Deep learning on Wavelets Analysis with Adam to predict Electricity load

Vuong Minh Chau[1a], Do Viet Bach[1b], Nguyen Quang Dat[1c], Vu Duc Hoan[2], Duong Xuan Bien[3]

[1]*Hanoi University of Science, Vietnam National University, Hanoi, Vietnam*
(a) vuongminhchau_hsgs2022@hus.edu.vn;
(b)dovietbach_hsgs2022@hus.edu.vn;
(c)nguyenquangdat@hus.edu.vn
[2]*The Air Force officer's College;*
ddnhat@lqdtu.edu.vn
[3]*Le Quy Don Technical University, Hanoi, Vietnam;*
duongxuanbien@lqdtu.edu.vn

***Abstract*—Hanoi is the biggest economic centre in Northern Vietnam with a population of nearly 10 million people. Its power supply system is supported and developed by the government, as part of the country's electrical distribution system. Because of this, forecasting Hanoi's electricity load is vital to improving the citizens' lives, especially when the power supply of Northern Vietnam is barely enough. In this research, we are proposing a new model that uses Wavelets along with Deep learning, with Adam as optimization in order to replace outdated manual statistical methods. Our model shows higher accuracy (best case is higher than 1.22%) when compared to the traditional methods of GRU and LSTM (without and with Adam optimization).**

***Keywords*—Time series forecasting, GRU, LSTM, Wavelets, Adam.**

I. INTRODUCTION

Electricity load of an area is an important factor as it directly affects the economy of that community. Power coordination, which is based on electricity load, poses as an urgent issue for the power generation industry and government.

In Vietnam, electricity load prediction is still operated on experience and uses traditional statistical tools. This has led to major errors in its operating system. That's why it's crucial that we come up with updated methods using machine learning in order to achieve higher accuracy, especially for short-term power load forecasting.

In this research, the authors brought out a deep learning method combined with pre-processing data using wavelets and Adam optimization. Our model had high accuracy and can be used alongside traditional methods to predict the electricity load with the highest possible accuracy rate.

II. RELATED WORKS

Neural Network methods have been used in past research for short-term time series forecasting. Some of these models have long running times and use complex algorithms only to produce modest results.

In 2018, Peng et al [1] worked on a way to predict electricity consumption changes, inspired by past works in the same field. Using LSTM, they garnered fairly high results in real-life test runs with error percentages of 4.01%, 5.37%, and 1.60%. An article written and published in 2018 by Chang et al [2] combined LSTM with Adam optimization with the goal to predict fluctuations in electricity prices. Experimenting on data from 2014 in New South Wales of Australia, their method had the highest MAPE rate of 6.61% in August.

In 2020, El-Hendawi et al [3] came up with a method of forecasting electricity load with very high accuracy. They achieved an error rate of 0.0191 when using Neural Network combined with MAPE, and an average error rate of 0.015 using the proposed wavelet neural network alongside MAPE.

In September 2021, Salleh et al [4] suggested a way of identifying abnormal points of progression in electricity consumption using LSTM along with several optimization techniques. Their model had the most successful results achieving error rates of 0.09 and 0.018 respectively for MSE and MAE when using the Adadelta optimizer.

Also in this year, Majeed et al [5] used Artificial Neural Networks merged with MAPE to predict the electricity load after 24, 48 and 72 hours with an error percentage of 1.87%, 1.98% and 1.78%.

In 2022, Kondaiah et al [6] suggested a way to predict short-term electricity load with a Novel Wavelet-Based Ensemble method. This method is tested with data from Ontario, Canada and had the highest MAPE rate of 1.911% in spring and summer.

III. METHODOLOGY

A. Wavelet analysis - WA

Wavelet Analysis is a powerful algorithm invented by Mallat that turns original detected the signals into non-noising ones. This method has been a crucial tool in demonstrating time frequency in time series domains. High-range filters generate detailed coefficients while approximation sequences are generated by low-pass filters. Wavelet transformation is divided into 3 separate classes of continuous, discrete, and multi-resolution based. Our research applies DWT (discrete wavelet transform) and uses wavelet theory, as shown in this equation below:

$$W_{(m,n)} = 2^{\frac{-m}{n}} \sum_{t-1}^{N-1} \psi\left(\frac{t-2^m n}{2^m}\right) \quad (1)$$

$2^m = s$ here represents the scaling parameter $2^m n = \tau$, which is the DWT translation parameter, calculates the degree of confinement. When $|s| > 1$, this means the wavelet signal has reached out from its root signal while $0 < |s| < 1$ means the original wavelet signal has been squeezed down from the original.

Fig. 1 shows the input data of first-order wavelet being split into two components of low-frequency and high-frequency. n is number of values in the original input time series.

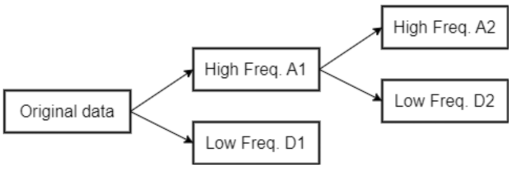

Fig. 1. Wavelet analysis

With A_1 a 2nd-order wavelet is used to split the low-frequency D_2 and the high-frequency A_2. As a result, we get the time series ($D_1, D_2, .., D_N, A_N$). The high-frequency component A_N is noisy and can be deleted after some splitting. The number N - Wavelet degrees can now be determined using an algorithm from [7]:

$$N = \lg(n) \quad (2)$$

B. Long-short term memory - LSTM

As mentioned above, RNNs show good performance with short-term dependencies in data series but are weak when dealing with longer ones. In 1997, Schmidhuber [[8] proposed the LSTM model, which is an updated version of RNN and has the ability to confront its existing problems [9], [10]. In fact, to store information during the training period, LSTM has been designed with 3 gates and a final processing step in the cells.

LSTM input value at $t - step$ is x_t, and output at the $t-1$ step is h_{t-1}. The input is filled by gates (with $sigmoid$ function) and the output value is in range $[0,1]$. If the output is 0, all inputs are erased. If not, the data stays the same. Then, the output at h_t and the cell state at t are determined by this process shown below:

Step 1: Forget gates remove data from the state's cell:

$$f_t = \sigma(V_f[h_{t-1}, x_t] + \text{bias}_f) \quad (3)$$

Step 2: New data is kept in the cell state. In the first phase, the input gate (with the function $sigmoid$) updates the node with new values. During the second phase, $\tanh$ class uses a new vector $\tilde{c}_t$:

$$i_t = \sigma(V_i[h_{t-1}, x_t] + bias_i) \quad (4)$$

$$\tilde{c}_t = \tanh(V_{\tilde{c}}[h_{t-1}, x_t] + bias_{\tilde{c}}) \quad (5)$$

Step 3: State cells are updated from C_{t-1} to C_t :

$$c_t = f_t \otimes c_{t-1} \otimes i_t \otimes \tilde{c}_t \quad (6)$$

Step 4: Calculating the output. Initially, the data outside of the LSTM unit are examined by the output gate evaluates. Then, state becomes a passed node that ranges from -1 to 1. The final output is then found through this formula:

$$o_t = \sigma(V_0[h_{t-1}, x_t] + bias_0) \quad (7)$$

$$h_t = o_t \otimes \tanh(h_t) \quad (8)$$

($V_f, V_i, V_{\tilde{c}}$ and V_0 are the LSTM parameter, and $bias_f, bias_i, bias_{\tilde{c}}$, $bias_0$ are LSTM model's biases).

C. Gated recurrent unit (GRU)

Kyung Hyun Cho et al., October 2014 [11] and J. Chung et al., 2015 [12] proposed gated recurrent unit (GRU), a simplified version of LSTM. Similar to LSTM, this model has some gate functionalities and can be widely applied. However, it doesn't have a memory cell. GRU functions can be summarized in these formulas:

$$h_t = (1 - z_t)h_{t-1} + z_t \tilde{h}_t \quad (9)$$

$$z_t = \sigma(V_z x_t + U_z h_{t-1}) \quad (10)$$

$$\tilde{h}_t = V_h x_t + U_{r_t h_{t-1}} \quad (11)$$

$$r_t = \sigma(V_r x_t + U_{r h_{t-1}}) \quad (12)$$

where h_t and z_t are the GRU outputs, r_t is the update and reset gate, $\tilde{h}_t$:candidate output; V_z, V_h, V_r and U_r are the GRU matrices.

D. Adam-optimization

Algorithm 1 ADAM-optimization

1: **Input** n : Stepsize

2: **Input** γ_1, γ_2 : Exponential decay rates for temporary estimation

3: **Input** $f(\theta)$: Stochastic objective function parameters θ

4: **Input** θ_0 : Initial parameter vector

5: $m_0 := 0$;

6: $v_0 := 0$;

7: $i := 0$;

8: **while** θ_t not converged **do**

9: $i := i + 1$

10: $g_i := \nabla_\theta f_i(\theta_{i-1})$

$m_i := \gamma_1 m_{i-1} + g_i(1-\gamma_1)$

$v_t := \gamma_2 v_{i-1} + g_i^2(1-\gamma_2)$

$\hat{m}_i := m_i \,/\, (1+\gamma_1^t)$

$\hat{v}_i := v_i \,/\, (1+\gamma_2^t)$

$\theta_i := \theta_{i-1} - n\hat{m}_i \big/ (\sqrt{v_i} + \varepsilon)$

end while

return θ_i

This is the pseudo-code of our proposed algorithm $Adam$ [13]. Consider $f(\theta)$ a noisy objective function: a stochastic scalar function of differentiable w.r.t parameters θ. We aim to minimize the expected value of this function, $E\left[f(\theta)\right]$ w.r.t its parameters θ. We denote the realizations of the stochastic function at subsequent time stamps $1,..,T$ with $f_1(\theta),..,f_T(\theta)$. The stochasticity might come from the evaluation of random datapoint mini samples, or arise from inherent function noise. With $g_t = \nabla_\theta f_t(\theta)$ the gradient is denoted, i.e. vector of partial derivatives of $f_t, w.r.t \;\; \theta$ evaluated at timestep t.

E. Proposed model: WA-NN-Adam

In our proposed model, the original data is firstly filtered by WA (Wavelet Analysis). The noisy part of the input dataset is deleted, which leaves behind smooth data that can be used in machine learning. LSTM and GRU models are then used. They both run with 8, 16, 32, 64, 128 hidden layers with a batch size of 100 and epoch of 10. The learning rate is set at 0.01. Adam is then used as an optimization method. Each model runs 10 times with each set of parameters. The end result is the average value of all machine runs.

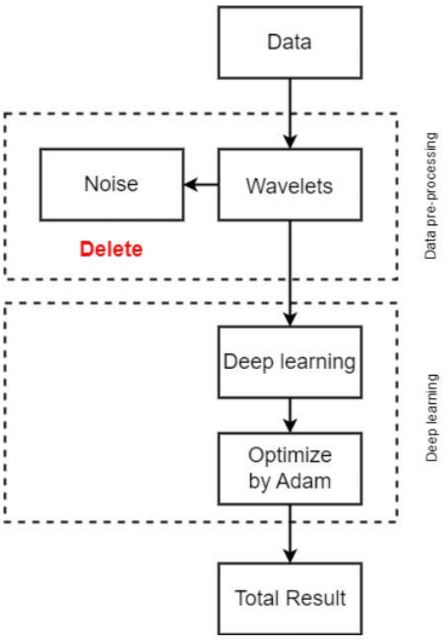

Fig. 2. Proposed flowchart

IV. Experiment and Results

A. Data

Our electrical load data is taken from a large urban area (all residential) in Northern Vietnam. The figures are taken from January 1, 2015 to August 30, 2019. The values of this datasets taken at 8 PM every day.

By running this dataset, we can generate indicators to forecast power consumption at peak times as well as off-peak times.

B. Criteria for comparison (Accuracy)

2 criteria are used in order to com pare the results: Root Mean Squared Error and Mean Absolute Percentage Error (RMSE and MAPE):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}; MAPE = \sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \qquad (13)$$

C. Results

The table below gives information on the result of the model after applying our actual data.

TABLE 1

RESULTS OF DATA.

Models	Compare	
	RMSE	MAPE
GRU	741	1.12%
LSTM	744	1.13%
GRU-Adam	690	1.04%
LSTM-Adam	656	1.01%
WA-GRU-Adam	619	0.98%
WA-LSTM-Adam	614	0.97%

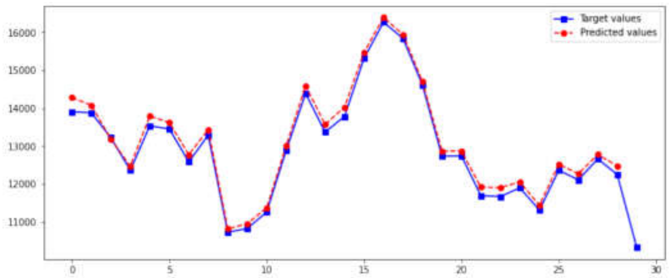

Fig. 3: Result of GRU model. This is the first 30 values

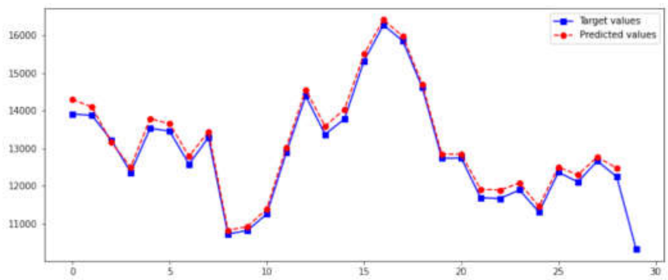

Fig. 4: Result of LSTM model. This is the first 30 values

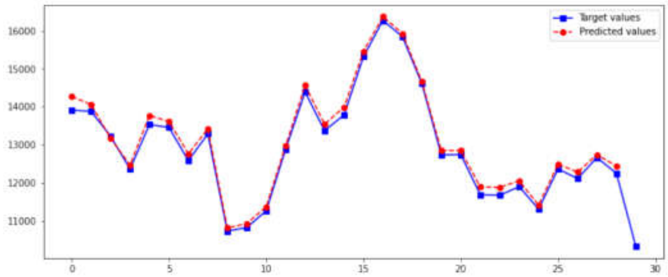

Fig. 5: Result of GRU-Adam model. This is the first 30 values

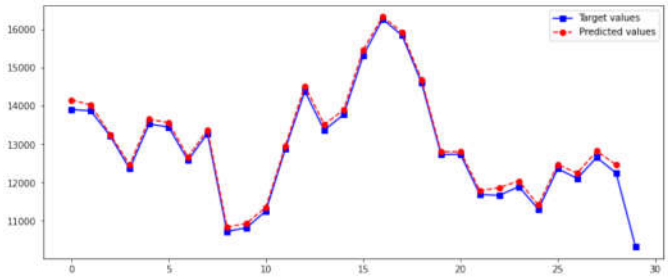

Fig. 6: Result of LSTM-Adam model. This is the first 30 values

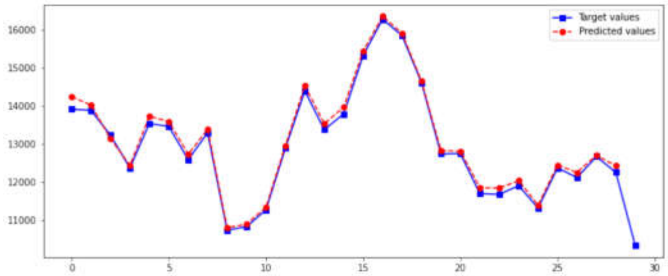

Fig. 7: Result of WA-GRU-Adam model. This is the first 30 values

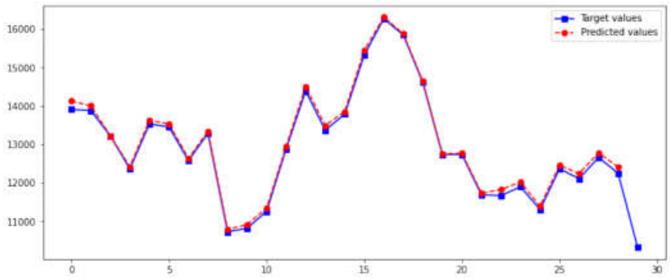

Fig. 8: Result of WA-LSTM-Adam model. This is the first 30 values

The Tab. 1 shows the results of the model after applying data of electricity load in Hanoi, biggest city in north Vietnam.

Each combination model's outcomes are better than those of the individual models, as illustrated in the table. The basic LSTM model using RMSE is the highest at 744, followed by GRU, which has an RMSE of 741 and a remarkably high MAPE of 1.12%, a sign of inaccuracy. (Aside from GRU and LSTM, the GRU-Adam model performs third worst in all criteria, with the exception of MAPE. However, this model's MAPE was higher than that of the hybrid LSTM-Adam model because the MAPE criteria are based on a relative error. The difference between 0.34 and 0.03 is comparatively insignificant).

The RMSE and MAPE differences between GRU-Adam and the LSTM-Adam model are just 0.34% RMSE and 0.03% MAPE, respectively. The suggested WA-GRU-Adam and WA-LSTM-Adam models had the lowest RMSE and MAPE values, with a 19.71% lower RMSE and a 14.3% better MAPE for GRU and 21.17% lower RMSE and a 16.5% better MAPE for LSTM than its hybrid adversary.

V. CONCLUSION AND FUTURE WORK

In this research, we have suggested some methods which use the core as a machine learning structure (Neural Network), combined with pre-processing and optimization tools. These combinations helped minimized weaknesses of the traditional methods, thereby increasing the accuracy of the final model. For this reason, our model of WA - Neural Network - Adam achieved higher accuracy rates compared to the traditional structures of LSTM and GRU.

We hope to improve our current machine learning method for future works, in order to achieve even higher accuracy rates. Some improvement directions can be considered below:

- Multi-time series for input data.
- More complicated hybrid models with more components.

ACKNOWLEDGMENT

We thank Hanoi University of Science, Vietnam National University, Hanoi, Vietnam for their help during our research.

REFERENCES

[1] L. Peng, S. Liu, R. Liu, L. Wang, "Effective long short-term memory with differential evolution algorithm for electricity price prediction", Energy, vol 162, pp 1301-1314, 2018

[2] Z. Chang, Y. Zhang, W. Chen, "Effective Adam-Optimized LSTM Neural Network for Electricity Price Forecasting", IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 2018

[3] M. El-Hendawia, Z. Wang, "An ensemble metho d of full wavelet packet transforms and neural network for short term electrical load forecasting", Electric Power Systems Research, Vol 182, 2020.

[4] N. S. M. Salleh, M. Saripuddin, A. Suliman, B. N. Jørgensen, "Electricity Anomaly Point Detection using Unsupervised Technique Based on Electricity Load Prediction Derived from Long Short-Term Memory", 2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS), 2021.

[5] M. A. Majeed, S. Mudgal, L. Tak, J. Popat, H. Kakadiya, N. K. Singh, V. Mahajan, "Artificial Neural Network Modelling for Short Term Load Forecasting", 2021 International Conference on Technology and Policy in Energy and Electric Power (ICT-PEP), 2021.

[6] V. Y. Kondaiah, B. Saravanan, "Short-Term Load Forecasting with a Novel Wavelet-Based Ensemble Method", Energies, vol 15, no. 14, 2022.

[7] W. Wang, S. Ding, "Wavelet network model and its application to the predication of hydrology", Natural and Science, vol 1, no. 1, pp 67-71, 2003.

[8] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[9] F. A. Gers, J. Schmidhuber and F. Cummins, "Learning to Forget: Continual Prediction with LSTM", Neural Computation, vol. 12, no. 10, pp. 2451-2471, 2000.

[10] R. Pascanu, T. Mikolov, Y. Bengio, "On the difficulty of training recurrent neural networks", Proceedings of the 30th International Conference on Machine Learning, PMLR 28 (3), pp. 1310-1318, 2013.

[11] K. Cho, B. Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk Y. Bengio, "Learning Phrase Representations using RNN Encoder-

Decoder for Statistical Machine Translation”, CoRR, pp 1724-1734, 2014.
[12] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, “Gated Feedback Recurrent Neural Networks”, Proceedings of the 32nd International Conference on Machine Learning, 2015.
[13] D. P. Kingma, J. Ba, “Adam: A Method for Stochastic Optimization”, 3rd International Conference for Learning Representations, San Diego, 2015.