



Article

Prediction of Electric Vehicles Charging Demand: A Transformer-Based Deep Learning Approach

Sahar Koohfar, Wubeshet Woldemariam * and **Amit Kumar ***

Civil and Environmental Engineering, University of Texas at San Antonio, San Antonio, TX 78249, USA

* Correspondence: wwoldema@purdue.edu (W.W.); amit.kumar@utsa.edu (A.K.)

Abstract: Electric vehicles have been gaining attention as a cleaner means of transportation that is low-carbon and environmentally friendly and can reduce greenhouse gas emissions and air pollution. Despite EVs' many advantages, widespread adoption will negatively affect the electric grid due to their random and volatile nature. Consequently, predicting the charging demand for electric vehicles is becoming a priority to maintain a steady supply of electric energy. Time series methodologies are applied to predict the charging demand: traditional and deep learning. RNN, LSTM, and transformers represent deep learning approaches, while ARIMA and SARIMA are traditional techniques. This research represents one of the first attempts to use the Transformer model for predicting EV charging demand. Predictions for 3-time steps are considered: 7 days, 30 days, and 90 days to address both short-term and long-term forecasting of EV charging load. RMSE and MAE were used to compare the model's performance. According to the results, the Transformer outperforms the other mentioned models in terms of short-term and long-term predictions, demonstrating its ability to address time series problems, especially EV charging predictions. The proposed Transformers framework and the obtained results can be used to manage electricity grids efficiently and smoothly.

Keywords: electric vehicles; time series; machine learning; deep learning; ARIMA; SARIMA; RNN; LSTM; transformers



Citation: Koohfar, S.; Woldemariam, W.; Kumar, A. Prediction of Electric Vehicles Charging Demand: A Transformer-Based Deep Learning Approach. *Sustainability* **2023**, *15*, 2105. <https://doi.org/10.3390/su15032105>

Academic Editor: Lei Zhang

Received: 17 December 2022

Revised: 15 January 2023

Accepted: 16 January 2023

Published: 22 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In addition to being an integral component of the economic development and daily life in many countries, the transportation sector is also a major source of energy consumption. This sector consumes an enormous amount of non-renewable energy, which has a negative impact on the environment and contributes to global carbon dioxide emissions. Particularly, CO₂ emissions from transportation have made up over 25% of all emissions worldwide since 2016 [1]. In recent years, electric vehicles (EVs) have become more widespread in the transportation industry as a cleaner means of transportation that is low-carbon and environmentally friendly. Based on current trends, EV transportation is likely to replace internal combustion engine (ICE) cars in the near future. The EV share is anticipated to increase exponentially up to 13.4% in 2030 under a Sustainable Development Scenario (SDS) 2020–2030 [2]. Promoting energy security by diversifying energy sources, stimulating economic growth by establishing new innovative sectors, and, most critically, protecting the environment and fighting climate change by reducing tailpipe emissions and decarbonizing are some benefits of EV penetration [3].

Even though EVs have many advantages compared to ICE vehicles, widespread adoption of electric vehicles among the drivers will have negative impact on the electric grid. Due to the randomness and volatility of EV charging loads, the power grid will be impacted by increasing peak powers, frequency and voltage variation, and overall energy demand.

Under these circumstances, accurate EV charging load prediction is an essential foundation for evaluating the impact of the EVs on the power grid and planning and operating

a highly penetrable power system [4]. Moreover, estimating EV charging demand is critical for accurately forecasting the implications of additional demand on grid constraints. Forecasting EV charging demand successfully preserves grid utility stability and resilience while also assisting in long-term investment planning and resource allocation for charging infrastructure [5].

This research focuses on forecasting EV charging load demand by applying time series algorithms and proposing an optimal model, which can be used by electric vehicle power suppliers and energy management systems for coordinated EV charging based on real world EV charging and weather data collected in Denver, Colorado.

Literature Review

Due to the stochastic nature of the EV charging demand, it can be significantly affected by a wide range of factors, making accurate load modeling and load forecasting challenging. It is increasingly critical to coordinate the operation of energy and transportation systems to facilitate the development of the energy-transportation nexus towards a low carbon future in order to enhance the industrial and social economy while reducing greenhouse gas emissions. A suitable solution is urgently needed due to the widespread use of electric vehicles as renewable sources of energy, which poses a threat to the stability of the distribution network. Consequently, load forecasting plays a significant role in the planning, scheduling, and operation of the power system [6].

Utility decision-making and development depend on the accuracy of EV charging forecasts. As a result of implementation of a highly accurate forecasting model, not only will it enhance prediction precision for optimal dispatching, but it will also support the development of EV charging and provide manufacturers with a strong incentive to promote EV use. Previous research has revealed some unique techniques to monitor EV charging, and a range of forecasting methods have been identified for modeling EV charging demand [7].

Studies of load forecasting for EVs can be classified into two parts based on forecasting technique: traditional statistical time series approaches and deep learning and machine algorithms. Time series is a method that predicts the future value of a variable based on its past. Autoregressive integrated moving average (ARIMA) is one of the statistical time series models that has been widely applied for load forecasting purposes. Amini et al. applied ARIMA for forecasting daily charging demand of EV parking lots. By adjusting the integrated and auto-regressive order parameters, they improved the ARIMA forecaster's accuracy in order to minimize the mean square error (MSE). Moreover, the daily charging demand profile of EV parking lots was decoupled from the load profile's seasonal variation. In addition, they forecasted EV charging demand in the scheduling problem and illustrated better unit commitment along with a reduction in operation cost [8]. Nevertheless, seasonal load patterns, which are typical of EV chargers and of energy time series can be better caught through seasonal time series. Buzna et al. applied different time series and ML models for forecasting daily EV charging demand up to 28 days in the Netherlands and compared their performance. The results illustrated that Seasonal Autoregressive Integrated Moving Average (SARIMA) outperformed random forest (RF) and gradient boosting regression tree (GBRT) [9]. Based on two years of time-stamped aggregate power consumption data from 2400 charging stations located in Washington State and San Diego, Louie et al. proposed a time-series seasonal autoregressive integrated moving average model for forecasting EV charging station load [10]. Although statistical time series models have straightforward structures and require minimal training, they fail to capture the nonlinear nature of the load series.

In recent years, deep learning (DL), and artificial neural networks (ANNs), as an essential branch of ML, have developed rapidly and have gained many applications in EV load forecasting. A neural network can effectively address the shortcomings of a time series model by capturing features and forming nonlinear mapping relationships [11]. They are also capable of performing automatic representation learning from big data and have strong adaptation [12].

Among the huge variety of approaches, ANNs, recurrent neural networks (RNNs) and their popular variant including long-short term memory (LSTM) are widely used by researchers for EV forecasting problem. As reported in [13], authors employed ANN forecasting algorithms in the Building Energy Management System (BEMS) to predict the charging profiles of electric vehicles. Jahangir et al. [14] applied three different neural networks—a simple ANN, a rough artificial neural network (R-ANN), and a recurrent rough artificial neural network (RR-ANN), approach to forecast 24-h EV load on distribution system based on travel behavior. Result implied that the RR-ANN model outperformed other models and generated the most accurate prediction.

In order to overcome the vanishing gradient problem of original RNNs, Hochreiter et al. developed a long short-term memory, which is an improved form of RNN [15]. Chang et al. applied the LSTM approach to forecast aggregated charging power demand in a multiple fast-charging station in Jeju, Korea, and compared its performance to other counterparts. The LSTM model achieved good accuracy and showed the best performance among other used deep learning models [16]. An LSTM model was used by Marino et al. for predicting building energy loads using neuron node numbers [17]. The LSTM showed the greatest performance in a study by Kong et al. that applied it to the forecasting of residential load [18]. Similarly, Lu et al. applied different neural network models for hourly level aggregated EV load forecasting, and according to the backtesting results, LSTM outperformed other models [19].

The literature review showed that the LSTM, RNN, and ANN methods have been successfully used on EV load forecasting. However, they have several limitations due to the sequential processing of input data, especially when dealing with datasets with long dependencies [20]. In order to deal with the long-term dependencies problem in time series forecasting, transformer-based solutions have been developed.

Transformer is a class of machine learning models that use the scaled dot-product operation or self-attention as its main learning mechanism [21]. A transformer has widespread applications and has been applied to tackle various problems in machine learning particularly natural language processing (NLP), speed recognition, and motion analysis, and has achieved state-of-the-art performance [22–24]. Recently, the attention mechanism has gained popularity in the time series task as well.

This literature review shows the lack of research on forecasting the EV charging using attention-based mechanisms despite its ability to outperform recurrent neural networks. This paper fills the gap in the existing literature by applying Transformer as a new method in forecasting charging demand of the EV by considering historical real-world data, weather, and weekend data in Boulder Colorado.

The paper studies attention-based mechanism to precisely forecast EV to generate more realistic results. The performance of Transformer, the state-of-the-art forecaster, is compared with ARIMA, SARIMA, RNN, and LSTM models, which are the main benchmarking methods in this field.

2. Materials and Methods

2.1. Forecasting Methods

This paper uses five models for forecasting EV energy load consumption. These models are Autoregressive Integrated Moving Average (ARIMA), Seasonal Autoregressive Integrated Moving Average (SARIMA), Recurrent Neural Network (RNN), Long short-term memory networks (LSTM), and Transformer. The following sections describe each model.

2.1.1. Autoregressive Integrated Moving Average (ARIMA) Model

An ARMA (p, q) model, which combines AR (p) and MA (q) models, is appropriate for modeling single-variable time series [25]. The future value of a variable is assumed in the ARIMA model to be a linear function of a number of prior observations and random errors. The ARIMA model assumes that the datasets are stationary, having a mean and variance

that remain constant across time. The following formula represents the ARIMA order (p, d, q) :

$$y = c + \varepsilon_t + \sum_{i=1}^p \varphi_i y_{t-i} + \sum_{j=1}^q \theta_j y_{t-j}, \quad (1)$$

where y_t and ε_t represent the actual value and random error (or random shock), respectively, at time interval t . θ_j and φ_i represent the autoregressive parameters, and c indicates a constant. p represents the model's order, d is the number of differentiations passes, and q is the number of moving averages, respectively.

2.1.2. Seasonal Autoregressive Integrated Moving Average (SARIMA)

Most of the time series records display a monthly or annual seasonal pattern. In order to cope with the seasonal data, Box and Jenkins developed an extension of ARIMA, which is called SARIMA. There are two types of variations in a seasonal time series: the first type is between consecutive observations, and the second type is between pairs of corresponding observations belonging to consecutive seasons. ARIMA (p, d, q) models can demonstrate the relationship between corresponding observation values of successive seasons, whereas Seasonal ARIMA $(p, d, q) \times (P, D, Q)_s$ models depict the relationship between corresponding observation values of consecutive seasons. In the Seasonal ARIMA, S denotes the number of the periods in each season, and the lowercase p, d , and q represent the autoregressive, differencing, and moving average terms for the nonseasonal part of the ARIMA model, and the uppercase P, D , and Q refer to the autoregressive differencing, and moving average terms for the seasonal part. Let L be the lag operator where:

$$L_{y_t}^i = y_{t-i} \quad (2)$$

The lag operator polynomials are:

$$\phi(L) = 1 - \phi_1 L - \phi_2^2 L^2 - \dots - \phi_P^P L^P, \quad (3)$$

$$\theta(L) = 1 - \theta_1 L + \theta_2^2 L^2 + \dots + \theta_P^P L^P, \quad (4)$$

$$\Phi(L) = 1 - \Phi_{p1} L^{p2} - \Phi_{p2}^2 L^{p2} - \dots - \Phi_{ps} L^{ps}, \quad (5)$$

$$\Theta(L) = 1 + \Theta_{q1} L^{q2} + \Theta_{q2}^2 L^{q2} + \dots + \Theta_{qs} L^{qs}, \quad (6)$$

$$\phi(L) = 1 - \phi_1 L - \phi_2^2 L^2 - \dots - \phi_P^P L^P \quad (7)$$

2.1.3. Artificial Neural Network (ANN)

A neural network composed of multiple layers of neurons interconnected among themselves is called an artificial neural network (ANN). Neural networks have an input layer that contains input values, a hidden layer that transforms those values, and an output layer that produces output values. These layers are connected together using weights. ANNs have many different classes and can be used for a wide variety of purposes [12]. There is a particular class of ANN, called a Recurrent Neural Network (RNN). RNNs have recurrent connections in their structure, which means that their current output is also reliant upon the previous output; more specifically, the network will memorize previous results to use them in calculating their current output. The difference between ANN and RNN's structure is shown in Figure 1:

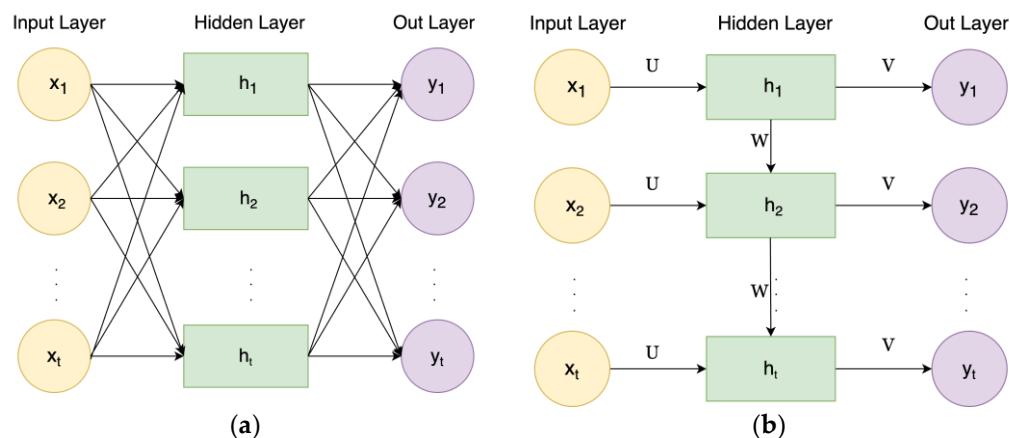


Figure 1. (a): ANN Architecture, (b): RNN Architecture.

As shown in Figure 1, despite ANN, RNN has recurrent connections between hidden layers. RNNs consist of multiple layers, where each layer is composed of one or more neurons that connect to the previous layer as well as the next layer in sequence. As for ANN, the information travels from one layer to another without touching a node twice.

Mathematically, for time t , RNN formula is:

$$h_t = f(Wh_{t-1} + Ux_t + b), \quad (8)$$

$$y_t = g(Vh_t) \quad (9)$$

In this equation, f and g are the non-linear activation function (sigmoid function), h_t and x_t are respectively the hidden layer and input layer in time step t , and W , U , and V are the weight matrices. y_t is the output, and b denotes the bias.

2.1.4. Long Short-Term Memory (LSTM)

Conventional RNNs are only capable of memorizing short time series data. They lose the important information of long-term input as data and time steps increase, causing vanishing gradients or expanding gradients. In order to overcome this problem, Long short-term memory (LSTM) was proposed by Horchreiter and Schmidhuber in 1997 [15], whose structure is shown in Figure 2. In LSTM networks, such problems are solved by incorporating specific gate mechanisms into the recurrent feedback loops.

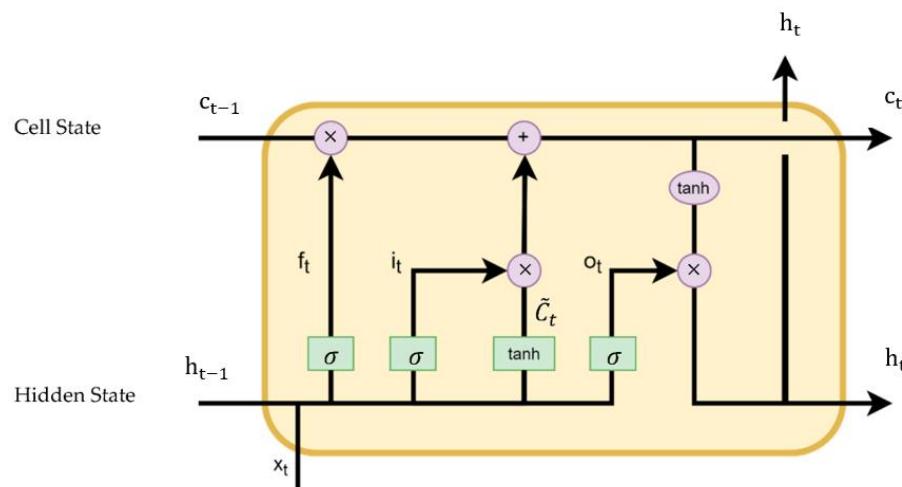


Figure 2. LSTM Architecture.

During the learning process, the neuronal network's gating mechanism determines what information to keep and what information to discard. The LSTM structure contains three gates: the forget gate (f), which decides what information is going to be thrown away from the cell state, the input gate (i), which decides which information should be stored in the cell state, and the output gate (o), which decides what parts of the cell state are going to be produced as output.

The formulation of updating the cell states and parameters can be written as:

$$f_t = \sigma(x_t U^f + h_{t-1} W^f), \quad (10)$$

$$i_t = \sigma(x_t U^i + h_{t-1} W^i), \quad (11)$$

$$\tilde{C}_t = \tanh(x_t U^g + h_{t-1} W^g), \quad (12)$$

$$C_t = \sigma(f_t \times C_{t-1} + i_t \times \tilde{C}_t), \quad (13)$$

$$h_t = \tanh(C_t) \times o_t \quad (14)$$

In the LSTM formula, i_t , f_t , and o_t are input gate, forget gate, and output gate at time t , respectively. x_t stands for input data at time step t , and h_{t-1} represents the hidden layer in time t . C_{t-1} and C_t are cell states in time $t - 1$ and t , respectively, and \tilde{C}_t denotes the internal memory unit. W , and U are the weight matrices associated to the corresponding gates.

2.1.5. Transformer

Vaswani et al. published the first transformer architectures in 2017 [21]. By implementing attention and self-attention mechanisms, this architecture eliminates the need for recurrent neural networks [26]. The Transformer model can improve time series forecasting accuracy through the use of an attention mechanism inspired by human perception, which enables the model to selectively focus on certain things while ignoring others. To sum up, attention is best utilized by capturing the core information in each task from a large amount of information. Unlike recurrent networks, the Transformer does not suffer from vanishing gradients and can access any point in the past no matter how far time steps are. As a result of this feature, the Transformer can identify long-term dependencies. Moreover, unlike recurrent networks that require sequential computation, the Transformer can also run completely in parallel at high speeds.

The Transformer model uses an encoder–decoder structure, which consists of stacked encoder and decoder layers. Input sequences can be mapped into output sequences with varying lengths using the encoder–decoder structure of the model. The encoder receives the input, a sequential sequence of time series data (x_1, \dots, x_n) , and transfers it into a continuous representation (z_1, \dots, z_n) . Based on z , the decoder calculates the output and then generates one symbol at a time (y_1, \dots, y_m) . The Transformer model is auto regressive, which means it uses the previous generated symbols as additional inputs. The output layer y_i at time step i is written as:

$$y_i = \sum_{j=1}^n a_{ij} (x_j W_V) \quad (15)$$

where y_i is the updated x_j , and a_{ij} is the attention score, measuring the similarity between x_i and x_j , which is calculated as:

$$a_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^n \exp(e_{ik})} \quad (16)$$

where e_{ij} measures the combability of two linearly transformed input elements, x_i and x_j :

$$e_{ij} = \frac{(x_i W_Q)(x_j W_K)^T}{\sqrt{h}}, \quad (17)$$

where W_K , W_Q , and W_K are three linear transformation matrices to increase the Transformer's expressiveness, and h is the dimension of the model. Figure 3 shows the structure of the Transformer model:

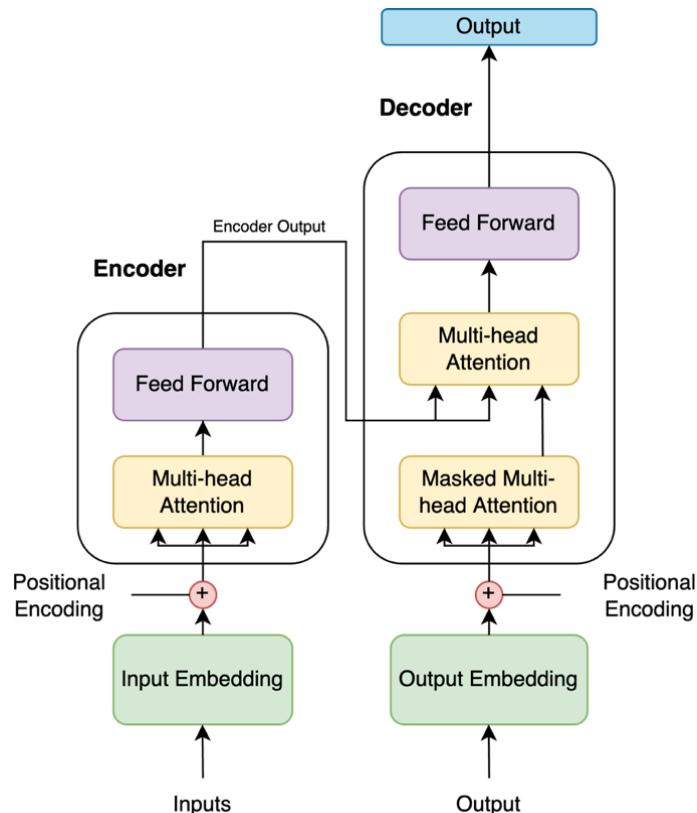


Figure 3. Transformer Architecture.

As shown in Figure 3, the encoder layer contains of two sub-layers: first and second fully connected multi-head self-attention mechanism and Feed Forward, respectively. Similarly, encoder layer can be deconstructed into three sub-layers. Masked Multi-head Attention, Multi-head Attention, and Feed Forward mechanism are three sub-layers of decoder. A masked Multi-head Attention ensures that no subsequent positions are involved in the predictor input. The Transformer uses residual connections surrounding each of the sublayers, followed by layer normalization, to accelerate training speed and convergence.

2.2. Data Analysis

An initial data analysis was performed before forecasting EV charging demand. In order to accurately forecast EV demand, we use charging load and weather data ranging from 1 January 2018 to 28 December 2021. Real-world EV charging load data from Boulder, Colorado was collected from 25 public charging stations, containing type 2 connectors of 22 kW. The EV charging data were obtained from the Colorado webpage, which contains historical data of electric charging facilities in Colorado [27]. The raw data of individual EV charging session such as type of the plug, address, arrival and departure time, date, and energy consumption in kW are available on the website for each charging record.

For this paper's objective of forecasting aggregated EV charging demand, raw individual EV charging session data must be transformed into an EV charging demand profile (in kW). In order to do that, the charging data were aggregated per day for all the charging facilities, and all the missing and negative data were removed from the dataset. The final

dataset contains 1425 records, which were fed to the machine learning models. Table 1 presents a statistical summary of the EV charging data in aggregated and disaggregated type.

Table 1. Statistical Summary of the EV Charging Data.

EV Stations	Type	Number of Record	Energy Usage
			Min
			Max
25	Disaggregated (EV charging Event)	29,780	0.001 (kW/Event)
	Aggregated (Daily EV charging)	1425	0.74 (kW/Day) 531.6 (kW/Day)

To develop the machine learning models, we divided the dataset into training and testing according to the standard 80/20 rule. The data of the first 38 months were used for training the models and those of the remaining nine months were used for testing. The range of test and training data is shown in Figure 4.

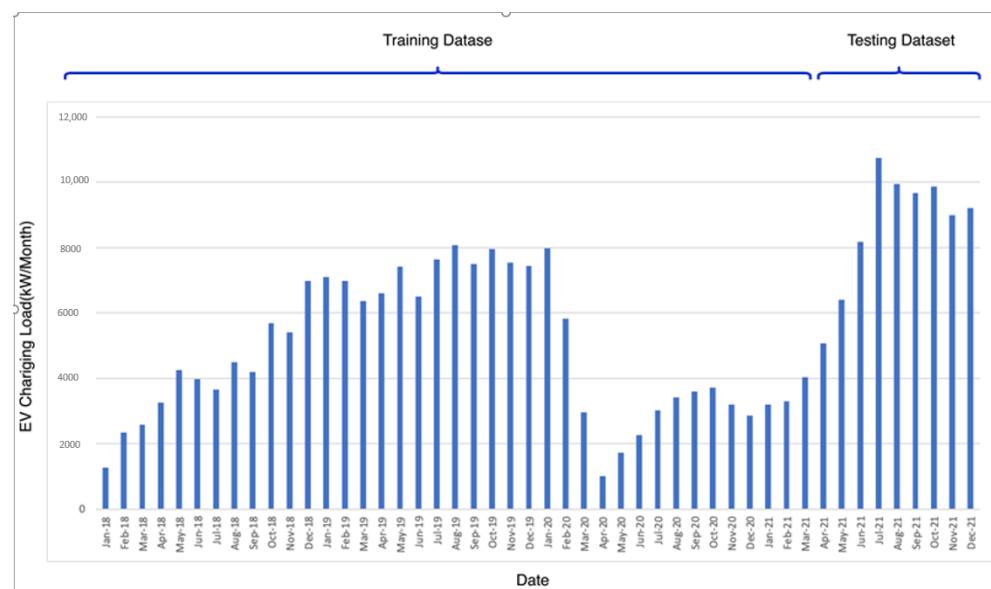


Figure 4. Training and Testing Range.

Next, the dataset was normalized before being fed into the models. Data were normalized using min–max normalization. The formula of min–max normalization is given by:

$$x_N = \frac{x_t - x_{\min}}{x_{\max} - x_{\min}}, \quad (18)$$

where x_N is normalized data, x_t is non-normalized data, and x_{\max} and x_{\min} are the maximum and minimum values of the entire dataset.

A time series plot was created for the EV charging load data to identify the overall pattern. From the start datapoint date to the summer of 2020, the charging load for EVs increases exponentially before starting to decline until the summer of 2021. Following that, the trend shows an increase in the overall EV charging load. The COVID-19 outbreak and accompanying quarantine, which reduced travel demand and EV charging load, was responsible for abrupt reducing of the data. In Figures 5 and 6, time series patterns are displayed. The dataset's decomposed pattern clearly shows a seasonal trend, and the residual pattern suggests that the data are non-stationary.

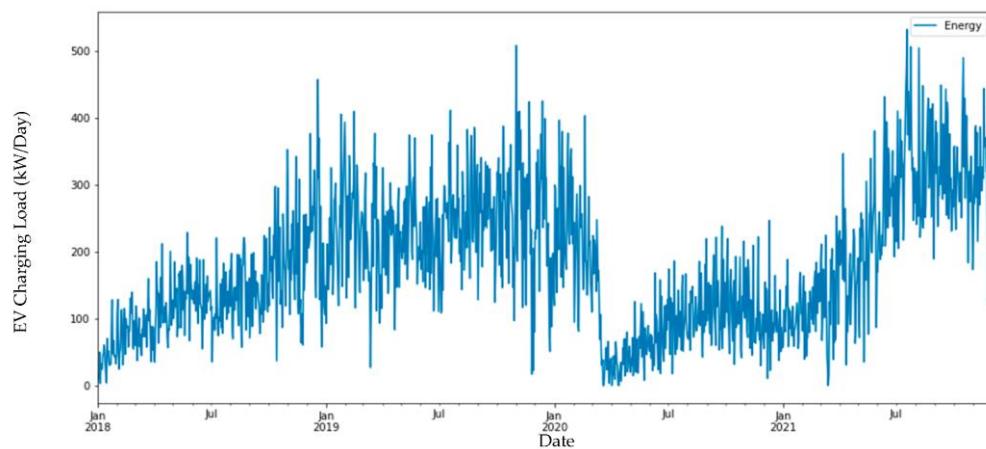


Figure 5. Time Series Plot of the EV Charging Data.

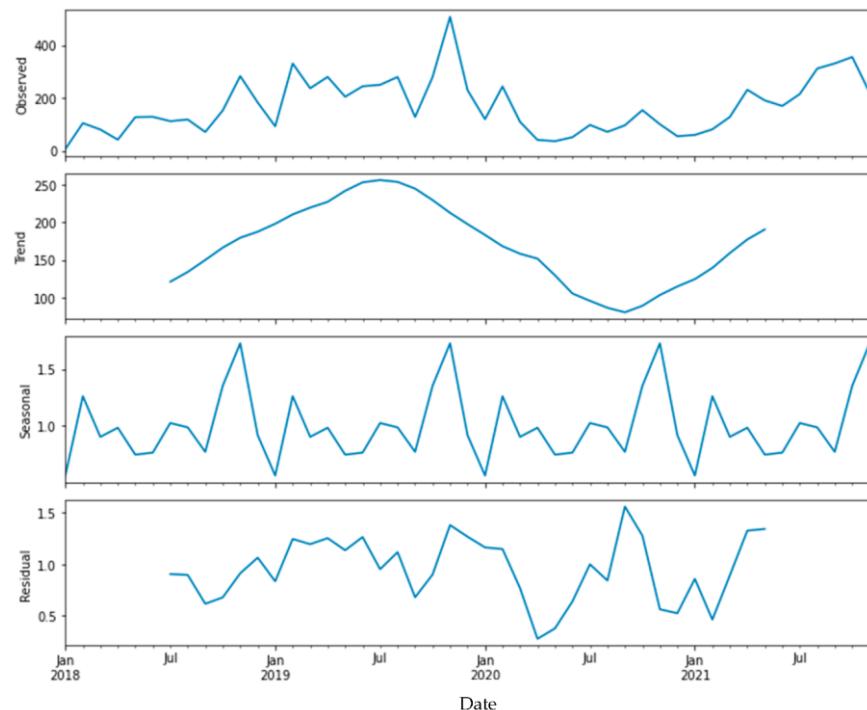


Figure 6. EV charging data Decomposition Pattern.

We use two different type of time series models for this paper, denoted as statistical and neural network models. ARIMA and SARIMA are the statistical time series, whereas RNN, LSTM, and Transformer are the deep learning time series approaches. EV charging demand, weekend, and weather data were applied as input features. We collected weather data of Boulder, Colorado form National Weather Service website. Weather data include maximum and minimum temperature, precipitation, and snow condition. The list of the features is given in Table 2.

Table 2. List of Dependent and Independent Variables.

Variable	Feature	ARIMA	SARIMA	RNN	LSTM	Transformers
Load Calendar	EV charging load (kW/day)	+ *	+	+	+	+
	Binary weekend (0 or 1)			+	+	+
	Max temperature (°F)					
Weather	Min temperature (°F)			+	+	+
	Snow (mm/day)					
	Precipitation (mm/day)					

* indicates the applied dataset in models.

By analyzing weather data, we can find that seasonal changes can noticeably affect the EV charging load. The dry season starts from September to March, and March to September is considered rainy season. Figure 7 shows the box plot of average monthly charging load distribution. In the dry season, the peak load, median load, upper quantile, and lower quantile are all higher than those of the rainy season. People tend to travel more during the dry season and consume more power, which results in a greater number of vehicles being charged.

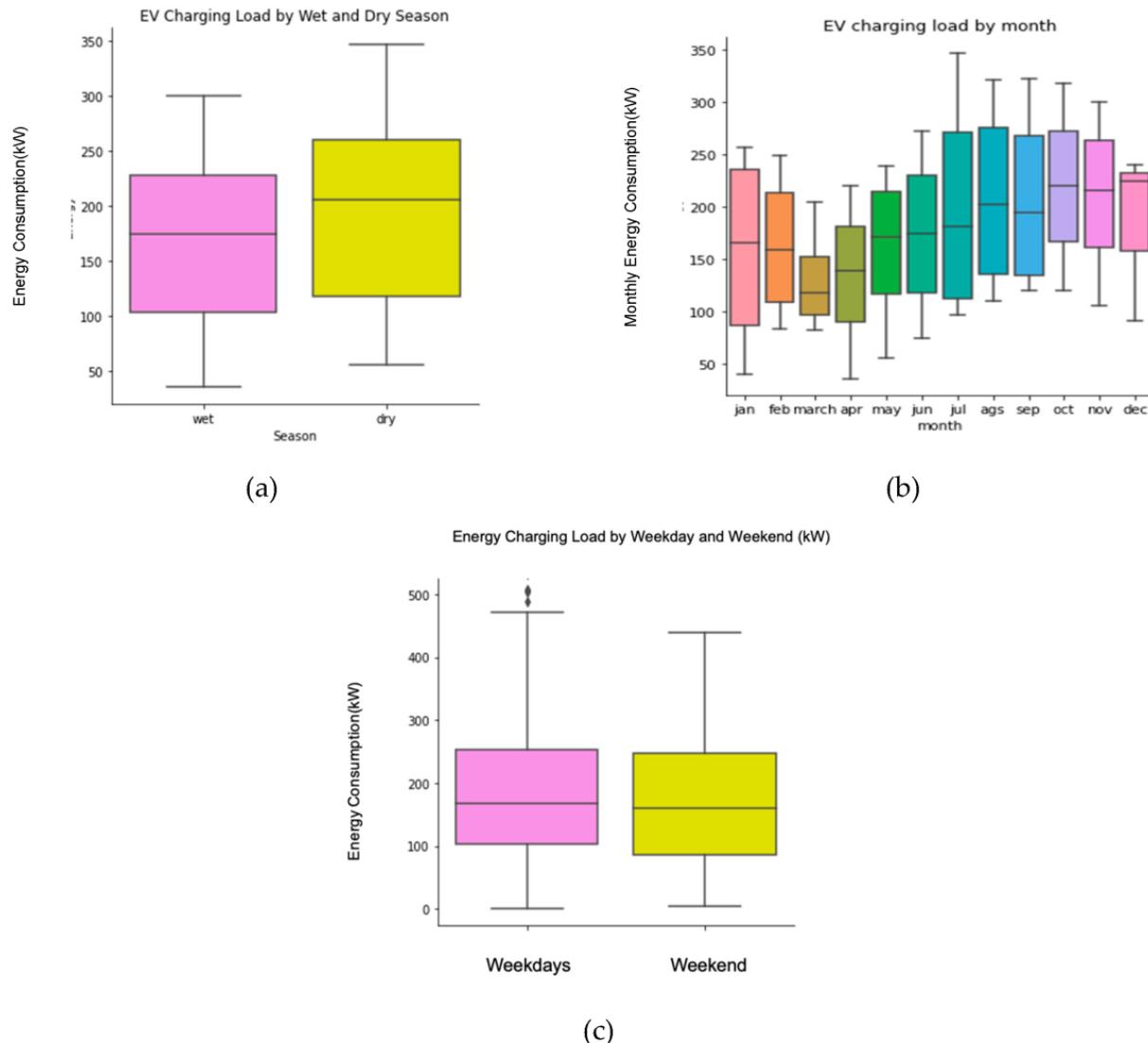


Figure 7. Box Plot of EV Charging Load, (a) EV Charging load by Seasons, (b) EV Charging Load by Month, (c) EV Charging Load by Weekend and Weekday.

Box plot of charging load based on weekday and weekend is shown in Figure 7. Analyzing charging load records by calendar data exhibits a slight difference between weekdays and weekends charging demand. It is observed that the median load on the weekdays, as well as the upper and lower loads, are higher than on weekends. Similarly, the plot shows a higher peak demand on weekdays compared with weekends. Considering the data are recorded at public charging stations, it can be explained by the fact that people typically charge their vehicles at public charging stations during working hours. In contrast, people typically use residential charging locations during weekends.

2.3. Performance Measurement

There are number of performance measures that are used for evaluating model accuracy. Among the commonly used error evaluation functions, we apply root mean squared error (RMSE) and mean absolute error (MAE) for evaluating model performance in load forecasting.

RMSE enables us to penalize outliers and clearly interpret the forecasted output, as they are in the same unit as the feature that the model is predicting. The equation of *RMSE* is:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (19)$$

We apply *MAE* as the second performance measurement mostly as a means of enhancing and confirming the confidence in the values obtained. Its formula is as follow:

$$MAE = \frac{1}{N} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (20)$$

where N is the number of the sample, y_i is the actual value and \hat{y}_i is forecasted value. A lower value of *RMSE* and *MAE* indicates better prediction performance.

3. Results and Discussion

ARIMA, SARIM, RNN, LSTM, and Transformer models were tested based on a dataset of 1425 points obtained over four years. In this study, four different time intervals were applied for forecasting EV charging load, from 7 days (short-term) to 1, 2, and 3 months (long-term). The different forecast horizons reflect different uses of the models, as follows: EVCS operators can use the 1, 2, and 3-month horizons for forecasting their system usage. As a result of the short horizons, energy providers can plan and optimize their short-term energy consumption in order to meet demand with clean energy.

For ARIMA and SARIMA, we train the model using target variable, which is EV charging load. For training deep learning models, Exogenous mentioned variables were used.

In the experiment, we implemented LSTM and RNN models in Keras deep learning framework and Transformer in Pytorch [28]. To facilitate the training process, the number of the hidden units and batch for the deep learning approaches are generally set as the power of 2. The deep learning model's hyperparameters are shown in Table 3.

Table 3. Applied Hyperparameters for Deep Learning Models.

Hyper Parameters	Value
Hidden Dimension	128
Number of Epoch	100
Number of Layer	1
Number of Head	8

The number of the hidden dimension in proposed models is set as 128, and the batch size is set as 64. For total number of 100 epochs, the training loss of neural network methods

is exhibited in Figure 8. The results indicate that Transformer achieves lower training loss and converges faster.

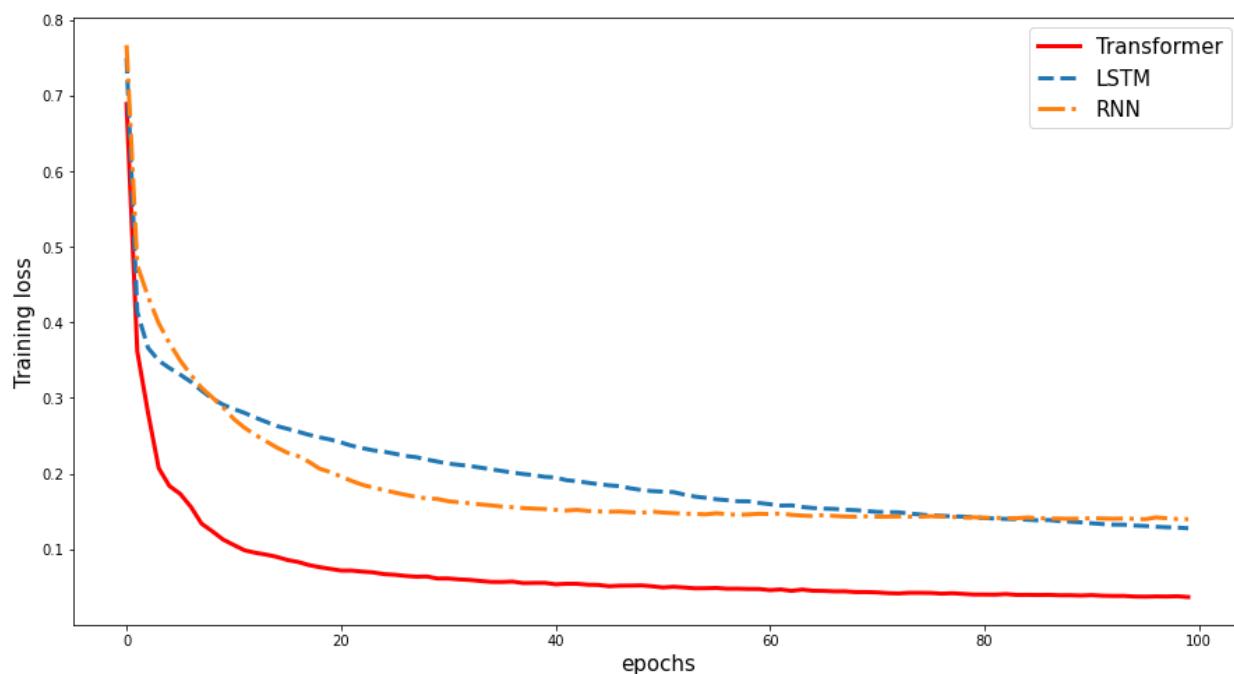


Figure 8. Training loss of Deep Learning Models.

A histogram of metrics and error comparisons is shown in Figure 9 for four scenarios. Based on the histograms, we can clearly see that the Transformer has the lowest error compared to the other four methods.

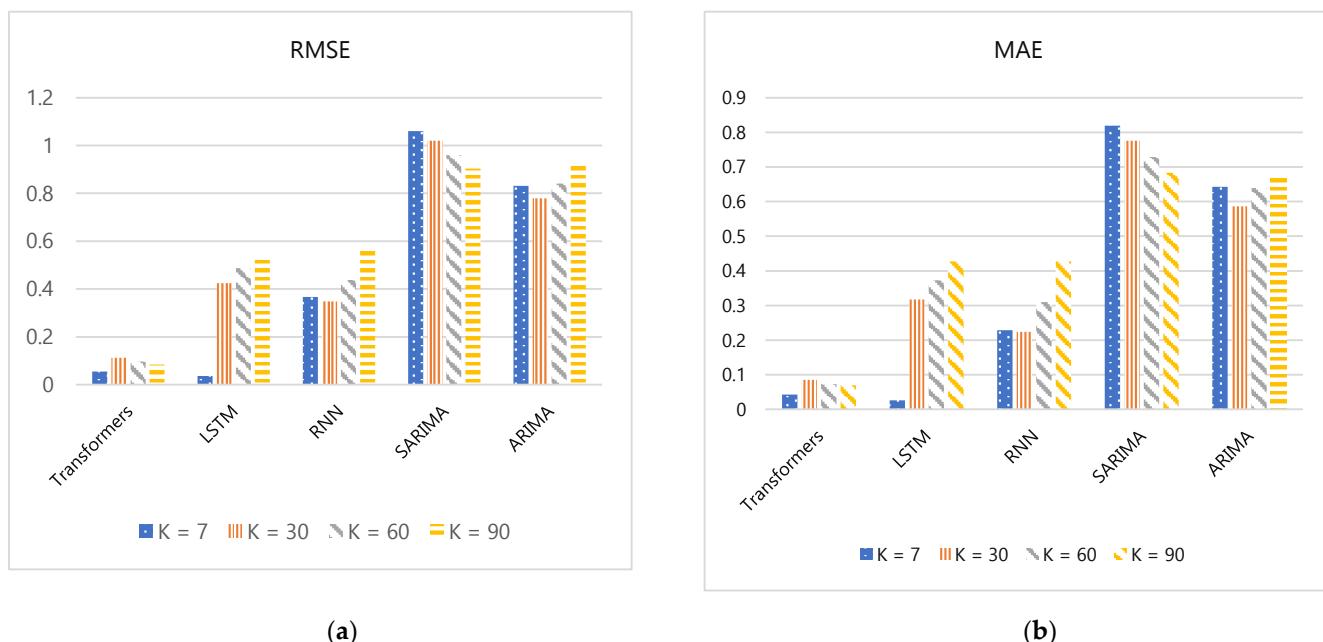


Figure 9. Bar Chart of Performance Measures of Proposed Models. (a) RMSE, (b) MAE.

In the next step, we evaluate and compare the performance of the trained models using the test set. The results for the performance accuracy of applied models are reported in Table 4, which indicates that the closer to zero the better, regardless of the evaluation indicator.

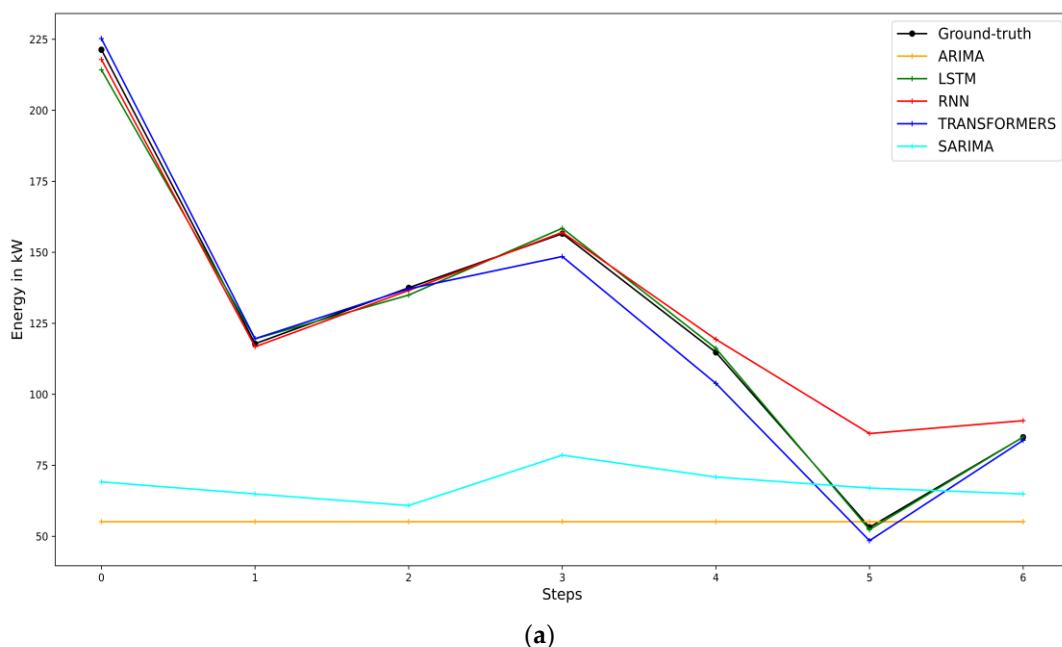
Table 4. Model Performance for Different Time Steps.

Steps Ahead (k)	Transformer		LSTM		RNN		SARIMA		ARIMA	
	RMSE	MSE	RMSE	MSE	RMSE	MSE	RMSE	MSE	RMSE	MSE
K = 7	0.055	0.043	0.036	0.026	0.367	0.229	1.06	0.819	0.831	0.643
K = 30	0.112	0.085	0.425	0.317	0.348	0.224	1.02	0.776	0.779	0.586
K = 60	0.096	0.073	0.488	0.373	0.438	0.310	0.960	0.729	0.841	0.639
K = 90	0.085	0.070	0.522	0.427	0.564	0.427	0.903	0.683	0.920	0.680

Among all the models assessed in Table 4, Transformer achieved the best prediction for 30, 60, and 90 days ahead. LSTM and Transformer results for 7 days ahead are comparable, as shown in Table 3. It is because LSTM is like Transformer when the prediction length is short. For 30, 60, and 90 days ahead, Transformer outperforms the next best model by 62%, 78%, and 84%, respectively. A key factor contributing to the model's higher performance is the ability of Transformer to incorporate any observation of the series (potentially skipping over non-relevant data points), which renders them for capturing similarities over more extended periods. These similarities are critical for accurate forecasting.

Despite the inferior results of LSTM and RNN when compared to Transformer, both models exhibit similar performance accuracy for long-term prediction. Moreover, they outperform ARIMA and SARIMA models in all prediction steps, which indicates the superior performance of deep learning methods over traditional statistical time series approaches. By looking at the results of Table 4, we notice that the Transformer model outperforms traditional time series methods, ARIMA and SARIMA, which are widely applied for addressing the EV charging load forecasting in both long-term and short-term prediction. It has been shown that the forecasting quality of ARIMA and SARIMA models, which predict the future value only based on the variable of interest without considering any complementary features, is significantly lower than that of Transformer, RNN, and LSTM models.

A histogram of metrics and error comparisons is shown in Figure 10 for four scenarios. Based on the histograms, we can clearly see that the Transformer model has the lowest error compared to four other methods.

**Figure 10. Cont.**

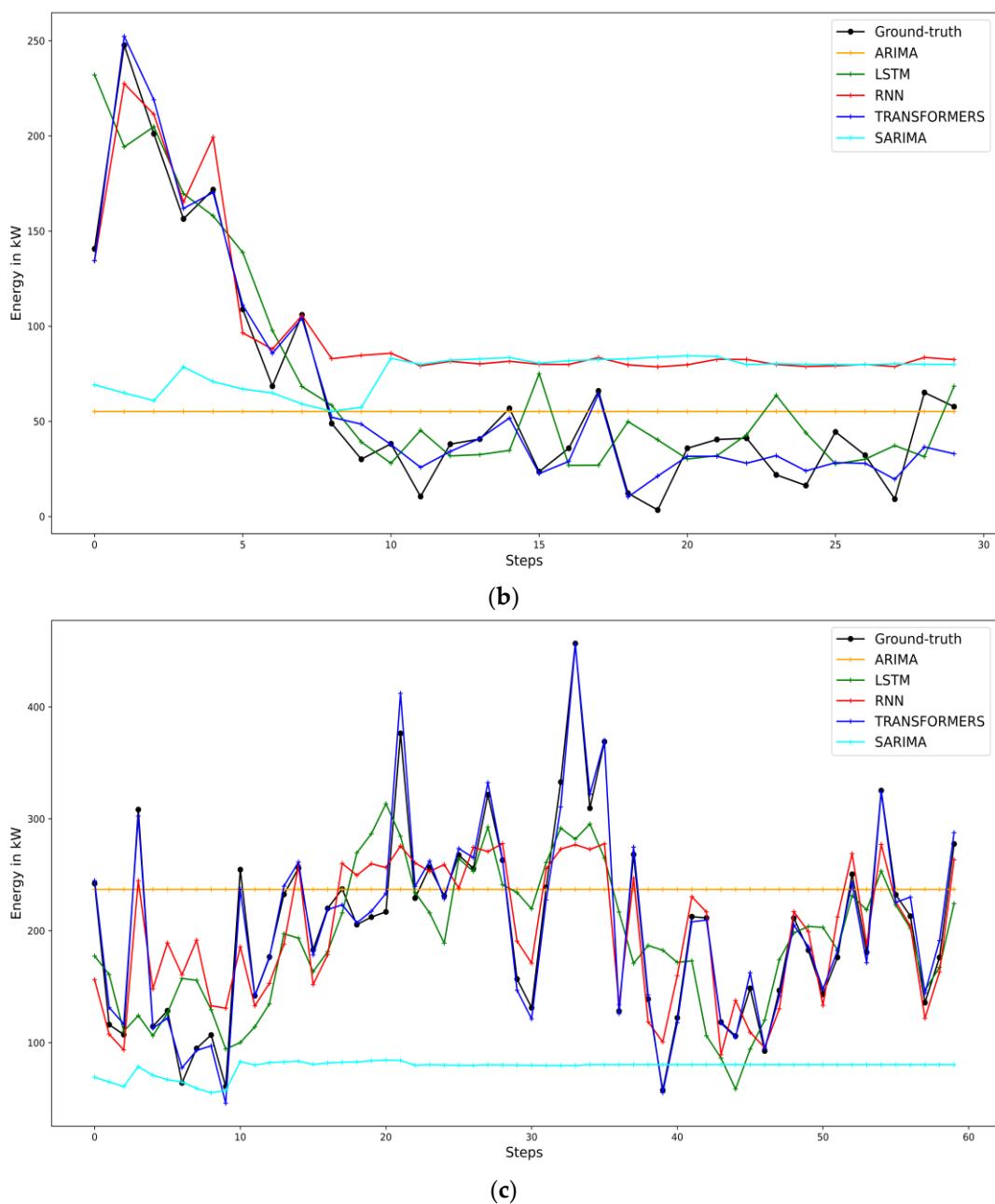


Figure 10. Cont.

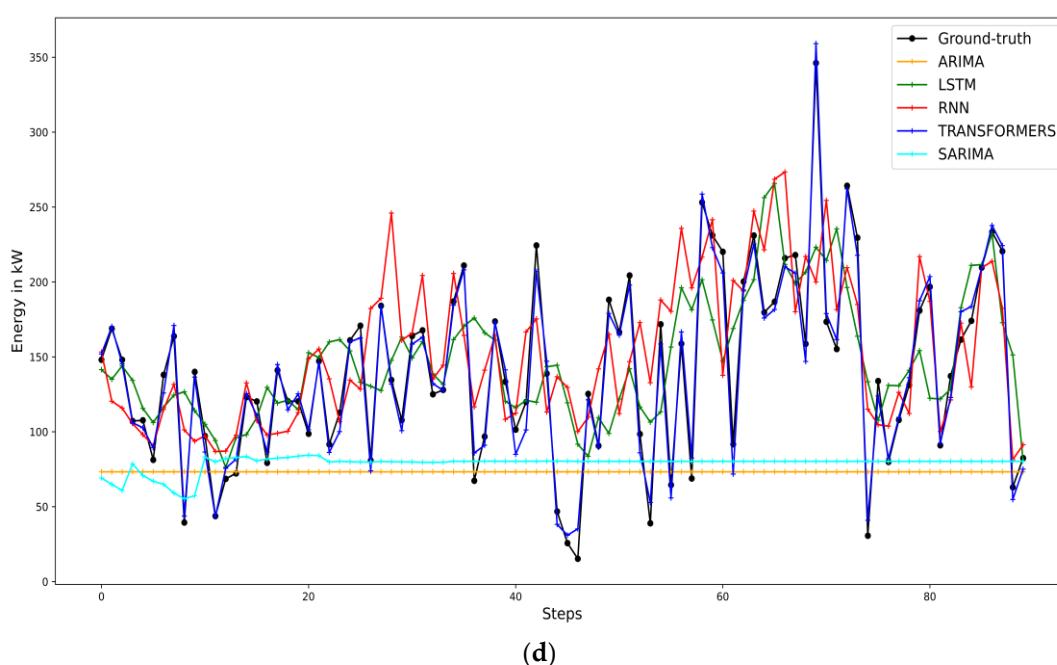


Figure 10. Forecasting Curve Comparison Graph. (a) 7-days ahead, (b) 30-days ahead, (c) 60-days ahead, (d) 90-days ahead.

On Figure 10, comparisons between the original test data and the predicted values converted into the original units are shown. The figure shows that the Transformer curve fits the real-world value precisely under each scenario and has perfectly predicted the load for each time step. For long-term horizons, both LSTM and RNN capture the general trend of the data, but still differ ominously from the test data. Traditional time series models, ARIMA and SARIMA, failed to capture fluctuation of time series data and created a straight-line prediction, which indicates the inability of traditional models compared with deep learning methods to predict time series dataset.

4. Conclusions

This article employs different time series methods to forecast EV charging load demand using historical real-world EV charging records of 25 public charging stations located at Boulder, Colorado for both short-term and long-term periods. This work is the first to use the Transformer model to address the EV charging load demand problem, and we benchmarked Transformer using ARIMA, SARIMA, LSTM, and RNN and compared its forecasting performance. According to the results, Transformer provided the best long-term prediction performance when compared to the time series and ML models stated above, with RMSE values of 0.085, 0.096, and 0.112 for $k = 30, 60$, and 90, respectively. Even while the model's performance for short-term prediction is slightly behind that of LSTM, it still performs much better than the other models indicated.

Upon reviewing the results, it is apparent that the Transformer model has excellent generalization capabilities, which can be applied effectively to EV charging record datasets. In addition, the Transformer model shows a greater ability to adjust to changes in the characteristics of EV records when compared with the LSTM and RNN models.

Due to the absence of a continuous hourly dataset in this project, the demand data are aggregated by day, which reduces the size of the training data for our proposed model and disables its capacity to forecast for shorter time steps (e.g., 15 min, 1 h). It is advised to use a larger real dataset of EV charging records with a smaller time step, such as a residential charging dataset in order to capture the EV charging demand more precisely and enhance the model's performance.

In this study, weather data and weekday datasets are utilized as independent variables for EV charging demand forecasting. Other related dataset, such as traffic distribution, which can affect the charging behavior of EVs, can be considered for future predication.

The primary objective of this research was to propose Transformer, a state-of-the-art deep learning method, for forecasting EV charging demand for the first time and compare its performance with models that are highly used by other studies, such as ARIMA, SARIMA, LSTM, and RNN. Applying other regression models, deep learning, and neural networks for future analysis is recommended.

The outcome of this study is beneficial for utility operators managing the operation of electric profiles in future power systems, by employing more precise models to forecast EV charging demand. This research also may help with the selection of investment and management strategies for flexible EV charging infrastructures based on EV charging demand.

Author Contributions: Conceptualization, S.K. and A.K.; methodology, S.K.; software, S.K.; validation, S.K., W.W.; formal analysis, S.K.; investigation, S.K.; data curation, S.K.; writing—original draft preparation, S.K.; writing—review and editing, W.W.; supervision, A.K., W.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: https://webappsprod.bouldercolorado.gov/opendata/ev_datadictionary.csv (accessed on 16 December 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zheng, X.; Streimikiene, D.; Balezentis, T.; Mardani, A.; Cavallaro, F.; Liao, H. A review of greenhouse gas emission profiles, dynamics, and climate change mitigation efforts across the key climate change players. *J. Clean. Prod.* **2019**, *234*, 1113–1133. [[CrossRef](#)]
2. Global EV Outlook 2021. 2021. Available online: <https://www.iea.org/reports/global-ev-outlook-2021> (accessed on 22 January 2022).
3. Yong, J.Y.; Ramachandaramurthy, V.K.; Tan, K.M.; Mithulananthan, N. A review on the state-of-the-art technologies of electric vehicle, its impacts and prospects. *Renew. Sustain. Energy Rev.* **2015**, *49*, 365–385. [[CrossRef](#)]
4. Dubey, A.; Santoso, S. Electric vehicle charging on residential distribution systems: Impacts and mitigations. *IEEE Access* **2015**, *3*, 1871–1893. [[CrossRef](#)]
5. Moon, H.; Park, S.Y.; Jeong, C.; Lee, J. Forecasting electricity demand of electric vehicles by analyzing consumers' charging patterns. *Transp. Res. Part D Transp. Environ.* **2018**, *62*, 64–79. [[CrossRef](#)]
6. Zhu, J.; Yang, Z.; Mourshed, M.; Guo, Y.; Zhou, Y.; Chang, Y.; Wei, Y.; Feng, S. Electric vehicle charging load forecasting: A comparative study of deep learning approaches. *Energies* **2019**, *12*, 2692. [[CrossRef](#)]
7. Caliwag, A.C.; Lim, W. Hybrid VARMA and LSTM method for lithium-ion battery state-of-charge and output voltage forecasting in electric motorcycle applications. *IEEE Access* **2019**, *7*, 59680–59689. [[CrossRef](#)]
8. Amini, M.H.; Kargarian, A.; Karabasoglu, O. ARIMA-based decoupled time series forecasting of electric vehicle charging demand for stochastic power system operation. *Electr. Power Syst. Res.* **2016**, *140*, 378–390. [[CrossRef](#)]
9. Buzna, L.; De Falco, P.; Khormali, S.; Proto, D.; Straka, M. Electric vehicle load forecasting: A comparison between time series and machine learning approaches. In Proceedings of the 2019 1st International Conference on Energy Transition in the Mediterranean Area (SyNERGY MED), Cagliari, Italy, 28–30 May 2019.
10. Louie, H.M. Time-series modeling of aggregated electric vehicle charging station load. *Electr. Power Compon. Syst.* **2017**, *45*, 1498–1511. [[CrossRef](#)]
11. Khodayar, M.; Liu, G.; Wang, J.; Khodayar, M.E. Deep learning in power systems research: A review. *CSEE J. Power Energy Syst.* **2020**, *7*, 209–220.
12. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
13. Kumar, K.N.; Cheah, P.H.; Sivaneasan, B.; So, P.L.; Wang, D.Z. Electric vehicle charging profile prediction for efficient energy management in buildings. In Proceedings of the 2012 10th International Power & Energy Conference (IPEC), Ho Chi Minh City, VietNam, 12–13 December 2012.

14. Jahangir, H.; Tayarani, H.; Ahmadian, A.; Golkar, M.A.; Miret, J.; Tayarani, M.; Gao, H.O. Charging demand of plug-in electric vehicles: Forecasting travel behavior based on a novel rough artificial neural network approach. *J. Clean. Prod.* **2019**, *229*, 10291044. [[CrossRef](#)]
15. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
16. Chang, M.; Bae, S.; Cha, G.; Yoo, J. Aggregated electric vehicle fast-charging power demand analysis and forecast based on LSTM neural network. *Sustainability* **2021**, *13*, 13783. [[CrossRef](#)]
17. Marino, D.L.; Amarasinghe, K.; Manic, M. Building energy load forecasting using deep neural networks. In Proceedings of the IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016.
18. Kong, W.; Dong, Z.Y.; Jia, Y.; Hill, D.J.; Xu, Y.; Zhang, Y. Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Trans. Smart Grid* **2017**, *10*, 841–851. [[CrossRef](#)]
19. Kong, W.; Dong, Z.Y.; Jia, Y.; Hill, D.J.; Xu, Y.; Zhang, Y. Ultra-Short-Term Prediction of EV Aggregator’s Demand Response Flexibility Using ARIMA, Gaussian-ARIMA, LSTM and Gaussian-LSTM. In Proceedings of the 2021 3rd International Academic Exchange Conference on Science and Technology Innovation (IAECST), Guangzhou, China, 10–12 December 2021.
20. Ahmed, S.; Nielsen, I.E.; Tripathi, A.; Siddiqui, S.; Rasool, G.; Ramachandran, R.P. Transformers in Time-series Analysis: A Tutorial. *arXiv* **2022**, arXiv:2205.01138.
21. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
22. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
23. Dong, L.; Xu, S.; Xu, B. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.
24. Liu, Y.; Zhang, J.; Fang, L.; Jiang, Q.; Zhou, B. Multimodal motion prediction with stacked transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021.
25. Box, G.E.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
26. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arxiv* **2014**, arXiv:1409.0473.
27. City of Boulder Open Data. 2021. Available online: https://open-data.bouldercolorado.gov/datasets/39288b03f8d54b39848a2df9f1c5fca2_0/explore (accessed on 28 December 2022).
28. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv* **2019**, arXiv:1912.01703.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.