Leveraging LSTM and Reinforcement Learning for Adaptive Sensing in CIoT Nodes

Sushmita Ghosh, Siamak Layeghy, Swades De, Shouri Chatterjee, and Marius Portmann

***Abstract*—Wireless sensor networks (WSNs) offer a broad range of applications in the Consumer Internet of Things (CIoT). The sensor nodes in a WSN are equipped with an array of sensors that often encounter limited energy availability. Thus, a joint Long Short-Term Memory (LSTM) and reinforcement-learning-based edge intelligence framework is proposed in this article for a multi-sensing node. This novel strategy aims to estimate an optimal set of active sensors during a measurement cycle by solving the trade-off between the cross-correlation between sensing signals and sensors' energy consumption using a Q-learning-based optimization function at the edge node. A prediction model based on LSTM is employed to predict the sensing signals monitored by the inactive sensors from the cross-correlated sensing signals monitored by the active sensors. To assess the performance of the proposed framework in CIoT nodes, the algorithm is simulated on an air pollution monitoring dataset. The simulation results confirm the effectiveness and efficiency of the proposed framework. In comparison to the current state-of-the-art approach, the proposed algorithm shows a 13% improvement in error performance and up to 27% improvement in sensing energy consumption, while maintaining a lower bound of cross-correlation coefficient between the inactive and active sensor set.**

***Index Terms*—Consumer Internet of Things (CIoT), Edge intelligence framework, Temporal correlation, long and short-term memory (LSTM), cross-correlation, Q-learning, adaptive sensing.**

I. Introduction

Wireless sensor networks (WSNs) offer multitude of applications in Consumer Internet of Things (CIoT) devices, such as smart wearables [1], smart healthcare systems [2], smart agriculture [3], Environmental monitoring devices [4], etc. with improved functionality, efficiency, and convenience. Such WSNs comprise numerous sensor nodes, primarily powered by batteries, which impose restrictions on the network's operational lifespan. The primary energy-intensive aspects within WSNs are the processes of sensing and data transmission. A sensor node encompasses a collection of sensors tasked with monitoring diverse environmental parameters. For instance, an air quality monitoring node incorporates sensors to monitor various aspects such as temperature, relative humidity, particulate matter (PM), and potentially harmful gases. It is important to highlight that the power consumption of these high-performance sensors can exceed the power consumption of the transmission module. To tackle the challenge of short battery capacity, these sensor nodes are equipped with energy-harvesting mechanisms to replenish their batteries. To minimize energy usage during sensing, an ideal sampling interval can be calculated for each parameter without compromising the data quality of data [5]. Despite the extension of WSN lifetimes through such mechanisms, the development of a sustainable wireless sensor node capable of efficiently monitoring multiple sensing signals with reduced energy consumption and high sensing quality remains a significant ongoing challenge.

In numerous WSNs, the field data is regularly transmitted to a nearby collection point, facilitating the analysis and establishment of a feedback mechanism for system parameter adaptation [6]. If multiple parameters of interest are sensing within an environment, even if the individual sensing elements differ significantly in their composition, these parameters frequently exhibit interrelationships or cross-correlations among them. This inherent connection can be leveraged to estimate the value of one parameter using data from one or more other parameters. Consequently, by carefully selecting the appropriate number of active sensors, it becomes possible to design an energy-efficient sensing strategy for the field node.

Within densely deployed WSNs, the monitored parameters often exhibit spatio-temporal correlations, which can be effectively utilised to eliminate redundancy. In such scenarios, it is possible to strategically activate only a few sensor nodes to monitor slowly changing signals [7]. Conversely, in sparsely deployed WSNs, commonly found in controlled smart city deployments, spatial data correlation is relatively minimal. In these cases, redundancy primarily occurs at the level of individual nodes due to the temporal correlation of the sensing signals and cross-correlation among such time series sensing signals. Consequently, the optimization of sensing in such scenarios is limited to the node level. This research specifically concentrates on such sparse deployment scenarios, to eliminate sensing redundancy at the node level.

Since the spatio-temporal and cross-correlation exploitation mechanism is computationally intensive, the complex algorithm can be programmed at the edge node. The edge node can transmit the output of the algorithm to the end nodes in the WSN to optimally operate the sensors. The battery operated field deployed sensor nodes have limited energy capacity. In contrast, the edge node is typically a high-end computing node, powered from the grid. Therefore, to reduce the energy consumption and thereby increase the lifetime of the field deployed sensor nodes, the processing complexity can be shifted to the edge node.

S. Ghosh is with the UQ-IITD Academy, Indian Institute of Technology Delhi, New Delhi, India (e-mail: sushmita.ghosh@uqidar.iitd.ac.in)

S. De and S. Chatterjee are with the Department of Electrical Engineering, Indian Institute of Technology Delhi, New Delhi, India (e-mail: swadesd@ee.iitd.ac.in, shouri@ee.iitd.ac.in)

S. Layeghy and M. Portmann are with the School of EECS, University of Queensland, Brisbane, 4072, QLD, Australia (e-mail: siamak.layeghy@uq.net.au, marius@ieee.org)

A. Related Work

A few studies in the existing literature have explored the concept of energy-efficient sensing in CIoT applications. In one instance, an adaptive sampling method was introduced for a snow monitoring application [8]. This algorithm dynamically determines the optimal sampling frequencies for sensors by employing the fast Fourier transform on a large dataset to compute the maximum detectable frequency of the sensed signal. The optimal sampling interval is decided from the Nyquist sampling rate, which is computed from the maximum frequency of the signal. After a specified number of samples have been collected, the sampling rate is updated/recalculated based on the change in the frequency.

Another study [5] presented three distinct data collection approaches for temperature and humidity sensing, each tailored to adapt the sampling rate to changing environmental conditions. The first method involved calculating the T-statistic value (which is the ratio of the variance determined from the collected temporal samples) using Bartlett test and one-way Anova model. This value is then used in a behavior function to adapt the sampling rate. The second and third models utilised the Jaccard similarity function and the Euclidean distance function, respectively, to assess dissimilarities between the consecutive data sets. Subsequently, these methods yielded new parameter values that were incorporated into the behavior function to calculate updated sampling rates for both models.

Furthermore, a decentralised approach to adaptive sampling was introduced in [9], employing a Kalman Filtering-based estimation technique. This approach autonomously adjusts the sampling periodicity within a specified range based on the estimation error derived from the Kalman Filter. The work in [10], proposed a learning-based adaptive sensing strategy for a multi-parameter sensor hub. Few optimal sensors are selected to turn on for a limited period to collect data based on the Upper Confidence Bound (UCB) algorithm. The inactive sensing parameter values are predicted from the data collected by the active sensors using the Gaussian process regression (GPR)-based prediction models.

In a large network, the edge node has to serve multiple end nodes. However, due to high processing complexity at the edge node and delay constraints of the end user packets, several packets may be dropped when the deadline of the packets is expired. To improve the QoS of the end users, model-free deep reinforcement learning-based task-offloading schemes were presented in [11], [12]. Similar to [11], [12], in [13], a deep-reinforcement learning-based joint task-offloading and resource allocation scheme was presented considering the long-term energy constraints of edge computing nodes while maintaining the QoE of end users.

In [14]–[17], several machine learning and deep learning-based methodologies were introduced. An IDW-BLSTM-based sensor data prediction model was presented in [16] for predicting the air pollution parameters, specifically PM2.5 and PM10, by leveraging their spatio-temporal correlations with meteorological factors such as temperature, wind speed, and humidity.

Although a few adaptive sensing frameworks exploit the cross-correlation among various parameters of a node in the WSN, the sensor data prediction performance and energy efficiency of the network need to be improved.

B. Motivation

A comparison of the state-of-the-art literature is presented in Table I, which shows the research gap that has been addressed in the proposed framework. While the above mentioned approaches help to significantly decrease energy consumption in devices like wireless sensor nodes, there remains a critical limitation due to the finite battery capacity of the small sensor hubs, especially in extensive deployments. Moreover, implementing complex adaptive sensing algorithms significantly increases the processing energy consumption of the sensor node. This constraint imposes restrictions on the overall lifespan of these networks. Furthermore, as high-quality sensors continue to evolve, their cost and energy requirements have been on the rise. Additionally, the trend is toward equipping a single node with an increasing number of sensing elements. As a result, this expansion in the number of sensors per node leads to higher data volumes and, subsequently, elevated energy consumption for the sensor nodes themselves. Thus, *an adaptive sensing framework can be developed at the edge node to shift the computational complexity of the sensor node and minimize the energy requirement of the sensor node.*

In [5], [8], and [9], the adaptations at the node level primarily focus on individual sensing parameters, without taking advantage of the potential cross-correlations between multiple parameters. Consequently, each sensor within the network is required to collect samples independently, regardless of the status of other sensors. Conversely, the research presented in [6], [20] pertains to densely deployed WSNs aimed at monitoring the same parameter. In this context, a subset of sensor nodes is strategically chosen from various sensor nodes in the sensor network for sampling during specific time intervals. The selection process is based on the spatio-temporal correlations observed among the sensing signals and the energy consumption patterns exhibited by the sensors at different time periods. This approach is suitable because the data collected in densely deployed WSNs are often sparse due to the spatio-temporal correlations. In contrast, the dynamics of multi-parameter sensing within a sensor hub are anticipated to be distinct from those in the scenarios studied in [7], [19].

Although an adaptive sensing strategy for a multi-parameter sensor hub was presented in [10], the error performance could be improved by including the past temporal samples of the inactive sensing parameter (as presented in [16]) along with the present cross-correlated samples of the other parameters, as the time series samples exhibit a strong temporal correlation. Moreover, neural networks can perform better than machine learning models in order to predict the cross-correlated inactive sensing parameters [21].

It appears that the studies mentioned in [14]–[17], and [22] have different primary objectives than energy efficiency of a multi-parameter sensor node. In [14]–[17], the focus is on weather forecasting, while in [22], the emphasis is on sensor selection in densely-deployed networks for monitoring

TABLE I: Comparison with the state-of-the-art

State-of-the-art	System model	Sensing Parameters	Correlation exploited	Learning-based prediction model	Reinforcement learning	Sampling interval
[8], [5], [9]	node level	single	not exploited	not used	not used	adaptive
[18]	network level	single	spatio-temporal correlation	Sparse Bayesian Learning	not used	fixed
[19]	network level	multiple	spatio-temporal and cross-correlation	Sparse Bayesian Learning	not used	fixed
[10]	node level	multiple	cross-correlation	Gaussian Process Regressor	UCB algorithm	fixed
Proposed framework	node level	multiple	temporal and cross-correlation	LSTM	Q-learning	adaptive

a single parameter. Indeed, *it seems that the specific challenge of energy-efficient quality sensing while considering the variations of process dynamics of sensor hub has not been thoroughly investigated in the existing literature.* This represents a scope where further research and exploration are needed.

When dealing with a single node equipped with multiple sensors for capturing environment parameters that exhibit random variations, the previous approaches may not be accurate in selecting the optimal set of sensors for that node. In such cases, it is beneficial to consider not only data from the immediate past measurement cycle but also historical data from previous cycles. To address this, a reinforcement learning model can be developed, allowing the system to learn and make informed decisions through experience. The sensor selection problem can be formulated as a multi-armed bandit (MAB) problem based on the system's dynamics [23]. Finding the best set of sensors to activate in the next measurement cycle is analogous to selecting the optimal "arm" in the MAB framework. MAB problems are well-established in the field of reinforcement learning and find extensive application in real-time systems for the purpose of online decision-making. Several algorithms, such as Q-learning, epsilon-greedy, Thompson sampling, UCB, etc. have been employed to solve MAB problems [23]. UCB is a deterministic algorithm that is based on the exploration policy and hence takes more time to estimate the probability function of rewards associated with each arm. Whereas, Q-learning is a model-less Reinforcement learning method, where a model of the environment is not required. It can handle problems by estimating the stochastic transitions and rewards automatically without requiring adaptations [23]. Moreover, the Q-learning algorithm settles faster than UCB. Thus, the Q-learning algorithm is adopted in the proposed framework to address the sensor selection challenge. This approach enables the system to iteratively improve its sensor selection strategy over time, enhancing the efficiency of sensor data collection.

Although a few sensing signal recovery schemes, such as Sparse Bayesian Learning (SBL) [18], [19] and GPR [10], [24], are reported in the literature, LSTM has better prediction performance [21]. SBL method is used to reconstruct the spatially varying sparse signal in a densely deployed WSN [18], [19]. In contrast, the proposed system model considers a single sensor hub with an array of sensors deployed in a geographical location, which is unaware of spatial variations of the sensing signals. Although GPR-based prediction model performs better than SBL, as discussed in [24], the prediction accuracy can be improved further using neural networks such as LSTM [16]. A cross-correlation based sensor selection model was presented in [19] for a multi-sensing node, however, the optimality of choosing the sensor set was not studied in that framework. An UCB-based sensor selection framework was proposed in [24], however, the reward achieved with the UCB-algorithm is lower than Q-learning. Therefore, as an advance an efficient deep learning-based prediction model to achieve better sensing accuracy with Q-learning-based sensor selection strategy is proposed in this framework.

C. Contributions

In this research endeavor, a novel edge intelligence-based approach to sensor data collection is introduced for applications, leveraging deep learning techniques for a compact sensor hub equipped with multiple sensors monitoring diverse environmental parameters. The inspiration for this approach stems from the impressive predictive capabilities of Long Short-Term Memory (LSTM) models [16]. The proposed method investigates an energy-aware sensing mechanism that dynamically activates a specific set of sensors in real time. This strategy is designed to optimize the utilisation of the sensor resources, thereby conserving energy while maintaining accurate data collection. This approach is comprehensively examined at the node level, and its applicability extends to distributed networks, offering potential benefits in terms of energy efficiency and data accuracy across a broader system.

The main features and contributions of this study are:

- The proposed edge intelligence-based sensor selection framework is conceptualised as an MAB problem. It leverages the interconnections and cross-correlations observed among the diverse sensing parameters within a node to intelligently activate a few optimal sensors to sense the environment. This decision-making process is adapted to account for the dynamic and non-stationary nature of the environment being monitored.
- The Q-learning algorithm is implemented to determine the activation of the optimal sensor set. In this process, rewards are calculated during each measurement cycle by taking into account several factors. These factors include the cross-correlation among various sensing parameters, the energy requirements for sensing, and the current energy available at the node. The Q-learning algorithm uses these reward calculations to make decisions about

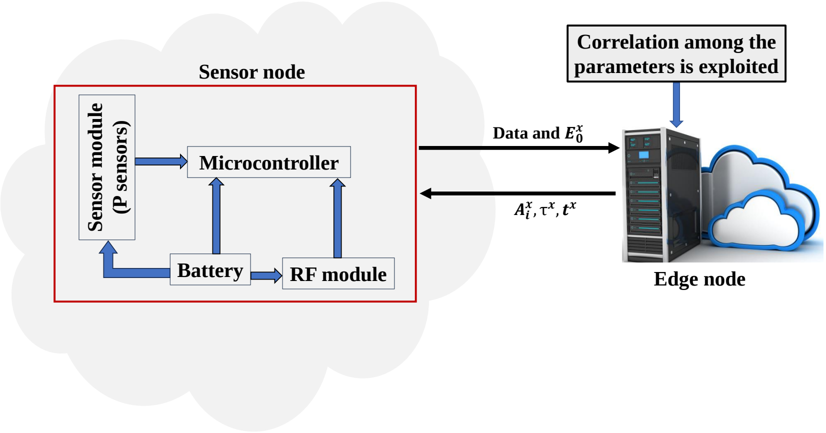


Fig. 1: System model.

which sensors to activate during each cycle to balance between data quality and energy conservation.

- The proposed approach involves utilising an LSTM network to predict missing parameters at the edge node. This prediction is achieved by analyzing the sampled parameters and their cross-correlations. The LSTM network is trained to make accurate predictions by learning from the relationships and patterns present in the data, enabling it to fill in missing parameter values effectively.
- The proposed algorithm is tested on an air pollution monitoring dataset. It achieves a 13% improvement in error performance and up to 20% improvement in sensing energy consumption compared to the nearest state-of-the-art approach, while maintaining a lower bound of cross-correlation coefficient between the inactive and active sensor set.

Paper Organisation: Section II describes the system model. Section III describes the LSTM-based prediction model, followed by the proposed Q-learning-based sensor selection strategy in Section IV. Section V presents the results and discussion, followed by the conclusion in Section VI.

Notations: A set is denoted as $\mathcal{A}$ and a matrix is denoted as $\underline{\mathbf{A}}$, where $\underline{\mathbf{A}} \in \mathbb{R}^{M\times M}$ denotes a $M \times N$ real valued matrix, and $\mathbf{A} \in \mathbb{R}^{M\times 1}$ denotes a vector with M elements. The cardinality/number of elements of set $\mathcal{A}$ is denoted as $|\mathcal{A}| = A$.

II. System Model

Fig.1 depicts a sensor node with multiple sensors integrated or assembled to the node. The sensor node is wirelessly connected to an edge node. The sensor module of the node equipped with P number to sense P distinct parameters in the environment. The microcontroller receives data from the sensors at every sampling instant for a period called the measurement cycle and transmits the data through the RF module to the edge node after every measurement cycle. The sensor node also transmits E_0^x, which denotes the energy available in battery at the end of x^{th} measurement cycle. Let $\mathcal{P} = \{P_p; 1 \leq p \leq P\}$ be the set of parameters. $\mathbf{z}^x \in \mathbb{R}^{P\times 1}$ denotes the measurement vector at a sampling instant containing the observations of each parameter in the x^{th} measurement cycle. Let I^x be the number of sampling instances in x^{th} measurement cycle. Thus, $\underline{\mathbf{Z}}^x \in \mathbb{R}^{P\times I^x}$ denotes the data matrix of the x^{th} measurement cycle.

Given that the sensors in the node are responsible for monitoring signals that change over time within the same environment, few parameters exhibit good cross-correlations. This allows the system to predict one parameter based on its correlations with other parameters. Thus, $N = (2^P - 2)$ is the total count of sensor sets that can be formed from P sensors, not including the empty set and the universal set. The i^{th} active set of sensors in the x^{th} measurement cycle is denoted as, $\mathcal{A}_i^x$. The respective inactive or sleep sensor set is denoted as $\mathcal{B}_i^x = \mathcal{P} - \mathcal{A}_i^x; 1 \leq i \leq N$. Therefore,

$$\mathcal{A}_i^x = \{P_{\mathcal{A}_i^x,m}; 1 \leq m \leq A_i^x\} \tag{1}$$

and

$$\mathcal{B}_i^x = \{P_{\mathcal{B}_i^x,k}; 1 \leq k \leq B_i^x\} \tag{2}$$

Let

$$\mathcal{S}^x = \{(\mathcal{A}_i^x, \mathcal{B}_i^x); 1 \leq i \leq N\} \tag{3}$$

be the set that contains all the active and inactive sensor sets, and $|\mathcal{A}_i^x| = A_i^x$, $|\mathcal{B}_i^x| = B_i^x$, $|\mathcal{S}^x| = N$, and $A_i^x + B_i^x = P$.

The sensing model described here is employed within the proposed framework outlined in Section IV. In this framework, the sensors belonging to the active set gather samples over a specific duration known as the measurement cycle. These collected samples are then stored in the node's memory. After the measurement cycle, the node transmits the accumulated data to the edge node. The intelligent adaptive sensing algorithm is programmed at the edge node, where the data is subsequently processed using the algorithm detailed in Section IV-C, which leads to the generation of feedback. As depicted in Fig. 1, the feedback in x^{th} measurement cycle consists of the active sensor set ($\mathcal{A}_i^x$), length of x^{th} measurement cycle (τ^x), and the sampling interval of the active sensors (t^x). This feedback is then sent back to the node, informing it of necessary adaptations to improve sensing quality and energy efficiency in subsequent cycles.

In the context of this study, it is essential to note that, in a given environmental scenario, the average number of communication exchanges and the associated energy requirements are relatively stable. Additionally, it is observed that the energy consumption for sensing in wireless nodes involved in environmental monitoring is notably high and can often dominate over communication-related energy costs [8]. Given the primary focus on optimizing sensing energy and quality through the exploitation of cross-correlations among parameter values, this study does not take into account the communication cost in its analysis.

III. Proposed LSTM-based Prediction Model

A Q-learning-based optimization method, discussed in Section IV, is used to choose an optimal active sensor set. According to the selected optimal set, certain sensors from the sensor array are to be turned on, while the remaining sensors data are predicted using the LSTM-based regressor model. This prediction is based on the active sensing parameters and the previous temporal values of the inactive parameter, as described in this section. LSTM is a variant of recurrent neural network (RNN) architecture that aims to address the shortcomings of conventional RNNs by better capturing and understanding distant relationships within sequential data. LSTMs are particularly effective in tasks involving sequential or time-series data, which makes it appropriate for predicting time series sensor data in this framework.

LSTM networks consist of various components that work together to process and learn from sequential data while addressing the vanishing gradient problem that often plagues traditional RNNs. The key components of an LSTM cell are:

- Cell State (c_t): This is the core memory of the LSTM. It runs linearly through the cell with minor interactions, which helps information to persist over long sequences. Information can be added to or removed from the cell state through various operations.
- Hidden State (h_t): The hidden state carries information about the previous time step and is used to make predictions and decisions. It can be thought of as the output of the LSTM cell.
- Input Gate (i_t): This gate controls whether and to what extent new information has to be stored in the cell state. It considers both the current input and the previous hidden state, and the output is a value between 0 and 1 for each component of the cell state.
- Forget Gate (f_t): This gate determines what information should be removed or forgotten from the cell state. It also considers both the current input and the previous hidden state, and gives output between 0 and 1 for each component of the cell state.
- Output Gate (o_t): The output gate decides what the next hidden state should be. It combines the current input and the previous hidden state to produce a value between 0 and 1 for each component of the cell state.
- Candidate Cell State: This is a new candidate value that could be added to the cell state. It is calculated based on the current input and the previous hidden state. The input gate determines how much of this candidate value gets added to the cell state.

The forget gate identifies which data from the prior cell state to omit, while the input gate determines what fresh information should be incorporated into the cell state. The candidate cell state generates a potential value for inclusion in the cell state. The update cell state function merges the outcomes of the forget gate, input gate, and candidate cell state to refresh the cell state. Ultimately, the output gate decides the output according to the refreshed cell state, which can serve as the hidden state for the present time step.

The LSTM architecture's ability to learn long-term dependencies and manage information flow through the cell state and gates makes it well-suited for various sequential tasks. It addresses the vanishing gradient problem by allowing information to flow more easily through the cell state, enabling the network to remember information over long sequences [15], [16]. Although GPR-based prediction model performs better than SBL, as discussed in [24], the prediction accuracy can be improved further using LSTM [16].

Input to the LSTM network is the design matrix $\underline{\mathbf{Z}}$, consists of n feature vectors. To predict p^{th} inactive parameter, each feature vector consists of the active parameters and one step previously collected/predicted value of the p^{th} inactive parameter. Consider $\mathbf{y}$ as the target vector with n number of target values of the p^{th} parameter from the training set. Thus, N sub-models can be created for P active sets $\mathcal{A}_i^x$ to predict the corresponding parameters of the inactive set $\mathcal{B}_i^x$. Intuitively, the temporal parameter has more significance in prediction, than the cross-correlated parameters. However, the cross-correlated parameters give information about the present condition of the environment.

As shown in Fig. 2, the n^{th} feature vector of the training matrix of the inactive parameter $k \in \mathcal{B}_i^x$ is:

$$\mathbf{z}^x_{\mathcal{A}_i}(n) = \{z^x_{\mathcal{A}_i(1)}(n), z^x_{\mathcal{A}_i(2)}(n), \cdots, z^x_{\mathcal{A}_i(A_i^x)}(n), z^x_{\mathcal{B}_i(k)}(n{-}1)\} \tag{4}$$

and the output is $z^x_{\mathcal{B}_i(k)}(n)$. Thus, the target vector is:

$$\mathbf{y}^x_{\mathcal{B}_i(k)} = \{z^x_{\mathcal{B}_i(k)}(1), \cdots z^x_{\mathcal{B}_i(k)}(2), \cdots, z^x_{\mathcal{B}_i(k)}(n)\} \tag{5}$$

The sub-models are trained at the beginning using n training samples. The i^{th} sub-model has B_i^x number of regressors to predict each parameter of $\mathcal{B}_i^x$.

IV. Proposed Sensor Selection Framework

As discussed in Section III, a set of sensors is selected to activate out of P sensors in the next measurement cycle based on the cross-correlation among all the sensors and the sensing energy consumption of the active sensors. Section IV-A describes the proposed Q-learning-based active sensor set selection framework. Each active sensor collects data at a certain optimum sampling interval decided based on the time series correlation of that sensing signal. The active sensors collect data for an optimum length of measurement cycle which is also decided based on the temporal correlation of the parameters, described in Section IV-B.

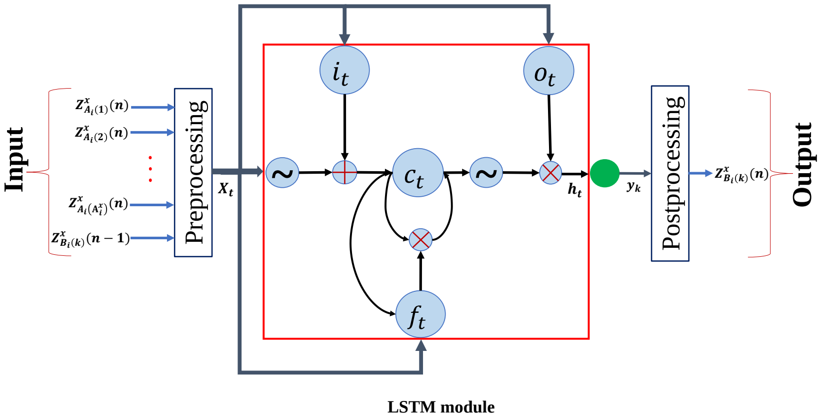


Fig. 2: LSTM-based prediction model.

A. *Q-learning-based optimization Framework*

As discussed in Section II, sensing parameters of a node experience the same environmental condition, hence exhibiting good cross-correlation. Therefore, few parameters can be predicted from the other parameters without turning On those sensors for a cycle. Since the environment is non-stationary, the optimum active sensor set changes over the period. Thus a set is selected at the end of x^{th} measurement cycle to collect data at $(x+1)^{th}$ measurement cycle. From the dataset collected at $(x+1)^{th}$ measurement cycle, a new set is selected to collect data at $(x+2)^{th}$ measurement cycle. With P sensors $N = 2^P - 2$ states can be created. At each state, $N = 2^P - 2$ actions can be taken, where each action refers to a unique vector containing the information of the set of active sensors for the next cycle. By taking action at a particular state, the system reaches the next state and gets some reward based on that action. Therefore, a reinforcement learning model can be created to choose an optimal action that maximises the reward of the system. According to [23], Q-learning is an algorithm in reinforcement learning operates without a model, that learns the value associated with taking a specific action in a given state. Thus, a Q-learning-based optimization framework, which is a model-less reinforcement learning framework, is proposed to find an optimal set of nodes based on the system parameters.

The reward of the system is computed based on the following performance parameters:

1) Cross-correlation factor of the i^{th} sensor set: Let the cross-correlation coefficient between p and q is denoted as $c(p,q)$. Thus, p and q can be called as cross-correlated if $|c(p,q)| \geq c_{th}$, where c_{th} denotes the threshold of cross-correlation coefficient. As discussed in Section II, the total count of active-inactive sensor sets that can be formed from P sensors is $N = (2^P - 2)$. Thus, $\mathcal{A}^x_i$ and $\mathcal{B}^x_i$ are cross-correlated if the sensing parameters of $\mathcal{B}^x_i$ is cross-correlated with the sensing parameter of $\mathcal{A}^x_i$. Thus, the cross-correlation factor C^x_i, which is the average cross-correlation between $\mathcal{A}^x_i$, is:

$$C^x_i = \frac{1}{A^x_i B^x_i} \sum_{k=1}^{B^x_i} \sum_{m=1}^{A^x_i} |c^x(m,k)|; \forall k \in \mathcal{B}^x_i \text{ and } m \in A^x_i \quad (6)$$

If $C^x_i < c_{th}$, C^x_i is assigned as 0.

2) Sensing energy consumption of $\mathcal{A}^x_i$: The energy consumed by the sensors of the set $\in \mathcal{A}^x_i$ is:

$$E^x_i = \sum_{m=1}^{A^x_i} M^x_m En_m; \forall m \in \mathcal{A}^x_i \quad (7)$$

M^x_m is denoted as the number of samples measured by the m^{th} parameter of $\mathcal{A}^x_i$ at x^{th} cycle and En_m is the energy consumed by the m^{th} sensor for collecting one sample.

3) Residual node energy: To introduce energy awareness into the optimization framework, the remaining energy is also taken into account as an additional performance metric to estimate the optimal sensor set. If E_{batt} is the battery capacity connected with the sensor node and E^x_0 is available node energy at the end of x^{th} measurement cycle, the normalised energy available at the sensor node is:

$$\lambda^x \triangleq \frac{E^x_0}{E_{batt}} \quad (8)$$

The objective is to maximise C^x_i, λ^x, and minimise E^x_i. If R^x_i is the reward obtained for choosing i^{th} action at state s in the x^{th} measurement cycle, R^x_i can be defined as:

$$R^x_i = \frac{\lambda^x C^x_i}{\nu^x E^x_i} \quad (9)$$

where

$$\nu^x = \max_{i \in \mathcal{Q}^x} \frac{C^x_i}{Es^x_i} \quad (10)$$

The reward is defined such that it is bounded to [0,1]. The new Q value of the state s for choosing i^{th} action is:

$$Q_{new}(s,i) = Q(s,i) + \alpha[R^x_i(s) + \beta \max_{i \in \mathcal{S}} Q(s',i) - Q(s,i)] \quad (11)$$

The Q-table is initiated with zeros. The rewards are computed for all N actions from the training data set and the Q values

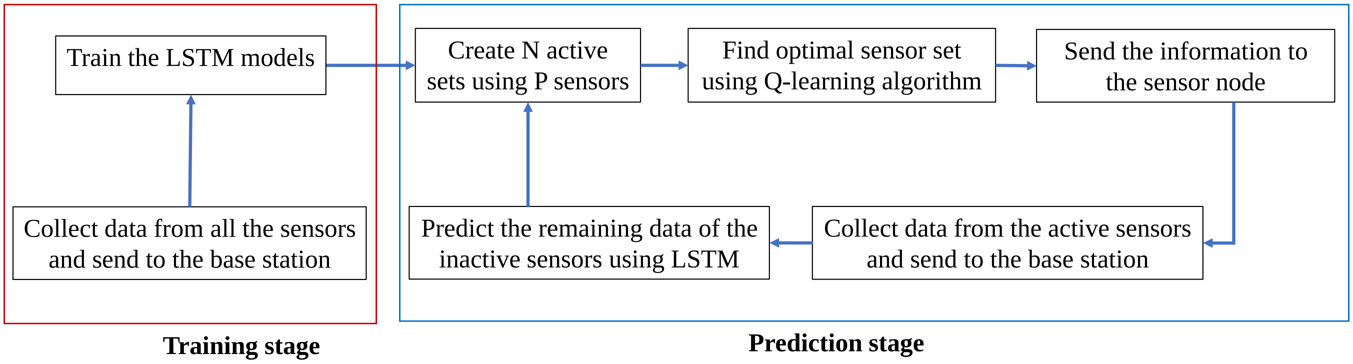

Train the LSTM models
Collect data from all the sensors and send to the base station
Create N active sets using P sensors
Find optimal sensor set using Q-learning algorithm
Send the information to the sensor node
Collect data from the active sensors and send to the base station
Predict the remaining data of the inactive sensors using LSTM
Training stage
Prediction stage

Fig. 3: Flow chart of the proposed LSTM and Q-learning-based adaptive sensing framework.

are computed using (11). The optimal active sensor set in the x^{th} measurement cycle is chosen as

$$A_{i^*}^x = \underset{i \in \mathcal{S}}{\text{argmax}} Q(s^x, i) \quad \text{s. t. } C_i^x > c_{th} \tag{12}$$

The inactive nodes will not participate in the data collection operation for this measurement cycle. The sensing signals of the inactive nodes are predicted using the LSTM, as discussed in Section III.

B. Finding Optimal Sampling Instants and Optimal Length of Measurement Cycle

As discussed in the [5], the sampling interval of the sensors is decided based on the Nyquist rate of the sensing signals. However, it has been observed that the data collected at Nyquist rate exhibits a very high temporal correlation [24]. Therefore, the sampling rate can be chosen based on the temporal correlation of the sensors. Let the temporal correlation of p^{th} parameter between $Z_p^x(i)$ and $Z_p^x(i-I)$ be $ct_p^x(I)$. If $ct_p^x(I) = ct_{th_1}$, the sampling interval of p^{th} parameter is set as $t_p^x = I$. To reduce the complexity, the sampling interval of all the sensors is set as $t^x = \min(t_p^x)$. Similarly, a new temporal correlation threshold ct_{th_2} is set to decide the retraining point. If $ct_p^x < ct_{th_2}$ of any parameter p, the model needs to be retrained.

The cross-correlation among the sensing parameters is time-varying in nature. Therefore, the length of measurement cycle τ should be chosen such that the active sensor set selected at that cycle is optimal for the entire duration. Hence, τ, in the proposed framework, is decided by exploiting the cross-correlation factor of the active sensor sets, such that the average cross-correlation factor of all the active sensor sets remains similar during the measurement cycle. The optimum number of samples collected by each sensor in the x^{th} measurement cycle is given by, $M^x = \frac{\tau^x}{t^x}$.

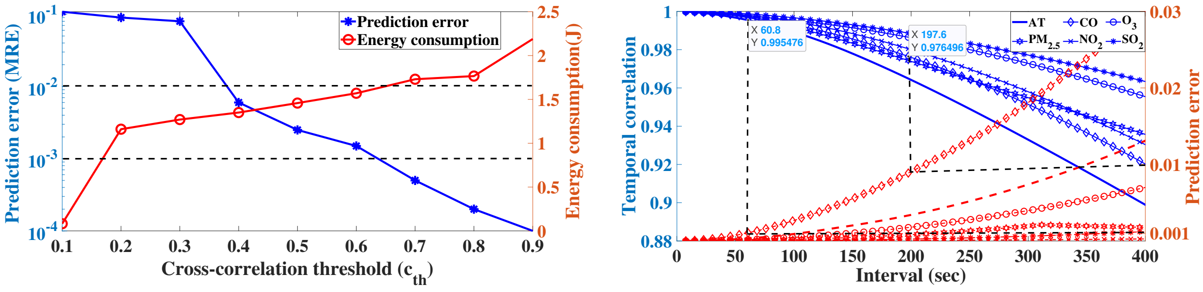
Algorithm 1: Adaptive sensing algorithm at the edge node

```
Input: $x = 0$, $e = 1$, $f_m$, $ct_{th_1}$, $ct_{th_2}$, and $c_{th}$.
if e=1 then
    Receive data for all sensing parameters from the sensor
      node and $E_0^x$
    Train/retrain and test the model with recently collected
      samples
    Exploit temporal correlation to find $t^{x+1}$ and $\tau^{x+1}$
    Construct $\mathcal{S}$ and calculate $C_i^x$ and $E_i^x; \forall i \in \mathcal{S}$ using (6)
      and (7), respectively
    Find optimal active sensor set $\mathcal{A}_i^{(x+1)}$ by solving (12)
    Set $e = 0$
else
    Receive $E_0^x$ and data from the active sensors of the
      sensor node
    Predict the missing samples using the LSTM model as
      described in Section III
    Exploit temporal correlation to find new $t^{x+1}$
    $\tau^{x+1} = \tau^x$
    Construct $\mathcal{S}$ and calculate $C_i^x$ and $E_i^x; \forall i \in \mathcal{S}$ using (6)
      and (7), respectively
    Find optimal active sensor set $\mathcal{A}_i^{(x+1)}$ by solving (12)
    Find $ct_p^x; \forall p \in \mathcal{P}$
    while $ct_p^x \ngtr ct_{th_2}; \forall p \in \mathcal{P}$ do
        Set $e = 1$, $t^{x+1} = \frac{1}{f_m}$, and $\tau^{x+1} = n \times t^{x+1}$
    end
end
Transmit $\mathcal{A}_i^{(x+1)}$, $t^{x+1}$, $\tau^{x+1}$, and $e$ to the sensor node
Set $x = x + 1$
```

C. Adaptive Sensor Selection framework

The proposed edge intelligence adaptive sensing framework is shown in Fig. 3, which is programmed at the edge node. Initially, The Nyquist sampling rate of all the sensing parameters is computed at the beginning by collecting data at a sampling interval of 1 sec. For simplicity, the Nyquist sampling rate is kept fixed.

At the beginning of the algorithm at the edge node, as presented in **Algorithm 1**, the node collects data from all the sensors at a sampling frequency $f_m = \max(f_1, f_2, \cdots, f_P)$, where f_p is the Nyquist sampling rate of the p^{th} sensor. The LSTM-based prediction model is trained at the edge node using a sufficient quantity of samples gathered from the node. The prediction model includes N sub-models that are constructed based on the input/output of N subsets in S, before exploiting the temporal correlations to find t^x and τ^x. Subsequently, the optimal sensor set is chosen based on the Q-learning method as discussed in Section IV-A. Finally, the outcome of the process consists of the optimal active sensor set $\mathcal{A}_i^x$ containing the sensors to be activated during



(a)

(b)

Fig. 4: (a)Variation of temporal correlation and prediction error with sensing interval, (b)Variation of prediction error and energy consumption with cross-correlation threshold.

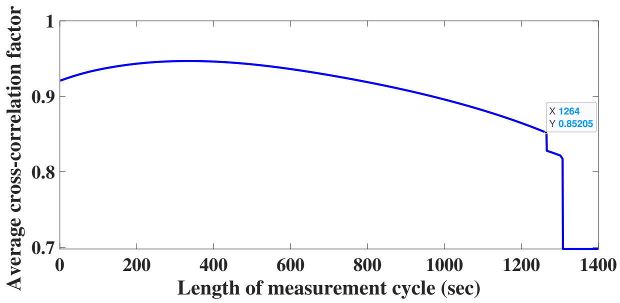


Fig. 5: Average cross-correlation factor versus measurement cycle length.

the x^{th} measurement cycle, sampling interval t^x, length of x^{th} measurement cycle τ^x, and a status flag e is transmitted to the sensor node. Based on the status of e the node activates the sensors of $\mathcal{A}_i^x$ and collects total M^x samples from each sensor at a sampling interval t^x.

At the end of the measurement cycle τ^x, the sensor node transmits the samples collected by its active sensors and its energy status E_0^x to the edge node. Upon receiving these values from the sensor node, the edge node executes the adaptive sensing algorithm and sends the required control information to the sensor node for executing in the cycle $\tau^{(x+1)}$.

To retrain the model, the temporal correlations among individual parameters are exploited. Given that these parameters represent slowly changing temporal signals, it is anticipated that there will be robust temporal correlation among the signals from one cycle to the next. Let $ct_{p,th}$ be the user-defined temporal correlation threshold of p^{th} parameter to maintain an acceptable range of prediction error. The temporal correlation ct_p^x is calculated by selecting an equal number of samples from both the currently reconstructed signals and those from past cycles at a consistent interval. When prediction errors increase, it indicates a reduction in the temporal correlations among the signals because these errors introduce randomness into the data. Therefore, all the sensors are activated in the $(x+1)^{th}$ measurement cycle if $ct_p^x \ngeq ct_{p,th} \forall p \in \mathcal{P}$. The LSTM network is retrained after collecting new samples from every sensor.

The proposed framework can be adopted to all the applications consisting of field sensor nodes with multiple sensors exhibiting a moderate to high cross-correlation among the sensing parameters.

V. Results and Discussion

The performance of the proposed LSTM and Q-learning-based adaptive sensing framework is evaluated in this section. Air pollution monitoring node is one of the widely used CIoT devices in smart city infrastructures. To analyze the efficiency of the proposed method in real-life applications, the algorithm is simulated in Python on an air quality monitoring data set from [25] where 6 sensors are used to monitor six parameters ($\mathcal{P}$ = {Temperature, $PM_{2.5}$, CO, NO_2, Ozone O_3, SO_2 }). According to [24], the sensing energy consumption of the sensor set $\mathcal{P}$ to turn ON and collect one sample is set as, $En = \{0.012, 29.55, 0.026, 0.02, 0.05, 0.026\}$ J.

Section V-A describes the estimation of optimum parameter values of the algorithm. A performance comparison of the proposed LSTM and Q-learninig-based framework with its competitive GPR-based framework [10] and SBL-based multi-sensing framework [19] is presented in Section V-B.

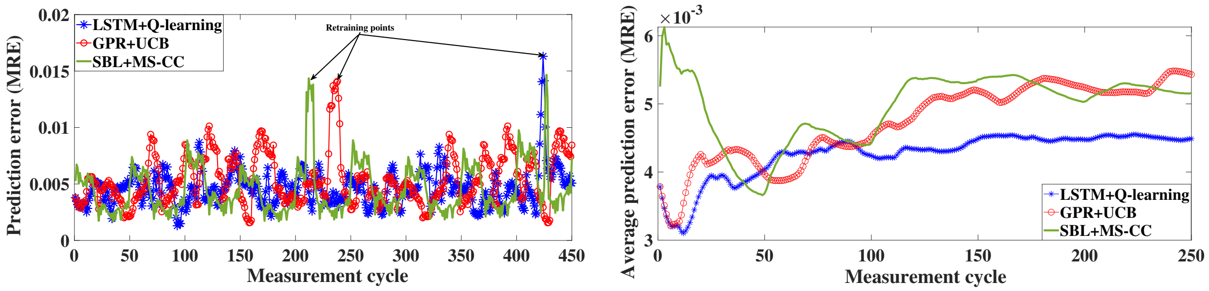
The prediction/reconstruction error of the inactive parameters is computed in terms of mean relative error (MRE). If $z_p(i)$ and $\widehat{z}_p(i)$ are the i^{th} samples of the actual and predicted data sequence of p^{th} parameter, respectively, the MRE for I number of samples is defined as,

$$\text{MRE}_p = \frac{1}{I}\sum_{i=1}^{I}\frac{|z_p(i) - \widehat{z}_p(i)|}{|z_p(i)|}. \tag{13}$$

According to [5], the sensing parameters can be predicted accurately if MRE$< 10^{-2}$. Thus, the prediction error threshold of the parameters is set as $e_{th} = 10^{-2}$, which is comparable to the error achieved from the GPR-based optimization framework described in [10]. The sub-models having MRE higher than 1% are discarded from the set $\mathcal{S}$ that contains the active-sleep subsets to reduce the computational complexity.

A. Finding Optimum System Parameters

The most important parameter to decide in the proposed framework is the cross-correlation threshold c_{th}. As shown in Fig.4(a), the sensing signal prediction error decreases with increasing c_{th}. Thus, in our study $c_{th} = 0.65$ is chosen such that the prediction errors lie within a range $[10^{-3}, 10^{-2}]$. It has been observed that the prediction error gradually increases with time. Considering the prediction error threshold as 10^{-2}, according to [8], the sub-models need to be retrained when the prediction error of any sensing parameter exceeds 10^{-2}.



(a)

(b)

Fig. 6: Variation of (a) prediction error (average error on all the sensing parameters), and (b) average prediction error (averaged over the past cycles) with measurement cycles in the proposed framework, GPR and UCB-based framework [10], and SBL-based collaborative multi-sensing framework [19].

TABLE II: Reconstruction error (MRE)

Sensing parameters	(SBL+MS-CC)-based framework [19]	(GPR+UCB)-based framework [10]	(LSTM+Q-learning)-based framework (Proposed)
Temperature	0.001	0.0007	0.002
$PM_{2.5}$	0.01	0.009	0.004
CO	0.009	0.009	0.002
NO_2	0.004	0.003	0.004
Ozone	0.007	0.006	0.009
SO_2	0.005	0.004	0.006

Although the Nyquist sampling rate always provides a faithful reconstruction of the signal, since it is based on the potential highest frequency component in the signal which may occur infrequently, the sensing energy consumption is very high in such cases as the sensors are mostly always turned ON to collect data at low sampling intervals. Therefore, the temporal correlation-based sampling interval selection method is proposed in this framework. The temporal correlation tend to decrease with the increased sampling interval, which increases the prediction error. In our study, to find the optimum sampling interval, the temporal correlation threshold ct_{th_1} is set as 0.995 for all the parameters, which provides a prediction error $\leq 10^{-3}$ for all the parameters, as shown in Fig. 4(b). The sampling interval from Fig. 4(b) is set as $t = 60$ sec for all the parameters at the first measurement cycle.

The temporal correlation is also exploited to decide whether to retrain the sub-models. From Fig. 4(b), the prediction error of a few parameters exceeds 10^{-2} when $ct < 0.976$. Accordingly, the sub-models are retrained when the temporal correlation of any p^{th}, $ct_p < ct_{th_2}$.

The length of a measurement cycle τ is decided by exploiting the cross-correlation factor of the active sensor sets of $\mathcal{S}$. From Fig. 5, it has been observed that the average cross-correlation factor of all the active sets decreases with the increasing length of the measurement cycle. Accordingly, $\tau = 21$ min is initially chosen in the proposed algorithm. However, it is recomputed during the subsequent retraining stages, when fresh samples are collected.

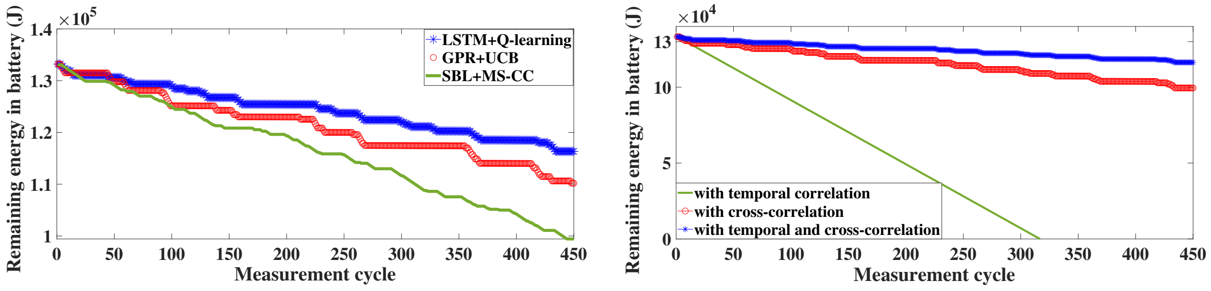
B. Performance Comparison with State-of-the-Art

The performance of the proposed joint LSTM and Q-learning-based adaptive sensing strategy is compared with the competitive state-of-the-art GPR and UCB-based optimization strategy presented in [10] and SBL-based multi-sensing strategy in [19]. The GPR and UCB-based optimization strategy in [10] was found to be the most recent and competitive state-of-the-art framework. The framework in [19] considered the network level performance, which we modified appropriately for additionally capturing the node level performance measures and comparing with the proposed framework. This section presents the relative error performance of the sensing parameters and energy consumption in the three competitive the frameworks.

The algorithms of all the three frameworks were simulated up to 450 measurement cycles for comparing the sensing quality and energy consumption performances. From Fig. 6, it can be observed that the prediction error increases with measurement cycle, which is intuitive. Fig. 6(a) shows the variation of average prediction error (average of prediction errors of all the sensing parameters) with time. The average prediction error, denoted as MRE_{avg}, is calculated by averaging the MRE of the current and past measurement cycles. Although the prediction error profile shows randomness, the mean value MRE_{avg} increases gradually, as shown in Fig. 6(b). Hence, the models are retrained when prediction errors exceed a predefined error threshold. As observed from Fig. 6(b), since LSTM has better prediction proficiency, the retraining requirement is reduced when using the LSTM model compared to GPR and SBL models. By exploring the data, $ct_{th_2} = 0.976$ is chosen for all the parameters such that the average prediction error remains within the error threshold. As shown in Fig. 6(a), the prediction error increases non-monotonically rather than linearly. Thus, the effect of high prediction error reflects on the temporal correlation of the sensing parameters after a few measurement cycles.

The average prediction error of individual sensing parameters in terms of MRE is listed in Table II. It can be observed that the prediction error of $PM_{2.5}$ and CO have improved. Thus, the average prediction error is improved in the case of LSTM-based sensor data prediction model compared to the GPR-based model and the SBL-based model.

The battery energy profile of the sensor node is shown in Fig. 7. Initially, the Q-learning method activates many sensors including the high energy-consuming sensors (such as the PM sensor) to update the Q-values of all the states until the Q-



(a)

(b)

Fig. 7: (a) Comparison of energy consumption of the proposed framework with GPR and UCB-based framework [10] and wth SBL-based collaborative multi-sensing framework [19]. (b) Energy consumption in the proposed framework with and without exploiting temporal and cross-correlation.

TABLE III: Performance comparison of the proposed framework with the competitive state-of-the-art methods

Performance parameters	(SBL+MS-CC)-based framework [19]	(GPR+UCB)-based framework ([10])	(LSTM+Q-learning)-based framework (Proposed)
Average energy consumption (J)	75	51	37
Average reconstruction error (MRE)	6×10^{-3}	5.2×10^{-3}	4.5×10^{-3}
Retraining interval	after 212 cycles	after 238 cycles	after 424 cycles
Average reward	-	0.18	0.2

values reach stability. After that, the learning agent judiciously activates the high energy consuming sensors. In contrast, UCB takes action based on the reward achieved in the previous measurement cycle. Since low energy-consuming sensors provide higher rewards compared to high energy-consuming sensors, UCB judiciously activates the sensor from the beginning. However, in the long run, it has been observed that Q-learning-based framework performs better than UCB, as depicted in Fig. 7(a). Since the optimal number of active sensors set in the SBL-based framework are chosen by comparing the cross-correlation coefficients only, the battery energy of the sensor node decays faster than the UCB-based framework as well as the proposed Q-learning-based framework. For fair comparison, in all the frameworks, c_{th} was set as 0.65 as the cross-correlation threshold.

Fig. 7(b) presents the battery energy profile of the sensor node in the proposed framework in three scenarios: while exploiting only the temporal correlation of individual sensors, only cross-correlation among the sensors, and both the temporal and cross-correlations of the sensors. It can be observed that the energy efficiency of the node significantly improves by exploiting the cross-correlation of the sensors and judiciously activating the sensors. The energy efficiency improves further by exploiting both the temporal and cross-correlation of the sensing signals.

Table III presents the performance comparison of the proposed framework with the competitive state-of-the-art approaches in [10] and [19]. The proposed algorithm shows up to 13% (25%) improvement in error performance and up to 20% (50%) improvement in sensing energy consumption, while maintaining a lower bound of cross-correlation coefficient $c_{th} = 0.65$ between the inactive and active sensor set. Further, considering an error threshold of 10^{-2}, the retraining interval in the proposed framework increases to 424 cycles from 238 cycles in [10] and 212 cycles in [19], as also visible from the trends in Fig. 6(a). Although the computational complexity of LSTM is higher than GPR and SBL due to multiple hidden layers, the algorithm is implemented in a high computing edge node, where the processing delay can be accommodated. The proposed learning-based framework is appropriate for semi-real-time or non-real-time systems, where energy sustainability of the field sensor nodes is prioritized over time criticality.

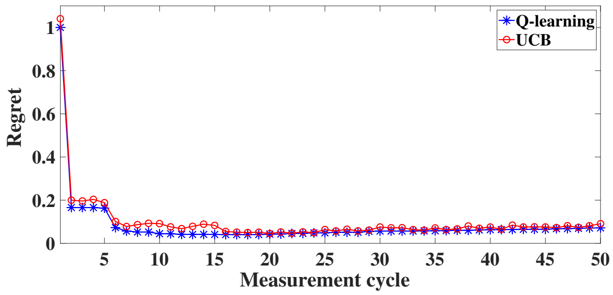


Fig. 8: Regret versus measurement cycle.

The convergence of the reinforcement learning algorithms is studied in terms of regret. Regret is the price paid by the learner for not choosing the optimal sensor set. It is computed as the difference between the maximum reward that could be achieved and the reward obtained at the current cycle based on the active sensor set chosen at that cycle. From Fig. 8 it can be observed that the Q-learning-based optimization framework converges faster (within 10 measurement cycles) than the UCB-based optimization framework (about 18 cycles). Thus, UCB requires to explore more number of cycles before converging compared to the Q-learning based approach.

VI. Concluding Remarks

In this article, an efficient edge intelligence adaptive sensor selection strategy has been introduced for the CIoT nodes that combine LSTM and model-less reinforcement learning (Q-learning). When dealing with an environment that undergoes constant change (referred to as non-stationarity), it is more enlightening to learn from all historical data rather than depending solely on the most immediate data. To measure the quality of sensed signals (as it is often impractical to determine actual prediction errors), we have defined a correlation factor, which serves as a robust performance indicator. The proposed edge intelligence framework has been designed to identify the most efficient sensor set by leveraging insights from the system's dynamics while carefully managing energy resources. Furthermore, the optimal sampling and data transmission intervals have been determined based on the temporal variation of the sensor parameters. Extensive investigations conducted using air pollution monitoring data have confirmed that our proposed algorithm significantly enhances error performance and energy efficiency compared to the current state-of-the-art approaches, while maintaining a similar correlation among the reconstructed parameters at the edge node.

References

[1] G. Hong and D. Shin, "Virtual connection: Selective connection system for energy-efficient wearable consumer electronics," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 4, pp. 299–307, 2020.

[2] P. Chanak and I. Banerjee, "Congestion free routing mechanism for iot-enabled wireless sensor networks for smart healthcare applications," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 3, pp. 223–232, 2020.

[3] P. S. Chatterjee, N. K. Ray, and S. P. Mohanty, "Livecare: An iot-based healthcare framework for livestock in smart agriculture," *IEEE Transactions on Consumer Electronics*, vol. 67, no. 4, pp. 257–265, 2021.

[4] P. Das, S. Ghosh, S. Chatterjee, and S. De, "Energy harvesting-enabled 5G advanced air pollution monitoring device," in *proc. IEEE 3rd 5G World Forum (5GWF), Bangalore, India, India*, 2020, pp. 218–223.

[5] H. Harb and A. Makhoul, "Energy-efficient sensor data collection approach for industrial process monitoring," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 661–672, 2017.

[6] J. Wang, S. Tang, B. Yin, and X.-Y. Li, "Data gathering in wireless sensor networks through intelligent compressive sensing," in *proc. IEEE INFOCOM, Orlando, FL, USA*. IEEE, 2012, pp. 603–611.

[7] S. Hwang, R. Ran, J. Yang, and D. K. Kim, "Multivariated Bayesian compressive sensing in wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 7, pp. 2196–2206, 2015.

[8] C. Alippi, G. Anastasi, M. Di Francesco, and M. Roveri, "An adaptive sampling algorithm for effective energy management in wireless sensor networks with energy-hungry sensors," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 2, pp. 335–344, 2009.

[9] A. Jain and E. Y. Chang, "Adaptive sampling for sensor networks," in *proc. of the 1st international workshop on Data management for sensor networks: in conjunction with VLDB 2004, Toronto, Canada*, 2004, pp. 10–16.

[10] S. Ghosh, S. De, S. Chatterjee, and M. Portmann, "Learning-based adaptive sensor selection framework for multi-sensing WSN," *IEEE Sensors J.*, pp. 1–1, 2021.

[11] M. Tang and V. W. Wong, "Deep reinforcement learning for task offloading in mobile edge computing systems," *IEEE Transactions on Mobile Computing*, vol. 21, no. 6, pp. 1985–1997, 2020.

[12] L. Huang, S. Bi, and Y.-J. A. Zhang, "Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks," *IEEE Transactions on Mobile Computing*, vol. 19, no. 11, pp. 2581–2593, 2019.

[13] H. Jiang, X. Dai, Z. Xiao, and A. Iyengar, "Joint task offloading and resource allocation for energy-constrained mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 4000–4015, 2022.

[14] Y. Zhang, Y. Wang, M. Gao, Q. Ma, J. Zhao, R. Zhang, Q. Wang, and L. Huang, "A predictive data feature exploration-based air quality prediction approach," *IEEE Access*, vol. 7, pp. 30 732–30 743, 2019.

[15] W. Qiao, W. Tian, Y. Tian, Q. Yang, Y. Wang, and J. Zhang, "The forecasting of PM2. 5 using a hybrid model based on wavelet transform and an improved deep learning algorithm," *IEEE Access*, vol. 7, pp. 142 814–142 825, 2019.

[16] J. Ma, Y. Ding, V. J. Gan, C. Lin, and Z. Wan, "Spatiotemporal prediction of PM2. 5 concentrations at different time granularities using IDW-BLSTM," *IEEE Access*, vol. 7, pp. 107 897–107 907, 2019.

[17] B. Zou, Q. Pu, M. Bilal, Q. Weng, L. Zhai, and J. E. Nichol, "High-resolution satellite mapping of fine particulates based on geographically weighted regression," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 4, pp. 495–499, 2016.

[18] V. Gupta and S. De, "SBL-based adaptive sensing framework for WSN-assisted IoT applications," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4598–4612, 2018.

[19] V. Gupta and S. De, "Collaborative multi-sensing in energy harvesting wireless sensor networks," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 6, pp. 426–441, 2020.

[20] Y. Xu, J. Choi, S. Dass, and T. Maiti, "Sequential Bayesian prediction and adaptive sampling algorithms for mobile sensor networks," *IEEE Trans. Autom. Control*, vol. 57, no. 8, pp. 2078–2084, 2011.

[21] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[22] L. Cai, M. Boukhechba, N. Kaur, C. Wu, L. E. Barnes, and M. S. Gerber, "Adaptive passive mobile sensing using reinforcement learning," in *proc. IEEE WoWMoM, Washington, DC, USA, USA*, 2019, pp. 1–6.

[23] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2020.

[24] S. Ghosh, S. De, S. Chatterjee, and M. Portmann, "Edge intelligence framework for data-driven dynamic priority sensing and transmission," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 1, pp. 376–390, 2022.

[25] CPCB. (2021) Air pollution in delhi. [Online]. Available: https://airquality.cpcb.gov.in/ccr/\#/caaqm-dashboard\\-all/caaqm-landing