IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 July 2024, accepted 2 August 2024, date of publication 5 August 2024, date of current version 16 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3438945

RESEARCH ARTICLE

Net Metering Prediction in Prosumer Building With Temporal Fusion Transformer Models

SORAWUT JITTANON, (Graduate Student Member, IEEE), YODTHONG MENSIN, NIPON KETJOY, AND CHAKKRIT TERMRITTHIKUN, (Member, IEEE)
School of Renewable Energy and Smart Grid Technology (SGtech), Naresuan University, Phitsanulok 65000, Thailand

Corresponding author: Chakkrit Termritthikun (chakkritt@nu.ac.th)

This work was supported in part by the Global and Frontier Research University Fund, Naresuan University, under Grant R2567C002; and in part by the Division of Research and Innovation, Naresuan University, under Grant R2566C011.

ABSTRACT The growing global demand for electrical energy, together with the significant problem of increasing carbon dioxide emissions, have become urgent issues. The inefficient use of electricity exacerbates these challenges. Smart grid technology is emerging as a solution, employing innovative approaches to solve these problems. Within this context, this research emphasizes the identification of the most effective forecasting model for demand prediction. Utilizing the Transformer-based model, Temporal Fusion Transformer (TFT), together with the Naresuan University, School of Renewable Energy and Smart Grid Technology (SGtech) net metering dataset, we explored the influence of additional features on forecasting models, categorizing them into net metering data, weather-related attributes including temperature, dew point, weather conditions, and wind direction, and supplementary features related to the operational behavior of SGtech, specifically workday and time-of-day. Our experimentation shows that integrating workday and time-of-day data alongside net metering data significantly enhances prediction precision compared to other combinations. The TFT model outperforms popular time series forecasting models, including Neural Basis Expansion Analysis for Time Series (N-BEATS) and Neural Hierarchical Interpolation for Time Series (N-HiTS), in accuracy and parameter efficiency while maintaining inference times.

INDEX TERMS Deep learning, sustainable development goals, transformer model, prosumer building, energy forecasting.

I. INTRODUCTION

Smart grid concept signifies a transformative evolution in the domain of electrical infrastructure, marking a new era of intelligent and interconnected energy systems. A smart grid is a sophisticated network integrating cutting-edge digital technologies, communication protocols, and real-time data analytics into the traditional electricity grid. The objective of the smart grid concept is to revolutionize electricity generation, distribution, and consumption by introducing two-way communication between utilities and end-users. By deploying smart meters, sensors, and automation, smart grids empower stakeholders with unprecedented insights and control over the grid's dynamic behavior. As the cornerstone of a modernized energy landscape, smart grids play a crucial role in addressing efficiency, reliability, and sustainability challenges, while accommodating the growing integration of renewable energy sources and enabling novel demand response strategies. In this context, understanding the intricacies of smart grid functionalities becomes crucial for unlocking the full potential of advanced energy management systems.

The integration of smart grid technology aligns closely with the United Nations' Sustainable Development Goals (SDGs), particularly Goal 7 (Affordable and Clean Energy) and Goal 13 (Climate Action). By enhancing energy efficiency, reducing carbon emissions, and promoting the use of clean and renewable energy sources, smart grids contribute significantly to achieving these global sustainability objectives. Within the context of sustainable energy and climate

The associate editor coordinating the review of this manuscript and approving it for publication was Wanqing Zhao.

action, the role of accurate net metering forecasting becomes even more crucial.

Net metering, a billing arrangement that allows consumers who generate their electricity, typically from renewable energy sources, to offset their consumption by feeding excess electricity back into the grid [1], is pivotal in the dynamic landscape shaped by demand response [2] and energy management system [3]. Given that electricity usage inherently follows a temporal pattern characterized by fluctuations over time, time series forecasting emerges as a crucial aspect of this endeavor. Accurate predictions of electricity demand are essential for utilities, grid operators, and energy stakeholders to efficiently plan and manage resources. The evolution of forecasting methods has witnessed a remarkable transition from traditional statistical approaches to the more sophisticated realms of time series forecasting, machine learning (ML), and deep learning [4], [5].

Statistical methods such as time series analysis and regression models have been commonly employed for energy forecasting. These models relied on historical patterns and factors such as seasonality and trends to make predictions. For instance, the Autoregressive Integrated Moving Average (ARIMA) models were widely used for their ability to capture temporal dependencies in energy consumption data [6].

With the advent of ML, predictive modeling for electricity consumption has become more nuanced and accurate [7], [8], [9]. ML algorithms, including decision trees, support vector machines, and random forests, have gained popularity [10], [11]. For example, a Random Forest model, trained on historical data, includes various features such as weather conditions, day of the week, and special events to predict future electricity consumption [12].

In recent years, deep learning techniques [13], [14], [15], particularly recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), have shown remarkable success in capturing complex patterns and dependencies in time series data [16]. An example includes using an LSTM-based model that can effectively capture long-term dependencies in energy consumption data [17], [18], making it well-suited for accurate forecasting, similar to the widely recognized N-BEATS [19] and N-HiTS [20].

Deep learning has found applications across various domains, and one of the most famous models, the Transformer, originally introduced for Natural Language Processing (NLP) tasks, has gained recognition due to its attention mechanisms. In the field of time series forecasting, the Temporal Fusion Transformer (TFT) emerges as another Transformer-based model harnessing attention mechanisms along with other state-of-the-art techniques in the field. Given proficiency in handling time series data, TFT is well-suited for tasks involving data such as net metering, which serves as the focus of this research.

In our research, the primary dataset comprised the data from the SGtech net metering dataset, which served as the cornerstone of our forecasting model. In addition to this fundamental data, we incorporated weather-related variables, including temperature, dew point, weather conditions, and wind direction, as interrelated features. We also incorporated the workday and time-of-day operational factors, specific to SGtech, as supplementary features in our multi-input feature training process. The purpose of incorporating these features was to enhance the overall accuracy of our forecasting model.

The contributions of this research can be summarized as follows:

- Our research results highlight the operational behavior-related features within SGtech as having a greater impact on enhancing model performance than the utilization of weather-related features.
- We successfully identified the optimal combination of correlation features for the TFT model, specifically SGtech's workday and time-of-day, contributing to its effectiveness in time series forecasting.
- The forecasting model developed in this work demonstrates greater accuracy and parameter efficiency than that demonstrated in established models like N-BEATS and N-HiTS, while maintaining efficient inference times.

II. RELATED WORK

In this section, we focus on methods related to our Transformer-based model, which has gained popularity across various fields of deep learning due to its capabilities.

A. TRANSFORMER MODEL

In 2017, Google's researchers published the Transformer model in their research paper titled "Attention is All You Need" [21]. The innovative architecture of the Transformer model has become mainstream in NLP and ML, reshaping the landscape of how sequential data is processed and modeled. The transformative impact of the Transformer model is evident in its unique self-attention mechanism, a feature that allows for the simultaneous consideration of contextual information across the entire input sequence. This capability enhances the network's effectiveness in learning complex relationships within the data, setting it apart from traditional sequence-to-sequence models. Beyond its initial introduction, the Transformer model continues to influence subsequent research, showcasing its enduring significance and ongoing evolution. With its remarkable ability to handle parallelization and capture long-range dependencies, the Transformer Neural Network remains at the forefront of advancements in NLP and ML, cementing its status as a pioneering framework.

The Attention Mechanism, the pivotal innovation at the core of the Transformer architecture, fundamentally transforms the way models weigh different parts of the input sequence during predictions. This mechanism represents a departure from traditional sequential processing, enabling the model to dynamically assign varying degrees of importance to different positions in the input sequence simultaneously. By doing so, the Transformer model can adeptly capture contextual information from all elements in the sequence,

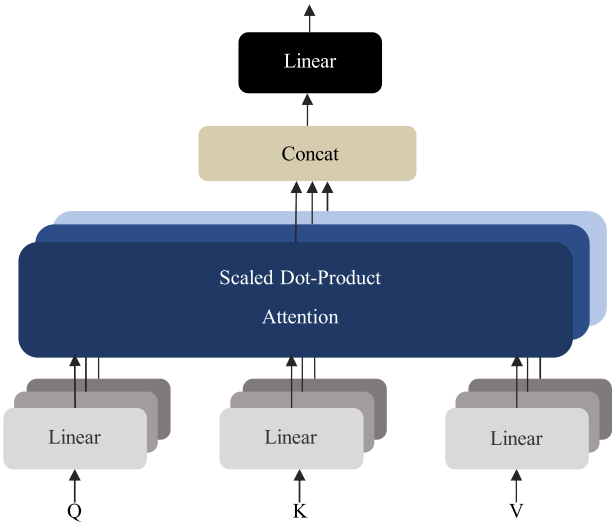


FIGURE 1. Attention layer in Transformer model [21].

offering a more nuanced understanding of relationships and dependencies within the data. The introduction of the attention mechanism stands as a paradigm shift in NLP, enhancing the model's capacity to discern and leverage intricate patterns in sequential information. The mathematical formula of the Attention mechanism is shown in (1).

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

where K is the Key, Q is Query, V is Value and $\sqrt{d_k}$ is dimension of metrix K

The Transformer architecture is structured around an encoder-decoder framework. The encoder processes the input sequence, and the decoder generates the corresponding output sequence. The key components include multi-head self-attention, where attention is applied simultaneously across different positions using multiple attention heads. To address the challenge of sequence order, positional encoding is introduced, providing the model with information about the position of elements in the sequence. Each encoder and decoder block incorporates position-wise feedforward networks, injecting non-linearity into the model. The use of layer normalization and residual connections further enhances stability during training. This combination of components results in a highly parallelizable architecture that excels in capturing dependencies across long distances in the input sequence.

One of the distinctive features of the Transformer model is the multi-head self-attention mechanism. By performing self-attention in parallel through multiple attention heads, as shown in Fig. 1, the model can attend to different parts of the input sequence with varying weights. The attention scores are computed using a scaled dot-product operation, measuring the similarity between elements and allowing the model to focus more on relevant information. This not only improves the efficiency of the attention mechanism but also

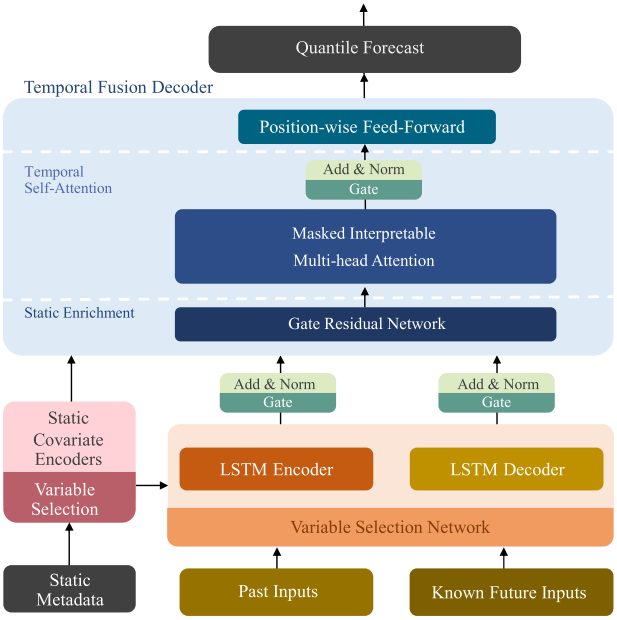


FIGURE 2. Temporal fusion transformer architecture.

enables the Transformer model to excel in tasks requiring the understanding of intricate relationships within the input data.

Positional encoding is introduced to address the challenge of maintaining the sequence order. This additional information helps the model distinguish between the positions of elements in the sequence, enabling the Transformer model to process sequential data effectively. Beyond its initial success in the NLP tasks, similar machines have demonstrated remarkable versatility and successfully applied to various domains, including image generation, speech recognition, and more. The architecture's adaptability and effectiveness have spurred ongoing research to optimize further and extend its capabilities for an even wider range of applications [22], [23], [24] including time series forecasting [25], [26], resulting in several models based on the original Transformer model being developed, including the Temporal Fusion Transformer Model (TFT), among others.

B. TEMPORAL FUSION TRANSFORMER MODEL (TFT)

The TFT is a deep learning model based on the original Transformer model and designed for time series forecasting [27]. Combining elements of both transformers and recurrent neural networks (RNNs), the TFT aims to efficiently capture long-term dependencies and temporal patterns in sequential data. While Transformer-based models have gained popularity for their parallelization capabilities, they may struggle with sequential data due to their lack of inherent temporal understanding. TFT addresses this limitation by introducing a novel architecture incorporating positional embeddings and attention mechanisms, allowing it to model sequential dependencies effectively. The TFT architecture is shown in Fig. 2.

The TFT model introduces an innovative gating mechanism, specifically the gate residual network (GRN), to refine

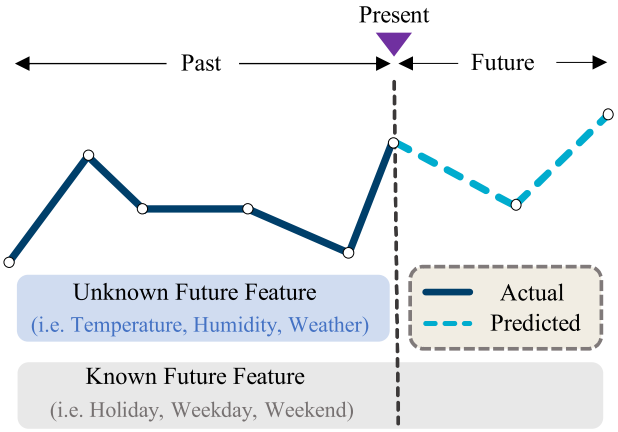


FIGURE 3. Data handling overview in TFT model.

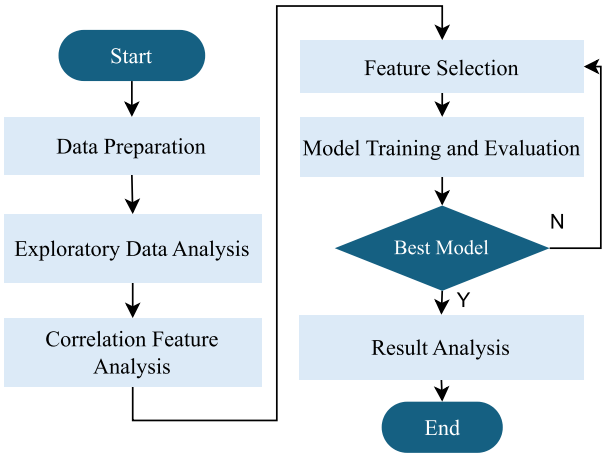


FIGURE 4. Flowchart of forecasting method.

its processing capabilities further. The GRN is crucial in handling the diverse complexities inherent in time series data. It operates by applying linear transformations to the input data, followed by a sigmoid gating layer that determines the relevance of each transformed feature. This gating mechanism adeptly filters and prioritizes information, allowing only the most pertinent features to influence the model's predictions. Also, the incorporation of residual connections within the GRN facilitates a smoother flow of gradients during backpropagation, thereby enhancing the model's ability to learn from both short and long-term temporal dependencies. This approach not only improves the model's accuracy in forecasting but also introduces a level of adaptability, enabling TFT to maintain performance across various types of sequential data with differing complexity levels. The mathematical formula of GRN is shown in (2), (3), and (4).

$$GRN_{\omega}(a, c) = LayerNorm(a + GLU_{\omega}(\eta_1)) \quad (2)$$

$$\eta_1 = W_{1,\omega}\eta_2 + b_{1,\omega} \quad (3)$$

$$\eta_2 = ELU\left(W_{2,\omega}a + W_{3,\omega}c + b_{2,\omega}\right) \quad (4)$$

where a is a primary input, c is an optional context, W is a weight metric, *LayerNorm* is standard layer normalization, *GLU* is the Gated Linear Units, η_1 and η_2 are intermediate layers, ω is an index to denote weight sharing, and *ELU* is the Exponential Linear Unit activation function.

The TFT model employs a variable selection process alongside the gating mechanism through a Variable Selection Network (VSN). These networks are instrumental in pinpointing the most relevant variables for each forecasting step. By assigning importance weights to inputs, VSN effectively filters out noise, allowing TFT to concentrate on data that is most predictive of future trends. This mechanism bolsters the model's forecasting precision and enhances interpretability by clearly identifying key influential variables in the time series data.

One important feature of TFT is its ability to handle irregularly sampled time series data, as shown in Fig. 3. It is versatile for real-world applications where observations may not occur at fixed intervals. This adaptability is crucial for forecasting tasks in various domains, such as finance, healthcare, and energy, where time series data can exhibit complex patterns and irregularities. Additionally, TFT utilizes a combination of global and local attention mechanisms to capture overall trends and fine-grained temporal patterns within the data.

The TFT model has succeeded in accurately predicting future values in time series datasets, outperforming traditional methods and showcasing its potential for applications requiring precise forecasting. Overall, the Temporal Fusion Transformer model represents a significant advancement in the field of time series forecasting [28], [29], [30], offering a powerful and flexible architecture for capturing temporal dependencies in sequential data.

III. METHODOLOGY

This section explains the research process for forecasting net metering within each building in an SGtech network. The research process comprises several steps as shown in Fig. 4, starting with preparing the net metering data from the SGtech building, considering the relationship of input features, and data grouping for model training. It also discusses methods for measuring the model's performance.

A. DATASET

The dataset utilized in this section belongs to the historical net metering of SGtech's prosumer building. This building can generate electricity from photovoltaic (PV) sources for self-consumption before resorting to grid electricity imports. The dataset comprises timestamps and corresponding net metering measurements in kilowatt-hours, encompassing the entire year from January 1, 2022, to December 31, 2022. The net metering profile is shown in Fig. 5.

B. CORRELATION FEATURE

In addition to using historical net metering data for training the model, the inclusion of other features related to net metering is another feature that can affect the accuracy

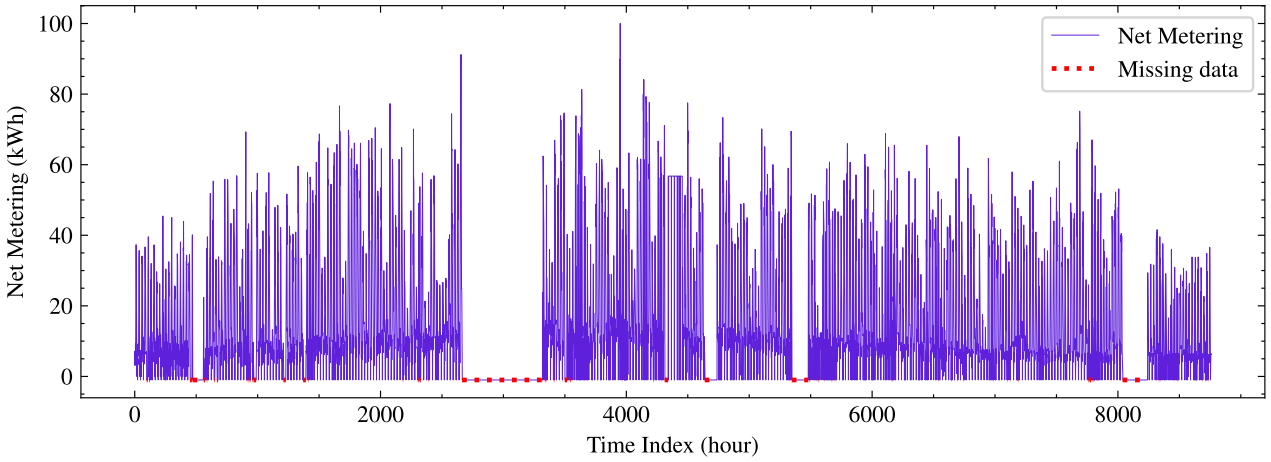


FIGURE 5. Net metering profile of SGtech prosumer building from January 1, 2022 to December 31, 2022, including 365 days (8,760 hours).

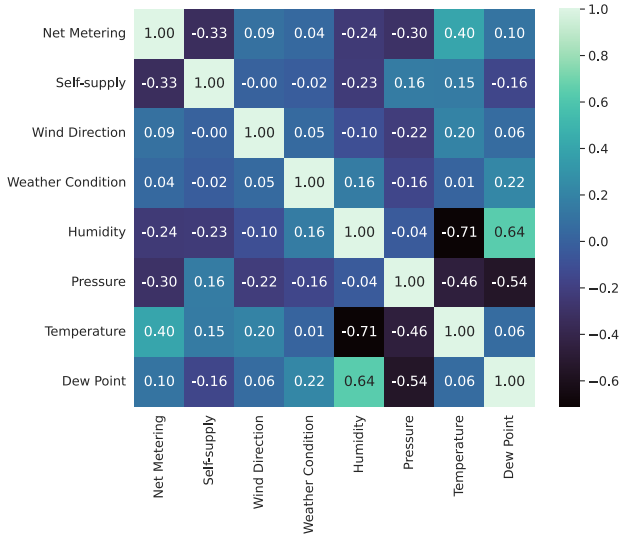


FIGURE 6. Correlation heatmap of additional feature.

of a forecasting model. In this section, we collected and utilized various additional features expected to influence the prediction. After gathering this data, we analyzed it to determine the correlation of each type of additional feature with the net metering.

To calculate the correlation coefficient of each input feature, the Pearson correlation method was used to find the coefficients of all the data. The correlation value from the range of -1 to 1, where a value close to 1 indicates a strong positive correlation between the data, a value close to 0 suggests no correlation, and a value close to -1 signifies a strong negative correlation between the data. The mathematical formula of Pearson Correlation is shown in (5).

$$\rho\left(X, Y\right) = \frac{cov\left(X, Y\right)}{\sigma_X \sigma_Y} \tag{5}$$

where $\rho\left(X, Y\right)$ is the Pearson Correlation, $cov\left(X, Y\right)$ is Covariant between X and Y, σ_X is standard deviation of X, and σ_Y is standard deviation of Y

The correlation heatmap illustrated in Fig. 6, shows the relationships between various input features, including supply, wind direction, weather conditions, humidity, pressure, temperature, and dew point, as we search their correlation with net metering. Our objective was to identify features that strongly correlate with net metering, which will be considered additional features for creating a forecasting model. Notably, the results showed that wind direction, weather conditions, temperature, and dew point all exhibited a positive correlation with net metering. Consequently, we incorporated all these features into our forecasting model.

The weather condition feature that is used in this research includes a range of terms commonly employed in meteorology to describe different weather conditions. "Fair" signifies clear and pleasant weather conditions. "Shallow Fog" indicates the presence of a thin layer of fog near the ground. "Partial Fog" implies fog that covers only part of the area. "Mist" represents light fog with good visibility. "Mostly Cloudy" describes a sky with a majority of cloud cover. "Partly Cloudy" indicates a sky with some cloud cover but also some clear areas. "Light Rain" refers to gentle and scattered rain showers. "Light Drizzle" represents very fine and light raindrops. "Cloudy" signifies an overcast sky with significant cloud cover. "Rain" represents steady and continuous rainfall. "Fog" indicates thick and dense fog reducing visibility. "T-Storm" stands for a thunderstorm with lightning, thunder, and heavy rain. "Haze" represents reduced visibility due to fine dust or pollution particles in the air. "Showers in the Vicinity" indicates rain showers occurring nearby without directly affecting the location. "Thunder" signifies the presence of thunder without rain. The wind direction symbols and descriptions used in this research are shown in Table 1.

Lastly, we integrated the two features of SGtech, workday and time-of-day features, into the model. These features provided valuable insights into SGtech's operational dynamics. Workday denotes the days when SGtech was actively working as shown in Fig. 7, enabling us to identify patterns

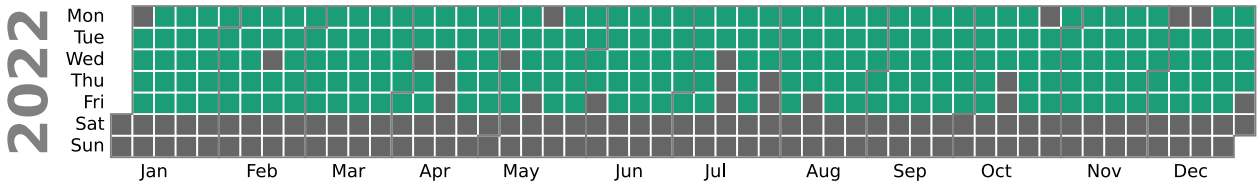


FIGURE 7. Illustrating SGtech's workday (green) and non-workday (gray) data from January 1, 2022, to December 31, 2022. The dataset starts on a Saturday, resulting in initial unfilled cells for days before January 1, 2022. Similarly, the dataset concludes on a Saturday, leaving cells beyond December 31, 2022, unfilled.

TABLE 1. Wind direction symbol explanation.

Symbol	Wind directions	Degrees/Description
N	North	0 to 23° and 337 to 360°
NNE	North-Northeast	24° to 33°
NE	Northeast	34° to 68°
ENE	East-Northeast	69° to 78°
E	East	79° to 113°
ESE	East-Southeast	114° to 123°
SE	Southeast	124° to 158°
SSE	South-Southeast	159° to 168°
S	South	169° to 203°
SSW	South-Southwest	204° to 213°
SW	Southwest	214° to 248°
WSW	West-Southwest	249° to 258°
W	West	259° to 293°
WNW	West-Northwest	294° to 303°
NW	Northwest	304° to 336°
NNW	North-Northwest	337° to 346°
VAR	Variable	Wind speed <5 m/s
CALM	Calm	Direction varies frequently

in net metering on these days, which may differ from non-working days. Meanwhile, the time-of-day feature informed us about the specific hours, which were divided into before and after midday, when SGtech officers were engaged in their tasks. This temporal information helped us capture variations in net metering that aligned with SGtech's working hours, enhancing our model's ability to predict and adapt to net metering patterns influenced by SGtech's operations.

C. EXPLORATORY DATA ANALYSIS

This section describes our use of Exploratory Data Analysis (EDA) as a crucial step in examining net metering data. EDA helped us discover important patterns, detect anomaly data, and discover our dataset's details. Through statistical and visual methods, we began our exploration of the net metering data to gain a fundamental understanding that would guide our subsequent analyzes.

Initially, we examined the correlation between our selected features and net metering data, including temperature, dew point, weather conditions, and wind direction, and identified any anomalies. To further understand the data distribution, we employed an hourly count of occurrences for each event, as shown in Fig. 8, which allowed us to gain insights into their patterns and relationships.

D. MODEL EVALUATION

Evaluating the efficiency of a forecasting model depends on the selected measurement method. This evaluation quantifies the difference between forecasted and actual net metering values, where smaller errors signify powerful model performance. We utilized several loss functions during both the training and evaluation phases to optimize the efficiency of the forecasting model, as discussed in upcoming sections.

Quantile loss is a statistical method used in regression analysis to evaluate a predictive model's accuracy, especially in the TFT model when estimating specific percentiles (quantiles) of a target variable's distribution, as shown in (6). Instead of focusing on minimizing the mean error, quantile loss measures how accurately a model predicts values at different quantiles of the distribution, giving insights into the dispersion of the data. This approach is particularly valuable in scenarios where accurately capturing prediction uncertainty and extreme values is essential, as it provides a more comprehensive assessment of a model's performance across the entire distribution of the target variable.

$$\mathcal{L}_{Quantile}(u_i, \bar{u}_i) = \begin{cases} \alpha \times (u_i - \bar{u}_i), & \bar{u}_i \leq u_i \\ (1-\alpha) \times (u_i - \bar{u}_i), & \bar{u}_i > u_i \end{cases} \tag{6}$$

where α is the quantile level, u_i is the actual value, and $\bar{u}_i$ is the predicted value.

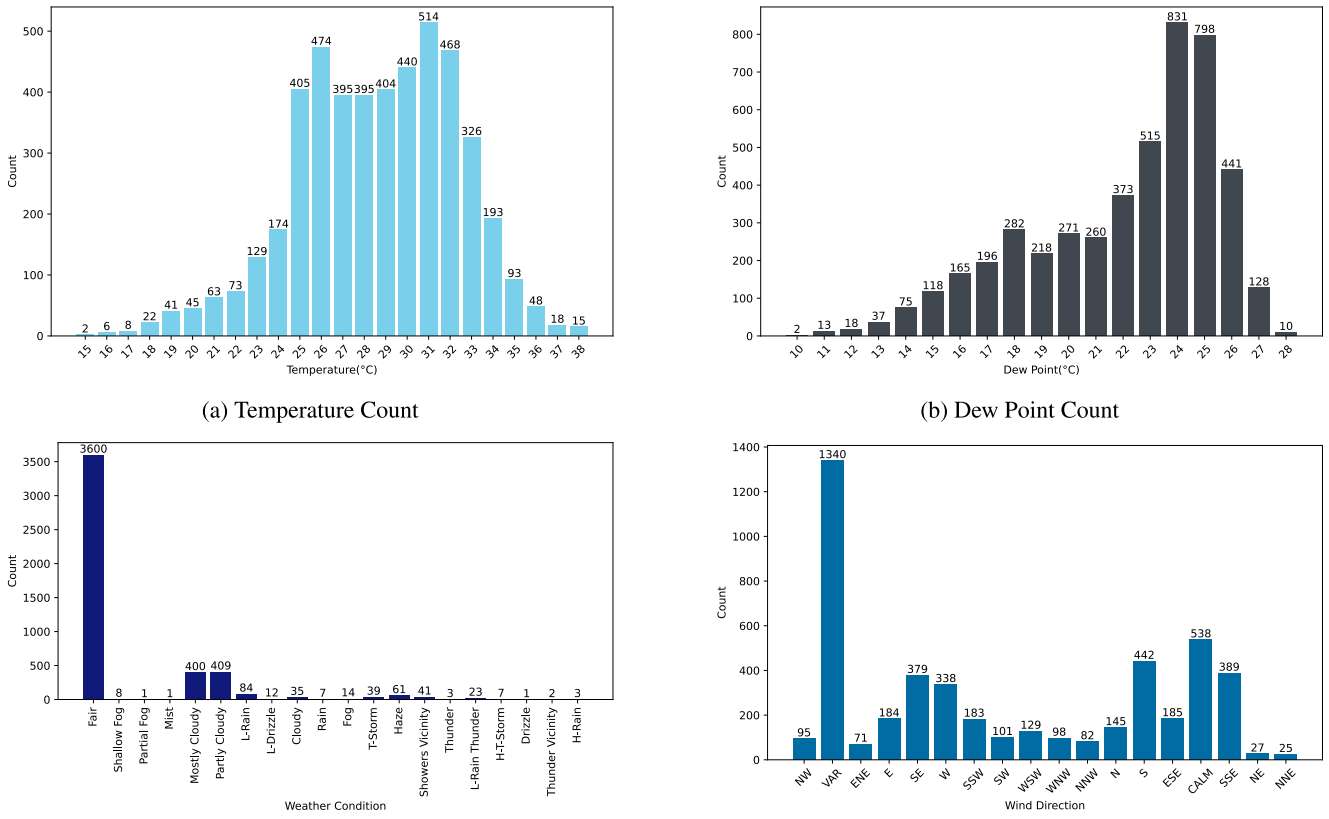
The Mean Absolute Error (MAE) measures the average error magnitude between predicted and actual values. It calculates the absolute difference between each prediction and the corresponding actual value and then averages these absolute differences. MAE provides a straightforward understanding of how far, on average, the model's predictions are from the actual values without considering the direction of the errors, as shown in (7).

$$MAE\ (u_i, \bar{u}_i) = \frac{1}{n} \sum_{i=1}^{n} |u_i - \bar{u}_i| \tag{7}$$

where u_i is the actual value and $\bar{u}_i$ is the predicted value.

The Mean Square Error (MSE) is another metric for evaluating the performance of a forecasting model. It calculates the average of the squared differences between predicted and actual values. Squaring the errors gives more weight to larger errors, which can be useful when penalizing larger deviations from the actual values. However, it tends to amplify the impact of outliers, as shown in (8).

$$MSE\ (u_i, \bar{u}_i) = \frac{1}{n} \sum_{i=1}^{n} (u_i - \bar{u}_i)^2 \tag{8}$$



(a) Temperature Count

(b) Dew Point Count

(c) Weather Condition Count

(d) Wind Direction

FIGURE 8. **Hourly count data of Temperature, Dew Point, Weather Condition, and Wind Direction.**

where u_i is the actual value and $\bar{u}_i$ is the predicted value.

The Root Mean Square Error (RMSE) is derived from MSE but is expressed in the same units as the original data. It is obtained by taking the square root of the MSE. RMSE provides a more interpretable measure of the model's error compared to MSE. Like MSE, RMSE also penalizes larger errors but presents them in a format that is easier to relate to the original data scale, as shown in (9).

$$RMSE\ (u_i, \bar{u}_i) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(u_i - \bar{u}_i)^2} \tag{9}$$

where p_i is the actual value and $\bar{p}_i$ is the predicted value.

The Mean Absolute Scaled Error (MASE) is a relative error metric that scales the MAE by the MAE in a sample of a naive forecasting method. This makes the MASE scale independent and easier to interpret across different datasets. A MASE value less than one indicates better performance than the naive method, while a value greater than one indicates worse performance, as shown in (10).

$$MASE\ (u_i, \bar{u}_i) = \frac{\frac{1}{n}\sum_{i=1}^{n}|u_i - \bar{u}_i|}{\frac{1}{n-1}\sum_{j=2}^{n}|u_j - u_{j-1}|} \tag{10}$$

where u_i is the actual value and $\bar{u}_i$ is the predicted value.

The Overall Percentage Error (OPE) measures the total error relative to the actual values, expressed as a percentage. This metric is useful for understanding the overall accuracy of the model in terms of percentage error, making it more interpretable for practical applications, as shown in (11).

$$OPE\ (u_i, \bar{u}_i) = \frac{\sum_{i=1}^{n}|u_i - \bar{u}_i|}{\sum_{i=1}^{n}|u_i|} \times 100 \tag{11}$$

where u_i is the actual value and $\bar{u}_i$ is the predicted value.

These five metrics are commonly used to assess the accuracy and performance of forecasting models, with lower values indicating better predictive capabilities. Each metric has its strengths and weaknesses, so choosing the most suitable metric depends on the specific characteristics and objectives of the forecasting task.

IV. RESULT

The results of this research can be divided into several phases. Initially, we focused on fine-tuning the hyperparameters to prepare for all experiments. We then fine-tuned the TFT model by incorporating correlation features to identify the most effective input features for building an optimized forecasting model. Finally, we conducted a comprehensive comparison with benchmark models to evaluate the overall performance of our proposed approach.

TABLE 2. All combinations of input feature sets used in the experiment.

No. of Correlation Feature	Net Metering	Temperature	Dew Point	Weather Condition	Wind Direction	Workday	Time-of-Day
1 Feature	✓	✓					
	✓		✓				
	✓			✓			
	✓				✓		
	✓					✓	
	✓						✓
2 Features	✓	✓	✓				
	✓	✓		✓			
	✓	✓			✓		
	✓	✓				✓	
	✓	✓					✓
	✓		✓	✓			
	✓		✓		✓		
	✓		✓			✓	
	✓		✓				✓
	✓			✓	✓		
	✓			✓		✓	
	✓			✓			✓
	✓				✓	✓	
	✓				✓		✓
	✓					✓	✓

TABLE 3. Hyperparameter tuning result of TFT model.

Hyperparameter			Result		
Hidden Layer	Learning Rate	Attention Head	MAE (kWh)	MSE (kWh^2)	RMSE (kWh)
8	0.01	1	13.45±0.51	280.66±10.91	16.73±0.28
16	0.01	1	7.76±0.24	105.51±9.87	9.66±0.30
32	0.01	1	7.50±0.28	98.55±8.45	9.46±0.15
32	0.005	1	8.16±0.14	137.86±12.29	10.01±0.12
32	**0.05**	**1**	**6.57±0.24**	**77.52±6.67**	**8.89±0.13**
32	0.05	2	7.46±0.27	95.02±6.97	9.44±0.15

TABLE 4. Forecasting result using net metering with one additional feature.

Additional Feature	MAE (kWh)	MSE (kWh^2)	RMSE (kWh)
Temperature	**11.29±0.40**	**277.24±7.84**	**16.79±0.27**
Dew Point	12.44±0.54	281.67±43.06	17.21±0.31
Weather Condition	18.39±0.65	459.39±24.24	21.20±0.42
Wind Direction	16.07±0.40	391.25±14.50	20.61±1.31
Workday	12.48±0.53	236.80±33.60	14.88±0.58
Time-of-Day	13.20±0.57	304.70±23.60	17.27±0.46

A. HYPERPARAMETER TUNING

We determined the hyperparameters for the TFT model by adjusting the three key hyperparameters; the number of hidden layers, the learning rate, and the number of attention heads. Through iterative experimentation, we identified a set of hyperparameters that performed well with the lowest error in terms of MAE, MSE, and RMSE. We conducted five rounds of experiments for each hyperparameter set to calculate the numerical average and standard deviation. Additionally, we created visual representations of the predictions with the least error in each experimental set. Among the various combinations tested, it became clear that the most optimal set of hyperparameters consisted of 32 hidden layers, a learning rate of 0.05, and 2 attention heads, as it resulted in the lowest prediction error. The results of hyperparameter tuning are shown in Table 3.

B. CORRELATION FEATURE RESULT

In this experiment, we considered various features related to net metering needs through a selection process based on correlation analysis. The features examined in the experiments included temperature, dew point temperature, weather conditions, wind direction, workday, and time-of-day. We conducted five rounds of experiments for each set of relevant features to calculate the numerical average and standard deviation of error values, as shown in Table 2. Furthermore, we generated visual representations of predictions with the least error for each experimental set. This comprehensive approach allowed us to explore these features' influence on net metering accurately. Additionally, the experiment was separated into two parts: testing one correlation feature and testing two correlation features.

The results for applying net metering in conjunction with an additional feature set, encompassing temperature, dew point, weather condition, wind direction, workday, and time-of-day, evaluated in terms of loss matrix, are displayed in Table 4. The findings show that by including the temperature feature as an auxiliary input during model training, forecasting accuracy was enhanced across all five error metrics; MAE, MSE, RMSE, MASE, and OPE.

In a further experiment, we utilized two correlated features alongside net metering as inputs for the forecasting model. Table 5 visually outlines the 15 combinations of input feature sets. Among these combinations, it was clear that incorporating both workday and time-of-day as input features, alongside net metering, consistently yielded the lowest error rates in terms of accuracy across all five loss metrics.

C. WORKDAY AND TIME-OF-DAY WITH OTHER FEATURE

We compared the combination of the workday and time-of-day features; the two-feature combination, which had previously demonstrated superior performance, with workday and time-of-day features combined variously with the additional

TABLE 5. Forecasting result using net metering with two additional features.

Additional Feature #1	Additional Feature #2	MAE (kWh)	MSE (kWh^2)	RMSE (kWh)
Temperature	Dew Point	10.16±0.18	236.26±13.36	15.37±0.44
Temperature	Weather Condition	10.78±0.43	227.08±8.91	15.07±0.30
Temperature	Wind Direction	11.20±0.42	241.61±7.46	15.54±0.24
Temperature	Workday	6.77±0.54	88.70±12.44	9.40±0.65
Temperature	Time-of-Day	11.97±0.33	229.26±11.86	15.14±0.39
Dew Point	Weather Condition	9.82±0.60	199.04±12.17	14.10±0.43
Dew Point	Wind Direction	10.93±0.57	240.19±12.55	15.49±0.40
Dew Point	Workday	8.09±28.56	118.28±11.76	10.86±0.55
Dew Point	Time-of-Day	10.65±0.37	178.80±12.91	13.36±0.48
Weather Condition	Wind Direction	9.92±0.38	194.81±12.69	13.95±0.46
Weather Condition	Workday	10.72±0.40	238.52±15.13	15.44±0.49
Weather Condition	Time-of-Day	11.08±0.41	232.08±13.53	15.23±0.44
Wind Direction	Workday	10.11±0.34	217.63±13.19	13.24±0.38
Wind Direction	Time-of-Day	11.05±0.45	225.31±14.25	14.61±0.45
Workday	Time-of-Day	**5.76±0.29**	**68.29±8.06**	**9.25±0.48**

features of temperature, dew point, weather condition, and wind direction; as three-feature combinations. The aim was to determine whether the two-feature combination outperformed the three-feature combinations. We found that the two-feature combination outperformed all of the three-feature combinations.

Table 6 shows the performance of the two-feature combination of workday and time-of-day combination compared to the three-feature combinations. Surprisingly, the least favorable result was observed when adding the temperature feature to the workday and time-of-day combination. This observation is intriguing, espec bially considering the strong correlation between the temperature feature and net metering. This result may come from the unique characteristics of the building used for historical net metering data. This building operates as a prosumer, with a PV system operating for several years, which can lead to contrasting effects when solar radiation increases. While higher radiation results in more energy generation from the PV system, it simultaneously increases the air-conditioning load due to rising temperatures. Adding temperature as a factor resulted in a greater forecast error, which did not align with the expected outcomes.

D. COMPARE WITH BENCHMARK

To ensure the effectiveness of our developed model in forecasting net metering, we compared well-established deep learning models commonly employed in time series forecasting tasks. The benchmark models used in this comparison were statistical methods, *i.e.*, ARIMA, Exponential Smoothing, regression methods, *i.e.*, Support Vector Regression (SVR), Random forest, and deep learning models, *i.e.*, LSTM, N-BEATS, N-HiTS, and TFT. The visualized results can be observed in Fig. 9 and Fig. 11, while the performance metric results in terms of MAE, MSE, RMSE, MASE, and OPE are presented in Table 7.

Experiments were conducted with prediction lengths of 24, 48, 72, 120, and 168 hours. Table 7 shows that the TFT model outperforms all other methods in all performance metrics for the 24- and 48-hour prediction lengths. For the 72-hour prediction length, the TFT model outperforms deep learning

TABLE 6. Compare result of workday and time-of-day with additional features.

Input Feature	MAE (kWh)	MSE (kWh^2)	RMSE (kWh)
Workday + Time-of-Day	**5.76±0.29**	**68.29±8.06**	**8.25±0.48**
Workday + Time-of-Day + Temperature	12.74±0.49	243.28±20.98	15.88±0.57
Workday + Time-of-Day + Dew Point	7.36±0.42	97.08±9.51	9.97±0.36
Workday + Time-of-Day + Weather Condition	10.36±0.37	219.80±15.73	14.83±0.60
Workday + Time-of-Day + Wind Direction	7.24±21	89.49±10.22	9.46±0.34

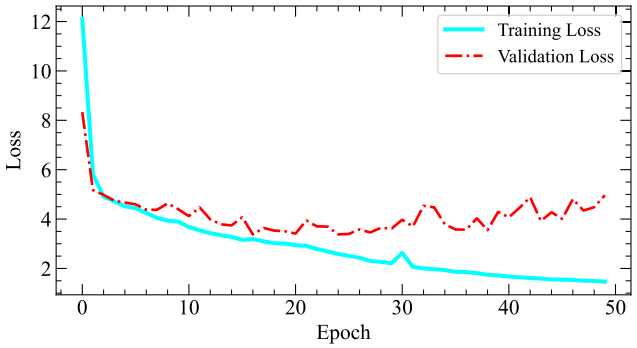


FIGURE 9. Training and validation loss over epochs of TFT model.

and statistical methods, though regression models have lower error metrics. For the 120-hour prediction length, the TFT model outperforms statistical methods. Fig. 11 illustrates that the TFT model's predictions are more accurate and align more closely with the actual values. The SVR model, despite its lower error metrics due to predictions being close to the mean of actual values, does not fit the actual plot as well.

We compared the computing time required in training and forecasting using various deep learning models. The N-BEATS, N-HiTS, and TFT models processed an 8,760 time index of training data. The training was conducted for 50 epochs using an AMD Ryzen 5 2600 CPU with 6 cores and 12 threads at 3.4 GHz and a GeForce RTX 3090 GPU with 10,496 CUDA cores and 24 GB of GDDR6X memory.

The results obtained for the statistical method, ARIMA, were 16.64 seconds for training and 0.01 seconds for forecasting. The regression method, SVR, required 47.18 seconds for training and 0.12 seconds for forecasting. For the deep learning models, the N-BEATS model took 87.43 seconds for training and 0.13 seconds for forecasting, followed by the N-HiTS model, which took 61.83 seconds for training and 0.14 seconds for forecasting. The TFT model had the longest training time of 228.09 seconds (2.61 times longer than N-BEATS and 3.69 times longer than N-HiTS), with a forecasting time of 0.19 seconds. The experimental results show that while the TFT model took the longest time for training, the forecasting times for all three models were roughly similar. In terms of memory usage, the N-BEATS model had 8.1 thousand parameters, the N-HiTS model had 2.1 thousand parameters, while the TFT model had 0.21 thousand parameters, making it approximately

TABLE 7. Comparison results in errors, number of parameters, training time, and forecasting time of statistical, regression, and deep learning methods.

Method		Statistical		Regression		Deep Learning			
Predict Length	**Metric**	**ARIMA**	**Exponential Smoothing**	**SVR**	**Random Forest**	**LSTM**	**N-BEATS**	**N-HiTS**	**TFT**
24 hours	MAE (kWh)	4.89	6.39	1.62	1.26	5.71	2.33	2.29	**1.13**
	MSE (kWh^2)	32.47	51.72	6.40	3.75	40.86	9.72	8.14	**3.69**
	RMSE (kWh)	5.70	7.19	2.53	1.94	6.39	3.12	2.85	**1.92**
	MASE	2.09	2.73	0.69	0.54	2.44	1.00	0.98	**0.48**
	OPE (%)	81.36	86.58	12.71	13.85	95.85	33.41	27.73	**10.27**
48 hours	MAE (kWh)	5.17	6.70	1.98	1.43	6.21	4.21	4.25	**1.16**
	MSE (kWh^2)	34.49	56.83	7.79	3.92	46.14	36.00	39.61	**3.44**
	RMSE (kWh)	5.87	7.54	2.79	1.98	6.79	6.00	6.29	**1.85**
	MASE	2.21	2.86	0.85	0.61	2.66	1.80	1.82	**0.50**
	OPE (%)	109.40	99.42	32.38	24.63	132.21	88.06	86.00	**15.46**
72 hours	MAE (kWh)	4.96	7.32	**2.01**	2.56	6.08	4.04	4.22	2.99
	MSE (kWh^2)	31.24	66.40	**8.05**	17.30	43.46	35.26	36.82	41.39
	RMSE (kWh)	5.59	8.15	**2.84**	4.16	6.59	5.94	6.07	6.43
	MASE	2.12	3.13	**0.86**	1.09	2.60	1.73	1.80	1.28
	OPE (%)	99.59	114.36	**28.58**	50.48	124.60	79.21	81.07	57.65
120 hours	MAE (kWh)	4.53	9.10	**2.23**	3.62	5.65	3.34	3.65	3.35
	MSE (kWh^2)	27.68	96.49	**12.57**	28.36	38.07	25.57	27.83	37.22
	RMSE (kWh)	5.26	9.82	**3.55**	5.33	6.17	5.06	5.27	6.10
	MASE	1.94	3.85	**0.96**	1.55	2.42	1.43	1.56	1.43
	OPE (%)	59.09	135.67	**1.80**	60.68	83.67	37.10	39.20	54.41
168 hours	MAE (kWh)	4.35	10.22	3.27	4.78	5.43	**3.18**	3.52	4.53
	MSE (kWh^2)	26.44	129.46	**18.07**	51.80	37.18	21.45	23.81	55.95
	RMSE (kWh)	5.14	11.38	**4.25**	7.20	6.10	4.63	4.88	7.48
	MASE	1.86	4.37	1.40	2.04	2.32	**1.36**	1.50	1.94
	OPE (%)	53.80	158.07	32.76	75.33	79.59	29.88	**25.76**	67.70
Total Parameters (k)		N/A	N/A	N/A	N/A	0.17	8.10	2.10	0.21
Training Time (second)		16.64	0.95	47.18	27.07	87.89	87.43	61.83	228.09
Forecasting Time (second)		0.01	0.09	0.12	0.07	0.08	0.13	0.14	0.19

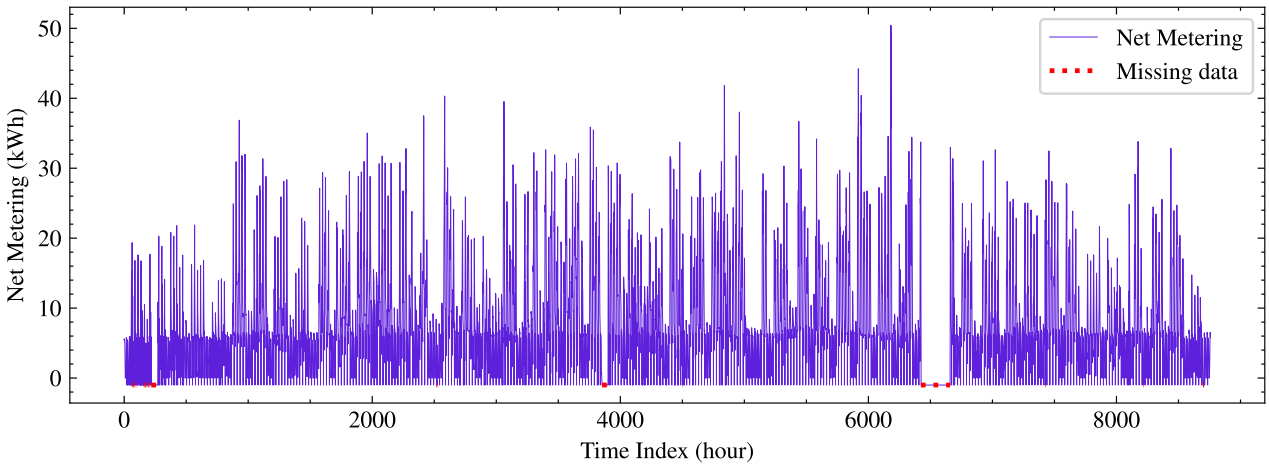


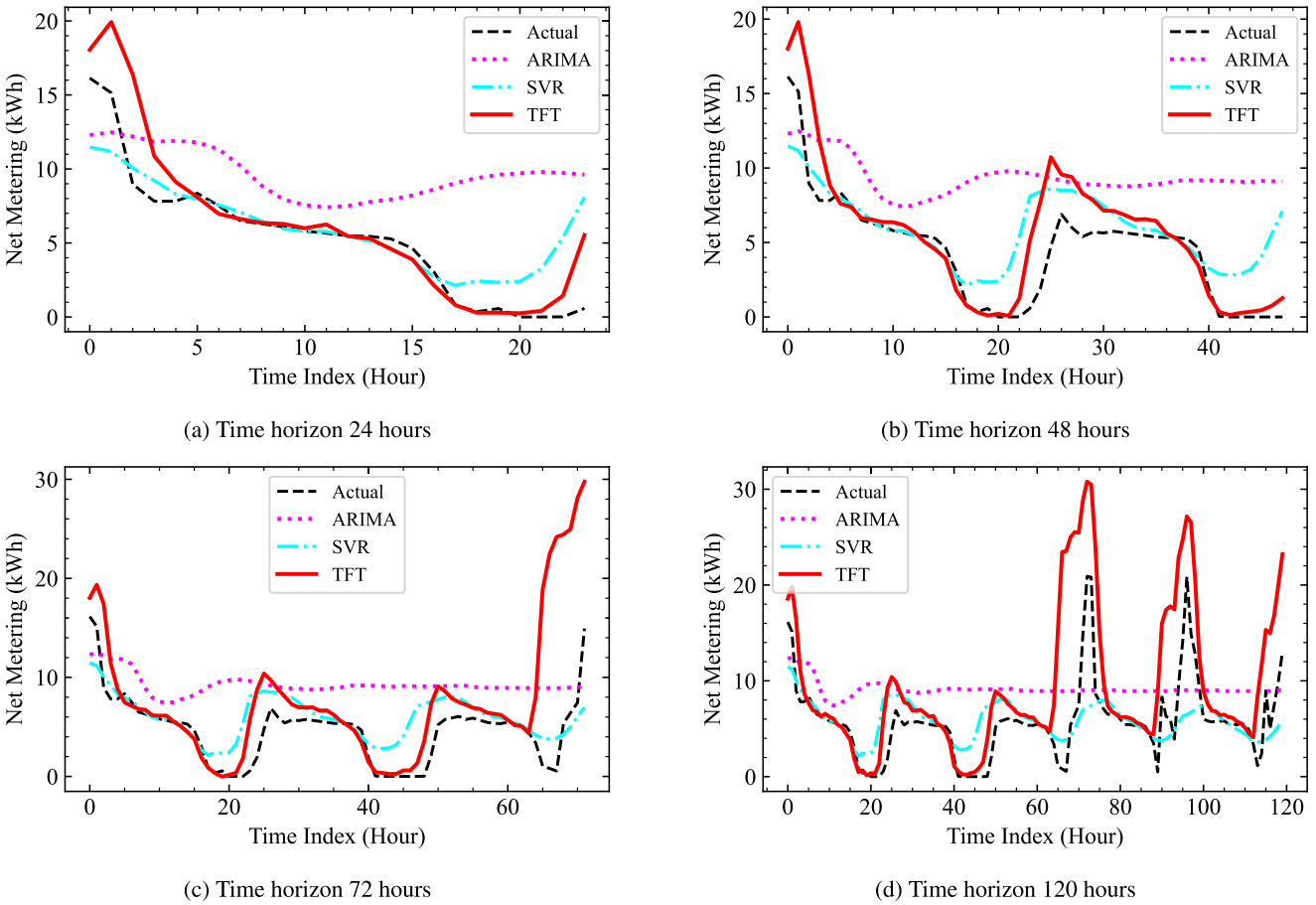
FIGURE 10. Net metering profile of SGtech prosumer building from January 1, 2023 to December 31, 2023, including 365 days (8,760 hours).

38.57 times more memory efficient than N-BEATS and 10 times more memory efficient than N-HiTS, as illustrated in Table 7.

The results in Table 7 show that the TFT model outperforms other methods in short-term forecasting (24 and 48 hours), achieving the lowest error metrics across all methods. However, for long-term forecasting horizons, the TFT model's accuracy slightly decreases, particularly compared to regression methods like SVR. However, the TFT model remains competitive, outperforming statistical and other deep learning models at 120 and 168 hours. In summary, the TFT model proves highly effective in short-term predictions, but further improvements are needed to enhance its long-term forecasting accuracy.

E. FORECAST ACCURACY WITH EXTENDING DATASET

In this experiment, we used an additional dataset to extend the training data for the TFT model, that is, the net metering dataset of 2023, as shown in Fig. 10. Initially, we trained the TFT model using only the 2023 dataset to establish a baseline. Subsequently, we trained the model using both the 2022 and the 2023 datasets in two different ways, that is, combining



(a) Time horizon 24 hours

(b) Time horizon 48 hours

(c) Time horizon 72 hours

(d) Time horizon 120 hours

FIGURE 11. Comparative forecasting accuracy of ARIMA, SVR, and TFT models over 24, 48, 72, and 120 hour time horizons.

TABLE 8. Mean absolute error (kWh) comparison for forecast horizons: 24, 48, 72, and 120 hours. Datasets: 2022, 2023, 2022&2023, and 2023 with pre-training on 2022 († indicates pre-training).

Experiment	Mean Absolute Error *(kWh)*				
	24 hrs	48 hrs	72 hrs	120 hrs	168 hrs
2022	1.13	1.16	2.99	3.35	4.53
2023	0.81	3.02	3.93	4.32	3.70
2022 & 2023	0.91	3.47	4.76	6.37	6.01
2023 †	0.58	3.07	3.64	4.23	2.60

TABLE 9. Mean absolute error for 24, 48, 72, 120, and 168 hours forecasting horizons in ablation experiment.

Experiment	Mean Absolute Error *(kWh)*				
	24 hrs	48 hrs	72 hrs	120 hrs	168 hrs
Baseline	1.13	1.16	2.99	**3.35**	4.53
No Dropout	**1.03**	1.23	**2.87**	3.84	4.68
MSE Loss	1.54	1.29	3.09	3.71	4.50
RMS Norm	1.13	**1.01**	3.00	3.48	**3.95**
AdamW Optimizer	1.12	1.08	3.64	4.23	4.49

the data from both years and using the 2022 dataset for pre-training followed by training with the 2023 dataset.

Table 8 shows the MAE for prediction lengths (24, 48, 72, 120, 168 hours). Generally, the pre-training method outperforms the baseline and the combined 2022-2023 datasets, except for the 48-hour prediction, where its performance is slightly worse but the difference is not significant. The combined dataset consistently gives higher MAE values than both the baseline and pre-training models.

The model's improved performance is due to pre-training on the 2022 dataset, allowing it to learn patterns and features that enhance its effectiveness on the 2023 dataset. This process helps the model capture new data patterns better, resulting in superior performance across most prediction lengths. Therefore, pre-training shows a clear advantage over baseline and combined dataset methods, showing the benefits of enhancing model performance with pre-training.

F. ABLATION EXPERIMENT

To evaluate our proposed TFT model, we conducted ablation experiments by tuning parameters including dropout, training loss, normalization layer, and optimizer. The results show the impact of different configurations on performance across prediction lengths. The forecasting periods were 24, 48, 72, 120, and 168 hours, with performance measured by MAE, as shown in Table 9.

In the first experiment, removing dropout from the model resulted in better accuracy than the baseline model for the

24- and 72-hour prediction lengths. In the second experiment, changing the loss function from quantile loss to MSE loss during the training phase resulted in lower accuracy across all prediction lengths. In the third experiment, replacing layer normalization with root mean square (RMS) normalization layer improved accuracy compared to the baseline model for the 48- and 120-hour prediction lengths. Lastly, changing the optimizer from Adam to AdamW slightly lowered the 24-hour prediction accuracy but showed better results for the 48- and 120-hour predictions.

Thus, for short-term prediction lengths (24 hours), removing dropout and using the AdamW optimizer are recommended for improving accuracy. For 48-hour predictions, RMS normalization is recommended. For the 72-hour prediction length, removing dropout is still our recommendation to improve the model. Finally, for 168-hour predictions, RMS normalization is recommended to achieve better accuracy.

V. EXPERIMENT ANALYSIS

Table 3 summarizes the various hyperparameter sets. The set of hidden layers, learning rate and attention head with values 32, 0.05 and 1 was optimal for the model, indicating that using 32 hidden layers was a good balance between complexity and efficiency during the training of the model and the learning rate of 0.05 was an appropriate step size for training. Interestingly, having just one Attention Head produced better results than using a higher number, as it prevents overfitting that results from excessive memorization of training data.

However, additional features, including temperature, dew point temperature, weather conditions, and wind direction, do not significantly improve predictive accuracy. Using too many related factors led to model confusion during training, making identifying essential contributors to the model's performance challenging.

Finally, segmenting the experiment periods based on SGtech's workday and time-of-day shows that using these features enhances prediction accuracy, particularly during non-working hours when net metering tends to decrease. Therefore, these factors, including weekends and holidays, play a crucial role in improving the accuracy of net metering predictions.

VI. CONCLUSION

In this study, we explored the effectiveness of the TFT model in net metering in smart grid prosumer buildings. We explored the impact of incorporating various features into the model, combining additional features, such as weather-related factors, with the usual workday and time-of-day operational features of the SGtech building. The TFT model demonstrated superior performance in handling these input features and showed good potential for accurate net metering forecasting. The integration of the additional operational features of workday and time-of-day was especially found to be a key factor in enhancing the model's precision, suggesting that those additional features played an essential role in net metering forecasting within the smart grid context.

In terms of all performance metrics for short-term prediction lengths of 24 hours and 48 hours, the TFT model is 100% better than other deep learning methods, regression methods, and statistical methods, while the forecasting time shows no significant difference from other methods. The TFT model is a promising tool for real-time net metering forecasting in smart grids. Our results contribute to enhancing the understanding of energy dynamics in smart grids and significantly inform future research into incorporating a wider range of data sources for improved real-time forecasting capabilities. Furthermore, Neural Architecture Search (NAS) can be considered for future work to improve the efficiency of the TFT model, making it more suitable for other datasets.

ACKNOWLEDGMENT

The authors would like to thank Roy I. Morien of the Naresuan University Graduate School for editing this manuscript's grammar, syntax, and general English expression.

REFERENCES

[1] A. S. Alahmed and L. Tong, "On net energy metering X: Optimal prosumer decisions, social welfare, and cross-subsidies," *IEEE Trans. Smart Grid*, vol. 14, no. 2, pp. 1652–1663, Mar. 2023.

[2] M. A. Khan, A. M. Saleh, M. Waseem, and I. A. Sajjad, "Artificial intelligence enabled demand response: Prospects and challenges in smart grid environment," *IEEE Access*, vol. 11, pp. 1477–1505, 2023.

[3] W. Ahmed, H. Ansari, B. Khan, Z. Ullah, S. M. Ali, C. A. A. Mehmood, M. B. Qureshi, I. Hussain, M. Jawad, M. U. S. Khan, A. Ullah, and R. Nawaz, "Machine learning based energy management model for smart grid and renewable energy districts," *IEEE Access*, vol. 8, pp. 185059–185078, 2020.

[4] T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, and H. Zareipour, "Energy forecasting: A review and outlook," *IEEE Open Access J. Power Energy*, vol. 7, pp. 376–388, 2020.

[5] A. Azeem, I. Ismail, S. M. Jameel, and V. R. Harindran, "Electrical load forecasting models for different generation modalities: A review," *IEEE Access*, vol. 9, pp. 142239–142263, 2021.

[6] J. C. López, M. J. Rider, and Q. Wu, "Parsimonious short-term load forecasting for optimal operation planning of electrical distribution systems," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1427–1437, Mar. 2019.

[7] X. Yao, X. Fu, and C. Zong, "Short-term load forecasting method based on feature preference strategy and LightGBM-XGboost," *IEEE Access*, vol. 10, pp. 75257–75268, 2022.

[8] B. Farsi, M. Amayri, N. Bouguila, and U. Eicker, "On short-term load forecasting using machine learning techniques and a novel parallel deep LSTM-CNN approach," *IEEE Access*, vol. 9, pp. 31191–31212, 2021.

[9] Q. Xiao, C. Li, Y. Tang, and X. Chen, "Energy efficiency modeling for configuration-dependent machining via machine learning: A comparative study," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 2, pp. 717–730, Apr. 2021.

[10] A. T. Eseye and M. Lehtonen, "Short-term forecasting of heat demand of buildings for efficient and optimal energy management based on integrated machine learning models," *IEEE Trans. Ind. Informat.*, vol. 16, no. 12, pp. 7743–7755, Dec. 2020.

[11] A. A. 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. 134911–134939, 2020.

[12] J. Son, J. Cha, H. Kim, and Y.-M. Wi, "Day-ahead short-term load forecasting for holidays based on modification of similar days' load profiles," *IEEE Access*, vol. 10, pp. 17864–17880, 2022.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[14] A. Umer, C. Termritthikun, T. Qiu, P. H. W. Leong, and I. Lee, "On-device saliency prediction based on pseudoknowledge distillation," *IEEE Trans. Ind. Informat.*, vol. 18, no. 9, pp. 6317–6325, Sep. 2022.
[15] C. Termritthikun, A. Umer, S. Suwanwimolkul, F. Xia, and I. Lee, "Explainable knowledge distillation for on-device chest X-ray classification," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, early access, May 2, 2023, doi: 10.1109/TCBB.2023.3272333.
[16] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *Int. J. Forecasting*, vol. 37, no. 1, pp. 388–427, Jan. 2021.
[17] F. Jamil, N. Iqbal, S. Ahmad, and D. Kim, "Peer-to-peer energy trading mechanism based on blockchain and machine learning for sustainable electrical power supply in smart grid," *IEEE Access*, vol. 9, pp. 39193–39217, 2021.
[18] Y. Hong, Y. Zhou, Q. Li, W. Xu, and X. Zheng, "A deep learning method for short-term residential load forecasting in smart grid," *IEEE Access*, vol. 8, pp. 55785–55797, 2020.
[19] B. N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 221–231.
[20] X. Wang, Y. Zhang, W. Liu, J. Yin, and J. Zhang, "N-HiTS: Neural hierarchical interpolation for time series forecasting," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, 2020, pp. 2663–2666.
[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[22] J. D. M.-W. C. Kenton and L. K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.
[23] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, and J. Uszkoreit, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2021, pp. 3146–3154.
[24] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 1, pp. 5485–5551, 2020.
[25] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 22419–22430.
[26] 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," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 12, pp. 11106–11115.
[27] B. Lim, S. Ö. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *Int. J. Forecasting*, vol. 37, no. 4, pp. 1748–1764, Oct. 2021.
[28] P. C. Huy, N. Q. Minh, N. D. Tien, and T. T. Q. Anh, "Short-term electricity load forecasting based on temporal fusion transformer model," *IEEE Access*, vol. 10, pp. 106296–106304, 2022.
[29] H. Zhang, Y. Zou, X. Yang, and H. Yang, "A temporal fusion transformer for short-term freeway traffic speed multistep prediction," *Neurocomputing*, vol. 500, pp. 329–340, Aug. 2022.
[30] M. López Santos, X. García-Santiago, F. E. Camarero, G. B. Gil, and P. C. Ortega, "Application of temporal fusion transformer for day-ahead PV power forecasting," *Energies*, vol. 15, no. 14, p. 5232, Jul. 2022.

SORAWUT JITTANON (Graduate Student Member, IEEE) received the B.Eng. degree in electrical engineering from Chiang Mai University, Chiang Mai, Thailand, in 2017, and the M.Sc. degree in smart grid technology from Naresuan University, Phitsanulok, Thailand, in 2023. He is currently pursuing the Ph.D. degree in smart grid technology.

He was an Electrical Engineer with Sino-Thai Engineering and Construction, from 2019 to 2020. His current research interests include time series forecasting and deep learning.

YODTHONG MENSIN received the B.Sc. degree in computer science and the M.Sc. degree in information technology from Naresuan University, Thailand, and the Ph.D. degree in energy, communities, and the environment from Chiang Mai Rajabhat University. He is currently the Deputy Director of Research and Academics Affairs with the School of Renewable Energy and Smart Grid Technology (SGtech), Naresuan University. He is also a Permanent Speaker of IEEE Conference in the Section of Power and Energy Society (PES) in Thailand. He has experience in the field of a microgrid systems, automated demand response (ADR), virtual power plant (VPP), energy management systems, smart grid data utilization, and energy trading platform with blockchain technology for more than 15 years. Currently, he is an advisory's team member of the mega-project implementation of Thailand utilities, the ministry of energy, and private sector. He also received the Certificate in Renewable Energy Technology from the University of Applied Science, Stralsund, Germany.

NIPON KETJOY received the B.Sc. degree in physics-energy from Naresuan University, Thailand, in 1997, the M.Sc. degree in energy technology from the King Mongkut's University of Technology, Thailand, in 2000, and the Dr.-Ing. degree (magna cum laude) in elektrotechnik from the University of Kassel, Germany, in 2005.

Since 2000, he has been with Naresuan University, where he is currently an Associate Professor and the Director of the School of Renewable Energy and Smart Grid Technology. He has authored or co-authored various articles in academic journals and international conferences. His current research interests include photovoltaic systems, renewable energy technology, smart grid systems, and battery energy storage systems. Since 2016, he has been a member of the Committee of the IEEE Power and Energy Society, Thailand.

CHAKKRIT TERMRITTHIKUN (Member, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees in computer engineering from Naresuan University, Thailand, in 2013, 2017, and 2021, respectively.

From 2018 to 2020, he was a Visiting Research Student with the University of South Australia, Adelaide, Australia. He is currently a Lecturer and the Assistant Director of Digital Innovation with the School of Renewable Energy and Smart Grid Technology, Naresuan University. His current research interests include deep learning, neural architecture search, and on-device artificial intelligence.