Forecasting of PV Plant Output Using Interpretable Temporal Fusion Transformer Model

Md Maidul Islam, Salman Sadiq Shuvo, Md Jamal Ahmed Shohan and Md Omar Faruque

September 12, 2023

Forecasting of PV Plant Output Using Interpretable Temporal Fusion Transformer Model

Md Maidul Islam
Electrical Engineering
Florida State University
Tallahassee, FL, US
mi19b@fsu.edu

Salman Sadiq Shuvo
Electrical Engineering
Pacific Northwest National Laboratory
Tampa, FL, US
salman.shuvo@pnnl.gov

Md Jamal Ahmed Shohan
Electrical Engineering
Florida State University
Tallahassee, FL, US
ms19bi@fsu.edu

Md Omar Faruque
Electrical Engineering
Florida State University
Tallahassee, FL, US
mfaruque@eng.famu.fsu.edu

***Abstract*—The stochastic nature of solar energy generation poses a challenge for grid operators, especially with higher penetration of solar-based renewables in the grid. This paper proposes an attention-based temporal fusion transformer (TFT) model for short-term (an hour ahead) photovoltaic (PV) power forecasting using available geographic data such as solar irradiation, temperature, and statistical features extracted from historical PV data. TFT utilizes a self-attention layer for long-term dependencies where recurrent networks are used for local processing. The model selects relevant features through a series of gating layers to achieve high performance for multi-horizon forecasting. The temporal fusion transformer model also provides interpretable insights into the temporal dynamics of different features. A real-world PV dataset has been utilized to compare the model performance with some other state-of-the-art forecasting models.**

***Index Terms*—PV forecasting, temporal fusion transformer, interpretable machine learning, multi-horizon forecasting.**

I. INTRODUCTION

Solar energy is the most popular and fastest-growing renewable resource to replace traditional fuel-based power plants. International Energy Agency (IEA) predicts the total global photovoltaic (PV) penetration will cross 1700 GW by 2030 [1]. Although solar is an infinite replenishable source of energy, the highly stochastic and unpredictable nature of solar makes optimal and reliable planning of utilities a challenging task. This issue is more relevant in recent times, as PV penetration in the power grid is expected to be doubled in the upcoming decade.

Accurate forecasting of the generation of PV plants is essential for the utilities as the deficit between demand and solar generation must be compensated with the rest of the energy sources. Especially, short-term accurate forecasting of PV output is a prerequisite for a renewable-dependent grid. PV forecasting depends directly on meteorological and geographical characteristics like solar irradiation, direct normal radiation, diffuse horizontal radiation, and global horizontal radiation. Researchers have used different methods to forecast the generation of solar plants and these methods can be divided into three major categories: statistical models, machine learning (ML) based models, and hybrid models. Researchers have utilized different regression models to forecast PV output by developing mathematical correlations among variables. The most popular among these methods is auto-regressive integrated and moving average (ARIMA) [2]. However, ARIMA does not adapt well to the non-stationary behavior of PV output. To compensate for this issue, an updated version including seasonality named seasonal auto-regressive integrated moving average (SARMIA) [3] and auto-regressive integrated and moving average with explanatory variables (ARIMAX) were developed by the researchers. XGBoost has shown impressive performances in pv forecasting and has been extensively used along with feature engineering by researchers [4]. The researchers in [4] presented XGBoost and feature engineering-based PV forecasting methods for a microgrid system with solar energy. Different empirical and Markov chain-based models have also been utilized by researchers to improve the accuracy of solar forecasting [5]. Recently, time series forecasting based on artificial intelligence (AI) based models has attained popularity due to higher accuracy and ease of implementation. Most of the AI/ML techniques use numerical weather prediction (NWP) as feature input for the algorithms. A hierarchical approach to achieve intraday PV forecast based on artificial neural network (ANN) and support vector regression (SVR) based model was presented in [6]. Authors in [7] presented long short-term memory (LSTM) based PV prediction using meteorological information and achieved a reduction of error rate up to 11.8% compared to convolutional neural network. The hybrid model can be defined as a combination of different statistical and/or ML-based models. Authors in [8] presented a hybrid LSTM and deep neural network (DNN) to predict multi-horizon pv output based on wavelet decomposition of solar data. Researchers used deep reinforcement learning and LSTM-based load and PV forecasting for aiding load and electrical vehicle (EV) dispatch [9]–[11]. Artificial recurrent neural networks, LSTM, and an auto-encoder (AE) based model were utilized by [12] to forecast the output power from 21 PV plants. Authors in [13] combined autoencoder-LSTM with persistence model to predict the day-ahead solar output. Due to the several layers required to increase the forecasting system's accuracy, this approach takes more time for training and testing. For fields connected to the PV grid with tracker facilities, LSTM with AE has a high error rate. However, it does not require the selectivity of necessary features. One of the major shortcomings of the ML models is that they are essentially black box models and the feature importance or explainability of results can not be comprehended. Recently attention-based mechanism proposed in [14] has gained popularity especially, in natural language processing. The transformer-based models have shown improved performance for PV forecasting [15], [16]. Authors in [17] presented temporal fusion transformer (TFT) based multi-horizon forecasting with improved performance. The multivariate, multi-horizon forecasting capability and feature importance indicator make this model a great fit for the PV forecasting task. Accurate hour-ahead predictions can be particularly useful for tasks like optimal load dispatch, energy management, and scheduling electric vehicle (EV) charging [18]. This paper utilizes the TFT architecture to forecast hour-ahead PV output based on meteorological data, and statistical features of PV time series data. The major contributions of this paper are:

1) Develop a temporal fusion transformer-based PV forecasting method based on weather data and statistical features from PV time series data from a real-world PV plant.
2) Present an analysis of feature importance to indicate the interpretability of the model.
3) Compare with existing popular methods to highlight the efficacy of the proposed attention-based TFT model.

The remainder of the paper is organized as follows. Section II presents the temporal fusion transformer model architecture. Section III discusses the proposed forecasting methodology. Section IV shows the forecasting performance of the proposed model. Finally, section V concludes the paper.

II. Temporal Fusion Transformer Model Architecture

The TFT utilizes LSTM models combined with attention head mechanism [17]. Fig. 1 presents a simplified version of the temporal fusion transformer model architecture. The model uses static covariate encoder outputs as context vectors to be used in the later part of the architecture, a gating mechanism to isolate less important features, an LSTM layer for processing observed inputs locally, and a temporal self-attention decoder for learning long-term dependencies.

Let in a time series data, $s_i \in \mathbf{R}^{m_s}$ are static variables, $x_{i.t} \in \mathbf{R}^{m_x}$ are inputs, and $y_{i.t} \in \mathbf{R}^{m_x}$ are target outputs in time step $t \in [0, T_i]$. The time-independent inputs are observed inputs $z_{i.t} \in \mathbf{R}^{m_z}$, and already known inputs $x_{i.t} \in \mathbf{R}^{m_x}$. The forecast thus can be expressed as :

$$\hat{y}_i(q, t, \tau) = f_q(\tau, y_{i,t-k:t}, z_{i,t-k:t}, x_{i,t-k:t}, s_i) \quad (1)$$

Here, $\hat{y}_i(q, t, \tau)$ is the prediction output for the q-th sample quantile with τ step ahead forecasting. The TFT architecture can be divided into the following main components as discussed below.

1) Gating mechanisms: This is a Gated Residual Network (GRN) block that serves as a filtering mechanism for less important features. This is especially helpful for datasets with large input features where not all features may have significant

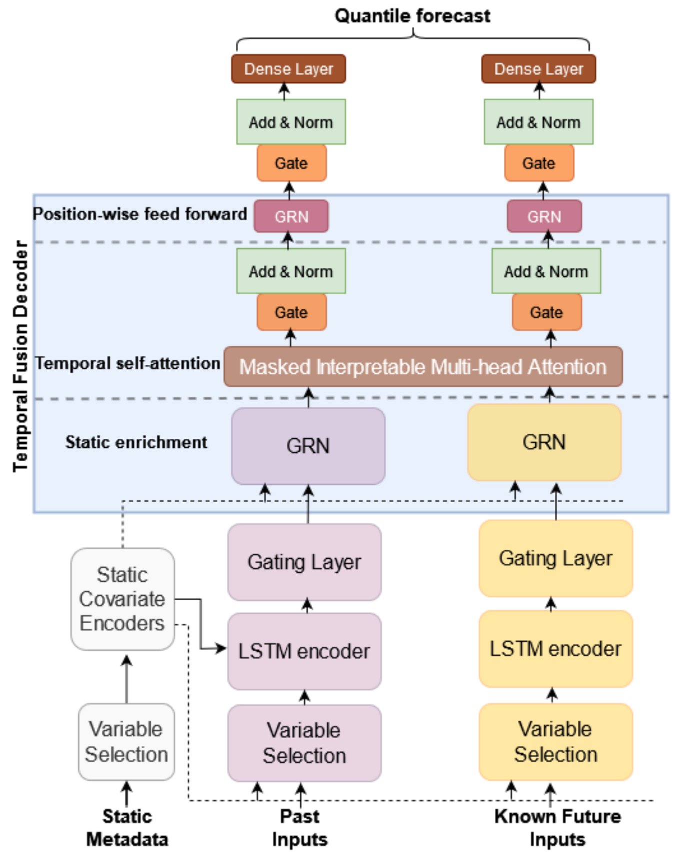


Fig. 1: Temporal fusion transformer model architecture

contributions to the forecasted output or contain noisy feature sets which can worsen the output. Let a be the primary input to GRN and c be an optional context vector. Then the GRN layer is as follows:

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

Where,

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

$$\eta_2 = ELU(W_{2,\omega}a + E_{3.\omega}C + b_{2,\omega}) \quad (4)$$

This GLU layer controls GRN contribution to the original input a and may skip the layer entirely if GLU outputs are close to 0.

2) Variable selection networks: This layer gives weight to the input features based on their variable importance. The transformed input, ζ_t is passed through a GRN and then to a softmax layer.

$$v_{xt} = softmax(GRN_{v_x}(\zeta_t, c_s)) \quad (5)$$

v_{xt} is a set of weights corresponding to the features and c_s is obtained from static covariate encoder. Finally, the processed features are weighted by their corresponding variable selection weights and combined as:

$$\bar{\zeta}_t = \sum_{j=1}^{m_x} v_{xt}^j \zeta_t^{-j} \quad (6)$$

3) Static covariant networks: To utilize static metadata, separate GRN encoders are used to produce four context vectors, temporal variable selection, c_s, local processing of temporal features, c_c, c_h, and enriching temporal features with static information c_e.

4) Multi-head attention mechanism: The TFT employs an attention mechanism to learn long-term relationships from the time series data. Let, the relationship between keys $K \in \mathbf{R}^{N \times d_{attn}}$ and queries $Q \in \mathbf{R}^{N \times d_{attn}}$ is described by attention mechanism values $V \in \mathbf{R}^{N \times d_v}$ as below:

$$attention(Q, K, V) = A(Q, K)V \tag{7}$$

Here, $A()$ is a normalization function. The multi-head attention layer is developed by employing different heads for different representation subspaces. In TFT, this multi-head attention is modified to share values in each head and can be expressed as :

$$IMH(Q, K, V) = \tilde{H}W_H \tag{8}$$

Here, $IMH()$ is interpretable multi-head attention function and $\tilde{H}$ is expressed as

$$\tilde{H} = \tilde{A}(Q, K)VW_V \tag{9}$$

where, $W_V \in \mathbf{R}^{d_{model} \times d_V}$ are shared weights across all heads.

5) Temporal fusion decoder: A sequence of LSTM encoder-decoder layers is employed to handle the anomalies or change points and feed $\zeta_{t-k:t}$ to the encoder and $\zeta_{t+1:t+\tau_{max}}$ to the decoder. This yields uniform temporal features, $\phi(t, n) \in \phi(t, -k) .... \phi(t, \tau_{max})$, and serves as the input to the decoder. Context vectors c_c, c_h from the static covariate encoders are used to initialize the cell and hidden state of the first LSTM in the layer.

6) Static enrichment layer: To emphasize the influence of static covariates on temporal dynamics, a static enrichment layer is introduced. The static enrichment layer can be expressed as:

$$\theta_{t,n} = GRN_\theta(\tilde{\phi}(t, n), c_e) \tag{10}$$

Here, c_e is a context vector from the static covariant encoder, and the weights from GRN_θ are shared across the layer.

7) Temporal self-attention layer: After the static enrichment layer, a self-attention layer is employed. First, all static temporal features are grouped together and multi-head attention is applied at each forecast time. To facilitate the training process, an additional gating layer is applied:

$$\delta(t, n) = LayerNorm(\theta(t, n) + GLU_\delta(\beta(t, n))) \tag{11}$$

This is then passed through additional non-linear processing through GRNs:

$$\psi(t, n) = GRN_\psi(\delta(t, n)) \tag{12}$$

The weights from GRN_ψ are shared across the entire layer.

8) Loss function: The loss function expressed by (13) is minimized by the TFT model during the training process.

$$\mathcal{L}(\Omega, W) = \sum_{y_t \in \Omega} \sum_{q \in Q} \sum_{\tau=1} \frac{QL(y_t, \tilde{y}(q, t-\tau, \tau), q)}{M\tau_{max}} \tag{13}$$

Here, Ω is the training data sample domain, W represents the weights of TFT, and Q is the set of output quartiles.

III. METHODOLOGY

A. Data preprocessing

The temporal fusion transformer model has been tested with real-world solar plant data from [19]. The data contains 15 min granular solar data generation for 34 days (May 15- June 17) from one PV plant in India. Meteorological data such as temperature ($°C$), irradiation (kWh/m^2), etc. are collected from station weather sensors. Plant 1 has a total of 23 inverters and the summation of these inverters give the total power generation of the plant. Fig. 2 shows AC power output from inverter 1 and irradiation data for the same period. We can see there is a high correlation between these two fields, increasing irradiation produces more power from the PV cells. The final AC power output also depends on the temperature and different inverter efficiency. The last 3 days from the dataset have been utilized as a testing data set (forecasting horizon) and the rest data is divided into 80:20 as training and validation datasets. The data was curated for any missing values. One hot encoding was applied to the inverter id features set. Apart from the time series features such as day of the month, and hour of the day, some statistical features including daily average, hourly average, etc were added.

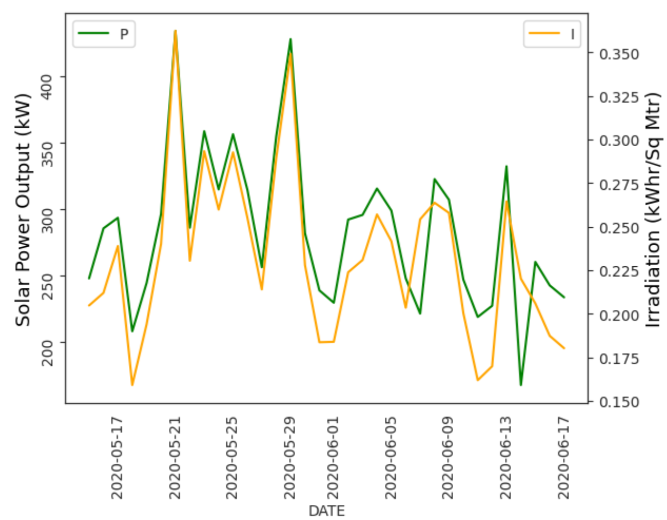
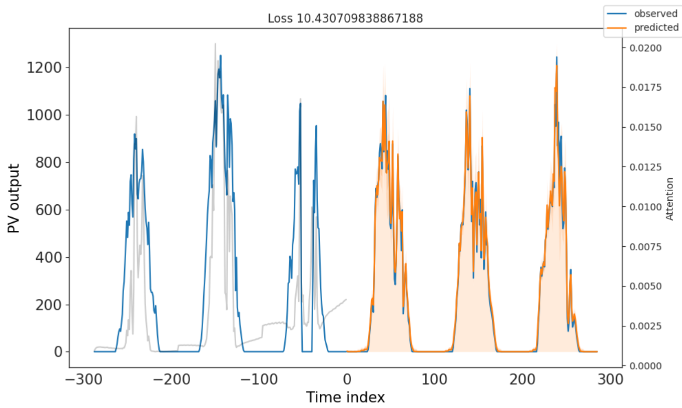


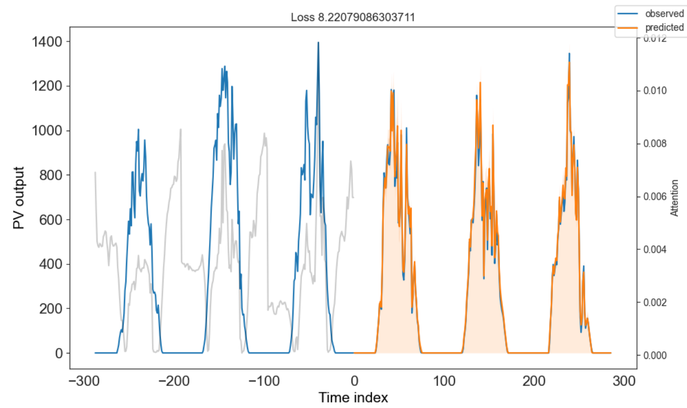
Fig. 2: PV output vs Irradiation

B. Temporal fusion transformer implementation

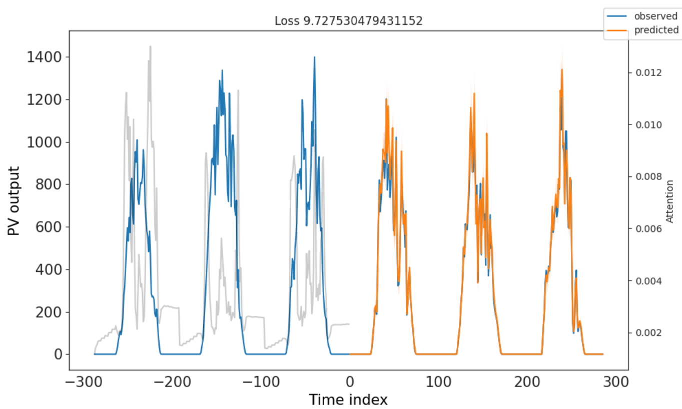
The temporal fusion transformer model excels other ML models as it supports different types of variables: time-varying known (date, day of week, month), time-varying unknown (irradiation, temperature), time-invariant real, time-invariant



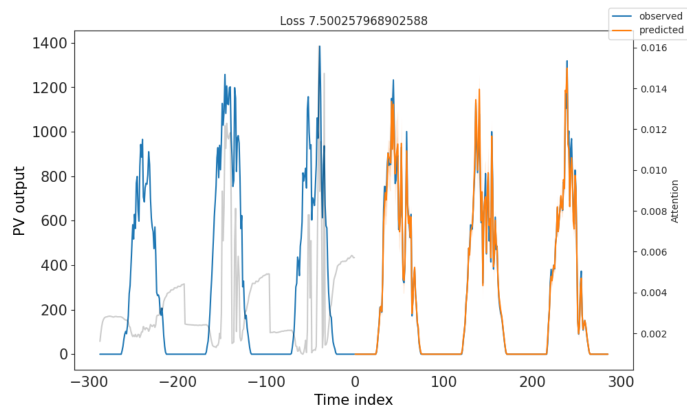
(a) Inverter 1



(b) Inverter 2



(c) Inverter 3



(d) Inverter 4

Fig. 3: Three-day prediction of solar output (kW) with temporal fusion transformer

categorical (inverter id). The attention mechanism allows the model to select the most important variables across the various time series feature sets through attention weights which help to improve forecasting accuracy. The model has been implemented on a computer with a 12^{th} Gen Core i7-1255U processor, 32GB ram, and an NVIDIA T550 graphics card. The model was implemented with Pytorch, and Pytorch optimize hyperparameter functionality was used to find the best values. The training time was about 42 minutes with the selected hyperparameters. Table I shows the hyperparameters used for the model where hidden size represents the dimensionality or size of the hidden states, and hidden continuous size refers to the dimensionality of the hidden continuous features. Attention head is selected based on the size of the dataset and dropout typically varies between 0.1 and 0.3. Among all the hyperparameters, hidden size, and learning rate are the most critical parameters.

IV. Forecasting Performance Evaluation

The three-day PV forecasted output from four different inverters out of 23 inverters using the proposed model is shown in Fig. 3. The gray lines in each figure represent the attention variance of the model during the training phase. The TFT shows exemplary performance for all different inverter outputs, which underlines its robustness for varying datasets. Fig. 4 shows the PV forecast comparison between LSTM and TFT for hour-ahead prediction over a day. It can be easily seen from the figure that TFT more closely matches the actual data than LSTM.

TABLE I: Hyperparameter for temporal fusion transformer network

Model Parameter	Value
Hidden size	103
Hidden continuous size	71
Attention head size	2
Dropout	0.17
Learning rate	0.089

In order to properly evaluate the performance of the proposed method four different performance metrics are selected. They are mean square error (MSE), mean absolute percentage error (MAPE), mean absolute error (MAE), and coefficient of determination (R^2). The equations of the metrics are provided below.

$$MAE = \frac{\frac{1}{n}\sum_{i=1}^{n}(\hat{p}_i - p_i)}{n} \tag{14}$$

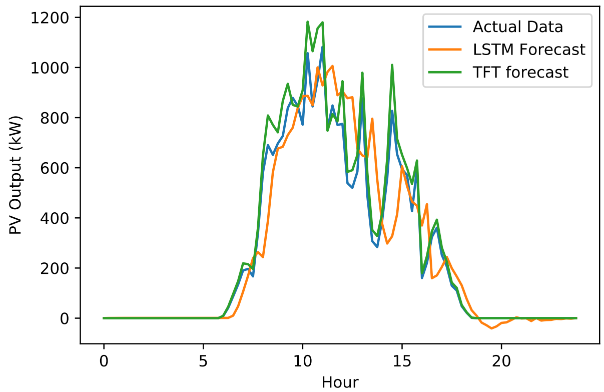


Fig. 4: PV output comparison between LSTM and Temporal Fusion Transformer for a day.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p_i - \hat{p}_i)^2 \quad (15)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\mid p_i - \hat{p}_i \mid}{c} \quad (16)$$

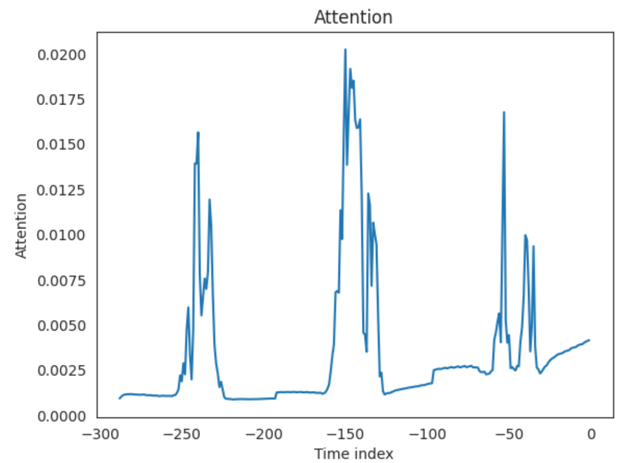
$$R^2 = 1 - \frac{\sum_{i=1}^{n}(p_i - \hat{p}_i)^2}{\sum_{i=1}^{n}(p_i - \bar{p}_i)^2} \quad (17)$$

Here, n is the number of samples, $\hat{p}_i$ is the predicted value, p_i is the actual value, and c is the capacity of the power plant. The MAE score, also known as Mean Absolute Error, calculates the average absolute deviation between the predicted values and the actual values. A smaller MAE implies a superior performance of the model. The MSE score, which stands for Mean Squared Error, computes the average of the squared differences between the predicted values and the actual values. MAPE calculates the average of the absolute percentage errors between actual values and predictions. Similar to the MAE, a lower MSE and MAPE score suggest better model performance. The R^2 score quantifies the percentage of the dependent variable's variability that can be accounted for by the independent variables. The close the value of R^2 is to 1, the better the model's performance. A comparison study for Plant 1 is performed with three other methods: LSTM, ARIMA, and Neural Prophet [20]. All metrics have been calculated based on the test samples. Table II summarizes these performance metrics for all the models. The proposed TFT model outperforms all the other models for all performance evaluation metrics.

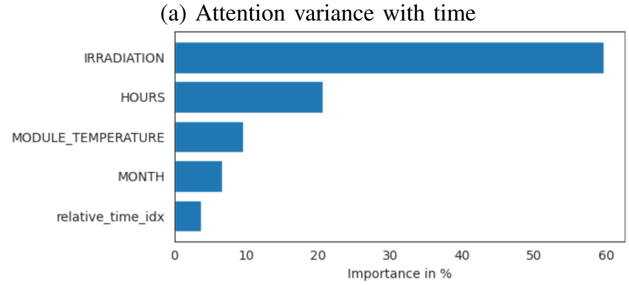
TABLE II: Performance comparison of the proposed model

Model Name	MSE	MAPE	MAE	R^2
ARIMA	24.25	0.78	28.23	0.89
LSTM	34.55	2.07	24.30	0.82
Neural Prophet	26.52	0.65	18.12	0.95
TFT	26.21	0.11	13.00	0.99

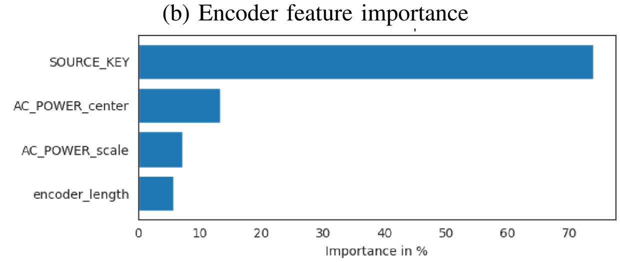
Finally, an analysis has been performed to find interpretable insights into temporal dynamics. Fig. 5a shows the development of the attention mechanism with time for the model. Fig. 5b, and 5c show the variable importance of encoder features and static variables respectively. From the figures, it can be found that irradiation is the most important feature in the encoder feature set, whereas inverter id is the most important static variable. These observances are aligned with physical significance as irradiation, the hour of the day is directly related to the PV output power and inverter efficiency plays a key role in the amount of AC power generated.



(a) Attention variance with time



(b) Encoder feature importance



(c) Static variable importance

Fig. 5: Feature variable importance and attention variance

V. Conclusion

This paper presented a temporal fusion transformer-based hour-ahead PV forecasting methodology. PV forecasting involves highly stochastic meteorological data like irradiation and temperature. The features contributing to solar output generation are mostly time-varying unknown covariates. The TFT architecture is capable of multi-horizon forecasting considering the complex mix of input variables including known and unknown future input variables. TFT produced better results in comparison with some of the state-of-the-art forecasting models for a real PV dataset. Finally, an analysis of TFT produced interpretable insights into the temporal dynamics of the input features has been presented.

References

[1] Das, U., Tey, K., Seyedmahmoudian, M., Mekhilef, S., Idris, M., Van Deventer, W., Horan, B. & Stojcevski, "A. Forecasting of photovoltaic power generation and model optimization: A review," *Renewable And Sustainable Energy Reviews*, **81**, pp. 912-928, 2018.

[2] Y. Zhang, M. Beaudin, H. Zareipour and D. Wood, "Forecasting Solar Photovoltaic power production at the aggregated system level," *2014 North American Power Symposium (NAPS)*, Pullman, WA, USA, pp. 1-6, 2014.

[3] Kushwaha, Vishal, and Naran M. Pindoriya, "Very short-term solar PV generation forecast using SARIMA model: A case study," *In 2017 7th international conference on power systems (ICPS)*, pp. 430-435, IEEE, 2017.

[4] Mohamed, M., Mahmood, F., Abd, M., Chandra, A. & Singh, B., "Dynamic Forecasting of Solar Energy Microgrid Systems Using Feature Engineering," *IEEE Transactions On Industry Applications*, **58**, pp. 7857-7869, 2022.

[5] Bai, Xiaoyang, Liang Liang, and Xueqin Zhu, "Improved Markov-chain-based ultra-short-term PV forecasting method for enhancing power system resilience," *The Journal of Engineering*, no. 2, pp. 114-124, 2021.

[6] Li, Z., Rahman, S., Vega, R. & Dong, B., "A Hierarchical Approach Using Machine Learning Methods in Solar Photovoltaic Energy Production Forecasting," *Energies*, **9**, pp. 55, 2016.

[7] Hossain, M. & Mahmood, H., "Short-Term Photovoltaic Power Forecasting Using an LSTM Neural Network and Synthetic Weather Forecast," *IEEE Access*, **8**, pp. 172524-172533, 2020.

[8] Ospina, J., Newaz, A. & Faruque, M., "Forecasting of PV plant output using hybrid wavelet-based LSTM-DNN structure model," *IET Renewable Power Generation*, **13**, pp. 1087-1095, 2019.

[9] Shuvo, S., Islam, M. & Yilmaz, Y., "DROP: Deep Reinforcement Learning Based Optimal Perturbation for MPPT in Wind Energy," *2022 North American Power Symposium (NAPS)*, pp. 1-6, 2022.

[10] Shuvo, S. & Islam, M., "LSTM Based Load Prediction for Distribution Power Grid with Home EV Charging," *2022 IEEE Kansas Power And Energy Conference (KPEC)*, pp. 1-5, 2022.

[11] Shuvo, Salman Sadiq and Yilmaz, Yasin. "Demand-side and Utility-side Management Techniques for Increasing EV Charging Load," *IEEE Transactions on Smart Grid*, 2023.

[12] Gensler, A., Henze, J., Sick, B. & Raabe, N., "Deep Learning for solar power forecasting — An approach using AutoEncoder and LSTM Neural Networks," *2016 IEEE International Conference On Systems, Man, And Cybernetics (SMC)*, pp. 2858-2865, 2016.

[13] Zhang, Y., Qin, C., Srivastava, A., Jin, C. & Sharma, R., "Data-Driven Day-Ahead PV Estimation Using Autoencoder-LSTM and Persistence Model," *IEEE Transactions On Industry Applications*, **56**, pp. 7185-7192, 2020.

[14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L. & Polosukhin, I., "Attention Is All You Need," *Advances in neural information processing systems*, 2017.

[15] Zhou, H., Zhang, Y., Yang, L., Liu, Q., Yan, K. & Du, Y., "Short-Term Photovoltaic Power Forecasting Based on Long Short Term Memory Neural Network and Attention Mechanism," *IEEE Access*, **7**, pp. 78063-78074, 2019.

[16] Kothona, D., Spyropoulos, K., Valelis, C., Sarigiannidis, C., Chatzisavvas, K. & Christoforidis, G., "Efficient 24-hour ahead PV energy production forecasting employing a transformer-based model," *2022 2nd International Conference On Energy Transition In The Mediterranean Area (SyNERGY MED)*, pp. 1-6, 2022.

[17] Lim, B., Arik, S., Loeff, N. & Pfister, T., "Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting," *International Journal of Forecasting 37.4*, pp. 1748-1764, 2021.

[18] Shuvo, S. & Islam, M. Optimal Dispatch for A Microgrid with Distributed Generations and EV Charging Load. *2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*. pp. 1-5, 2023.

[19] Kannal, A., "Solar Power Generation Data," 2020. Available at http://www.kaggle.com/datasets/anikannal/solar-power-generation-data/code (2023/07/30)

[20] Shohan, Md Jamal Ahmed, Md Omar Faruque, and Simon Y. Foo. "Forecasting of electric load using a hybrid LSTM-neural prophet model," *Energies*, **15**, no.6, pp. 2158, 2022.