Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7096561, 20 pages
https://doi.org/10.1155/2022/7096561

Research Article

A Novel Machine Language-Driven Data Aggregation Approach to Predict Data Redundancy in IoT-Connected Wireless Sensor Networks

G. C. Jagan [1] and P. Jesu Jayarin [2]

[1] *Faculty of Electronics Engineering, Sathyabama Institute of Science and Technology, Chennai, India*
[2] *Institute of Information Technology, Department of Data Analytics, Saveetha School of Engineering, SIMATS, Chennai-602105, Tamil Nadu, India*

Correspondence should be addressed to G. C. Jagan; jagan11gc@gmail.com

Received 30 June 2022; Accepted 20 September 2022; Published 18 October 2022

Academic Editor: Yan Huang

Copyright © 2022 G. C. Jagan and P. Jesu Jayarin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Real world data aggregation and delivery in Internet of Things (IoT) technology are essential to predict and retrieve target data in short time so that the end user feels no delay but ensures a high quality of information. In addition to habitat monitoring and disaster management, these networks have a wide range of other uses, including security and military operations. The processing capabilities of sensor nodes are restricted due to the fact that they have a limited battery life and hence a modest size and processing capacity. WSNs are also susceptible to failure as a result of the limited battery power available. In WSNs, data aggregation is practiced as an energy efficient strategy to reduce computing and transmission latency. It is because of sensor node distribution density that shares the same data at a time data redundancy comes to exist. It is possible to reduce redundancy by adopting a suitable machine learning algorithm while executing the data aggregation process. Researchers are still chasing behind algorithms and modeling strategies effectively to ease the process of developing an effective and acceptable data aggregation strategy from existing wireless sensor network (WSN) models. A three stage framework is proposed for an efficient data aggregation mechanism, and the stages are Modified LEACH, extreme learning machines (ELM), adaptive Kalman filter, and Bi-LSTM. This experiment result shows better performance than the existing methods.

1. Introduction

Today, the world is experiencing a tremendous use of digital data due to which the growth in data collection and distribution processes has been rising in a rapid pace which in turn leads to be a vital decisive factor in developing IoT system architectures [1]. It is evident that the amount of data collected from various digital communication infrastructures gets doubled at the rate of two times a year. Several efficient data collection models are developed to upgrade the performance of many IoT-assisted sensing applications. The sensor systems perform data acquisition which forms the first part of data collection process to provide better services to users. It is known that there are several numbers of sensors deployed in wireless sensor networks to perform data collection from industrial and other natural or artificial environments. The main goal of sensor deployment is to sense data of intent from a hostile area. As these sensing mechanisms consume significant energy, different network-based data processing mechanisms are considered to develop sensor centric IoT applications. Such data aggregation techniques in IoT-enabled networks conserve energy with network longevity.

WSN finds its applications in areas such as home automation, monitoring different types of surroundings, healthcare, and industrial control to mention a few. The communication between constituent sensors will lie in a short range only [2, 3], wherein the sensor nodes have limited bandwidth and other associated resources. The sensor works on collecting signals from sources such as light, temperature, and heat and passes it to data conversion units called microcontrollers. Overall communication effectiveness of sensor nodes would be dependent on type of data aggregation techniques employed. Data aggregation is carried out with few objectives so that it reduces energy consumption along with other resource utilizations. The network lifetime would also increase when the data aggregation algorithm is carefully chosen [4]. Data aggregation is preferred in cases where multiple sensor nodes are in operation to fetch signals on a same parameter under a high node density scenario.

In this research work, energy minimization is carried out in two stages such that at first, data prediction and, secondly, a statistical prediction modeling are performed out, respectively. Data prediction is done in an IoT network to predict future data coming from all live nodes. A given IoT network contains a sensing node known as aggregator to collect data and broadcast it to remaining nodes. The aggregator node sends only the required amount of data instead of sending all received data after processing it using suitable data precipitation techniques. Besides, a sufficient data reduction is further achieved during the second phase using a statistical data prediction model to identify neighboring nodes which periodically generate data.

Figure 1 portrays various applications of IoT, where data aggregation plays a major role in a large scale. Thus, this manuscript focuses on developing efficient data aggregation mechanisms for IoT-enabled machines available in industries.

2. Literature Survey

WSN is prominently used in IoT, to collect environmental data because of its large-scale deployment, low cost, and low-energy consumption. It is time-consuming and difficult to reduce the amount of data gathered and transferred across a network without affecting data integrity. Literature [5] proposes a two-stage model in which the first stage is for effective data collection by unmanned aerial vehicle (UAV) where as the next stage is NP hard maximization problem to model the full or partial collection of data by hovering of the UAVs.This proposed two stage model tries to maximize data collection with minimal energy consumption.

Researchers on [6] proposed an Energy-Efficient Data Aggregation Mechanism (EEDAM) to save energy at the cluster level. Edge computing is used to give on-demand trusted services to IoT devices with the least amount of delay, and blockchain is incorporated into a cloud server for verifying the edge in order to provide secure services to IoT devices with the least amount of delay. In work [7], author presents methods on how to effectively deal with the data veracity issue that arises due to the existence of misbehaving nodes, outliers, missing readings, and redundancy in the raw IoT sensor data by using a data aggregation technique. The data aggregation methodology is intended for use with extremely uncertain raw IoT sensor data acquired by device to device connection, as opposed to other methods.

In [8], authors presented a review paper for aggregation which proposes LTE-WLAN aggregation (LWA) that is implemented using a Software-Defined Networking- (SDN-) based technique to manage aggregation across LTE and WLAN Access Points eliminating excess connection attempts thus serving users only with essential services. The genetic algorithm is used to pick the best WLAN access point. This reduces the traffic demand in licensed spectrum and increases the UE throughput. Another paper [9] describes several energy efficient data aggregation techniques employed in sensor networks.

In [10], authors reported a review article which covers a detailed analysis of methodical analysis of data aggregation in WSN. Here, they discussed about challenges in data aggregation and various methods and tools used. In [11], a survey is found on various data aggregation models which used machine learning techniques and finally, authors proposed a novel priority-based data aggregation (PbDA) technique which is machine learning based on confront emergency situations. Again, in [12], research work proposes an adaptive event differential privacy (Re-ADP) system, and all the collected sensor information at different timings may be protected sequentially through an unlimited stream of data in real time, without compromising performance. They are meant to provide aggregated data to cloud storage; it may reduce the processing load on cloud storage servers, enhance communication efficiency, and preserve the privacy of data sent to them.

Another unique data aggregation approach with network clustering and an extreme learning machine (ELM) that effectively removes unwanted and erroneous data is identified in [13]. The instability during training phase of ELM is tackled using basis functions. All the sensed data are preprocessed to filter noise using Kalman filter before delivering data to a particular CH. This supports to accuracy improvements. Again, an energy-efficient data aggregation strategy is proposed in [14] in which IoT nodes encode sensor input into a binary format before routing. Next, the data is compressed at the edge node and then pumped into the IoT cloud through the shortest route. An accurate data aggregation and prediction model improves performance of the cloud.

An energy efficient LEACH protocol is introduced [15, 16] to improve network routing by reducing resource consumption. Clustering points out an appropriate cluster head to reschedule TDMA slots of a particular sensor node and to balance all sensors data transmission such that each node sends the same quantity of data. This reduces energy consumption of nodes and hence increases life of the network.

The research contributions in [17] focus on theoretical analysis of extreme learning machine (ELM) to achieve improvements in the context of stability, efficiency, and accuracy of WSNs. The work present in [18] aims at fast traffic forecasting to predict vehicle count that is anticipated in the successive time period in a traffic junction. During uncertain signal transmissions, adaptive Kalman filters offer reasonable prediction intervals according to empirical evidence, along with a better adaptability in a variable traffic scenario. It is inferred that during sensitivity analysis, the adaptive Kalman filter performance is stable as its memory capacity increases.

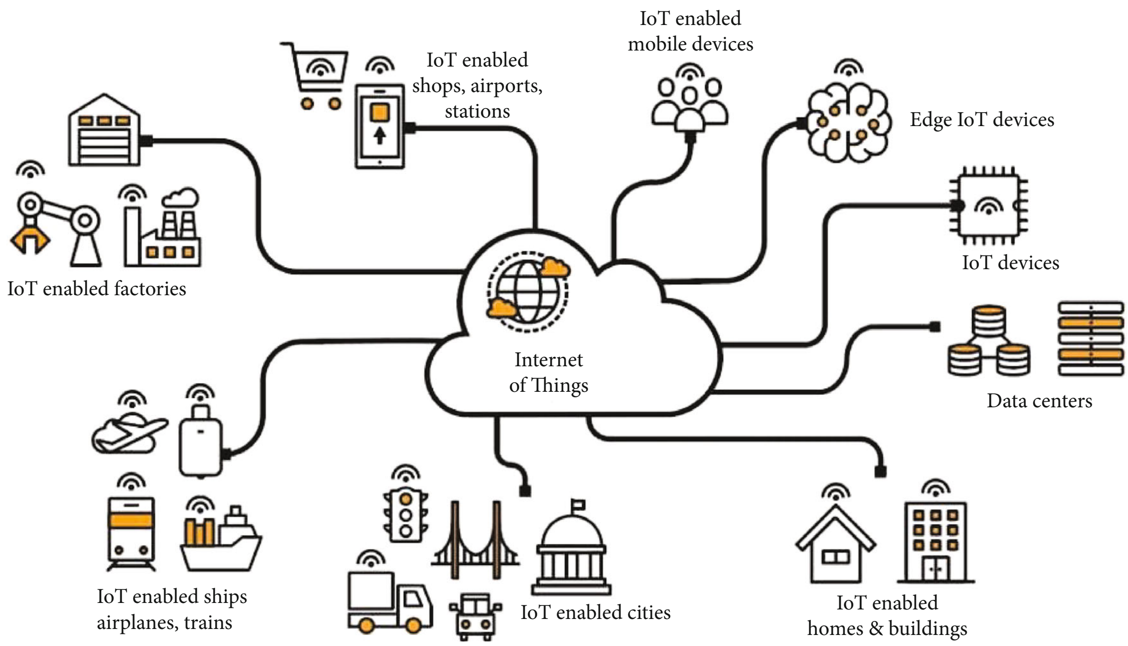

Figure 1: Structure of an IoT model (source: http://www.tibco.com).

In [19], a CLSTM-based model is proposed to precipitate the nowcasting issues. It uses both input and prediction output as spatiotemporal sequences. The resultant correlations outperform FC-LSTM and operational ROVER algorithm. Research reports in [20] used an LSTM network enhanced with word embeddings which is already trained using a significant number of Twitter message samples. It applied tokenization, word normalization, segmentation, and spell correction to optimize the identification of the most significant words.

Study in work [21] reveals that the research work implements aggregate sales forecasting using the deep learning algorithm ConvLSTM, which is developed by the University of Michigan. When looking at the sales forecasting, it might take geographical correlations between neighboring shops into consideration. It is discovered that the proposed ASFC method reduces errors and improves prediction quality. It is the goal of this effort to develop aggregate sales forecasting using the deep learning algorithm ConvLSTM.

Research in [22] uses a new prediction approach to develop sensor-connected IoT applications. Bi-LSTM and 1-D CNN are used to extract characteristics with distinct features, resulting in one-step prediction. After recursively combining previous data with new prediction findings, a multistep prediction model is arrived that significantly improved the performance.

An IoT system for Wireless Medical Sensor Network (WMSN) is proposed [23] to monitor medical data transmitted to central storages. The work deals with offering highly secured data aggregation and then transmit to desired locations. This prevents the intrusion of undestined users. The current schemes are significantly complex due to the use of complex product functions to generate batch keys. Thus, it makes the systems experience high computational complexity and large memory utilization. Authors presented a new lightweight Secure Aggregation and Transmission Scheme (SATS) that has a low complex EXOR logic to find out the batch key by eliminating the tedious multiplication steps. In addition, the work includes Aggregator Node Receiving Message Algorithm (ARMA) for effective data aggregation. This collective approach is found to be one of the preferred choices for SATS to mitigate security threats, viz., denial of service, man in the middle, and reply kind of attacks in a given IoT-WSN scenario. Simulations in NS2 show that the proposed SATS presents a lightweight type of data transmission minimal computations and allied communication costs along with an improved memory size and low energy consumption.

As the security of WSN data transmission is a key factor to determine the quality of service, there are data aggregation (DA) schemes [24] framed with a suitable security mechanism to offer safe and reliable data delivery. This presents a review of secure data aggregation (SDA) focusing only on security threats.

This work [25] presents an energy aware and secured data aggregation algorithm. End results show that the proposed approach preserves nodal energy significantly besides achieving a prolonged network lifetime. Data aggregation becomes very effective in large scale of WSNs wherein a huge volume of data is involved and out of which only a particular size is useful, whereas the remaining are said to be redundant [26]. An increased redundancy will decrease the system performance in the context of additional computational overhead and unnecessary transmission besides memory wastage. Data aggregation aims at data mining where only the useful data is precipitated to ensure data transmission with better consistency, accuracy, and efficiency. Data mining plays a pivotal role in wireless sensor networks combined with Internet of Things to achieve remote data communication. A new redundancy checking approach is proposed in this work that performs better redundant data mining compared to other counterparts. Table 1 illustrates the comparison of some existing methods.

3. Proposed Method

With the help of WSN, we have developed an effective data aggregation technique for utilization prediction in IoT operated machines. To predict the following three stages, Modified LEACH, extreme learning machines (ELM), adaptive Kalman filter, and Bi-LSTM are used as shown in Figure 2.

3.1. M-LEACH. The LEACH method requires few nodes to assign as cluster heads which are more distant from the BS than they are, in order for the process to work. The sensor nodes transmit their sensed data to the centralized access points. Extra transmissions are communications that squander the energy of the network and are thus referred to as such. The suggested protocol operates over a large area, similar to that of a wide area network but simultaneously reducing the complexity of communication and the complexity of time management. All sensor nodes are deployed into a scalable distributed cluster environment in accordance with the suggested protocol, and the region is divided into various numbers of clusters. All the clusters are assigned with fixed cluster nodes. Any individual node should be attached with any of the cluster available in the network. Cluster heads are selected based on the modified LEACH algorithm. Only CHs will communicate with base stations.

Steps to be followed in modified LEACH:

Step 1. Deployment of sensor nodes.

Step 2. Formation of cluster.

Step 3. Calculate cluster head threshold for all nodes.

Step 4. Check the threshold value. If threshold is not higher, then the node is not a cluster head (CH) else proper selection of cluster head (CH).

Step 5. CH waits for join request messages.

Step 6. Broadcast a message from one node to all CHs.

Step 7. Modify TDMA schedule duration based on the largest cluster and send its cluster members.

Step 8. Sensor nodes send sensed data to its CH.

Step 9. Data aggregation on CH.

Step 10. Sent the data to extreme learning machines (ELM).

3.2. Extreme Learning Machines (ELM). The LEACH output is fed into an ELM to eliminate excess and error prone data. As illustrated in Figure 3, the ELM is a feed forward neural network with two stages of learning. The projection stage is nontrainable, and the input weights are chosen at random. No iterative calculation is required. This feature reduces computing time for training the model, but random selection of biases and weights causes prediction instability. To overcome ELM's flaw, a Mahalanobis distance-based radial basis function (MDRBF) is suggested to be integrated with ELM's network.

$$\text{TD (Training Data)} = \left\{ \left(c_j, d_j \right) \right\}, (j = 1, 2, 3, \cdots, n), \text{where } c_j \longrightarrow \text{input data}, d_j \longrightarrow \text{target data}. \tag{1}$$

The following is the formula for the ELM network with k hidden nodes:

$$\alpha_j(c) = \sum_{j=1}^{k} f^1 \left(w_j.c \right) + b_i, \tag{2}$$

where w is the weight and b is the bias.

$$\begin{gathered} d(c_i) = \sum_{j=1}^{k} \left(w'_j \alpha_j \right) + b'_j, \\ Xw' = d, \end{gathered} \tag{3}$$

$$\begin{gathered} X = \begin{bmatrix} \alpha(w_1 c_1 + \alpha_1) & \cdots & \alpha(w_n c_1 + \alpha_k) \\ \vdots & \ddots & \vdots \\ \alpha(w_1 c_n + b_k) & \cdots & \alpha(w_n c_n + b_k) \end{bmatrix}, \\ X \longrightarrow \text{output of the hidden layers}. \end{gathered} \tag{4}$$

The following equation is solved by least-square fitting:

$$F = \min \left(X w'_j - y_{ji} \right). \tag{5}$$

3.3. Adaptive Kalman Filter (AKF). In the field of data fusion, AKF is one of the most often utilized approaches. It decreases the amount of noise in the data and provides an accurate approximation of the state vector containing

Table 1: Reference comparison table.

Author details	Journal details	Observation
Li et al. [5]	IEEE Transactions on Mobile Computing (2021)	Approximation of data collection from sensor nodes fully and partially and heuristic approach maximization of collected data with NP-hard problem are used. Proposed methods proved promising results.
Sanyal and Zhang [7]	IEEE Access (2018)	Data veracity problem is addressed. Generated true sensor data matrix and proved higher efficiency in the presence of noise and outliers.
Huo et al. [12]	Wireless Communications and Mobile Computing (2018)	A real-time stream data aggregation framework with adaptive-event differential privacy is proposed. Authors used privacy protection and smart grouping based on K -means clustering. Results shown outperform the existing works.
Anbalagan et al. [8]	Future Generation Computer Systems (2020)	Alternate complementary network is utilized. Results proved that LWA-SA aggregates data with minimal latency and selects an optimal AP as prescribed in the GA-based EWS algorithm.

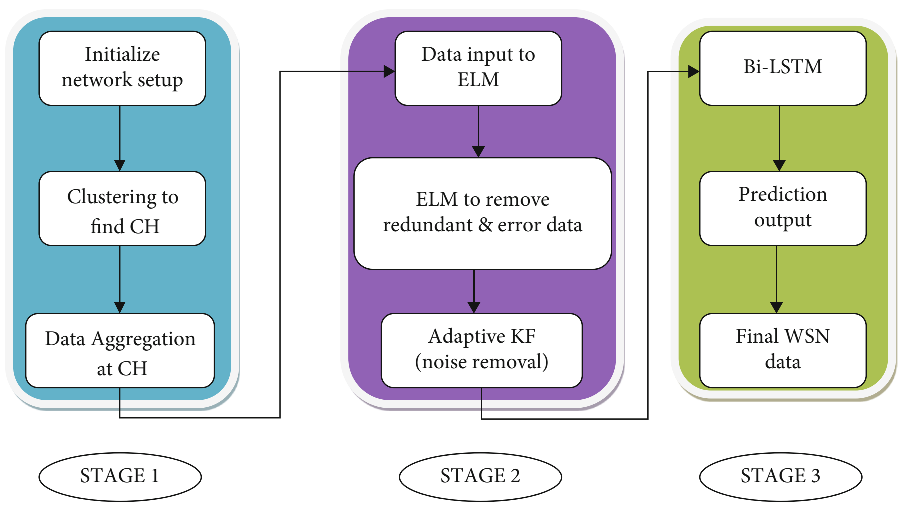

Figure 2: Three stages of data aggregation and prediction.

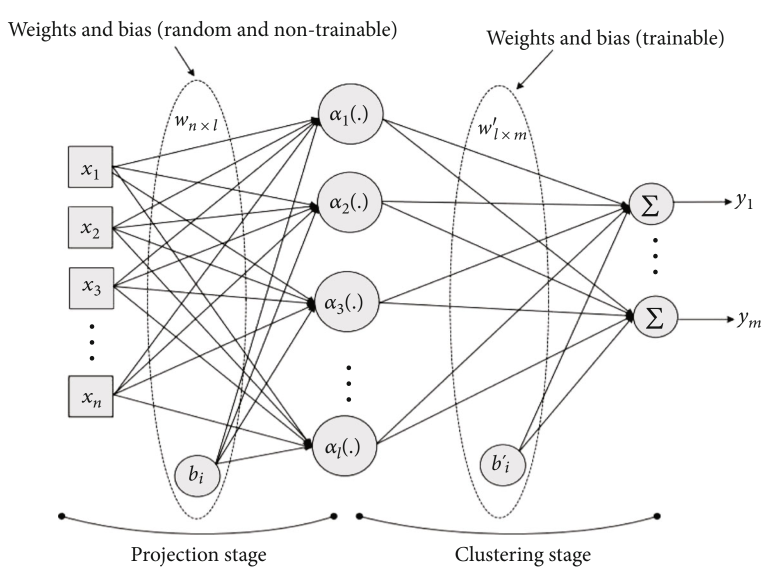

Figure 3: Structure of ELM network.

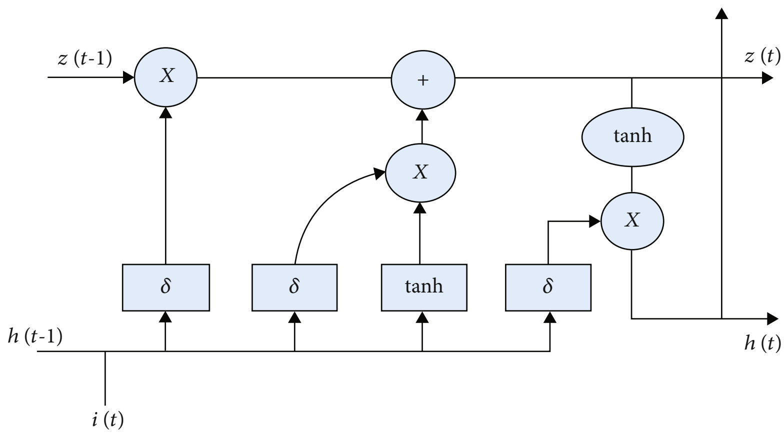

Figure 4: Structure of one cell.

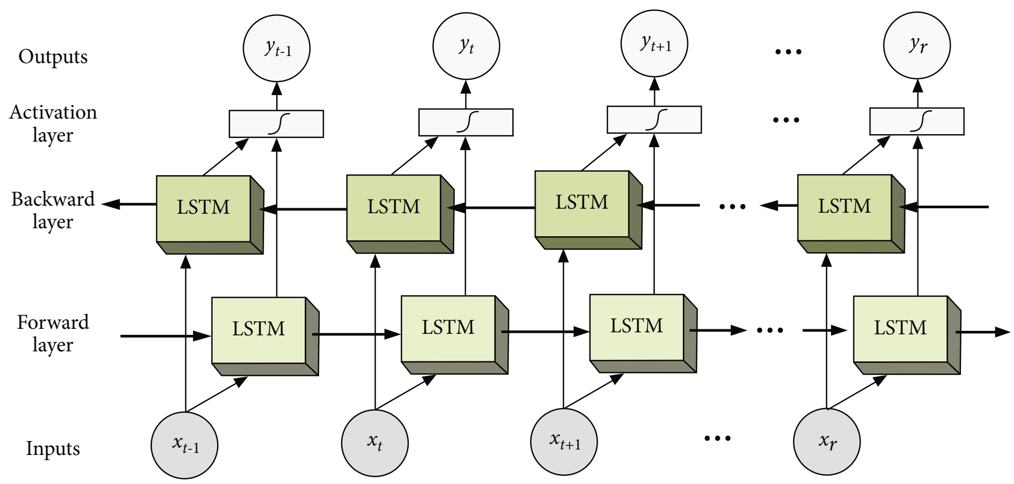

Figure 5: Structure of Bi-LSTM.

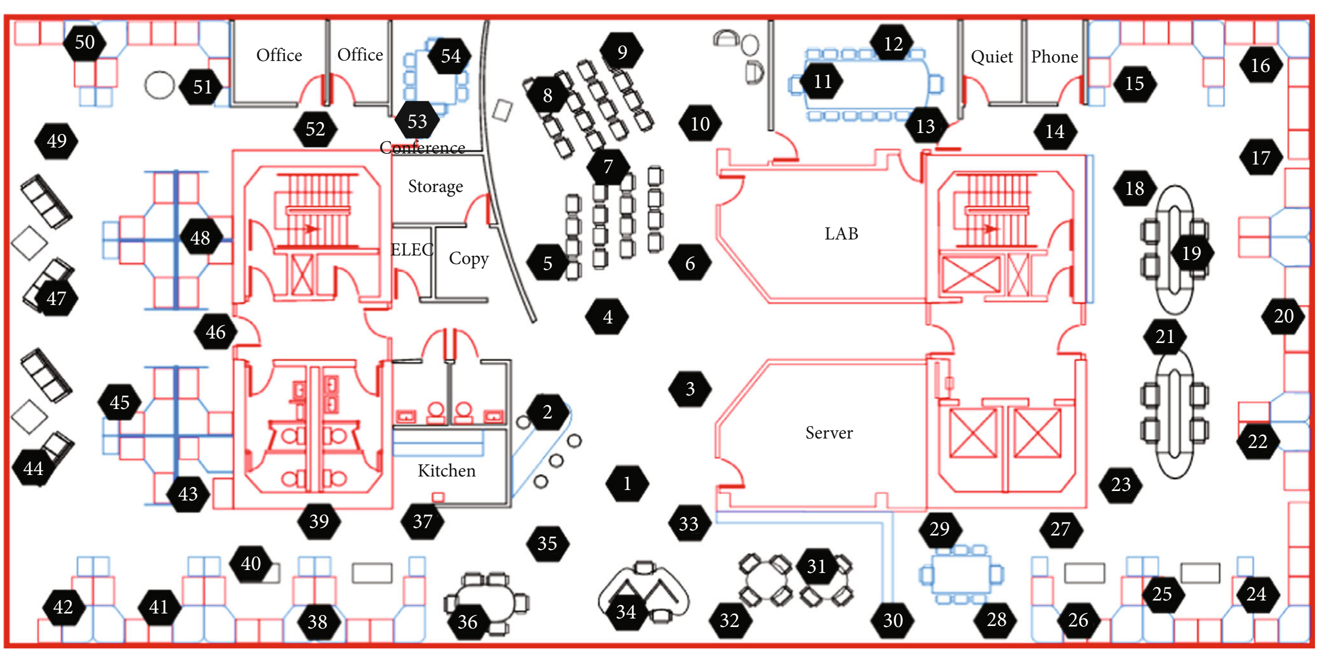

Figure 6: Sensor arrangement (courtesy: Intel Lab Berkeley).

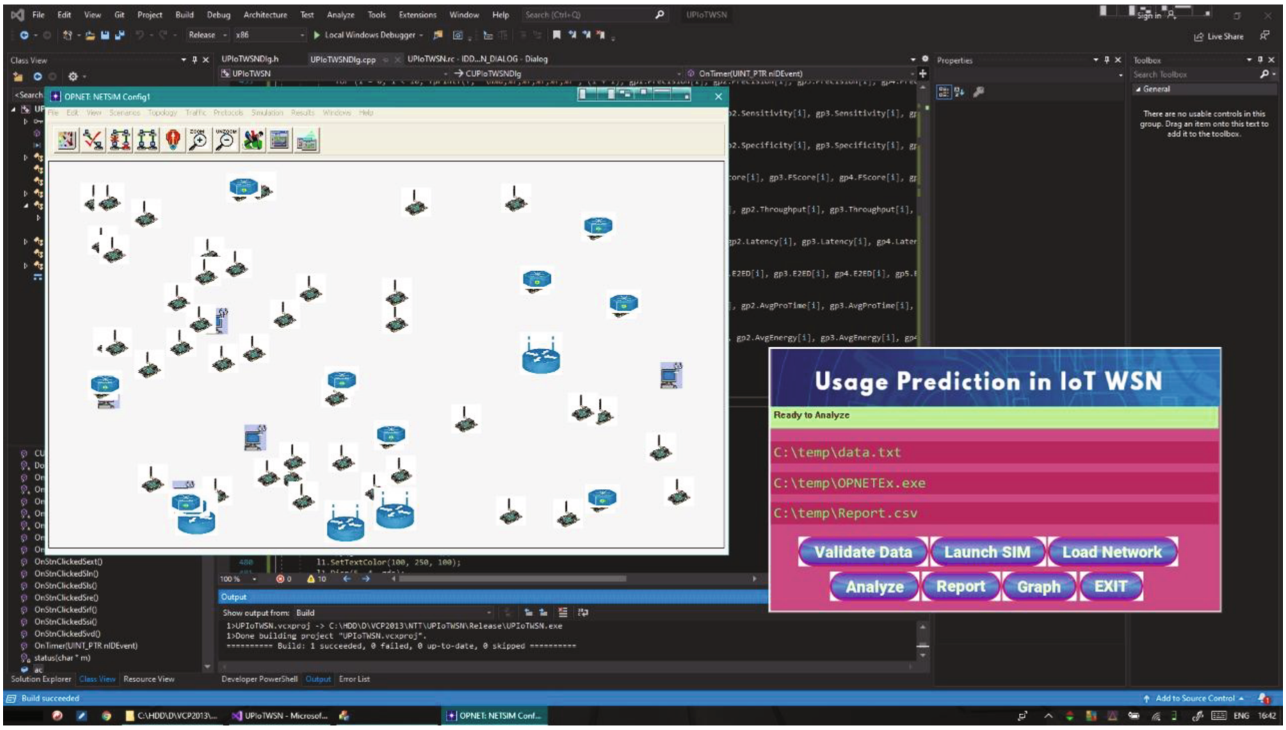

Figure 7: Experimental setup.

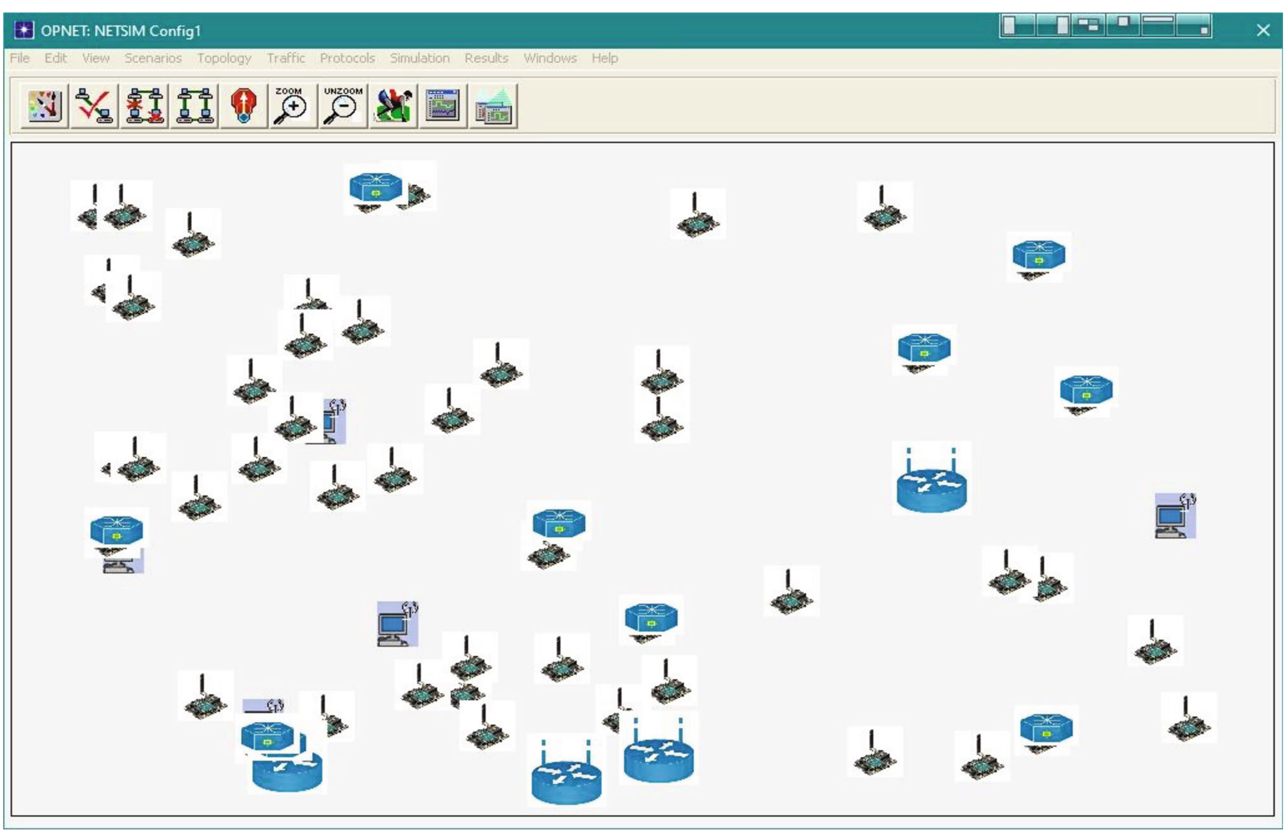

Figure 8: OPNET-network environment setup.

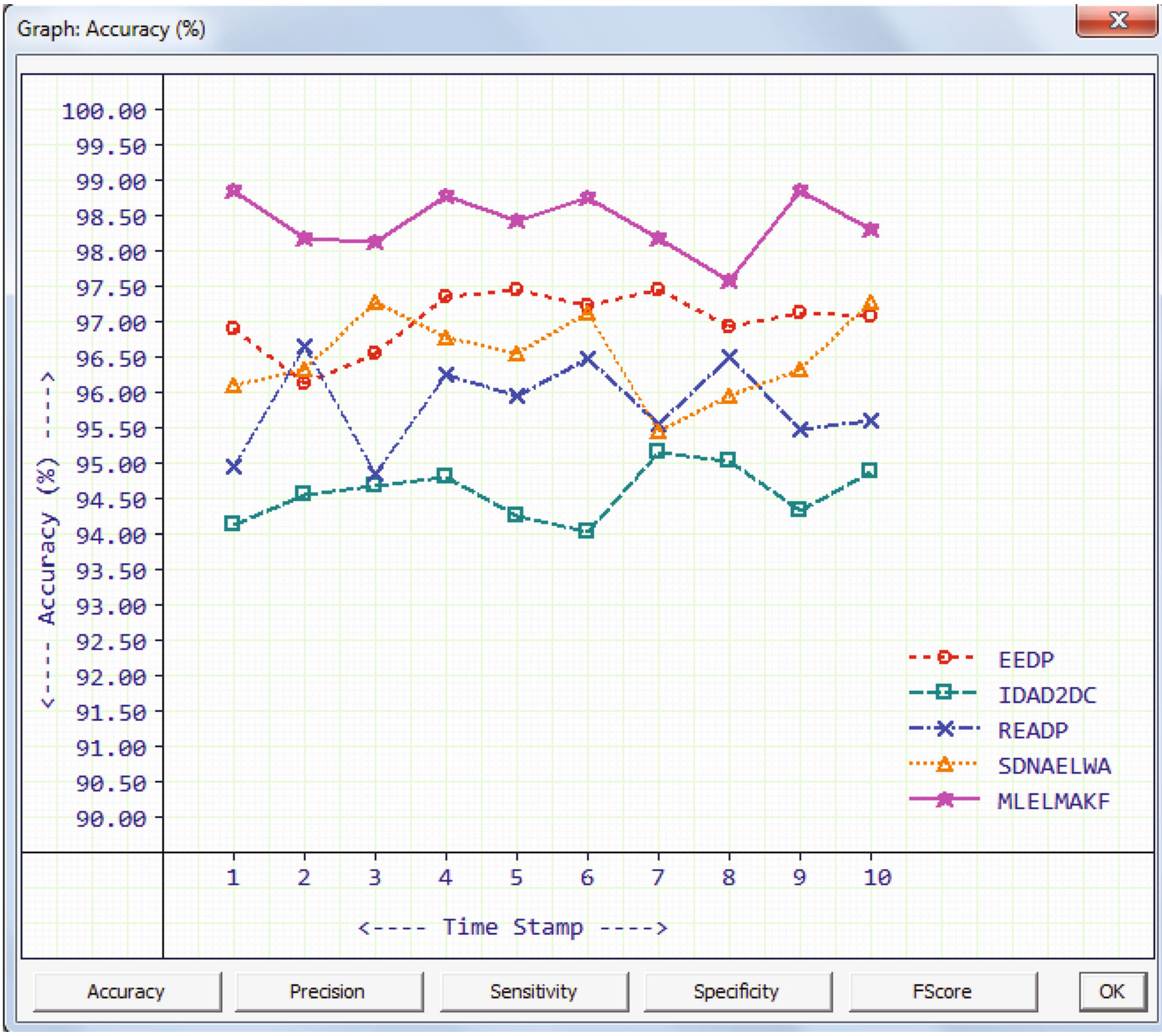

Figure 9: Accuracy.

valuable information. It has been widely used for a variety of applications, including estimate, tracking, and sensor fusion.

Step for AKF is given below:

Step 1. Find the prior state estimation error covariance

$$\widehat{S}^{-}_{k|k-1} = \widehat{\varnothing}\widehat{S}^{+}_{K-1|K-1}, \qquad \widehat{G}^{-}_{k|k-1} = \varnothing\widehat{G}^{+}_{K-1|K-1}\varnothing^{T} + R_k, \tag{6}$$

$G^{-}_{k|k-1}$: Prior State Estimation error covariance.

Step 2. Compute the errors

$$E_k = Z_k - Y^{T}_{k}\widehat{S}^{-}_{k|k-1}, \tag{7}$$

E_k : Obseravtion of errors.

Step 3. Update observation process covariance matrix C_k

$$\widehat{E} = \frac{1}{M}\sum_{i=1}^{M} E_{k-i+1},$$

$\widehat{E}$: Average observation errors,

$$C_k = \frac{1}{M}\sum_{i=1}^{M}\left\{\left(E_{k-i+1} - \widehat{E}\right)\left(E_{k-i+1} - \widehat{E}\right)^{T} - \frac{M-1}{M} Y_{k-i+1}\widehat{G}^{-}_{k-i+1|k-i} Y^{T}_{k-i+1}\right\}, \tag{8}$$

where M is the AKF memory size.

Step 4. Compute the gain of the AKF

$$KG_k = \frac{\widehat{G}^{-}_{k|k-1} Z_k}{Z^{T}_{k}\widehat{G}^{-}_{k|k-1} Z_k + C_k}, \tag{9}$$

KG_k : Kalman gain at time k.

Table 2: Accuracy in %.

Time stamp	11 : 07 am to 12 : 07 pm (60 minutes divided into each 6 minutes)										Average
	1	2	3	4	5	6	7	8	9	10	
EEDP [5]	97.105	97.635	97.14	96.59	96.39	96.2	98.005	96.46	96.135	97.26	96.892
IDAD2DC [7]	94.965	94.295	93.76	94.79	94.675	94.83	94.305	95.175	94.6	96.04	94.7435
READP [12]	96.195	94.7	95.835	96.085	96.03	96.225	95.52	95.8	95.895	94.585	95.687
SDNAELWA [8]	95.91	96.15	96.715	96.55	97.055	95.77	96.415	97.305	97.195	97.27	96.6335
MLELMAKF	98.18	98.395	97.79	98.125	98.54	97.925	97.93	98.32	97.6	97.725	98.053

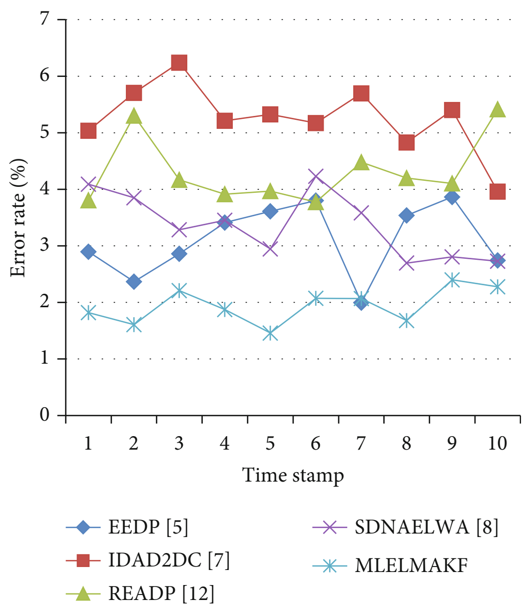

Figure 10: Error rate.

Step 5. Estimate the posterior state and its covariance error

$$\begin{aligned}\widehat{S}^{+}_{k|k} &= \widehat{S}^{-}_{k|k-1} + KG_k E_k,\\ \widehat{G}^{+}_{k|k} &= \left(1 - KG_k Y_k^T\right)\widehat{G}^{-}_{k|k-1},\end{aligned} \tag{10}$$

$G^{+}_{k|k}$: Posterior State Estimation error covariance.

Step 6. State estimation error computation

$$b_t = \widehat{S}^{+}_{k|k} - \varnothing \widehat{S}^{+}_{k-1|k-1}, \tag{11}$$

b_t : system state estimation errors.

Step 7. Update state process covariance matrix Q_k

$$\widehat{b} = \frac{1}{M}\sum_{i=1}^{M} b_{k-i+1}, \tag{12}$$

$\widehat{b}$: average system state estimation errors,

$$Q_k = \frac{1}{M}\sum_{i=1}^{M}\left\{\left(b_{k-i+1} - \widehat{b}\right)\left(b_{k-i+1} - \widehat{b}\right)^T - \frac{M-1}{M}\varnothing_{k-i+1}\widehat{G}^{+}_{k-i|k-i}\varnothing^{T}_{k-i+1} - \widehat{G}^{+}_{k-i+1|k-i+1}\right\}. \tag{13}$$

3.4. Bidirectional LTSM. RNN is a type of conventional LSTM approach with its module contains a singular neuronal structure to represent the human brain. In LSTM, the module is made up of cells that each have three gates. These two modules are organized into a chain structure. The three gates of a cell are named as input, hidden, and output, respectively, as illustrated in the following Figure 4.

The mathematical models for these gates mentioned are as follows. The input is defined as

$$x(t) = \delta(W_i i(t) + V_i h(t-1) + a_i), \tag{14}$$

in which $h(t-1)$ denotes previous gate, $i(t)$ is the current input cell, δ is the sigmoid function, and W_i and V_i are -the weights of the input gates.

$$g(t) = \delta\left(W_f i(t) + V_f h(t-1) + a_f\right), \tag{15}$$

where $g(t)$ is the forget gate in the cell, and W_f and V_f are -the weights of the forget gates.

$$\tilde{z}(t) = \tanh\left(W_c i(t) + V_c h(t-1) + a_c\right), \tag{16}$$

where $\tilde{z}(t)$ will update the memory unit, which will update the alternate information.

$$z(t) = g(t) * z(t-1) + x(t) * \tilde{z}(t), \tag{17}$$

where $z(t)$ will update the cell.

Information from the forgotten gate is merged with the updated information which results in a new state, where W_c and V_c are weights of forgotten gates and updated information, respectively, * is the Hadamard product, and a new alternative state is

$$\text{Output}(t) = \delta(W_o x(t) + V_o h(t-1) + a_o), \tag{18}$$

$$h(t) = \text{output}(t) * \tanh\left(z(t)\right), \tag{19}$$

where W_o and V_o are the weights of the output gate.

Here, equations (18) and (19) are used to calculate the output gate.

Table 3: Error rate (%).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	2.895	2.365	2.86	3.41	3.61	3.8	1.995	3.54	3.865	2.74	3.108
IDAD2DC [7]	5.035	5.705	6.24	5.21	5.325	5.17	5.695	4.825	5.4	3.96	5.2565
READP [12]	3.805	5.3	4.165	3.915	3.97	3.775	4.48	4.2	4.105	5.415	4.313
SDNAELWA [8]	4.09	3.85	3.285	3.45	2.945	4.23	3.585	2.695	2.805	2.73	3.3665
MLELMAKF	1.82	1.605	2.21	1.875	1.46	2.075	2.07	1.68	2.4	2.275	1.947

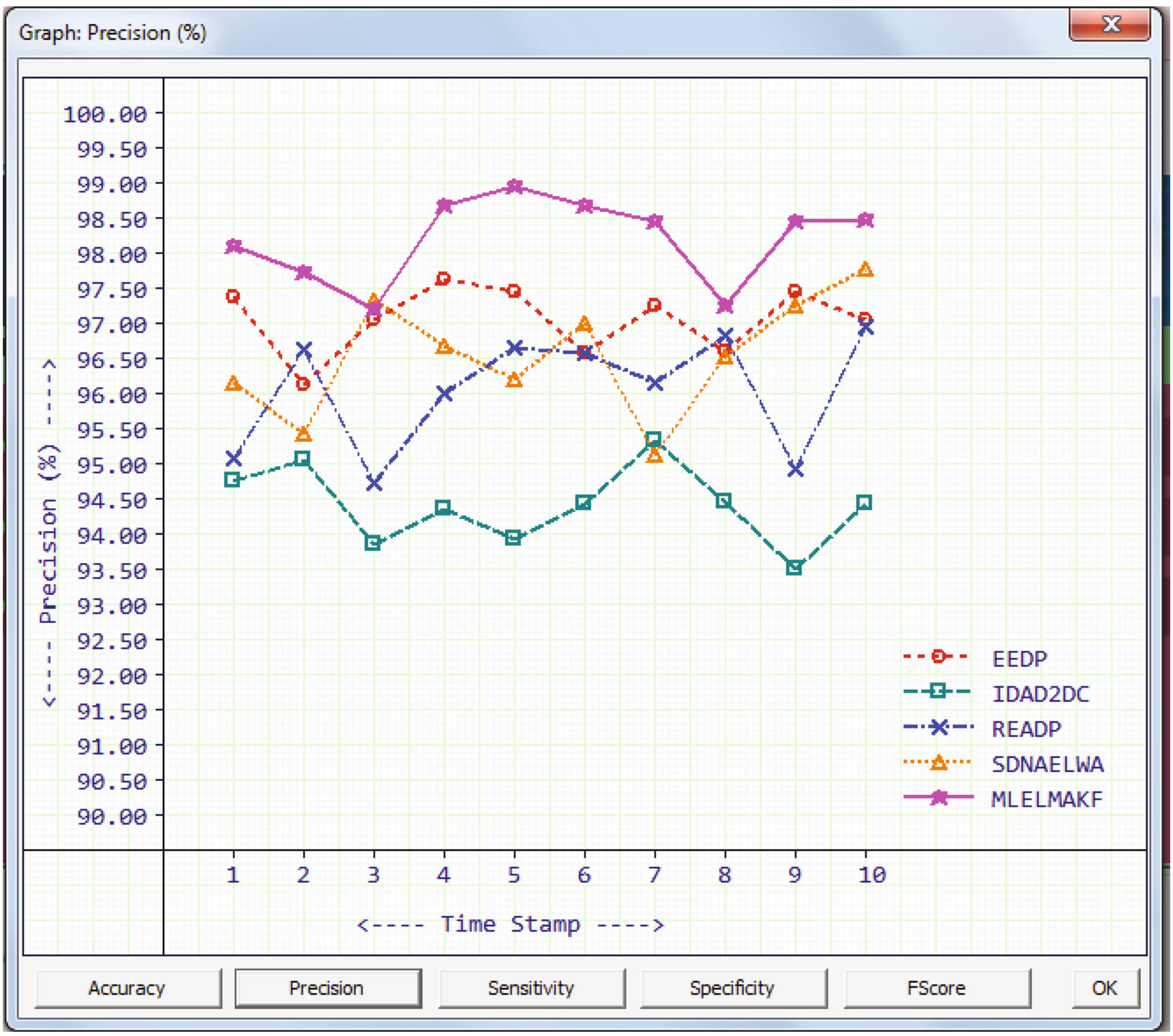

Figure 11: Precision.

The input data is processed by both forward and backward layers using activation functions, and the final output is created as a result of this processing as illustrated in Figure 5.

4. Experiment Setup

This work uses the Intel Indoor dataset [27] which comprises four types of data that are acquired using 54 nodes from Intel Research Lab, Berkeley, as shown in Figure 6. The data is divided in to four categories, viz., temperature, humidity, light, and voltage, respectively.

Mica2Dot types of sensors are used in this setup. The sensor board collects timely topological data along with temperature, light, humidity, and voltage information in every 31 seconds such that on one sensor per 31 seconds, the data was gathered from a tiny database developed on a TinyOS.

Table 4: Precision (%).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	96.25	97.84	96.96	96.93	96.22	96.04	97.97	96.01	96.01	98.04	96.827
IDAD2DC [7]	93.88	95.14	93.37	95.4	95.37	95.43	94.35	94.85	94.07	95.85	94.771
READP [12]	96.43	94.87	96.1	96.21	96.69	96.03	95.04	96.81	96.96	94.62	95.976
SDNAELWA [8]	96.03	95.68	95.28	96.1	95.96	95.76	95.43	96.95	96.58	97.17	96.094
MLELMAKF	97.62	97.46	97.4	97.31	98.02	97.48	97.97	98.44	97.31	97.41	97.642

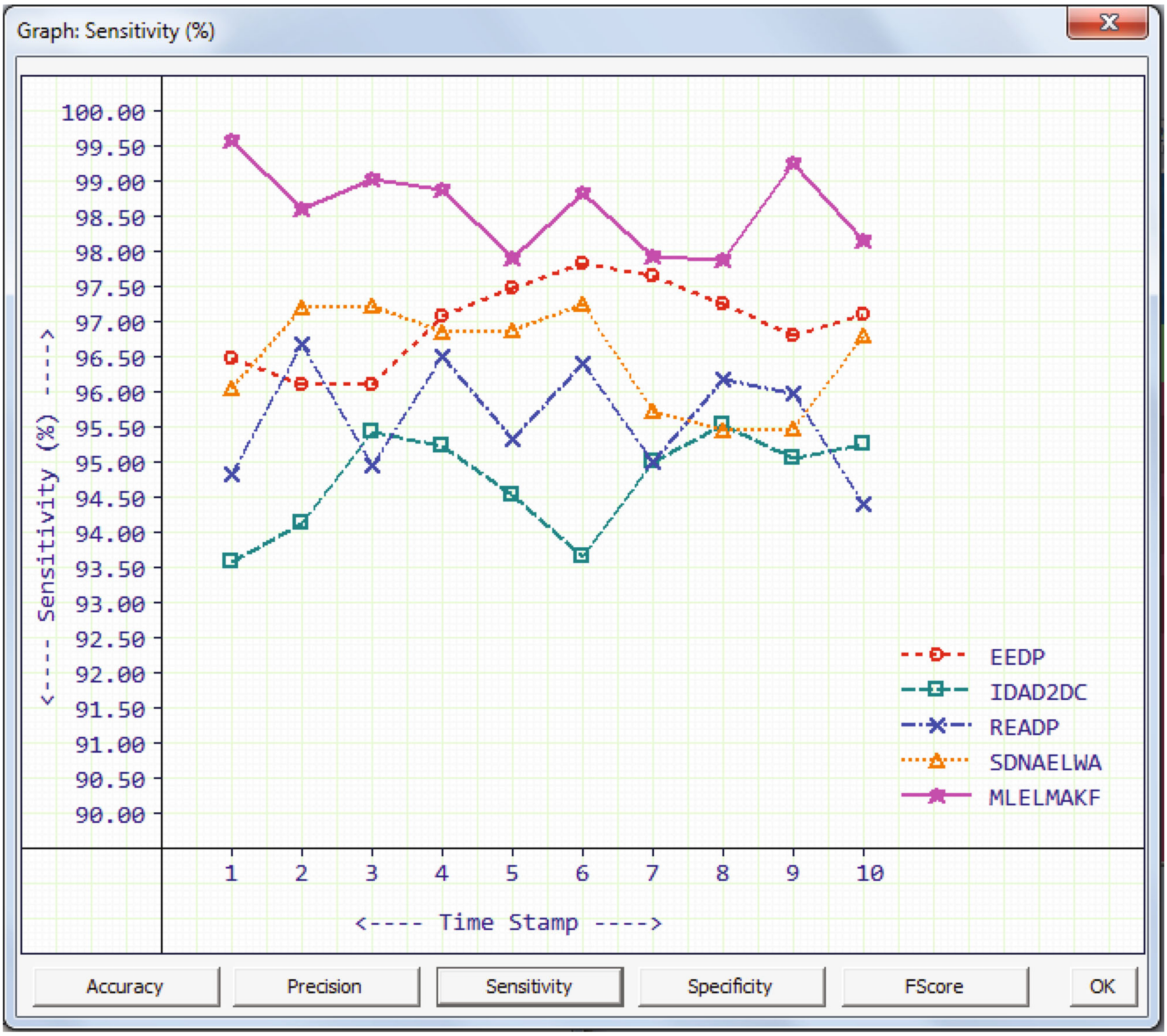

Figure 12: Sensitivity.

This dataset contains almost 2.3 million recorded values received from sensor outputs. Compressed file size is 34 MB, whereas the uncompressed file has 150 MB size. Dataset is divided and taken into account for every 6minutes. The model will analyze and provide results for every one hour. Figures 7 and 8 show the network environment setup in OPNET.

5. Results and Discussion

In this work, the following performance metrics are analyzed and proved that the propose method is the better one when compared with existing works which are EEDP, IDAD2DC, READP, and SDNAELWA.

Table 5: Sensitivity (%).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	97.9245	97.4405	97.3103	96.2753	96.5483	96.3483	98.0386	96.8819	96.2506	96.5341	96.9552
IDAD2DC [7]	95.9624	93.5589	94.104	94.2501	94.0625	94.2984	94.2652	95.4706	95.0778	96.2156	94.7266
READP [12]	95.9789	94.5485	95.5934	95.9701	95.4303	96.406	95.9612	94.8932	94.9378	94.5538	95.4273
SDNAELWA [8]	95.8001	96.5879	98.0953	96.9728	98.1086	95.7792	97.3478	97.6433	97.7827	97.3647	97.1482
MLELMAKF	98.7257	99.3172	98.1657	98.9224	99.0501	98.3554	97.8917	98.2043	97.8777	98.0276	98.4538

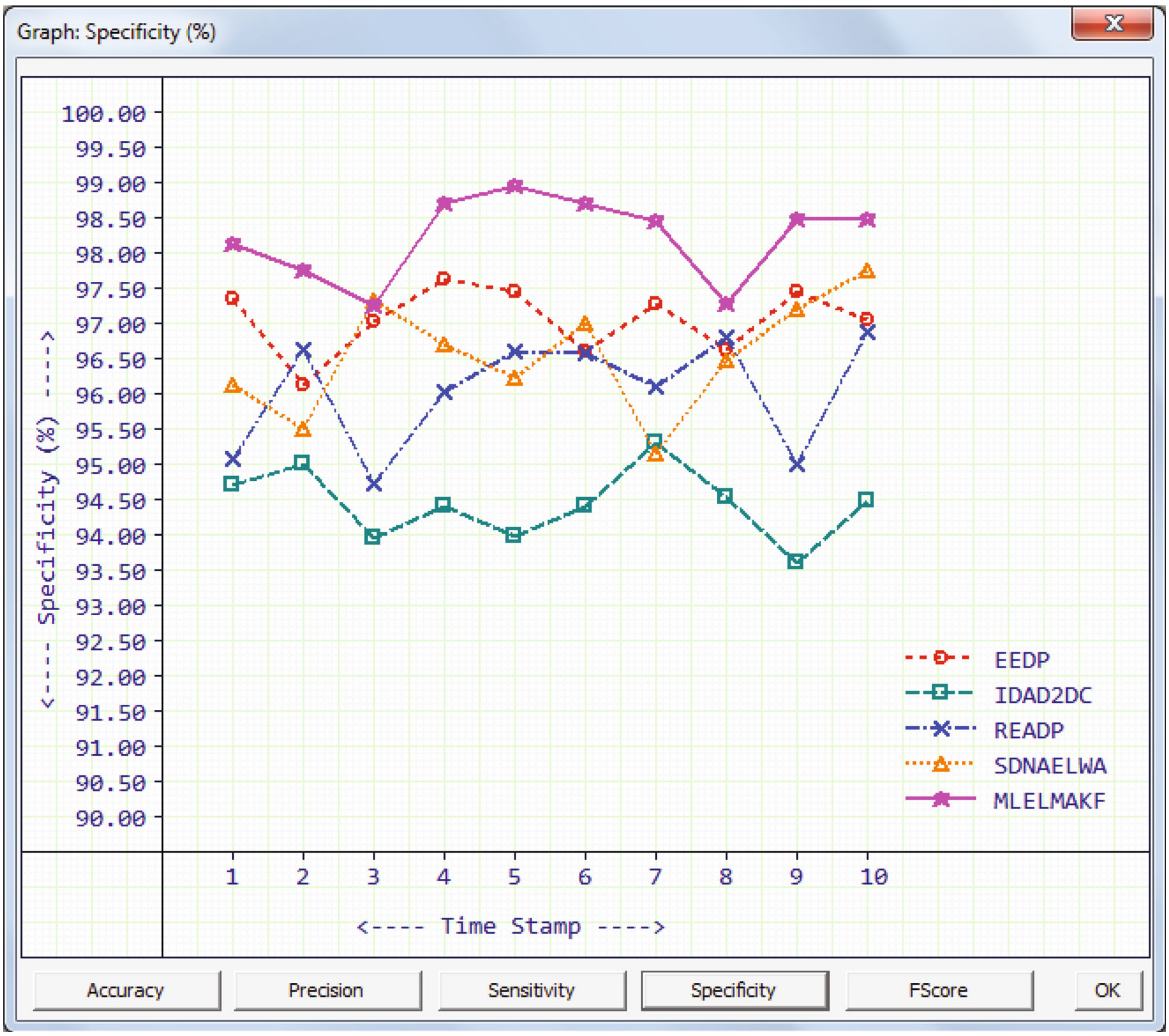

Figure 13: Specificity.

Figure 9 and Table 2 show the percentage of accuracy calculated from the given method and is compared with other existing methods. The average accuracy percentage of the proposed MLELMAKF method is 98.053. The average percentage difference of the proposed method (MLELMAKF) is improved by 1.19%, 3.43%, 2.44%, and 1.45 than the existing methods like EEDP, IDAD2DC, READP, and SDNAELWA, respectively.

Figure 10 and Table 3 show the percentage of error rate of the proposed method with the comparison of the four existing methods. The average error rate percentage of the proposed method is 1.947. The average percentage difference of the proposed method (MLELMAKF) is improved by 45.93%, 91.88%, 75.59%, and 53.42% than the existing methods like EEDP, IDAD2DC, READP, and SDNAELWA, respectively.

Table 6: Specificity (%).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	96.313	97.8311	96.9709	96.909	96.2328	96.0526	97.9714	96.0456	96.02	98.0089	96.8355
IDAD2DC [7]	94.01	95.0565	93.4213	95.3432	95.3047	95.3745	94.3449	94.8833	94.1322	95.8657	94.7736
READP [12]	96.4131	94.8525	96.0792	96.2005	96.6457	96.0454	95.0872	96.7442	96.8938	94.6162	95.9578
SDNAELWA [8]	96.0204	95.7202	95.4117	96.1348	96.0466	95.7608	95.5183	96.9715	96.6216	97.1756	96.1382
MLELMAKF	97.6464	97.5066	97.4201	97.3531	98.0404	97.5022	97.9684	98.4362	97.3255	97.4262	97.6625

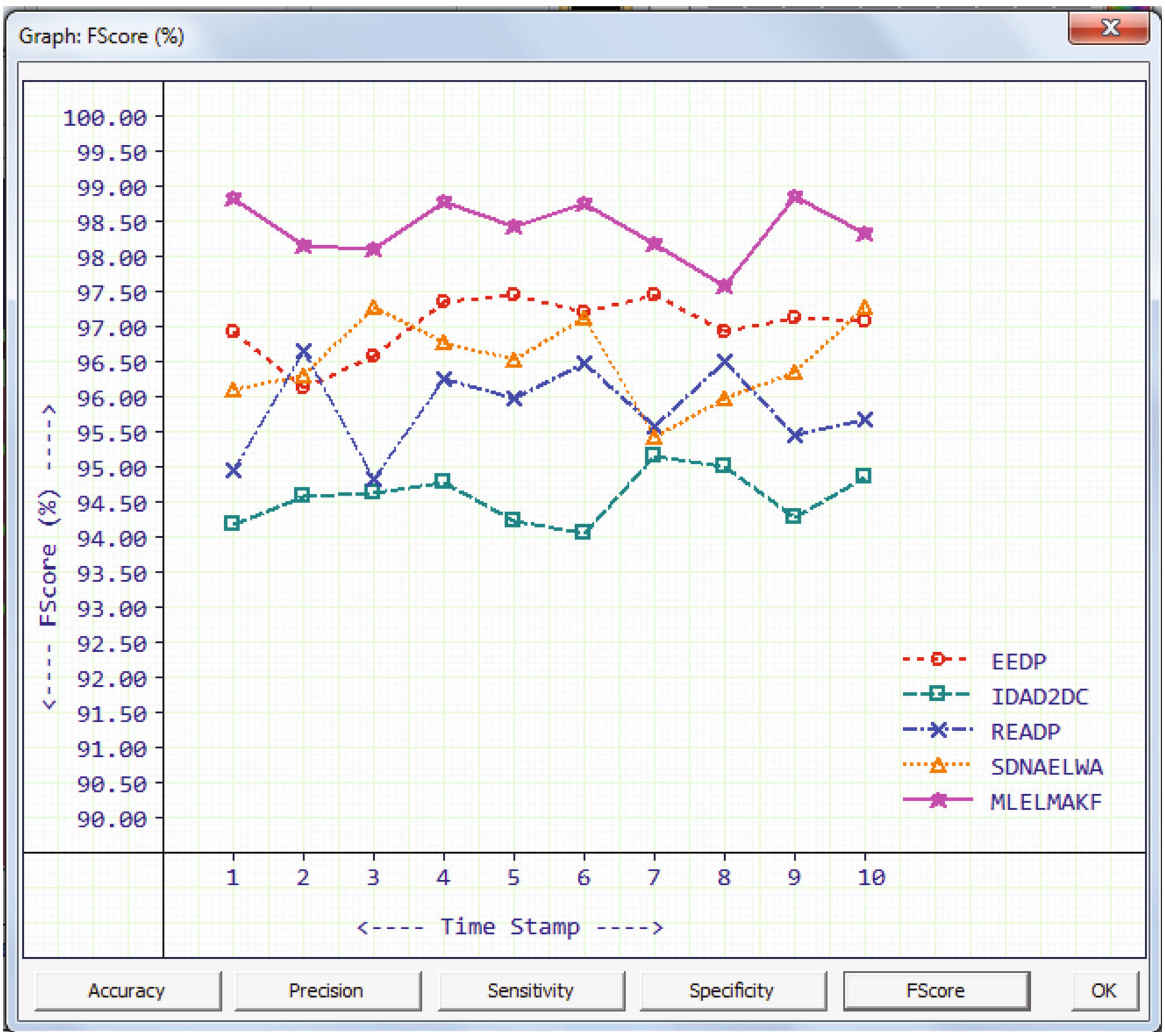

Figure 14: *F*-score.

Figure 11 and Table 4 show the precision comparisons as percentage. The average precision percentage of the proposed method is 98.053. The average percentage difference of the proposed method (MLELMAKF) is improved by 0.838%, 2.98%, 1.72%, and 1.59% than the existing methods like EEDP, IDAD2DC, READP, and SDNAELWA, respectively.

Figure 12 and Table 5 show the percentage of sensitivity of the proposed method with the comparison of the four existing methods. The average sensitivity percentage of proposed method is 98.053. The average percentage difference of the proposed method (MLELMAKF) is improved by 1.53%, 3.85%, 3.12%, and 1.33% than the existing methods

Table 7: F-score (%).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	97.08	97.6398	97.1349	96.6016	96.3839	96.1939	98.0043	96.444	96.1302	97.2812	96.8894
IDAD2DC [7]	94.9098	94.3428	93.7356	94.8216	94.7118	94.8608	94.3076	95.1593	94.5712	96.0325	94.7453
READP [12]	96.2039	94.709	95.846	96.0899	96.056	96.2176	95.4984	95.842	95.9383	94.5869	95.6988
SDNAELWA [8]	95.9149	96.1318	96.6672	96.5344	97.0224	95.7696	96.3793	97.2954	97.1776	97.2673	96.616
MLELMAKF	98.1697	98.3799	97.7813	98.1096	98.5324	97.9157	97.9308	98.322	97.593	97.7178	98.0452

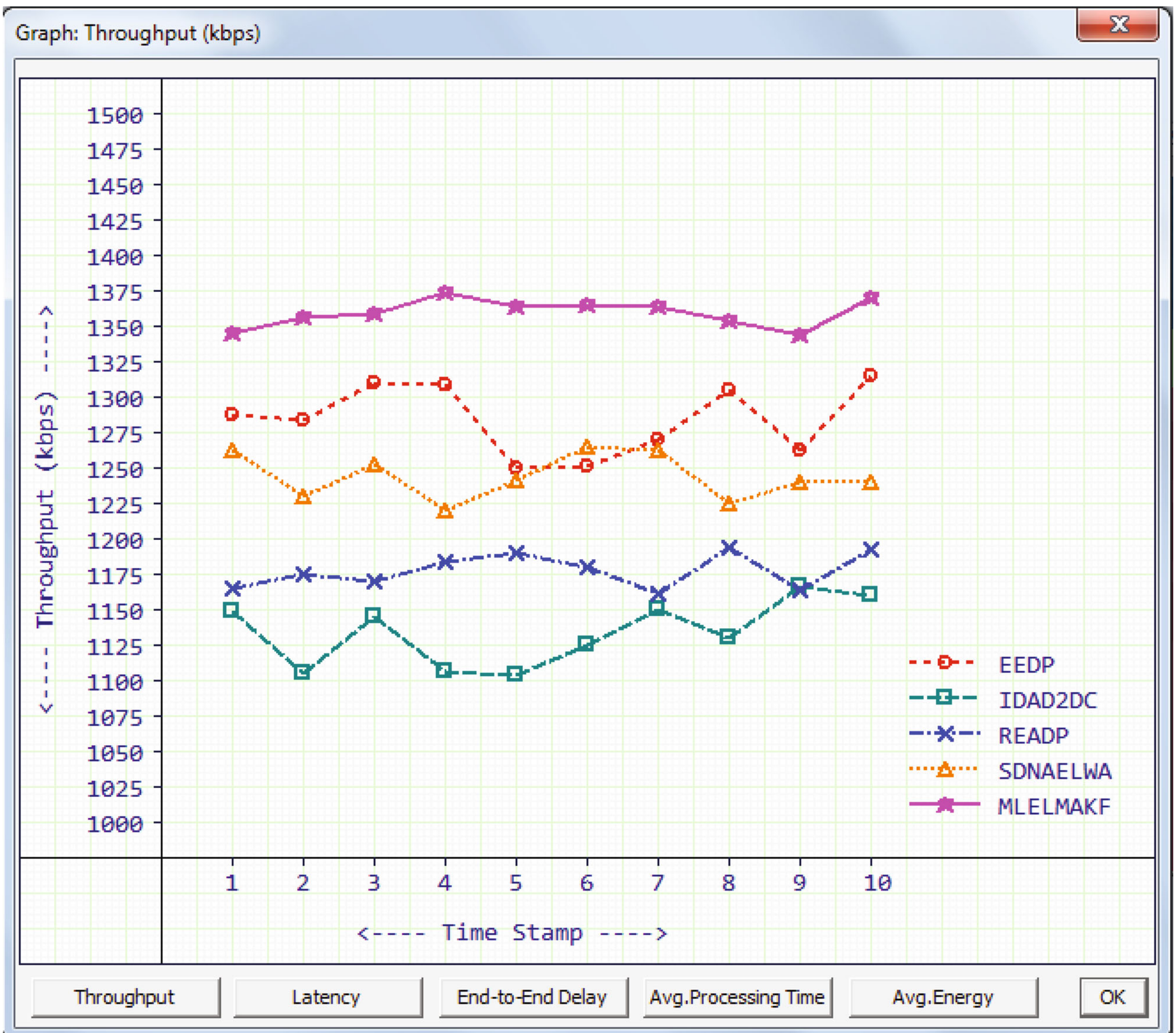

Figure 15: Throughput.

like EEDP, IDAD2DC, READP, and SDNAELWA, respectively.

Figure 13 and Table 6 show the percentage of specificity of the proposed method with the comparison of the four existing methods. The average specificity percentage of the proposed method is 97.66. The average percentage difference of the proposed method (MLELMAKF) is improved by 0.85%, 3.00%, 1.76%, and 1.57% than the existing methods like EEDP, IDAD2DC, READP, and SDNAELWA, respectively.

Figure 14 and Table 7 show the percent F-score comparisons. The average specificity percentage of the proposed method is 98.04. The average percentage difference of the proposed method (MLELMAKF) is improved by 1.18%,

Table 8: Throughput (kbps).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	1272	1251	1280	1292	1254	1288	1272	1287	1264	1280	1274
IDAD2DC [7]	1124	1107	1148	1148	1153	1131	1146	1123	1167	1152	1139.9
READP [12]	1179	1159	1189	1168	1171	1151	1165	1154	1213	1153	1170.2
SDNAELWA [8]	1253	1239	1204	1246	1249	1226	1236	1246	1213	1205	1231.7
MLELMAKF	1342	1354	1361	1375	1366	1344	1362	1369	1353	1354	1358

Figure 16: Latency.

3.42%, 2.42%, and 1.46% than the existing methods like EEDP, IDAD2DC, READP, and SDNAELWA, respectively.

Figure 15 and Table 8 show the throughput comparisons with four existing methods from which it is inferred that the proposed method delivers 1358 kbps packets which is far better compared to the values 84, 218,187, and 126 of their respective EEDP, IDAD2DC, READP, and SDNAELWA counterparts.

Figure 16 and Table 9 show that the latency of the proposed method is very low when compared to other existing methods.

Table 9: Latency (ms).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	44	52	44	40	53	46	48	45	54	47	47.3
IDAD2DC [7]	98	97	91	93	85	89	88	98	78	82	89.9
READP [12]	79	89	74	77	77	88	80	88	70	85	80.7
SDNAELWA [8]	50	58	68	56	58	60	55	60	67	69	60.1
MLELMAKF	25	24	15	16	13	21	12	14	15	21	17.6

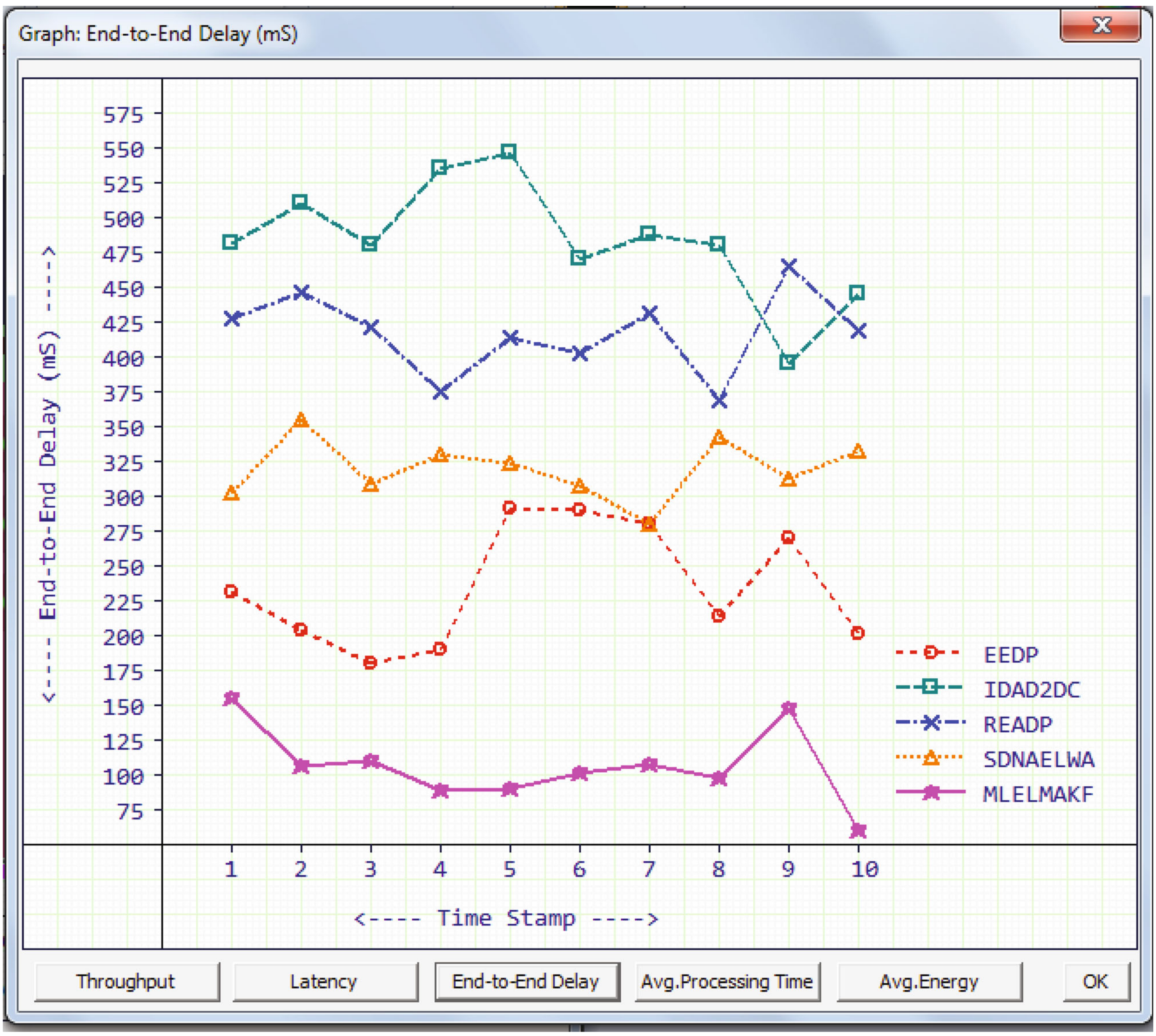

Figure 17: End-to-end delay.

Table 10: End to-end delay (ms).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	234	272	262	216	269	255	276	260	274	263	258.1
IDAD2DC [7]	495	509	471	477	469	475	444	518	423	447	472.8
READP [12]	410	453	376	397	412	470	427	458	383	450	423.6
SDNAELWA [8]	260	329	381	303	309	324	298	335	369	371	327.9
MLELMAKF	169	152	103	102	75	113	64	113	83	114	108.8

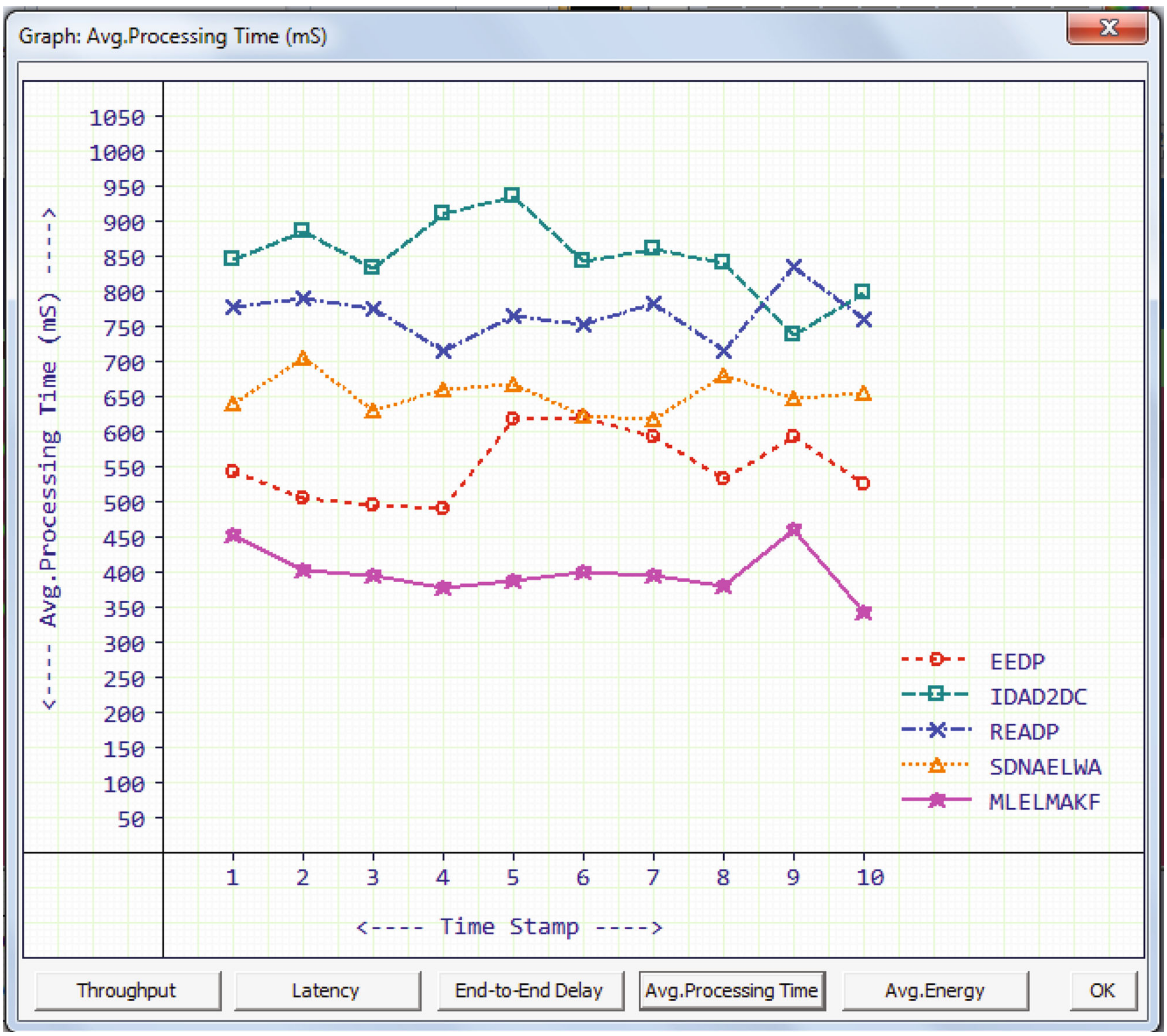

Figure 18: Average processing time.

Table 11: Average processing time (ms).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	567	599	590	522	586	586	606	583	611	583	583.3
IDAD2DC [7]	881	885	847	835	833	826	796	880	761	814	835.8
READP [12]	754	816	712	762	750	844	795	820	716	800	776.9
SDNAELWA [8]	585	673	730	647	638	660	635	680	703	708	665.9
MLELMAKF	471	465	403	405	363	410	352	391	362	424	404.6

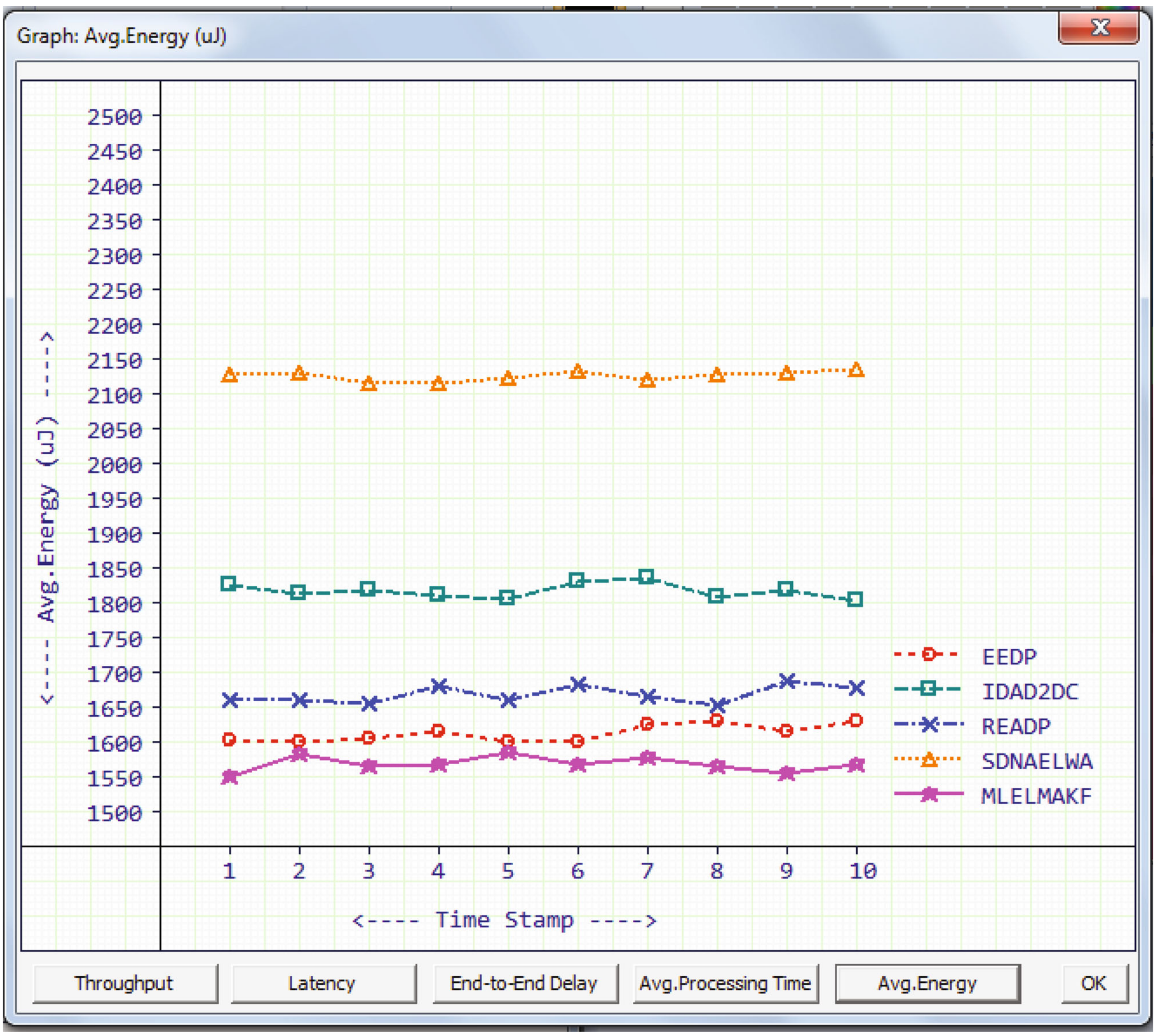

Figure 19: Average energy consumption.

Table 12: Average energy consumption (μJ).

Time stamp	1	2	3	4	5	6	7	8	9	10	Average
EEDP [5]	1627	1633	1623	1605	1623	1616	1632	1613	1628	1608	1620.8
IDAD2DC [7]	1833	1821	1825	1819	1811	1807	1824	1804	1819	1828	1819.1
READP [12]	1686	1653	1681	1674	1671	1678	1676	1682	1686	1655	1674.2
SDNAELWA [8]	2138	2107	2130	2134	2120	2112	2110	2102	2131	2118	2120.2
MLELMAKF	1561	1565	1572	1569	1579	1553	1584	1583	1562	1581	1570.9

Both Figure 17 and Table 10 show that the end-to-end delay is very low comparatively.

Figure 18 and Table 11 reveal that the average processing time is found to be very low when compared with four of the existing methods.

Figure 19 and Table 12 show that average energy consumption is significantly reduced when compared with the existing methods.

6. Conclusion

We conclude that the proposed approach significantly contributes to the development of efficient neural network architecture to perform data aggregation and prediction from the IoT enabled services in an industrial background which employs a typical wireless sensor network. After comparisons across the four existing methods, the proposed prediction model exhibits a significant performance improvement across all metrics of a neural network due to the inclusion of the three steps, viz., ELM-based redundancy removal, AKF-based noise removal, and Bi-LSTM-based prediction, respectively. The proposed approach MLELMAKF has an average accuracy rate of 98.053% which positively differs by 1.19%, 3.43%, 2.44%, and 1.45% compared to current methods, viz., EEDP, IDAD2DC, READP, and SDNAELWA, respectively. Additionally, other performance metrics such as average latency, end-to-end delay, average processing time, and average energy consumption and throughput are obtained as 17.6 ms, 114 ms, 404 ms, and 1570.9nJ and 1358 kbps, respectively. It is evident from Results that the proposed data aggregation scheme outperforms existing EEDP, IDAD2DC, READP, and SDNAELWA methods significantly. This research can be taken to next levels by adding up the number of layers in ELM and Bi-LSTM stages such that the prediction accuracy of both redundant and error data becomes better with considerable reduction in the number of computations.

Data Availability

The data used to support the finding of this research are accessed from http://db.csail.mit.edu/labdata/labdata.html.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. P. Plageras, K. E. Psannis, C. Stergiou, H. Wang, and B. B. Gupta, "Efficient IoT-based sensor big data collection-processing and analysis in smart buildings," *Future Generation Computer Systems Journal*, vol. 82, pp. 349–357, 2018.

[2] H. O. Tan and I. Korpeoglu, "Power efficient data gathering and aggregation in wireless sensor networks," *ACM SIGMOD Record*, vol. 32, no. 4, pp. 66–71, 2003.

[3] M. Liu, J. Cao, Y. Zheng, H. Gong, and X. Wang, "An energy-efficient protocol for data gathering and aggregation in wireless sensor networks," *The Journal of Supercomputing*, vol. 43, no. 2, pp. 107–125, 2008.

[4] S. Thomas and T. Mathew, "Secure data aggregation in wireless sensor network using Chinese remainder theorem," *International Journal of Electronics and Telecommunications*, vol. 68, no. 2, pp. 329–336, 2022.

[5] Y. Li, W. Liang, W. Xu et al., "Data collection maximization in IoT-sensor networks via an energy-constrained UAV," *IEEE Transactions on Mobile Computing*, pp. 1–16, 2021.

[6] A. Ahmed, S. Abdullah, M. Bukhsh, I. Ahmad, and Z. Mushtaq, "An energy-efficient data aggregation mechanism for IoT secured by blockchain," *IEEE Access*, vol. 10, pp. 11404–11419, 2022.

[7] S. Sanyal and P. Zhang, "Improving quality of data: IoT data aggregation using device to device communications," *IEEE Access*, vol. 6, pp. 67830–67840, 2018.

[8] S. Anbalagan, D. Kumar, J. Mercy Faustina, G. Raja, W. Ejaz, and A. K. Bashir, "SDN-assisted efficient LTE-WiFi aggregation in next generation IoT networks," *Future Generation Computer Systems*, vol. 107, pp. 898–908, 2020.

[9] S. S. Geetika Dhand, "Data aggregation techniques in WSN:survey," *Procedia Computer Science*, vol. 92, pp. 378–384, 2016.

[10] K. Gulati, R. S. K. Boddu, D. Kapila, S. L. Bangare, N. Chandnani, and G. Saravanan, "*A review paper on wireless sensor network techniques in Internet of Things (iot)*," *Materials Today: Proceedings*, vol. 51, pp. 161–165, 2022.

[11] V. V. Ghate and V. Vijayakumar, "Machine learning for data aggregation in wireless sensor networks: a survey," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 24, pp. 1–12, 2019.

[12] Y. Huo, C. Yong, and Y. Lu, "Re-ADP real-time data aggregation with adaptive ω-event differential privacy for fog computing'," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 6285719, 13 pages, 2018.

[13] I. Ullah and H. Y. Youn, "Efficient data aggregation with node clustering and extreme learning machine for WSN," *The Journal of Supercomputing*, vol. 76, no. 12, pp. 10009–10035, 2020.

[14] X. Du, Z. Zhou, Y. Zhang, and T. Rahman, "Energy-efficient sensory data gathering based on compressed sensing in IoT

networks," *Journal of Cloud Computing*, vol. 9, no. 1, p. 19, 2020.

[15] M. Elshrkawey, S. M. Elsherif, and M. E. Wahed, "An enhancement approach for reducing the energy consumption in wireless sensor networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 2, pp. 259–267, 2018.

[16] A. Upadhyay, R. Kumar, and S. K. Tiwari, "Modified LEACH protocol for sensor network," in *1st International Conference on Futuristic trend in Computational Analysis and Knowledge Management (ABLAZE-2015)*, pp. 301–304, Greater Noida, India, 2015.

[17] S. Ding, X. Xu, and R. Nie, "Extreme learning machine and its applications," *Neural Computing and Applications*, vol. 25, no. 3-4, pp. 549–556, 2014.

[18] J. Guo, W. Huang, and B. Williams, "Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 50–64, 2014.

[19] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, "Convolutional LSTM network: a machine learning approach for precipitation nowcasting," *Advances in Neural Information Processing Systems*, vol. 1, pp. 802–810, 2015.

[20] C. Baziotis, N. Pelekis, and C. Doulkeridis, "'Data stories at Sem Eval-2017 Task 4: deep LSTM with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop on Semantic Evaluations (Sem Eval-2017)*, pp. 747–754, Vancouver, Canada, 2017.

[21] Y. Liu, K. Lan, F. Huang, X. Cao, B. Feng, and B. Zhu, "An aggregate store sales forecasting framework based on conv LSTM," in *The 5th international conference on compute and data analysis (ICCDA 2021)*, pp. 67–72, New York, NY, USA, 2021.

[22] H. Cheng, Z. Xie, Y. Shi, and N. Xiong, "Multi-step data prediction in wireless sensor networks based on one-dimensional CNN and bidirectional LSTM," *IEEE Access*, vol. 7, pp. 117883–117896, 2019.

[23] G. Said, A. Ghani, A. Ullah, M. Azeem, M. Bilal, and K. S. Kwak, "Light-weight secure aggregated data sharing in IoT-enabled wireless sensor networks," *IEEE Access*, vol. 10, pp. 33571–33585, 2022.

[24] X. Liu, J. Yu, F. Li, W. Lv, Y. Wang, and X. Cheng, "Data aggregation in wireless sensor networks: from the perspective of security," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6495–6513, 2019.

[25] L. Zhou, C. Ge, S. Hu, and C. Su, "Energy-efficient and privacy-preserving data aggregation algorithm for wireless sensor networks," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3948–3957, 2019.

[26] S. Kumar and V. K. Chaurasiya, "A strategy for elimination of data redundancy in internet of things (IoT) based wireless sensor network (WSN)," *IEEE Systems Journal*, vol. 13, no. 2, pp. 1650–1657, 2018.

[27] "Intel Berkeley Research lab," http://db.csail.mit.edu/labdata/labdata.html.