Hybrid Model Based on an SD Selection, CEEMDAN, and Deep Learning for Short-Term Load Forecasting of an Electric Vehicle Fleet

Ahmad Mohsenimanesh, Evgueniy Entchev * and Filip Bosnjak

Hybrid Energy Systems, CanmetENERGY Ottawa Research Centre, Natural Resources Canada, 1 Haanel Drive, Ottawa, ON K1A 1M1, Canada
* Correspondence: evgueniy.entchev@nrcan-rncan.gc.ca; Tel.: +1-613-220-9517

Abstract: Forecasting the aggregate charging load of a fleet of electric vehicles (EVs) plays an important role in the energy management of the future power system. Therefore, accurate charging load forecasting is necessary for reliable and efficient power system operation. A hybrid method that is a combination of the similar day (SD) selection, complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN), and deep neural networks is proposed and explored in this paper. For the SD selection, an extreme gradient boosting (XGB)-based weighted k-means method is chosen and applied to evaluate the similarity between the prediction and historical days. The CEEMDAN algorithm, which is an advanced method of empirical mode decomposition (EMD), is used to decompose original data, to acquire intrinsic mode functions (IMFs) and residuals, and to improve the noise reduction effect. Three popular deep neural networks that have been utilized for load predictions are gated recurrent units (GRUs), long short-term memory (LSTM), and bidirectional long short-term memory (BiLSTM). The developed models were assessed on a real-life charging load dataset that was collected from 1000 EVs in nine provinces in Canada from 2017 to 2019. The obtained numerical results of six predictive combination models show that the proposed hybrid SD-CEEMDAN-BiLSTM model outperformed the single and other hybrid models with the smallest forecasting mean absolute percentage error (MAPE) of 2.63% Canada-wide.

Keywords: EV; charging load; SD selection; CEEMDAN; GRU; LSTM; BiLSTM; short-range EV; mid-range EV; long-range EV

1. Introduction

1.1. Problem Statement

The global EV market is growing exponentially. The World Economic Forum says worldwide sales of EVs reached 6.6 million in 2021, almost doubling from the previous year [1]. The International Energy Agency estimated that the global EV fleet could reach 250 million by 2030 [2]. With the significant growth of the EV industry, it is bound to bring new challenges to the power systems due to the large battery capacity and uncertain charging behaviors of EV users [3]. This would result in significant peak–valley differences in load in featured time slots, particularly in a super-short-term time scale. Therefore, utilities and other power producers need to be prepared to meet the increased loads as transportation electrification grows, and to be able to forecast required electricity with a minimum error, to maintain stable and effective power system operation.

1.2. Proposed Solutions

Over the past few decades, scientists have proposed various solutions to improve the accuracy of short-term load forecasting (STLF) methods including traditional methods, the SD selection, the EMD methods, artificial intelligence (AI), and hybrid forecasting models [4–13]. With energy grid diversification, an increasing number of factors impact the
load demand such as weather, holidays, and electricity prices [3]. Traditional load forecasting methods are unable to provide prediction models with sufficient accuracy [14]. These models are based on mathematical methods that often perform poorly when predicting non-linear problems [15].

The SD selection is based on historical days that have comparable features to capture the features of load. To predict short-term load, a weekday index and weather events similar to the forecasted day were used [16]. The SD selection was applied to assess the attribute weights using the XGB algorithm and to compute the distance between the chosen day and the day that depends on different measurement attributes in different weights [12]. The authors also used the k-means algorithm to merge SDS into one cluster and applied it as input data for the succeeding load prediction based on the XGB distance. However, scientists found that the SD method cannot sufficiently capture the complex electric load features alone and it should be combined with other methods [12,16–18].

AI and machine learning (ML) modeling techniques are broadly used by many electric utility companies to handle accurate load forecasting problems [19]. Although comprehensive research has been accomplished, accurate STLF remains a challenge due to its non-stationary electric load data and long-term dependencies estimating horizon [12]. The LSTM model, which is a special type of recurrent neural network (RNN), was used to predict the aggregated demand-side electric load over the short- and medium-term horizons [20]. The BiLSTM has been applied in several areas of study to provide accurate aggregated electric load prediction results [21–23]. The GRU model was used to forecast the short-term load of EV charging stations and the state of charge of batteries [3,24]. A comparative study of ML approaches using LSTM, BiLSTM, and GRU was evaluated for day-ahead charging of the EV fleet in Canada [25]. The results showed that the BiLSTM algorithm has the lowest error among the used models and was best suited for load prediction of the EV fleet. However, in view of the complexity and non-stationarity of the aggregate load of the EV fleet, it is difficult to obtain accurate results using neural networks.

The EMD method has been used by many scholars in a wide range of applications including electric load, wind speed, solar radiation, and crude oil price to improve prediction accuracy [5–7,12,26,27]. The EMD method can sufficiently extract the features of the original data from non-stationary and unstable time series into a series of frequency components [28]. There are many types of EMD methods that can be applied to the time series including ensemble EMD (EEMD), complete EEMD (CEEMD), and CEEMD with adaptive noise (CEEMDAN). Compared with EMD and EEMD, the CEEMDAN method can perform a better spectral separation of the IMFs at a lower computational cost [29]. More recently, CEEMDAN was used to reconstruct the original input/output variables for electricity demand forecasting [30]. They found that the accuracy of the load model with the CEEMDAN method is greater than that without it.

Based on the above-mentioned solutions, a hybrid approach consisting of the SD selection, the CEEMDAN, and deep neural networks is proposed to forecast the aggregated load of an EV fleet. The SD method is used to capture the features of load using the XGB algorithm and cluster them using the k-means method. The CEEMDAN technique is introduced to decompose historical data into a set of IMFs and a residue. Three ML methods comprising LSTM, BiLSTM, and GRU were applied to compare and select the best-suited method for predicting the charging load of the EV fleet.

The contribution of the presented paper is the application of the CEEMDAN, the SD methods, and deep neural networks applied to the problem of STLF in EV fleet research with:

- a unique heterogeneous fleet of 1000 EVs, grouped into long-range, mid-range, and short-range EVs, in terms of their battery capacity;
- the three-day-ahead predictions with high accuracy achieved on real data provided by FleetCarma Inc. (Waterloo, ON, Canada);
- an evaluation with various methods comprising the SD selection, the CEEMDAN method, and different RNN architectures.
The following section provides the theoretical background of our method. Section 3 describes data pre-processing and feature analysis. Section 4 describes the experimental results and validation. Section 5 presents future work. Section 6 makes a conclusion.

2. Materials and Methods

This section describes the SD approach, the signal-processing algorithm called CEEMDAN, and the three structures of RNN comprising LSTM, BiLSTM, and GRU investigated in this study.

2.1. SD Approach Using XGB and k-Means

This section estimates the weights of the features, i.e., vehicle groups, charging levels, charging locations, temperature, electricity rates, seasonal category, weather events, day type, etc. [25], using the XGB algorithm. Then, the weighted features are integrated using the k-means clustering.

XGB is a decision tree-based ensemble ML algorithm that applies a gradient boosting framework. XGB is a supervised algorithm that assembles all base learners into strong learners. The prediction result by XGB is equal to the total of all base learners.

The XGB process is as follows:

\[ \hat{y}_i = \sum_{n=1}^{N} f_n(x_i), \quad f_n \in F, \]  

where \( f_n \) represents the nth decision tree, \( N \) is the number of decision trees, \( F \) is the space that covers a set of decision trees. The objective function of XGB includes the loss function and the regularization term, which are defined as:

\[ \text{obj} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{m=1}^{n} \Omega (f_n), \]  

\[ \Omega (f_n) = \gamma T + 1/2 \beta \sum_{j=1}^{T} w_j^2, \]

where \( l (y_i, \hat{y}_i) \) denotes the loss function, \( y_i \) and \( \hat{y}_i \) are the prediction and the target values, \( \Omega \) states the regularization term, \( n \) is the number of targets \( y_i \), \( T \) specifies the leaf node number in a decision tree, \( w_j \) is the score of leaf nodes, \( \gamma \) and \( \beta \) act as the penalty factors. Then, k-means clustering, which is an unsupervised learning algorithm, is used to partition \( n \) observations into different clusters, so that similar data points are grouped together and underlying patterns can be discovered.

2.2. Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN)

CEEMDAN is an algorithm based on EMD [31], that can effectively solve the EMD mode mixing and the EEMD residual noise problems. The decomposition process is as follows:

i. Add white noise \( V^i(t) \) with a normal distribution to the original signal \( S(t) \). The signal for the i-th decomposition, where \( i \) denotes the number of decompositions, is represented as

\[ S^i(t) = S(t) + V^i(t), \quad i = 1, \cdots, I. \]  

ii. The EMD decomposes the trial signal \( S^i(t) \) to obtain \( IMF^i_1 \) correspondingly, so

\[ IMF_1 = 1/I \sum_{i=1}^{I} IMF^i_1(t), \quad \text{residual } r_1(t) = S(t) - IMF_1. \]  

iii. Add white noise \( V^i(t) \) to the residual \( r_1(t) \), execute the trial \( i \) times \((i = 1, 2, \cdots, I)\), and each trial adopts EMD to decompose

\[ r^i_1(t) = x(t) + v^i(t) \]
to find its first-order component $IMF_1$.

\[
IMF_2 = \frac{1}{T} \sum_{t=1}^{T} IMF_1(t)
\]  

Residual
\[
r_2(t) = S(t) - IMF_2
\]

iv. Then, the signal is further decomposed by EMD to calculate the second IMF mode and the relating residue by repeating the above decomposition process. When the residual cannot be decomposed by EMD, the program ends. The original signal can be represented as:
\[
S(t) = \sum_{i=1}^{n} IMF_i + r_n(t).
\]  

2.3. Long Short-Term Memory (LSTM) and Bidirectional LSTMs

LSTM is a specific RNN architecture used to prevent gradient vanishing and exploding problems. LSTMs can acquire long-term dependencies by a gate mechanism that can control the flow of information. These gates can realize which data in the sequence are important to preserve or forget. There are three different gates in the LSTM cell: input, forget, and output (Figure 1).

![Figure 1. The architecture of LSTM cell. Numbered values correspond to the equation numbers below.](image)

The following equations are the LSTM cell states and parameters’ updating scheme [12,20,32]:

\[
f_t = \delta \left( W_f [h_{t-1} * x_t] + b_f \right),
\]

\[
i_t = \delta \left( W_i [h_{t-1} * x_t] + b_i \right),
\]

\[
\tilde{C}_t = \text{tanh} \left( W_c [h_{t-1} * x_t] + b_c \right),
\]

\[
C_t = i_t * \tilde{C}_t + f_t * C_{t-1},
\]

\[
O_t = \delta \left( W_o [h_{t-1} * x_t] + b_o \right),
\]

\[
h_t = o_t * \text{tanh}(c_t),
\]

where $i_t, f_t, \text{ and } O_t$ represent the activation of input, forget, and output gate at time step $t$, respectively, $H_{t-1}$ is the output at time step $t - 1$, $x_t$ is the input at the present moment, and $C_{t-1}$ is the memory cell from the previous state. The forget gate ($f_t$) can input the information in $h_{t-1}$ and $x_t$, and outputs a vector, with values ranging from 0 to 1 for the cell state $C_{t-1}$, 1 means completely reserved and 0 means completely discarded. The input gate $i_t$ decides what new information is going to be stored in the cell state, and the first part is...
the sigmoid layer called the “input gate layer” that decides which values will be updated. The second part is that the \(\text{tanh}\) layer creates a vector of a new candidate value, \(\hat{C}_t\) which is added to the state. The current block memory \(C_t\) is created by the accumulation of the items from the previous block and input gate. Finally, the output gate \(O_t\) determines the cell state value, where \(W\) and \(b\) are weight and bias. The following equations are sigmoid and hyperbolic tangent functions

\[
\delta(x) = 1 / \left(1 + e^{-x}\right), \quad (16)
\]

\[
tanh(x) = (e^x - e^{-x}) / (e^x + e^{-x}). \quad (17)
\]

The idea of BiLSTM is to combine input information in the past and future time step in LSTM models. In BiLSTM, information can be preserved from both the past and future at any point in time [33].

2.4. Gated Recurrent Unit (GRU)

GRU is a newer generation of RNN architecture and is similar to LSTM. GRU does not have the cell state, instead it uses a hidden state to transfer information. Therefore, GRU has only two gates: reset and update gate (Figure 2).

![The architecture of the GRU cell. Numbered values correspond to the equation numbers below.](image)

The key equations for GRU are shown below [24]:

\[
z_t = \delta_g(W_z x_t + U_z h_{t-1} + b_z), \quad (18)
\]

\[
r_t = \delta_g(W_r x_t + U_r h_{t-1} + b_r), \quad (19)
\]

\[
\hat{h}_t = \Phi_h(W_h x_t + U_h(r_t \ast h_{t-1}) + b_h), \quad (20)
\]

\[
h_t = (1 - z_t) \ast h_{t-1} + \hat{h}_t, \quad (21)
\]

where \(x_t, h_t, \hat{h}_t, z_t\) and \(r_t\) are input, output, activation function, update gate, and reset gate. \(W, U,\) and \(b\) are parameter matrices and vectors.

\(\delta_g\) and \(\Phi_h\) are sigmoid and hyperbolic tangent functions. GRUs have fewer parameters and thus may train faster than LSTM.

2.5. The Proposed SD-CEEMDAN-BiLSTM Prediction Model

In the BiLSTM model, the rectified linear unit (ReLU) function is used as an activation function of the stack of a fully connected layer and the mean square error (MSE) is used as a loss function.

\[
\text{Loss} = \text{MSE} = 1 / N \sum_{n=1}^{N} (d_n - y_n)^2 \quad (22)
\]
where \( N \) is the sum of estimated number of days, \( y_n \) is the estimated value, and \( d_n \) is the actual value.

The proposed model generally comprises the following steps, which are shown in Figure 3.

I. The feature weight is calculated by the XGB method and is merged with the k-means algorithm to establish the SD cluster.

II. The CEEMDAN method is utilized to decompose the charging load into several IMF sequences \( C_i(t) (i = 1, 2, \ldots, M) \) and one residue \( R_M(t) \).

III. Each IMF and the residue item are normalized and used as the input to the BiLSTM model for training and obtaining the predicted values, respectively. The results of the test set predictions are \( C_i(t) (i = 1, 2, \ldots, M) \) and \( R_M(t) \).

IV. Then, the final forecast results are adjusted using the below formula.

\[
S_i(t) = C_i(t) + R_M(t) \quad (t = 1, 2, \ldots, L)
\]  

where \( L \) is the test series length, and \( S_i(t) \) is the final predictive series of the test set.

The architecture of the BiLSTM was derived by testing different configurations of units in each layer and calculating the MAPE on training and testing datasets. Table 1 shows the MAPE for a different number of layers and units for data with one-day-ahead prediction. Increasing the capacity of the neural network can be seen by increasing the number of layers and units, which improves error on the training and testing dataset. However, the

![Figure 3. Hybrid forecast model based on SD-CEEMDAN-BiLSTM.](image-url)
proposed model executes well on the training data by a 2-layer network with 100 units in layer 2 and 30 units in layer 1.

Table 1. MAPE (%) of the proposed model architecture.

<table>
<thead>
<tr>
<th>Layers</th>
<th>Units</th>
<th>MAPE (Testing)</th>
<th>MAPE (Training)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>3.24</td>
<td>3.04</td>
</tr>
<tr>
<td>1</td>
<td>30</td>
<td>3.10</td>
<td>2.99</td>
</tr>
<tr>
<td>1</td>
<td>50</td>
<td>3.15</td>
<td>2.98</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>3.12</td>
<td>2.95</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>2.95</td>
<td>2.96</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td>2.84</td>
<td>3.02</td>
</tr>
<tr>
<td>2</td>
<td>50</td>
<td>2.79</td>
<td>3.03</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>2.63</td>
<td>2.93</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>2.99</td>
<td>3.11</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>2.89</td>
<td>3.05</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>2.87</td>
<td>3.02</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>2.83</td>
<td>3.01</td>
</tr>
</tbody>
</table>

3. Data Pre-Processing and Feature Analysis

3.1. Data Description and Input Variable

A large dataset with over 727,000 charging load events was collected in nine provinces in Canada (Figure 4) by FleetCarma Inc., in partnership with 10 electric utility companies and the University of Waterloo, over a three-year period using FleetCarma data loggers via the OBD II port of 1000 EVs. The collected data include charging loads of thirty-five vehicle models for three charging locations and three charging levels. The data include charging load start and end times, energy consumption, energy loss, and the state of charge of the battery at start time and end time.

Figure 4. Number and distribution of EVs used in this project.

For the charging load prediction, two individual models were developed to process battery size and weather events into feature input parameters, and the full results have been published recently [25].

3.2. XGB Feature Importance

Several features described in Section 2.1 are used as inputs for the XGB algorithm to compute the feature importance with a charging load for a fleet of EVs (Figure 5). The increase in the feature importance of the individual predictors in the tree is then visualized. A higher value of the metric “Feature gain score”, compared to another feature, implies it is more important for generating a prediction. It can be seen from Figure 5 that the charge load is sensitive to temperature variables. In addition, the SD charging load, which has the
highest feature gain score, is an important feature for load prediction. This assumption is consistent with the data analysis results.

![Figure 5. XGB feature importance.](image)

3.2. XGB Feature Importance

Several features described in Section 2.1 are used as inputs for the XGB algorithm to compute the feature importance with a charging load for a fleet of EVs (Figure 5). The increase in the feature importance of the individual predictors in the tree is then visualized. A higher value of the metric “Feature gain score”, compared to another feature, implies it is more important for generating a prediction. It can be seen from Figure 5 that the charge load is sensitive to temperature variables. In addition, the SD charging load, which has the highest feature gain score, is an important feature for load prediction. This assumption is consistent with the data analysis results.

3.3. Data Decomposition

The CEEMDAN method is utilized to decompose the aggregated charging load at low- and high-frequency waves, which correspond to daily and seasonal, respectively. All graphs in Figure 6 are on the same scale, thus enabling the evaluation of the contribution of each extracted IMF on the daily, weekly, monthly, and seasonal scales (Figure 6b–e). It can be seen that increasing the range of data for decomposition from daily to seasonally would decrease the amplitude of the fluctuation. Therefore, it presents improvements in the extracted IMFs and provides a valuable abstract for data visualization. For load forecasting, the generated IMFs were divided into training and testing sets, with the ratio selected as 76/24 (16 months of training data, and 5 months of testing data).

For time interval processing, Anaconda, which is a distribution of the Python programming language, was used to integrate all the charging load event data based on the time stamps and split them into 24 h time stamps every single day.

3.4. Evaluation Indicators

To evaluate the performance of the models and the prediction errors, three commonly used metrics, including root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percent error (MAPE), were employed. The error indicators are defined as below:

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2},
\]

(24)

\[
MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|,
\]

(25)

\[
MAPE = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{y_i - \hat{y}_i}{y_i} \right|,
\]

(26)

where \(y_i\) is the actual value, \(\hat{y}_i\) is the prediction value, and \(N\) is the total amount of data.
Figure 6. The original data sequence of the aggregated daily charging load (a) and the result of CEEMDAN; daily (b) weekly (c), monthly (d), seasonally (e).

4. Results

Six predictive combination models including the SD selection, the CEEMDAN method, and three different RNN architectures (GRU, LSTM, BiLSTM) were selected for comparative
analysis and for verifying the predictive performance of the proposed model. The six predictive combination models in this research are implemented using Python Anaconda.

Figures 7 and 8 illustrate the results of the aggregated charge power, in kilowatts (kW), during the three-day-ahead prediction period for all single and hybrid prediction models in spring and winter, respectively. The training period is from 20 August 2017 to 31 December 2018 and the three-day-ahead prediction period is from 30 April 2019 to 3 May 2019 and from 26 January 2019 to 29 January 2019 for spring and winter, respectively. It can be seen that the peak charging load during the three-day-ahead period occurred around 7 p.m., and the real error, which is the difference between actual and predicted load, is the largest at that point (Figures 7 and 8). However, the proposed hybrid model that includes the SD selection and decomposition method outperformed the single (Figures 7b and 8b) and other hybrid models (Figures 7c and 8c) at the peak points. Although all six models show good predictive results on the three-day-ahead period, the SD-CEEMDAN-BiLSTM model achieves a comparatively better performance with the smallest forecasting MAPE of 2.63% Canada-wide. Such trends in terms of improvements of the models using hybrid methods for electricity load, wind speed, solar radiation, and crude oil price prediction performance have also been observed in reported experimental studies [5–7,12].

The aggregated charge power (kW) in spring and winter are compared and the results show that the peak periods in spring are shorter than in winter, resulting in higher overall charge power in winter (Figures 7 and 8). This might be due to decreased battery efficiency, a higher charging load for cabin heating, and more charging load per season in cold winter weather. Comparisons between the forecasting curve of the single models in spring and winter (Figures 7b and 8b) show that the BiLSTM curves follow the actual data better than other models. Comparisons between the prediction curves of the hybrid models in spring and winter (Figures 7c and 8c) show that the predicting curve of the proposed SD-CEEMDAN-BiLSTM model is closer to the original charge power curve than the other models.

Table 2 shows the error values of the six prediction models. The training period is from 20 August 2017 to 31 December 2018, and the prediction period is from 1 January 2019 to 30 May 2019. Among the six prediction models, the results obtained by using SD-CEEMDAN-BiLSTM are more fitted to the original data. The RMSE, MAPE, and MAE values are 14.5, 2.6, and 13.1, respectively, which are much smaller than other prediction models. These data indicate that the SD-CEEMDAN-BiLSTM model has better stability and accuracy and can be well applied to EV load prediction. LSTM has the maximum MAPE value. Although all three single models and other hybrid models followed the general trend of the raw data, their forecasting errors were high.

<table>
<thead>
<tr>
<th>Model</th>
<th>Train MAE</th>
<th>Test MAE</th>
<th>Train MAPE</th>
<th>Test MAPE</th>
<th>Train RMSE</th>
<th>Test RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRU</td>
<td>23.57</td>
<td>22.50</td>
<td>6.26</td>
<td>6.42</td>
<td>30.47</td>
<td>30.34</td>
</tr>
<tr>
<td>LSTM</td>
<td>25.70</td>
<td>25.14</td>
<td>6.44</td>
<td>7.26</td>
<td>28.04</td>
<td>33.70</td>
</tr>
<tr>
<td>Bi-LSTM</td>
<td>20.09</td>
<td>22.37</td>
<td>5.03</td>
<td>6.43</td>
<td>27.80</td>
<td>29.81</td>
</tr>
<tr>
<td>SD-BiLSTM</td>
<td>18.46</td>
<td>18.93</td>
<td>3.98</td>
<td>4.33</td>
<td>21.59</td>
<td>23.77</td>
</tr>
<tr>
<td>CEEMDAN-BiLSTM</td>
<td>15.23</td>
<td>16.90</td>
<td>3.42</td>
<td>3.81</td>
<td>18.01</td>
<td>18.24</td>
</tr>
<tr>
<td>SD-CEEMDAN-BiLSTM</td>
<td>11.02</td>
<td>13.10</td>
<td>2.54</td>
<td>2.63</td>
<td>13.64</td>
<td>14.51</td>
</tr>
</tbody>
</table>
Figure 7. Predictions and actual aggregated charge power during the three-day-ahead prediction period in spring for (a) all prediction models, (b) single prediction model, (c) hybrid prediction model.

Tables 3 and 4 compare the MAPE values for all the models by month for the one-day-ahead and the three-day-ahead prediction. The training period is from 20 August 2017 to 31 December 2018, and the prediction period is from 1 January 2019 to 30 May 2019. The results indicate that the proposed model is significantly superior to the single models and other hybrid models. MAPE of the SD-CEEMDAN-BiLSTM model is the lowest among all the models. The average forecasting accuracy of the proposed model reaches 97.93% and 97.29% in the one-day-ahead and the three-day-ahead prediction, respectively.
Figure 8. Predictions and actual aggregated charge power during the three-day-ahead prediction period in winter for (a) all prediction models, (b) single prediction model, (c) hybrid prediction model.

The aggregated charge power (kW) in spring and winter are compared and the results show that the peak periods in spring are shorter than in winter, resulting in higher overall charge power in winter (Figures 7 and 8). This might be due to decreased battery efficiency, a higher charging load for cabin heating, and more charging load per season in cold winter weather.
Table 3. MAPE (%) of all the models per month for the one-day-ahead prediction.

<table>
<thead>
<tr>
<th></th>
<th>GRU</th>
<th>LSTM</th>
<th>Bi-LSTM</th>
<th>SD-BiLSTM</th>
<th>CEEMDAN-BiLSTM</th>
<th>SD-CEEMDAN-BiLSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>January</td>
<td>5.68</td>
<td>6.45</td>
<td>5.32</td>
<td>3.63</td>
<td>3.18</td>
<td>2.21</td>
</tr>
<tr>
<td>February</td>
<td>10.13</td>
<td>7.84</td>
<td>6.16</td>
<td>3.78</td>
<td>3.01</td>
<td>2.00</td>
</tr>
<tr>
<td>March</td>
<td>4.66</td>
<td>5.58</td>
<td>5.82</td>
<td>3.73</td>
<td>3.10</td>
<td>1.76</td>
</tr>
<tr>
<td>April</td>
<td>7.48</td>
<td>6.51</td>
<td>5.99</td>
<td>4.10</td>
<td>3.13</td>
<td>2.76</td>
</tr>
<tr>
<td>May</td>
<td>7.25</td>
<td>6.94</td>
<td>4.83</td>
<td>3.57</td>
<td>2.71</td>
<td>2.54</td>
</tr>
<tr>
<td>Avg.</td>
<td>7.04</td>
<td>6.66</td>
<td>5.62</td>
<td>3.76</td>
<td>3.02</td>
<td>2.25</td>
</tr>
</tbody>
</table>

Table 4. MAPE (%) of all the models per month for the three-day-ahead prediction.

<table>
<thead>
<tr>
<th></th>
<th>GRU</th>
<th>LSTM</th>
<th>Bi-LSTM</th>
<th>SD-BiLSTM</th>
<th>CEEMDAN-BiLSTM</th>
<th>SD-CEEMDAN-BiLSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>January</td>
<td>7.27</td>
<td>7.29</td>
<td>6.41</td>
<td>4.46</td>
<td>3.72</td>
<td>2.23</td>
</tr>
<tr>
<td>February</td>
<td>7.10</td>
<td>9.01</td>
<td>5.77</td>
<td>3.84</td>
<td>4.00</td>
<td>2.14</td>
</tr>
<tr>
<td>March</td>
<td>6.70</td>
<td>4.93</td>
<td>4.60</td>
<td>4.72</td>
<td>3.95</td>
<td>2.88</td>
</tr>
<tr>
<td>April</td>
<td>8.64</td>
<td>5.38</td>
<td>5.61</td>
<td>3.18</td>
<td>3.16</td>
<td>2.57</td>
</tr>
<tr>
<td>May</td>
<td>9.82</td>
<td>6.73</td>
<td>5.08</td>
<td>3.20</td>
<td>3.70</td>
<td>2.66</td>
</tr>
<tr>
<td>Avg.</td>
<td>7.90</td>
<td>6.67</td>
<td>5.49</td>
<td>3.88</td>
<td>3.70</td>
<td>2.49</td>
</tr>
</tbody>
</table>

5. Future Work

Issues that could be addressed in future work:
1. Examine reinforcement learning approaches for dealing with real large datasets composed of time series.
   To learn the individual EV user energy consumption in order to reduce peak power on an aggregate level.
   To investigate the individual and cumulative impact of battery capacity and time of use rate on the charging behavior of EV users.
2. Apply smart charging strategy to minimize overall vehicle energy use costs.
3. Develop a methodology that facilitates the decision making in real-time, using models that can be applied to a wide range of vehicle models and/or groups.

6. Conclusions

This study proposes a hybrid model based on the SD selection, CEEMDAN signal processing technique, and the BiLSTM network. The proposed approach was compared with LSTM, GRU, SD-BiLSTM, and CEEMDAN-BiLSTM models to assess the effectiveness of the proposed model on hourly EV charging load forecasting. To achieve the best performance, fine-tuning and proper hyper-parameters were investigated. The performance of these prediction models was evaluated in terms of MAE, RMSE, and MAPE. The main conclusions of this study can be summarized as follows:
1. The hybrid approach is feasible and reasonable for the three-day-ahead load forecasting of a real-life dataset that was collected from 1000 EVs in nine provinces in Canada from 2017 to 2019.
2. The SD algorithm applied for optimizing the single models and the CEEMDAN technique applied in extracting the various components could both improve the prediction performance of the single models.
3. Overall, the proposed SD-CEEMDAN-BiLSTM model shows a competitive technique for enhancing the charging load prediction accuracy of the EV fleet.
Author Contributions: Conceptualization, A.M.; methodology, A.M.; software, A.M. and F.B.; formal analysis, A.M., E.E. and F.B.; investigation, A.M. and E.E.; writing—original draft preparation, A.M., E.E. and F.B.; writing—review and editing, A.M., E.E. and F.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research is financially supported by Natural Resources Canada, the Program of Energy Research and Development under grant CEO-19-112.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within this review article.

Acknowledgments: The authors would like to thank the Office of Energy Research and Development (OERD) of Natural Resources Canada for their valuable financial support.

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

Nomenclature

EV Electric Vehicle
STLF Short-Term Load Forecasting
SD Similar Day
AI Artificial Intelligence
ML Machine Learning
RNN Recurrent Neural Network
LSTM Long Short-Term Memory
BiLSTM Bidirectional LSTM
GRU Gated Recurrent Unit
XGB Extreme Gradient Boosting
EMD Empirical Mode Decomposition
IMFs Intrinsic Mode Functions
EEMD Ensemble EMD
CEEMDAN CEEMD with Adaptive Noise
MR Mid-Range
SR Short-Range
fi the nth decision tree
F space
l loss function
yi and ŷi prediction and target values
Ω regularization term
n the number of targets
T number of leaf nodes
wij the score of leaf nodes
γ and β penalty factors
V(t) white noise
S(t) original signal
S′(t) experimental signal
r(t) residual
i(t) input gate
f(t) forget gate
O(t), h(t) output gate
δ and tanh activation functions
z(t) update gate vector
H(t−1) output at t−1 time slot
r(t) reset gate vector
x(t) input gate at the current moment
6s, Φh sigmoid and hyperbolic function
C(t) the current state of the cell
C(t−1) cell state
References


7. Lin, H.; Sun, Q. Crude Oil Prices Forecasting: An Approach of Using CEEMDAN-Based Multi-Layer Gated Recurrent Unit Networks. Energies 2020, 13, 1543. [CrossRef]

8. Lin, Y.; Yan, Y.; Xu, J.; Liao, Y.; Ma, F. Forecasting Stock Index Price Using the CEEMDAN-LSTM Model. Entropy 2020, 22, 140. [CrossRef]


