Wiley
International Journal of Distributed Sensor Networks
Volume 2025, Article ID 5573818, 13 pages
https://doi.org/10.1155/dsn/5573818

Research Article

Quality of Experience Aware Service Selection Model to Empower Edge Computing in IoT

Bipasha Guha Roy,[1] **Deepsubhra Guha Roy**,[1] **Piyali Datta**,[2] **Surbhi Bhatia Khan**,[3] **Abdullah Albuali**,[4] and **Ahlam Almusharraf**[5]

[1]*IEM Centre of Excellence for Cloud Computing & IoT, Department of CSE (AIML), Institute of Engineering & Management, University of Engineering & Management, Kolkata, India*
[2]*IEM-IIT Mandi Centre for Joint Research on Human Computer Interaction, Department of CSE (AIML), Institute of Engineering & Management, University of Engineering & Management, Kolkata, India*
[3]*School of Science, Engineering and Environment, University of Salford, Manchester, UK*
[4]*Department of Computer Networks and Communications, College of Computer Sciences and Information Technology, King Faisal University, Al Ahsa, Saudi Arabia*
[5]*Department of Management, College of Business Administration, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia*

Correspondence should be addressed to Surbhi Bhatia Khan; s.khan138@salford.ac.uk and Abdullah Albuali; aabuali@kfu.edu.sa

Received 28 February 2025; Accepted 25 April 2025

Academic Editor: Shikha Jain

Quality of experience-aware service selection can significantly remove well-known scalability issues of an Internet of Things (IoT) architecture. In traditional IoT architecture, several heterogeneous data streams from connected nodes are transmitted through gateways to the remote mobile cloud servers. The entire procedure is time- and energy-consuming if the target dataset is comparatively small and uninterrupted. Also, using this conventional technique, the reliability grade drops significantly to meet additional security-related quality of service (QoS) requirements compared to the service cost. We propose a quality of experience-aware task rescheduling model using edge modules that offer territory-based three-layered edge IoT data analysis and service selection. The observation module at the application layer takes a near-optimal remark upon each usage metric having distinct QoS components. Meanwhile, the QoS manager at the network layer handles network traffic due to the load associated with heterogeneous service needs. The precision of the knowledge is assured to the service manager through the sensing layer with few adaptability characteristics towards assorted service requests. The proposed three-layered energy-efficient model helps minimize data delivery time with minimal cost and optimized quality assurance for service-based IoT infrastructures like smart agriculture, patient monitoring, and student monitoring.

Keywords: data computation; edge computing; Internet of Things; quality of experience; quality of service

1. Introduction

IoT is fundamentally a composite of wireless and wired connectivity networks or organizations that are developed with the help of wireless sensor/mess network, radio frequency identification (RFID), and WLAN-like traditional network connectivity [1]. IoT connectivity can coherence with any functional device, which causes a huge amount of heterogeneous data and makes the network troublesome to fulfil unnumbered assorted user demands. Most of the time, user application claims for accurate structured data than inconvenient and miscellaneous unstructured data. In these circumstances, data analysis effectively helps the IoT applications to become self-revised human assistants decently. Fog servers, cloud servers, and edge servers are usually providing data analytical platforms through wireless connectivity accordingly. Several regional-based IoT network like smart home, smart grid, and smart farming mostly prefer edge IoT platforms rather than remote server activity due to high service cost, higher energy consumption of IoT nodes, elevated response time, and low data security [2, 3]. On the other hand, edge servers, having lower databases and short battery lifetime, lag in making accurate decisions within the stipulated time. Some of the recent articles explained the probable exploration of fog-IoT [4, 5] and edge IoT [6, 7], whereas very few papers thought about the quality of service (QoS) metric to satisfy individual demand for service [8, 9]. Some research works consider QoS-aware cloud computing for traditional wireless networks [10, 11]. However, those are not equally effective for IoT networks due to their scalability issues. Recent research works conclude that edge computing is a highly recommended model for small subject-based IoT networks [12, 13] having some early discussed issues. Therefore, this paper suggests a QoS model consisting of three layers to support service-based edge IoT that includes some features as follows:

A. A promising quality data analysis and publishing module has been developed for edge IoT architecture.

B. Towards optimization, a decision-making algorithm has also been designed to make choices for edge-service selection.

C. Supportive analysis of the proposed model has been done through an emulation platform design in the laboratory environment.

D. A thorough comparison test has been performed, revealing that the proposed architecture is more efficient than previous related works. The packet delivery rate is also under consideration.

We followed a three-layered analysis of QoS provisioning for easy data analysis-based decision-making over the service, network, and mobile-edge layers. Practical fields of industrial IoT like agriculture, education, and medicine can easily adopt this top–down analysis approach used in decision-making to guarantee QoS through edge computing. The proposed framework helps select the most likely service with the help of different service parameters like data load, available bandwidth, and future probable data hazard, which are the guiding attributes for optimizing the QoS metric in IoT.

Apart from these, the article consists of details of the proposed QoS-aware edge IoT architecture and definitions of service-metrics parameters for knowledge-oriented IoTs in Section 2. Section 3 is about the result outcomes measurement. This section also analyzes the feasibility of the proposed solution. Section 4 concludes with some future works related to this study.

2. Quality Aware Service Selection

A service selection mechanism rarely emphasizes the quality of experience over service selection infrastructure. Actual service experience has a huge difference from expected or promised service [14, 15]. IoT networks are highly intriguing in nature, which makes any service provider difficult to match or predict expected service quality in real-time. The only possible way out is to consider a live QoS manager module for near-optimal service scheduling [16]. This model proposed to consider the alive-QoS management unit after the task queue and before the decision-making module. This module is a connecting module between two other layers, called the sensing layer and the application layer. This module acts as a quality of experience monitoring and assuring the unit. This unit considers both the QoS matrices with real-time resource consumption and is mapped to the user expectation from the historical data. We can define the overall circumstance in this manner: assuming A denotes the resultant metric of provided QoS and γ denotes the confidence level possible resultant metric with an option of update. These metrics depend upon other two metrics: $E(A)$ and $D(A)$. $D(A)$ and $E(A)$ are experience metric and expected experience metric, respectively.

On the other hand, the considered confidence interval is $\left(E(A) \pm \sqrt{(D(A)/(1-\gamma))}\right)$ in each service attempt. The composite architecture of QoS-aware edge IoT has a clear selection for three layers. The modules into the layers in top to bottom approach are as follows:

A. Edge IoT sensor data collection layer

B. Edge IoT server at the network layer

C. Web application layer

Depending upon the QoS provision and cost factors, we have considered user demand, network layer capacity, and available bandwidth to authorize the QoS optimization unit of our proposed edge IoT model. Apart from these real-time factors, we also have considered a few predefined factors of IoT networks like the number of nodes, sensing range, and subject of sensing at the same time. These factors altogether make the proposed edge IoT network more accurate and adaptive in nature. These characteristics have been suitably formulated and defined in different layers in the next section. Figure 1 shows the layer dependent prototype model with decision-making module and convenient components in various layers. Table 1 consists of short description of each variables initialized through proper mathematical description.

2.1. Edge IOT Sensor Data Collection Layer. Sensor data collection is the first and foremost step of any IoT network. A designed IoT network for any specific area having distinct applications can face fundamental concerns such as limited processing units, low battery capacity, and small lifetime. Also, random deployment of sensor nodes can damage the possible utilization of the overall network. To exemplify, an IoT network designed for a healthcare unit of a hospital can give better results if the IoT nodes are motionless. However, designing an optimization algorithm for dynamically changing IoT devices in a vehicular network is quite challenging [17, 18].

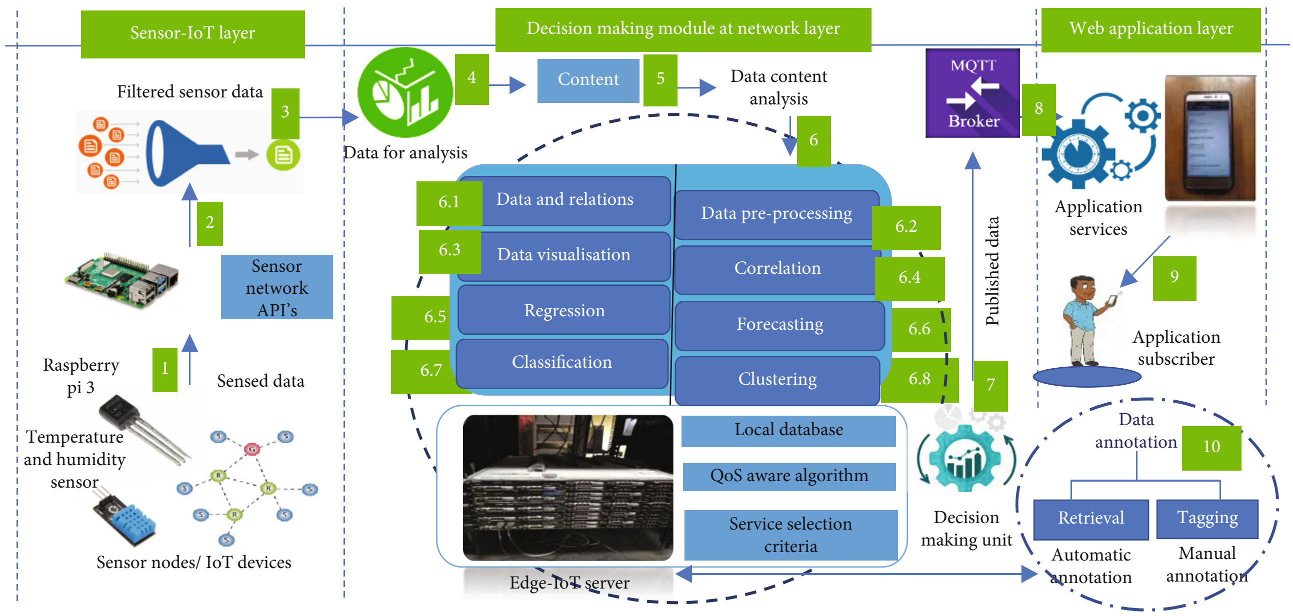

Figure 1: Three-layered architecture of edge IoT service selection method for statically allocated IoT devices.

TABLE 1: Variable description.

Variable term	Description
S	Set of states
P	Probable action utility function
$F^{\rho}(s)$	A sequence of states which fits to the policy ρ
t	Time instance of IoT service consumption
l	The costing in term of bandwidth selection over state transition
e	The measured power needed to transit one service state to another
E	Consumed energy by the edge node
α	Quality of service reward associated with individual edge
bw_{mn}	Available bandwidth
e_{mn}	Possible consumed power

2.2. Edge IOT Server at Network Layer. This layer of the proposed architecture consists of a decision-making module [19] with the assistance of edge server programmed with a designed QoE-aware algorithm (Algorithm 1), previous data related to the subscribers, regional database, and service selection criteria. The decision-making module primarily maps user demand with the available resources [8, 9]. The service criterion metrics and users' previous history help the module to make a satisfied end-user service selection in a faster energy and cost-effective manner than other cloud service methods. A backend machine learning process helps to turn this module into a self-revised one itself.

2.3. Web Application Layer. This is the adjacent layer to the end-user or subscriber community [2, 8, 9]. Preferably, two data matrices are collected from this layer. One is at the beginning of a service provision and another is after a data subscription. After sign up for a paid subscription to a service package from the broker, a basic survey matrix is reported to both the service ends according to the terms and conditions [12, 15]. Another matrix is generated when the user subscribes to some data from the resource pool. In both cases, the generated data will automatically acquiesce within the local database with a unique primary key, that is, subscription ID.

2.3.1. Task Model With Service Quality Measurement. Let us consider a DMP (decision-making procedure) possessing four distinguished tuples $D(S, X, P, R)$, where X forms a set of states termed as S. P having two possible states, such as "on" and "off" and being able to convert from one state to another, is called probable action utility function $P : \in [0, 1]$. If an action earns some reward points compared to the growth from the last stage, then it will be defined as $R : S \times X \longrightarrow R$, from last status s' to a new "on" or "off" state.

The proposed model is highly capable of focusing on quality of experience policy. This measurement indirectly sums up the service rewards and upgrades itself as per the expected range [20]. Sometimes, the compensation could be done via service discount without compromising the quality assurance. Finite upgradation of the policy then counts as increased $E(R)$.

$$F^{\rho}(S) = \arg\max_{x \in X} \left(R(s, x) + \alpha \sum_{s^{\alpha} \in S} p\left(s, x, s'\right) F^{\gamma}(s) \right). \quad (1)$$

From Equation (1), we can achieve $F^{\rho}(S)$, a sequence of states which fits to the policy ρ. Another equation can be considered, that is,

```
Initialization: F_0(S), i = 0, α, f_0, e;
Output: F_i(S);
Repeat For i = 0 to I as:
    Repeat For n = 1 to N as:
        For s_i ∈ S calculate ∑_{s_{i-1}∈S} R(s_i, x, s_{i-1}). Calculate
        F^ρ(S) = arg max_{x∈X} (R(x, s) + α ∑_{s^α∈S} p(s, x, s′) F^γ(s));
        Until ||f_n − f_{n−1}|| ≤ e;
        End For;
        f(s_i) ⟵ arg max_{x∈X} f′(s) = {f′(s_1), f′(s_2), ⋯ f′(s_N)};
        i ⟵ i + 1;
    End For;
        F_i(S) = max_{x∈X} (R(x, s) + α ∑_{s′} p(s, x, s′).F_{x−1}(S) + F_0(S));
End
```

ALGORITHM 1: Edge-service selection for iterative model.

$$R(s, x) = \sum_{s' \in S} R(s, x) \times p\left(s, x, s'\right). \tag{2}$$

Equation (2) helps to obtain the optimal policy of decision-making. Further, Equation (2) can be evaluated through the following equation:

$$F(S) = \max_{x \in X} \left(R(x, s) + \alpha \sum_{s' \in S} p\left(s, x, s'\right) F^{\gamma}(s) \right). \tag{3}$$

At a point of $x \longrightarrow \infty$, the function $F(S)$ obtain its optimal policy value.

Considering a real-time scenario, based on delay consideration, three tuples can be picked: (t, l, e). Consumed IoT service can be defined with these three tuples where t designates the time instance of IoT service consumption and can be defined as $t \in \{t_1, t_2, \cdots t'\}$ over the service utilization. e is the measured power needed to transit from one service state to another. l reflects the cost in terms of bandwidth selection over state transition. As per the user demand, if a service state transit from s to s' with an action of $x \in X$, then the probability transition matrix will be defined as $p(s, x, s')$, where $s = (t_i, l_i, e_i)$ and $s' = \frac{(t_j, l_j, e_j)}{(t_k, l_k, e_k)}$; $j = i + x, k = i - x$. Similarly, other service variables are measured to map the availability with the user demand, like available bandwidth probability and probable extra energy consumption, as $p_{bw}(bw_i, bw_{\pm x})$ and $p_{xt}(xt_i, xt_{\pm x})$, where

$$p_I\left(i_j, i_k\right) = \begin{cases} 1 & \text{when } j = k = 1, \\ -1 & \text{when } j + 1 = k, \\ 0 & \text{for otherwise.} \end{cases} \tag{4}$$

We have considered Equation (4) as a predefined state model and has been modified towards bandwidth selection as

$$p_{\mathrm{BW}}\left(bw_j, bw_k\right) = \begin{cases} p_{\mathrm{BW}}^{\mathrm{same}} & \text{when} \quad j == k \quad , \\ 2p_{\mathrm{BW}}^{\mathrm{change}} & \text{when} \quad j = BW, \quad j = 2\mathrm{BW}, \\ p_{\mathrm{BW}}^{\mathrm{change}} & \text{when} \quad |j + k| = 1 \quad , \\ 0 & \text{Otherwise} \quad . \end{cases} \tag{5}$$

However, Equation (6) helps to measure probable energy consumption for the accomplishment of a data operation when

$$x \in X : \ p_E\left(e_j, e_k\right) = \begin{cases} 1(a) & \text{when} \quad j == E \quad , \\ p_e & \text{when} \quad j = BW, \quad , \\ 1 - p_e^{(a)} & \text{when} \quad j + 1 = k \quad , \\ 0 & \text{Otherwise} \quad . \end{cases} \tag{6}$$

Effective bandwidth is the major factor in terms of the reward function estimation. Edge nodes help us with this information. Therefore, the ultimate reliability aspect of the proposed model can be summed up by the below-defined Equation (7):

$$\text{Reward, } R((t, l, e), x) = \begin{cases} -R^{\mathrm{powerout}} & (\text{if } e = E, t < T, l < L), \\ k_R.t.l.e & \text{otherwise.} \end{cases} \tag{7}$$

If a single edge node is consumed at a time for a single task confirmation, once in a cycle, then Equation (6) can be solved effortlessly. Equation (7) reflects the reward value with k_R, proportionality constant, initialized on a state, with $F_{0(s)}$. This summative resultant value is one of the effective factors as it updates continuously for an iteration of $(i \leq I, i > 0)$ shown in Equation (8). α denotes the QoS reward associated with an individual edge.

$$F_i(S) = \max_{x\in X}\left(R(x,s) + \alpha\sum_{s'} p\left(s,x,s'\right).F_{x-1}(S) + F_0(S)\right). \tag{8}$$

Furthermore, this reward function helps in the quality of experience assurance. For a particular IoT network, it is considered in the following Equation (9):

$$R(s,x) = \sum_{s,s'\in S} R\left(s,x,s'\right)p\left(s,x,s'\right). \tag{9}$$

where the optimal function of service selection calculated through the following Equation (10):

$$F_n(s) = \sum_{x\in X} p\left(s,x,s'\right)\left[R(s,x) + \alpha F_{n-1}\left(s'\right)\right]. \tag{10}$$

The optimal policy function for the IoT services β is defined as $f'(s) = \{f'(s_1), f'(s_2), \cdots f'(s_N)\}$.

2.3.2. Decision Formation at the Network Layer. The network layer consists of two kinds of service selection options. The first consists of four phases: data and relations gathering, data visualization, regression, and classification. Another service selection strategy has four stages: data preprocessing, correlation finding, forecasting, and clustering. Depending upon the real-time data traffic, the network layer service selection module instantly makes the right decision over the two mentioned phases. Further, these two service selection procedures for our proposed prototype model have been named execution-based real-time service selection and z-based service selection.

1. *Execution-based real-time services*: Decision is taken after a service cycle completion. Most of the jitter-sensitive services are observed by their expected service completion time and delay overhead. Another measurement policy is called z-based service.

2. *z -based services*: These IoT services especially focus on long-term basis service measurement factors such as peer-to-peer applications, location, and other non-predetermined variables. A minimum $\mathrm{Rate}_{\min}^{(i)}$ for $i = 1, 2, \cdots k$. A rate of provisional connectivity is required to maintain this kind of network model, where k stands for different nontiming variables.

IoT networks suffer from privacy issues which are tough to overcome while proposing dynamic network flow considering IoT applications such as vehicular networks [13]. There, we need to depend on customized models like the "trust cascading–based emergency message dissemination" model [21] and the "privacy-preserving reputation updating scheme for cloud-assisted vehicular networks" model [22]. Also, advanced fields like federated learning-based digital twin solutions must incorporate IoT networks securely [23]. Our proposed model has been developed into an environment of private edge connectivity and needs to be improved, such as for secured dynamic data handling. In this state, the proposed model is beneficial and advantageous in static IoT network implementation areas like agriculture [18], patient monitoring, and student monitoring [1]. The proposed model has been designed to ensure the same QoS to those subscribers who are paying the equivalent [9]. Therefore, we need to optimize the proportional fairness of subscribed services [24, 25]. Considering the direction, we compared service selection and QoS in terms of different days over different activities captured from the same IoT network, graphically represented in the result outcome section. Though the provider is the same during different data collection phases, the criteria to fulfil distinct quality experiences are different. We incorporated encryption methods and safe data transfer protocols (e.g., TLS/SSL) to secure IoT sensitive data. Furthermore, the designed QoS manager should be updated regarding the real-time service consumption and should be able to decide on the mapping of the edge node with a suitable resource solution. The N numbers of layer-2 nodes are connected to the M numbers of sensor layer edge nodes in a mess network connectivity. The calculated throughput depends upon two major factors: available bandwidth (bw_{mn}) and possible consumed power (e_{mn}). Therefore, the resource is allocated depending upon the reward factor associated with the possible allocation of resources upon energy consumption at a time instance i. We have considered network connectivity between n numbers of network layer nodes with m numbers of sensing layer nodes where $n \in N$ and $m \in M$. The calculated throughput is shown in Equation (11).

$$r_n = \sum_{m=1}^{M}(1-\eta_{mn})\delta_n.bw_{mn}.\log_2\left(1 + \frac{g_{mn}}{a_{mn}}.e_{mn}\right). \tag{11}$$

Here, in the above Equation (11), from the right side, the variables are as follows: e_{mn} is the transmission power with an instant channel gain of g_{mn}, where at the time instance i, the active channel between edge node n and network layer access module m, passes any random variable of data a_{mn} to access all t_n. η_{mn} is the average connective bit error rate having a selected bandwidth bw_{mn} between m and n in the range of $n = 1, 2, \cdots N$. The service module ensures bandwidth connectivity during data transfer with a rate of

$$r_n \geq \mathrm{Rate}_{\min}(n). \tag{12}$$

In the case of heterogeneous IoT network, the service module at the network layer furthermore helps to maximize the QoS by modifying the overall system capacity. Therefore, considering the energy consumption with data transfer quality, our module follows the following equation:

$$(\max\ (\mathrm{Rate}(a,e)) = \max\ \sum_{n}^{N}\sum_{m}^{M}(1-\eta_{mn}).\delta_n.a_{mn}.\log_2\left(1 + \frac{g_{mn}}{a_{mn}}.e_{mn}\right), \tag{13}$$

such that

$$a_{mn} \leq A_n \forall n \tag{14}$$

and

$$e_{mn} \leq E_n \forall n. \tag{15}$$

The above three equations apply the convex optimization method to be solved, where node n consumes the maximum energy E_n.

2.3.3. Decision Formation at the Application Layer. In the web service layer [2, 3], some service parameters help to create a utilization matrix which further assists the decision-making module more accurate in nature. The proposed model has considered the following parameters:

1. *Service completion time*$I(\beta)$: Mean consumed time by the network from the request submission to fulfil the requirement. In the case of a remote server subscription, the completion time is longer than a local server execution [14]. $E(I(\beta)) = (\pi \rho_k)/((1-\pi)^2 \varepsilon + 1/\mu)$ for $M/M/m$ queue model, using a service channel β. However, for a small amount IoT data analysis, it is time consuming to make it happen into a cloud server than an edge server because the queue of a cloud server is comparatively longer than any other local edge server [11].
2. *Service load*$L(\beta)$: Measured by $L(\beta) = \varepsilon\mu$, over the providing service rate μ, with arrival request-to-response rate ε.
3. *Service reputation*$Rp(\beta)$: Reputation is highly associated with market demand and quality feedbacks. The selection of service is proposed by the providers whereas decided by the user upon the edge-level trust factor.
4. *Service reliability*$\mathrm{Re}\,(\beta)$: It is measured by the risk of failure over edge/web level service upon demand. It is denoted as $\mathrm{Re}\,(\beta) = \sum_{n=1}^{N} (a_{mn}/n)$ and varies over D.
5. *Service subscription cost*$C(\beta)$: Generally, a fixed cost per use-based table is provided by the service provider or broker side. It ensures a role-based access control to allow only legitimate users to access valuable information.

In this paper, the prototype emphasizes the subsequent importance of data accuracy checking of collected data, coverage of the considered IoT network, energy efficiency of the connected network, and adaptability increment. Apart from the continuous quality monitoring of service matrices, the wireless network of IoT behaves exceptionally due to some factors. Our model considered the following aspects to deliver a quality experience to the subscribers: it makes the service provider module more accurate.

1. *Accuracy measurement*: Even if all the edge nodes are deployed at the same time due to damages or other functional issues, they are heterogeneous in nature. We consider three main accuracies called spatial [26], data, and sensing time accuracy of each node [27, 28]. To reduce information loss, the data are transmitted as metadata or packed within a packet while traveling the network layer.
2. *Consumed energy measurement*: In the case of IoT, energy consumption is inadequate due to the low battery capability of the edge nodes. Therefore, each node should be programmed to report about themselves if any damage has occurred to the neighbor node or the QoS manager.
3. *Coverage calculation*: Coverage calculation is also important to optimize data accuracy and energy consumption within a specific network range.

Ultimately, the service provider module considers more accurate information based on historical data and user directives. This accuracy depends upon the data loss and matching of subscription-level data with sensing-level data. Let us consider a service channel β between any edge nodes among $m \in M$ to layer-2 device ($n \in N$) having nN numbers of possible connectivity. For a randomly generated data variable, $a_{mn} \in A(\beta)$, the probability of incorrect data attainment is directly mapped to the reliability factor. Therefore, the probable information accuracy experienced by the user is

$$p(\theta) = 1 - p\left(a'_{mn}\right), \tag{16}$$

where $p(a'_{mn})$ reflects data loss during network traffic management and exponential path-loss fronting. Therefore,

$$p(\theta) = 1 - p\left(1 - \sum_{n=1}^{N} \frac{a_{mn}}{n} .s_n\right). \tag{17}$$

Ultimately, the resultant value supports to generating decision over a segment connectivity state s_n which has been mentioned previously.

3. Result Outcomes

An emulation scenario of weather data prediction with the help of edge devices and cloud server in the IEM Centre of Excellence for Cloud Computing and IoT (C2IoT), Department of CSE (AIML), Institute of Engineering and Management, India, having a latitude and longitude of 22.57 and 88.43, respectively, has been deployed. The emulator is specifically based on Java programming which can be manually modeled by using the amazing graphical user interface. The GUI consists of actuators, sensors, and instances, along with the connectors. The sensors are acting as IoT devices that are present in a connected mess network, subscribed by more than one user. The service of data transfer from one IoT layer to another happens as per the recommendation of

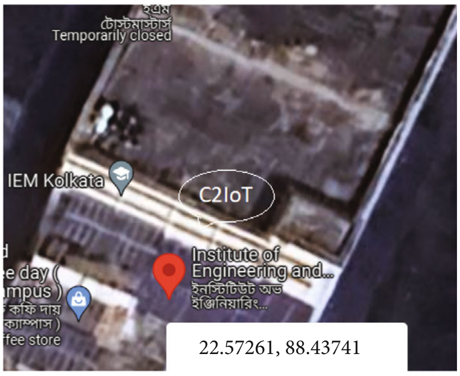

Figure 2: Sensor data collection centre: cloud server in IEM Centre of Excellence for Cloud Computing and IoT (C2IoT), Department of CSE (AIML), Institute of Engineering and Management, India.

the QoS manager [9, 29]. Therefore, we have arranged the overall network connectivity as per [4, 10] and [6], which articles offer general cloud computing, fog computing, and edge computing prototypes, in the absence of QoS metrics analyzer, respectively. To measure updated data of the target region, shown in Figure 2, an IoT network has been designed, and the collected data has been gathered into C2IoT. Identical data from the sensor devices directly goes for further analysis through four different gateways to the four different data analysis platforms placed into Amazon public cloud server, private fog server, without QoS aware edge server, and proposed QoS aware edge IoT server. Data analysis has done through some relevant steps shown in Figures 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, and 15 also as follows:

1. Data and relations
2. Data processing
3. Data visualization
4. Correlation
5. Regression
6. Forecasting
7. Classification
8. Clustering

Figure 2 is the satellite position sensor data collection centre: cloud server in IEM Centre of Excellence for Cloud Computing and IoT (C2IoT), Department of CSE (AIML), Institute of Engineering and Management, India, having the latitude and longitude of 22.57 and 88.43, respectively.

Figure 3 shows real-time sensor data plotting of temperature, dew point, and humidity altogether, through deployed sensor nodes, whereas real-time temperature data collection through deployed sensor nodes ([TMP36], omega) has been shown in Figure 4, and real-time humidity data collection through deployed sensor nodes ([LM335Z/NOPB], Texas) has been shown in Figure 8 individually. A histogram of collected temperature data at C2IoT against Figure 4 has been shown in Figure 5. Similarly, 3 consecutive days of collected temperature data plotting has been shown in Figure 6, and the histogram plotting has been done in Figure 7. The correlation between collected temperature data and humidity data has been plotted in Figure 9. We compared our proposed

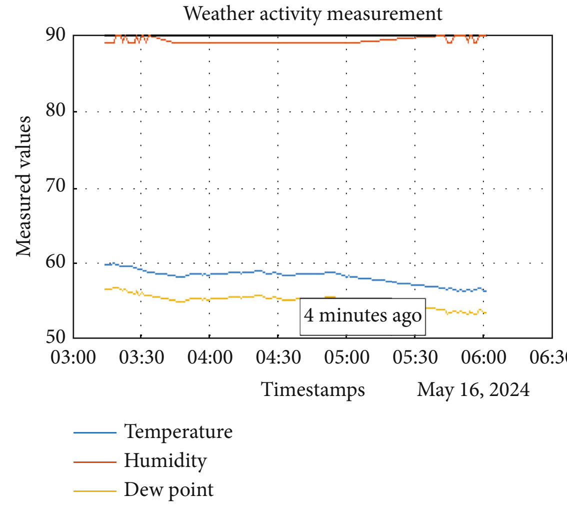

Figure 3: Real-time sensor data collection through deployed sensor nodes.

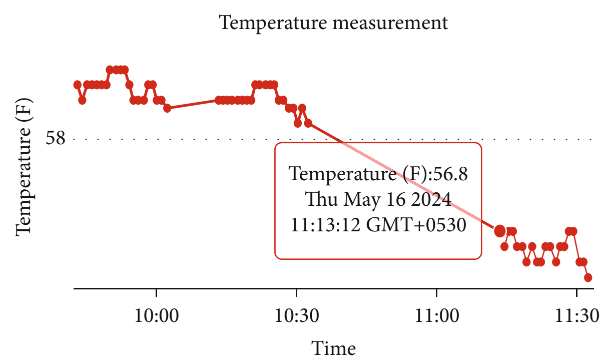

Figure 4: Temperature data collection by deployed sensor node at C2IoT.

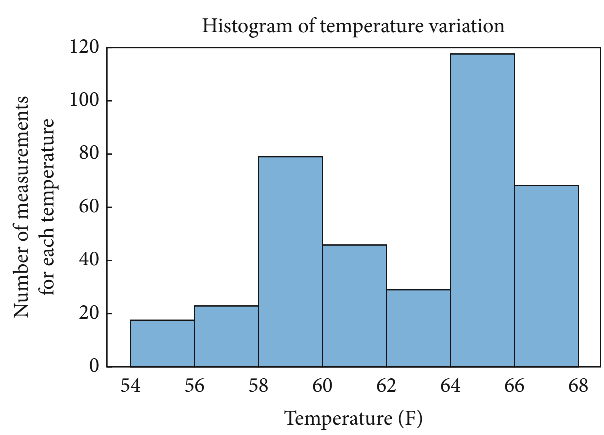

Figure 5: Histogram of collected temperature data at C2IoT.

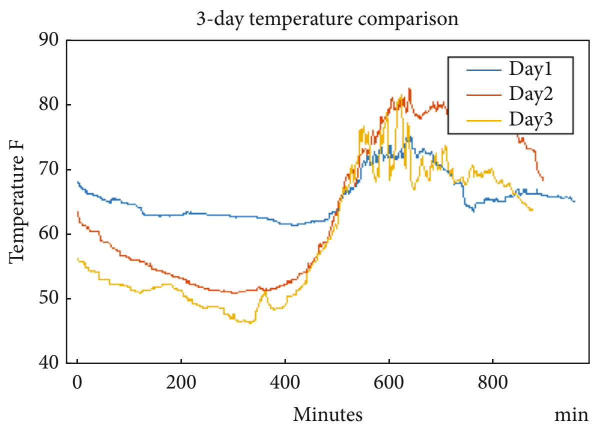

Figure 6: Three days consecutive temperature data collection by deployed sensor node for C2IoT.

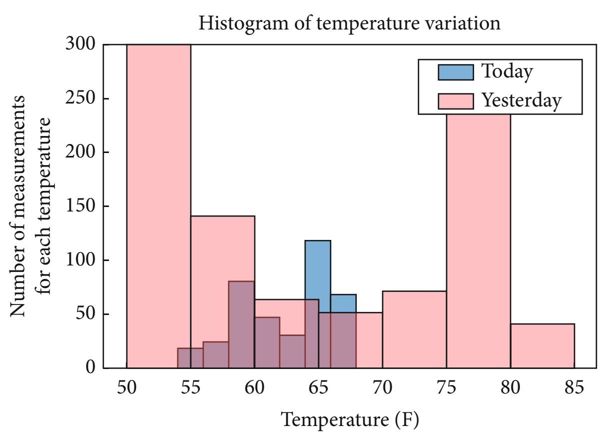

Figure 7: Histogram of collected 3 days temperature data into C2IoT.

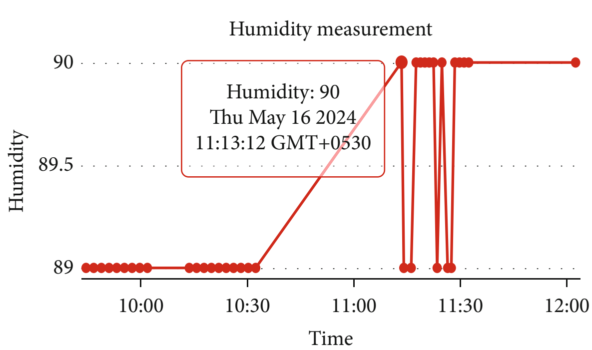

Figure 8: Deployed sensor node's collected temperature data and humidity data.

prototype (QoS-aware edge IoT) with the other three, considering models concerning delay, energy consumption, and alive node count. The resultant values with graphs indicate better outputs in terms of low latency, less power consumption, accurate decision-making, and higher lifetime of the sensor nodes using our proposed solution. The mapping of the procedural outcome of the proposed methodology as

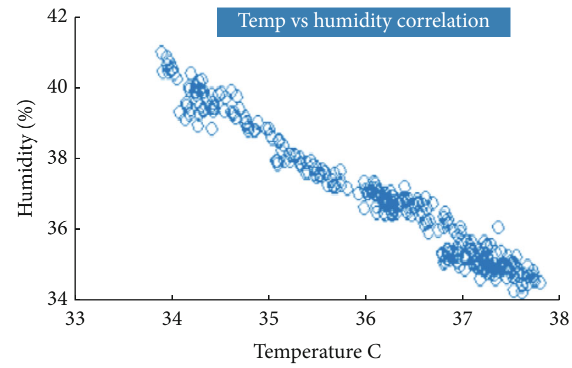

Figure 9: Correlation between collected temperature data and humidity data.

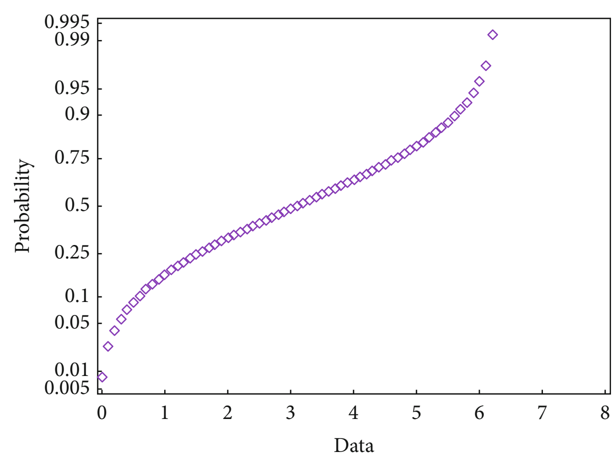

Figure 10: Probability matrix measurement for the upcoming temperature data.

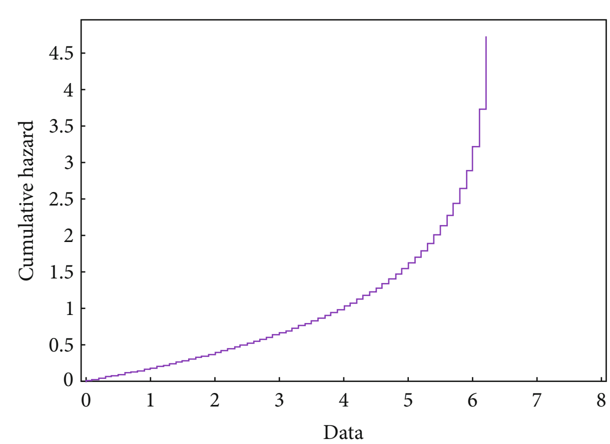

Figure 11: Possible hazard calculation for the future temperature data collection.

per Figure 1 and the step-by-step implementation of Section 2 are shown in the result outcome section with examples in the following fashion:

We adopted experimental and subsequent observation methodology into our proposed data scope analysis and

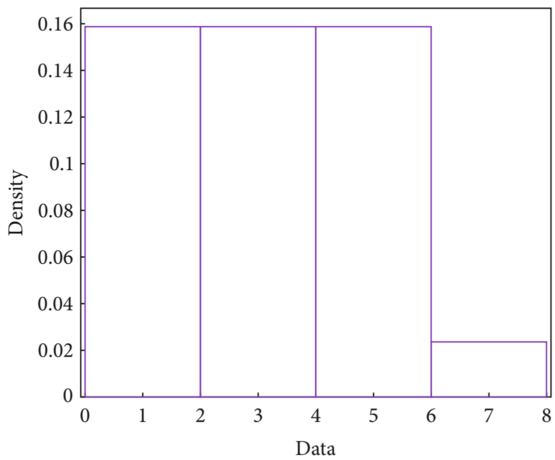

Figure 12: Density of the collected temperature data at C2IoT.

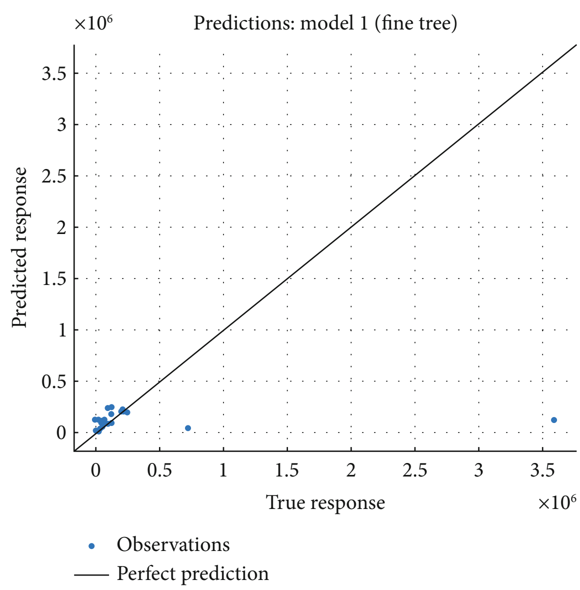

Figure 13: Weather data prediction through data analysis into C2IoT.

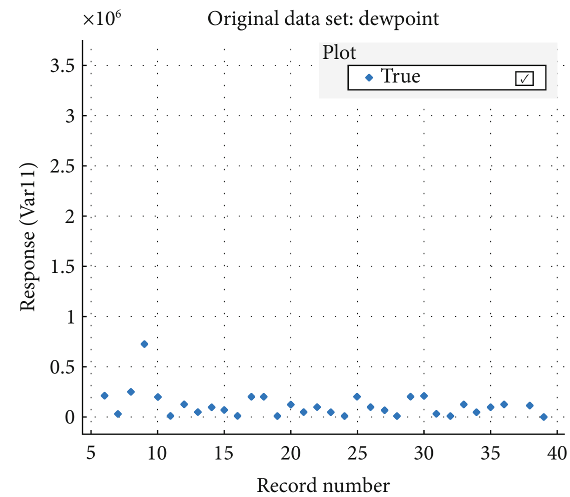

Figure 14: Real-time dew point measurement by the deployed sensor node at C2IoT.

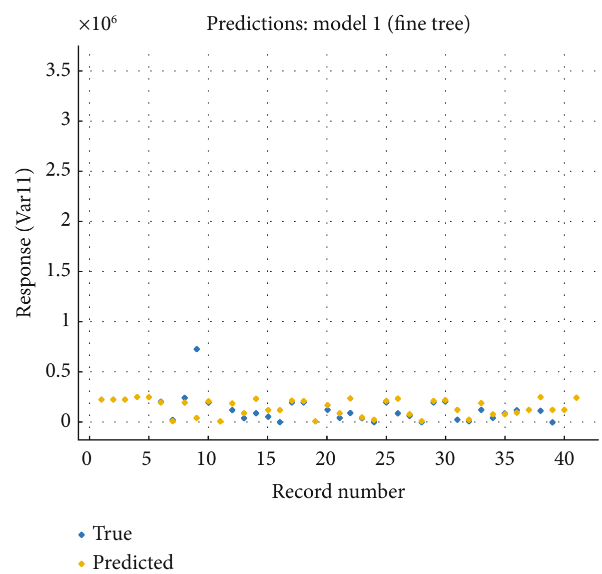

Figure 15: Prediction of dew point measurement by the prediction model (fine tree).

process module. We have done a tuning check of the collected sensor data using conventional methods like histograms. It helped identify whether there is any uncertain hardware or connection error occurrence during sensor data collection. Figure 5 shows the histogram of temperature data collection against Figure 4, where temperature (F) is the independent value collected concerning the continuous time factor. We found our data collection module works satisfactorily as no significant gap is in the range of the Figure 5 graph. We plotted consecutively collected 3 days' sensor data to justify further visualization accuracy, as shown in Figure 6. The histogram plotting against Figure 6 is illustrated in Figure 7. The continuous temperature range in Figure 7 likewise confirms that the visualization of collected sample data is composed of a correct frequency distribution. As per the principal aim of our scheme, to provide quality experienced-aware service selection, we chose quantitative data collection following predictive data analysis. We used the proposed edge-service selection model to consider different category data like temperature, dew point, and humidity to predict upcoming weather conditions. The prediction module merges individual sensor data collected into the edge IoT server. Further, compute the correlation between corresponding data. Figure 9 shows the correlation graph between temperature and humidity data from Figure 8.

Next, the quality-assurance module ensures the experienced-based service selection using three-step decision-making via QoS aware algorithm. The decision-making unit makes quality-aware service selections depending on three major factors: service availability, possible service failure due to data hazard, and promised service providence. The

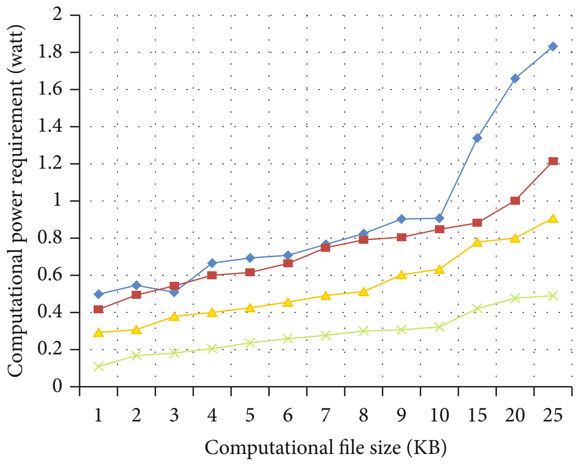

FIGURE 16: Computational power comparisons of the proposed method with existing methods.

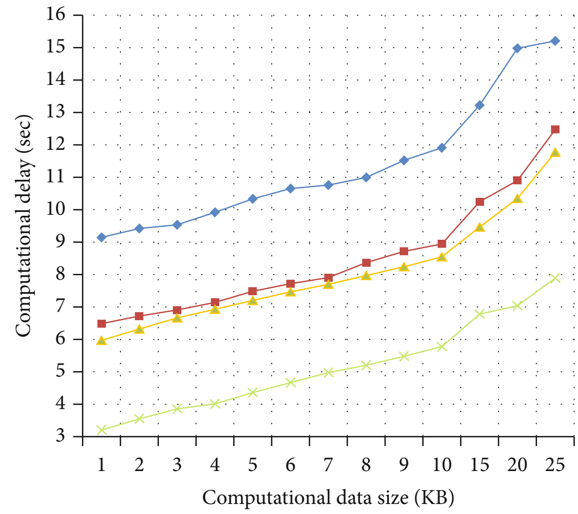

FIGURE 17: Computational delay comparisons of the proposed method with existing methods.

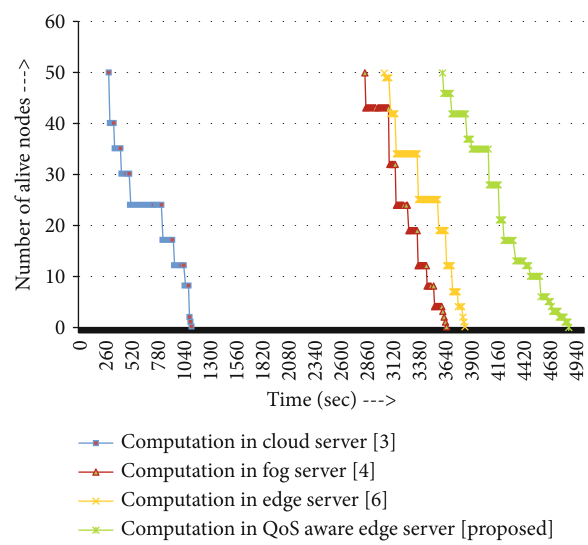

FIGURE 18: Lifetime of edge nodes (sensor nodes) of the proposed method with existing methods.

mathematical backbone of the module uses the previously mentioned equations to measure these three factors.

Here, the module considered data density as service availability and plotted the graph in Figure 12 with the help of Equation (12). Similarly, possible service failure due to data hazards on temperature data collection has been plotted with the help of Equation (16), as shown in Figure 11. The third variable, promised service providence, has been calculated with the help of Equation (17) and the plotted graph as a probability matrix measurement, as shown in Figure 10. Furthermore, the fine tree inspired our proposed analysis model, which publishes the predicted results, as demonstrated for temperature data in Figure 13. Following similar steps, the expected dew point measurement is shown in Figure 15 concerning the collected real-time measurement dew points, shown in Figure 14. This experienced-aware service selection mechanism furthermore achieved time- and energy-efficient service providence until data has been published from data collection.

Here, the module considered data density as service availability and plotted the graph in Figure 12 with the help of Equation (12). Similarly, possible service failure due to data hazards on temperature data collection has been plotted with the help of Equation (16), as shown in Figure 11. The third variable, promised service providence, has been calculated with the help of Equation (17) and the plotted graph as a probability matrix measurement, as shown in Figure 10. Furthermore, the fine tree inspired our proposed analysis model, which publishes the predicted results, as demonstrated for temperature data in Figure 13. Following similar steps, the expected dew point measurement is shown in Figure 15 concerning the collected real-time measurement dew points, as shown in Figure 14. This experienced-aware service selection mechanism furthermore achieved time- and energy-efficient service providence until data has been published from data collection.

The entire procedure shows how our proposed solution has been designed to analyze heterogeneous QoS demand at the application layer, network traffic into the network layer, and reliable factors at the sensing layer. Comparative analysis graphs are shown in Figures 16, 17, and 18. Table 2 briefly reveals the emulation setup of the designed test bed in the centre of cloud computing and IoT, India. The comparison graph in Figure 16 shows computational power requirements for all four methods ([3, 4], and [6]), including our proposed model, while delivering various amounts of data packets. Figure 17 shows time comparisons for mentioned all four methods while delivering heterogeneous data packets from various edge nodes. Figure 18 shows the lifetime of 40 nodes applying the four discussed methods separately. Comparative discussion in Table 3 has briefly revealed the contribution and achievement of our proposed method concerning other existing related works in the field of sensor networks. The lifetime of the IoT nodes

TABLE 2: Emulation setup.

Equipment	Description
Temperature sensor	[TMP36], omega. (18 pcs.)
Humidity sensor	[LM335Z/NOPB], Texas, (22 pcs.)
Fog instance	Intel Xeon CPUES-26670 @2.60 GHz (Hexa Core), 16 GB, 2 TB
Edge instance	Intel Xeon CPU ES-2667 0 @2.60 GHz (Hexa Core), 8 GB, 1 TB
Public cloud	Amazon cloud server (AWS)
Receiver node/application end	Lenovo Ideapad 500, Intel Core I7, 6th Gen @2.5 GHz-Dual Core, 2 GB, 1 TB (4 pcs.)
Implementation language	Java using NetBeans 8.0.2

TABLE 3: Comparative study between some correlated works.

Properties	Cloud computing–based IoT network [3]	Fog computing–based IoT network [4]	Edge computing–based IoT network [6]	Proposed QoS-aware edge IoT
Contributions	A cloud computing based IoT model has designed for smart agriculture	A survey of IoT protocols, technologies, and applications has done	Extended mobile-edge computing model for IoT network	QoS-aware edge IoT computing for service-oriented IoT users
Operational speed	Slow	Medium	High	High
Delay and jitter	High	Medium	Less	Less
Energy efficient	Less	Medium	Medium	High
QoS metric dependency	×	×	×	✓
Efficient resource allocation	×	✓	×	✓
Scalability	×	✓	✓	✓
Packet delivery delay	21% more than proposed	17% more than proposed	9% more than proposed	NA
Energy consumption	20% more than proposed	19% more than proposed	15% more than proposed	NA

has significantly increased due to less power consumption of the proposed method. Along with the novelties, discussed in Table 3, it is also significant that the projected model is capable to deliver sensor data 21%, 17%, and 9% faster to the subscriber than [4, 11] and [6], respectively. It is also revealed that the proposed architecture is 20%, 19%, and 15% more efficient in terms of energy concerning [3, 4] and [6].

4. Conclusion

Extensive IoT networks like smart agriculture, smart education, and smart healthcare systems produce huge amounts of IoT evidence [12, 13, 17–19]. Due to increasing simultaneous connectivity among the IoT nodes, one must choose quick, responsive, and cost-effective data analysis through a cloud server to make the entire procedure convenient. Regarding enormous data control, appropriate data analysis and storage platforms offered by cloud servers can significantly decrease the time for necessary data search, delay, energy, and cost. However, in the case of local and regional data analysis of IoT nodes, a QoE-aware edge server module is needed, which can dedicatedly involve making the whole IoT network more available to the users in an efficient way [8, 9]. Edge computing for IoT networks is ideal for small data computations [2, 17, 18] with less risk of global attacks [13]. Data analysis of previous user demand, prediction of future data utilization, and constant monitoring of designed IoT networks can make the scenario competent and resourceful. Therefore, the proposed model is certainly helpful for efficient data analysis into edge servers for local IoT data analysis. The proposed prototype can be revised in the near imminent with the following prospects of the current edge network to overcome the limitations of our proposed work:

A. To make the IoT service more convenient in terms of essential service subscription cost, which should be decreased by giving the functioning effort on availability factors;

B. Service quality amplification of IoT nodes should be emphasized for robustness increment, regularity maintenance, and the growth of the reputation of the entire edge IoT network.

C. Data annotation can further be modified in two manners: automatic and manual. The autoprocess should be accomplished at an affordable price.

D. To overcome one of the significant shortcomings, security issues while using an IOT network. Both the software- and hardware-related issues should be individually identified and avoided [21–23].

With an incessant and collaborative effort in research and development areas, we can be optimistic about overcoming these complications shortly. With the help of incorporating other effective solutions [30, 31], the result will be an effort towards unquestionably enlarging the acceptance of IoT networks in every aspect of human life, where a waiting time estimation system based on QoE-aware multi-source data is needed.

Data Availability Statement

The data are available from the corresponding authors on reasonable request. The original contributions presented in the study are included in the article material, and further inquiries can be directed to the corresponding authors.

Disclosure

No third-party individuals or external services were involved in the research or preparation of the manuscript.

Conflicts of Interest

The authors declare no conflicts of interest.

Author Contributions

We confirm that all authors have contributed significantly to the manuscript preparation and research activities. Idea generation: Bipasha Guha Roy and Deepsubhra Guha Roy. Wrote the main manuscript: Bipasha Guha Roy and Deepsubhra Guha Roy. Prepared figures: Bipasha Guha Roy and Piyali Datta. Methodology: Bipasha Guha Roy and Piyali Datta. Comparative analysis: Surbhi Bhatia Khan and Deepsubhra Guha Roy. Overall organization: Deepsubhra Guha Roy, Abdullah Albuali, and Ahlam Almusharraf. Editing and formatting: Deepsubhra Guha Roy, Abdullah Albuali, and Ahlam Almusharraf. Supervision: Deepsubhra Guha Roy and Surbhi Bhatia Khan.

Funding

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (Grant No. KFU251757). This work was supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project (Number PNURSP2025R432), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. We would like to express our heartfelt gratitude to the IEM Centre of Excellence for Cloud Computing and IoT for invaluable support and resources from the IEM Grant in Aid Project IEMT(S)/2024/02-G19 and IEMT(S)/2023/02-G05.

Acknowledgments

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (Grant No. KFU251757). This work was supported by the Princess Nourah bint Abdulrahman University Researchers Supporting Project (Number PNURSP2025R432), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. We would like to express our heartfelt gratitude to the IEM Centre of Excellence for Cloud Computing and IoT for invaluable support and resources from the IEM Grant in Aid project IEMT(S)/2024/02-G19 and IEMT(S)/2023/02-G05. No AI software was used at any stage of manuscript preparation, data collection, writing, or editing.

References

[1] B. Kar, W. Yahya, Y. D. Lin, and A. Ali, "Offloading Using Traditional Optimization and Machine Learning in Federated Cloud–Edge–Fog Systems: A Survey," *IEEE Communications Surveys and Tutorials* 25, no. 2 (2023): 1199–1226, https://doi.org/10.1109/COMST.2023.3239579.

[2] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, and D. O. Wu, "Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges," *IEEE Communications Surveys and Tutorials* 22, no. 4 (2020): 2462–2488, https://doi.org/10.1109/COMST.2020.3009103.

[3] S. Hamdan, M. Ayyash, and S. Almajali, "Edge-Computing Architectures for Internet of Things Applications: A Survey," *Sensors* 20, no. 22 (2020): 6441, https://doi.org/10.3390/s20226441.

[4] H. Sabireen and V. Neelanarayanan, "A Review on Fog Computing: Architecture, Fog With IoT, Algorithms and Research Challenges," *ICT Express* 7, no. 2 (2021): 162–176, https://doi.org/10.1016/j.icte.2021.05.004.

[5] U. K. Saba, S. ul Islam, H. Ijaz, J. J. P. C. Rodrigues, A. Gani, and K. Munir, "Planning Fog Networks for Time-Critical IoT Requests," *Computer Communications* 172 (2021): 75–83, https://doi.org/10.1016/j.comcom.2021.03.002.

[6] F. Murtaza, A. Akhunzada, S. ul Islam, J. Boudjadar, and R. Buyya, "QoS-Aware Service Provisioning in Fog Computing," *Journal of Network and Computer Applications* 165 (2020): 102674, https://doi.org/10.1016/j.jnca.2020.102674.

[7] D. Zhao, Q. Zou, and M. Boshkani Zadeh, "A QoS-Aware IoT Service Placement Mechanism in Fog Computing Based on Open-Source Development Model," *Journal of Grid Computing* 20, no. 2 (2022): 12, https://doi.org/10.1007/s10723-022-09604-3.

[8] R. Wang and J. Lu, "QoS-Aware Service Discovery and Selection Management for Cloud-Edge Computing Using a Hybrid Meta-Heuristic Algorithm in IoT," *Wireless Personal Communications* 126, no. 3 (2022): 2269–2282, https://doi.org/10.1007/s11277-021-09052-4.

[9] D. Guha Roy, B. Mahato, D. De, and S. N. Srirama, "Quality of Experience Aware Recommendation System With IoT Data Filtration for Handshaking Among Users Using MQTT-SN Protocol," *Journal of Ambient Intelligence and Humanized Computing* 14, no. 7 (2023): 8811–8826, https://doi.org/10.1007/s12652-021-03644-5.

[10] D. Guha Roy, B. Mahato, A. Ghosh, and D. De, "Service Aware Resource Management Into Cloudlets for Data Offloading Towards IoT," *Microsystem Technologies* 28, no. 2 (2022): 517–531, https://doi.org/10.1007/s00542-019-04450-y.

[11] B. Mahato, D. Guha Roy, and D. De, "Distributed Bandwidth Selection Approach for Cooperative Peer to Peer Multi-Cloud Platform," *Peer-to-Peer Networking and Applications* 14, no. 1 (2021): 177–201, https://doi.org/10.1007/s12083-020-00917-2.

[12] K. Cao, Y. Liu, G. Meng, and Q. Sun, "An Overview on Edge Computing Research," *IEEE Access* 8 (2020): 85714–85728, https://doi.org/10.1109/ACCESS.2020.2991734.

[13] S. Bhatia, R. I. Alsuwailam, D. G. Roy, and A. Mashat, "Improved Multimedia Object Processing for the Internet of Vehicles," *Sensors* 22, no. 11 (2022): 4133, https://doi.org/10.3390/s22114133.

[14] G. Carofiglio, G. Grassi, E. Loparco, L. Muscariello, M. Papalini, and J. Samain, "Characterizing the Relationship Between Application QoE and Network QoS for Real-Time Services," in *Proceedings of the ACM SIGCOMM 2021 Workshop on Network-Application Integration* (ACM, 2021), 20–25, https://doi.org/10.1145/3472727.3472800.

[15] P. Lai, Q. He, G. Cui, et al., "QoE-Aware User Allocation in Edge Computing Systems With Dynamic QoS," *Future Generation Computer Systems* 112 (2020): 684–694, https://doi.org/10.1016/j.future.2020.06.029.

[16] N. Abu-Amssimir and A. Al-Haj, "A QoS-Aware Resource Management Scheme Over Fog Computing Infrastructures in IoT Systems," *Multimedia Tools and Applications* 82, no. 18 (2023): 28281–28300, https://doi.org/10.1007/s11042-023-14856-6.

[17] S. S. Vellela, V. L. Reddy, D. Roja, G. R. Rao, K. B. Sk, and K. K. Kumar, "A Cloud-Based Smart IoT Platform for Personalized Healthcare Data Gathering and Monitoring System," in *Proceedings of the 2023 3rd Asian Conference on Innovation in Technology (ASIANCON)* (IEEE, 2023), 1–5, https://doi.org/10.1109/ASIANCON58793.2023.10270407.

[18] M. A. Zamir and R. M. Sonar, "Application of Internet of Things (IoT) in Agriculture: A Review," in *Proceedings of the 2023 8th International Conference on Communication and Electronics Systems (ICCES)* (IEEE, 2023), 425–431, https://doi.org/10.1109/ICCES57224.2023.10192761.

[19] L. Lin, J. Liu, N. Huang, S. Li, and Y. Zhang, "Multiscale Spatio-Temporal Feature Fusion Based Non-Intrusive Appliance Load Monitoring for Multiple Industrial Industries," *Applied Soft Computing* 167 (2024): 112445, https://doi.org/10.1016/j.asoc.2024.112445.

[20] G. Sun, Y. Wang, H. Yu, and M. Guizani, "Proportional Fairness-Aware Task Scheduling in Space–Air–Ground Integrated Networks," *IEEE Transactions on Services Computing* 17, no. 6 (2024): 4125–4137, https://doi.org/10.1109/TSC.2024.3478730.

[21] Z. Liu, L. Wan, J. Guo, et al., "AuthUp: A Privacy-Preserving V2I Authentication Scheme for Vehicular Ad Hoc Networks," *IEEE Transactions on Vehicular Technology* 74, no. 2 (2025): 1–15, https://doi.org/10.1109/TVT.2025.3567603.

[22] Z. Liu, J. Weng, J. Ma, et al., "TCEMD: A Trust Cascading-Based Emergency Message Dissemination Model in VANETs," *IEEE Internet of Things Journal* 7, no. 5 (2020): 4028–4048, https://doi.org/10.1109/JIOT.2019.2957520.

[23] J. Guo, Z. Liu, S. Tian, et al., "TFL-DT: A Trust Evaluation Scheme for Federated Learning in Digital Twin for Mobile Networks," *IEEE Journal on Selected Areas in Communications* 41, no. 11 (2023): 3548–3560, https://doi.org/10.1109/JSAC.2023.3310094.

[24] M. Babar, M. A. Jan, X. He, M. U. Tariq, S. Mastorakis, and R. Alturki, "An Optimized IoT-Enabled Big Data Analytics Architecture for Edge–Cloud Computing," *IEEE Internet of Things Journal* 10, no. 5 (2023): 3995–4005, https://doi.org/10.1109/JIOT.2022.3157552.

[25] V. Hayyolalam, M. Aloqaily, O. Ozkasap, and M. Guizani, "Edge Intelligence for Empowering IoT-Based Healthcare Systems," *IEEE Wireless Communications* 28, no. 3 (2021): 6–14, https://doi.org/10.1109/MWC.001.2000345.

[26] H. Li, K. Qiu, L. Chen, X. Mei, L. Hong, and C. Tao, "SCAttNet: Semantic Segmentation Network With Spatial and Channel Attention Mechanism for High-Resolution Remote Sensing Images," *IEEE Geoscience and Remote Sensing Letters* 18, no. 5 (2021): 905–909, https://doi.org/10.1109/LGRS.2020.2988294.

[27] Y. Yang, X. Yang, M. Heidari, et al., "ASTREAM: Data-Stream-Driven Scalable Anomaly Detection With Accuracy Guarantee in IIoT Environment," *IEEE Transactions on Network Science and Engineering* 10, no. 5 (2023): 3007–3016, https://doi.org/10.1109/TNSE.2022.3157730.

[28] S. Bilakeri and A. K. Karunakar, "Multi-Object Tracking by Multi-Feature Fusion to Associate All Detected Boxes," *Cogent Engineering* 9, no. 1 (2022): 2151553, https://doi.org/10.1080/23311916.2022.2151553.

[29] H. Chen, Y. Bei, W. Huang, S. Chen, F. Huang, and X. Huang, "Graph Cross-Correlated Network for Recommendation," *IEEE Transactions on Knowledge and Data Engineering* 37, no. 2 (2025): 710–723, https://doi.org/10.1109/TKDE.2024.3491778.

[30] Y. Rong, Z. Xu, J. Liu, et al., "Du-Bus: A Realtime Bus Waiting Time Estimation System Based on Multi-Source Data," *IEEE Transactions on Intelligent Transportation Systems* 23, no. 12 (2022): 24524–24539, https://doi.org/10.1109/TITS.2022.3210170.

[31] D. Zhu, Z. Han, X. Du, D. Zuo, L. Cai, and C. Xue, "Hybrid Model Integrating Fuzzy Systems and Convolutional Factorization Machine for Delivery Time Prediction in Intelligent Logistics," *IEEE Transactions on Fuzzy Systems* 33, no. 1 (2025): 406–417, https://doi.org/10.1109/TFUZZ.2024.3472043.