Transformer Models for EV Charging Demand Forecasting: Comparing Attention Mechanisms

Taniya Manzoor
Dept. of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi, India
eez228195@ee.iitd.ac.in

Brejesh Lall
Dept. of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi, India
brejesh@ee.iitd.ac.in

B.K. Panigrahi
Dept. of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi, India
bkpanigrahi@ee.iitd.ac.in

***Abstract*—Electric Vehicle (EV) adoption is surpassing predictions, with a sharply ascending trajectory. A robust EV charging ecosystem is imperative for grid stability, which faces potential strain from the increasing demand of EV charging loads. Accurate forecasting of EV charging demand is crucial for maintaining a resilient vehicle-grid interaction. EV charging demand forecasting, can be conceptualized as a long sequence time-series forecasting (LSTF) problem. However, majority of the forecasting models struggle to capture long-range dependencies effectively. This work represents an early attempt at leveraging the merits of Transformer-based models for forecasting EV charging demand. Additionally, we present a comparative analysis between two widely-used attention mechanisms, Self-attention and ProbSparse attention, aiming to identify the optimal solution for the EV charging demand forecasting problem. Experiments on real-world Adaptive Charging Network (ACN) dataset demonstrate that the computationally inexpensive, ProbSparse attention based model outperforms the canonical self-attention model, providing a new solution to the EV charging demand LSTF problem.**

***Index Terms*—EV charging demand forecasting, Transformer, Informer, Self-attention, ProbSparse attention**

I. Introduction

The rapid electric vehicle (EV) adoption rates fit with the global urgency to decarbonize, fostering a sustainable and environmentally responsible future. Electric vehicles crossed 14% of total car sales in 2022, and are expected to reach 30% by the end of this decade [1]. Rapid advancements in various sectors of EV's like battery technology (solid-state batteries, lithium iron phosphate (LFP) cathodes, magnesium-ion with twin-graphene), charging technology (V2G charging, wireless charging, smart charging) and revised legislations (from taking initiatives to boost local manufacturing, setting performance standards to providing incentives for EV purchase) are only going to accelerate the growth of EV sector. However, the lack of charging infrastructure continues to remain a significant bottleneck for extensive uptake of EVs.

Anticipating a swift growth in the electric vehicle (EV) sector, it is expected that the power grids will experience a substantial surge in demand. Additionally, unregulated charging of electric vehicles results in significant fluctuations in electrical load within the grid, adversely affecting the power quality of the distribution network [2]. Furthermore, the uncontrolled charging patterns contribute to significant imbalance, resulting in violations of voltage constraints, decreased network hosting capacity, elevated energy losses, and increased demands for power generation. [3]. Hence, charging load forecasting is one of the crucial challenges in the development of EV charging infrastructure. Accurate EV charging load forecasting can help in designing intelligent scheduling algorithms for the purpose of future demand planning, infrastructure planning, user-experience improvement (minimization of range anxiety and queuing time) and optimization of operational efficiency of charging networks. Researchers have shown considerable interest in addressing the challenge of forecasting EV charging load. From the implementation of traditional load forecasting models like Pattern Sequence Forecasting (PSF) [4] and autoregressive integrated moving average (ARIMA) [5], to the utilization of machine learning models such as Support Vector Machines (SVM) [6], Random Forest (RF) [7], Gradient Boosting Regression Tree (GBRT) [8] and Artificial Neural Network (ANN) [9], it is evident that a comprehensive range of forecasting methodologies has been extensively explored. Furthermore, in recent studies on EV load forecasting, there has been a notable focus on incorporating deep learning approaches, including Long Short-Term Memory (LSTM) [10], Gated Recurrent Units (GRU) [11] and Recurrent Neural Networks (RNNs) [12]. However, these models face major technical challenges, such as gradient vanishing, while also struggling to effectively capture long-range dependencies [13]. As EV charging load exhibits recurring patterns over both short and long periods, the effectiveness of these models for forecasting applications may be limited.

Many real-world applications such as long-term energy planning, traffic management and extreme weather predictions require the prediction of long sequence time-series. In 2017, Vaswani et.al proposed a new deep learning architecture, the Transformer [14] which revolutionized the field of artificial intelligence, delivering unparalleled performance across a range of applications, such as natural language processing (NLP), computer vision and speech recognition. Recently, there has also been a surge in utilizing Transformer-based solutions for time-series forecasting problems. Several studies have indicated that, in the field of load forecasting problems, Transformers outperform traditional techniques such as ARIMA and SARIMA [15], as well as deep learning methods like sequence-to-sequence algorithms [16], RNNs, and LSTM [17].

The primary functional capability of Transformers lies in the multi-head self-attention mechanism, demonstrating an exceptional ability to extract semantic correlations among elements in long sequences. The self-attention mechanism prioritizes and emphasizes relevant information improving both accuracy and computational efficiency. However, recent research highlights several issues with the Transformer, limiting its direct use in long sequence time-series forecasting (LSTF). The most significant limitations while employing Transformers for LSTF problems are the quadratic time complexity, memory issues due to stacking of encoder/decoder layers, dynamic step-by-step decoding affecting the inference speed [18]. In view of this, extensive research has been focused on improving efficiency and reducing the complexity of self-attention mechanism. In 2021, Zhou et al. proposed Informer [18], a transformer-based model for LSTF, utilizing ProbSparse self-attention to handle challenges of memory and complexity in the canonical Transformer.

Acknowledging the imperative for comprehensive research in electric vehicle (EV) load forecasting tasks, we outline the following contributions in this work:

1) To the best of our knowledge, this work is one of the first few attempts in utilizing Transformer-based architecture in the domain of EV load forecasting problem.
2) We present a comparative study between canonical dot-product self-attention and Probsparse self-attention mechanisms to ascertain their respective efficacy.
3) Experiments on real-world dataset illustrate the utility of making the model lighter. The lighter Informer model demonstrates better accuracy compared to its self-attention variant. Especially, its ability to precisely predict longer sequences stands out as a significant advancement, effectively tackling a bottleneck in EV load forecasting.

II. PRELIMINARIES

A. Transformer

Transformer [14] utilizes an encoder-decoder framework, employing stacked self-attention mechanisms and fully connected layers for both the encoder and decoder components . Each encoding layer contains a multi-head self-attention mechanism and feed-forward neural networks. The decoding layer introduces a cross-attention layer alongside the existing sub-layers in the encoder, to perform multi-head attention over the output of the encoding stack. The main features of the transformer model are as follows:

1) Positional Encoding: The Transformer model dispenses with recurrence and convolution, hence it relies on positional encoding to capture sequential information within the data. The Vanilla Transformer uses the following sine and cosine functions for positional encoding:

$$PE(P,2i) = \sin\left(\frac{P}{10000^{2i/D_m}}\right) \tag{1}$$

$$PE(P,2i+1) = \cos\left(\frac{P}{10000^{2i/D_m}}\right) \tag{2}$$

where P is the position, i is the dimension, and D_m is the model dimension

2) Feed-forward and Residual Network: The feed-forward network essentially comprises of two linear transformations with a ReLU activation in between.

$$FFN(H_0) = \text{ReLU}(H_0W_1 + b_1)W_2 + b_2 \tag{3}$$

where H_0 is the output of the previous layer, $\mathbf{W}_1 \in \mathbb{R}^{D_m \times D_f}$, $\mathbf{W}_2 \in \mathbb{R}^{D_f \times D_m}$, $\mathbf{b}_1 \in \mathbb{R}^{D_f}$, $\mathbf{b}_2 \in \mathbb{R}^{D_\text{m}}$ are trainable parameters.

Residual connections link the output of a preceding layer to the input of the subsequent layer, followed by layer normalization to stabilize training [35].

$$H_\mathbf{0} = \textit{LayerNorm}(\textit{SelfAttn}(X) + X) \tag{4}$$

$$H = \textit{LayerNorm}(\textit{FFN}(H_\mathbf{0}) + H_\mathbf{0}) \tag{5}$$

where *SelfAttn* denotes self-attention module and *LayerNorm* denotes the layer normalization operation.

3) Multi-head Attention : The core innovation in Transformer lies in the self-attention mechanism in the multi-head self-attention layers, allowing the model to assess the significance of various input tokens during prediction. The relationship between different input tokens are considered simultaneously rather than relying on fixed-size context windows [36]. Transformer model uses the Query-Key-Value model, to calculate the scaled dot-product attention given by:

$$Attn_{\textit{Dot-Pro}}(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{D_k}}\right)V \tag{6}$$

where queries $\mathbf{Q} \in \mathbb{R}^{L_\text{q} \times D_k}$, keys $\mathbf{K} \in \mathbb{R}^{L_\text{k} \times D_k}$, and values $\mathbf{V} \in \mathbb{R}^{L_\text{k} \times D_v}$. L_q and L_k denote the lengths of queries and keys (or values), and D_k and D_v denote the dimensions of keys (or queries) and values. The output values of different heads functioning in parallel are concatenated resulting in the final values.

$$\text{MultiHeadAttn}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{Concat}(\text{h}_1,\ldots,\text{h}_n)\mathbf{W}^O \tag{7}$$

where $\text{h}_i = \text{Attention}(\mathbf{QW}_i^Q, \mathbf{KW}_i^K, \mathbf{VW}_i^V)$.

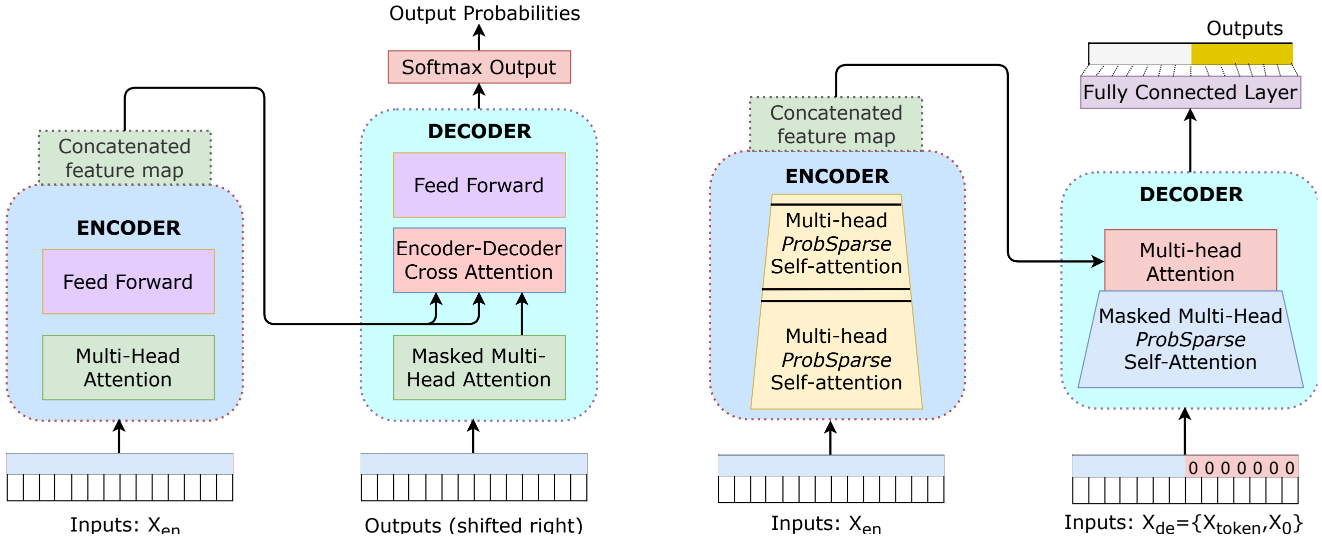
B. Informer

Informer [18] follows an architecture similar to that of the Transformer, with a specific focus on addressing LSTF problems. The characteristics unique to the Informer model, distinguishing it from the conventional Transformer are as follows:

1) Query sparsity measurement: Adhering to the Query-Key-Value model, Informer ranks queries based on a query sparsity measurement obtained through the Kullback-Leibler divergence equation :

$$S(q_i,K) = \ln\left(\sum_{j=1}^{L_k} e^{\frac{q_i \cdot k_j^T}{\sqrt{D}}} - \frac{1}{L_k}\sum_{j=1}^{L_k}\frac{q_i \cdot k_j^T}{\sqrt{D}}\right) \tag{8}$$

A high value of S for the ith query indicates a higher attention probability, suggesting a greater likelihood of containing dominant dot-product pairs.



(a) The Transformer-model architecture

(b) The Informer-model architecture

Fig. 1: Overview of Transformer and Informer model architectures

2) Probsparse attention: Utilizing the sparsity inherent in the self-attention mechanism to its advantage, Informer model uses Probsparse attention, in which each key attends to only $u = c.\ \ln L_q$ dominant queries.

$$\mathrm{Attn}_{Prob}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{Softmax}\left(\frac{\bar{\mathbf{Q}}\mathbf{K}^T}{\sqrt{D}}\right)\mathbf{V} \tag{9}$$

where $\bar{Q}$ is a sparse matrix and contains only the Top-u queries, under the query sparsity measurement S.

3) Self-attention distilling: A self-attention distillation operation is incorporated to extract the dominating attention scores by removing redundant information in the value matrix V. The distilling procedure involves forwarding the feature map from the M-th layer into the (M + 1)-th layer as follows:

$$X^t_{M+1} = \mathrm{MaxPoolELU}(\mathrm{Conv1d}([X^t_M]_{\mathrm{AT}})) \tag{10}$$

where $[.]_{\mathrm{AT}}$ represents the attention block and ELU(.) is the activation function. Max-pooling layer is added to reduce the overall memory usage.

4) Generative decoding: Employing generative-style decoding facilitates the generation of long sequence outputs in a single forward step, as opposed to relying on sequential inference. The final output is obtained through a fully connected layer, with its size determined by the type of forecasting, whether univariate or multivariate.

TABLE I: Computation statistics comparison for each layer: Probsparse vs. Canonical Scaled Dot-Product Self-Attention

	Training		**Testing**	
	Time	**Memory**	**Steps**	**Overall Memory**
Prob.	$\mathcal{O}(N \log N)$	$\mathcal{O}(N \log N)$	1	$\mathcal{O}((2-\epsilon)N \log N)$
Dot.	$\mathcal{O}(N^2)$	$\mathcal{O}(N^2)$	N	M.N^2

[1] *N- Input/output length.*

[2] *M- No.of encoder/decoder stacks*

In conclusion, Table I provides a comparative compute complexity order analysis emphasizing the computational efficiency of the self-attention and ProbSparse attention mechanisms. Fig. 1 depicts the architectural overview of both the Transformer and Informer models.

III. Experiment

A. Load Forecasting for EV charging stations

EV charging load forecasting can be conceptualized as a LSTF problem, essentially predicting future load values based on historical observations. Given the load data from past charging records $[c_{i,1}, \ldots, c_{i,t_0-2}, c_{i,t_0-1}] = c_{i,1:t_0-1}$ our goal is to forecast $[c_{i,t_0}, c_{i,t_0+1}, \ldots, c_{i,T}] = c_{i,t_0:T}$, where t_0 denotes the time point from which $c_{i,t}$ is assumed to be unknown at the prediction time. To enhance the process, a covariate matrix $X_{1:T}$ is assumed to be known which includes both time-independent factors (such as user ID, charging pile ID) or/and time-dependent variables (such as month of the year). Hence, the EV load forecasting problem can be reduced to modeling the subsequent conditional distribution, which comprises the product of T likelihood values.

$$\begin{aligned} &P(c_{i,t_0:T} \,|\, c_{i,1:t_0-1}, X_{1:T}; \boldsymbol{\theta}) \\ &\quad = \prod_{t=t_0}^{T} P(c_{i,t} \,|\, c_{i,1:t-1}, X_{1:t}; \boldsymbol{\theta}) \end{aligned} \tag{11}$$

where $\boldsymbol{\theta}$ denotes the set of trainable parameters of the time-series model.

B. Dataset

We perform our experiments on the real-world Adaptive Charging Network (ACN) dataset [19], a dynamic dataset of

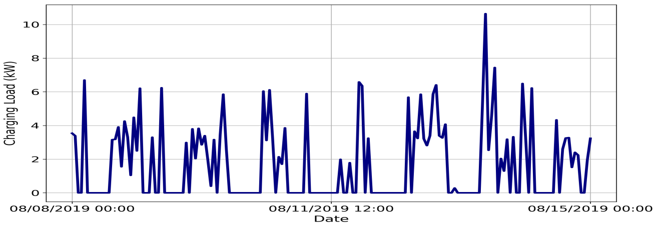


Fig. 2: 7-day charging load utilization at Caltech university

EV charging sessions collected from 54 EVSEs at Caltech University, California, to verify the effectiveness of both canonical dot-product self-attention and Probsparse attention. Although the charging site is accessible to the public, it primarily serves faculty, staff, and students. The dataset comprises information regarding the time of connection, disconnection, and total energy consumption for each charging session. We covert the given data into a new dataset wherein the average charging load (kW) is computed for each hour. The dataset includes 16489 data points from April 2018 - March 2020 with an average load of 1.92 kW and peak load of 10.63 kW. Fig. 2 depicts a 10-day charging profile extracted from the dataset.

C. Implementation Details

1) Hyperparameters: To facilitate the comparison, we incorporate both the Probsparse self-attention (Informer) model and the canonical self-attention variant (Informer†). For implementation, the input of the dataset is processed using standard normalization. The architecture contains a 2 encoder layers and 1 decoder layer. Adam optimiser is used with learning rate initialized from 1e^{-4}. The dataset is divided into training, validating and testing subsets by 0.7:0.1:0.2. We employ Random Search [19] to adjust the model's hyperparameters. The ranges for these hyperparameters are determined based on domain expertise, and ultimately, the selection is based on achieving the best performance. Table II presents a detailed overview of the hyperparameters used for the experiment.

2) Metrics: The evaluation metrics that we use include $MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2$ and $MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i|$. These metrics are computed for each prediction window, and the entire set is then rolled with a stride of 1 for evaluation.

TABLE II: Values of hyperparameters employed in the experiments

Hyperparameter	**Range**
Number of training epochs	1-50 **(6)**
Number of heads	[1, 2, 4, **8**]
Embedding dimension	[64, 128, 256, **512**, 1024, 2048]
No of encoder layers	[1, **2**, 3, 4, 5, 6]
No of decoder layers	[**1**, 2, 3, 4, 5, 6]
Feedforward network dimension	[64, 128, 256, 512, 1024, **2048**]
Padding	[**0**,1]
Dropout Percentage	0-0.8 **(0.05)**
Activation function	[ReLU, **GELU**]
Data sample batch size	[8, 16, **32**, 64, 128, 256]

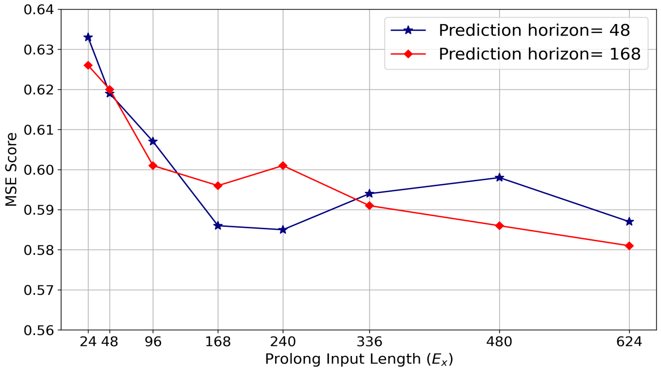


Fig. 3: Comparison of MSE scores for different input lengths in Informer model

3) Platform: The models are trained on six NVIDIA GeForce GTX 1080 GPU's.

IV. Results and Analysis

Table III presents the results of load variable predictions over time series. As a requirement of higher prediction capacity, we extend the prediction horizon progressively, keeping the encoder input length fixed. The results from Table III indicate a significant improvement in inference performance across the dataset with the Informer model. The best results are indicated in bold font. A lower MSE or MAE signifies improved prediction precision. The last column illustrates the percentage change in MSE of the Informer model in comparison to Informer† model.

The Informer model surpasses the canonical Informer† model in winning counts, with 8>2, thereby reinforcing the hypothesis of query sparsity assumption offering an equivalent attention feature representation. Moreover, it can be observed that as the prediction horizon increases gradually, the prediction error rises slowly and smoothly, improving prediction capacity in a LSTF problem such as EV charging load forecasting. Furthermore, Fig. 3 shows the variation of MSE with prolonging encoder length *(E_x)*, and decoder token *(D_{token})* in the Informer model. Here $\text{E}_x \in \{24, 48, 96, 168, 240, 336, 480, 624\}$ representing the *Conditioning range* $[1, t_0 - 1]$ of our EV load forecasting problem and $\text{D}_{token} \in \{24, 48, 48, 96, 168, 240, 336, 336\}$, indicating

TABLE III: Long-Sequence time-series forecasting results for prolonged prediction horizons

Models	**Informer**		**Informer†**		
Metric	**MSE**	**MAE**	**MSE**	**MAE**	**ΔMSE%**
24	**0.576**	**0.529**	0.600	0.544	-4.00
48	**0.581**	**0.533**	0.623	0.567	-6.74
168	**0.592**	**0.540**	0.639	0.595	-7.36
336	**0.612**	**0.568**	0.646	0.617	-5.27
720	0.687	0.657	**0.664**	**0.598**	+3.46
Count	**8**		**2**		

the integration of augmented local information into the decoder. The observed trend suggests that the Informer model demonstrates superior performance when forecasting long sequences. As it is evident from Fig.2, when predicting short sequences (like 48), the performance shows large variations before ultimately reducing the MSE. This reduction in error is likely due to the recurrent occurrence of short-term patterns. Conversely, the MSE shows significant reduction while predicting long sequences (like 168) with longer encoder inputs. This improvement can be attributed to the longer encoder input (conditioning range), which encompasses more dependencies, and the extended decoder token, which incorporates richer local context.

V. Conclusion and Future Work

Ensuring the resilience of an electric vehicle charging network depends significantly on accurate forecasts of charging demand. Utilizing Transformer neural network architecture proves ideal for EV load forecasting, offering parallelization of inputs, handling long-range dependencies, and scalability for larger datasets. This study presents Transformer-based models to address the challenge of electric vehicle (EV) charging forecasting. Furthermore, our comparative analysis showcases the superiority of a lighter ProbSparse attention mechanism, which not only streamlines computational complexity but also demonstrates superior generalization compared to its self-attention counterpart. The notable absence of overfitting underscores the efficacy of our model. Consequently, this model stands as a valuable asset for stakeholders such as EV charging station operators as well as power grid operators, facilitating efficient planning, maintenance, and operation of existing charging infrastructure, as well as aiding in the strategic development of new charging stations.

In future research, we aim to expand upon this study by integrating additional variables that influence EV charging behavior into our multivariate analysis. Given the high variability of EV datasets, we seek to investigate attention mechanisms capable of capturing sub-series periodicity more effectively. By doing so, we anticipate a deeper understanding of underlying patterns in EV charging data and thus, more accurate forecasting models.

References

[1] International Energy Agency (IEA), "Global EV Outlook 2023," IEA, Paris, Tech. Rep., 2023.

[2] K. Clement-Nyns, E. Haesen, and J. Driesen, "The impact of charging plug-in hybrid electric vehicles on a residential distribution grid," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 371–380, 2010.

[3] M. Islam, H. Lu, M. Hossain, and L. Li, "An iot- based decision support tool for improving the performance of smart grids connected with distributed energy sources and electric vehicles," 01 2020.

[4] M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, "Modified pattern sequence-based forecasting for electric vehicle charging stations," in *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2014, pp. 710–715.

[5] H. Louie, "Time-series modeling of aggregated electric vehicle charging station load," *Electric Power Components and Systems*, vol. 45, pp. 1–14, 12 2017.

[6] Q. Sun, J. Liu, X. Rong, M. Zhang, X. Song, Z. Bie, and Z. Ni, "Charging load forecasting of electric vehicle charging station based on support vector regression," in *2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, 2016, pp. 1777–1781.

[7] Y. Lu, Y. Li, D. Xie, E. Wei, X. Bao, H. Chen, and X. Zhong, "The application of improved random forest algorithm on the prediction of electric vehicle charging load," *Energies*, vol. 11, 2018.

[8] L. Buzna, P. De Falco, S. Khormali, D. Proto, and M. Straka, "Electric vehicle load forecasting: A comparison between time series and machine learning approaches," 05 2019, pp. 1–5.

[9] "Comparison of electric vehicle load forecasting across different spatial levels with incorporated uncertainty estimation," *Energy*, vol. 283, p. 129213, 2023.

[10] J. Zhu, Z. Yang, Y. Chang, Y. Guo, K. Zhu, and J. Zhang, "A novel lstm based deep learning approach for multi-time scale electric vehicles charging load prediction," 05 2019, pp. 3531–3536.

[11] L. Guo, P. Shi, Y. Zhang, Z. Cao, Z. Liu, and B. Feng, "Short-term ev charging load forecasting based on ga-gru model," 03 2021, pp. 679–683.

[12] M. Abumohsen, A. Y. Owda, and M. Owda, "Electrical load forecasting using lstm, gru, and rnn algorithms," *Energies*, vol. 16, 2023.

[13] 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," 2020.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023.

[15] S. Koohfar, W. Woldemariam, and A. Kumar, "Prediction of electric vehicles charging demand: A transformer-based deep learning approach," *Sustainability*, vol. 15, 2023.

[16] A. L'Heureux, K. Grolinger, and M. A. M. Capretz, "Transformer-based model for electrical load forecasting," *Energies*, vol. 15, 2022.

[17] X. Huang, D. Wu, and B. Boulet, "Metaprobformer for charging load probabilistic forecasting of electric vehicle charging stations," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, pp. 1–11, 10 2023.

[18] 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.

[19] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.