A Comparison of Various Deep Learning Methods for Household Load Forecasting

Deivamani, Karthikeyan; Norouzi, Farshid; Shekhar, Aditya; Bauer, Pavol

DOI
10.1109/ISGTEUROPE56780.2023.10407916

Publication date
2023

Document Version
Final published version

Published in
Proceedings of 2023 IEEE PES Innovative Smart Grid Technologies Europe, ISGT EUROPE 2023

Citation (APA)
Deivamani, K., Norouzi, F., Shekhar, A., & Bauer, P. (2023). A Comparison of Various Deep Learning Methods for Household Load Forecasting. In *Proceedings of 2023 IEEE PES Innovative Smart Grid Technologies Europe, ISGT EUROPE 2023* (IEEE PES Innovative Smart Grid Technologies Conference Europe). IEEE. https://doi.org/10.1109/ISGTEUROPE56780.2023.10407916

A Comparison of Various Deep Learning Methods for Household Load Forecasting

Karthikeyan Deivamani
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Delft, The Netherlands
K.Deivamani@student.tudelft.nl

Farshid Norouzi
Department of DC Systems Energy Conversion and Storage
Delft University of Technology
Delft, The Netherlands
f.norouzi@tudelft.nl

Aditya Shekhar
Department of DC Systems, Energy Conversion and Storage
Delft University of Technology
Delft, The Netherlands
a.shekhar@tudelft.nl

Pavol Bauer
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Delft, The Netherlands
P.Bauer@tudelft.nl

2023 IEEE PES Innovative Smart Grid Technologies Europe (ISGT EUROPE) | 979-8-3503-9678-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISGTEUROPE56780.2023.10407916

***Abstract*—Forecasting energy consumption is vital for smart grid operations to manage demand, plan loads, and optimize grid operations. This work aims at reviewing and experimentally evaluating six univariate deep learning architectures to forecast load for a single household using a real-world dataset. Multi-layer perceptron (MLP), Convolutional neural network (CNN) and recurrent neural networks (Simple RNN, Long Short Term Memory (LSTM)) were the neural network methods that were analysed along with robust LSTM architectures like Bi-directional LSTM and CNN-LSTM Hybrid. All the models were tuned using Bayesian optimization and evaluated using root mean squared error (RMSE) as the metric. In addition to neural network models, Seasonal ARIMA (SARIMA) a statistical model is also presented to observe the performance. As a result, Bi-directional LSTM was observed to have achieved the best performance with the smallest value of RMSE; however, it was also observed that differences in performances between other neural network models were quite low, especially between the RNN architectures. Additionally, although machine learning methods performed better than SARIMA the former model was more complex and computationally intensive.**

***Index Terms*—electric load forecasting, smart grid, time-series forecasting, univariate, deep learning**

I. Introduction

Microgrids are a promising solution in making the electric grid more reliable and green by improving energy reliability, energy sharing and demand-side management aspects. To leverage the full capabilities of a microgrid, accurate load forecasting becomes a critical task either from a consumer perspective to reduce consumption or from a grid operator perspective for a better decision-making process or for efficient energy storage system management. With the rise in advanced monitoring infrastructure more granular and extensive data is being collected. Deep learning forecasting methods have demonstrated significant potential in effectively managing larger and more intricate datasets [1].

979-8-3503-9678-2/23/$31.00 ©2023 IEEE

Short-term load forecasting (STFL) is the process of predicting the power demand of a power system over a short-term period, typically ranging from a few minutes to a few hours. Deep learning has demonstrated improved performance in modeling complex patterns for individual household load profiles, which tend to be more volatile due to their dependence on individual behavior, as opposed to aggregate level modeling [2].

In literature, different types of models both linear and non-linear have been used for STFL. Family of Auto-regressive moving average (ARMA) models were pioneers in STFL [3] which was then evolved into SARIMA to account for seasonal variance [4]. The limitation of this set of statistical methods is that it assumes a linear system whereas most often real-world cases exhibit non-linear properties. In order to solve this shortcoming, models like feed-forward neural networks have become attractive as they show capabilities in modelling complex non-linear systems such as load forecasting [1]. Neural network techniques range from simple MLP to convolutional methods to recurrent neural networks [5] along with their variants LSTM [6] and Gated recurrent units (GRU). Hybrid architectures have also been proposed in the literature between neural networks as well as between statistical and machine learning methods such as CNN-LSTM hybrid presented in [7] and a hybrid LSTM-Exponential smoothening [8] respectively.

The scope of this paper is to provide a comparative analysis of basic deep-learning architectures for STFL of a single household and compare using standard error metrics such as RMSE and MAE to input into an energy management system.

This paper is organized as follows: Section II reviews the load dataset, preprocessing steps, error metrics and the details of the models being evaluated. Next, section III presents the results and discussion of parameter search and load forecasting models. Finally, the conclusions are provided in section IV.

II. Methodology

A. Data Collection and preparation

The dataset used in this analysis is the hourly load from a single household which was retrieved from the IHomeLab RAPT dataset [9]. The data is from a household in the greater Lucerne region in Switzerland and the data spans from 1 December 2016 to 31 July 2019 which is around two and a half years.

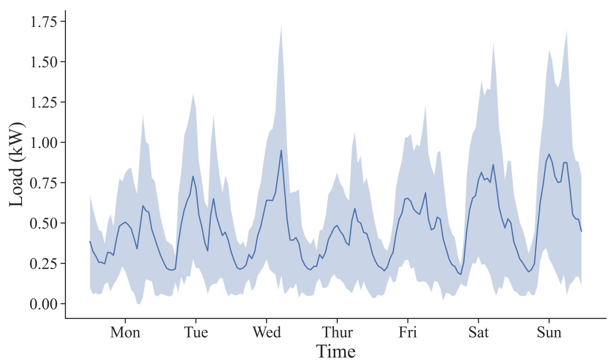

Fig. 1. Weekly statistics for the load in the whole IHomeLab dataset. The bold line is the mean and the blue area covers one standard deviation from the mean

In Fig. 1 the bold line represents the mean of that hour over every week in the dataset, here it could be observed that there are high fluctuations and a higher mean during Sunday,Saturday and Wednesday. However, higher consumption during the weekend is expected but Wednesday is an interesting observation.

The missing values in the dataset were located and filled using the previous value and to deal with non-stationarity the data was differenced from the previous value [10]. Stationarity of the data was checked using Augmented Dickey-Fuller (ADF) test [11].

B. Forecasting models

1) Baseline:
A seasonal naive model was used which is using the same value of the previous season as the predicted value this is also called a persistent model. In this analysis, since the daily seasonality is strong as inferred from auto-correlation values the previous day were used as the forecasting value for the coming day.

2) Statistical Model (SARIMA):
SARIMA is a class of time series forecasting techniques that predicts the future values based only on the past behavior of the variable being modelled along with accounting for seasonalities [4]. The mathematical representation of the model is given below.

$$y_t = c+\sum_{n=1}^{p} \phi_n y_{t-n}+\sum_{n=1}^{q} \theta_n \epsilon_{t-n}+\sum_{n=1}^{P} \Phi_n y_{t-sn}+\sum_{n=1}^{Q} \Theta_n \epsilon_{t-sn}+\epsilon_t \tag{1}$$

SARIMA (p,d,q)(P, D, Q)s is a parametric model, and (p,d,q) is the auto-regressive lag order, order of differencing, moving average lag order respectively and (P, D, Q) is for the seasonal terms. In (1) the c is a constant term, ϵ_t is the error term and ϕ_n,θ_n,Θ_n,Φ_n are the coefficients of lag terms .

3) Artificial Neural Network (ANN) Methods:
Neural Networks are part of a family of machine-learning techniques inspired by the functioning of the human brain which consists of interconnected nodes (neurons) compartmentalized into layers [12]. A typical neural network architecture consists of an input, output layer along with hidden layers where the abstractsa are learned . For this forecasting problem, 24 lags were utilized as the feature due to its strong diural pattern [13]. Different architectures of neural networks built on the foundational structure are discussed below.

a) Multi layer perceptron (MLP):
Multi-layer perceptron is a type of feed-forward ANN where the information flows in only one direction from the input layer to the output layer. This brings the limitation of not capturing the temporal dependencies regardless MLP has shown to have competitive performance in a few load forecasting use cases [14].

b) Convolutional Neural Network(CNN):
Convolutional neural networks are a family of ANN which works with a grid-like structure and have been extensively used in image recognition and natural language processing [15]. The key property of CNNs to extract features has been leveraged for univariate time-series forecasting, where a filter/kernel is passed through the series to extract relevant features. Although CNN has shown to be effective in NLP and image recognition, they are not widely used in time series forecasting as CNN cannot model sequential data, which has been addressed recently by combining them with recurrent neural networks [2]. A vanilla CNN and a CNN-LSTM hybrid architectures have been analyzed in this work.

c) Recurrent Neural Network (RNN):
Recurrent neural networks is a NN architecture that modifies feed-forward neural networks to handle sequential data and capture patterns better, which makes it a powerful tool for time series forecasting [5]. Compared to feed-forward neural networks, RNNs maintain an internal state that allows them to remember information from previous inputs.RNN's internal feedback loop allows it to use its previous outputs as inputs to the current step, making it capable of modeling sequences and capturing temporal dependencies in the data.

d) Long Short Term Memory (LSTM):
LSTM is a variant of RNN that was developed to address the issue of vanishing or exploding gradients during backpropagation. This problem occurs when the gradient values become too small or too large, making it difficult for the network to update the weights effectively [16].LSTMs have a hidden state and a cell state which store short-term dependencies and long-term information. This ability of LSTM to store long-term dependencies has it a popular choice in load forecasting [6]. One extension of LSTM is the Bi-directional LSTM, as unidirectional LSTM processes the input sequence only in the

TABLE I
HYPERPARAMETERS AND PARAMETER RANGES FOR DEEP LEARNING MODELS

Hyperparameter	Range
Number of hidden layers	[1,2,3]
Number of neurons	(1,120)
Dropout ratio	(0,0.5)
Learning rate	(0.00001,0.01)

forward direction, and Bi-LSTM processes both ways forward and backward [17]. This allows the model to capture a better sequential relationship in the input sequence. Unidirectional and Bi-directional LSTMs both have been explored in this analysis.

C. Hyperparameter Tuning

Hyperparameter tuning is important in time series forecasting to find the best model hyperparameters that minimize forecast error. RMSE is a commonly used target metric for this optimization process.

For SARIMA, the autoregressive, differencing, moving average terms $(p, d, q)x(P, D, Q)$ were analyzed using a correlogram, after which a grid search was done to obtain best-fit hyperparameters [4].

Two common methods for hyperparameter tuning are grid search and random search. However, as the number of hyperparameters to tune increases, the number of possible combinations can grow exponentially, making the search computationally intensive and sometimes infeasible. Bayesian optimization is a probabilistic model which finds an optimal set of hyperparameters in fewer evaluations. After every iteration, it updates the search algorithm and avoids the low-performing region [18]. For this comparative study, a simple Bayesian algorithm was performed for the hyperparameter in the range as mentioned in table II.

D. Performance Metrics

The accuracy of models and neural network architectures is typically assessed using scale-dependent error metrics like root mean squared error (RMSE) and mean average error (MAE) [19]. RMSE is particularly effective as it places greater weight on large errors and can handle values close to zero. Since they are scale-dependent, they cannot be used to compare different datasets. R-squared (R^2) is a statistical measure that represents the proportion of the variance in predicted values is explained by the true values, (4) shows the mathematical representation.

1) Mean Absolute Error (MAE):

$$MAE = \frac{1}{n}\sum |\mathbf{\hat{y}_t} - \mathbf{y_t}| \tag{2}$$

2) Root Mean Squared Error (RMSE):

$$RMSE = \sqrt{\frac{1}{n}\sum |\mathbf{\hat{y}_t} - \mathbf{y_t}|^{\mathbf{2}}} \tag{3}$$

3) R-Squared:

$$R^2 = 1 - \frac{\sum(\mathbf{y_i} - \mathbf{\hat{y}})^2}{\sum(\mathbf{y_i} - \mathbf{\bar{y}})^2} \tag{4}$$

In (2) and (3), $\mathbf{\hat{y}_t}$ is the predicted value and $\mathbf{y}$ is the actual value. To assess model accuracy, a combination of metrics is needed to inspect different aspects of the forecast. In this paper, RMSE is chosen as the metric over which models are optimized as well as for evaluation.

A holdout validation was undertaken for evaluation where the dataset was split into 70% for training 20% for validation and 10% for testing without shuffling of data points.

III. RESULTS AND DISCUSSION

In this section, the results of the above mentioned models using the IHomeLab dataset and accuracy on the test data using RMSE and MAE are analyzed.

TABLE II
SUMMARY OF PREDICTION ERRORS (RMSE AND MAE) FOR ABOVE MODELS

Models	MAE (kW)	RMSE (kW)	$\mathbf{R^2}$
Persistence Model (Baseline)	0.251	0.410	-0.356
SARIMA	0.231	0.322	0.149
Deep Learning Models			
Multilayer Perceptron (MLP)	0.195	0.281	0.359
Convolutional Neural Network(CNN)	0.179	0.275	0.363
Recurrent Neural Network (RNN)	0.183	0.277	0.361
Long Short Term Memory (LSTM)	0.175	0.270	0.374
CNN-LSTM Hybrid	0.181	0.271	0.404
Bidirectional LSTM	0.173	0.269	0.409

A. SARIMA

For the SARIMA model, after analyzing autocorrelation and partial autocorrelation plot followed by a grid search the optimum parameters were found to be $p = 2, d = 1, q = 1, P = 3, D = 0, Q = 2$ with the seasonality of 24 hours. As seen in the table II RMSE and MAE were 0.231 kW and 0.322 kW, on comparing this with the baseline performance SARIMA had lower RMSE and MAE. In Fig. 2, the predicted values capture the daily variation but fail to capture the increase during the weekend as the model is limited to one seasonality. As it assumes a linear relationship and stationarity in data which states constant variance so the model fails to capture the non-linearity. A key advantage of SARIMA and statistical models, in general, is that they need less data to train compared to machine learning methods which makes them robust when fewer data is available. This model is introduced to compare a well-popular statistical model with deep learning methods.

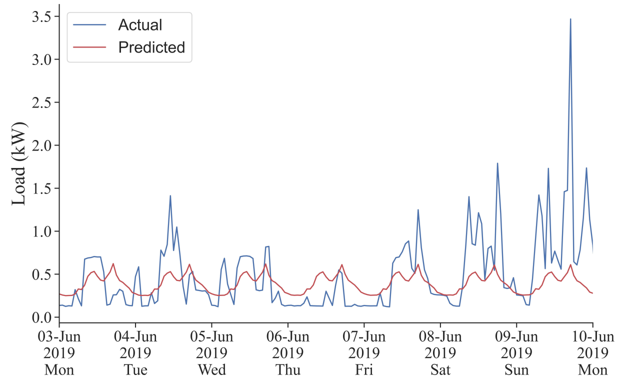

Fig. 2. Forecasted (Red) and Actual (Blue) Load profile for a week using SARIMA Model

B. Multilayer Perceptron (MLP)

In Fig. 3, it can be seen that the MLP model is able to capture the trend of the actual data and fit better than the SARIMA model with a RMSE of 0.281 kW. It can also be noticed that the model does capture variance significantly better visually and by inferring R-squared value in Table II . The parameter chosen were 2 dense layers with the first layer of 90 neurons and the second layer of 20 neurons, a learning rate of 0.0001, dropout ratio of 0.2 for 100 epochs.

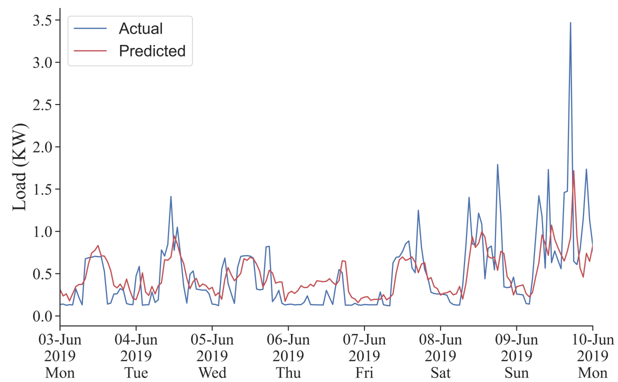

Fig. 3. Forecasted (Red) and Actual (Blue) Load profile for a week using Multilayer Perceptron (MLP)

One of the main drawbacks of using MLP for load forecasting is that it does not model the long-term dependencies and the temporal relationships between the data. Compared to other architectures mentioned, MLPs are heavily parameterized due to the fully connected nature of dense layers which lead to overfitting of the training data. Regularization by a dropout layer was introduced to tackle this problem and generalize the model.

C. Convolutional Neural Network (CNN)

Optimization of CNN hyperparameters yielded 16 filters, kernel size of 4 and 30 neurons in the dense layer as the optimal hyperparameters from bayesian optimization. It performed better than MLP with a RMSE of 0.275 kW with a slight reduction in RMSE and better capturing of vairance as seen in Fig.4 and Table II . The architecuture of CNN is to capture local patterns through filters explains the variance in the predicted data and it also emphasizes the limitation of not capturing the trend and long-term dependencies in the data.

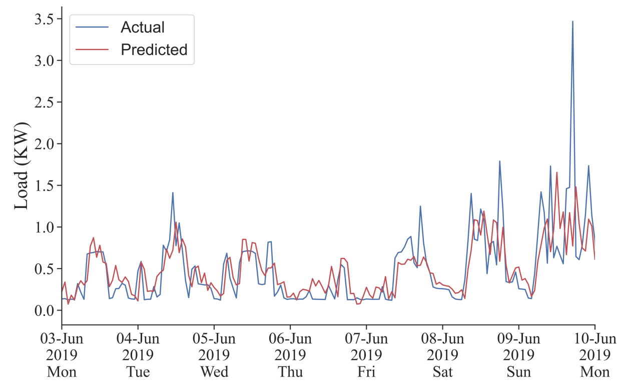

Fig. 4. Forecasted (Red) and Actual (Blue) Load profile for a week using Multilayer Perceptron (CNN)

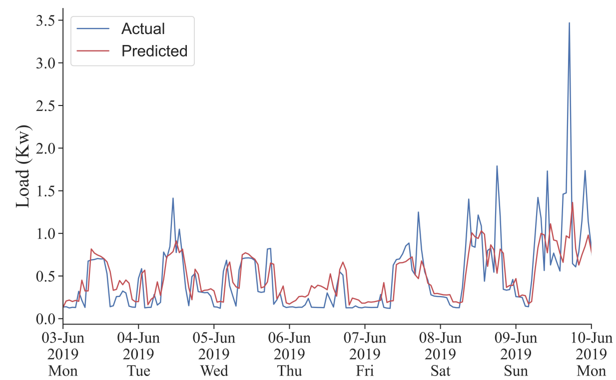

Fig. 5. Forecasted (Red) and Actual (Blue) Load profile for a week using Multilayer Perceptron (Bi-Directional LSTM)

D. Bi-Directional LSTM

In this analysis, Bidirectional LSTM has shown to be the model with the least RMSE of 0.269kW, but the difference between this and other recurrent neural networks (LSTM, CNN-LSTM) is very low as seen in Table II. In Fig. 5, it can be observed that some predictions clearly reflect the influence of the actual value of the previous time step. This is mainly due to the sequential nature of the data and high correlation with the previous time step even after differencing and removing non-stationarity. Adding external features that influence the load could improve the performance further by providing additional information to the model like for instance weather conditions, holidays, and special events can all impact the electricity consumption of the household. A shallow architecture with a Bi-directional LSTM with 64 neurons and a learning rate of 0.001 was chosen and trained for 80 epochs.

Although, LSTMs have shown good results they suffered from overfitting during experimentation due to smaller train data and a high number of parameters. This was addressed by introducing a regularization dropout layer with a dropout ratio of 0.2.

In this analysis, it was observed that RNNs outperformed the baseline and other neural network methods as seen by their lower RMSE. It was also noted that LSTM based models performed better than simple RNNs with Bi-directional LSTM having a slight edge over others in terms of RMSE, as shown in 6. The problem of overfitting was constantly experienced during the empirical testing for all the models as the data set used to train was limited to only 2 years, and neural networks are complex models that require large amounts of data to achieve better performance.

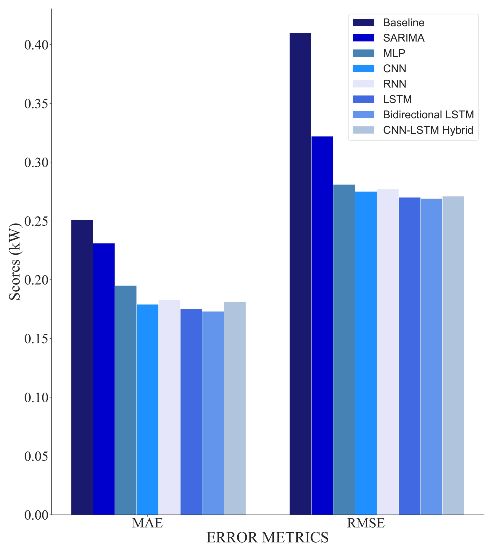

Fig. 6. Comparison of RMSE and MAE for all the discussed models

IV. Conclusion

In this paper, we have compared the univariate single-step load forecasting performance of a statistical model (SARIMA) and primary neural network architectures such as MLP, CNN, RNN and LSTM using rRMSE as the evaluation metric and Bi-Directional LSTM performance was superior to other models.

Although there were only slight variations in the performance of different neural network models, Bi-LSTM exhibited the best results in terms of RMSE, as depicted in Fig. 6 and Table II. Nonetheless, the differences in the performance among these models were not significant. It should also be noted that neural network methods on average performed 12% better than SARIMA but the implementation for the former was complex and computationally intensive than the later. Overall, the findings suggest that deep learning approaches are effective for energy consumption forecasting and can be useful in smart grid operations for managing demand, planning loads, and optimizing grid operations.

References

[1] A. Gasparin, S. Lukovic, and C. Alippi, "Deep learning for time series forecasting: The electric load case," *CAAI Transactions on Intelligence Technology*, vol. 7, no. 1, pp. 1–25, 2022.

[2] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 1087–1088, 2017.

[3] Y. Chakhchoukh, P. Panciatici, and L. Mili, "Electric load forecasting based on statistical robust methods," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 982–991, 2010.

[4] A. K. Dubey, A. Kumar, V. García-Díaz, A. K. Sharma, and K. Kanhaiya, "Study and analysis of sarima and lstm in forecasting time series data," *Sustainable Energy Technologies and Assessments*, vol. 47, p. 101474, 2021.

[5] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *International Journal of Forecasting*, vol. 37, no. 1, pp. 388–427, 2021.

[6] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on lstm recurrent neural network," *IEEE transactions on smart grid*, vol. 10, no. 1, pp. 841–851, 2017.

[7] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid cnn-lstm model for short-term individual household load forecasting," *Ieee Access*, vol. 8, pp. 180 544–180 557, 2020.

[8] G. Dudek, P. Pełka, and S. Smyl, "A hybrid residual dilated lstm and exponential smoothing model for midterm electric load forecasting," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 7, pp. 2879–2891, 2021.

[9] P. Huber, M. Ott, M. Friedli, A. Rumsch, and A. Paice, "Residential power traces for five houses: the ihomelab rapt dataset," *Data*, vol. 5, no. 1, p. 17, 2020.

[10] J. C. López, M. J. Rider, and Q. Wu, "Parsimonious short-term load forecasting for optimal operation planning of electrical distribution systems," *IEEE transactions on power systems*, vol. 34, no. 2, pp. 1427–1437, 2018.

[11] B. Nepal, M. Yamaha, A. Yokoe, and T. Yamaji, "Electricity load forecasting using clustering and arima model for energy management in buildings," *Japan Architectural Review*, vol. 3, no. 1, pp. 62–76, 2020.

[12] J. C. B. Gamboa, "Deep learning for time-series analysis," *arXiv preprint arXiv:1701.01887*, 2017.

[13] M. Elsaraiti and A. Merabet, "Solar power forecasting using deep learning techniques," *IEEE Access*, vol. 10, pp. 31 692–31 698, 2022.

[14] M. Massaoudi, S. S. Refaat, I. Chihi, M. Trabelsi, F. S. Oueslati, and H. Abu-Rub, "A novel stacked generalization ensemble-based hybrid lgbm-xgb-mlp model for short-term load forecasting," *Energy*, vol. 214, p. 118874, 2021.

[15] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 international conference on engineering and technology (ICET)*. Ieee, 2017, pp. 1–6.

[16] Z. Hu, J. Zhang, and Y. Ge, "Handling vanishing gradient problem using artificial derivative," *IEEE Access*, vol. 9, pp. 22 371–22 377, 2021.

[17] H. Jahangir, H. Tayarani, S. S. Gougheri, M. A. Golkar, A. Ahmadian, and A. Elkamel, "Deep learning-based forecasting approach in smart grids with microclustering and bidirectional lstm network," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 9, pp. 8298–8309, 2020.

[18] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[19] C. Bergmeir and J. M. Benítez, "On the use of cross-validation for time series predictor evaluation," *Information Sciences*, vol. 191, pp. 192–213, 2012.