Scalable and Lightweight Machine Learning Based Load Forecast: Netload versus Disaggregrated Forecast

Alexander N. Ndife
Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden
nnamdi@chalmers.se

David Steen
Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden
david.steen@chalmers.se

Le Anh Tuan
Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden
tuan.le@chalmers.se

***Abstract*— This paper develops lightweight and adaptive demand forecast models for a residential building integrated with solar photovoltaics using scalable and adaptive deep learning algorithms, i.e., long short-term memory (LSTM) and gated recurrent units (GRU). First, the forecast models have been trained using the real measurement data from a residential building. Then, the models have been used in case studies using the real-time data for two forecasting approaches: i) netload forecast; ii) disaggregated forecasts, i.e., forecasting the load and PV generation separately. The performance of the two forecasting approaches have been compared. The results from case studies showed that disaggregated forecast approach was superior (with an overall RMSE of 2.03 kW for the building with max demand of 10.53 kW) than the aggregated forecast approach (with an overall RMSE of 2.63 kW). Case studies results have also demonstrated that the models are scalable with more data, and are lightweights, hence, suitable for resource-constraint devices. Although LSTM shows advantages in accuracy, GRU shows better scalability in terms of computational efficiency. The models can be utilized by various stakeholders, such as building owners, grid operators, etc., and can be adapted to other types of buildings.**

Keywords*— *forecasting, netload, lightweight, scalability, neural networks, deep learning, energy optimization

I. INTRODUCTION

A. Background and Motivations

Demand forecasting for buildings is becoming one of the important elements in the energy management system of buildings. It is more so when more renewable energy sources (e.g., solar photovoltaics - PV), new types of loads such as heat-pumps, battery energy storages, electric vehicles, etc., are being integrated to buildings. Accurate load forecasts could lead to a reduction in both operational and maintenance costs for buildings. From the end-users' perspective, reliable forecasts will help in daily energy management and scheduling with potential cost and energy saving. Therefore, there is need to develop advanced demand forecast model in such way that potential high volatility and uncertainty associated with energy loads and building integrated PVs could be addressed. Forecasting of load demand in PVs integrated buildings can be done by forecasting the netload or by forecasting load demand and PV production separately. The main question here is which approach would be better and in which conditions.

B. Related Work

Efforts have been made in load forecasting but not much in the direction of netload. Several forecasting strategies exist for individual components of the netload and the netload itself [1]. However, it is not clear whether aggregating several forecasts to obtain the netload is more beneficial or to aggregate the input data to forecast the netload with one approach directly. With netload, the knowledge of the appropriate time to apply balancing efforts between demand and supply is known. Fig. 1 is an illustration of a building's netload.

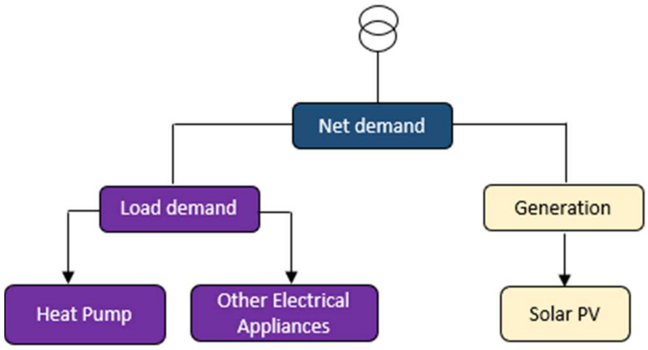

Fig. 1. Representation of a building's load and generation

Recognizing the importance of individual electric users in grid planning and operation, [2] carried out a user level load forecasting to analyze integrated energy system influential factors and proposed a new user level load forecast method. [3] made a short-term individual households forecast and added up the resulting values as aggregated users in a large-scale prediction. [4] used data-driven approach to decompose the netload for easy modeling and forecasting. A blend of these forecast strategies is implemented in [5], focusing on both single household and low aggregate levels loads.

1) Individual components forecasting

This approach implies forecasting netload indirectly, that is, forecasting PV production and electric load individually, then subtract the PV forecast from the electric load forecast. Achieving such a forecast requires a time series of PV production, electricity and netload, complemented with certain explanatory variables, such as weather. However, the stochastic nature and high uncertainty associated with these variables, makes it difficult for accurate prediction, especially for PV production [6]. In addition, calendar variables, such as the day type (workday or weekend), hour of the day, and season of the year and temperature, consumption behavior by the consumers, etc., have strong influence on electricity load forecast.

Several forecasting techniques, such as in [7-9], had been deployed in electric load forecasting, covering processes of problem formulation, data transformation to supervised learning, and feature learning. Short-term forecasting strategy to learn residents' consumption behavior has been introduced in [10, 11], while [12, 13] predicted a building heat load using

The work leading to this paper is part of the project GENTE funded by the framework of the joint programming initiative ERA-Net Smart Energy Systems with support from the European Union's Horizon 2020 research and innovation programme under grant agreement No 883973.

an integrated machine learning to address the flexibility between the use of district heating systems and heat pumps.

2) Netload forecasting

Netload forecast provides the knowledge of the difference between energy demand and PV production and helps ensure that sufficient energy is made available even when PV production is not available. The motivation for netload forecast is the sparsity of existing research in this area, particularly at residential and low-aggregate levels. This necessitates a more in-depth investigation to be carried out. Spatial relationships between various residential building loads should be considered and the forecast targets low-aggregate level netload.

Aggregated strategy directly forecasts the netload, i.e., the difference between the electricity demand and the electricity supply from renewable energy sources [1]. Similar model configurations to individual forecast are used here to justify their comparison. The netload is represented by (1).

$$P_{Net} = \mathrm{D} - \mathrm{S}, \quad (1)$$

where P_{Net} is the netload, D is the electrical load at user or low aggregate levels, and S is the electricity supply from renewable sources. The aggregated value of all the individual renewable energy generators in the system makes up the expected renewable energy supply. The forecast is performed on the time series after subtracting the supply from the demand. The assumed advantage of the strategy is that more aggregation leads to a time series that is simpler to forecast as it fluctuates less with fewer extreme values and more explicit recurring patterns [1].

One disadvantage of this aggregated strategy is that the relationship between netload and exogenous variables, such as the weather, might not be straightforward as the weather primarily influences the output of the generators and not the netload itself. Hence, there is a need to closely investigate the weather impacts on netload. It is important to note that almost none of the reported netload forecasts in literature [1-4], [7-11] made a forecast using real-time data. The models were only tested on historical data. Moreover, the methods in literature [2-4], [14-15] focused more on model's efficiency in terms of accuracy without considering how deployable they can be in the real-world. High computational complexity of forecast models, especially neural networks models result in high resource utilization making their deployment difficult. This is a challenge which needs to be addressed.

C. Main contributions and organization of the paper

To address the above presented challenges in load forecasting, this paper proposes an hourly netload forecast using neural networks algorithms. The choice of neural networks is due to its robustness to noise and its ability to capture complex and non-linear relationships in a time-series data. The chosen forecast horizon is best suited for real time control, ramp rate control, variability management as well as demand response scheduling in a building. Factors affecting the forecast results are investigated using the proposed model, the models' scalability and adaptability to new data, as well as problem of high resource utilization (due to complexity) associated with neural networks algorithms are also addressed. The main contributions of this paper can be summarized as follows:

- Development of an hour ahead load forecast models for both netload forecast as well as individual forecast, with light weight, low error, and high computational speed using scalable and adaptive deep learning (DL) algorithms. The models are thus suitable for on-device implementation in applications such as real-time energy monitoring and management.
- Validation of the models using hourly data from a real residential building, HSB Living Lab and comparisons of forecasting approaches, netload forecast versus individual component forecast (or disaggregated forecast).
- Analysis of the effects of weather and other external factors on the forecast results.

This paper delves properly into the conversation on the best option between individual and netload forecast by testing both. Two components such as PV and electric load were used in the proposed individual forecasting strategy.

This paper is organized as follows: Section I gives the motivation for the study, general overview of netload and individual load forecasting, the existing research efforts and the main contributions of this paper. The proposed forecast methodology was discussed in Section II. Section III presents the case studies and results. The conclusions and suggestions for future work are presented in Section IV.

II. PROPOSED FORECASTING METHOD

In this study, efforts are made to analyze the PV and electric load characteristics of an integrated energy system in relation to the netload. Netload can be forecasted using (2) based on the energy balance within an integrated energy system having both PV, electrical load, and battery storage.

$$P_{Net}(t) = \mathrm{P}_{Elect}(t) - (\mathrm{P}_{PV}(t) - \mathrm{P}_{Storage}(t)) \quad (2)$$

where $P_{storage}(t) = \mathrm{P}_{discharge}(t) - \mathrm{P}_{charge}(t)$

Firstly, a regression-based approach was used to predict PV output considering weather conditions especially solar irradiance as the most influential factor. The electric load on the other hand is predicted considering the influence of factors such as day type (workday or weekend), hour of the day, and ambient air temperature on the actual electricity load.

$$PV_{pred.} = f(C, H, T_a(t), S(t); \theta) \quad (3)$$

where $PV_{pred.}$ denotes the predicted PV, f is a neural network function parameterized by θ (weights and biases of the neural network), C is the cloud-cover, H relative humidity, $T_a(t)$ is the ambient air temperature at time *t*, and $S(t)$ is the solar radiation at time *t*.

$$P_{Elect\_pred.} = f(D, h, T_a(t); \theta) \quad (4)$$

where $P_{Elect\_pred.}$ is the predicted electric load, D is the day type (D = 1 for workdays and D = 0 for weekends), *h* is the hour of the day, $T_a(t)$ is the ambient air temperature at time *t*. Applying a multi-input single-output (MISO) neural networks on these influential weather/calendar variables firstly transforms their features linearly before passing them to the activation function. Therefore, the predicted electric load (4) is expanded further, and could now be expressed as:

$$P_{Elect\_pred.} = \sigma(W_1.[D, h, T_a(t), S(t)] + b_1).W_2 + b_2 \quad (5)$$

where W is the weight matrix, b is the bias vector and σ is the activation function. Netload is predicted by combining the influential factors for electric load and PV, expressed as:

$$P_{Net\_pred.} = f(D, h, H, C, S(t), T_a(t); \theta) \quad (6)$$

A. System framework for the forecasting model

The forecast models' framework here comprises three stages as shown in Fig. 2. Firstly, a decomposition algorithm was developed to separately collect electric load from the integrated energy system, as well as the historical weather data through application programming interface (API). Secondly, the disaggregated components and netload were trained individually using a lightweight model (achieved with an improved squeeze layer technique of [16]). At this stage, a baseline model is obtained. And the training error for the models were found to be less than two percent, which is satisfying. Finally, the base learners were optimized and made to make predictions using hourly data collected through API, from HSBLL and Numerical Weather Predictions (NWP) sources. A complete 1year data split into training and testing was used in training the deep learning algorithms, while 7 days was used for final prediction. The scalability of the models is defined in terms of their performance to the increase in dataset size, time and space complexity and deployment environment.

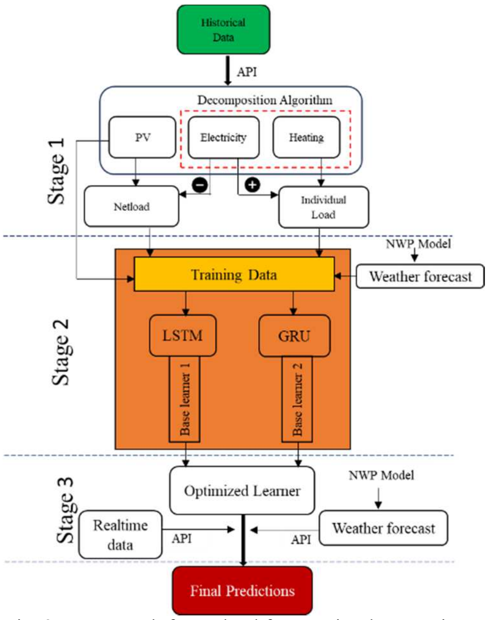

Fig. 2. Framework for netload forecast implementation

B. Forecast Horizon

A short-term forecast was leveraged. An hourly forecast for 7 days was tested.

Hourly forecast: This prediction is based on a two-step modelling chain, where the first step is the prediction of meteorological variables using Numerical Weather Predictions (NWP) [17]. Two deep learning algorithms were used: LSTM [18], GRU [19]. The algorithms were chosen because they accept multi-input-variables and can give single-output prediction that catches the nonlinearity in input variables. This multi-step hourly prediction for 24hrs was made from time step t+1 to t+24, at t = 1hr. For n-step forecast ahead, the vectors of A, B, C, D, and K in (7) and (8) can be described in a state-space form [20].

$$x_{t+1} = Ax_t + Bu_t + Ke_t \tag{7}$$

$$y_t = Cx_t + Du_t \tag{8}$$

where, u_t is input vector, y_t is the output vector, and x_t is the state vector.

Unfortunately, the historical data of the previous hours' actual values are not always available for all timestamp of the 24-hour horizon. Therefore, the "Walk-forward" optimization approach [21] that leverages the last predicted hour for the next timestamp prediction for 24-hour period was used. This approach is crucial because it enables the model performance to be assessed by recursively augmenting the training data with recent observations and re-evaluating the model over extended horizon [14].

C. Model Configuration

Two deep learning algorithms, LSTM [18] and GRU [19], with different model configurations and parameters were used in learning the characteristics of the training data. LSTM model has 250 neurons (100 at first layer, and 50 neurons at each hidden layer) and 1 in the output layer, with filter output size of 64, kernel size of (1, 3), dropout of 50% of the layers. GRU consists of a similar network but with only 200 neurons, filter output size of 200, dropout of 20%, and a stochastic optimization gradient (Adam). An improved version of [16] that compressed the networks' hidden layers was used in reducing the models' size i.e., making it lightweight.

D. Test Scenarios

Since the major objective of the study is to determine the most effective forecast method, a test using five different forecast models shown in Table I (presented in Section III), was carried out. *Disaggreg.1* and *aggreg.1-4* are test cases. The test assessed the effect of forecast data type (historical and real-time) and time, influence of weather and calendar variables, and other external factors to prediction outcomes. The test started by using historical data for prediction without recurse to the stage 3 process (involving optimized learner, and hourly data for PV, electric load and weather) as depicted in Fig. 2 of the framework. The performance of these base learners was assessed with an increase in the test sets' time scale. As soon as performance of the base learners is ascertained, the test is completed by optimizing the algorithms with hourly data, and weather and/or calendar data as the case of PV production and electric load.

The relationships between aggregated/disaggregated netloads and weather/calendar variables were analyzed using feature importance test. Pearson correlation analysis showed that weather and calendar variables in disaggregated netload have higher correlation coefficients in relation to their respective targets (i.e., actual electric load and actual PV production) than that of the aggregated netload. The final prediction of the aggregated netload across the tested models incidentally was affected by this correlation drop when compared against disaggregated netload. Apart from correlation drop, *aggreg.4* that used only historical data as shown in Table I, has the worst performance. This goes to show that other models understand and predict variations in netload better with comprehensive input data comprising of weather and calendar variables.

III. Case Studies: Results and Discussions

The disaggregated forecast versus netload forecast approaches are demonstrated using a case study using data from a real residential building in Sweden, a HSB Living Lab.

A. Description of Case Studies

The dataset for this paper is acquired through API from HSBLL's energy system, while the meteorological data is acquired from Numerical Weather Predictions (NWP) [17]. The data included hourly electricity load, PV output and local weather data containing seven variables collected from 00:00:00 hours of 15th of October 2021 to 23:00:00 hours of 13th of October 2022. HSBLL is a smart residential building consisting of 29 apartments. The building contains 2 electric vehicle chargers with two 3-phase outlets, a heating system that is composed of 2 heat pumps and 3 hot water storage tanks. It also contains a washing machine, dish washer, tumbler dryer and other non-controllable loads. The actual PV and electric load data are used as the targets of the individual prediction models. The measured netload data is used as the target for the direct netload forecast. All model inputs and targets are standardized to be within the range (0, 1). The forecast horizon is 1-hour ahead and previous 24-hour observations were used as inputs to the models.

Applying (1) on HSBLL dataset for July 2023, the netload for building is obtained. Exploratory analysis to ascertain the distribution of the loads as depicted in Fig. 3 showed that variation between individual load and netload decreased significantly around the hours 48 – 72, 504 – 552, and 576 – 600 corresponding to day 2 – 3, 21 – 23, 24 – 25, respectively.

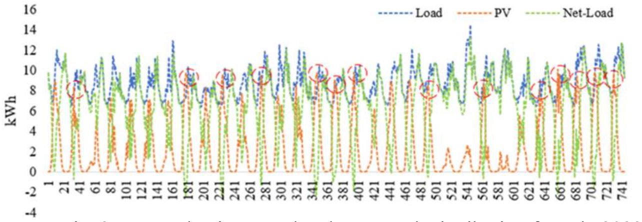

Fig. 3. PV production, Load and Net-Load Distribution for July 2023

B. *Forecast Evaluation*

Training and testing of the models' performance was done using the entire one-year historical data, but predictions analyzed here are for 7 consecutive days, i.e., 1-week predictions between April 11th to 18th 2024.

Two major standard error metrics: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were used for model evaluation. These metrics are the most suitable and commonly used metrics for neural networks models evaluation. Both error metrics use the same scale as the measured data, which implies that the error is of the same unit with the predictions and can range from 0 to ∞.

1. RMSE measures the average magnitude of the error. It is the square root of the average of squared differences between prediction and actual observation.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{9}$$

2. MAE measures the average magnitude of the absolute errors in a set of predictions, without considering their direction. It is mostly used for regression tasks, especially for accuracy evaluation.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|x_i - y_i| \tag{10}$$

where x_i is the observed data, y_i denotes the predicted data, and n is the number of observations.

Table I shows the results analysis of the models tested, while Table II and III show the results regarding the lightweight and scalability, respectively, demonstrated on both aggregated and disaggregated forecast approaches using two algorithms. Table III shows no significant difference in the models' accuracy as the training data size is increased. Based on Table I, the disaggregated approach using LSTM achieved lower error because it leveraged finer granularity and detailed modeling of individual components of netload. However, GRU outperformed LSTM in aggregated forecasts tested. The differences in RMSE and MAE across the input variables of Table I indicate the importance of calendar variables in netload forecasting. As shown, *Aggreg.1* model was good in netload because it captured the effects of historical patterns, weather variations, and calendar-related behavioral patterns. However, it is important to state that the results shown in Table I are highly dependent on the accuracy of the weather variables used in the forecast model training.

TABLE I. PERFORMANCE ANALYSIS BASED ON THE TEST CASE

Metrics	***Disaggreg.1*** *Input: weather, historical PV, historical Electric load, Calendar*	***Aggreg.1*** *Input: weather, historical Netload, Calendar*	***Aggreg.2*** *Input: weather, historical Netload*	***Aggreg.3*** *Input: historical Netload, Calendar*	***Aggreg.4*** *Input: historical Netload*
RMSE (kW) LSTM	**1.2057**	1.8008	3.0928	2.5718	3.9898
RMSE (kW) GRU	**1.2932**	1.7312	2.97705	2.3984	3.7107
MAE ((kW) LSTM	**0.8438**	1.2976	2.4078	1.9751	3.1584
MAE ((kW) GRU	**0.9251**	1.2663	2.4155	1.8522	2.8856

TABLE II. MODELS' LIGHTWEIGHT DEMONSTRATION

		Aggregated	***Disaggregated***	
			PV	***Elect.***
Computational time	Training	16.87s	0.75s	14.89s
	Forecast	0.81s	0.1s	0.90s
Model size	LSTM	2.9332 MB		
	GRU	0.5269 MB		

TABLE III. SCALABILITY DEMONSTRATION ON INDIVIDUAL LOADS

Scalability Metrics	***LSTM***		***GRU***	
	6-month data	***1-year data***	***6-month data***	***1-year data***
Accuracy (%)	90.70	92.00	90.80	91.00
Time of train (s)	11.60	16.87	4.66	10.16

Lack of access to datasets made it difficult to compare this proposed forecast approach with similar methods in literature, most especially [15] which reported a RSME of 7.967 kW. However, on comparing with statistical methods tested on the same dataset, an improvement of 12% over statistical methods was achieved. Apart from the performance improvement, it also demonstrated scalability, fast to train with compressed model size without a considerable loss in accuracy. Moreover, the models with weather variables proved not susceptible to weather variation when used in predicting a specific hour without including weather variables for that hour. This implies that the model is adaptive, hence, can be tested on any dataset. However, inclusion of only calendar variables significantly affected the outcome of netload prediction. It captured the temporal variations and behavioral patterns in electricity consumption, ultimately leading to more accurate forecasts.

Fig. 4 shows the 7-day ahead forecasts between the aggregated and disaggregated netload against the actual netload. Disaggregated forecast performs better than aggregated when compared with the actual netload and showed it can predict the peakload. Disaggregated forecast has a variation of 11.5% against 27.11% recorded in aggregated.

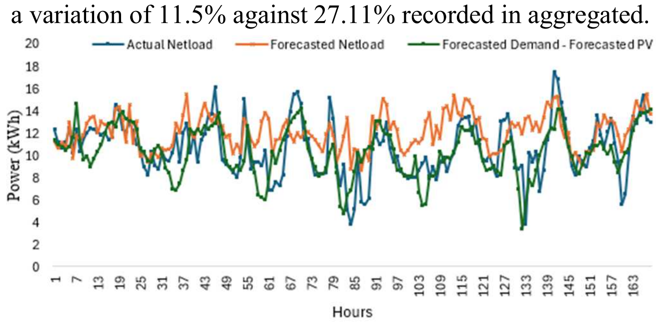

Fig. 4. Comparison of forecasts against actual netload

Scalability of the model was tested using the models accuracy and time-of-train as metrics over different training data size and computing infrastructure (on-premise, cloud).

Figs 5 and 6 show the forecasting error distribution versus normal distribution for disaggregated and aggregated netload forecasts, respectively. As can be seen, the error distribution in Fig. 5 is relatively more concentrated around the mean (0), compared with Fig. 6. This means lower variance. Secondly, the peak of the forecasting error distribution in Fig. 5 aligns well with the peak of the normal distribution, though it has a higher density. Looking at the error alignment of Fig. 5 with the normal distribution curves, it has a significant overlap. The error distribution of Fig. 6 is wider and less peaked compared to Fig. 5. This wider spread indicates a larger variance, and the lower peak shows a flatter distribution. The error of Fig. 6 does not align as closely with the normal distribution curve compared to Fig. 5. The increase in deviations, particularly on the left side, indicates skewness in the error distribution.

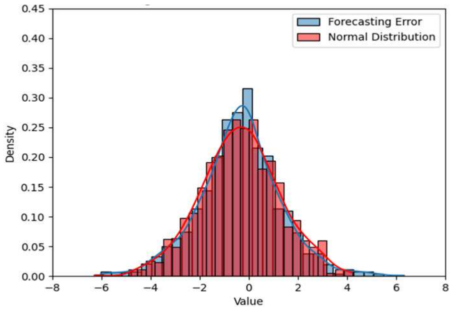

Fig. 5. Error distribution for disaggregated forecast

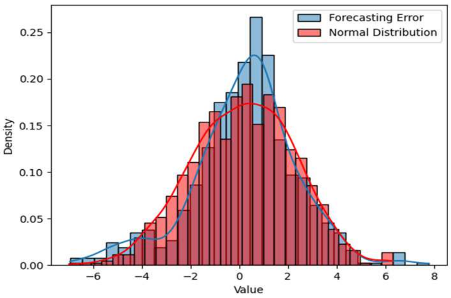

Fig. 6. Error distribution for aggregated forecast

C. *Factors affecting the forecast results*

Further investigation to understand the reason for the underperformance of aggregated netload showed that weather and calendar variables influence is a major affecting factor. The decrease of these variables in the correlation test is a strong proof. The highest decrease was recorded in solar irradiance, hour of the day, temperature, and relative humidity. A further probe showed that these variables are strongly correlated with PV production. Also, the export power that occurs when PV production is more than the electricity demand (circled), as can be seen in Fig. 3, is another factor.

IV. CONCLUSIONS

This paper developed load forecast models using deep learning algorithms to determine the most effective approaches between disaggregated and aggregated netload forecast. The case studies results showed that disaggregated forecast is better than aggregated (netload) forecast. The performance of disaggregated forecast was found to be linked to weather and calendar variables effects, and the power export that occurs when PV production is higher than the electricity demand. The developed forecast models demonstrated scalability, and their lightweights make them deployable on resource-constraint devices. They also proved that they are adaptive to new data and can handle noise. The models can be adopted and extended to various types of buildings and could be used by building owners by integrating them in building energy management systems, as well as by the grid operators to predict the demands by grid users to better plan for grid operation and congestion management.

V. REFERENCES

[1] Beichter, M., et al., "Net load forecasting using different aggregation levels". *Energy Informatics*, 2022. 5(1): p. 19.

[2] Wang, Y., et al. "Load Forecasting Method of Integrated Energy System Based on CNN-BiLSTM with Attention Mechanism" in 2021 3rd International Conference on Smart Power & Internet Energy Systems (SPIES). 2021.

[3] Kong, W., et al., "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network," *IEEE Transactions on Smart Grid*, 2019. 10(1): p. 841-851.

[4] Wang, Y., et al., "Data-Driven Probabilistic Net Load Forecasting With High Penetration of Behind-the-Meter PV," *IEEE Transactions on Power Systems*, 2018. 33(3): p. 3255-3264.

[5] Razavi, S.E., et al., "From Load to Net Energy Forecasting: Short-Term Residential Forecasting for the Blend of Load and PV Behind the Meter," *IEEE Access*, 2020. 8: p. 224343-224353.

[6] Das, U.K., et al., "Forecasting of photovoltaic power generation and model optimization: A review," *Renewable and Sustainable Energy Reviews*, 2018. 81: p. 912-928.

[7] Liang, F., et al., "Deep Learning-Based Power Usage Forecast Modeling and Evaluation," *Procedia Computer Science*, 2019. 154: p. 102-108.

[8] Chandramitasari, W., B. Kurniawan, and S. Fujimura, "Building Deep Neural Network Model for Short Term Electricity Consumption Forecasting," in 2018 International Symposium on Advanced Intelligent Informatics (SAIN). 2018.

[9] Berriel, R.F., et al.," Monthly energy consumption forecast: A deep learning approach," in 2017 International Joint Conference on Neural Networks (IJCNN). 2017.

[10] Wen, L., K. Zhou, and S. Yang, "Load demand forecasting of residential buildings using a deep learning model," *Electric Power Systems Research*, 2020. 179: p. 106073.

[11] Kong, W., et al., "Short-Term Residential Load Forecasting Based on Resident Behaviour Learning," *IEEE Transactions on Power Systems*, 2018. 33(1): p. 1087-1088.

[12] Eseye, A.T. and M. Lehtonen, "Short-Term Forecasting of Heat Demand of Buildings for Efficient and Optimal Energy Management Based on Integrated Machine Learning Models," *IEEE Transactions on Industrial Informatics*, 2020. 16(12): p. 7743-7755.

[13] Saloux, E. and J.A. Candanedo, "Forecasting District Heating Demand using Machine Learning Algorithms," *Energy Procedia*, 2018. 149: p. 59-68.

[14] Suradhaniwar, S., et al., "Time Series Forecasting of Univariate Agrometeorological Data: A Comparative Performance Evaluation via One-Step and Multi-Step Ahead Forecasting Strategies," Sensors, 2021. 21(7): p. 2430.

[15] Zazoum, B., "Solar photovoltaic power prediction using different machine learning methods," Energy Reports, 2022. 8: p. 19-25.

[16] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," Proceedings of the IEEE conference on computer vision and pattern recognition 2018, Pages: 7132-7141.

[17] Rebase Energy meterological forecast platform; accessed on 10th April, 2023, available from: https://www.rebase.energy/.

[18] Mokarram, M.J., et al., "Net-load forecasting of renewable energy systems using multi-input LSTM fuzzy and discrete wavelet transform," *Energy*, 2023. 275: p. 127425.

[19] Zheng, J., et al. "Short-term power load forecasting of residential community based on GRU neural network," in 2018 IEEE International conference on power system technology (POWERCON). 2018.

[20] Durbin, J. and S.J. Koopman, "Time series analysis by state space methods", Oxford University Press United Kingdom, Vol. 38. 2012.

[21] Falessi, D., et al., "On the need of preserving order of data when validating within-project defect classifiers," Empirical Software Engineering, 2020. 25: p. 4805-4830.