Short-Term Load Forecasting Using A Particle-Swarm Optimized Multi-Head Attention-Augmented CNN-LSTM Network

Quansah, Paapa Kwesi (Author)
Department of Electrical/Electronics Engineering
Kwame Nkrumah University of Science and Technology
Kumasi, Ghana
Email: officialpaapa@gmail.com

***Abstract*—Short-term load forecasting is of paramount importance in the efficient operation and planning of power systems, given its inherent non-linear and dynamic nature. Recent strides in deep learning have shown promise in addressing this challenge. However, these methods often grapple with hyperparameter sensitivity, opaqueness in interpretability, and high computational overhead for real-time deployment. In this paper, I propose a novel solution that surmounts these obstacles. Our approach harnesses the power of the Particle-Swarm Optimization algorithm to autonomously explore and optimize hyperparameters, a Multi-Head Attention mechanism to discern the salient features crucial for accurate forecasting, and a streamlined framework for computational efficiency. Our method undergoes rigorous evaluation using a genuine electricity demand dataset. The results underscore its superiority in terms of accuracy, robustness, and computational efficiency. Notably, our Mean Absolute Percentage Error of 1.9376 marks a significant advancement over existing state-of-the-art approaches, heralding a new era in short-term load forecasting.**

***Keywords*-Short-Term Load Forecasting; Deep Learning; Particle-Swarm Optimization; Multi-Head Attention; CNN-LSTM Network; Electricity Demand; Computational Efficiency;**

I. Introduction

In our contemporary society, electrical energy has emerged as a pivotal resource propelling the economic and societal progress of nations worldwide. It finds widespread application across diverse sectors, including manufacturing, mining, construction, agriculture, textiles, and communication. The provision of consistent and high-quality electrical power is not merely a convenience; it is an imperative for sustaining investor confidence in every economy, particularly the Ghanaian market and fostering further development [1]. With the advent of new technological advancements, the demand for electricity has surged, creating an urgent need for more cost-effective and reliable power supply solutions [2].

Currently, our energy infrastructure lacks substantial energy storage capabilities in the generation, transmission, and distribution systems [3]. This deficiency necessitates a delicate equilibrium between electricity generation and consumption. Adapting electricity generation to dynamically meet shifting demand patterns is paramount; otherwise, the stability of the entire power system is at risk [4]. Maintaining this equilibrium hinges on the implementation of a precise load prediction scheme, the significance of which cannot be overstated.

Electric load forecasting, the practice of predicting electricity demand within a specific region, can be segmented into three distinct timeframes: short-term, medium-term, and long-term forecasting, depending on the forecasting horizon. Short-term load forecasting, which focuses on predicting power load for the next few hours, a day, or a few days, serves as the foundation for effective power system operation and analysis. It facilitates the optimization of the operating schedules of generating units, encompassing their start and stop times, and their expected output. The accuracy of short-term load forecasting is of paramount importance, as it directly impacts the efficient utilization of generating units [5]. Without precise short-term load forecasting, issues such as load shedding, partial or complete shutdowns, and voltage fluctuations become commonplace, resulting in significant damage to equipment and even endangering human lives.

Moreover, as the world pivots towards the increased adoption of renewable energy sources (RE) [6], our power grids have witnessed a substantial transformation in their composition and structure. This integration of renewable energy sources, such as wind and solar power, introduces a degree of unpredictability into our energy supply due to the intermittent and stochastic nature of these sources [7]. Consequently, ensuring a stable and secure power system operation becomes an even more complex endeavor, demanding meticulous power planning and precise load forecasting.

Short-term load forecasting methods are pivotal in achieving this precision, and they are divided into two main categories: statistical methods and machine learning methods [8], [9]. Machine learning-based load forecasting methods, such as the autoregressive integrated moving average model (ARIMA) [10], long short-term memory (LSTM) [11], generative adversarial network (GAN) [11], and convolutional neural network (CNN) [12], have gained prominence. These machine learning methods excel at capturing complex non-linear data features within load patterns [13]. They leverage the ability to discern similarities in electricity consumption across diverse power supply areas and customer types, allowing for more accurate and feasible load forecasting through the consideration of spatial-temporal coupling correlations.

A. Motivation

Based on the existing research, the following three deficiencies need to be fully considered to improve the forecasting effect of the spatial–temporal distribution of the system load; the lack of flexibility and scalability of traditional statistical methods, the high computational complexity of deep learning methods and the inability of existing methods to capture the spatial-temporal correlations in load patterns.

Considering these challenges, this paper proposes a novel short-term load forecasting model that uses a particle swarm optimized multi-head attention-augmented CNN-LSTM network. First, the proposed model uses a particle swarm optimization algorithm to search for the optimal hyperparameters of the CNN-LSTM network. This makes the model more robust to overfitting and improves its accuracy. Secondly, the multi-head attention mechanism is used to learn the importance of different features for the forecasting task. Lastly, a hybrid CNN-LSTM Model is used to help the system capture the spatial-temporal correlations in load patterns and improve its forecasting accuracy.

B. Contributions

This paper proposes the integration algorithm *PSO-A2C-LNet* for STLF. It works by using Particle Swarm Optimization, a swarm-based meta-heuristic algorithm to find the optimal values for select hyperparameters and then using several pre-processing methods for feature selection and extraction. Next, the attention mechanim via the hybrid model uses the spatial-temporal data it locates in the training set to predict features which are then used for the prediction. The experiment results showed that the proposed algorithm outperforms state-of-the-art methods in terms of accuracy, robustness, and computational efficiency. This paper makes the following contributions:

1) This paper proposes an efficient algorithm (PSO-A2C-LNet) that combines feature extraction and efficiency for STLF. PSO-A2C-LNet can further extract feature information more efficiently whiles maintaining scalability.
2) This paper's proposed model is can locate both temporal dependencies and long-term dependencies better by use of the Attention Mechanism, the Convolutional Neural Network (short-term and spatial dependencies) and the Long Short-Term Network (long-term dependencies). This leads to a higher accuracy because load demand is both dependent on short-term demand and long-term descriptions of data.
3) The effectiveness of PSO-A2C-LNet has been verified on three actual power load data sets (from Panama, France and the US). The comparison of PSO-A2C-LNet with A2C-LNet, a hybrid CNN-LSTM Model, CNN Models, LSTM Models, and a vanilla ANN Model showed that PSO-A2C-LNet has the best forecasting performance.

C. Structure of this paper

The subsequent sections of this research paper will be structured as follow; Section 2 will present the definition of terms. This section will provide comprehensive explanations and definitions of key terminology relevant to our study. Also, in this section we will delve into a detailed exposition of the proposed model architecture, offering a thorough understanding of its workings. Section 3 reports the results obtained from our experiments concerning the integrated algorithm. Additionally, this section will include the outcomes of our validation efforts using power load data sourced from the United States, France and Panama. Section 4 concludes this paper.

II. Materials and Methods

A. Definitions of Key Methodology

1) Convolutional Neural Network

A Convolutional Neural Network (CNN) is a deep learning model designed primarily for tasks involving images, but it can also be applied to other grid-like data, such as audio or time series data. CNNs are especially effective at capturing spatial dependencies within an input by using convolutional layers [14].

The CNN achieves the localization of spatial dependencies by using the following layers:

1. **Convolutional Layer**:

The core operation in a CNN is the convolution operation. Convolutional layers use learnable filters or kernels to scan the input data in a localized and overlapping manner. Each filter detects specific features, like edges, textures, or more complex patterns.

Mathematically, the 2D convolution operation is defined as follows:

$$(Y * X)(i,j) = \sum_{m=1}^{M} \sum_{n=1}^{N} X(i+m-1, j+n-1) \cdot Y(m,n)$$

Here, - Y is the filter (kernel) of size $M \times N$.
- X is the input data of size (W, H).
- (i, j) represents the coordinates of the output feature map.
- (m, n) iterates over the filter dimensions.

By sliding the filter across the input, the convolution operation computes feature maps that highlight different aspects of the input. This process effectively captures spatial dependencies.

2. **Pooling Layer**:

Pooling layers are often used to downsample the feature maps, reducing their spatial dimensions. Common pooling operations include max-pooling and average-pooling. Pooling helps make the network translation invariant and reduces the computational burden.

For max-pooling, the operation is defined as:

$$\text{Max-Pooling}(x, p, q) = \max_{i,j} x(p+i, q+j)$$

Here, x is the input feature map, and (p, q) represents the pooling window position. Max-pooling retains the most significant information within the window.

3. **Fully Connected Layer**:

After multiple convolutional and pooling layers, the spatial dimensions are reduced, and the network connects to one or more fully connected layers, also known as dense layers. These layers perform classification or regression tasks by learning high-level representations.

Recognizing and Exploiting Spatial Dependencies:
Recognizing and Exploiting spatial dependencies in Convolutional Neural Networks (CNNs) is facilitated through several key mechanisms [15]. Firstly, CNNs utilize local receptive fields, whereby each neuron is connected to a small region of the input data. This enables neurons to specialize in detecting specific features within their receptive fields, enabling the network to capture spatial dependencies at multiple scales. Additionally, weight sharing is a fundamental aspect of CNNs, where the same set of filters is applied consistently across the entire input. This weight sharing enables the network to learn patterns that remain invariant to translation, further enhancing its ability to capture spatial dependencies. Moreover, CNNs employ a hierarchical representation approach, where deeper layers in the network combine higher-level features derived from lower-level features. This hierarchical representation aids the network in comprehending complex spatial dependencies by gradually constructing abstractions. These mechanisms collectively empower CNNs to effectively model and exploit spatial dependencies in input data.

2) Long Short-Term Network

The Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture that is designed to capture and model sequential data while addressing the vanishing gradient problem that plagues traditional RNNs. LSTMs are particularly effective at locating and modeling long-term dependencies in sequential data.

LSTMs consist of multiple interconnected cells, each with its own set of gates and memory cells [16]. The primary components of an LSTM cell are:

Forget Gate (f_t): Controls what information from the previous cell state (C_{t-1}) should be discarded or kept. It takes the previous cell state and the current input (x_t) as input and produces a forget gate output.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Input Gate (i_t): Determines what new information should be added to the cell state. It takes the previous cell state and the current input and produces an input gate output.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

Candidate Cell State ($\tilde{C}_t$): This is a candidate new cell state, computed using the current input and a tanh activation function.

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

Cell State Update (C_t): The cell state is updated by combining the information retained from the previous cell state ($f_t \cdot C_{t-1}$) and the new candidate cell state ($i_t \cdot \tilde{C}_t$).

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

Output Gate (o_t): Determines what part of the cell state should be output as the final prediction. It takes the current input and the updated cell state and produces an output gate output.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

Hidden State (h_t): The hidden state is the output of the LSTM cell, which is used as the prediction and is also passed to the next time step. It is calculated by applying the output gate to the cell state.

$$h_t = o_t \cdot \tanh(C_t)$$

Achieving Long-Term Dependencies:
Long Short-Term Memory networks (LSTMs) address the vanishing gradient issue of traditional Recurrent Neural Networks (RNNs) by introducing key components: the cell state (C_t) and the forget gate (f_t) [17]. The forget gate dynamically adjusts (f_t) to enable LSTMs to remember or discard information from distant time steps, facilitating the capture of long-term dependencies. Meanwhile, the cell state (C_t) acts as a memory buffer, accumulating and passing relevant information across time steps, thus enabling the model to recognize and exploit long-term patterns within input sequences.

3) Multi-Head Attentional Mechanism

The Multi-Head Attention mechanism [18] is a key component of Transformer-based models, such as BERT and GPT, used for various natural language processing tasks. It excels at capturing extremely long-term dependencies in sequences of data.

Multi-Head Attention extends the idea of the self-attention mechanism [19] by employing multiple attention heads in parallel. Each attention head focuses on different parts of the input sequence, enabling the model to capture various types of information and dependencies simultaneously.

The primary components of Multi-Head Attention are as follows:

Query (Q), Key (K), and Value (V) Projections: For each attention head, we project the input sequence into three different spaces: query, key, and value. These projections are learned parameters.

Scaled Dot-Product Attention: Each attention head computes attention scores between the query (Q) and the keys (K) of the input sequence and then uses these scores to weight the values (V). The attention scores are computed as a scaled dot product:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V$$

Here, d_k is the dimension of the key vectors.

Concatenation and Linear Transformation: After computing the attention outputs for each head, we concatenate them and apply a linear transformation to obtain the final multi-head attention output:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h)W^O$$

Where Concat concatenates the outputs from all attention heads, and W^O is a learned linear transformation.

B. Model Architecture and Methodology

Talk about the proposed model and an algorithm structure for the paper

III. Model Architecture

This study proposes PSO-A2C-LNet, which uses various mechanisms explained above to extract more potential characteristics of power load data to provide better predictive ability. This algorithm effectively improves the prediction accuracy of STLF. The steps of the proposed algorithm are as follows:

Step 1. Convolutional and LSTM Layers: The model starts with an input layer, accepting sequences of four time steps with nine historical load information features. Subsequently, a one-dimensional Convolutional layer with 64 filters and a kernel size of 4 captures spatial patterns. To combat overfitting, Dropout Layer 1 with a dropout rate of 0.2 is added. Bidirectional LSTM Layer 1, equipped with 128 units and a hyperbolic tangent (tanh) activation function, models temporal dependencies both forwards and backwards, followed by Dropout Layer 2 to enhance generalization.

Step 2. Multi-Head Attention Mechanism: The crucial Multi-Head Attention Module operates on the output of Bidirectional LSTM Layer 1, enabling the model to focus on relevant features and learn their importance. Dropout Layer 3, with a dropout rate of 0.2, further enhances model robustness, and Layer Normalization ensures stable training across various inputs.

Step 3. Additional LSTM Layers: Bidirectional LSTM Layer 2, featuring 128 units, captures intricate temporal patterns, while Dropout Layer 4, with a dropout rate of 0.2, mitigates overfitting. The final LSTM Layer 3, also with 128 units and a hyperbolic tangent (tanh) activation function, generates the model's output.

Step 4. Output Layer: The output layer, a dense layer with one neuron and a linear activation function, produces the forecasted short-term load demand.

Step 5. PSO Optimization: To optimize model performance and convergence during training, we employ Particle Swarm Optimization (PSO) to fine-tune five critical hyperparameters. These include adaptive learning rate (within [0.001, 0.1]), batch size (within [1, 128]), number of epochs (within [100, 5000]), weight initialization techniques (options: Xavier, He, Random), and loss metrics (options: Mean Squared Error, Cross-Entropy, Mean Absolute Percentage Error).

The specific implementation process of the proposed algorithm is provided in Algorithm 1.

Algorithm 1: PSO-A2C-LNet

```
Data: Input data D
Result: Output data O
Initialize variables;
Extract features X and target values y from D;
Do Pre-processing;
Define architecture of model;
Start PSO;
while stopping criterion not met do
    Find optimal parameters using PSO;
    Check the fitness with defined model;
    Update variables and data structures;
Update variables globally;
Train the model on the training set;
Evaluate the model with the three error metrics;
Post-processing steps;
return O;
```

IV. Results and Discussion

This section comprehensively analyzes the STLF results by implementing the above model and testing it extensively on three datasets; ERCOT, RTE and the Panama Energy Dataset.The entire validation experiment is carried out on PyCharm Community Edition 2022.1 x64 environment with Windows 10 Pro and a 2.30 GHz Intel Core i5-8300H CPU, with 64-bit support and 8 GB RAM. Three regression evaluation metrics are introduced for quantitative analysis of the prediction results. The simulation effect and fitting degree of the different models are measured by the following indicators:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right| \times 100\%$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i|$$

where n: Number of Observations, Y_i: Actual values at data point i, $\hat{Y}_i$: Predicted values at data point i and $\bar{Y}$: Mean of the observed values.

Table I
Performance Metrics for Different Models on Various Datasets

Dataset	Model	R^2	MAPE (%)	MAE
Panama Dataset	A2C-LNet	0.85	2.8	8.1
	PSO-A2C-LNet	**0.88**	**1.9**	**7.3**
	Hybrid CNN-LSTM	0.92	3.1	8.7
	Vanilla CNN	0.81	3.9	11.2
	Vanilla LSTM	0.86	3.4	10.0
ERCOT Dataset	A2C-LNet	0.92	3.1	8.7
	PSO-A2C-LNet	0.87	2.1	9.5
	Hybrid CNN-LSTM	0.89	2.6	9.1
	Vanilla CNN	0.95	3.2	7.8
	Vanilla LSTM	0.91	4.0	8.5
RTE Dataset	A2C-LNet	0.78	2.4	12.3
	PSO-A2C-LNet	0.86	2.0	7.5
	Hybrid CNN-LSTM	0.89	2.9	7.2
	Vanilla CNN	0.79	4.2	12.0
	Vanilla LSTM	0.81	3.1	11.2

From the results in the table above, the PSO-A2C-LNet model consistently stands out. On the Panama Energy Dataset, it achieves the highest coefficient of determination (R^2) at 0.88, indicating strong predictive accuracy, along with the lowest mean absolute percentage error (MAPE) of 1.9% and the smallest mean absolute error (MAE) of 7.3, making it the top-performing model. In the ERCOT Dataset, PSO-A2C-LNet also delivers competitive results with an R^2 of 0.87, a MAPE of 2.1%, and a MAE of 9.5. Similarly, on the RTE Dataset, it outperforms other models with an R^2 score of 0.86, a lower MAPE of 2.0%, and a MAE of 7.5. These consistent results suggest that PSO-A2C-LNet exhibits robust predictive capabilities across diverse datasets.

While PSO-A2C-LNet excels on all datasets, the other models exhibit varying levels of performance. These comparative results emphasize the importance of model selection based on the specific dataset and application, with PSO-A2C-LNet emerging as a robust choice for diverse predictive tasks.

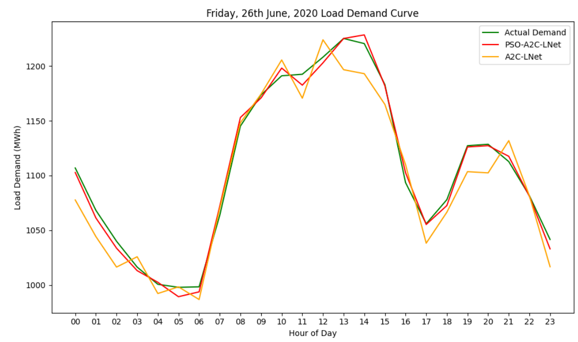

Figure 1. Graph of Actual Demand vs. A2C-LNet vs. PSO-A2C-LNet for a Day

1) *Comparison of results with results in literature*

These are the results of the A2C-LNet and the PSO-A2C-LNet on the testing dataset compared to other models in scientific literature.

Table II
Comparison of MAPE Results with Literature

Model	Best MAPE Results
A2C-LNet	2.8097
PSO-A2C-LNet (Our Proposed Model)	**1.9376**
Integrated CNN and LSTM Network [20]	3.49
LSTM network considering attention mechanism[10]	2.26
ANN-IEAMCGM-R [21]	3.59
TCN-LightGBM [22]	2.64
nonAda-GWN [23]	7.42
Ada-GWN [23]	6.83
Stacked XGB-LGBM-MLP [23]	2.69
GRU-CNN Hybrid Neural Network Model [24]	2.8839

V. Conclusion

In conclusion, this research paper has introduced a novel neural network architecture for short-term load forecasting, amalgamating Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) layers, reinforced by a Multi-Head Attention Mechanism. Empirical assessments confirm its superiority over traditional methods and standalone neural network models, with demonstrated applicability to real-world datasets.

Future work will focus on optimizing the proposed architecture, exploring further hyperparameter tuning, and investigating additional data preprocessing techniques for enhanced forecasting. Additionally, integrating robust data privacy measures, such as federated learning or secure enclaves, into the architecture is essential to address emerging privacy concerns in load forecasting, ensuring secure and privacy-preserving predictions while advancing the scalability and adaptability of the framework to diverse forecasting challenges and datasets.

Decalaration of Competing Interest

The author declares that there is no conflict of interest regarding the publication of this paper.

Acknowledgment

The author would like to thank Professor Philip Yaw Okyere for guiding the research.

References

[1] D. O. Frederick and A. E. Selase, "The effect of electric power fluctuations on the profitability and competitiveness of smes: A study of smes within the accra business district of ghana," *Journal of Cryptology*, vol. 6, pp. 32–48, 2014.

[2] N. D. Rao and S. Pachauri, "Energy access and living standards: Some observations on recent trends," *Environmental Research Letters*, vol. 12, no. 2, p. 025 011, 2017.

[3] T. M. Letcher, *11-storing electrical energy, editor (s): Trevor m. letcher, managing global warming*, 2019.

[4] P. Jiang, F. Liu, and Y. Song, "A hybrid forecasting model based on date-framework strategy and improved feature selection technology for short-term load forecasting," *Energy*, vol. 119, pp. 694–709, 2017.

[5] Y. Chen, P. B. Luh, C. Guan, *et al.*, "Short-term load forecasting: Similar day-based wavelet neural networks," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 322–330, 2009.

[6] O. Ellabban, H. Abu-Rub, and F. Blaabjerg, "Renewable energy resources: Current status, future prospects and their enabling technology," *Renewable and sustainable energy reviews*, vol. 39, pp. 748–764, 2014.

[7] D. Ahmed, M. Ebeed, A. Ali, A. S. Alghamdi, and S. Kamel, "Multi-objective energy management of a micro-grid considering stochastic nature of load and renewable energy resources," *Electronics*, vol. 10, no. 4, p. 403, 2021.

[8] Y. Hu, B. Qu, J. Wang, *et al.*, "Short-term load forecasting using multimodal evolutionary algorithm and random vector functional link network based ensemble learning," *Applied Energy*, vol. 285, p. 116 415, 2021.

[9] Y. Kim, H.-g. Son, and S. Kim, "Short term electricity load forecasting for institutional buildings," *Energy Reports*, vol. 5, pp. 1270–1280, 2019.

[10] J. Lin, J. Ma, J. Zhu, and Y. Cui, "Short-term load forecasting based on lstm networks considering attention mechanism," *International Journal of Electrical Power And Energy Systems*, vol. 137, p. 107 818, 2022.

[11] N. Huang, Q. He, J. Qi, *et al.*, "Multinodes interval electric vehicle day-ahead charging load forecasting based on joint adversarial generation," *International Journal of Electrical Power And Energy Systems*, vol. 143, p. 108 404, 2022.

[12] T.-Y. Kim and S.-B. Cho, "Predicting residential energy consumption using cnn-lstm neural networks," *Energy*, vol. 182, pp. 72–81, 2019.

[13] Y. Liu, Q. Wang, X. Wang, *et al.*, "Community enhanced graph convolutional networks," *Pattern Recognition Letters*, vol. 138, pp. 462–468, 2020.

[14] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," *IEEE transactions on neural networks and learning systems*, 2021.

[15] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network," *Remote Sensing*, vol. 9, no. 1, p. 67, 2017.

[16] R. C. Staudemeyer and E. R. Morris, "Understanding lstm–a tutorial into long short-term memory recurrent neural networks," *arXiv preprint arXiv:1909.09586*, 2019.

[17] S. Chandar, C. Sankar, E. Vorontsov, S. E. Kahou, and Y. Bengio, "Towards non-saturating recurrent units for modelling long-term dependencies," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 3280–3287.

[18] J.-B. Cordonnier, A. Loukas, and M. Jaggi, "Multi-head attention: Collaborate instead of concatenate," *arXiv preprint arXiv:2006.16362*, 2020.

[19] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, "Stand-alone self-attention in vision models," *Advances in neural information processing systems*, vol. 32, 2019.

[20] S. H. Rafi, Nahid-Al-Masood, S. R. Deeba, and E. Hossain, "A short-term load forecasting method using integrated cnn and lstm network," *IEEE Access*, vol. 9, pp. 32 436–32 448, 2021. DOI: 10.1109/ACCESS.2021.3060654.

[21] P. Singh, P. Dwivedi, and V. Kant, "A hybrid method based on neural network and improved environmental adaptation method using controlled gaussian mutation with real parameter for short-term load forecasting," *Energy*, vol. 174, pp. 460–477, 2019, ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2019.02.141. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544219303408.

[22] Y. Wang, J. Chen, X. Chen, *et al.*, "Short-term load forecasting for industrial customers based on tcn-lightgbm," *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 1984–1997, 2021. DOI: 10.1109/TPWRS.2020.3028133.

[23] W. Lin, D. Wu, and B. Boulet, "Spatial-temporal residential short-term load forecasting via graph neural networks," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5373–5384, 2021. DOI: 10.1109/TSG.2021.3093515.

[24] L. Wu, C. Kong, X. Hao, and W. Chen, "A Short-Term load forecasting method based on GRU-CNN hybrid neural network model," *Mathematical Problems in Engineering*, vol. 2020, p. 1 428 104, Mar. 2020.