load L^d, and modular load L^m. Therefore, the primary load L^p is uninterruptible and also cannot be deferrable. However, the deferrable load L^d depends on the task load and once the task execution is being started then this type of load cannot be interrupted. Finally, the characteristics of modular load L^m are interruptible and do not require the constant power supply.

Our propose system model is presented in Figure 1. In this system model, we have considered the base station (BS) is connected to the microgrid controller and access points (APs) are connected within the BS. Therefore, each AP consists of the edge server F. Finally, heterogeneous tasks $J = \{1, ..., J\}$ request are generated by the various smart users.

In our problem description, we assume that our system model in the cellular network and also tasks are already received by the edge servers F. In addition, all the edge servers have the equal computational capacity α for tasks execution. Moreover, for the temporal measurement, we consider a finite time horizon set $T = \{1, ..., T\}$ with the uniform length and each T presents with t. Total energy consumption load at time slot t as follows,

$$L_t = \sum_{\forall \in F}(L_t^p + L_t^d + L_t^M) \qquad (1)$$

The value of the primary load follows the constant value for the each and individual edge server and also $\sum_{\forall \in F} L_t^p$ is the total amount of load for all the edge servers under the one BS. Therefore, the deferrable load is $L_t^d = \sum_{\forall \in J} \beta \frac{\gamma}{\alpha}$, whereas β is the energy coefficient and the value of β depends on the edge server capacity. In addition, γ is the amount of total computational request into one edge server at time slot t.

The modular load depends on one binary indicator ω_t where $\omega_t = 1$ indicates the edge server has the modulate load at time slot t and the modular load for all edge server represents as $\sum_{\forall \in F} L_t^m \omega_t$.

The objective of the problem (2) is to minimize the root mean square error between the true value L_t and predicted value h_t.

$$\delta_t = \sqrt{\frac{\sum_{i=1}^{N}(L_t - h_t)}{N}} \qquad (2)$$

Where N is the number of data point at time slot t.

3. Solution Approaches

Our proposed temporal energy demand extrapolation for mobile edge solution architecture is shown in figure 2. We have proposed a three-layer architecture where the first layer is for energy feature extraction from the task requests into edge server. To extracts the primary load energy features we have consider the edge server characteristics for the idle state. However, the deferrable load depends on the tasks request into edge server.

Therefore, the modular load contingents with the status of edge server execution status.

The second layer usages the features, those are the output from the first layer and preprocess these input values into a matrix format. Additionally, preprocessed data are scaled by min-max normalization.

Finally, in the layer three, we have adopted the long short-term memory (LSTM) neural networks model for our solution [6]. That is the optimized version of the recurrent neural network (RNN) and also LSTM handles properly with the vanishing gradient problem.
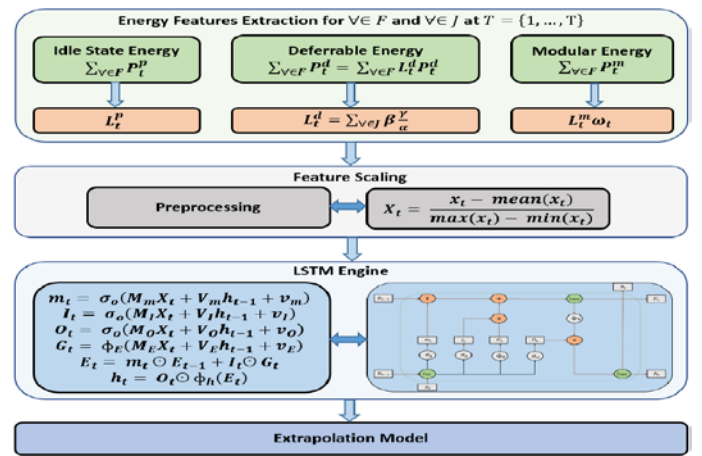

Figure 2: Extrapolation Architecture

Following is the temporal energy demand extrapolation algorithm basic steps,

Algorithm: Temporal Demand Extrapolation
1. Input : Task List J_t , Edge F_t
2. Output: Extrapolation vector h_t
3. **Repeat{**
4.   **Repeat{**
5.   Step 1: Feature Extraction:
6.   Find $\sum_{\forall \in F} P_t^p + \sum_{\forall \in F} L_t^d P_t^d + \sum_{\forall \in F} P_t^m$
7.   Calculate L_t using eq. (1)
8.   Step 2: Preprocessing:
9.   Scale X_t
10.   Step 3: Modelling(LSTM):
11.   Find: $h_t = O_t \odot \sigma_h(E_t)$
12.   } **Until** $\forall J \in F$
13.   } **Until** $\forall F \in B$

4. Performance Evaluation

In this research, we have implemented our proposed model on python platform and also, used the nyupoly/video dataset [7] for performance analysis. However, we have considered the Raspberry Pi 3 Model B as an edge server into one AP and also

we have used 10 APs with one BS. Additionally, we have executed our model for 3000 time slot and each time slot consists of 15 minutes duration.
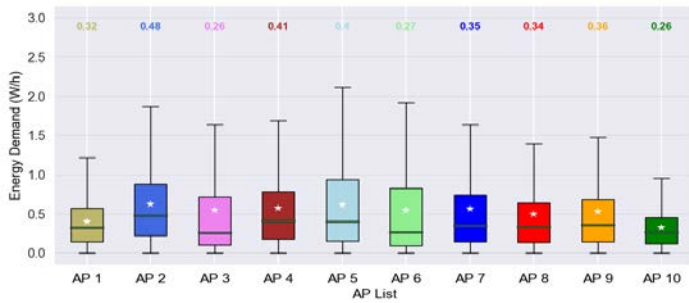

Figure 3: Energy Demand for Individual APs

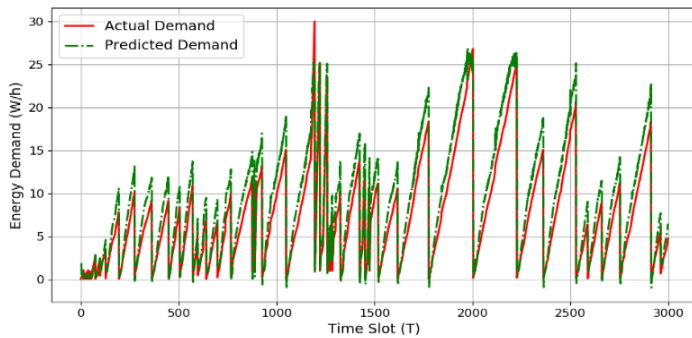Figure 3 depicts the extrapolated energy demand for each AP into 3000 time slot.
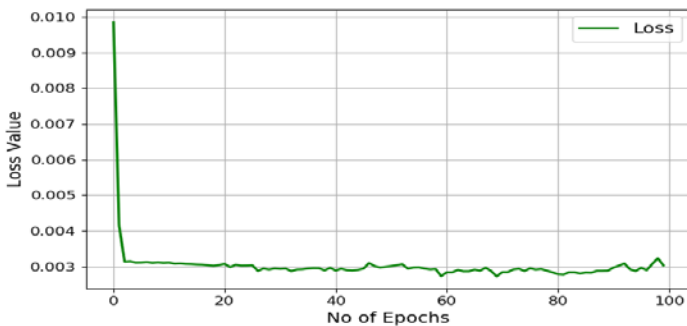

Figure 4: BS Energy Demand Forecasting

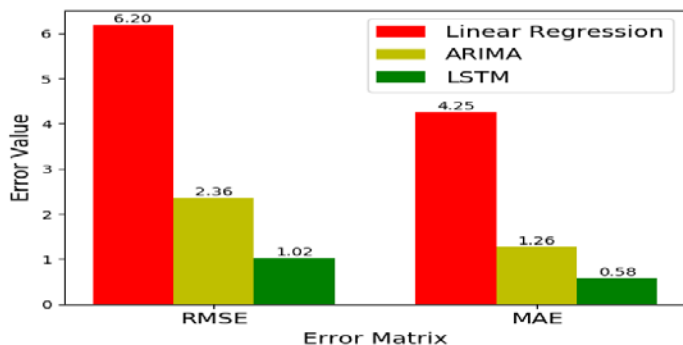Figure 5: Loss Convergence for LSTM Learning

Figure 6: Comparison of Different Model

BS energy demand forecasting for 3000 time slot represents in Figure 4 and it is observed that, the proposed model gain (green dot) higher accuracy for the demand extrapolation. Loss convergence for LSTM learning model is described in Figure 5. Finally, in Figure 6 presents, the root means square error (RMSE) and mean absolute error (MAE) is very less instead of linear regression and ARIMA so that we choose the LSTM model.

5. Conclusion

The temporal energy demand extrapolation for mobile edge based on computational task is a novel and contemporary approach that enables the energy load extrapolation for the network resources under the cellular networks. The proposed method substantially helps for seamless energy management for mobile networks with the smart grid environment. Additionally, this approach will significantly reduce the risk of energy failure due to task load in dynamic cellular networks.

References

[1] M. S. Munir, S. F. Abedin, M. G. R. Alam, N. H. Tran and C. S. Hong, "Intelligent service fulfillment for software defined networks in smart city," *2018 International Conference on Information Networking (ICOIN)*, pp. 516-521, 2018. (in Chiang Mai, Thailand).

[2] S. F. Abedin, M. G. R. Alam, R. Haw and C. S. Hong, "A system model for energy efficient green-IoT network," *2015 International Conference on Information Networking (ICOIN)*, pp. 177 – 182, 2015. (in Cambodia)

[3] M. S. Munir, S. F. Abedin, M. G. R. Alam, D. H. Kim, and C. S. Hong, " Smart Agent based Dynamic Data Aggregation for Delay Sensitive Smart City Services," *Journal of KIISE*, vol. 45, no. 4, pp. 395-402, April 2018.(in Korea)

[4] J. Kelly and W. Knottenbelt, "Neural nilm: Deep neural networks applied to energy disaggregation," *in Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, 2015, pp. 55–64.

[5] K. He, L. Stankovic, J. Liao, and V. Stankovic, "Non-intrusive load disaggregation using graph signal processing*," IEEE Transactions on Smart Grid, 2017*.

[6] M. S. Munir, S. F. Abedin, M. G. R. Alam, D. H. Kim, and C. S. Hong, "RNN based Energy Demand Prediction for Smart-Home in Smart-Grid Framework," *Korea Software Congress 2017*, pp. 437-439, 2017

[7] Online: F. Fund, C. Wang, Y. Liu, T. Korakis, M. Zink, Shivendra Panwar, CRAWDAD dataset nyupoly/video (v. 2014-05-09), downloaded from https://crawdad.org/nyupoly/video/20140509.

[8] Online: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/