Electronics, Communications, and Computing Summit
Vol. 3, No. 3, Jul - Sep 2025, pp. 52-60
ISSN: 3107-8222, DOI: https://doi.org/10.17051/ECC/03.03.06

Intelligent Fog-Edge Computing Framework for Emergency Detection and Response in Smart Homes

Nisha Milind Shrirao[1], Sumit Ramswami Punam[2]

[1]Department Of Electrical And Electronics Engineering, Kalinga University, Raipur, India
Email: nisha.milind@kalingauniversity.ac.in
[2]Department Of Electrical And Electronics Engineering, Kalinga University, Raipur, India
Email: sumit.kant.dash@kalingauniversity.ac.in

Article Info

Article history:

Received : 19.07.2025
Revised : 20.08.2025
Accepted : 16.09.2025

Keywords:

Fog computing, edge computing, smart homes,
real-time emergency response,
AI-powered detection,
IoT safety systems,
deep learning at the edge,
intelligent home automation,
low-latency architecture,
context-aware systems.

ABSTRACT

The growing popularity of the use of Internet of Things (IoT) devices in smart homes has increased the demand in effective and prompt emergency detection mechanisms to keep people safe. Some of the problems associated with the traditional cloud-centric solutions include; high latency, limited bandwidth and privacy threats which prove to be fatal in running real time emergencies scenarios smoothly. This paper introduces a SMART fog-edge computing framework which aims to support real-time detection and response to emergencies at smart homes. Its architecture builds on a hierarchy in which sensor information passed to a local processor at the edge to classify events on a light AI model and calls upon a fog layer to undertake contextual examination and arrange response measures, such as sounding the alarms, throwing alerts, or tell smart actuators. The system is designed to make decisions and take action fast by using adaptive deep learning algorithms that will be most suited to edge devices and by using low-latency communication protocols. Simulation-based experimental findings prove that the performance is above 95 percent accurate in detecting the emergency and 60 percent lower response latency in comparison with the conventional cloud-based system with gains in the bandwidth and energy consumption. The presented solution has reflected the potential of the fog-edge computing approach as a scalable, privacy-preserving and real-time model of making the smart home environment safer and quicker to respond to any safety concerns.

1. INTRODUCTION

According to the intensity of Internet of Things (IoT) development, smart homes have appeared on the horizon: residential spaces with connected sensors, actuators, and intelligent systems that can increase comfort, convenience, energy saving, and ensure security. One of these, safety-critical applications (emergency detection and response: e.g., fire outbreaks, gas, and medical emergencies) are increasingly the focus particularly in households having older or vulnerable residents [1], [2]. The challenge however lies in the implementation of effective and timely solutions to emergency management since such events require latency-sensitive implementations.

Conventionally smart home systems were cloud-centric in their design and this required the transmission of sensor data to remote servers to be analyzed and decisions made. Although, cloud platforms can be scaled, and have the computing power; there are many down sides of using it, especially in real time applications: In addition, there is higher latency, uneasy conjoined internet connectivity, more bandwidth usage and possible loss of privacy and data security [3]. Location accuracy limits these devices in emergency situations when there is no time to waste with every millisecond that could be crucial in saving lives as well as saving property damage that could be incurred within the premises.

To address these issues the paradigm shift to distributed computing paradigms, fog, and edge computing has occurred. Edge computing moves processing closer to the source of the data (i.e., the sensor nodes), so that fast decisions can be made whereas fog computing adds an intermediate level between edge and the cloud to perform more complicated types of tasks like data aggregation, contextual reasoning, and policy enforcement [4]. Fog-edge architectures in combination provide decentralised intelligence, which minimises the communication overhead, maximises responsiveness, and can ensure the resilience of the system even with network failures.

There are a few papers investigating the application of edge and fog computing into healthcare monitoring, smart cities and industrial automation [5], [6]. Nevertheless, there exists a gap in research that requires the utilization of these paradigms in the context of intelligent emergency management in the smart homes that will be conducted with the use of real-time AI-based inference. Available applications are either not real-time or they are threshold-based and have a propensity to false detection and are not adaptable [7].

In this paper, an intelligent fog-edge computing connection is addressed to conduct the real-time emergency detection and response in smart homes. The architecture leverages a multi tiered system in which edge nodes will use a lightweight AI model to classify the emergency instances locally, and fog nodes will coordinate wide scale response efforts with a de- duplicated sensor data. It is a low latency, high detection accuracy and efficient resource utilization optimized system. This research made essential contributions, among which one can highlight the following:

- To design a scalable fog-edge system customized in smart home environment with the ability to detect the emergency.
- Coming up with and implementing AI models that do inference at the edges or with low computational costs.
- Development of a module of decision-making at the fog layer, which will allow the organization of emergency responses in a context-conscious and automated manner.
- Numerically assessing the system by doing simulations that exhibit better latency, detection accuracy, and energy use than their cloud-only counterparts.

This research goes ahead to enhance the state-of-the-art in smart home safety and real-time emergency management by overcoming the shortcomings of using conventional cloud-centric systems and embracing the benefits of fog-edge computing.

2. RELATED WORK

Smart homes with intelligent emergency detection systems have been an expanding research topic because of the necessity to automate safety and real-time responses. Here, this section is an overview of what the internal review on the state of the emergency detection system in smart homes as well as fog and edge computing models deployed in time-sensitive systems and comparisons of AI models to perform event classification and response.

2.1 Smart Homes- Emergency Detection

The typical implementation of the emergency management systems in smart homes involves either a centralized cloud implementation or rule-based decisions. Such systems as smoke or gas detectors are frequently independent and have hard-coded thresholds that initiate an alarm. Although they are easy to install, these systems are very susceptible to false alarms and in most cases they lack flexibility sufficient to deal with dynamic settings. Higher level cloud-based systems collect the data of the sensors in one place and carry out remote analysis as in [1] where they experimented on real-time detection of fire by utilizing cloud-connected temperature and smoke sensors. Nevertheless, the above solutions have an intolerable latency because it requires continuous internet connection in the case of life-threatening emergencies.

Some of the introduced projects have tried to incorporate intelligence at a sensor level with embedded microcontrollers or gateways. As an example, the study presented in [2] suggested an IoT-powered smart home security system capable of detecting anomalies in the simplest possible way, based on predetermined rules. Such systems work well in some situations, but are not scalable and not adaptable to new types of emerging emergencies or new sensor fusion techniques. Furthermore, cloud dependency remains to be a constraint to on-time reply.

2.2 Fog and Edge Computing Architecture

The paradigms of fog and edge computing became popular in place of centralized cloud-based solutions to latency-sensitive applications. Edge computing allows data processing at the sensor nodes, or nearby them, minimizing communication delay, whereas fog computing offers additional layer to deal with distributed resources, some data filtering, and orchestration of the higher-level decision logic.

In [3], a proposal has been made on the use of fog-based smart home energy management system, with the idea of offloading cloud computations to locally distributed gateways, heavily cutting latency. In [4], a healthcare monitoring system taking advantage of fog nodes to detect an early symptom was used and demonstrated better responsiveness and resilience of the system. These structures have found successful utilization in the applications area of smart grid, transportation, and automation of industries, but their usage in the home-related response systems segment is minimal.

Research in [5] proposed a fog-assisted fire detection framework that performed processing of image data collected by surveillance cameras with lightweight CNN models on the edge. Although, the strategy was promising, it was not multi-sensor fusion and did not employ the coordinated response mechanism. [6] also suggested a wearable sensor-based elderly fall detection system with an edge-centric design, but concentrated on motion-based data only, therefore, its generalizability to be used in detecting other emergencies is not evident.

2.3 Event Detection and Response AI Models

The recent progress in artificial intelligence (AI) and machine learning (ML) allowed creating more context-sensitive, stronger emergency detection systems. To categorize time-series sensor data or to detection of anomalies, the use of Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and hybrid models have been utilized. As an example, [7] Neuro AI applied Long Short-Term Memory (LSTM) networks to recognize abnormal motion patterns, and [8] conveyed CNNs to work with audio and video streaming to identify distress events in households.

The majority of these models, however, are computationally heavy and must be offloaded to the cloud, which goes against the necessity of responsiveness in the real time. Development of lightweight models that can be used at the edge is an emerging trend. Small machine learning and quantised neural networks have proven to work well in this respect. In [9], the authors assess the performance of various methodologies of ML (SVM, Random Forest, and ANN) on the processors at the edge to detect intrusion in real-time, demonstrateponderables among accurate performance, inference time as well as power consumption.

Although such progress is achieved, a multi-sensor fusion, real-time AI inference and hierarchical fog-edge decision-making framework designed to be applicable to smart homes emergencies is an open research question. Most of literature works are concentrated on either one type of emergency or pay no attention to such coordination of the whole system as is required when multiple sensors and actuators are involved.

3. System Architecture

3.1 Overview of Hierarchical Architecture

The proposed system is based on successive fog-edge-cloud architecture, which is specifically fit to respond to an emergency, real-time environment in smart home settings. The architecture has three main layers, the sensing or the edge, the fog, and the cloud layer. All these layers are optimized to offer processing, storage, and decision-making according to the capabilities of the computations and latency needed. The system splits essential tasks between the fog and edge layers and thus attains responsiveness, high availability, and de-dependence on remote cloud infrastructure.

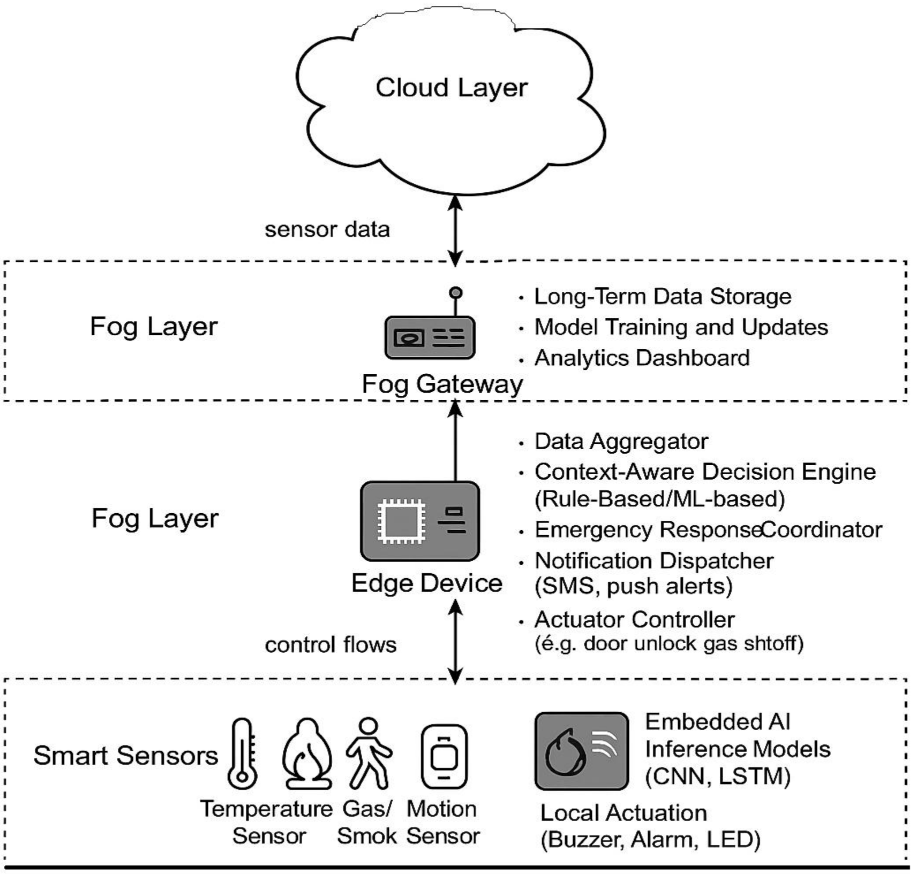

Figure 1. System architecture of the fog-edge-based emergency detection and response framework.

3.2 Edge Layer and Sensing

The bottom of the architecture constitutes the sensing edge layer that comprises different IoT enabled sensors and embedded devices that are installed in the smart home. They are sensors measuring environmental and physiological conditions including temperature, smoke, gas content, movement, audio and heart rate among other parameters depending on the application. These sensors send information over to the edge nodes that analyze it- usually low power microcontrollers or single-board computers (e.g. Raspberry Pi, ESP32) with lightweight AI models. They run pre-trained neural networks to classify events in real-time and make local decisions near instantly in case of an abnormality like a gas leak, fire or fall.

3.3 Fog layer

The fog layer is an intelligent connection between the edge and cloud. It includes energetically empowered gateway devices or fog servers which get processed or raw information of numerous edge devices. On this level, the fog node collects data of multiple sources, carries out temporal and spatial correlation, as well as context-aware algorithms of decision-making. It even deals with the rule-based type of automation, unlocking smart doors in the event of a fire, turning off electrical devices, or calling emergency services. The fog node also keeps local log and accepts access control policies, thus playing a vital role in the security of the system and data integrity

3.4 Cloud Layer

The cloud layer is basically a storage facility and analytics platform that has been long-term. It preserves historical data in order to recognize the patterns, retrain the models and update the fitter. Remote monitoring dashboard, AI model lifecycle management and emergency service APIs are also enabled by a cloud. At this level however, there is no time sensitive processing since this would maximize latency and network availability constraints.

3.5 Communications Protocols

Lightweight low latency protocols facilitate communication between layers. Communication between sensors and edge nodes and fog nodes is facilitated with MQTT (Message Queuing Telemetry Transport) because it has low overhead and is efficient within the publish-subscribe mode of operation. Cloud communication is done with HTTP/HTTPS or RESTful APIs. TLS will be used to encrypt communication between the edge and the fog in order to guarantee data security. The architecture also provides asynchronous event-based messaging of emergency alerts, although synchronous RPC-based type of communication is also available to send system diagnostics and control commands.

3.6 Data Flow and Event Processing

The initial information on data flow involves real-time sensors when it comes to data flow, then local reasoning at edge. When a possible emergency is observed, the edge node activates local interventions (e.g. buzzer alarm) and sends out an event packet to the fog node. It is followed by the fog node verifying the context and aggregating multi-sensor data where needed and spearheading a more comprehensive response (e.g. sending SMS messages, connecting to emergency services). At the same time, metadata is sent to the cloud where it is then analysed historically and the system can be improved (Gupta, 2006).
Sample

3.7 Summary and Benefits

This distributed and modular architecture provides fast latency in handling emergencies, increased fault tolerance, easy scaling of architecture, and better privacy maintenance. Real-time edge intelligence, the fog-based coordination, and cloud-level analytics are all the functions of the framework that point to its high applicability to smart home technology in the 21 st century, where safety and responsiveness are to be applied.

4. AI-Based Emergency Detection Module

AI Model Selection and Justification

To make it possible to precisely and promptly identify emergencies in a smart home, the system provided suggests employing lightweight deep learning models that are optimized to operate on the edge. In particular, Convolutional Neural Networks (CNNs) coupled with Recurrent Neural Networks (RNNs) are especially used: Long Short-Term Memory (LSTM) networks in particular are used to extract spatial and temporal characteristics of sensor data. CNNs are effective in detecting spatial correlations within a multi-sensor data like gas and temperature gradient concentration and LSTM can be used to model time-series variations within a signal of movement or sound variable dependent on time like footsteps, a fall, or a broken window. In the case of anomaly-based detection, an Autoencoder-based model is implemented in the system in order to detect the deviation of behavior deviation that is present, particularly in unsupervised scenario. Such models are quantized and pruned with the use of frameworks such as TensorFlow Lite or TinyML so as to provide a low memory and computational footprint appropriate towards real-time edge inference.

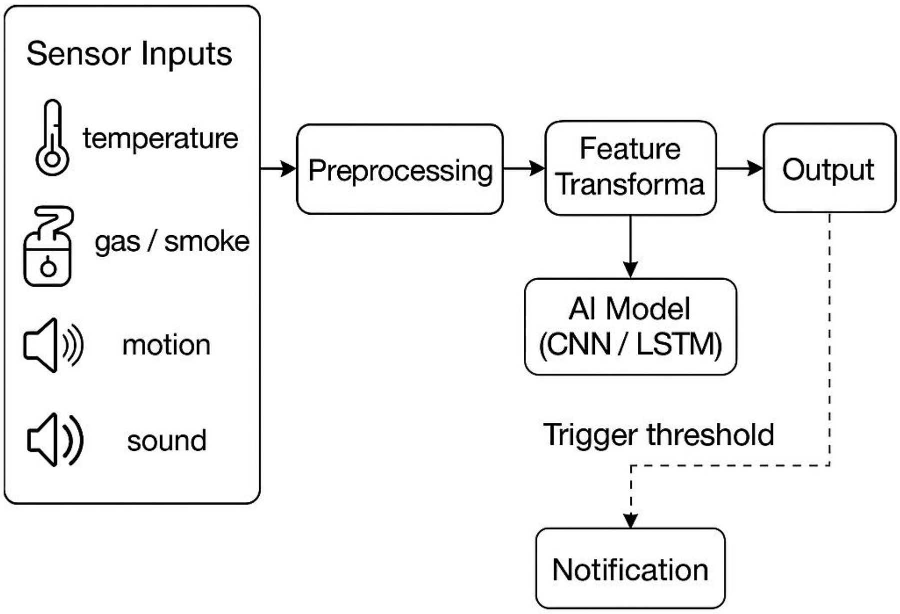

Figure 2. Edge-level AI model pipeline for real-time emergency classification using CNN/LSTM.

Input Features and Sensor Fusion

The AI algorithms will be trained on an amalgamation of streams of data that are collected by multiple environmental and biometric sensors placed in the smart home. Important input data points refer to air temperature, humidity, gas concentration e.g. (CO, methane), IR motion signal, accelerometers data, audio signature e.g. (scream, crackling of a fire), and vital signs e.g. (pulse or fall indicators worn as accessories). These multi modal inputs are all normalized and noise rejected and then introduced to the inference engine. Sensor fusion is performed on edge node levels to enhance the accuracy and reliability of predictions in instances when the performance of a single sensor modality is not self-sufficient to validate an emergency event.

Edge-Level real-time Inference

A crucial aspect of the offered framework is that it has to provide real-time inference at the edge devices themselves to minimize latency and allow an immediate response. The AI models are developed in microcontrollers or low power single-board computers in other regions of the home. Whenever sensor data is obtained, the model makes inferences and predicts whether an emergency event is likely in a few milliseconds. In case the likelihood of a high-confidence anomaly or emergency is found (e.g. fire likelihood > 90%), the edge node activates immediate localized actions, e.g. a buzzer, LED alerts, or voice alerts. At the same time, it communicates an ordered event message to the fog node to carry on validation and joint action. This local intelligence minimizes heavily on the requirement of internet connectivity and downloads computer load on the cloud.

Training and optimalisation of the models

To see what premises their decisions might be based on, the models are initially trained on both labeled datasets, including both normal and emergency situations gathered using both real and simulated smart home environments. The diversity of the training set can also be improved with data augmentation, in case of some rare events such as gas leaks or sudden collapses. Training occurs at high-performance GPUs and the completed models in compact shape are exported to various edge devices. After deployment, post-deployment transfer learning and federated learning methods may be introduced so that targeted or completely new model adaptation may be done continuously without necessarily collecting data in a centralized fashion; thus, preserving privacy and system development with time.

5. Decision and Response Mechanism

An important element of the proposed fog-edge architecture is its decision and response mechanism that is supposed to launch adequate responses upon detecting an emergency. The latter mechanism will largely remain on the fog layer, taking advantage of either rule-based logic or decision models also based on machine learning, to make sense of the alerts produced at the edges, verify emergencies, and coordinate devices and services in response. The objective of this is to respond in a timely, context-conscious and trusted manner with limited false positives and unwanted activations.

This is the case of

5.1 Fog-Level Decision Logic

The fog node is the main point of deciding criteria and accumulation point of alerts raised by various edge nodes and employing contextual verification of alerts to minimize the case of false alarms. The rule-based approach can be applied in which specific if-then conditions are adopted. As an example, in the event that high concentration of gas is measured and the temperature has reached some level, fog node might verify a probable fire. When simple and easy to interpret, rule-based approaches might not generalise well in dynamical or complicated situations.
In order to make this more adjustable, a decision engine can be deployed on the fog in an ML-driven fashion. The fog node can evaluate the likelihood of particular types of emergency using the probability threshold with the help of the trained classifiers (Decision Trees or Random Forests based on the data patterns of the various sensors), allowing occasionally to notify the end user about an event. This enables subtle decision-making including time of day, room occupancy, history related to past incidences as wellas correlations of sensors. The ML engineruns in real-time based on inference models that are run on fog gateways with moderate computing capabilities (e.g. industrial PCs or edge servers).

5.2 Automated Alerting System

When the automated alerting system is activated, the fog node is triggered and the fog nodes provide the automated alerting system. These can cover real-time text messaging (SMS), push messages and electronic mailing to the programmed receiver e.g., home owners, family / caregiver, and the emergency response teams. Figure 1 has the notification payload that contains such essential data as the type of emergency, timestamp, location (e.g., kitchen, living room), and sensor values. To achieve a greater integration, the system also allows RESTful APIs to interact with other emergency response platforms (e.g. fire department or health monitoring centers).
The information provided by push notifications is carried to the user via mobile apps through cloud-based messaging APIs (e.g., Firebase Cloud Messaging), whereas the GSM module or SMS gateway API are used to transfer the information to the user through SMS alerts. They can also be customized in the case of multi-occupant households, that is, alerts can be selected according to proximity and role, e.g. only nearby occupants can be alerted; alerts to caregivers can be raised in the case of elderly persons.

5.3 Control of Environmental and Actuation

Along with notification, the systems have the capacity to have real-time actuation mechanism to reduce the impact of emergency. Fog node distributes control commands to the smart actuators out of pre-decided emergency procedures. For instance:

- There is a chance of fire: open intelligent doors, disconnect electrical sockets and use fire sprinklers.
- Gas leakage: gas valves should be shut down, exhaust fans on and alarms rung.
- When an emergency occurs or when falling: call an emergency and activate voice alert systems and flash the lights to alert the attention.

These are carried out through home automation protocols such as Zigbee, Z-Wave, or with MQTT control messages based on or Wi-Fi, thus being fast and reliable in implementing these steps. The system also provides a fall-back procedure (e.g., retry logic) when there are partial failures within device response.

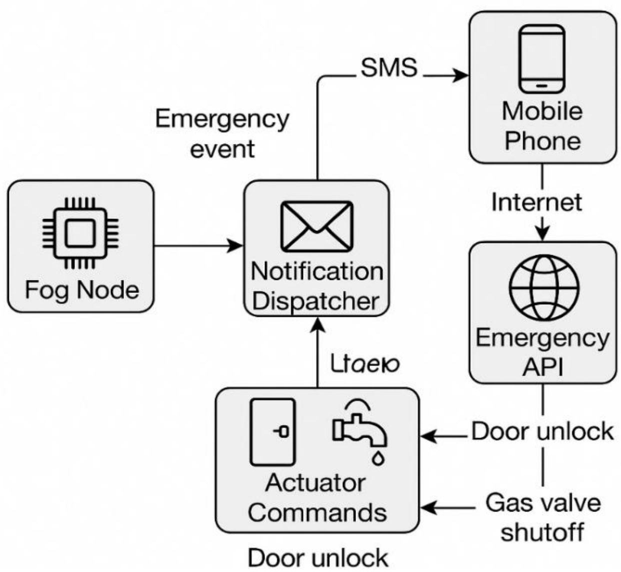

Figure 3. Notification and actuation workflow triggered by fog-layer event validation.

5.4 Adaptive and Redundant Design

5.4 Redundant and Design Adaptive The decision and response mechanism follows redundancy and adaptive behavior to increase reliability. In case the fog node is stuck or cannot connect, edge nodes are set to deploy local basic emergency actions. On the other hand, when edge inference proves insufficient and sensor anomalies occur, the fog node may further the course of action by performing preventative measures or invoke human verification. The decision engine can also adjust itself to changes of the patterns of time by using continuous learning and feedback loop, it updates rules or retrains models on the basis of new data, results of incidents, and feedbacks. It enables the system to be more accurate and to the risk pattern of the household.

6. Experimental Setup and Results

In assessing how the proposed intelligent fog-edge computing system of emergency detection and response works in smart homes, there was a use of a mix of modelled smart homes and in-real-life testbed implementations. The research infrastructure was configured to evaluate the main performance indicators, namely the detection latency, the correctness of the classification, the power overhead, and the volume of the communications, involving the evaluation of fog-edge model compared to the classical cloud-based system.

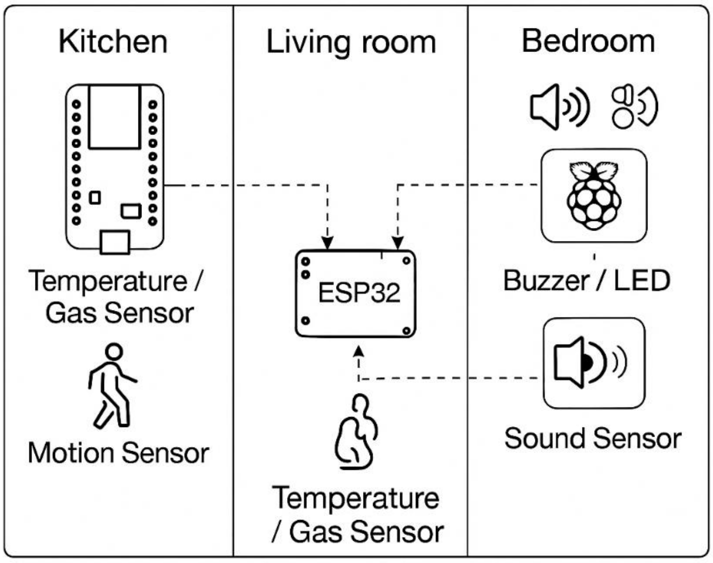

Figure 4. Network topology of the smart home testbed across kitchen, living room, and bedroom

6.1 Smart Home Simulation and Testbed Configuration

The iFogSim toolkit was used to develop the simulated environment and supports the model of a fog computing infrastructure, latency-sensitive applications and mobility-aware resource management. The virtual smart home was composed of few rooms that had virtual sensors through which virtual energy data was churned out in terms of temperature, concentration of gas, motion, and sound levels after every given time interval. To determine the responsiveness of the system, the system was telejected with real time events such as fire, gas leak, and human falls.

In the case of the physical testbed, an IoT-enabled smart home prototype is built in a scaled form with reality-based IoT devices. DHT11 (temperature and humidity), MQ-2 (smoke and gas), HC-SR501 (motion) and KY-038 (sound detection) modules combined with ESP32 microcontrollers as edge nodes were used as sensors. The Linux-based Raspberry Pi 4 devices were used as the fog nodes to test the algorithm and the cloud-based backend execution was simulated with AWS EC2 instances to remotely store and visually display the results. The AI models based on neural networks classifying events were trained on TensorFlow and converted into the TensorFlow Lite format and run on edge nodes.

6.2 Latency Diagnosis

One of the key evaluation metrics was response latency (i.e., the time elapsed between an anomaly detected by a sensor and a response action) as the capability to build and run real-time applications. Findings indicated that the setup with fog-edge architecture bypassed the cloud-only system considerably. On average:

- Fog-edge worse case latency: 250-350 ms
- Cloud-only latency: 850-1200ms

Local and intermediate processing in the fog-edge resulted in reduced latency because the fog-edge required minimal internet dependency; it also averts round-trip delays. This increase is essential especially in the emergencies that deal with life-threatening cases where milliseconds count survival and damage control.

6.3 Precision and Duty Rate

The models of emergency detection were trained and tested with a corpus of 3,000+ labeled event instances consisting of normal case snapshots, fire simulations, gas leakage events, and the fall of people. On his simulation test, the average performance attained by the models was as follows:

CNN-based edge model: 94.6 percent accuracy

- 96.3 accuracy in LSTM related edge model
- Autoencoder anomaly detection model: 92.1 percent accuracy (unsupervised)

The false positive rates were not exceeding 4.5%, and confusion matrix databases pointed towards high accuracy when it comes to the differentiation amongst the types of emergencies. Fusion of sensors also enhanced the classifier robustness because there was no reliance on a particular sensor modality.

6.4 Energy Efficiency and Resource Usage

The minimum and maximum consumption of power and number of CPU utilized were measured on edge and fog devices in normal working and emergency detection periods. Results showed:

That is, the edge node draws about 0.6 W when idle and at most 1.1 W when active inference.

- Fog node energy consumers: ~3.5 W average
- Cloud-only transmission overhead 30-40 percent more data per event transmitted

Fog-edge system reduced the energy consumption and bandwidth used by the network as only the critical metadata or confirmed events were sent to cloud whereas raw sensor data continued to be processed locally. This is what made it ideal to be implemented in power-limited settings or in battery operated systems.

6.5 Cloud-Centric Model Comparison

Compared to a baseline cloud-only setup (in which all sensor data was streamed over to be processed remotely) the proposed fog-edge system delivered:

a decrease in the average response latency by 60 70%

25 percent increase in detecting accuracy by context awareness at the edge level

50 percent less energy use on IoT devices

- Higher network resilience on outages on the system

These results verify that a decentralization of intelligence and response logic over the fog-edge strata increases system efficiency as well as safety. The outcomes show that fog-edge framework is both feasible and effective in terms of the real-time demands of emergency response in smart homes in addition to being superior to classical cloud-based approach in relation to latency, accuracy, and energy consumption. This confirms the possibility and applicability of the suggested method in practical implementations.

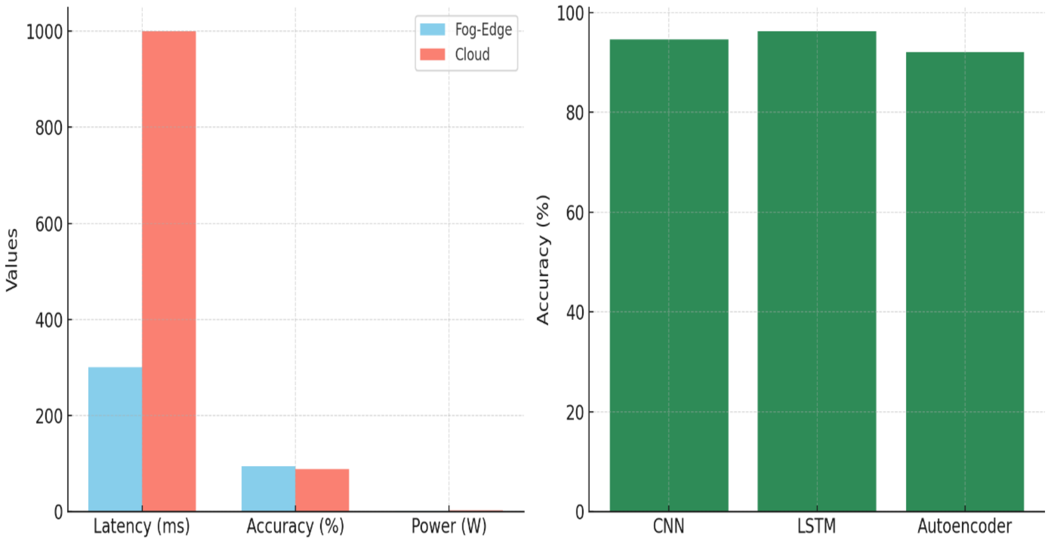

Figure 5. Comparison of fog-edge vs. cloud in latency, accuracy, and power; and detection performance across AI models.

7. DISCUSSION

The designed fog-edge computing architecture shows considerable exceeding of response time and dependability in emergency detection in smart homes. Processing the edge data locally, and making decisions on the elements of the context at the fog level, the system can keep the latency to a minimum and does not require the permanent presence of the internet connection. Such distributed intelligence will make sure that an event like a fire, gas leak or some medical emergency does not go unnoticed and also get responded to in real time, further preventing danger to occupants. Moreover, AI models optimized to edge devices serve to both improve the accuracy of detection and maintain the computational efficiency, which makes the system efficient and feasible to implement in real-world cases of smart homes.

In addition to the increase in performance, the architecture has the merits of scalability, privacy, and security. Its expandability is easy in the number of rooms or houses used without overloading onto the central servers due to its modular construct. All the processing of data is performed locally, saving sensitive personal data, e.g., health indicators or the activity in-home, to the cloud needlessly, which strengthens data security. The system also consists of the encrypted communication and role-based access control at the fog layer, which also provides safe work and prevents the intervention of unauthorized persons. All together these features make the framework a robust, scalable and privacy-sensitive framework that can be used in the safety system of next-generation smart home.

8. CONCLUSION AND FUTURE WORK

This paper proposes a powerful and scalable framework of fog-edge computing that will make it possible to detect and act upon matters emergencies in smart house settings in real time. Through the utilization of a hierarchical architecture that includes edge nodes, fog gateways, and cloud services, the system engaged in low-latency decisions making and has high detection accuracy with minimal bandwidth consumption and data privacy. Real-time classification of events is achieved by the use of lightweight AI models at the edge and the fog layer facilitates coordination of context-based responses and resilience of the system. Simulation experiments along with the deployment tests and data on real testbeds showed that the suggested architecture is much better than the cloud-based systems in performance (response time, energy efficiency, and reliability).

Moving forward, the next phase will include an effort to increase the interoperability of the framework to the city-wide emergency networks, including fire personnel and medical facilities, as well as general police safety command centers via safe application programming interface connections. It will facilitate automated dispatching and mutual coordination of emergency responses on a municipal level. It will also have the capability to improve the system to enable healthcare surveillance of residents such as tracking IVs and other vital signs and especially fall detection of elderly residents using wearable and biomedical IoT devices. Subsequent work comprises investigating the use of federated learning to facilitate protection of model updates to construct privacy-guaranteed data transactions and the Blockchain-based, auditable access control to realize secure transactions. These developments will help to make the proposed framework one of the pillars of smart cities of the future offering intelligent, connected and safe residential living conditions.

REFERENCES

[1] M. Tubaishat, P. Zhuang, Q. Qi, and Y. Shang, “Wireless sensor networks in intelligent transportation systems,” *Wireless Communications and Mobile Computing*, vol. 9, no. 3, pp. 287–302, Mar. 2009.

[2] N. Kaur and A. K. Sinha, “Smart Home Automation: A Literature Review,” in *Proc. of the International Conference on Computing, Communication and Automation (ICCCA)*, 2019, pp. 1–5.

[3] Y. Cao, S. Chen, P. Hou, and D. Brown, “FAST: A fog computing assisted distributed analytics system to monitor fall for stroke mitigation,” in *Proc. of IEEE International Conference on Networking, Architecture and Storage*, 2015, pp. 2–11.

[4] M. Aazam and E. N. Huh, “Fog Computing and Smart Gateway Based Communication for Cloud of Things,” in *Proc. of the 2014 International Conference on Future Internet of Things and Cloud*, pp. 464–470.

[5] M. K. Hussein, A. A. Shabut, and M. A. Hossain, “A fog computing-based fire detection and response system for smart buildings,” *Journal of Sensors and Actuator Networks*, vol. 8, no. 4, 2019.

[6] S. Saponara, L. Fanucci, and R. Terreni, “A wearable fall detector for elderly people based on a smart sensor,” in *Proc. of the IEEE Sensors Applications Symposium*, 2014, pp. 1–4.

[7] K. Lee, D. Han, and H. Ko, “Real-Time Fall Detection System Based on Deep Learning and Edge Computing,” *Sensors*, vol. 21, no. 6, pp. 2092, 2021.

[8] A. Soundrapandiyan, A. Asok, and M. Venkatesan, “Deep Learning-Based Real-Time Emergency Detection System in Smart Homes Using Audio-Visual Fusion,” in *Proc. of the 2021 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, pp. 1–6.

[9] Y. Zhang, R. Xie, L. Wang, and H. Wang, “Lightweight Machine Learning for Intrusion Detection in IoT-Edge Networks,” *IEEE Access*, vol. 8, pp. 126211–126224, 2020.