Chapter 5

Energy Consumption Prediction of Smart Buildings: A Federated Learning and XAI from Consumer Endpoint

This chapter focuses on the second contribution of this thesis. We provide an introduction and overview of the proposed electricity prediction framework in Section 5.1. The high-dimensional learning (HDL) models utilized for prediction are described in Section 5.2. Section 5.3 demonstrates the proposed work for efficient EM. The experimental setup and results obtained using the proposed technique are presented in Section 5.4. Finally, Section 5.5 summarizes the findings and key outcomes of this work.

5.1 Introduction

Energy efficiency plays a vital role in sustainable EM by ensuring similar or improved output with reduced energy usage [1]. Practices such as utilizing efficient appliances, optimizing industrial processes, and encouraging energy-conscious behavior not only cut emissions but also improve energy security and reduce utility costs. Buildings, accounting for 40–50% of global energy consumption [1, 68–70], present major opportunities for energy savings. SBs, equipped with IoT devices and automation systems, allow real-time monitoring and control of energy systems [71, 72]. These systems adapt to occupancy, environmental conditions, and system performance, helping to reduce unnecessary energy usage. However, forecasting energy consumption (EC) remains complex due to non-linear usage patterns. Traditional models like ARIMA struggle with such variability [5, 73]. To address this, DL models such as LSTMs have shown promise, especially when enhanced through hybrid approaches and optimization algorithms like genetic algorithms or GANs [74–76]. This study proposes a TCN-BiLSTM-based federated learning framework for EC forecasting using multimodal time-series data [77]. Validated on real-world office building data, it outperforms existing models, highlighting the significance of advanced DL architectures in improving smart building EF accuracy.

Key contributions of this study include:

- The use of a Temporal Convolutional Network TCN-Bi-Lstm based DL model for energy consumption forecasting in SBs.
- Evaluation of the model's LT prediction capability.
- Comparative performance analysis using publicly available DSs and metrics like MAE, RMSE, and R^2 score.

5.2 Theoretical Background

This section presents a comprehensive overview of the problem definition and the methodology proposed to address it.

5.2.1 Problems and Motivations

EC forecasting plays a crucial role in the proactive EM and mitigation of greenhouse gas emissions, emphasizing the importance of early warning systems. The main goal is to optimize energy efficiency, enhance occupant comfort, decrease operational costs, and boost productivity, ultimately leading to greater building resilience at various observation points over specific time intervals. The observation period commonly spans fifteen minutes, determined by the sensor systems. Fig. **??** provides a representation of typical EC data.

EC prediction problem is discussed as follows. In this approach, the development of the model relies on using time series (TS) data of EC. Let $X = \{x_1, ..., x_n\}$ represent the EC TS data, spanning a time interval T between two consecutive data points x_i and x_j. Given a time T, the projection model expects the EC value at time $T + 1$ / $T + n$. The prediction model anticipates the output considering various factors, such as air temperature, solar radiation, HVAC operational data, and other relevant multimodal conditions, as shown in Fig. 3.3. Hence, the mentioned data characteristics are crucial to the process of the EC forecasting task.

The pivotal aspect of EC forecasting lies in effectively processing and capturing the spatial-temporal patterns (STP) of the aforementioned EC data elements. Contextual information about the observation sites exists within the EC data and other circumstances, and the development of future trends is influenced by the past condition. In other words, the periodic intervals and adjacent data points in the EC TS data frequently show a strong association with one another.

Moreover, EC data exhibit sharp non-linearities, particularly during transitions from weekdays, weekends, holidays, and working hours. The task of EC forecasting is challenging due to the rapid and unpredictable changes in weather conditions and greenhouse gas emissions directed through a multitude of factors. Moreover, these factors, such as temperature, humidity, and occupants themselves, exhibit non-linear and dynamic behavior. The complexity arises from the highly non-linear interactions among these factors, making precise EC prediction for specific times and locations a difficult endeavor. Given the inherent interdependence among these influencing factors, effectively handling and harnessing this interdependence from the multivariate EC TS data poses another key challenge in EC prediction.

This paper introduces a proposed EC prediction method utilizing a federated learning architecture. Dealing with EC TS data poses challenges due to their varying statistical characteristics, where different TS exhibit distinct representations and related structures. As a result, traditional shallow ML models struggle with fusion modeling in this context. To address this issue, researchers have explored DL models, which have shown promise in enhancing the performance of classic ML models [69, 71, 72, 78].

5.2.2 Model Overview

Fig. 5.1 presents the overall framework of the proposed system, detailing four main tiers and the EC scenario in smart residential sectors. The office building data set used in this study was obtained from Berkeley, California [77]. The building was constructed in 2015. The data set comprises variables that represent instrument-produced data values, which exhibit changes over time. The data accounting was performed from a variety of IoT sensors that are integrated into various office-use equipment that primarily detects properties like temperature, humidity, and solar radiation. IoT sensors continuously monitor the production system provided data points every 1 to 15 minutes. For this study, the sensor data used has already been collected and is readily available for analysis.

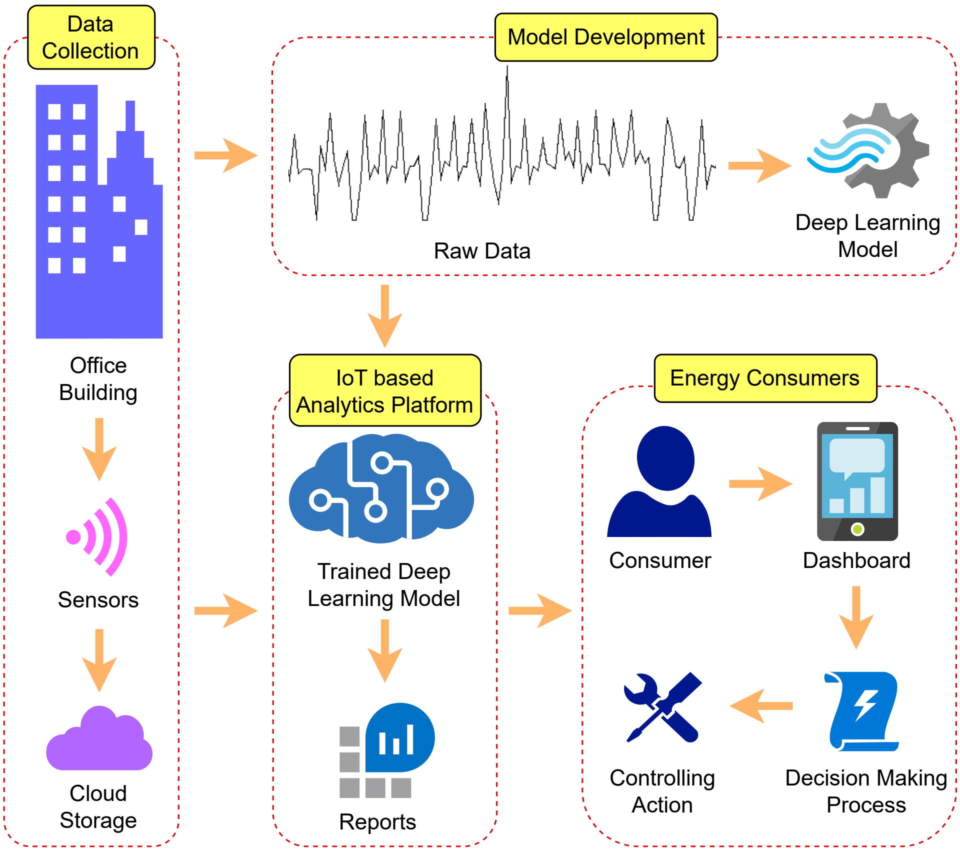


FIGURE 5.1: Architectural overview of proposed system.

The model development tier plays a crucial role in predicting EC and efficiently managing it. To facilitate this process, a cloud server acts as a service provider to consumers. This cloud server stores, analyzes, and relays demands from households, ensuring a proper energy supply to the respective consumers. The IoT-based analytics platform tier is central to the framework, where consumer parties utilize resource-constrained instruments for forecasting their upcoming energy needs. It is significant to highlight that particular information on energy production resources is outside the purview of this article, and we presumptively take for granted that the grid station obtains enough energy from the available resources.

The energy consumers tier has witnessed significant advancements in sensor technology and software, enabling end users to accurately predict their required EC. This capability serves as a valuable preliminary assessment tool for consumers,

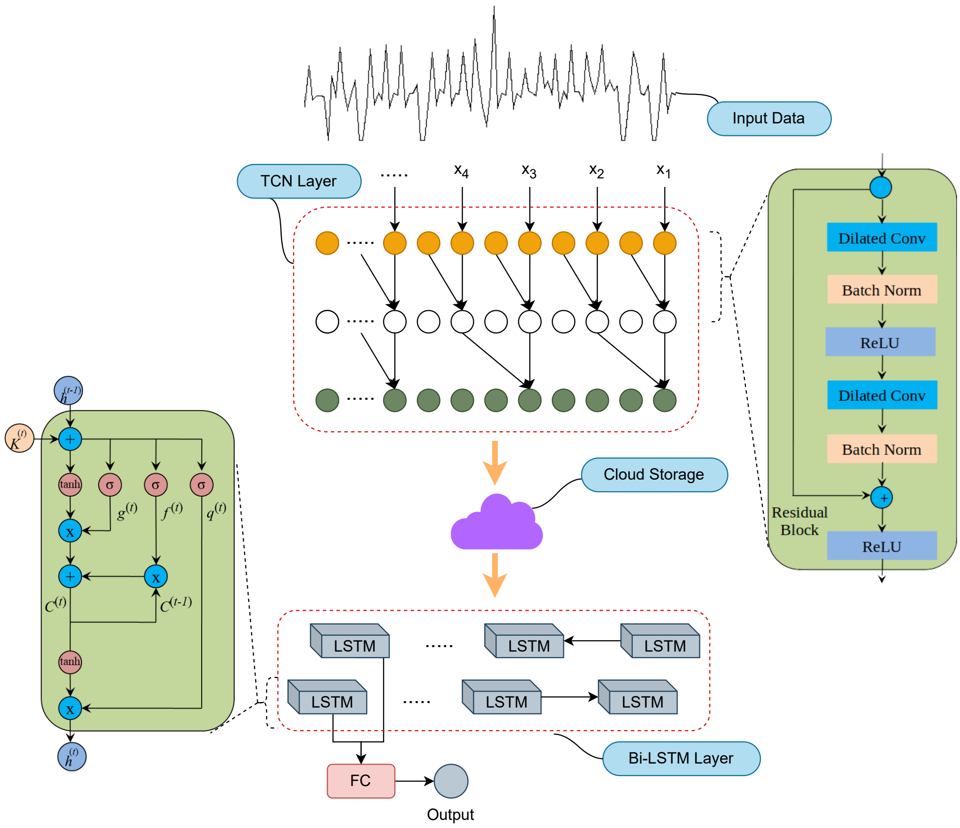


FIGURE 5.2: Proposed hybrid DL model for EC prediction.

allowing them to strategically plan their energy usage based on external and indoor multimodal conditions. By utilizing these predictive capabilities, consumers can have a direct impact on various aspects, including financial aspects, efficient energy-saving management, occupant comfort, and global environmental sustainability. The ability to make informed decisions about energy usage not only helps in optimizing costs but also contributes to a more environmentally sustainable approach to EC. Ultimately, this integration of sensor technology and software empower consumers to play a proactive role in EM and its positive impacts on both a micro and macro scale.

5.3 Proposed Methodology

Our framework offers significant technical contributions, primarily focused on future EC prediction from consumer endpoint. These contributions include achieving a reduced error rate in predictions by developing the federated learning model. Several essential steps are involved to achieve a functional trained model applicable in real-world scenarios. The initial step involves pre-processing the raw data from an existing data set. Subsequently, our novel federated DL mechanism is implemented to obtain the optimized EC prediction model. The proposed model process is detailed below, enabling the model to adapt and improve iteratively.

We present the EC prediction framework, which is based on an proposed architecture. It combines TCN 4.2.1 and Bi-LSTM to account for the spatial-temporal dependence in EC TS data. Due to the interrelationship between local patterns and the LD within the EC TS data, the EC TS is also influenced by other multimodal conditions data, forming inherent interdependencies. Fig. 5.2 provides a graphical illustration of the EC forecasting framework. The overall model comprises two main components: (1) TCN: These layers are responsible for learning high and low-frequency correlation features from the TS data of different wings. (2) Bi-LSTM: This component extracts LD temporal attributes from the respective TS data. The combination of these components allows proposed model to effectively capture and leverage the patterns present in the EC data of various wings, resulting in improved EC predicting performance of building.

In order to capture the dependency features present in EC TS data of different wings, the initial step involves training the TCN for each wing. These TCNs are utilized to extract high and low-frequency information patterns and identify potential interrelationship features from the TS data of multimodal conditions of various wings. Our model receives inputs as multiple 1D TS. Rather than individually learning the features of each TS, we collectively learn from all the TS data across multiple conditions. This approach ensures that we efficiently exploit the various patterns and dependencies among the EC data for improved prediction accuracy.

Subsequently, the extracted features, comprising the high and low-frequency information trend features and the potential correlation features from multivariate EC TS data, are concatenated at server. This combined feature set is then passed as input into specific Bi-LSTMs. These bi-LSTMs are capable of learning STP from both past and future contexts by processing the TS data in both forward and backward directions simultaneously. This bidirectional processing enhances the model's ability to capture and leverage the temporal relationships within the data, leading to more comprehensive and accurate EC prediction. With the input data X, the process of acquiring dependency features from the multivariate TS data can be given as follows:

$$\psi(X) = \psi_K\left(\cdots\psi_2\left(\psi_1\left(X, \gamma^{(1)}\right), \gamma^{(2)}\right)\cdots\gamma^{(K)}\right) \rightarrow \varpi \tag{5.1}$$

In this representation, ψ represents the different operations performed by the TCN at each stage K, and γ represents the TCN's learning parameters for each stage K. The TCN generates the output ϖ, i.e., high and low-frequency information trend features. Similar to Eq. 5.1, Bi-LSTM learns ϖ with ϕ and ζ, i.e., various operations and learning parameters of Bi-LSTM. It is expressed as

$$Bi - LSTM(\varpi, \phi, \zeta) \rightarrow O_t \tag{5.2}$$

The objective function of the proposed technique training can be expressed as follows:

$$\underset{\Theta}{argmin}\Gamma = \frac{1}{m}\sum_{a=1}^{l}\|\hat{y_a} - y_a\| \tag{5.3}$$

The ultimate goal of proposed model training is to minimize the overall error Γ of training samples. In this context, a denotes the number of input samples within a time window, and Θ is the learning parameter of the layer. In this manner, the proposed model integrates TCN and bi-LSTM into an federated learning architecture. This unified framework allows for the simultaneous extraction of both the high and low-frequency pattern and the STP from the multivariate TS data related to EC.

5.4 Experimental Setup and Result Discussion

The models are evaluated and compared using various window sizes, specifically 4, 12, 20, and 28, along with different time stamps represented in hours, namely 1, 3, 5, and 7. For each time stamp, the best-performing model is selected. This comparison allows us to determine the most effective model for different temporal contexts. Various types of statistical, DL, ML, hybrid-DL (HDL), federated learning algorithms are applied in the analysis. To compare the models' performance, three commonly used metrics are considered: MAE, RMSE, and R-squared (R2). The data set is partitioned in such a way that 90.0% of the data is treated as training data, while the remaining 10.0% is allocated as testing data. The experimental results were obtained on a compute node equipped with a 2.60GHz Intel(R) Core(TM) i7-10750H CPU, which has 6 cores, and 8 GB of RAM.

5.4.0.1 Univariate ML model

Table 5.1 shows the performance of three different models (LR, ARIMA, and ANN) in terms of three metrics (MAE, RMSE, and R-squared) at different time horizons (1 hour, 3 hours, 5 hours, and 7 hours). At the all-time horizon, ANN achieved the lowest MAE, followed closely by ARIMA, and then LR. However, at the 1-hour time horizon, MAE value of ANN is followed closely by ARIMA, and then LR. Similarly, at all time horizons, ANN consistently achieved the lowest RMSE values, indicating its better performance compared to the other models. At the 1-hour and 5-hour time horizon, ANN had the highest R-squared value, followed by ARIMA, and LR. At the 3-hour and 7-hour time horizons, LR consistently achieved the highest R-squared values, indicating its better fit to the data compared to the other models.

Overall, the performance comparison of the models varies depending on the specific metric and time horizon. ANN tends to perform well in terms of R-squared, while ARIMA shows competitive performance in both MAE and RMSE. LR generally exhibits the least favorable results among the three models, but its performance is still close to the other models in some cases. Among all the methods, ANN performance varies significantly with various time horizons for different metrics.

5.4.0.2 Univariate DL model

The comparative analysis of the results for the different DL models (RNN, LSTM, Bi-LSTM, and TCN) is provided in the Table 5.1 using various performance metrics and time horizons. Except for the 3-hour time horizon, TCN achieved the lowest MAE, followed by Bi-LSTM, LSTM, and RNN. At the 3-hour time horizon, Bi-LSTM achieved the lowest MAE, followed by LSTM, TCN, and RNN. It is also observed that the MAE value has decreased due to DL models by at least 9.7%, 7.3%, 3.1%, and 4.49%, respectively, for time horizons of 1, 3, 5, and 7 hours compared to ML models. Besides, TCN has reduced the MAE value by at least 5.7% compared to DL sequential models like Bi-LSTM, LSTM, and RNN for all the time horizons. Among DL sequential models, Bi-LSTM performs better on 1-hour, 3-hour, and 7-hour time horizons, while LSTM is superior in the 5-hour time horizon. But, LSTM and Bi-LSTM MAE values are very close compared to RNN.

At the 1-hour and 5-hour time horizons, TCN had the lowest RMSE value, followed by Bi-LSTM, LSTM, and RNN. Hence, TCN has reduced the RMSE value by at least 12.9% compared to DL models. Similarly, at the 3-hour and 7-hour time horizons, Bi-LSTM had the lowest RMSE value, followed by TCN, LSTM, and RNN. It drops the RMSE by approximately 1.5% compared to LSTM, TCN, and RNN. Like MAE, the DL method reduces the RMSE value in comparison to ML models in all time horizons, and it is approximately 1.7%. Similar to the RMSE

Table 5.1: Lag values of univariate models with error metrics.

Error Metric	**Models**	**Lag**			
		1hour	3hour	5hour	7hour
	LR	8.74512	8.53271	8.86124	8.11642
	ARIMA	8.4851	8.13512	8.43250	7.991
	ANN	8.21412	7.5853	8.181	7.86431
	RNN	7.41153	7.02414	7.92416	7.5112
	LSTM	6.64989	5.95055	7.04188	7.81044
MAE	Bi-LSTM	6.47734	5.67300	6.69240	5.80187
	TCN	4.43684	6.67421	5.48321	5.46874
	CNN-LSTM	5.24025	6.38688	7.31434	6.25339
	TCN-RNN	7.54041	6.02413	8.49475	5.97155
	TCN-LSTM	4.225	5.66571	5.48672	4.47624
	TCN-Bi-LSTM	4.44217	4.79700	5.44283	**4.10695**
	LR	11.0125	10.4257	10.4581	10.13282
	ARIMA	10.9151	11.7033	10.7450	9.695
	ANN	10.811	10.0812	10.0321	9.2602
	RNN	9.6953	10.4489	9.3073	9.4214
	LSTM	8.4225	8.0277	9.0937	9.8971
RMSE	Bi-LSTM	8.3654	7.1703	9.5781	7.1340
	TCN	6.3241	8.1634	8.3420	7.2471
	CNN-LSTM	7.1709	8.9409	9.3660	8.7516
	TCN-RNN	9.0866	7.9761	10.8916	7.3169
	TCN-LSTM	6.2470	6.8141	7.3157	5.7747
	TCN-Bi-LSTM	6.1449	6.1208	7.0228	**5.418**
	LR	0.795	0.827	0.821	0.796
	ARIMA	0.811	0.784	0.81	0.788
	ANN	0.825	0.802	0.84	0.784
	RNN	0.864	0.82	0.863	0.793
	LSTM	0.88	0.876	0.863	0.822
R^2	Bi-LSTM	0.872	0.917	0.867	0.918
	TCN	0.922	0.879	0.883	0.861
	CNN-LSTM	0.918	0.877	0.866	0.829
	TCN-RNN	0.865	0.918	0.827	0.881
	TCN-LSTM	0.928	0.93	0.915	0.958
	TCN-Bi-LSTM	0.925	0.927	0.911	**0.952**

value, the R-squared value of DL models has shown a matching trend, and both TCN and Bi-LSTM are competing to achieve good performance.

Overall, the performance comparison shows that TCN and Bi-LSTM consistently outperformed the other models across most time horizons and metrics. TCN achieved the lowest MAE and RMSE values and the highest R-squared values, indicating its overall superior performance. Also, noteworthy deviation in the performance between various time horizons is observed for TCN compared to other DL methods for different metrics. Bi-LSTM also demonstrated competitive performance, especially in terms of R-squared. LSTM performed well in R-squared but outperformed TCN and Bi-LSTM in terms of MAE and RMSE. RNN achieved the lowest performance among the four models in most cases.

5.4.0.3 Univariate HDL model

Based on the results obtained from the individual DL models for predicting EC TS data, several hybrid models are proposed in this work to further enhance performance prediction. By designing and implementing these hybrid models, we aim to achieve more accurate and reliable EC forecasts, taking advantage of the complementary characteristics of individual models. Table 5.1 shows a comparative analysis of the different hybrid models (CNN-LSTM, TCN-RNN, TCN-LSTM, and TCN Bi-LSTM) using multiple time horizons and performance metrics. At the 1-hour time horizon, TCN-LSTM achieved the lowest MAE and slightly superior to TCN-Bi-LSTM. At the 3-hour, 5-hour, and 7-hour time horizon, TCN-BiLSTM achieved the lowest MAE, followed by TCN-LSTM, CNN-LSTM, and TCN-RNN. However, TCN-Bi-LSTM displays the dominant performance in RMSE and R-squared compared to other hybrid models.

Overall, the performance comparison shows that TCN-LSTM and TCN Bi-LSTM consistently outperformed the other hybrid models across most time horizons and metrics. TCN-Bi-LSTM achieved the lowest MAE and RMSE values and the highest R-squared values, indicating its overall superior performance. Hybridizing

TCN and Bi-LSTM models exploits TCN's effective local modeling of temporal dependencies along with the Bi-LSTM's strong LT memory retention. TCN-LSTM also demonstrated competitive performance, especially in terms of MAE and RMSE values. CNN-LSTM performed well but was outperformed by TCN-based models in most cases. TCN-RNN achieved the lowest performance among the four hybrid models in most cases. Despite its hybrid architecture, the CNN-LSTM model did not surpass the performance of DL models like Bi-LSTM.

5.4.1 Multivariate ML model

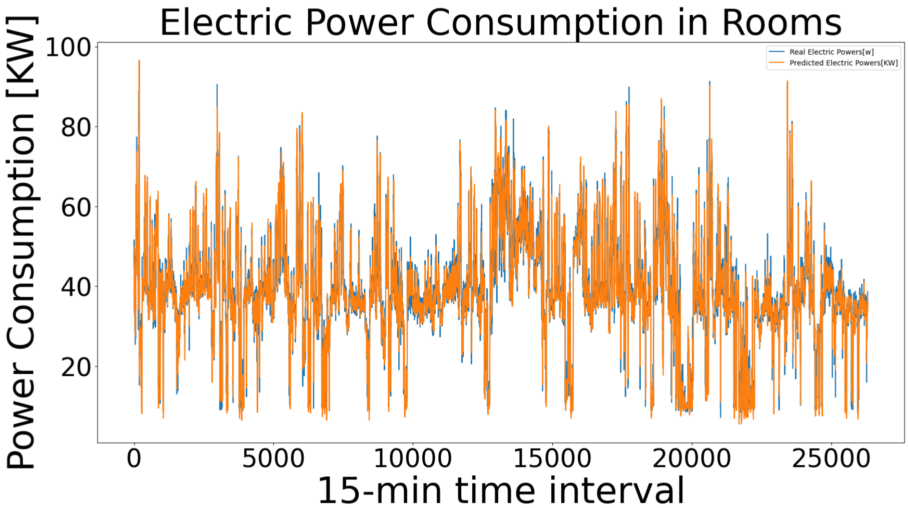
Table 5.2 discuss the comparative analysis of multivariate data set of EC. At the all time horizon, ANN achieved the lowest MAE, followed by ARIMA, and then LR. The similar trend is observed for RMSE and R-squared metrics. LR exhibits a significant variation in MAE and RMSE performance across all time horizons. Overall, the performance comparison shows that ANN consistently outperformed the other models across most time horizons and metrics. The MAE performance of the EC prediction model improved significantly, with an enhancement of at least 13.0% and 12.0% for the 1-hour and 5-hour time horizons, respectively, when using multivariate EC data compared to univariate EC data. Similar kind of enhanced is observed with RMSE and R-squared metrics.

5.4.1.1 Multivariate DL model

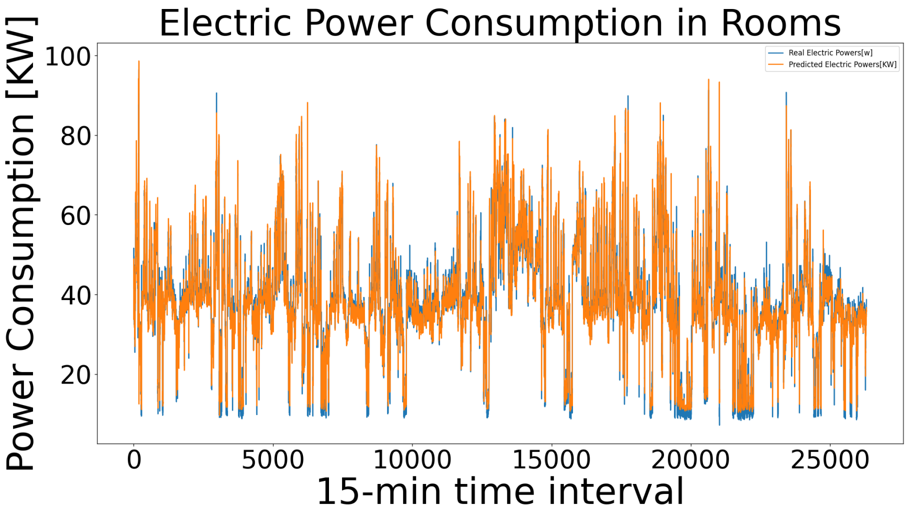
Except for 3-hour time horizon, TCN achieved the lowest MAE, followed by Bi-LSTM, LSTM, and RNN. At the 3-hour time horizon, Bi-LSTM achieved the lowest MAE, followed by LSTM, TCN, and RNN. Similar trends are observed for the RMSE and R-squared metrics as well. Overall, the performance comparison shows that TCN consistently outperformed the other models across most time horizons and metrics. TCN achieved the lowest MAE and RMSE values and the highest R-squared values, indicating its overall superior performance for EC forecasting. TCN outperforms the DL sequential model by at least 5.8% in terms of MSE, 7.8%

Table 5.2: Lag values of multivariate models with error metrics.

Error Metric	**Models**	**Lag**			
		1hour	3hour	5hour	7hour
MAE	LR	7.14215	8.24051	7.16502	7.86114
	ARIMA	6.975	7.92514	7.1042	7.121
	ANN	6.8102	7.331	6.565	6.86725
	RNN	6.22614	6.41681	6.01244	6.38657
	LSTM	5.42614	4.4275	6.88537	5.74255
	Bi-LSTM	5.39512	4.201	5.3172	5.05974
	TCN	3.6617	5.2041	5.00452	4.09775
	CNN-LSTM	4.821	5.4224	6.78438	6.82599
	TCN-RNN	6.9426	5.74359	7.82275	5.13301
	TCN-LSTM	3.1385	4.3182	4.96758	3.61358
	TCN-Bi-LSTM	3.8158	3.8862	4.66857	**3.2016**
RMSE	LR	10.0074	11.361	10.1792	9.1481
	ARIMA	9.222	9.7793	9.9498	8.805
	ANN	8.4345	9.379	9.595	8.4705
	RNN	8.3625	7.4022	9.2574	8.121
	LSTM	6.8601	6.6709	8.5607	6.39
	Bi-LSTM	7.2083	5.961	7.4296	7.0124
	TCN	5.261	6.5183	6.8465	6.3301
	CNN-LSTM	7.6434	7.8437	9.6364	8.4135
	TCN-RNN	7.4875	7.8851	9.1351	7.5585
	TCN-LSTM	5.3082	5.9533	6.0205	5.2585
	TCN-Bi-LSTM	5.0179	5.3207	5.226	**4.2051**
R^2	LR	0.821	0.752	0.832	0.861
	ARIMA	0.85	0.85	0.836	0.87
	ANN	0.873	0.84	0.85	0.886
	RNN	0.887	0.926	0.859	0.892
	LSTM	0.93	0.911	0.873	0.928
	Bi-LSTM	0.917	0.942	0.915	0.885
	TCN	0.946	0.924	0.922	0.934
	CNN-LSTM	0.921	0.906	0.852	0.884
	TCN-RNN	0.928	0.917	0.86	0.926
	TCN-LSTM	0.943	0.95	0.937	0.941
	TCN-Bi-LSTM	0.948	0.951	0.947	**0.968**



(a)



(b)

FIGURE 5.3: (a)
Actual v/s predicted graph using TCN-Bi-LSTM on univariant TS data and (b) Actual v/s predicted graph using TCN-Bi-LSTM on multivariant TS data.

in terms of RMSE, and 0.7% in terms of R-squared. Bi-LSTM and LSTM also performed competitively, demonstrating strong predictive capabilities. RNN achieved relatively lower performance compared to the other models in most cases. Indeed, both TCN and Bi-LSTM models using multivariate EC TS data have demonstrated substantial improvements compared to the univariate approach. Specifically, TCN has consistently achieved R-squared values greater than 0.92 across all time horizons, indicating its superior performance in capturing the variations in the EC data. These results highlight the effectiveness of leveraging multivariate information and the capabilities of TCN and Bi-LSTM models in enhancing the accuracy of EC forecasting.

5.4.1.2 Multivariate HDL model

Except for the 1-hour time horizon, TCN-Bi-LSTM achieved the lowest MAE, followed by TCN-LSTM, CNN-LSTM, and TCN-RNN. At the 1-hour time horizon, TCN-LSTM achieved the lowest MAE, followed by TCN-Bi-LSTM, CNN-LSTM, and TCN-RNN. At the all-hour time horizon, the models' RMSE scores ranked from lowest to highest as follows: TCN-Bi-LSTM, TCN-LSTM, CNN-LSTM, and TCN-RNN. A similar pattern is observed for the R-squared value, and the best R-squared value of 0.968 is achieved for the 7-hour horizon on multivariate TS data. Also, the best R-squared value of 0.952 is attained for a 7-hour horizon on univariate TS data. Thus, overall improvement of 1.65% has been realized by moving from univariate to multivariate TS data. Overall, the performance comparison shows that TCN Bi-LSTM consistently outperforms the other models across most time horizons and metrics. TCN Bi-LSTM achieved the lowest MAE and RMSE values and the highest R-squared values, indicating its overall superior performance for EC forecasting. TCN-LSTM and TCN-RNN also performed competitively, demonstrating strong predictive capabilities. CNN-LSTM achieved relatively lower performance compared to the other models in most cases.

Fig. 5.3 compares the actual EC values with the corresponding predicted values using the TCN Bi-LSTM model in different scenarios. By comparing Figs. 5.3(a) and 5.3(b), we can assess the performance of the TCN-Bi-LSTM model in two different settings: univariate and multivariate. This comparison enables us to evaluate the model's ability to capture patterns and make accurate EC predictions based on the multivariate input data configuration. The analysis reveals a strong correlation

Table 5.3: Lag values of federated learning models with error metrics.

Error Metric	Models	Lag			
		1hour	3hour	5hour	7hour
MAE	CNN-LSTM	4.01159	7.0579	5.3215	7.2111
	TCN-RNN	4.3281	7.2411	6.46118	5.2214
	TCN-LSTM	4.0016	3.8513	4.02248	3.88621
	TCN-Bi-LSTM	3.8815	4.11758	3.68628	**2.3316**
RMSE	CNN-LSTM	6.3082	9.1513	7.2183	8.8135
	TCN-RNN	5.9205	9.9164	7.5011	6.4351
	TCN-LSTM	5.3085	5.3112	4.9631	5.3219
	TCN-Bi-LSTM	5.0941	4.7231	5.1123	**3.5412**
R^2	CNN-LSTM	0.93	0.861	0.915	0.893
	TCN-RNN	0.937	0.868	0.926	0.934
	TCN-LSTM	0.952	0.945	0.96	0.951
	TCN-Bi-LSTM	0.957	0.956	0.962	**0.976**

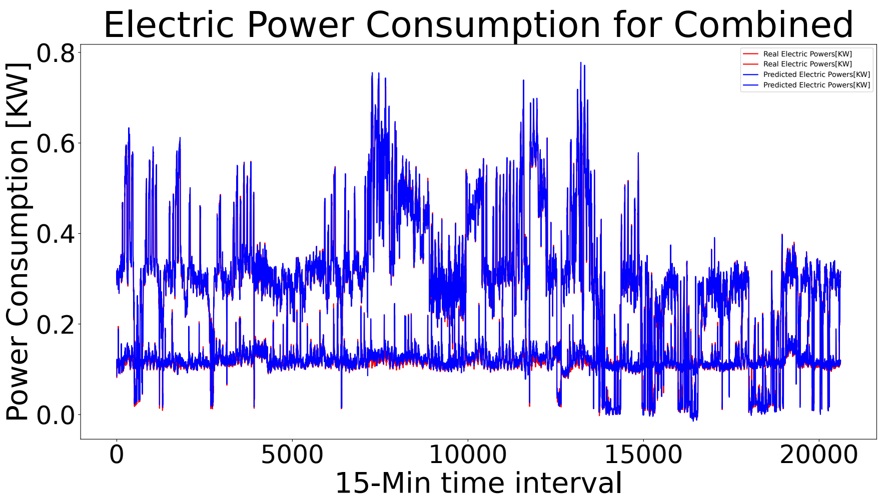


Figure 5.4: Actual v/s predicted graph using federated learning approach of TCN-Bi-LSTM.

between the projected values and the actual values, indicating that the proposed model exhibits good performance.

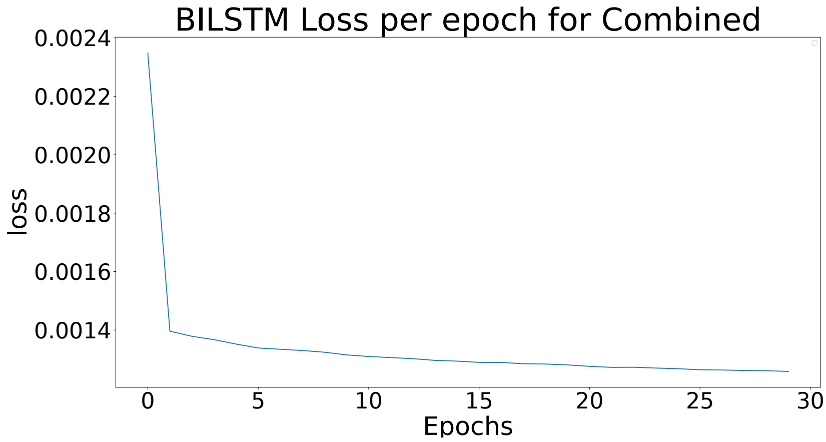


FIGURE 5.5: Loss per epoch values using Bi-LSTM model.

5.4.1.3 Federated learning model

Only HDL models were used for development of federated learning model from section **??**. When examining the MAE and RMSE values, TCN-Bi-LSTM consistently demonstrates the lowest values across all time horizons, as shown in Table 5.3. Notably, at the 7-hour time horizon, TCN-Bi-LSTM achieves the lowest MAE of 2.3316 and RMSE of 3.5412, indicating its superior predictive accuracy. Following this trend, TCN-LSTM also performs well, showing competitive results across all time intervals. CNN-LSTM and TCN-RNN, however, exhibit comparatively higher MAE values, with TCN-RNN showcasing relatively better performance than CNN-LSTM, especially at the 5-hour and 7-hour time horizons.

TCN-Bi-LSTM consistently stands out with the highest R-squared values among all models and time horizons. Notably, at the 5-hour and 7-hour intervals, TCN-Bi-LSTM achieves an R-squared of 0.962 and 0.976, respectively. Thus, overall improvement of 0.8% has been realized by moving from multivariate DL to federated learning approach. TCN-LSTM also maintains strong R-squared values, closely following TCN-Bi-LSTM's performance. CNN-LSTM and TCN-RNN, while still displaying competitive R-squared values, exhibit slightly lower scores compared to the TCN-based models.

Fig. 5.4 illustrates that even during instances of unexpected variation in EC, the projected EC values closely align with the actual EC values. This observation underscores the superiority of our suggested federated learning model compared to competing approaches, as it accurately captures fluctuations and provides reliable predictions even under challenging conditions. Fig. 5.5 represents the change in the model's loss function over successive epochs during the training process. The graph usually starts at a relatively high value for the loss in the initial epochs, indicating that the model's predictions are far from accurate. As the training progresses, the loss typically decreases, signifying that the model is learning and improving its predictions and eventually stabilizes.

5.4.2 Explainable artificial intelligence (XAI) analysis

In the final step of our analysis, we employed the XAI model LIME (Local Interpretable Model-agnostic Explanations) [?] to investigate the contribution of different wings to the overall EC in the building. LIME is a popular method used to interpret the predictions of complex ML models. It provides local and interpretable explanations for individual predictions, allowing us to understand which features or variables are influencing the model's output. Upon applying LIME to HDL model, we discovered that the HVAC_S exhibited a higher EC compared to other wings in the building. This insight is further supported by the visualization shown in Fig. 5.6, where the HVAC_S is represented with a positive value. This positive value suggests that the HVAC_S has a significant impact on contributing to the higher EC in the building.

Overall, using LIME as an XAI tool provided us with valuable insights into the relative contribution of each wing to the total EC, highlighting the prominent role played by the HVAC_S in driving the higher EC in the building. This information can be useful for building management and energy optimization strategies.

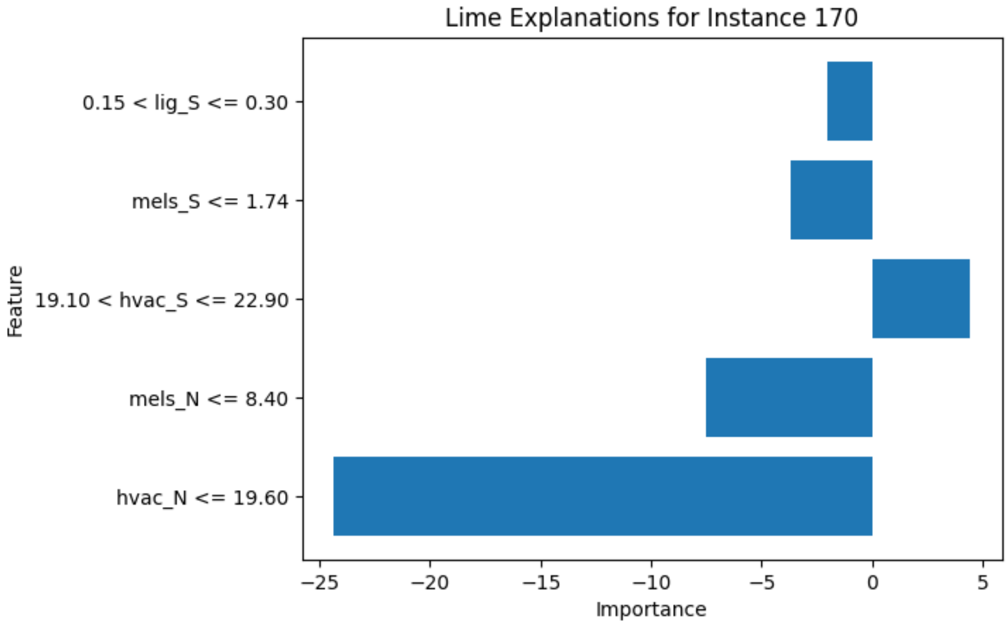


FIGURE 5.6: LIME model for EC prediction.

5.5 Summary and Conclusion

In conclusion, the increasing cost of electricity and the depletion of natural resources have become significant challenges in the generation of electricity. To address these issues and promote energy-efficient practices, the development of smart energy-efficient buildings has become a crucial focus. Many countries are actively monitoring and seeking ways to minimize energy costs while promoting environmental sustainability. In this paper, we have proposed various methodologies and their combinations for predicting energy consumption based on time series data. Through extensive testing on different time lags, we have found that our proposed model performs optimally with a 15-minute lag for different wings. This demonstrates the effectiveness of our approach in forecasting energy consumption accurately. Looking ahead, we have explored various federated learning algorithms in our research to uncover more hidden features and enhance the accuracy of energy consumption predictions. Through the use of these predictive capabilities, consumers gain the power to directly influence various critical aspects, such as cost control, systematic energy-saving strategies, occupant comfort, and global environmental sustainability.

With access to informed insights on energy usage, individuals and organizations can optimize costs and adopt a more environmentally sustainable approach to energy consumption.

Additionally, expanding data sets to include diverse electrical products and scenarios will further improve the algorithms' ability to anticipate energy use in SBs. By continually refining and innovating our predictive models, we can contribute to the development of more efficient and sustainable energy practices, leading to a brighter and more environmentally conscious future.