Deep Recurrent Neural Network for Time Series Modelling of Energy in Wireless Sensor Network Nodes

Zoren P. Mabunga
School of Graduate Studies and College of Engineering
Mapúa University and Southern Luzon State University
Manila, Philippines
zorenmabunga@gmail.com

Jennifer C. Dela Cruz
School of Electrical, Electronics and Computer Engineering
Mapúa University
Manila, Philippines
jcdelacruz@mapua.edu.ph

***Abstract*— This study addresses the challenge of predicting energy consumption in wireless sensor network (WSN) nodes using advanced AI techniques, specifically deep recurrent neural networks (DRNN). By collecting datasets from both static and dynamic sensor nodes, we aim to reflect diverse real-world energy consumption patterns. We developed and trained LSTM-based models, leveraging their proficiency in processing time-series data and learning temporal dependencies. The results demonstrated that the LSTM model significantly outperformed other models, including stacked LSTM, Prophet, and exponential smoothing. For static nodes, the LSTM model achieved an RMSE of 213.39 joules, MAE of 491.75 joules, and MAPE of 2.53%, while for dynamic nodes, it achieved an RMSE of 293.26 joules, MAE of 212.76 joules, and MAPE of 1.46%. In comparison, the stacked LSTM, Prophet, and exponential smoothing models showed higher error rates. The LSTM model's superior performance is attributed to its ability to balance complexity and generalization, making it well-suited for the relatively simple dataset used in this research.**

Keywords – long short-term memory networks, wireless sensor network, energy, forecasting, dynamic nodes, static nodes

I. INTRODUCTION

In today's world, where technological advancements are everywhere, wireless sensor networks (WSNs) have become very important in many areas, such as environmental monitoring, education, industrial automation, agriculture and healthcare [1]–[6]. WSNs are spatially distributed sensors that are autonomous and capable of monitoring different physical or environmental conditions and sending their data to a central point [7]. Implementing these networks into these sectors is an important steppingstone towards achieving more effectiveness and automation. However, various challenges come with the use of WSNs. One of them is the limited power supply to the sensor nodes, which directly affects the network's lifetime operationally and efficiency. Many energy-saving forms in WSNs like effective routing and data fusion [8]–[14], often do not take care of these networks dynamicity.

In recent years, the integration of Artificial Intelligence (AI), particularly Deep Learning (DL) and Recurrent Neural Networks (RNNs), has shown promise in enhancing the energy management of WSNs. AI's ability to learn and adapt to varying conditions presents a novel approach to managing the energy consumption of sensor nodes. Studies like those by Alqaralleh et al. [15] have developed energy-efficient tracking models in Wireless Multimedia Sensor Networks using DL and RNNs, demonstrating the potential of AI in optimizing energy usage. Similarly, Alameen and Gupta [16] focused on intrusion detection in Wireless Body Sensor Networks using Deep RNN, highlighting the versatility of AI applications in WSNs. However, these studies primarily concentrate on specific applications or aspects of WSNs. There is a notable gap in the comprehensive energy modeling of sensor nodes using deep recurrent neural networks, which can be universally applied to various WSN applications.

This research's primary problem is the lack of a dynamic and efficient model for predicting the energy consumed by WSN nodes using advanced AI techniques. Existing methods are limited in adaptability and efficiency, and there is a critical need for a model that can predict energy requirements in real time, considering both static and dynamic sensor nodes in various WSN applications. For a more widespread adoptation and a more sustainable operation of a WSNs, this gap must be address.

This study is aims to tackle these challenges through the following objectives: Firstly, by gathering datasets from both static and dynamic sensor nodes, with a focus on the time-series energy levels of these nodes. This step is important as it provides the necessary real-world data, reflecting the energy consumption patterns of sensor nodes in an actual scenario. Secondly, the study involves developing and training a deep recurrent neural network (DRNN) model, which is uniquely suited for this task due to its proficiency in processing time-series data and learning temporal dependencies. These capabilities are crucial for accurately predicting the energy consumption of sensor nodes. Lastly, the study includes a comparative analysis of various DRNN variations and traditional statistical forecasting models to determine the most effective model for energy prediction in WSNs.

The main significance of this paper is its potential on the improvement of energy management in wireless sensor

networks. By accurately modeling and predicting energy consumption, the proposed DRNN-based approach can assist in the selection of optimal paths during data transmission that can significantly extend the operational efficiency and longevity of WSNs. This research could lead to more sustainable network operations, reduce maintenance costs, improve network reliability, and pave the way for more AI-driven solutions in WSNs.

II. Methodology

A. Data Gathering and Pre-Processing

A prototype that measures, collects, and transmits air quality parameters was developed to ensure an accurate and realistic dataset. The prototype was developed using a LoRa module as the transceiver, an air quality sensor (CCS811 Module), a NEO6M GPS module, and a Lithium-Ion battery with a capacity of 5000 mAh. LoRa is a wireless technology that offers advantages for wireless data transmission such as long range, low power and secure data communication which are ideal for IoT applications and M2M communication. In this study, the TTGO v2 ESP32 LoRa Module was used for the transmitter and receiver device. Another device is constructed that will serve as the receiver for the data-gathering setup. The receiver was constructed using the same LoRa module, an RTC module for time logging, and a battery supply with a capacity of 5000 mAh. Table I tabulates the specifications for the Li-ion battery used as the prototype's power source.

TABLE I. Li-ion Battery Specifications

Model	Lii-50A
Nominal Capacity	5000 mAh
Nominal Voltage	3.65 V
Internal Resistance	<20 mohms
Charging Cut-off Voltage	4.2 V
Discharge Cut-off Voltage	3.0 V
Operating Temperature	Charge: 0 – 45 °C Discharge: -20 – 60 °C

The transmitting device was first installed in a fixed position until the energy supply is depleted. This setup was used to gather the static sensor node time series energy data used for testing and training the different models for a static sensor node. The device gathered the area's carbon dioxide and volatile organic compounds measurements. The device then transmits the sensor measurement every 30 seconds and the battery percentage and remaining energy of the battery supply in Joules. The receiving device received this data through its LoRa transceiver module and saved it to its SD card. After one battery discharging cycle, 3503 data sets were gathered and saved in the SD card for further processing. For the collection of dynamic sensor node time series energy data, the transmitting device was installed in five different locations that mimic the mobility of the wireless sensor node. This process is done until the energy supply of the sensor node is depleted. After one battery cycle, we gathered a total of 3976 sets of data for the dynamic sensor node setup. Figures 1 and 2 show the graphical representation of the gathered energy over time for a static and dynamic sensor node. This dataset was used to train, validate, and test the sensor nodes' energy level model.

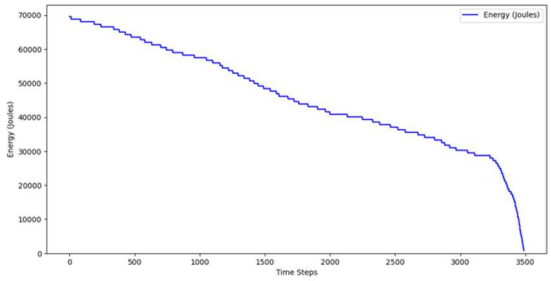

Fig. 1. Battery Energy (Joules) over Time for Static Sensor Nodes

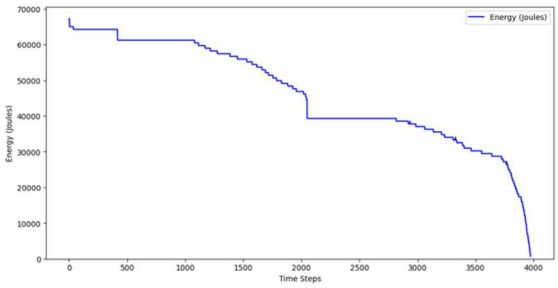

Fig. 2. Battery Energy (Joules) over Time for Dynamic Sensor Nodes

After the data were gathered, they were manually cleaned and preprocessed before the energy model's training process. Data visualization was first conducted to check for inconsistencies, missing values, and outliers on the dataset. The gathered data also undergoes min-max scaling. This process is crucial to ensure that the dataset has uniform values and that each value contributes equally to the analysis and model development. Min-max scaling also allows faster training time since each value will now have a uniform scale (0-1). One discharge cycle of the battery will serve as the training data for each model, and a different battery discharge cycle data will be used for the testing.

B. Exponential Smoothing (Holt's Trend Model)

One forecasting method for univariate time series data is utilizing past observations to predict future values. In the exponential smoothing method [17], its prediction is based on the weighted linear sum of past values or observations. This forecasting method is accurate when the dataset exhibits seasonality or trend. In this paper, we employ a type of exponential smoothing called the Holt's trend method. This type of exponential smoothing is best suited for our dataset since it exhibits a clear declining trend of battery energy with minimal seasonal variation. Mathematical prediction calculation using Holt's Trend Method are performed using equations 1, 2 and 3 as shown below. Equation 1 is the level equation where l_t represents the level at time t, y_t is the observed values at time t, α represents the parameter for smoothing the level, and b_{t-1} is the estimated trend at time t-1. Equation 2 is the trend equation where b_t serves as the trend parameter for a particular time t, and β represents the smoothing parameter for that particular trend. Lastly, equation 3 is the forecast equation for h periods ahead.

$$l_t = \alpha y_t + (1-\alpha)(l_{t-1} + b_{t-1}) \quad (1)$$

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \quad (2)$$

$$y_{t+h} = l_t + hb_t \quad (3)$$

C. Prophet Forecasting Model

The Prophet forecasting model [18], developed by Facebook, is designed to handle complex time series data typical in business contexts, such as multiple seasonality, trend changes, and outliers. Prophet uses the generalized additive framework (GAM) that divides a time series data into three parts namely, holiday, trend and seasonality. The core model is represented by equation 4 below.

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \quad (4)$$

where g(t) represents the non-periodic trend, s(t) is the periodic seasonal effects, h(t) includes holiday impacts, and ε_t accounts for noise. The trend component g(t) can be modeled as either a piecewise linear function or a logistic growth curve, with changepoints automatically detected to allow for shifts in the trend. Seasonal effects are modeled using the Fourier series, enabling the model to capture multiple seasonalities, such as weekly and yearly cycles. This flexibility, combined with an interactive analyst-in-the-loop approach, allows domain experts to adjust model parameters and improve forecast accuracy, making Prophet a robust tool for business forecasting at scale.

D. Long Short-Term Memory Networks

The LSTM architecture [19], a specialized form of DRNN designed for processing time series or sequential data. These type of DRNN consists of three gates that control the information flow in each cell. These gates are the forget gate, input gate and output gate. This design effectively addresses the exploding and vanishing gradient issues encountered during training. The previous hidden state and the current input affect the status of the forget gate into what information must be retained or forgotten. On the other hand, the input gate, utilizing the same inputs, determines which new information or data should be accepted. These gates collectively update the status of each cell in the network by integrating both past and new added information. The output gate then select which portion of the cell state will becomes the output. This mechanism enables LSTMs to manage and extract important information or data across each time step, allowing them to extract patterns from the time series data. The structure of a simple LSTM unit is presented in figure 3.

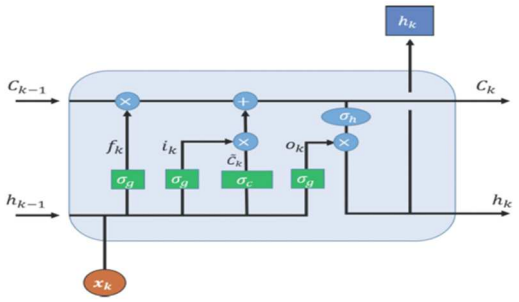

Fig. 3. LSTM unit Structure [20]

In this study, two DRNN based model were developed. One energy model was developed using the standard LSTM, and a second energy model was developed using a stacked LSTM (S-LSTM). Stacked LSTM networks consist of multiple LSTM layers arranged sequentially, where the input a the next layer is the output from the previous layer. This structure allows the network to capture deeper information from the given dataset. The lower layers typically capture simple patterns and short-term dependencies, while the higher layers can determine more complex patterns and longer-term dependencies. This multi-layered approach significantly enhances the model's capacity to learn complex temporal information that may be present in the data. A graphical representation of a stacked LSTM is presented in figure 4.

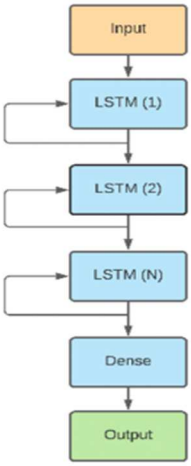

Fig. 4. Stacked-LSTM Architecture [21]

E. Hyperparameter Tuning

The selection of the hyperparameter or simply hyperparameter tuning is a critical process in AI-based modeling. The goal of this process is to find the best set of hyperparameters that can maximize the model's performance on a particular dataset. For LSTM and stacked LSTM, typical parameters involved in the tuning process are the number of units, optimizers, dropout rates, batch sizes, and activation functions. RandomizedSearchCV was used to select the optimal parameters. The search range for each parameter is tabulated in Table II. The same parameters and search range are used for the normal LSTM model and stacked LSTM model.

TABLE II. HYPERPARAMETER TUNING SEARCH RANGE

Units	50, 100, 200, 500
Dropout Rate	0.1, 0.2, 0.3, 0.4, 0.5
Activation Function	Relu, tanh
Optimizer	Adam, rmsprop
Batch Size	32, 64, 128
Number of Layers (for stacked LSTM)	2-4

F. Model Evaluation Metrics

In this study, we used three evaluation metrics to assess the performance of the four developed energy forecasting models for static and dynamic nodes. The metrics that was used are the mean absolute percentage error (MAPE), mean

absolute error (MAE), and root mean square error (RMSE). Equations 6, 7 and 8 were used to calculate each metric,

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(o_i - p_i)^2} \quad (6)$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|o_i - p_i| \quad (7)$$

$$MAPE = \frac{\sum_{i=1}^{N}\left|\frac{o_i - p_i}{p_i}\right| x100}{N} \quad (8)$$

where N represents the sample number, o_i is the actual/true value and p_i is the predicted/forecasted value.

III. Results

This section summarizes the performance of the different forecasting models using exponential smoothing, Prophet model, LSTM, and stacked LSTM to forecast the energy level of a wireless sensor node. The different models were evaluated using MAPE, RMSE, and MAE metrics. For the LSTM and stacked LSTM, the different hyperparameters were selected using the hyperparameter tuning process, which results in the hyperparameters presented in Table III.

TABLE III. Hyperparameters for the LSTM and S-LSTM Model

Hyperparameter	LSTM	S-LSTM
Number of Units	50	500
Dropout Rate	0.1	0.1
Activation Function	tanh	tanh
Optimizer	adam	adam
Batch Size	32	32
Number of Layers	N/A	2

A. Forecasting Model's Performance for Static Node

The performance of the four models for forecasting static sensor node energy was tabulated in Table IV and visualized in Figure 5. The exponential smoothing model obtained the largest RMSE, MAE, and MAPE, with values of 5317.89 joules, 4687.97 joules, and 22.82%, respectively. On the other hand, the LSTM-based energy forecasting model provided the least amount of RMSE, MAE, and MAPE, with values of 213.39 joules, 491.75 joules, and 2.53%, respectively.

TABLE IV. Forecasting Models for Static Node

Model	RMSE	MAE	MAPE
Exponential Smoothing	5317.89	4687.97	22.82%
Prophet	4359.80	3507.17	20.64%
LSTM	213.39	491.75	2.53%
Stacked-LSTM	673.45	535.11	3.43%

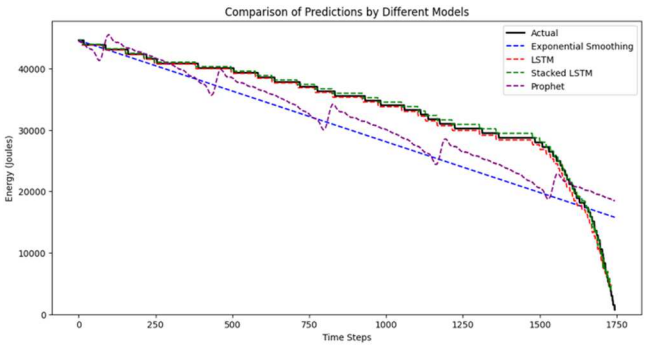

Fig. 5. Static Node's Energy Forecast

B. Forecasting Model's Performance for Dynamic Node

The same models used for the static sensor node were evaluated for the dataset gathered using a dynamic sensor node. The performance of each model for a dynamic sensor node is tabulated in Table V and presented in Figure 6. The exponential smoothing model provides the largest MAE and MAPE with values of 4619.27 joules and 24.61%, respectively, while the Prophet model gives the largest RMSE of 5486.02 joules. The RMSE, MAE, and MAPE of the LSTM-based energy forecasting model give the lowest values among the four different forecasting techniques, having values of 293.26 joules, 212.76 joules, and 1.46%, respectively.

TABLE V. Forecasting Models for Dynamic Node

Model	RMSE	MAE	MAPE
Exponential Smoothing	5242.16	4619.27	24.61%
Prophet	5486.02	3856.53	22.08%
LSTM	293.26	212.76	1.46%
Stacked-LSTM	605.41	486.19	2.28%

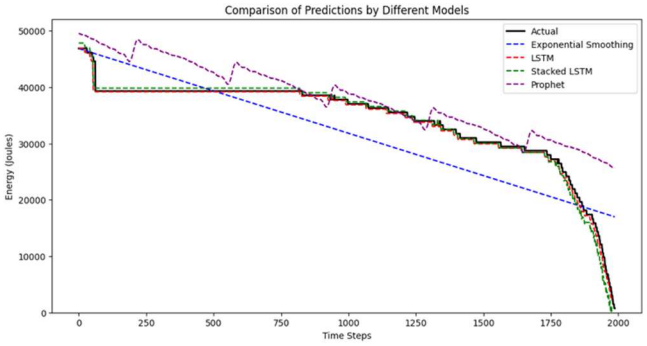

Fig. 6. Dynamic Node's Energy Forecast

IV. Discussion

This research reveals the superiority of artificial intelligence-based forecasting models over statistical-based forecasting models for the forecasting of wireless sensor node energy levels. The LSTM and stacked LSTM (S-LSTM) based model's superior performance compare with statistical based forecasting model is attributed to its capability to extract long-term dependencies and learn patterns from its training dataset. As evident in LSTM and S-LSTM architecture that includes gates for controlling the flow of information, this allows information to be maintained and

updated over long data sequences, leading to a more accurate forecast.

However, S-LSTM, which is a more advanced and complex variation of the standard LSTM model, was outperformed by the standard LSTM model. The S-LSTM model achieved RMSE of 673.45 joules, MAE of 535.11 joules, and MAPE of 3.43% for static node test data and RMSE of 605.41 joules, MAE of 486.19 joules, and MAPE of 2.28% for the dynamic sensor node test data. These results are close to the results of the standard LSTM but still provide a larger forecast error. The added complexity of the S-LSTM, which includes additional layers, may have led to overfitting, which results in a poor generalization of the test data. The simplicity of the wireless sensor node energy data that was used in this research is another factor contributing to the better performance of the simpler LSTM model, wherein the added complexity in the architecture of the S-LSTM doesn't provide additional benefits to its performance.

V. Conclusion and Future Works

The study successfully addressed the gaps in the current literature in predicting energy consumption in wireless sensor networks (WSNs) using advanced AI techniques. The LSTM model consistently outperformed the other models, including stacked LSTM, Prophet, and exponential smoothing. The superior performance of the LSTM model, reflected in its lower RMSE, MAE, and MAPE values, highlights its effectiveness in capturing complex temporal dependencies in the energy consumption data. This suggests that simpler models like LSTM, which balance complexity and generalization, are better suited for this application than more complex models like stacked LSTM, which may overfit the data. The comparative analysis also showed that traditional statistical models like exponential smoothing and Prophet struggled with the adaptability and efficiency required for accurate energy prediction. In contrast, the LSTM model's ability to learn from sequential data and adapt to both static and dynamic conditions makes it a robust choice for real-time energy forecasting in WSNs. These findings confirm that using LSTM-based models can bridge existing research gaps, providing a dynamic and efficient solution for predicting energy consumption in WSNs. The study's outcomes pave the way for more reliable and sustainable operations of WSNs in critical applications, ultimately facilitating their wider adoption.

Future research can explore the use of other LSTM variations, such as bidirectional LSTM and transformer-based models. The ensemble concept can also explore combining AI-based and statistical-based forecasting models. Regarding the dataset, future research can also incorporate other factors that might affect the energy level of a WSN, such as the weather conditions in the area of deployment.

Acknowledgement

The authors would like to express their gratitude to Mapua University, School of Graduate Studies, and the Department of Science and Technology—Engineering Research and Development for Technology (DOST-ERDT) for the financial support they provided.

References

[1] M. Ma, B. He, N. Wang, and R. Shen, "A method for monitoring the solar resources of high-scale photovoltaic power plants based on wireless sensor networks," *Sustain. Energy Technol. Assessments*, vol. 53, no. PC, p. 102678, 2022, doi: 10.1016/j.seta.2022.102678.

[2] S. Sadeghi, N. Soltanmohammadlou, and F. Nasirzadeh, "Applications of wireless sensor networks to improve occupational safety and health in underground mines," *J. Safety Res.*, vol. 83, pp. 8–25, 2022, doi: 10.1016/j.jsr.2022.07.016.

[3] V. Chowdary, D. Deogharia, S. Sowrabh, and S. Dubey, "Forest fire detection system using barrier coverage in wireless sensor networks," *Mater. Today Proc.*, vol. 64, pp. 1322–1327, 2022, doi: 10.1016/j.matpr.2022.04.202.

[4] S. Muruganandam, R. Joshi, P. Suresh, N. Balakrishna, K. H. Kishore, and S. V. Manikanthan, "A deep learning based feed forward artificial neural network to predict the K-barriers for intrusion detection using a wireless sensor network," *Meas. Sensors*, vol. 25, no. December 2022, p. 100613, 2023, doi: 10.1016/j.measen.2022.100613.

[5] C. Jamroen, P. Komkum, C. Fongkerd, and W. Krongpha, "An intelligent irrigation scheduling system using low-cost wireless sensor network toward sustainable and precision agriculture," *IEEE Access*, vol. 8, pp. 172756–172769, 2020, doi: 10.1109/ACCESS.2020.3025590.

[6] Z. Mabunga and G. Magwili, "Greenhouse Gas Emissions and Groundwater Leachate Leakage Monitoring of Sanitary Landfill," 2019, doi: 10.1109/HNICEM48295.2019.9072872.

[7] F. Zhao and L. J. Guibas, "Introduction," *Wirel. Sens. Networks*, pp. 1–21, 2004, doi: 10.1016/b978-155860914-3/50001-8.

[8] M. Y. Cheng, Y. Bin Chen, H. Y. Wei, and W. K. G. Seah, "Event-driven energy-harvesting wireless sensor network for structural health monitoring," *Proc. - Conf. Local Comput. Networks, LCN*, pp. 364–372, 2013, doi: 10.1109/LCN.2013.6761268.

[9] R. Nagaraju *et al.*, "Secure Routing-Based Energy Optimization for IoT Application with Heterogeneous Wireless Sensor Networks," *Energies*, vol. 15, no. 13, 2022, doi: 10.3390/en15134777.

[10] H. M. Baradkar and S. G. Akojwar, "Implementation of energy detection method for spectrum sensing in cognitive radio based embedded wireless sensor network node," *Proc. - Int. Conf. Electron. Syst. Signal Process. Comput. Technol. ICESC 2014*, pp. 490–495, 2014, doi: 10.1109/ICESC.2014.92.

[11] M. Rami Reddy, M. L. Ravi Chandra, P. Venkatramana, and R. Dilli, "Energy-Efficient Cluster Head Selection in Wireless Sensor Networks Using an Improved Grey Wolf Optimization Algorithm," *Computers*, vol. 12, no. 2, 2023, doi: 10.3390/computers12020035.

[12] F. Zhu and W. Wang, "A Distributed Unequal Clustering Routing Protocol Based on the Improved Sine Cosine Algorithm for WSN," *J. Sensors*, vol. 2022, 2022, doi: 10.1155/2022/7382098.

[13] A. Hossan and P. K. Choudhury, "DE-SEP: Distance and Energy Aware Stable Election Routing Protocol for Heterogeneous Wireless Sensor Network," *IEEE Access*, vol. 10, pp. 55726–55738, 2022, doi: 10.1109/ACCESS.2022.3177190.

[14] X. Xue, R. Shanmugam, S. K. Palanisamy, O. I. Khalaf, D. Selvaraj, and G. M. Abdulsahib, "A Hybrid Cross Layer with Harris-Hawk-Optimization-Based Efficient Routing for Wireless Sensor Networks," *Symmetry (Basel).*, vol. 15, no. 2, 2023, doi: 10.3390/sym15020438.

[15] B. A. Y. Alqaralleh, S. N. Mohanty, D. Gupta, A. Khanna, K. Shankar, and T. Vaiyapuri, "Reliable Multi-Object Tracking Model Using Deep Learning and Energy Efficient Wireless Multimedia Sensor Networks," *IEEE Access*, vol. 8, pp. 213426–213436, 2020, doi: 10.1109/ACCESS.2020.3039695.

[16] A. Alameen and A. Gupta, "OPTIMIZATION-DRIVEN DEEP RECURRENT NEURAL NETWORK FOR INTRUSION DETECTION AND HEALTH RISK ASSESSMENT IN WIRELESS BODY SENSOR NETWORK," *Biomed. Eng. Appl. Basis Commun.*, vol. 33, no. 1, 2021, doi: https://doi.org/10.4015/S1016237220500441.

[17] E. S. Gardner, "Exponential Smoothing part 1," *Journal of Forecasting*, vol. 4. pp. 1–28, 1985.

[18] S. J. Taylor and B. Letham, "Business Time Series Forecasting at Scale," *PeerJ Prepr. 5e3190v2*, vol. 35, no. 8, pp. 48–90, 2017, [Online]. Available:

https://peerj.com/preprints/3190/%0Ahttp://ezproxy.bangor.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=c8h&AN=108935824&site=ehost-live%0Ahttps://peerj.com/preprints/3190/%0Ahttps://peerj.com/preprints/3190.pdf.

[19] J. Hochreiter, S., & Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] F. Yang, X. Song, F. Xu, and K. L. Tsui, "State-of-Charge Estimation of Lithium-Ion Batteries via Long Short-Term Memory Network," *IEEE Access*, vol. 7, pp. 53792–53799, 2019, doi: 10.1109/ACCESS.2019.2912803.

[21] R. R. Maaliw, Z. P. Mabunga, and F. T. Villa, "Time-Series Forecasting of COVID-19 Cases Using Stacked Long Short-Term Memory Networks," *2021 Int. Conf. Innov. Intell. Informatics, Comput. Technol. 3ICT 2021*, pp. 435–441, 2021, doi: 10.1109/3ICT53449.2021.9581688.