A Machine Learning Approach to Reduce Latency in Edge Computing for IoT Devices

Muddassar Ali
Department of Computer Science, Faculty of Computer Science & IT, Superior University Lahore, 54000, Lahore, Pakistan
muddassar.ali7927@gmail.com

Hamayun Khan
Department of Computer Science, Faculty of Computer Science & IT, Superior University Lahore, Lahore, Pakistan
hamayun.khan@superior.edu.pk (corresponding author)

Muhammad Tausif Afzal Rana
School of Information Technology, King's Own Institute, Sydney, Australia
touseef.rana@koi.edu.au

Arshad Ali
Faculty of Computer and Information Systems, Islamic University of Madinah, Al Madinah, Al Munawarah, Saudi Arabia
a.ali@iu.edu.sa

Muhammad Zeeshan Baig
Department of Information Technology, Wentworth Institute of Higher Education, Sydney, Australia
muhammad.baig@win.edu.au

Saif ur Rehman
Faculty of Electrical Engineering and Technology, Superior University Lahore, Lahore, Pakistan
saifurrehman@superior.edu.pk

Yazed Alsaawy
Faculty of Computer and Information Systems, Islamic University of Madinah, Al Madinah, Al Munawarah, Saudi Arabia
yalsaawy@iu.edu.sa

Received: 11 July 2024 | Revised: 28 July 2024 | Accepted: 3 August 2024

ABSTRACT

Nowadays, high latency in Edge Computing (EC) for Internet of Things (IoT) devices due to network congestion and online traffic reduces the acquired precision, performance, and processing power of the network. Data overload in IoT significantly impacts the real-time capabilities of user experience, decision-making efficiency, operational costs, and security in EC. By combining EC innovation and three Machine Learning (ML) models, namely Decision Trees (DT), Support Vector Machines (SVMs), and Convolutional Neural Networks (CNNs), this research aims to tackle the inactivity of IoT devices and information cleaning from errors. Its purpose is to preserve information astuteness and highlight the efficacy of each model's execution by using the essential components of previous approaches. The proposed model evaluates the precision, performance, and quality enhancement by measuring the Mean Square Error (MSE), coefficient of determination (R^2), and accuracy.

Keywords-edge computing; machine learning; data analysis; model evaluation; IoT; convolutional neural networks; latency reduction; mean square error

I. INTRODUCTION

The communication between Internet of Things (IoT) devices is based in reliable accessibility and data exchange between them. As IoT expand, those two parameters become crucial for the correct results of different functions, because they effect the respond time of devices. So, monitoring and decreasing idleness are especially challenging in IoT organizations [1]. In IoT frameworks, inactivity, or the time it takes to send and get information, could be a significant execution metric. High inactivity can cause serious problems in IoT applications that require real-time information preparing and prompt reaction, such as independent vehicles, healthcare, and mechanical robotization [2]. Edge Computing (EC) has emerged as a promising reply to these problems. EC moves computation and data storage to the "edge" of the network which means closer to users and devices and most critically, as close as possible to data sources. The main idea is to reduce the information that must transport from and to the device, in order to decrease inactivity [3]. EC has the potential to essentially boost IoT execution and viability by preparing information, either locally or near to the information source. Much consideration has been paid to the combination of EC with Machine Learning (ML) [4]. ML models are perfect for optimizing situations with EC since they can analyze large packages of information and make predictions or take decisions. This article explores the utilization of three ML models, namely Decision Trees (DTs), Support Vector Machines (SVMs), and Convolutional Neural Networks (CNNs) to work on the latency (time delay between input arrival and corresponding output generation) of EC [5, 6]. DTs work by subdividing information based on real values. This model has been effectively utilized in [7]. SVMs are suitable for classification, safe to over fitting, and work well with large amounts of data [8]. The capacity of CNNs to handle nonlinear data, makes them a perfect choice to use for analyzing time arrangement and other non-visual information. CNNs are exceptionally successful for assignments that require understanding complex designs since they can learn spatial pecking orders from input information [9-11]. The aim of this study is to implement and assess these three models.

II. RELATED WORK

The limitations of conventionally distributed computing, particularly in terms of latency and transmission capacity utilization, make EC look as a fundamental innovation [12]. EC reduces the time required to process and follow up on information by handling it at or close to the source of the information. This is important for continuous applications in the IoT context. This proximity to the data source leads to faster data dealing with and response times, making EC indispensable to reduce latency in fragile applications [13, 14].

A. Role of Edge Computing in Latency Reduction

In past studies, different approaches to decrease latency of EC have been presented. The part of EC within the IoT and the potential benefits of decreasing latency and expanding efficiency were discussed in [15-17]. The authors emphasized the noteworthiness of decentralized data management and the part that smaller-than-expected server clusters, play in moving forward IoT frameworks' that make utilization of EC. Comparable to this, authors in [18, 19] proposed a multitier architecture EC design for applications in smart cities that utilize both cloud and local servers to decrease latency and optimize information handling.

B. Machine Learning for Latency Optimization

DTs are easy to set and apply, since they divide the information recursively into subsets and have been utilized effectively in prediction assignments, counting classification and relapse [20, 25, 26]. SVM and CNN ML models have found broad application and are known for their simplicity and interpretability. SVMs are particularly capable in high-dimensional spaces. SVMs create hyper planes to arrange information of interest, making them sensible for classification [21, 22]. Their capacity to deal with non-linear data contributes to their flexibility and set them suitable for complex datasets [8, 21-24, 27-29].

III. METHODOLOGY

A. Dataset

The data in [30] were utilized.

B. Data Cleaning and Preparation

Stacking of the dataset into a Pandas data frame is the primary step. Pandas is a Python information control library with capable information structures and capacities for controlling time arrangement and numerical tables. With Pandas, each dataset column is changed over to the suitable numerical sorts. This step guarantees that numerical operations and show preparing forms are error- and typo-free. Investigation and data predictions can be off base due to lost values. Strategies like filling lost values with mean, median, or mode, or disposing of lines or segments with lost values were utilized to clean the dataset, thus ensuring it is suitable for analysis and presentation.

C. Exploratory Data Analysis

In order to comprehend the data's structure, transfer and relations, EDA was utilized. Transfer of the different parameters and their relationship to the target variable, were visualized utilizing histograms, diffuse plots, and box plots. The target variable (latency) and the different features' relationships were inspected. The EDA's relationship analysis was utilized to choose parameters that relate directly with latency. The protocols tcp.ack, tcp.len, and udp.time_delta of the internet protocol suite, were recognized as vital parameters. Those parameters were subjected to normalization and scaling, to increase performance.

D. Model Training and Evaluation

Three ML models were considered: DTs, SVMs, and CNNs. The target variable was latency. Accuracy, R-squared (R^2), and Mean Squared Error (MSE) were considered to assess each model's execution. Accuracy measures the extent of the expected values. MSE measures the amount of the normal squared distinction between the real and the expected values. R^2 shows how well the expected values interpret the data's variance.

- Accuracy can be calculated as:

$$\text{Accuracy}=\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (1)$$

where: TP: True Positives, TN: True Negatives, FP: False Positives, FN: False Negatives.

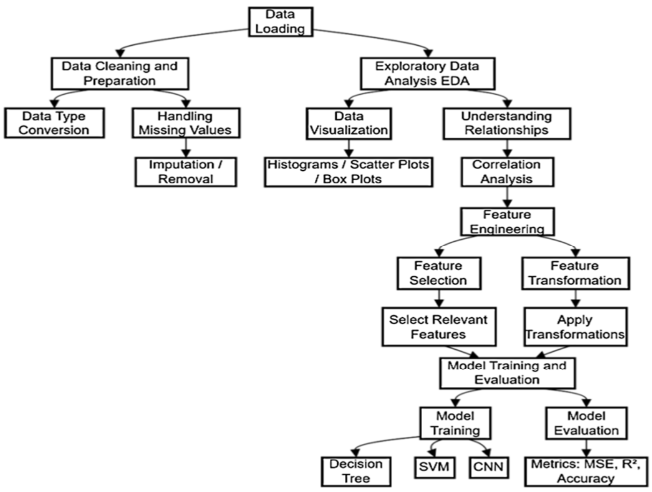

Fig. 1. Methodology block diagram.

- MSE is the average of the squares of the errors—that is, the average squared difference between the predicted values and the actual values**.** It is defined by:

$$\text{MSE}=\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\hat{y}_i\right)^2 \quad (2)$$

where: y_i is the real value, $\hat{y}_i$ is the predicted value, n is the number of observations.

- The proportion of the dependent variable's variance that can be predicted from the independent variables is shown by the R^2 score. It goes from 0 to 1, with higher qualities demonstrating better model execution. It is defined by:

$$R^2=1-\frac{\sum_{i=1}^{n}\left(y_i-\hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i-\bar{y}\right)^2} \quad (3)$$

- Decision Boundary: It is defined by:

$$\text{Margin}=\frac{2}{\|w\|} \quad (4)$$

where w is the weight vector of the decision boundary.

- Loss Function: For classification, the Cross-Entropy Loss is commonly used:

$$L=-\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}y_{i,c}\log\left(\hat{y}_{i,c}\right) \quad (5)$$

where: N is the number of instances, C is the number of classes, $y_{i,c}$ is a binary indicator (0 or 1) if class label c is the correct classification for instance i, $\hat{y}_{i,c}$ is the predicted probability that instance i belongs to class c.

For models like SVM, DT, and CNN, the loss function can be adapted to include latency as a factor.

- Logistic Regression Cost Function (Adapted for Latency):

$$J(\theta)=\frac{1}{m}\times \sum_{i=1}^{m}\left[y^{(i)}\log h_\theta\left(x^{(i)}\right)+\left(1-y^{(i)}\right)\log\left(1-h_\theta\left(x^{(i)}\right)\right)\right]\times(1+LR) \quad (6)$$

where: m is the number of training examples, $y^{(i)}$ is the true label, $h_\theta(x^{(i)})$ is the predicted probability and LR is the latency rate.

- Gradient Descent with Latency. To minimize the cost function, we can use gradient descent. The gradient descent update rule will include the latency factor. The gradient of the Cost Function (Adapted for Latency) is defined by:

$$\frac{\partial J(\theta)}{\partial \theta_j}=\frac{1}{m}\sum_{i=1}^{m}\left[\left(h_\theta\left(x^{(i)}\right)-y^{(i)}\right)x_j^{(i)}\right]\times(1+LR) \quad (7)$$

Figure 1 shows the complete layout for the data collection, data cleaning, model training, and model evaluation process.

IV. RESULTS AND DISCUSSION

Step 1: Initially the dataset was stacked into a CSV-based Pandas data frame. Different IoT device-related measurements and parameters were utilized to compile this set of information. When the initial dataset has a lot of information or some missing values, it requires pre-processing. We used Panda to change over all columns to the numeric sort, change over errors to zeros, and fill in any lost values with zeros. This step confirms the data's cleanliness and that the dataset is ready for the consequent examination.

Step 2: EDA was applied to realize the data's structure and connections. We started by analyzing the dataset's transfer of different parameters and constructed tally plots for each parameter. In expansion, a relationship that showed how different parameters relate to one another was made. A heat-map of the relationship between the target variable, latency, and time was created.

A. Decision Tree Model

DTs are easy to implement. We assessed the model's execution on the test set and got the results for MSE, R^2-score, and accuracy metrics in respect to the number of epochs (epoch: one complete pass of the training dataset through the algorithm).

- The acquired accuracy can be seen in Figure 2. The maximum obtained value was 99.21%.
- The low MSE of 0.008 indicates that the DT model's predictions are very close to the actual latency values. The low error margin indicates high precision and dependability in latency prediction, which is essential for improving edge computing performance.
- The acquired R² score of 0.992 means that 99.2% of the variability in latency can be explained by the features used in the DT model. With such a high score, the model is clearly very successful at capturing the underlying patterns in the data and has significant explanatory power.

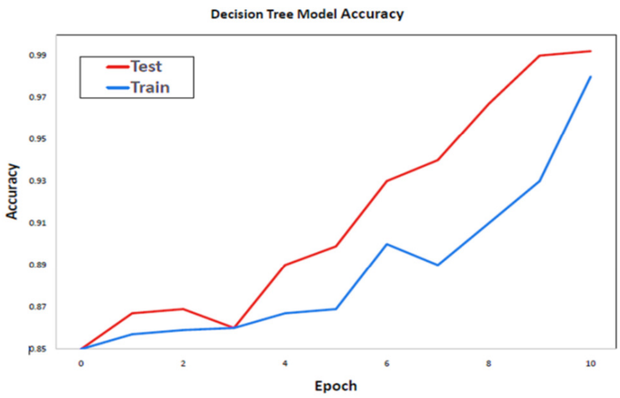

Fig. 2. Accuracy plot of the DT model.

B. SVM Model

Support Vector Regression (SVR) was used for this model. SVR is effective in high-dimensional spaces and works on the same principles as SVM but is customized for regression problems.

- Accuracy: The result of accuracy of the SVM is 88% (Figure 3), indicating that while the SVM model is generally effective, it is less precise than the other models. The relatively lower accuracy suggests that the SVM model's handling of the data's complexity needs to be improved.

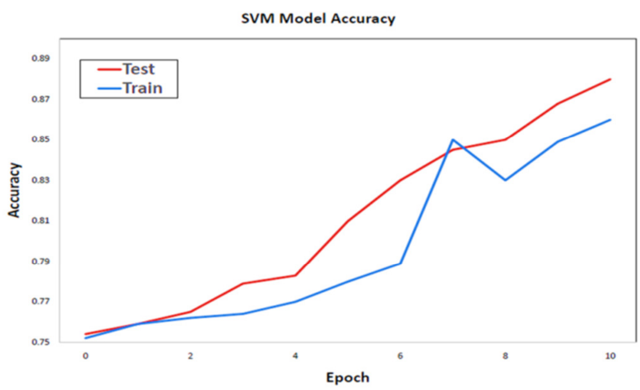

Fig. 3. Accuracy Plotting of SVM Model.

- Decision Boundary can be seen in Figure 4.

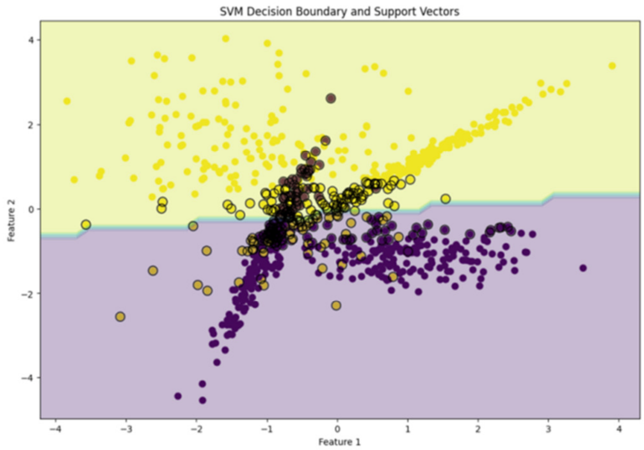

Fig. 4. SVM decision boundary.

- The SVM's predictions deviate more from the actual values than the DT and CNN models, as evidenced by its higher MSE of 0.112.
- The SVM model accounts for 88% of the variability in latency, as evidenced by its R^2 score of 0.880. Even though this has a fair amount of explanatory power, it is significantly lower than the scores obtained by the CNN and DT models. This suggests that the SVM model may be missing some important data patterns.

C. CNN Model

To adopt a CNN model, we reshaped the input data. High-dimensional data patterns and feature recognition are hallmark of CNNs. Keras with TensorFlow backend was utilized to fabricate and prepare the CNN model.

- The CNN model's acquired accuracy is 96.84% (Figure 5). The CNN model is great at monitoring latency with an accuracy of 96.84%, which is marginally less than the DT value. This high accuracy prescribes that the CNN model can be used in IoT EC, preparing circumstances, ensuring a high degree of performance.
- The acquired MSE was 0.031.

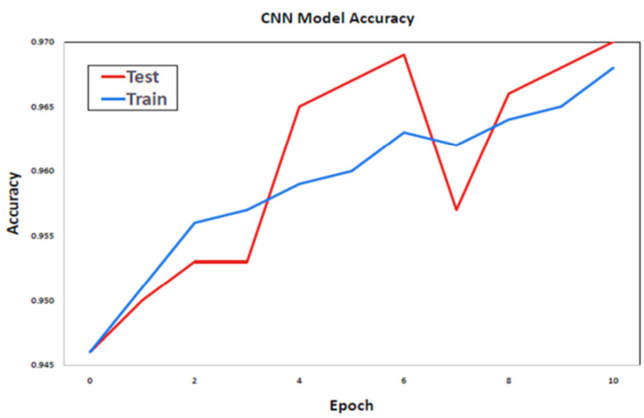

Fig. 5. Accuracy of the CNN model.

- The R^2 score of the CNN model was 0.968. The CNN model's predictions can explain 96.8% of the latency variability.
- The loss function Cross-Entropy Loss Function of the CNN model can be seen in Figure 6.

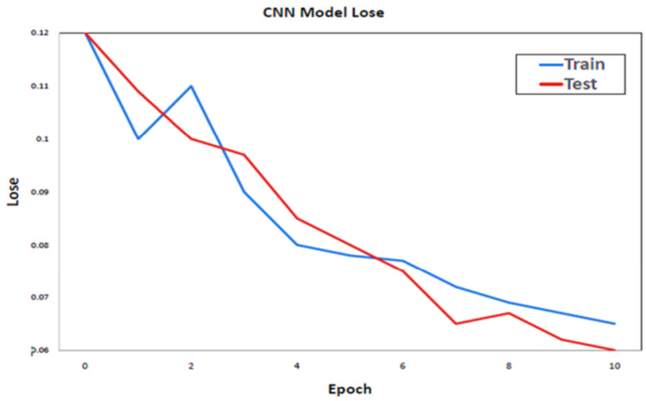

Fig. 6. Loss of the CNN model.

D. Comparative Analysis

The definite examination of each model's presentation features their assets and shortcomings in predicting latency for IoT. Table I includes the basic parameters for each model.

TABLE I. COMPARATIVE ANALYSIS

Model	Accuracy (%)	MSE	R^2 score
Decision Tree	99.21	0.008	0.992
SVM	88.00	0.112	0.880
CNN	96.84	0.031	0.968

- DT Model

Advantages: The DT model has the lowest MSE and the highest accuracy. Its R^2 score suggests that it captures nearly all latency variability, making it a solid option for comprehending and predicting patterns of latency.

Disadvantages: DTs can be prone to overfitting, especially with complex datasets. However, this can be mitigated through techniques such as pruning.

- SVM Model

Advantages: SVM is effective in high-dimensional spaces and is robust to over fitting, particularly in cases where the number of dimensions exceeds the number of samples.

Disadvantages: The SVM model has the larger MSE and the lowest R^2 score. This suggests that the information is more intricately captured. More time may be needed for training, particularly with larger datasets.

- CNN Model

Advantages: The CNN model has a balanced approach, strong prediction performance, low MSE, and high accuracy. It is helpful for comprehending data patterns because, as indicated by its R^2 score, it captures most of the latency variability. The remarkable capability of CNNs to identify spatial hierarchies in data might be advantageous for complex feature extraction.

Disadvantages: CNNs are harder to train and demand more processing resources than DTs and SVMs. They also require a large amount of data to function properly, which might provide challenges in particular circumstances.

V. CONCLUSION

The DT model outperformed the other two models when it came to foreseeing inactivity in IoT Edge Computing (EC). It had the lowest mean squared error of 0.008, the highest R^2 score 0.992, and the highest accuracy of 99.21%. CNN performed also very well, with an R^2 score of 0.968 and an accuracy of 96.84%, proving to be a practical alternative for complex information designs.

REFERENCES

[1] I. Elshair and T. J. S. Khanzada, "hFedLAP: A Hybrid Federated Learning to Enhance Peer-to-Peer," *Engineering, Technology & Applied Science Research*, vol. 14, no. 3, pp. 14612–14618, Jun. 2024, https://doi.org/10.48084/etasr.7331.

[2] N. K. Al-Shammari, T. H. Syed, and M. B. Syed, "An Edge – IoT Framework and Prototype based on Blockchain for Smart Healthcare Applications," *Engineering, Technology & Applied Science Research*, vol. 11, no. 4, pp. 7326–7331, Aug. 2021, https://doi.org/10.48084/etasr.4245.

[3] R. Rajamohanan and B. C. Latha, "An Optimized YOLO v5 Model for Tomato Leaf Disease Classification with Field Dataset," *Engineering, Technology & Applied Science Research*, vol. 13, no. 6, pp. 12033–12038, Dec. 2023, https://doi.org/10.48084/etasr.6377.

[4] S. R. Gopi and M. Karthikeyan, "Effectiveness of Crop Recommendation and Yield Prediction using Hybrid Moth Flame Optimization with Machine Learning," *Engineering, Technology & Applied Science Research*, vol. 13, no. 4, pp. 11360–11365, Aug. 2023, https://doi.org/10.48084/etasr.6092.

[5] M. Y. A. Khan, U. Khalil, H. Khan, A. Uddin, and S. Ahmed, "Power Flow Control by Unified Power Flow Controller," *Engineering, Technology & Applied Science Research*, vol. 9, no. 2, pp. 3900–3904, Apr. 2019, https://doi.org/10.48084/etasr.2587.

[6] H. Khan, I. Din, A. Ali, and M. Husain, "An Optimal DPM Based Energy-Aware Task Scheduling for Performance Enhancement in Embedded MPSoC," *Computers, Materials & Continua*, vol. 74, no. 1, pp. 2097–2113, 2022, https://doi.org/10.32604/cmc.2023.032999.

[7] S. Khan *et al.*, "Green synthesis of AgNPs from leaves extract of Saliva Sclarea, their characterization, antibacterial activity, and catalytic reduction ability," *Zeitschrift für Physikalische Chemie*, vol. 238, no. 5, pp. 931–947, May 2024, https://doi.org/10.1515/zpch-2023-0363.

[8] H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, and Q. Bashir, "Offline Earliest Deadline first Scheduling based Technique for Optimization of Energy using STORM in Homogeneous Multi-core Systems," *International Journal of Computer Science and Network Security*, vol. 18, no. 12, pp. 125–130, Dec. 2019.

[9] R. Waleed *et al.*, "An Efficient Artificial Intelligence (AI) and Internet of Things (IoT's) Based MEAN Stack Technology Applications," *Bulletin of Business and Economics (BBE)*, vol. 13, no. 2, pp. 200–206, Jun. 2024, https://doi.org/10.61506/01.00316.

[10] Q. Noor, A. Ilyas, Z. Javaid, and H. Khan, "Framing a Knowledge Domain Visualization on Green Human Resource Management: A Bibliometric Analysis from 2008-2022," *Pakistan Journal of Humanities and Social Sciences*, vol. 11, no. 4, pp. 4200–4212, Dec. 2023, https://doi.org/10.52131/pjhss.2023.1104.0684.

[11] T. M. Gondal, Z. Hameed, M. U. Shah, and H. Khan, "Cavitation Phenomenon and Its Effects in Francis Turbines and Amassed Adeptness of Hydel Power Plant," in *2019 2nd International Conference on*

Computing, Mathematics and Engineering Technologies (iCoMET), Jan. 2019, pp. 1–9, https://doi.org/10.1109/ICOMET.2019.8673454.

[12] H. Khan, I. Din, A. Ali, and S. Alshmrany, "Energy-Efficient Scheduling Based on Task Migration Policy Using DPM for Homogeneous MPSoCs," *Computers, Materials & Continua*, vol. 74, no. 1, pp. 965–981, 2022, https://doi.org/10.32604/cmc.2023.031223.

[13] M. Shah, S. Ahmed, K. Saeed, M. Junaid, H. Khan, and Ata-ur-rehman, "Penetration Testing Active Reconnaissance Phase – Optimized Port Scanning With Nmap Tool," in *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, Jan. 2019, pp. 1–6, https://doi.org/10.1109/ICOMET.2019.8673520.

[14] M. Y. A. Khan, "A GSM based Resource Allocation technique to control Autonomous Robotic Glove for Spinal Cord Implant paralysed Patients using Flex Sensors," *Sukkur IBA Journal of Emerging Technologies*, vol. 3, no. 2, pp. 13–23, 2020.

[15] M. Y. A. Khan, "A high state of modular transistor on a 105 kW HVPS for X-rays tomography Applications," *Sukkur IBA Journal of Emerging Technologies*, vol. 2, no. 2, pp. 1–6, 2019, https://doi.org/10.30537/sjet.v2i2.475.

[16] H. Khan, S. Ahmed, N. Saleem, M. U. Hashmi, and Q. Bashir, "Scheduling Based Dynamic Power Management Technique for offline Optimization of Energy in Multi Core Processors," *International Journal of Scientific & Engineering Research*, vol. 9, no. 12, pp. 6–10, Dec. 2018.

[17] M. Y. A. Khan, "Enhancing Energy Efficiency in Temperature Controlled Dynamic Scheduling Technique for Multi Processing System on Chip," *Sukkur IBA Journal of Emerging Technologies*, vol. 2, no. 2, pp. 46–53, 2019, https://doi.org/10.30537/sjet.v2i2.477.

[18] H. Khan *et al.*, "An Efficient Scheduling based cloud computing technique using virtual Machine Resource Allocation for efficient resource utilization of Servers," in *2020 International Conference on Engineering and Emerging Technologies (ICEET)*, Oct. 2020, pp. 1–7, https://doi.org/10.1109/ICEET48479.2020.9048233.

[19] A. Hassan, H. Khan, I. Uddin, and A. Sajid, "Optimal Emerging trends of Deep Learning Technique for Detection based on Convolutional Neural Network," *Bulletin of Business and Economics (BBE)*, vol. 12, no. 4, pp. 264–273, Dec. 2023, https://doi.org/10.61506/01.00114.

[20] H. Sarwar, H. Khan, I. Uddin, R. Waleed, and S. Tariq, "An Efficient E-Commerce Web Platform Based on Deep Integration of MEAN Stack Technologies," *Bulletin of Business and Economics (BBE)*, vol. 12, no. 4, pp. 447–453, Dec. 2023, https://doi.org/10.61506/01.00147.

[21] H. Khan, M. R. Usman, B. Ahmed, A. Khan, M. Usman, and S. Ahmed, "Thermal-Aware Real-Time Task Schedulabilty test for Energy and Power System Optimization using Homogeneous Cache Hierarchy of Multi-core Systems," vol. 14, Mar. 2019, https://doi.org/10.26782/jmcms.2019.08.00036.

[22] M. Y. A. Khan, F. Khan, H. Khan, S. Ahmed, and M. Ahmad, "Design and Analysis of Maximum Power Point Tracking (MPPT) Controller for PV System," vol. 14, no. 1, pp. 276–288, Jan. 2019, https://doi.org/10.26782/jmcms.2019.02.0001.

[23] H. Khan *et al.*, "Enhanced Resource Leveling Indynamic Power Management Techniqueof Improvement in Performance for Multi-Core Processors," vol. 14, no. 6, pp. 956–972, Dec. 2019, https://doi.org/10.26782/jmcms.2019.12.00073.

[24] H. Khan, S. Ahmed, N. Saleem, M. U. Hashmi, and Q. Bashir, "Scheduling Based Dynamic Power Management Technique for offline Optimization of Energy in Multi Core Processors," *International Journal of Scientific & Engineering Research*, vol. 9, no. 12, pp. 6–10, Dec. 2018.

[25] M. Y. Ali Khan, M. Ibrahim, M. Ali, H. Khan, and E. Mustafa, "Cost Benefit Based Analytical Study of Automatic Meter Reading (AMR) and Blind Meter Reading (BMR) used by PESCO(WAPDA)," in *2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, Jan. 2020, pp. 1–7, https://doi.org/10.1109/iCoMET48670.2020.9073904.

[26] A. Naz, H. Khan, I. U. Din, A. Ali, and M. Husain, "An Efficient Optimization System for Early Breast Cancer Diagnosis based on Internet of Medical Things and Deep Learning," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 15957–15962, Aug. 2024, https://doi.org/10.48084/etasr.8080.

[27] S. Khan *et al.*, "Inorganic-polymer composite electrolytes: basics, fabrications, challenges and future perspectives," *Reviews in Inorganic Chemistry*, Feb. 2024, https://doi.org/10.1515/revic-2023-0030.

[28] M. S. A. Razak, S. P. A. Gothandapani, N. Kamal, and K. Chellappan, "Presenting the Secure Collapsible Makerspace with Biometric Authentication," *Engineering, Technology & Applied Science Research*, vol. 14, no. 1, pp. 12880–12886, Feb. 2024, https://doi.org/10.48084/etasr.6400.

[29] H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, and A. Saleem, "Performance Evaluation for Secure DES-Algorithm Based Authentication & Counter Measures for Internet Mobile Host Protocol," *IJCSNS International Journal of Computer Network and Information Security*, vol. 18, no. 12, pp. 181–185, Dec. 2018.

[30] M. A. Ferrag, "Edge-IIoTset Cyber Security Dataset of IoT & IIoT." [Online]. Available: https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot/discussion?sort=undefined.