current studies have not explored the collaboration and work-sharing mechanisms between device-based resource providers and nodes at the cloud and fog levels, taking into account diverse overheads, performance consequences, battery considerations, and other offloading characteristics.

A recent study by Fernando et al. [7] on the integration of cloud, fog, and edge has shown that little research has examined the interconnections among these three models within a 3-tier cloud-fog-edge architecture. Their study focused on cloud-based and edge-based smart cities. Furthermore, many works have reviewed fog computing application in smart urban environments, of which only one adhered to a certain design. To address the latency and location-aware vehicular use of smart urban environments via fog computing, we decided to examine the current literature to mitigate the delay and latency associated with fog computing in the development of urban environments. These papers are predicated on a comprehensive investigation of fog computing in urban environments.

Park and Yoo [8] proposed an agreement-based method for validating the correctness and stability of the vehicular information system in a fog computing network to reduce latency and ensure the integrity of the vehicular network system. They presented methodologies for resource allocation and offloading decisions inside a fog model. The enhancement issue was articulated as a hybrid integer nonlinear optimization problem. A unique three-tier design was developed in [9], minimizing each user's task reaction time by offloading tactics utilizing an extended Nash equilibrium grounded in queueing theory. In [10], Ali et al. introduced Volunteer Supported FC (VSFC), which reduced intrinsic communication latency, power expenditure, and networking use. VSFC lowers the expenses associated with sustaining Real-time IoT computing.

Singh and Singh [11] provided a comprehensive architecture for IoT-Fog-Cloud applications to enable ultra-fast services, rapid feedback times, and real-time queries by Fog task delegation to enhance Quality of Service (QoS). They introduced the dynamic optimum task delegation challenge in programmable access networks, which provides low latency and adaptable processing. They introduced an offloading strategy for dynamic computing to the network's edge, aimed at minimizing offloading costs while mitigating substantial network strain. The theoretical research indicated that this technique might minimize offloading costs and limit queue length.

Wang and Chen [12] presented a technique for delay-optimized offloading decisions and resource assignment in fog-integrated IoT networks to mitigate the strain on core network communication. They also added a mechanism of unloading to balance load of Fog. Their approach reduces latency through the Fog-to-Fog cooperation model that aims to assign requests as well as distinguish between IoT-heavy and lightweight requests. A Software-Defined Networking (SDN)-based hybrid normal data propagation architecture is used.

Hybrid Fog-Cloud Architectures for IoT

The hybrid infrastructure leverages the power of the cloud and the centralProcessing system with low latency offered by fog nodes. In addition, it fulfills a broad range of applications, including consumer services, to non-discretional industrial operations. The ecosystem can efficiently support a vast amount of data produced by IoT devices to allow real-time decision-making and analytics.

Khan et al. [13] suggested an EcoTaskSched approach that presents a BiLSTM, DNN-based network. This network is an improvement of the unidirectional LSTM. It examines the complex input patterns, such as temporal schemes, in both directions in the data of task scheduling. It contributes knowledge on the long-term time-varying dependencies that increase energy efficiency, lead to higher levels of accuracy and ensure quality of services in the fog-cloud networks. This novel extension approach utilizes the memory-based sequence modeling by accounting for both forward (historical-to-predictive) and backwards (predictive-to-historical) data relationships. It more effectively captures long-term relationships, resulting in enhanced work scheduling forecasts.

Chang et al. [14] presented a cloud-fog intrusion detection system (CFIDS) that utilizes cloud and fog resources to detect threats in contemporary research. Nonetheless, the assault impacts data packets originating from Fog layer nodes. The bulk of invasive cyber-attacks are closely related variations of previously recognized cyber assaults, sharing same data and properties. Furthermore, the data packet requires analysis to ascertain the breach. SYN packets, used to establish TCP connections, inundate a victim's network in a SYN flood attack. To alleviate the effects of SYN assaults in an FC environment, it is essential to execute the requisite security procedures.

Deep learning-based intrusion detection systems are effective for safeguarding fog computing layers from intrusions and resolving existing challenges. The authors of this study propose a novel hybrid architecture for deep learning intrusion detection in fog computing, integrating several deep learning models.

Data Aggregation Techniques in IoT Networks

As described by Tong, Hamzei, and Jafari [15], data aggregation (DA) undermines the efficacy of dispersed raw data from various sources inside the network. The aggregation within the network substantially influences the quantity and duration of sent packets, thereby minimizing energy usage and extended network longevity. The 3 major scheduling protocols for DA encapsulated in [16]: Quad basic consolidation requires that each router regularly gather all data objects and compile them after a certain interval. (ii) Periodic dispersion aggregation functions similarly to a basic periodic approach, except it transmits data immediately upon receiving information from all its subordinate nodes. (iii) Ultimately, in a regulated accumulation at intervals, every node inside the aggregate tree alters 1 hour of time according to its location and transmits the resultant conclusion.

In [17], the primary objective for using data aggregation techniques in LLNs (low-power and lossy networks) and the IoT is to reduce energy consumption and extend network longevity. Provided that every networking node has limited capability for data generation, processing, and storage, while simultaneously facilitating data interchange with neighboring nodes, the need for optimal resource usage within the network is self-evident. To achieve this objective, many data aggregation approaches are suggested to eliminate duplication from superfluous and irrelevant data, hence reducing transmission costs as well. **Fig. 1** shows the method and infrastructure of data compilations in IoT and LLN systems.

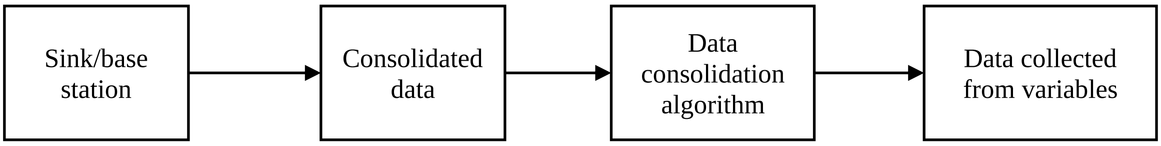

Fig 1. Data Aggregation Structure

The research conducted by Surenther, Sridhar, and Roberts [18] highlighted the application of ML for data consolidation and clustering in WSNs (wireless sensor networks) to minimize energy usages by decreasing data transmission volume. Numerous machines learning approaches, including neural networks, reinforcement learning, and swarm intelligence, have been employed. The efficacy of these approaches for network longevity, data precision, coverage, and energy use were examined, revealing that Q-learning and decision trees exhibited minimal delays. This work presented enhanced data aggregation and similarity-centric clustering through ICA (independent component analysis), which may mitigate computational demands, and energy usage. The scholars emphasized the significance of data consolidation in WSNs to diminish energy usage and enhance network adaptability in bottleneck network topology.

Ahmad et al. [19] suggested a three-layer data aggregation architecture, termed the dynamic data aggregation scheme (PDDA), which prioritizes data for sensor networks due to the substantial volume of duplicated data collected by sensors. The suggested PDDA system is a hybrid methodology that employs clustered and tree-based strategies contingent upon application types. Consequently, the suggested PDDA methodology attains energy efficiency while minimizing data processing duration and overhead at the large data server level.

III. EXPERIMENTAL SETUP

The experimental set up of the research aims at comparing the results of three different systems namely fog computing, cloud computing, and a hybrid fog-cloud system. All these architectures perform the processing of data arriving at IoT sensors differently and the experiments compare the efficiency and execution time as well as the communication costs and the accuracy of machine learning algorithms being applied to the data.

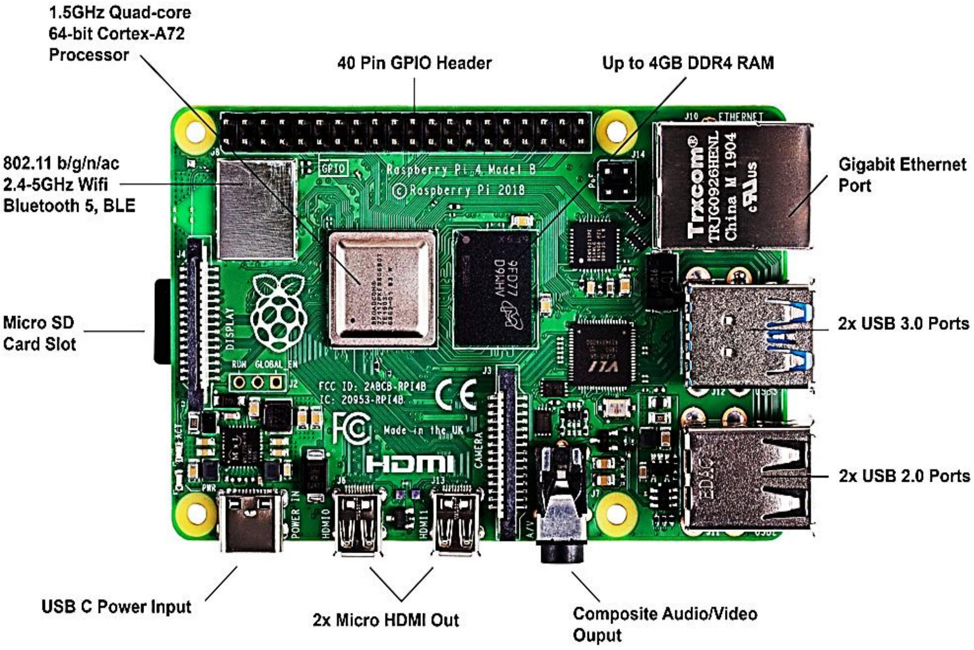

Fig 2. Raspberry Pi Model B

Hardware Configuration

The hardware implementation of the experiments was underpinned by both cloud infrastructure and local fog machines (**Fig. 3**). Fog node was developed around a Raspberry Pi single-board processor (Model B) having 4 GB of RAM (see **Fig. 2**). Such a device selection is characteristic of IoT systems where edge devices must often compute and make decisions locally without accessing a centralized cloud service. Data aggregation (data transformation) and machine learning (ML) algorithms locally through the Raspberry Pi computer simulated an IoT fog environment where computational resources are limited.

On the cloud side, a powerful computing environment was utilized, which offers the computing capacity prerequisite to run the analytics algorithms effectively. A cloud service like AWS or Google Cloud was used to process raw and transformed data, which was executed much faster because of high-performance machines. The cloud infrastructure allowed offloading of computationally-intensive actions like the execution of machine learning algorithms that needed more processing power than Raspberry Pi could support.

The experiment was designed to test the efficiency of the architectures to manage a similar dataset in the same conditions with different computational resources and data transmission constraints. The selected configuration enabled a reasonable comparison of the three configurations.

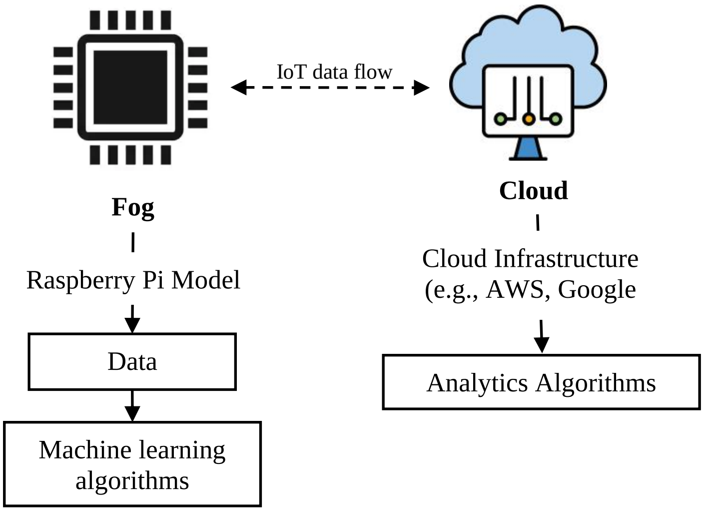

Fig 3. Hardware Configuration

Software and Tools Used

The experiments were carried out in the software environment recreating the settings of real world IoT data processing scenarios. Five different analytics algorithms (see **Fig. 4**) were implemented and executed using the WEKA Toolbox, a set of machine learning algorithms. The algorithms incorporated popular approaches in machine learning, and this aspect guaranteed the generality of the outcomes to other related IoT analytics problems. The experiments were carried out by applying the algorithms on raw data and on aggregated and feature extracted data to determine how data preprocessing (by the fog node) would affect the results.

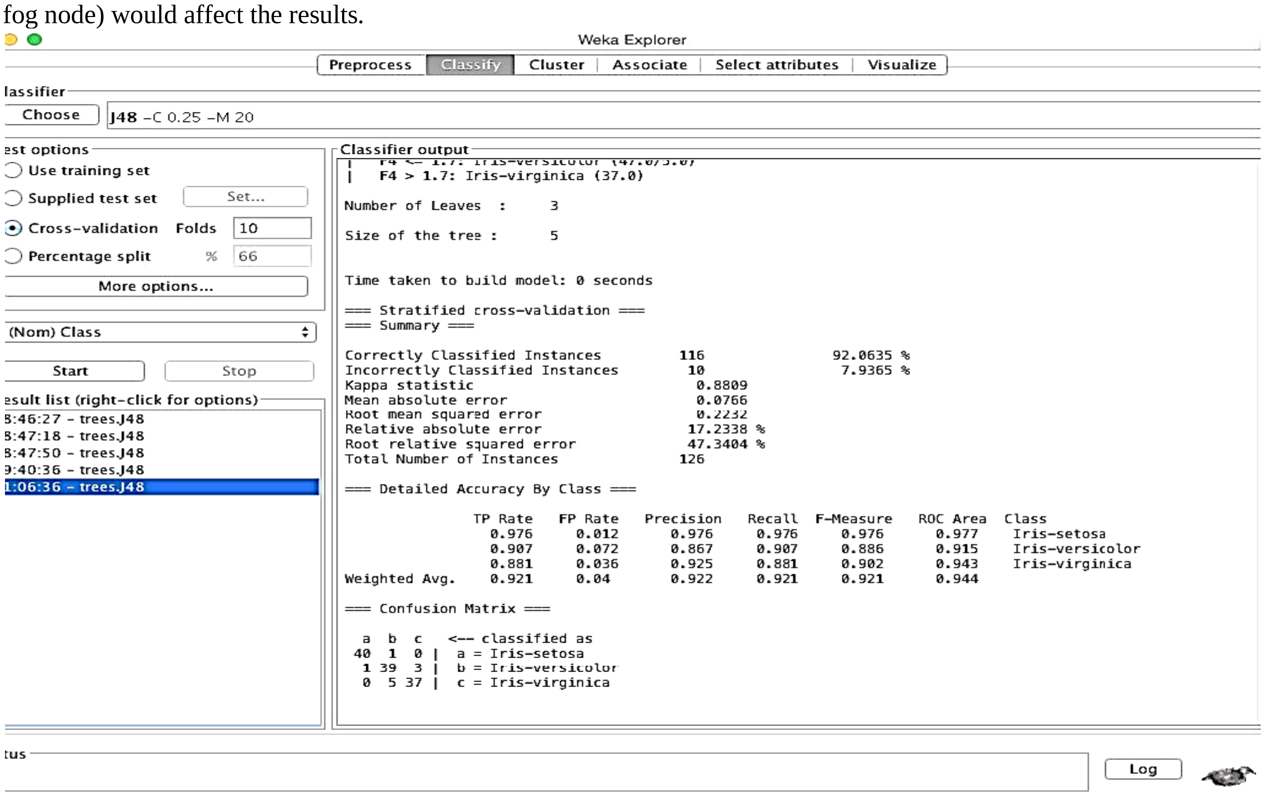

Fig 4. WEKA Toolbox

The methods of data cleaning, feature engineering, and model application were applied in Python and its corresponding libraries, such as NumPy, pandas, and scikit-learn. The processing scripts were run on Raspberry Pi of the fog architecture and on cloud-based nonphysical machines of the cloud architecture. A stable internet connection that had 1 Mbps as the upload speed was used in all experiments, to ensure that the time taken on communication between the fog and the cloud remained the same throughout the trials.

Dataset and Data Transformation Process

The dataset employed in the experiments comprises the IoT sensor data simulating data used in the real-world monitoring systems, such as environmental sensors and smart devices. This dataset consisted of about 1 million rows, and approximately 50MB in the raw state (Dataset in **Table 1**). The raw information was transformed into three formats to suit the various set ups:

- *Raw Data*: In the cloud-only and hybrid systems, raw data has been sent to the cloud to perform machine-learning, so the time to process raw data and the delay in communication could be measured.
- *Transformed Data:* In the fog and hybrid approaches, data was processed on the Raspberry Pi by summing up sensor data into meaningful features. It was during this data aggregation process that the size of the dataset was reduced tremendously, being shed off of 50MB to the current 1.2MB, and the number of rows dropped by the same margin, with the current number of rows being approximated at 5, 418. This change compressed the data being sent to the network, emulating an edge processing scenario where data can be processed at the fog node to be later analysed in the cloud.

The data transformation procedure not only provided a test of local aggregation via data transformation to minimize network congestion but also to allow a comparison between unextracted and processed data in terms of execution time and accuracy on corresponding machine learning operations. This enabled a thorough comparison of advantages and cost balances of the various data processing schemes in fog, cloud and hybrid computing.

Table 1. Synthetic IoT Sensor Data for Experimentation

Time stamp	Sensor_Id	Temperature	Humidity	Pressure	Motion_Detected
2025-01-01 00:00:00	52	22.65	64.80	1034.18	1
2025-01-01 00:01:00	93	17.91	71.79	977.25	0
2025-01-01 00:02:00	15	16.35	62.37	1005.36	1
2025-01-01 00:03:00	72	26.28	73.67	953.51	0
2025-01-01 00:04:00	61	18.24	78.00	978.40	0

IV. RESULTS

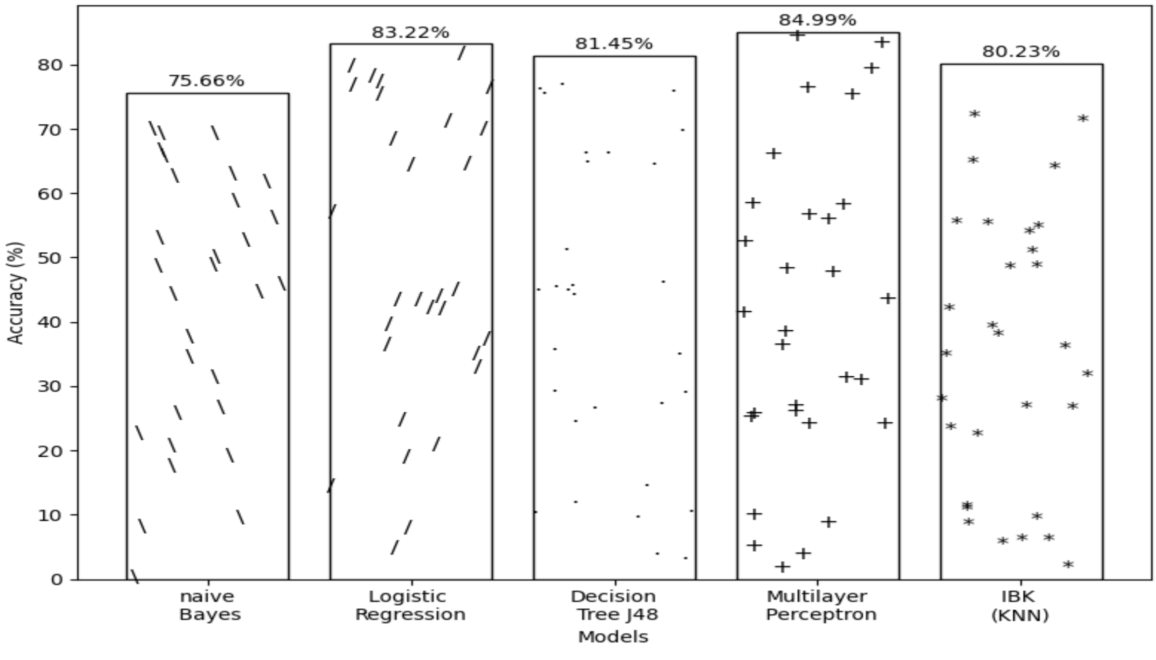

Fig 5. Precision Results

Fig. 5 presents the precision outcomes of the five analytical algorithms used on the modified data. The findings indicate that MLP (multilayer perceptron) achieves the maximum precision levels, with 100% being the optimal outcome from the cloud assessment of raw dataset. **Fig. 6** illustrates the data transmission duration from the fog (Raspberry Pi single-board computer) to the cloud system. Two bars are displayed: one representing raw dataset and the other representing processed dataset.

During the execution of this trial, the internet data transmission rate was 1 Mbps. The fog-only device incurs no data connection costs, since processing occurs inside the device without any interaction with the cloud. In the cloud method, the transmission of raw network data from fog to cloud is significantly high, but in the hybrid approach, the transmission of converted fog-to-cloud data transmission is minimal. This unsurprising outcome verifies that substantial savings on data transfers may be achieved by compiling and conditioning data early in the process.

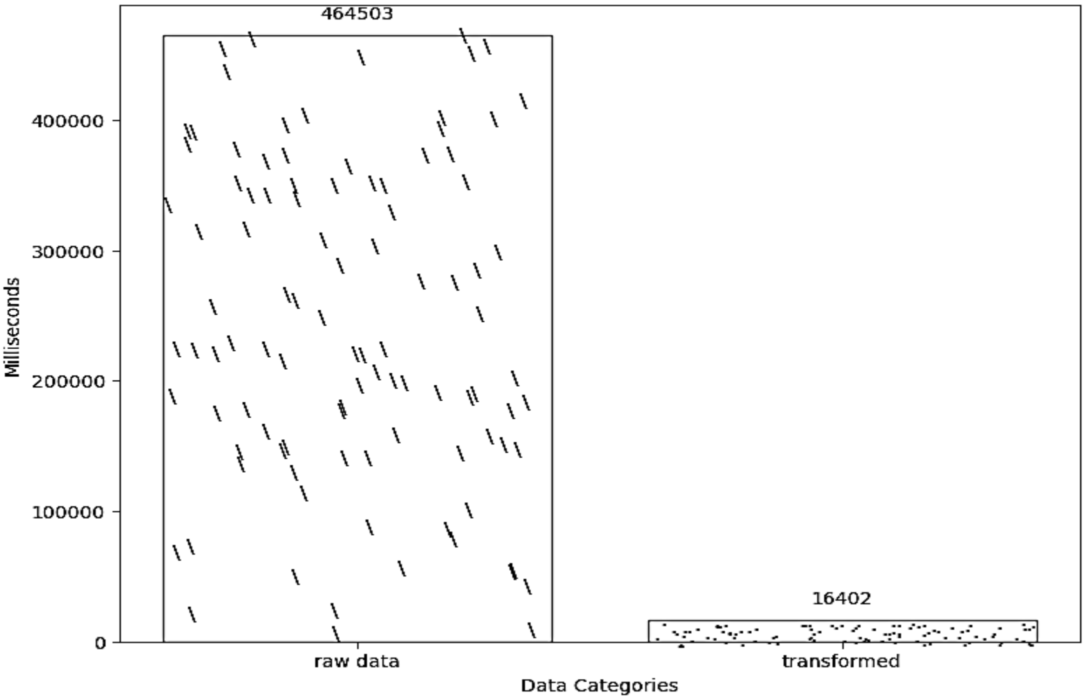

Fig 6. Data Communication

Fig. 7 depicts the execution duration of the five analytics techniques in both fog and cloud environments. The findings indicate that two techniques (MLP and logistic regression) exhibit considerable discrepancies between both sides. The IoT node inherently takes longer than cloud processing to execute analytics techniques due to its resource limitations.

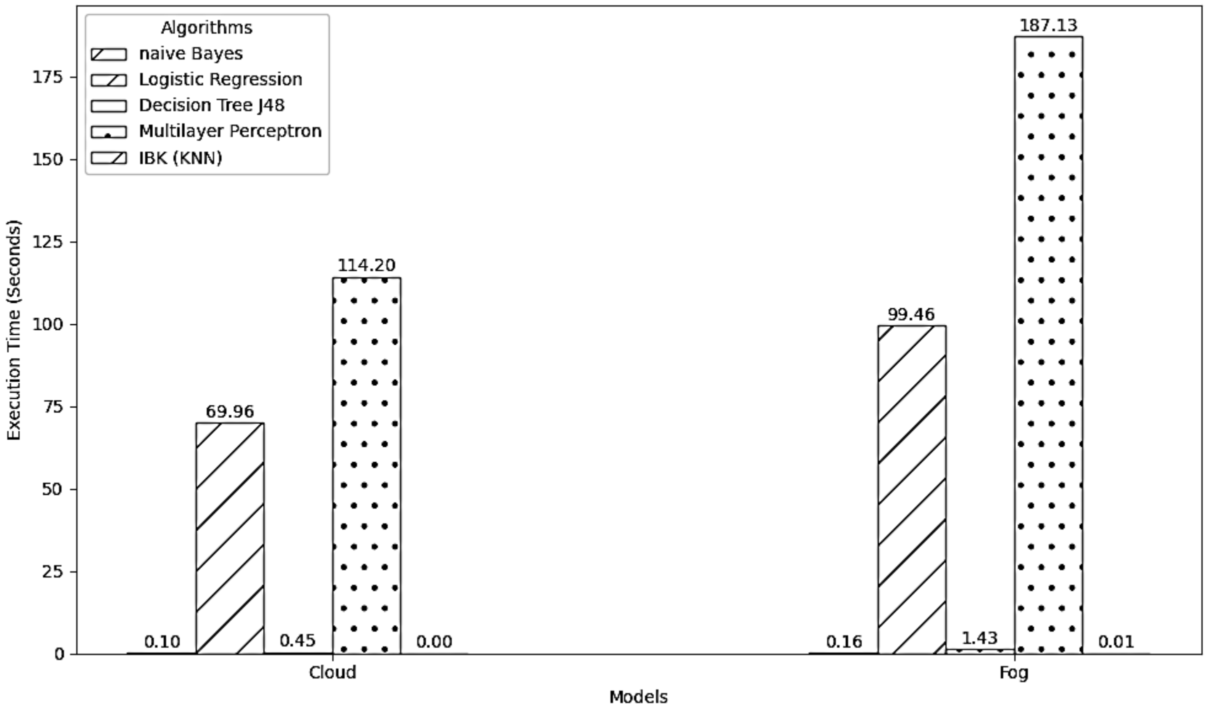

Fig 7. Execution Duration

Fig. 8 illustrates the cumulative execution duration for the 3 designs. Each architecture is evaluated based on three metrics: the execution duration of machine learning algorithms, the duration of the data conversion process, and the data transfer duration between the cloud and the local device. This graph requires more clarification, since the findings are intriguing; the specifics are as follows:

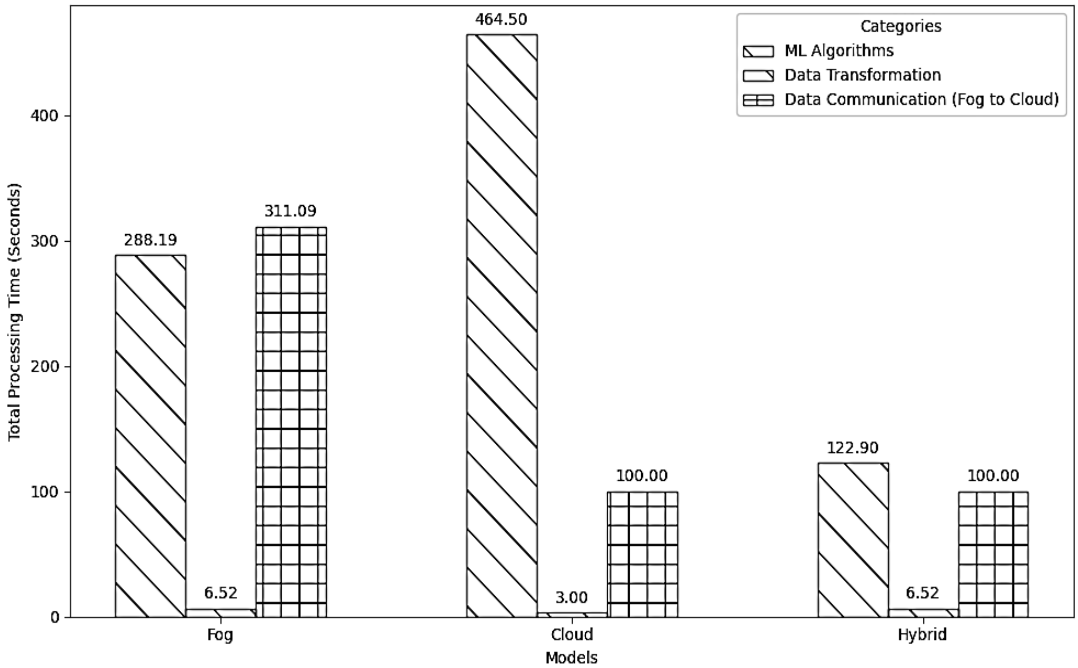

Fig 8. Total Processing Duration

Discussion of Results

Fog (Raspberry PI)

The data conversion procedure occurs nearby, and it is evident that the processing duration exceeds that of cloud-based solutions. The analytics methods were executed locally and required much more duration than cloud analysis due to limited computational capability. Nonetheless, data communication (duration required to transmit data to the cloud) is minimal as only consolidated information is forwarded to storage. The total processing time is average among the evaluated methods.

Cloud

Data communication refers to the duration required for raw data to traverse from an IoT node to the cloud system that is notably extensive due to the transmission of all raw data. Communication is the essence of the Internet of Things. It facilitates seamless device interaction, establishing a network that provides real-time information and insights. The significance of communication in IoT is in its capacity to enable prompt action and decision-making based on the disseminated data. The data conversion procedure was executed on the cloud system and, owing to the present resources, was completed rapidly. The analytics algorithms were executed on the cloud, requiring much less time than in the fog due to superior processing energy. Nevertheless, owing to the considerable transmission latency, the cloud represents the slowest method in this context.

Hybrid

The data modification procedure is executed at the edge on the Raspberry Pi, accompanied by standard observation. Data transmission refers to the duration required for modified data to traverse from an IoT device to the cloud, which remains minimal. The analytics algorithms were executed on the cloud on the changed data, requiring much less duration than local processing due to enhanced computational capacity. By integrating the numerous strengths, this results in an excellent execution time.

In most instances, fog and hybrid techniques seem similar. Nonetheless, they vary about the location for implementing the ML algorithms on the changed data, namely in both cloud and fog devices [20]. The findings indicate data. In the fog computing paradigm, all processing, including ML algorithms and data fusion, occurs at the node level that may be seen as a decentralized infrastructure. Conversely, in the hybrid method, the raw data is first integrated inside the fog node to obtain features, which are then transferred to the cloud for the application of ML algorithms. The primary advantage of the hybrid strategy compared to the fog model is the use of cloud computing capabilities for executing ML algorithms, which necessitate more processing energy.

Consequently, this phase aids in minimizing the processing duration, so contributing to the total data processing time. The preliminary findings indicate that the suggested hybrid methodology is efficacious for the selected dataset and analytical techniques. The findings clearly indicate that data transmission is efficient and yields substantial benefits.

Observations Recorded

Data usage over the network

It is acknowledged that larger data volumes incur higher consumption costs. The raw dataset included around 1 million rows, equating to roughly 50MB. Subsequent to the aggregation of data into features using data aggregation techniques, the row count diminished to 5,418 and the size lowered to 1.2MB. Early aggregation may provide substantial savings in data transmission and storage. This finding will progressively acquire significance. With the fast development in the quantity and quality of sensors, the speed and resolution of data delivery also escalate swiftly. By aggregating the data locally prior to transmission to the cloud, we are not only minimizing the volume of data but also identifying and transmitting just the relevant information. This will decrease the energy usage of fog and sensor equipment, which often get internet access via 3G, 4G, or 5G. Consequently, the batteries in these gadgets will have an extended lifespan.

Accuracy

Aggregated data results in decreased accuracy relative to the use of raw data. All proposed methodologies are similarly impacted. The overall decline in accuracy is not significant: the minimum accuracy is 75%, while the maximum is around 93%. The used analytic approach significantly affects accuracy, with trade-offs including local processing power and optimization techniques tailored to the specific environment. An appropriate equilibrium regarding resource usage, accuracy, privacy, data conversion, and energy use must be established, and our next research will delve further into this domain.

From these trials and observations, we may deduce that a crucial element for future endeavors is the formulation of a solution that integrates decentralized data compilation with analytic techniques that may also be efficiently distributed. The phrase seems incomplete; please offer some context for a fuller response. operate in low power settings. For thoroughness, we want to indicate that we first endeavored to do the studies using a RPi B 512MB board. Nonetheless, this was unable to execute some analytical methods from the WEKA data mining tool owing to RAM limitations. Consequently, we used a little more potent variant, as previously stated.

V. CONCLUSION

We evaluated the performance of fog, cloud, and fog-cloud architectures in processing Internet of Things sensor data, focusing on execution time, network efficiency, and the efficacy of machine learning techniques. Although cloud-only systems possess significant computational capacity, they incur minimal data transfer cost due to the reception of raw data. The fog-only strategy offers advantages in local processing, whereas its shortcomings, previously noted, pertain to execution time limitations imposed by hardware constraints. The hybrid architecture attains an optimal equilibrium between local data aggregation at the fog node and the offloading of computationally demanding activities to the cloud, hence minimizing communication expenses and execution duration.

CRediT Author Statement

The author reviewed the results and approved the final version of the manuscript.

Data Availability

The datasets generated during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interests

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Funding

No funding was received for conducting this research.

Competing Interests

The authors declare no competing interests.

References

[1]. U. O. Matthew, O. Asuni, and L. O. Fatai, “Green Software Engineering Development Paradigm,” in Advances in systems analysis, software engineering, and high performance computing book series, 2024, pp. 281–294. doi: 10.4018/979-8-3693-3502-4.ch018.
[2]. O. I. Abiodun, E. O. Abiodun, M. Alawida, R. S. Alkhawaldeh, and H. Arshad, “A review on the security of the Internet of Things: Challenges and Solutions,” Wireless Personal Communications, vol. 119, no. 3, pp. 2603–2637, Mar. 2021, doi: 10.1007/s11277-021-08348-9.
[3]. E. M. Migabo, K. D. Djouani, and A. M. Kurien, “The narrowband Internet of Things (NB-IoT) resources management performance state of art, challenges, and opportunities,” IEEE Access, vol. 8, pp. 97658–97675, Jan. 2020, doi: 10.1109/access.2020.2995938.
[4]. X. Li, Y. Liu, H. Ji, H. Zhang, and V. C. M. Leung, “Optimizing resources allocation for FOG Computing-Based Internet of Things networks,” IEEE Access, vol. 7, pp. 64907–64922, Jan. 2019, doi: 10.1109/access.2019.2917557.
[5]. C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, “A comprehensive survey on FoG Computing: State-of-the-Art and Research challenges,” IEEE Communications Surveys & Tutorials, vol. 20, no. 1, pp. 416–464, Nov. 2017, doi: 10.1109/comst.2017.2771153.
[6]. N. A. Angel, D. Ravindran, P. M. D. R. Vincent, K. Srinivasan, and Y.-C. Hu, “Recent advances in evolving computing paradigms: cloud, edge, and fog technologies,” Sensors, vol. 22, no. 1, p. 196, Dec. 2021, doi: 10.3390/s22010196.
[7]. N. Fernando, S. Shrestha, S. W. Loke, and K. Lee, “On Edge-Fog-Cloud collaboration and reaping its benefits: a heterogeneous Multi-Tier Edge Computing architecture,” Future Internet, vol. 17, no. 1, p. 22, Jan. 2025, doi: 10.3390/fi17010022.
[8]. S. Park and Y. Yoo, “Network Intelligence based on network state information for connected vehicles utilizing FOG computing,” Mobile Information Systems, vol. 2017, pp. 1–9, Jan. 2017, doi: 10.1155/2017/7479267.
[9]. Y. Chen, J. Zhao, Y. Wu, J. Huang, and X. Shen, “QOE-Aware Decentralized Task Offloading and Resource Allocation for End-Edge-Cloud Systems: A Game-Theoretical Approach,” IEEE Transactions on Mobile Computing, vol. 23, no. 1, pp. 769–784, Nov. 2022, doi: 10.1109/tmc.2022.3223119.

[10]. B. Ali, M. A. Pasha, S. U. Islam, H. Song, and R. Buyya, "A Volunteer-Supported FOG computing environment for Delay-Sensitive IoT applications," IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3822–3830, Sep. 2020, doi: 10.1109/jiot.2020.3024823.

[11]. P. Singh and R. Singh, "Energy-Efficient Delay-Aware task offloading in FOG-Cloud Computing System for IoT sensor applications," Journal of Network and Systems Management, vol. 30, no. 1, Oct. 2021, doi: 10.1007/s10922-021-09622-8.

[12]. Q. Wang and S. Chen, "Latency-minimum offloading decision and resource allocation for fog-enabled Internet of Things networks," Transactions on Emerging Telecommunications Technologies, vol. 31, no. 12, Jan. 2020, doi: 10.1002/ett.3880.

[13]. A. Khan, F. Ullah, D. Shah, M. H. Khan, S. Ali, and M. Tahir, "EcoTaskSched: a hybrid machine learning approach for energy-efficient task scheduling in IoT-based fog-cloud environments," Scientific Reports, vol. 15, no. 1, Apr. 2025, doi: 10.1038/s41598-025-96974-9.

[14]. V. Chang et al., "A survey on intrusion detection Systems for fog and cloud computing," Future Internet, vol. 14, no. 3, p. 89, Mar. 2022, doi: 10.3390/fi14030089.

[15]. X. Tong, M. Hamzei, and N. Jafari, "Towards Secure and Efficient data aggregation in Blockchain-Driven IoT Environments: A Comprehensive and Systematic study," Transactions on Emerging Telecommunications Technologies, vol. 36, no. 2, Feb. 2025, doi: 10.1002/ett.70061.

[16]. X. Xu, X.-Y. Li, P.-J. Wan, and S. Tang, "Efficient scheduling for periodic aggregation queries in multihop sensor networks," IEEE/ACM Transactions on Networking, vol. 20, no. 3, pp. 690–698, Sep. 2011, doi: 10.1109/tnet.2011.2166165.

[17]. J. V. V. Sobral, J. J. P. C. Rodrigues, R. a. L. Rabêlo, J. Al-Muhtadi, and V. Korotaev, "Routing protocols for low power and lossy networks in internet of things applications," Sensors, vol. 19, no. 9, p. 2144, May 2019, doi: 10.3390/s19092144.

[18]. I. Surenther, K. P. Sridhar, and M. K. Roberts, "Enhancing data transmission efficiency in wireless sensor networks through machine learning-enabled energy optimization: A grouping model approach," Ain Shams Engineering Journal, vol. 15, no. 4, p. 102644, Jan. 2024, doi: 10.1016/j.asej.2024.102644.

[19]. I. Ahmad et al., "Adaptive and Priority-Based data Aggregation and scheduling model for wireless sensor network," Knowledge-Based Systems, vol. 303, p. 112393, Aug. 2024, doi: 10.1016/j.knosys.2024.112393.

[20]. A. A. Alli and M. M. Alam, "SecOFF-FCIoT: Machine learning based secure offloading in Fog-Cloud of things for smart city applications," Internet of Things, vol. 7, p. 100070, Jun. 2019, doi: 10.1016/j.iot.2019.100070.