Toward Enhancing Mid-Term Load Forecasting: RNN-Based Models vs Transformer-Based Models

Quang Huy Giap[1], Thi Thanh Quynh Nguyen[1,*], Trung Hieu Trinh[1], and Ky Trung Nguyen[2]

ARTICLE INFO

Article history:
Received: 7 September 2023
Revised: 30 November 2023
Accepted: 27 December 2023
Online: 20 June 2024

Keywords:
Mid-term load forecasting
Transformer-based model
Recurrent neural network
informer

ABSTRACT

Enhancing accuracy in mid-term load forecasting contributes to improving the efficiency of electricity system management and operation. This paper thoroughly examines different models to facilitate accurate mid-term load forecasting and identifies the key factors influencing the forecast results. Real world load data is employed to assess the effectiveness of popular models, including variants of regression neural network (RNN) models and transformer-based models. The experimental findings demonstrate that the transformer-based model outperforms RNN models, achieving remarkable accuracy in capturing load variations with a Mean Absolute Percentage Error (MAPE) of less than 3% for forecasts up to six months ahead. Meanwhile, the RNN models have demonstrated remarkable effectiveness in forecasting months characterized by significant load volatility. Another advantage of the transformer-based model is its reduced time complexity compared to other regression models with the same input length. This feature highlights the practicality and efficiency of the transformer-based model for mid-term load forecasting.

1. INTRODUCTION

Amid climate change and the development of renewable energy sources, Mid-Term Load Forecasting (MTLF) plays a crucial role in ensuring energy security and environmental protection.

By providing important information on electricity production decisions, system operation planning, and risk management in electricity supply, mid-term load forecasting can lead to more profitable electricity production and reduced risk in the production process. Furthermore, mid-term load forecasting can also provide valuable insights into electricity consumption trends and future energy demand, thereby helping investors and managers plan sustainable and energy efficient power systems. In fact, forecasting errors can have a significant impact on the cost of generating electricity, whether they are positive or negative. Even a small improvement in load forecasting accuracy, such as a 1% reduction in mean absolute percentage error (MAPE), can lead to substantial cost savings on the generation side, ranging from 0.1% to 0.3% [1]

However, mid-term load forecasting is very difficult for accurate results. Mid-term load forecasting will depend on many factors such as historical data, the influence of uncertain input factors, rapid changes in the environment, and social factors. In addition, building mid-term load forecasting models needs to be selected appropriately for the specific conditions of each region. For example, a mid-term load forecasting model for a large city may not be suitable for a rural area, as the factors affecting electricity demand in these two areas may differ. This requires necessary evaluation of the characteristics of the regions and selection of appropriate datasets and models to provide accurate forecasting.

There are numerous proposed approaches in the problem of load forecasting. It can be time series method, the econometric approach, the consumer investigation approach, the parametric techniques, the Artificial Intelligent techniques or Other combinative methods. The econometric approach does not suitable well with MTLF because timeline forecasting lasts several weeks to months, it's too short to estimate the relation between the energy demand and economic effective variables correctly. The time series forecasting method is based on the assumption that historical data exhibits correlation. These series can be classified as linear, polynomial, or logarithmic. The parametric technique such as Error-Trend-Seasonal model (ETS), Auto Regressive (AR), Moving Average (MA), AR Integrated MA (ARIMA), and ARIMA with External Variables (ARIMAX) invest to establish a linear mathematical relationship between the inputs (load and effective variables on load) and output (the predicted load).

[1]*The University of Danang - University of Science and Technology, Danang City, Viet Nam.*
[2]*School of Computer Science and Engineering, International University – Vietnam National University, Ho Chi Minh City, Viet Nam.*
**Corresponding author:* Thi Thanh Quynh Nguyen; Phone: +84-369-360-592; Email: nttquynh@dut.udn.vn.

If the determination of input variables is incomplete, the prediction model using these techniques is likely to exhibit a significant margin of error. Meanwhile, Artificial Intelligence (AI) technique referred to fuzzy logics, Artificial Neural Network (ANN) and Support Vector Regression (SVR). The most popular type of Artificial Neural Network (ANN) used for load forecasting is the multi-layer perceptron (MLP). MLP have a higher classification accuracy in comparison to fuzzy method [2]. There are many proposals called variations of ANN models to improve forecasting accuracy such as combining many methods such as ETS-ANN [3], ARIMA-ANN [4]. Evolutionary optimization algorithms have also been used in training and optimizing neural networks (ANN) and fuzzy systems. The evolutionary algorithms such as PSO, GA [5] used to fine-tune the parameters of the model can cause prolong the training time. Consumer investigation approach classify load effect factors into compound approach and autonomous approach. In compound approach, load effect factors include grid load in previous time, climate data, the social-economic index and energy politics. In autonomous approach, load effect factors include weather and load data, other social economic variables are not considered. The MTLF has a distinct time horizon, typically ranging from several months to one or two years [5], [6]. If the horizon is less than one year, load in previous time and meteorology data are the main variables, thus the autonomous approach is suitable [7]. However, precisely forecasting weather conditions over extended periods, like several months, presents noteworthy challenges. The incorporation of weather variables into the training model raises concerns about the practical accuracy of the forecasting model.

Aside from weather and economic factors that are known to impact electricity consumption, patterns related to trend and seasonality are also taken into account as training parameters to enhance prediction accuracy and minimize computation time [8]. In recent years, time-series methods have great investigation in load forecasting domain. When processing long sequences, RNNs can suffer from a significant issue known as vanishing or exploding gradient. To address this problem, a specific type of RNN called Long Short-Term Memory (LSTM) was introduced [13]. The LSTM architecture consists of a cell and multiple non-linear gates that regulate the flow of data within the cell, determining which information should be retained and which should be propagated to the next time step. Research has demonstrated that LSTMs outperform classical statistical and machine learning models, including ARIMA, support vector machines, and classical neural networks [9]. However, LSTM is a feedforward architecture with simple repeating memory units, which can only learn from previous data. Gated Recurrent Unit (GRU) has a simpler architecture than LSTM, both LSTM and GRU have equivalent accuracy but GRU has better performance cause its computations speed is faster. To fully capture the information in the data, the bi-directional LSTM architecture [10] is proposed. Bi-directional LSTM uses both past and future information of a sequence to predict the current result. The use of both directions allows the model to learn complex relationships among the components of the data sequence, thus increasing the accuracy of the prediction. Another alternative to RNN-based models is the encoder-decoder attention mechanism [11] and Transformer [12]. Attention-based architectures, including the Transformer, can address the vanishing or exploding gradient problem and are able to focus on any part of the input sequence during forecasting. Although Transformers have shown impressive accuracy in forecasting in natural language processing and image processing [12], one of their drawbacks is their high computational cost, which increases quadratically with the input size. Besides, Transformer isn't dedicated to predict for data that exhibit cyclical patterns [13].

Unlike LSTM, N-beat uses a time-aggregation architecture and can be optimized for multi-step forecasting [14] but the improvement in accuracy is not impressive in comparison to Transformer so it is not focused on this paper. There are many time series Transformers with different attention modules such as FEDformer, Reformer, LogTrans, etc. Paper [15] proposes an efficient transformer-based model for Long Sequence Time-series Forecasting named Informer. Informer can deal with severe issues of Transformer such as quadratic time complexity, high memory usage, and the inherent limitation of the encoder-decoder architecture. Informer has superior performance in short-term load forecasting in comparison to other Transformer-based models [16]. Briefly, there are various methods proposed in the literature to improve the accuracy of time-series forecasting. However, these methods are often scattered across different papers, making it difficult to compare and evaluate which approach is superior. Moreover, methods designed for differential contexts may not be suitable for mid-term load forecasting, and datasets with limited variability may not be representative of specific regions.

This paper aims to address these challenges by focusing on mid-term load forecasting for a six-month period. Specifically, the paper aims to: (i) provide an overview of load forecasting and mid-term load forecasting, (ii) focus on the effectiveness of an autonomous approach only using grid load for mid-term load forecasting, and (iii) determine which deep learning model outperforms others in forecasting real data in agriculture province.

In this paper, section 1 presents the current state of load forecasting. The structure of deep learning models and a description of the dataset are provided in section 2 and section 3, respectively. In section 4 experimental results as well as relative evaluations are discussed. Finally, the conclusion and future work are presented in section 5.

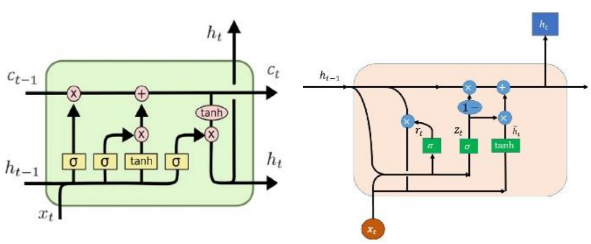
2. ARCHITECTURE OF RNN-BASED MODELS VS. TRANSFORMER-BASED MODELS

As aforementioned, there are various methods proposed for load forecasting. In terms of this paper, we focus on outperforming methods: variant of RNN-based models and Transformer-based model specify Informer.

2.1. RNN-based models

2.1.1 Sequences of memory cells - LSTM, GRU

LSTM or GRU represents an artificial recurrent neural network (RNN) where the output from the previous step serves as input for the current step, establishing a crucial "feedback" connection. This network demonstrates the ability to grasp extensive dependencies over extended intervals and retain information over prolonged durations, facilitated by a memory cell linked to the prior time step. The structure of LSTM or GRU manifests as a sequence of cells, enabling controlled information passage through gates, as depicted in Figure 1.



(a)LSTM unit (b)GRU unit

Fig. 1. Cells description of LSTM and GRU implementation

In the t_{th} cell, the input is composed of three components: the current input information x_t, the previous state information of the cell h_{t-1}, and the memory cell c_{t-1}. Within the LSTM framework, information traverses through three distinct gates: the update gate G_u, the forget gates G_f (utilized for cell memory c_t updating), and the output gate G_o (employed to update the state cell h_{t-1}). On the other hand, the GRU architecture employs a simpler design with two gates: the update gate G_u and the reset gate G_r . The principle of LSTM cell and GRU cell are defined in equation (1) and (2) respectively.

$$\begin{cases} \widetilde{c_t} = \tanh\left(\mathrm{K_c}\left(\mathrm{h_{\{t-1\}}}, \mathrm{x_t}\right) + \mathrm{b_c}\right) \\ \mathrm{G_u} = \sigma\left(\mathrm{K_u}\left(\mathrm{h_{\{t-1\}}}, \mathrm{x_t}\right) + \mathrm{b_u}\right) \\ \mathrm{G_f} = \sigma(\mathrm{K_f}(\mathrm{h_{t-1}}, \mathrm{x_t}) + \mathrm{b_f}) \\ \mathrm{G_o} = \sigma\left(\mathrm{K_o}\left(\mathrm{h_{\{t-1\}}}, \mathrm{x_t}\right) + \mathrm{b_o}\right) \\ c_t = G_u . \widetilde{c_t} + G_f . c_{t-1} \\ \mathrm{h_t} = \mathrm{G_o}.\tanh(\mathrm{c_t}) \\ \hat{y} = \mathrm{K_s}.\mathrm{h_t} + \mathrm{b_s} \end{cases} \quad (1)$$

$$\begin{cases} \mathrm{G_u} = \sigma(\mathrm{K_u}(\mathrm{c_{t-1}}, \mathrm{x_t}) + \mathrm{b_u}) \\ \mathrm{G_r} = \sigma(\mathrm{K_r}(\mathrm{c_{t-1}}, \mathrm{x_t}) + \mathrm{b_r}) \\ \mathrm{c_t} = \mathrm{G_u}.\widetilde{c_t} + (1 - \mathrm{G_u}).\mathrm{c_{t-1}} \\ \mathrm{h_t} = \mathrm{c_t} \end{cases} \quad (2)$$

Here, σ and $tanh$ function as activation functions, while $K_c, K_u, K_f, K_o, K_s, K_r$ denote the weight matrix parameters. The parameters $b_c, b_u, b_f, b_o, b_r, b_s$ represent bias vectors, and $\widetilde{c_t}$ signifies a novel candidate for the cell state.

2. 1.2 Attention-based long short-term memory (At-LSTM)

Being a component of the Seq2Seq model [17], the At-LSTM has exhibited intriguing attributes across various domains. Notably in language translation, it excels in seizing vital segments of the input sequence. This refinement amplifies the linkage between input and output data, culminating in heightened precision of model forecasts. From a technical perspective, the attention mechanism is employed within the uppermost layer of LSTM as presented in Fig. 2.

The At-LSTM's realization is laid out in equation (3), with H representing the features derived from the prediction mode, $H = [h_{t-T+1,}, \dots, h_t]$; The attention score function of H is expressed as e, and α is employed to signify the attention weights.

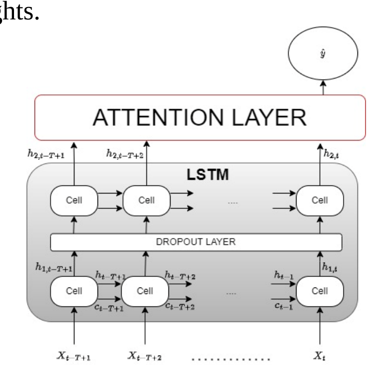


Fig. 2. At-LSTM implementation.

The context vector of the attention layer is represented by r. Lastly, the parameter V_α, which harmonizes with the remainder of the system, is acquired through learning.

$$\begin{cases} \widehat{y_t} = \tanh\left(\mathrm{K_c}(\mathrm{l_t}, \mathrm{h_t})\right) \\ \mathrm{l_t} = \sum_{i=1}^{T} \alpha_{ti} . \mathrm{h_i} \\ \alpha_{ti} = \dfrac{\mathrm{e^{e_{tj}}}}{\sum_{j=1}^{T} e^{e_{tj}}} \\ \mathrm{e_{ti}} = \mathrm{V_a^T}\tanh\left(\mathrm{K_a}(\mathrm{h_t}, \mathrm{H})\right) \end{cases} \quad (3)$$

2.1.3 Bidirectional LSTM

A Bidirectional LSTM, commonly abbreviated as bi-LSTM, is a sequence processing model that comprises two distinct

LSTM layers. One-layer processes the input in a forward direction, while the other processes it in a reverse direction. This configuration significantly enhances the volume of information accessible to the network, thereby enriching the contextual understanding available to the algorithm.

Incorporating a bi-LSTM involves introducing an additional LSTM layer that reverses the flow of information. In this arrangement, the input sequence is processed in a backward manner within the supplementary LSTM layer. The outcomes from both LSTM layers are then combined using various methods such as averaging, summation, multiplication, or concatenation.

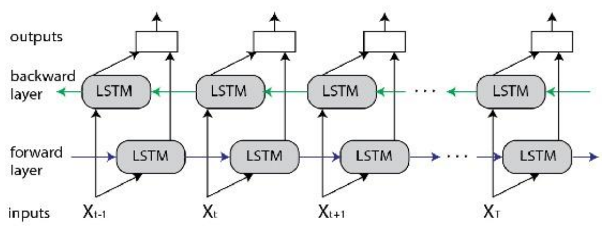


Fig. 3. Bidirectional LSTM implementation.

2.2 Informer: A transformer-based model

Informer's architecture is completely different to RNN. It uses stacked self-attention layers and fully connected layers applied to both the encoder and decoder components. Informer is the optimized version of the Transformer model designed to address the high predictability of long sequence time-series forecasting (LSTF) [18]. It replaces the conventional self-attention mechanism with the recently introduced ProbSparse self-attention. The blue trapezoid in Figure 4 represents the self-attention distillation process, extracting dominant attention and significantly reducing the network size. Additionally, the utilization of a generative-style decoder designed for direct multi-step (DMS) predictions assists Informer in enhancing the speed of inference for long sequence forecasts, departing from the conventional step-by-step approach.

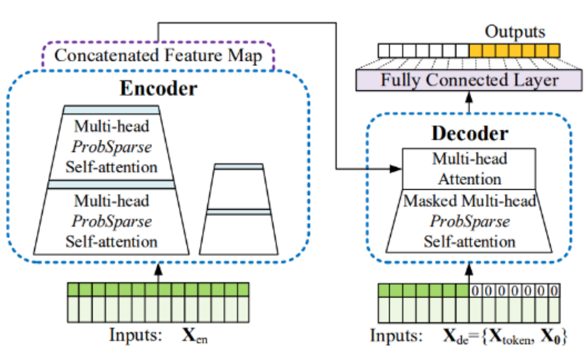


Fig. 4. Informer implementation [15].

The distillation process conveys information from the j_{th} layer to the $(j+1)^{th}$ layer as shown in equation (4):

$$X_{t+1}^{t} = MaxPool\left(ELU\left(Con1d\left[X_j^t\right]_{AB}\right)\right) \tag{4}$$

Here $[.]_{AB}$ denotes the attention block that encompasses ProbSparse self-attention and the fundamental operations as defined in equation (5). ELU is an activation function and $MaxPool$ is a downsampling.

$$M(Q,K,V) = softmax\left(\frac{\bar{Q}K^T}{\sqrt{d}}\right)V \tag{5}$$

where, Q, K, V denote matrix of query, key, value, $\bar{Q}$ is a square matrix and d is a input dimension.

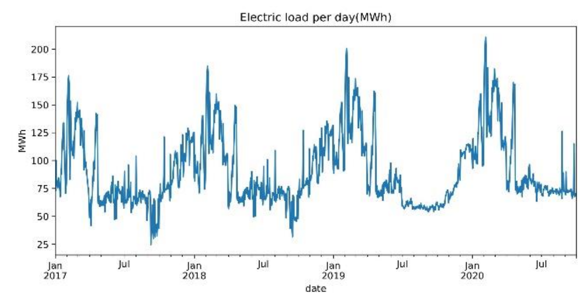


Fig. 5. Electric consumption in IA Grai district from 01/2017.

3. DATASET

The data of electrical consumption was collected daily from feeder 471/110 Dien Hong in IA Grai district, an agricultural area in Vietnam, by Distribution Management and Data System (DMDS).

3.1 Data analysis

The electricity consumption data of IA Grai district collected from January 2017 to September 2020 is presented in figure 5.

Upon examination of the graph, it is evident that power consumption exhibits seasonal variations. This is due to the fact that the primary economic activity in IA Grai district is agriculture and forestry, making electric consumption highly dependent on climate and weather factors (i.e., undefined factors in the future). During the dry season, the increased demand for pumping water for coffee plants in January, February, March, April, November, and December results in a higher electric load. Conversely, during the rainy season months from May to October, electric consumption is lower as it is mainly for household needs.

Generally, electric consumption has been gradually increasing from 2017 to 2020 due to economic development and growth in Gross Domestic Product (GDP).

The distribution of average annual electricity consumption, as depicted in figure 6, is not following a normal distribution, making it difficult to achieve high accuracy in predicting this data.

The data collected at feeder 470/110 in Ia Grai district over a 4-year period (2017-2020) will be split into 3 datasets, as shown in fig. 7:

- Data from 01/2017 - 12/2018 used for training
- Data from 01/2019 - 12/2019 used for validation

• Data from 01/2020 - 06/2020 used for testing

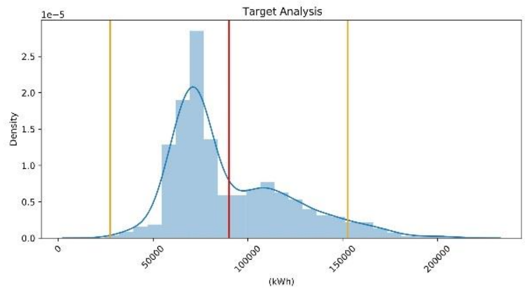


Fig. 6. Annual electric consumption distribution.

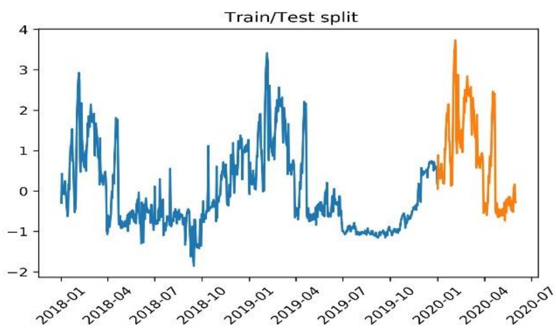


Fig. 7. Data split for training and testing.

For regression predictive modeling problems, such as electric load forecasting, the loss function can be Mean Absolute Error (MAE), Mean Squared Error (MSE), or Mean Squared Logarithmic Error (MSLE).

Mathematically, MSE is preferred under the maximum likelihood inference framework when the target variable's distribution is Gaussian. However, in our case, the target variable's distribution is mostly Gaussian with outliers, so MAE is a more appropriate loss function as it is more robust to outliers.

3.2 Influencing factors and model parameters

For mid-term forecasts lasting from a few months to a year, the impact of economic indicators and social development projections on changes in energy consumption is very small. Besides, weather forecasts sometimes lack reliability in the medium term. Therefore, to increase the accuracy of mid-term load forecasting models, we ignore influencing factors such as weather, economic factors, and social development, the forecasting model can only base on historical grid load indicators.

The forecasting program is written in Python code using the Keras and Torch library. Hyperparameters in deep learning models are referenced from articles or trial and error methods. For recurrent network models such as LSTM, and GRU, one layer is suggested to be better performance than multiple layers [19].

For Attention LSTM, multiplicative is chosen for the score function. Bidirectional model uses two layers one LSTM and one GRU. Informer model comprises an Encoder with three layers and a Decoder with two layers [15]. Dimension of the model and feedforward are 512 and 2048 respectively. The batch size for each
epoch is 32 for all models.

To evaluate the performance of models and determine their accuracy in predicting future outcomes, we use three fundamental metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) (%). Suppose n is the number of data points, y_i denotes the true value, and $\widehat{y_i}$ stands for the predicted value. The equations of metrics are shown in equations (6), (7), (8):

$$MAE = \frac{1}{n}\Sigma|y_i - \widehat{y_i}| \tag{6}$$

$$MSE = \frac{1}{n}\Sigma(y_i - \widehat{y_i})^2 \tag{7}$$

$$MAPE = \frac{1}{n}\Sigma\left|\frac{y_i - \widehat{y_i}}{y_i}\right| * 100 \tag{8}$$

A lower metric indicates better accuracy.

4. EXPERIMENTAL EVALUATION AND DISCUSSION

We conduct preliminary empirical studies on the IA GRAI dataset, which is a typical benchmark dataset for an agriculture province in Vietnam, to analyze how deep learning models perform on this time series dataset. Since the classic statistical ARIMA/ETS models [20,21,22] performed inferiorly compared to deep learning models in short/mid-term load forecasting issues, we focus on assessing the performance of RNN-based models and Transformer-based models in the experiment. It should be noted that DLinear [23] challenges the need for using Transformers in long-term time series forecasting and demonstrates through empirical studies that a simpler MLP-based model can outperform certain Transformer baselines, yielding better results. Thus, in this experiment, a simple MLP model is also taken into account for comparison.

4.1 Evaluation the accuracy

Examining forecasting results are presented in table 1, it becomes evident that the Informer model exhibits superior performance. Among the RNN models, both LSTM and GRU methods yield comparable predictive capabilities, while their variants in cooperating attentional and bidirectional mechanisms demonstrate promising outcomes.

Furthermore, the inclusion of the simpler Multilayer Perceptron (MLP) method in the analysis highlights the transformative power of the Transformer-based approach, specifically the Informer model, which achieves remarkable results with the proposed dataset. These findings challenge certain assertions made in paper [23], thereby contributing novel insights to the field of load forecasting methodologies.

Table 1. Performance metrics of deep learning models

Model	Metrics		
	MAE	MSE	MAPE
MLP	0.309	0.196	8.1
LSTM	0.209	0.078	6.0
GRU	0.221	0.081	6.7
AtLSTM	0.236	0.100	6.93
Bi-LSTM	0.164	0.053	5.04
Informer	**0.126**	**0.047**	**3.06**

The overall prediction result is illustrated in Figure 8.

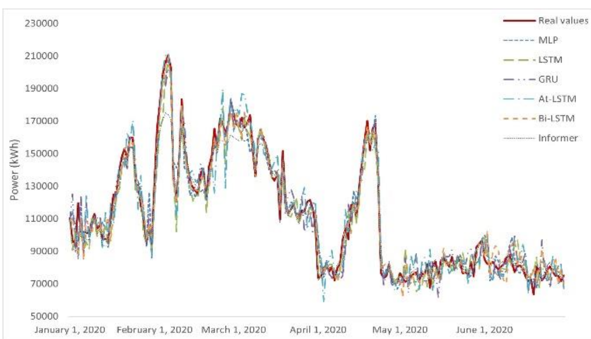
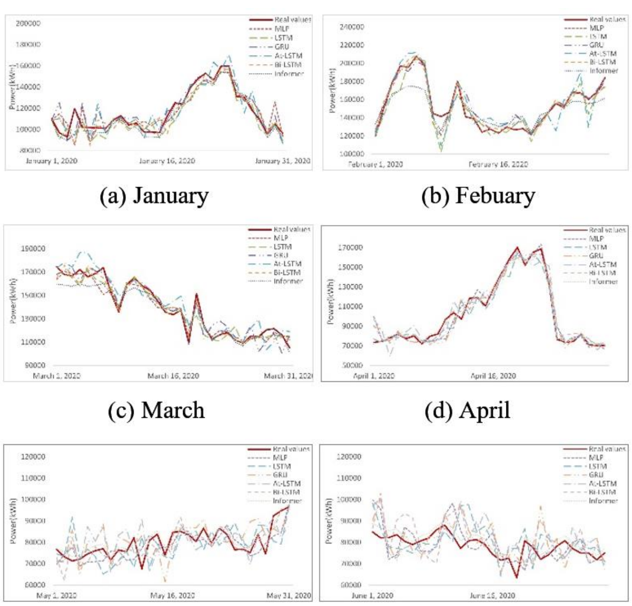


Fig. 8. Power forecasting of 6 months ahead.



(a) January (b) Febuary

(c) March (d) April

(e) May (f) June

Fig. 9. Prediction of each month.

When evaluating model accuracy, the Mean Absolute Percentage Error (MAPE) index is commonly favored over the other two indices (MSE and MAE). The reason is that MAPE provides a more comprehensive insight into the deviation of predictions from the actual values while enabling performance evaluation of the model on the same percentage scale across different data values. On the other hand, MSE and MAE can be influenced by the scale of the data, making it less effective to compare models with datasets of different scales, unlike MAPE. Thus, for a detailed comparison, we propose to use only MAPE index.

The predictions of each method for individual months (from January to June) have been thoroughly examined in Figure 9, metric's evaluation is shown in Table 2.

Table 2. Assessment the accuracy (MAPE) by month

Month	**MAPE**					
	MLP	LSTM	GRU	At-LSTM	BiLSTM	Informer
Jan	6.83	5.43	6.8	7.26	5.89	2.62
Feb	4.15	4.85	3.15	5.51	2.61	5.75
Mar	3.19	3.69	4.88	5.69	2.45	2.78
Apr	6.38	4.91	7.37	8.41	4.26	2.25
May	5.47	8.56	8.36	8.8	5.00	1.98
Jun	7.81	8.4	9.48	9.98	10.05	3.12

As seen, the Informer method yields the most accurate overall forecasts. However, it should be noted that the Informer's load prediction for February falls short of the accuracy achieved by the BiLSTM method. This discrepancy can be attributed to the distinctive traits of agricultural regions engaged in coffee cultivation, leading to substantial consumption fluctuations in February. Consequently, the Informer model encounters challenges in accurately forecasting such fluctuated consumption patterns.

Furthermore, RNN-based models such as LSTM, GRU, At-LSTM, and BiLSTM are constrained by memory and network architecture, causing their accuracy to diminish when dealing with long prediction horizons. As observed in table 2, the forecasts for May and June display inferior performance compared to those for January, February, March, and April. Additionally, when the load curves exhibit low variability, these RNN-based models struggle to capture trends effectively. Conversely, the Informer model demonstrates better support for long-range forecasts. Leveraging its self-attention mechanism, it adeptly captures the variability patterns in the load curve.

4.2 Robustness analysis

Real-time learning can significantly enhance the model's accuracy. Ensuring fast training is of paramount importance when training the model in real-time. In this context, we investigate the training time of different machine learning models. The time complexity of models is resumed in table 3. For RNN-based deep learning model, their complexity is contingent upon factors such as the number of layers, the units within each layer (h), and the length of the input (L). These parameters increase the time complexity in each iteration. Meanwhile, Transformer-based model depends mainly on the input length (L). In table 3, the complexity assumes each RNN-based model has one layer, and Attention mechanism is not taken into account in At-LSTM.

Table 3. Complexity comparison [24]

Models	Training time
MLP	$O(hL + h)$
LSTM	$O\big(4h(L + 3 + h)\big)$
GRU	$O\left(3h\left(L + \frac{8}{3} + h\right)\right)$
At-LSTM	$O\big(4h(L + 3 + h)\big)$
BiLSTM	$O\big(8h(L + 3 + h)\big)$
Informer	$O(LlogL)$

Fig. 10. Execution time of deep learning models.

The training time of models using GPU are presented in figure 10 with look-back window size is seven days for all models. As seen, the training time of the simple MLP model is the shortest, attributed to its straightforward architecture. However, its predictive accuracy falls short. On the other hand, regression neural network models demand more extensive training time, particularly Bi-LSTM, due to its two-layer architecture while other models like LSTM and GRU have a single-layer. At-LSTM model employs its unique attention mechanism to effectively filter crucial information during training, resulting in a reduction in training time. In contrast, Informer leverages uses self-attention and distilling mechanisms to constrain memory usage while maintaining the model's characteristics. Remarkably, the training time for the Informer model is notably shorter compared to other regression neural network models.

4. 3 Look-back window size analysis

Next, we continue to evaluate the effect of look-back window size (sequence length) to performance of model. The sequence length will be changed from week (7 days) to month (30 days), semester (90 days), half-year (180 days) and a year (365 days) to observe the accuracy of the models.

The results of the experiment are shown in table 4. As evident from the results, the accuracy (MAPE) of the simple MLP and regression neural network models does not improve significantly when the sequence length is increased. In contrast, the Informer method demonstrates improved accuracy with increasing sequence lengths. However, this improvement reaches a limit at a sequence length of 90; beyond this point, further increases in sequence length do not lead to additional improvements in accuracy.

Table 4. Assessment of look-back window size

		MAPE					
		MLP	LSTM	GRU	AtLSTM	BiLSTM	Informer
Seq_Len	7	8.1	6.0	6.7	6.9	5.04	3.06
	30	8.15	6.92	6.51	6.55	5.75	2.47
	90	8.45	10.3	9.76	9.67	7.67	2.35
	180	9.46	7.39	8.07	12.27	8.03	2.46
	365	7.53	10.52	7.76	8.98	NaN	2.53

5. CONCLUSION

Assessing factors influencing forecasting and comparing the feasibility of various time series forecasting methods provides a comprehensive perspective on the context of mid-term load forecasting. In the experimental phase, forecasting methods utilizing historical load data were executed. The comparison mainly revolves around different variations of RNN-based models and Informer - the most efficient architecture within the transformer-based model.

Informer model consistently produces robust forecasting results when compared to RNN-based models. This is particularly evident in its ability to achieve a remarkable Mean Absolute Percentage Error (MAPE) as low as 3% in medium-term forecasts. Additionally, the Informer model significantly reduces training time, thanks to its integrated self-attention mechanism. Nevertheless, it's important to note that during months characterized by substantial fluctuations in the load curve, the Bi-LSTM model demonstrates superior performance over the Informer model. Besides, in agricultural production areas, weather parameters play a critical role in influencing energy consumption. However, the future prediction of these parameters presents a challenge due to their inherent uncertainty. Consequently, we propose to omit them as dependent variables in this study. For future research directions, it's worth considering the incorporation of seasonal and temporal factors to further enhance the Informer model's predictive accuracy in the realm of mid-term forecasting.

ACKNOWLEDGEMENTS

This work was supported by The University of Danang, University of Science and Technology, code number of Project: T2022-02-35.

REFERENCES

[1] W. Gao, A. Darvishan, M. Toghani, M. Mo-hammadi, O. Abedinia, and N. Ghadimi, “Different states of multi-block based forecast engine for price and load prediction” International Journal of Electrical Power Energy Systems, vol. 104, pp. 423–435, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061517319865

[2] A. Al Mamun, M. Sohel, N. Mohammad, M. S. H. Sunny, D. R. Dipta, and E. Hossain, “A comprehensive review of the load forecasting techniques using single and hybrid predictive models,” IEEE Access, vol. 8, pp. 134 911–134 939, 2020.

[3] S. Panigrahi and H. Behera, “A hybrid ets–ann model for time series forecasting,” Engineering Applications of Artificial Intelligence, vol. 66, pp. 49–59, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197617301550

[4] C. Tarmanini, N. Sarma, C. Gezegin, and O. Ozgonenel, “Short term load forecasting based on arima and ann approaches,” Energy Reports, vol. 9, pp. 550–557, 2023.

[5] M. Askari and F. Keynia, “Mid-term electricity load forecasting by a new composite method based on optimal learning mlp algorithm,” IET Generation, Transmission & Distribution, vol. 14, no. 5, pp. 845–852, 2020.

[6] U. Amato, A. Antoniadis, I. De Feis, Y. Goude, and A. Lagache, “Forecasting high resolution electricity demand data with additive models including smooth and jagged components,” International Journal of Forecasting, vol. 37, no. 1, pp. 171–185, 2021.

[7] H. Wang, K. A. Alattas, A. Mohammadzadeh, M. H. Sabzalian, A. A. Aly, and A. Mosavi, “Comprehensive review of load forecasting with emphasis on intelligent computing approaches,” Energy Reports, vol. 8, pp. 13 189–13 198, 2022.

[8] G. Dudek and P. Pełka, “Pattern similarity-based machine learning methods for mid-term load forecasting: A comparative study,” Applied Soft Computing, vol. 104, p. 107223, 2021.

[9] 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.

[10] N. Mughees, S. A. Mohsin, A. Mughees, and A. Mughees, “Deep sequence to sequence bi-lstm neural networks for day-ahead peak load forecasting,” Expert Systems with Applications, vol.175, p. 114844, 2021.

[11] T. Bashir, C. Haoyong, M. F. Tahir, and Z. Liqiang, “Short term electricity load forecasting using hybrid prophet-lstm model optimized by bpnn,” Energy Reports, vol. 8, pp. 1678–1686, 2022.

[12] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X.Wang, and X. Yan, “Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting,” Advances in neural information processing systems, vol. 32, 2019.

[13] B. Li, W. Cui, L. Zhang, C. Zhu, W. Wang, I. Tsang, and J. T. Zhou, “Difformer: Multi-resolutional differencing transformer with dynamic ranging for time series analysis,” IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 1–12, 2023.

[14] B. N. Oreshkin, G. Dudek, P. Pełka, and E. Turkina, “N-beats neural network for mid-term electricity load forecasting,” Applied Energy, vol. 293, p. 116918, 2021.

[15] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, “Informer: Beyond efficient transformer for long sequence time-series forecasting,” 2021.

[16] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, “Transformers in time series: A survey,” arXiv preprint arXiv:2202.07125, 2022.

[17] G. Zhang, X. Bai, and Y. Wang, “Short-time multi-energy load forecasting method based on cnn-seq2seq model with attention mechanism,” Machine Learning with Applications, vol. 5, p. 100064, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666827021000323

[18] X. Li, Z. Zhou, L. Chen, and L. Gao, “Residual attention-based lstm for video captioning,” World Wide Web, vol. 22, pp. 621–636, 2019.

[19] A. G. Salman, Y. Heryadi, E. Abdurahman, and W. Suparta, “Single layer & multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting,” Procedia Computer Science, vol. 135, pp. 89–98, 2018.

[20] Phuangpornpitak, N. and W. Prommee. “A Study of Load Demand Forecasting Models in Electric Power System Operation and Planning,” GMSARN International Journal, vol.10, pp. 19-24, 2016.

[21] D. D. Le. “An Enhancement to Time Series Model for Short-Term Forecasting of Multi-Regional Electricity Load,” GMSARN International Journal, vol.13, pp. 202 – 207, 2019.

[22] Q. H. Giap, D. L. Nguyen, T. T. Q. Nguyen, và T. M. D. Tran, “Applying Neural Network And Levenberg - Marquardt Algorithm for Load Forecasting in IA-Grai District, Gia Lai Province”, UD-JST, vol.20, pp. 13–18, 2022.

[23] A. Zeng, M. Chen, L. Zhang, and Q. Xu, “Are transformers effective for time series forecasting?” in Proceedings of the AAAI conference on artificial intelligence, vol. 37, no. 9, 2023, pp.11 121–11 128.

[24] S. Zhang, Y. Wu, T. Che, Z. Lin, R. Memisevic, R. R. Salakhutdinov, and Y. Bengio, “Architectural complexity measures of recurrent neural networks,” Advances in neural information processing systems, vol. 29, 2016.