Research Article

Short-Term Demand Forecasting Method in Power Markets Based on the KSVM–TCN–GBRT

Guang Yang, Songhuai Du, Qingling Duan, and Juan Su

College of Information and Electrical Engineering, China Agricultural University, Beijing, China

Correspondence should be addressed to Guang Yang; yangguang2018@cau.edu.cn

Received 5 February 2022; Revised 18 March 2022; Accepted 11 April 2022; Published 30 April 2022

Academic Editor: Shahid Mumtaz

Copyright © 2022 Guang Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the consumption of new energy and the variability of user activity, accurate and fast demand forecasting plays a crucial role in modern power markets. This paper considers the correlation between temperature, wind speed, and real-time electricity demand and proposes a novel method for forecasting short-term demand in the power market. Kernel Support Vector Machine is first used to classify real-time demand in combination with temperature and wind speed, and then the temporal convolutional network (TCN) is used to extract the temporal relationships and implied information of day-ahead demand. Finally, the Gradient Boosting Regression Tree is used to forecast daily and weekly real-time demand based on electrical, meteorological, and data characteristics. The validity of the method was verified using a dataset from the ISO-NE (New England Electricity Market). Comparative experiments with existing methods showed that the method could provide more accurate demand forecasting results.

1. Introduction

The inherent volatility and uncertainty of renewable energy sources, such as wind power and photovoltaics, may lead to large deviations between the bid output and the actual output when renewable energy sources participate in the market. Also, customers change their electricity load when they receive a price change or incentive signal from the supply side, taking into account their own production or consumption. Hence, timely and accurate information on the demand changes has imposed higher requirements on the accuracy of demand forecasting [1].

In current electricity forecasting, it is necessary to fully integrate economic, meteorological, and other multidimensional information; use advanced data-driven processing means; and deeply analyze the change patterns and laws of renewable energy and demand [2]. This creates the need to study demand response characteristics under multidimensional variables in the electricity market environment to improve forecasting accuracy [3]. For power system operation, short-term demand forecasts can predict demands from 1 hour to 168 hours in advance [4]. To obtain accurate forecasting results, different forecasting models such as regression, statistical, and state-space methods are used. Expert system-based, evolutionary process, fuzzy systems, and artificial intelligence algorithms have also been introduced. Wang et al. [5] proposed a method that decomposes the demand into the trend series and the fluctuating series and then builds the corresponding forecasting models separately. To improve the internal clustering performance, Dongyeon et al. [6] proposed the logistic mixed vector autoregression model, which combines clustering and prediction into one model through an expectation-maximization algorithm. Dong et al. [7] proposed an indicator variable to capture the abnormal information on special days, such as national holidays, proximity days, and bridging days. In Yin et al. [8], the multispatiotemporal convolutional network was applied to short-term demand prediction tasks, which reduced the noise error of demand data and enhanced the time-series characteristics of demand data. In [9], the demand was decomposed into different frequency components using an integrated empirical modal decomposition algorithm; the low-frequency components were then predicted using multivariate linear regression, the high-
frequency components were predicted using a long short-term memory neural network (LSTM), and the components were finally combined to obtain the demand forecast. Han et al. [10] presented a short-term individual residential demand forecasting model based on a combination of deep learning and k-means clustering, which is capable of effectively extracting the similarity of residential demand and performing residential demand forecasting accurately at the individual level. It first makes full use of k-means clustering to extract the similarity among residential demand and then employs deep learning to extract complicated patterns of residential demand. Lv et al. [11] designed a LightGBM-optimized LSTM to realize short-term stock prices. To improve the demand prediction accuracy in the case of single sample data, Mei et al. [12] proposed a model based on multiscale temporal features for LSTM. First, wavelet decomposition decomposes historical data into stable components, trend demand, and periodic series such as the response peak-valley magnitude and duration, highlighting different time-scale features. Second, LSTM is used to achieve further extraction of time-series characteristics and data fitting. Finally, the model directly outputs predicted values for multiple moments. M. Hadi Amini et al. [13] proposed an autoregressive integrated moving average method for forecasting conventional electrical loads and electric vehicle parking charging demand. Kianoosh et al. [14] proposed to model the nonseasonal and seasonal cycles of load data using regression (AR) and moving average (MA) components, which have been used to forecast electricity demand at different time scales. Moreover, Zhang et al. [15] used Singular Spectrum Analysis (SSA) to preprocess the data and then used a support vector machine (SVM) optimized by the cuckoo search (CS) algorithm to model the resulting sequence with different prediction strategies. Salah et al. [16] used wrapper and embedding feature selection methods to select the optimal features and a genetic algorithm (GA) to find the optimal time lag and number of layers to optimize the predictive performance of the LSTM model, which was used to construct a short-to-medium-term cumulative load forecasting model. LU et al. [17] proposed a hybrid model short-term load forecasting method based on the convolutional neural network (CNN) and LSTM network. The CNN was first used to extract the feature vector, and the feature vector was constructed in a time-series sequence and used as the input data for the LSTM network. In [18], a convolutional long- and short-term memory network (Conv-LSTM) was proposed for electrical load data, and it achieved better accuracy than the traditional prediction algorithm based on linear regression. CHEN et al. [19] derived the respective predictions based on LSTM and LightGBM training. The optimal weighting combination method was used to determine the weighting coefficients and derive the prediction values of the combined model to improve the accuracy of load prediction. In [20], predictions were extrapolated by calculating correlations between potential variables and outputs and predicting the future consumption of high performance. In [21], a hybrid method combining empirical mode decomposition (EMD), particle swarm optimization (PSO), and a fuzzy inference system based on the adaptive network (ANFISs) for short-term load prediction of microgrids was proposed. In [22], sample entropy was used to identify the nonlinearity and uncertainty of the original time series, and the optimal mode of the original series and the optimal input form of the model were determined by the feature selection method. Finally, the least square SVM adjusted by the multiobjective sines and cosines optimization algorithm was used to predict the power demand sequence. In [23], the parameters of the LSTM were first optimized using the Sparrow Search Algorithm (SSA); then, the dataset was preprocessed, and finally, the processed data were used for residential load training and prediction. Li et al. [24] combined LSTM with quantile regression to generate multiple quantile results and introduced a combinatorial layer that considers the constraint relationships between quantile prediction values to ensure that the quantile prediction values are reasonable.

The TCN was proposed in 2018 to offer greater performance advantages over recurrent neural networks (RNNs) in temporal data processing tasks [25]. Following the motivation above, we propose a novel method based on Kernel Support Vector Machine (KSVM)–TCN–Gradient Boosting Regression Tree (GBRT) for improving the short-term demand forecasting accuracy of power markets. The contributions of this paper are as follows:

1. The KSVM is used to extract and train classification features for real-time electricity demand on historical data by the features of temperature and wind speed, and a numerical calculation method is used to automatically select the parameters of the KSVM to derive a classification sequence of real-time electricity demand for future days as the feature sequence.

2. A multivariable TCN method is used to capture the fluctuation trend of day-ahead demand in the day-ahead market to predict the real-time demand series in the real-time market.

3. The TCN–GBRT method integrating time-domain processing, integrated learning, and parallel feature processing is proposed. TCN is able to extract features and temporal relationships owing to its residual network and convolutional structure, avoiding gradient disappearance and gradient explosion in deep learning. GBRT can combine multiple weak classifiers into a single strong classifier that takes the best of all the weak classifiers and achieves optimal performance.

2. Materials and Methods

2.1. KSVM Model. In this paper, a Gaussian Kernel SVM is used in data feature extraction to solve the classification problem of the presence of nonlinear separability. That is, given a training sample set \( D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}, y_i = \{-1, 1\} \), the basic type of Kernel SVM is defined as in the following equation:
\[
\min_{w,b} \frac{1}{2} \|w\|^2,
\]
\[s.t. \ y_i (w^T \phi(x_i) + b) \geq 1, \quad i = 1, 2, \ldots, m, \tag{1}\]
\[y = \frac{\|w\|^2}{2}, \]

where \(y\) is the sum of the distances from the support vector to the hyperplane, called the “margin.” Finding the maximizing interval is equivalent to finding the minimum \(\|x\|^2\).

\(\omega = (\omega_1; \omega_2; \ldots; \omega_d)\) is the normal vector of the hyperplane, \(b\) is the displacement term, and \(\phi(x)\) denotes the feature vector after mapping \(x\).

The dual problem is defined in the following equation:
\[
\max \sum_{i=1}^{m} \delta_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \delta_i \delta_j y_i \phi(x_i)^T \phi(x_j),
\]
\[s.t. \sum_{i=1}^{m} \delta_i y_i = 0, \quad \delta_i \geq 0, \quad i = 1, 2, \ldots, m. \tag{2}\]

Here, \(\kappa(\cdot, \cdot)\) is the kernel function given in
\[
\kappa(x_i, x_j) = \phi(x_i)^T \phi(x_j). \tag{3}\]

The kernel function maps the data from the original space into a high-dimensional Hilbert space, where a more efficient classification hyperplane exists than in the original space.

Suppose that \(\{x^{(i)}_j\}_{j=1}^{N} \subset R^d\) is the set of training samples in class \(i\), where \(N_i\) represents the training samples in class \(i\) \((i = 1, 2, \ldots, L)\). The Gaussian Radial Basis Function (RBF) kernel is defined as in the following equation:
\[
\kappa(x, z, \sigma) = \exp \left(-\frac{\|x-z\|^2}{2\sigma^2}\right), \quad \sigma \in (0, \infty), \tag{4}\]

where \(x, z \in R^d\), and \(\sigma \in R - \{0\}\) are the corresponding parameters.

### 2.2. TCN Model

TCN is a one-dimensional full convolution network, which combines the structure of causal convolution, extended convolution, and the residual network. Causal convolution means that the output time is convolved only with the elements of the previous layer of time, which ensures that there will be no information leakage in the future. Dilated convolution is designed to capture a sufficiently long history of information, and the depth of the network model increases dramatically.

Define the one-dimensional sequence input \(X = (x_1, x_2, \ldots, x_n)\), where \(F: \cdot [0, 1, \ldots, k-1]\) is the function to the dilatative convolution, and the TCN convolution operation is defined as in the following equation:
\[
F(s) = (XCon_d)f(s) = \sum_{i=0}^{k-1} f(i)x_{s-d+i}, \tag{5}\]

where \(Con\) is the convolution operation; \(d\) is the dilatative convolution parameter; \(s\) is the current number of sequences; \(k\) is \(Con\ Size\); \(s-d\) is past directions; and a convolution with parameter \(d = 1, 2, 4\) and filter size \(k = 3\) is shown in Figure 1.

As shown in Figure 2, the mezzanine mapping is within the residual connection, as shown in equation (6). The residual network fits several nonlinear layers between the input and output portion. The more the features that are extracted, the closer the residual \(F(x)\) is to 0. Hence, when the network reaches an optimal structure, \(F(x)\) is pushed to 0 as the network layers deepen, leaving only the identity mapping \(x\). This overcomes the problem of TCN degradation due to increasing network layers. Using the residuals, each order of derivative plus the constant term 1 is as in equations (6) and (7). The error can still be effectively backpropagated at this point, even if the derivatives \(\partial f/\partial x\) are small.
2.3. GBRT Model. The GBRT algorithm is able to compensate for the tendency of the cart algorithm to overfit small sample data or produce instability and low prediction accuracy. The algorithm is an iterative decision tree algorithm, which consists of three parts: cart, Gradient Boosting algorithm, and reduction idea.

The basic idea of the algorithm is to use the Boosting method to iterate multiple weak learners with low prediction accuracy to form a strong learner with high prediction accuracy, that is, to reduce the residuals of the previous model by learning again so that the next generated model has a smaller error. The gradient iteration makes the combined model continuously improved, which is a kind of decision tree integrated learning algorithm designed to improve the model learning rate and prevent overfitting. That is, it does not fully trust each residual tree and uses gradual approximation to learn through multiple residual trees.

For continuous data types, the loss function is the classical loss function in Boosting, that is, the sum of squared errors, which is calculated as shown in (4). After M-th iterations, the prediction is shown in equations (8) and (9).

\[ L(y, f(x)) = \sum_{i=1}^{n} (y_i - f(x_i))^2, \quad (8) \]

\[ f(x) = \sum_{i=0}^{M} f_i(x), \quad (9) \]

where \( x \) is the input variables, \( y \) is the output variables, and \( i \) is the iteration count.

The key to improving the prediction accuracy of the GBRT model is the calculation of the residuals, and this paper uses the method proposed by Friedman [26] to calculate the residuals based on the negative gradient of the loss function.

In this paper, the negative gradient of the loss function is used as an approximation to the residuals in the boosted tree algorithm. Hence, the i-th sample of the m-th round \( g_{im} \) is calculated as shown in the following equation:

\[ g_{im} = -\left( \frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right)_{f(x_i)=f_{m-1}(x_i)}. \quad (10) \]

\( f_{m}(x) \) can be calculated from \( \beta_{m} \) and \( h_{m} \), where \( \beta_{m} \) denotes the optimal step for each iteration and is calculated as shown in (13). \( H_{m}(x) \) is the decision tree created in the m-th iteration. The prediction is shown in (11) and (12).

\[ f_{m}(x) = f_{m-1}(x) + \beta_{m}h_{m}(x). \quad (11) \]

\[ \beta_{m} = \arg \min_{\beta} \sum_{i=1}^{n} L(y_i, f_{m-1}(x) + \beta_{m}h_{m}(x)), \quad (12) \]

2.4. Short-Term Demand Forecasting Method Based on K SVM–TCN–GBRT. This paper takes into account day-ahead demand, real-time demand, date features, and meteorological features. An RBF kernel-SVM method is used to get the relationship between temperature, wind speed, and real-time power demand.

The TCN–GBRT method is also proposed to forecast the next day’s real-time demand and the next week’s real-time demand. Processing datasets are shown in Table 1.

### Table 1: Processing datasets.

<table>
<thead>
<tr>
<th>Type of data</th>
<th>Input VARIABLE t</th>
<th>Units</th>
<th>Processing method</th>
<th>Output variable t</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrical features</td>
<td>t – 24/168 day-ahead cleared demand</td>
<td>MW</td>
<td>Feature extraction based on TCN</td>
<td>t + 24/168 MW</td>
</tr>
<tr>
<td>Meteorological feature</td>
<td>t – 24/168 condition</td>
<td>°F</td>
<td>Feature extraction based on KSVM</td>
<td>t + 24/168 condition</td>
</tr>
<tr>
<td>Date features</td>
<td>t – 24/168 Week</td>
<td>Day</td>
<td>Feature extraction based on KSVM</td>
<td>t + 24/168 Week</td>
</tr>
<tr>
<td>Classification features</td>
<td>t + 24/168 classification ID</td>
<td></td>
<td>Feature extraction based on KSVM</td>
<td></td>
</tr>
</tbody>
</table>
The KSVM–TCN–GBRT method processing steps for power markets are shown in Figure 3. The short-term demand forecasting framework based on the KSVM–TCN–GBRT method is shown in Figure 4.

In this paper, parameters are selected automatically for RBFkernel-SVM by the means of temperature and wind speed. Temperature and wind speed are the main influencing factors in the new energy power market, including photovoltaic and wind power. In this paper, temperature and wind speed are proposed for measuring real-time demand class dissociative in the feature space. The same classes’ temperatures and wind speeds are as close as possible. For the different classes’ temperatures and wind speeds, the greater the distance that can be
created between them, the better. Hence, the mean of values applied by the normal kernel function on the samples in the same class is as shown in equation (13):

\[
\beta = \frac{1}{\sum_{i=1}^{L} |\Omega_i|^2} \sum_{i=1}^{L} \sum_{x \in \Omega_i} \sum_{x \in \Omega_i} \kappa(x, z, \gamma),
\]

(13)

where \(|\Omega_i|\) is the samples in class \(i\). \(\beta\) is \(w(\gamma)\), which is close to 1 [29]. The RBF kernel function is as in the following equation:

\[
b(\gamma) = \frac{1}{\sum_{i=1}^{L} \sum_{j \neq i} |\Omega_i||\Omega_j|^2} \sum_{i=1}^{L} \sum_{j \neq i} \sum_{x \in \Omega_i} \sum_{x \in \Omega_j} \kappa(x, z, \gamma).
\]

(14)
Hence, $\gamma$ should be determined such that it is closest to $b(\gamma)$. The same classes are closest to 0. The different classes are closest to 1.
It can be seen that $0 \leq w(c) \leq 1$ and $0 \leq b(c) \leq 1$ if $k(x, y, c) \geq t$, which is equivalent to the optimization problem, as in the following equation:

$$
\min_{\beta} f(y) = (1 - w(y)) + (b(y) - 0) = 1 - w(y) + b(y).
$$

(15)

In this paper, the KSVM–TCN–GBRT parameter settings are shown in Table 2.

Consider the impact of meteorological factors such as temperature and wind on new energy sources such as photovoltaics and wind power. In this paper, hourly

### Table 3: The comparison parameter settings.

<table>
<thead>
<tr>
<th>Name</th>
<th>Layer (type)</th>
<th>Output shape</th>
<th>Param #</th>
<th>Connected to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input (InputLayer)</td>
<td>[None, 5, 2]</td>
<td>0</td>
<td>input_1[0][0]</td>
<td></td>
</tr>
<tr>
<td>conv1d (Conv1D)</td>
<td>(None, 5, 16)</td>
<td>48</td>
<td>conv1d[0][0]</td>
<td></td>
</tr>
<tr>
<td>max_pooling1d (MaxPooling1D)</td>
<td>(None, 1, 16)</td>
<td>17</td>
<td>max_pooling1d[0][0]</td>
<td></td>
</tr>
<tr>
<td>Dropout (dropout)</td>
<td>(None, 1, 16)</td>
<td>48</td>
<td>input_1[0][0]</td>
<td></td>
</tr>
<tr>
<td>Dense (dense)</td>
<td>(None, 1, 16)</td>
<td>17</td>
<td>multiply[0][0]</td>
<td></td>
</tr>
<tr>
<td>input (InputLayer)</td>
<td>[None, 5, 2]</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>conv1d (Conv1D)</td>
<td>(None, 5, 16)</td>
<td>48</td>
<td>input_1[0][0]</td>
<td></td>
</tr>
<tr>
<td>max_pooling1d (MaxPooling1D)</td>
<td>(None, 1, 16)</td>
<td>17</td>
<td>multiply[0][0]</td>
<td></td>
</tr>
<tr>
<td>Dropout (dropout)</td>
<td>(None, 1, 16)</td>
<td>48</td>
<td>input_1[0][0]</td>
<td></td>
</tr>
<tr>
<td>Dense (dense)</td>
<td>(None, 1, 16)</td>
<td>17</td>
<td>multiply[0][0]</td>
<td></td>
</tr>
<tr>
<td>input (InputLayer)</td>
<td>[None, 5, 2]</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>conv1d (Conv1D)</td>
<td>(None, 5, 16)</td>
<td>48</td>
<td>input_1[0][0]</td>
<td></td>
</tr>
<tr>
<td>max_pooling1d (MaxPooling1D)</td>
<td>(None, 1, 16)</td>
<td>17</td>
<td>multiply[0][0]</td>
<td></td>
</tr>
<tr>
<td>Dropout (dropout)</td>
<td>(None, 1, 16)</td>
<td>48</td>
<td>input_1[0][0]</td>
<td></td>
</tr>
<tr>
<td>Dense (dense)</td>
<td>(None, 1, 16)</td>
<td>17</td>
<td>multiply[0][0]</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4: Real-time demand projections on March 31, 2021.

<table>
<thead>
<tr>
<th>Predicted time</th>
<th>Actual value (MW)</th>
<th>CNN-LSTM</th>
<th>CNN-LSTM-attention</th>
<th>CNN-Bi-LSTM-attention</th>
<th>TCN</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0:00</td>
<td>2340</td>
<td>2697</td>
<td>357</td>
<td>2627</td>
<td>287</td>
<td>2779</td>
</tr>
<tr>
<td>1:00</td>
<td>2263</td>
<td>2564</td>
<td>301</td>
<td>2380</td>
<td>417</td>
<td>2471</td>
</tr>
<tr>
<td>2:00</td>
<td>2219</td>
<td>2469</td>
<td>250</td>
<td>2286</td>
<td>67</td>
<td>2369</td>
</tr>
<tr>
<td>3:00</td>
<td>2204</td>
<td>2404</td>
<td>200</td>
<td>2282</td>
<td>78</td>
<td>2311</td>
</tr>
<tr>
<td>4:00</td>
<td>2266</td>
<td>2360</td>
<td>94</td>
<td>2310</td>
<td>44</td>
<td>2291</td>
</tr>
<tr>
<td>5:00</td>
<td>2453</td>
<td>2369</td>
<td>84</td>
<td>2375</td>
<td>78</td>
<td>2360</td>
</tr>
<tr>
<td>6:00</td>
<td>2766</td>
<td>2539</td>
<td>227</td>
<td>2579</td>
<td>187</td>
<td>2571</td>
</tr>
<tr>
<td>7:00</td>
<td>2996</td>
<td>2788</td>
<td>208</td>
<td>2889</td>
<td>107</td>
<td>2827</td>
</tr>
<tr>
<td>8:00</td>
<td>3090</td>
<td>2950</td>
<td>140</td>
<td>3026</td>
<td>64</td>
<td>2999</td>
</tr>
<tr>
<td>9:00</td>
<td>3111</td>
<td>3052</td>
<td>59</td>
<td>3081</td>
<td>30</td>
<td>3037</td>
</tr>
<tr>
<td>10:00</td>
<td>3118</td>
<td>3242</td>
<td>124</td>
<td>3115</td>
<td>3</td>
<td>3111</td>
</tr>
<tr>
<td>11:00</td>
<td>3090</td>
<td>3355</td>
<td>265</td>
<td>3093</td>
<td>3</td>
<td>3154</td>
</tr>
<tr>
<td>12:00</td>
<td>3076</td>
<td>3409</td>
<td>333</td>
<td>3062</td>
<td>14</td>
<td>3129</td>
</tr>
<tr>
<td>13:00</td>
<td>3083</td>
<td>3312</td>
<td>229</td>
<td>3061</td>
<td>22</td>
<td>3207</td>
</tr>
<tr>
<td>14:00</td>
<td>3013</td>
<td>3265</td>
<td>252</td>
<td>3240</td>
<td>227</td>
<td>3218</td>
</tr>
<tr>
<td>15:00</td>
<td>3004</td>
<td>3237</td>
<td>233</td>
<td>3151</td>
<td>147</td>
<td>3194</td>
</tr>
<tr>
<td>16:00</td>
<td>3103</td>
<td>3134</td>
<td>31</td>
<td>3027</td>
<td>76</td>
<td>3097</td>
</tr>
<tr>
<td>17:00</td>
<td>3215</td>
<td>3125</td>
<td>90</td>
<td>3036</td>
<td>179</td>
<td>3087</td>
</tr>
<tr>
<td>18:00</td>
<td>3243</td>
<td>3169</td>
<td>74</td>
<td>3168</td>
<td>75</td>
<td>3146</td>
</tr>
<tr>
<td>19:00</td>
<td>3252</td>
<td>3207</td>
<td>45</td>
<td>3218</td>
<td>34</td>
<td>3206</td>
</tr>
<tr>
<td>20:00</td>
<td>3138</td>
<td>3209</td>
<td>71</td>
<td>3157</td>
<td>19</td>
<td>3172</td>
</tr>
<tr>
<td>21:00</td>
<td>2922</td>
<td>3320</td>
<td>398</td>
<td>3239</td>
<td>317</td>
<td>3256</td>
</tr>
<tr>
<td>22:00</td>
<td>2663</td>
<td>3154</td>
<td>491</td>
<td>3144</td>
<td>481</td>
<td>3127</td>
</tr>
<tr>
<td>23:00</td>
<td>2441</td>
<td>2837</td>
<td>396</td>
<td>2803</td>
<td>362</td>
<td>2859</td>
</tr>
</tbody>
</table>

**MAPE (%)** 0.1844 0.1806 0.1876 0.1752 0.0315

**MAX APE (MW)** 491 481 464 410 84
datasets are extracted to create quarterly and similar day datasets based on actual electricity spot market data and weather data. This paper classifies real-time electricity demand by numerical intervals, using wind speed and temperature as the main characteristics and using 100 as the unit of measurement. A classification method using RBF SVMs with hyperparameters is used to find the optimum using numerical optimization. The wind speed and temperature are divided into groups that are particularly close to each other; the greater the difference between the different groups, the better. This master feature classification sequence is fed as a feature column into the succeeding deep learning neural network. KSVM can effectively solve machine learning problems with small samples and has good generalization ability; it can compensate for the problems of neural network structure selection and local minima.

In the electricity market, a real-time electricity demand forecast is a multivariate time series consisting of the day-ahead demand and day-ahead price. In this paper, a multivariate TCN model is developed for supervised learning, and the dynamic relationships between its variables are extracted. We take the $T$ time series of the day-ahead demand and day-ahead price of the day-ahead market as the cause and the time series of real-time demand series $T+24$ and $T+168$ of the real-time market as the effect. By increasing the number of layers, changing the expansion coefficients, revising the filters, and adjusting the length of the historical sequence, we can avoid the gradient dispersion and gradient explosion problems in the RNN model prediction, and longer-term memory and dynamic analysis capabilities can be obtained. Also, as a convolutional structure, TCNs can slide a one-dimensional convolutional kernel to receive inputs of an arbitrary length and can be massively processed in parallel for faster training and verification. This effectively guarantees the timeliness of power prediction.

In this paper, one hot numerical treatment of meteorological features and date features is used to process the electricity market dataset into a non-high-dimensional non-sparse set of values suitable for GBRT forecasting. GBRT is integrated learning, using decision trees as weak classifiers and iterative learning based on the residuals of the decision tree predictions. It allows the GBRT model to be highly interpretable and robust, automatically discovering higher-order relationships between day-ahead market sequences and real-time demand characteristics.

2.5. Performance Evaluation. In this paper, absolute error (APE) is shown in equation (15), mean absolute percentage error (MAPE) is shown in equation (16), root mean square error (RMSE) is shown in equation (17), and mean absolute error (MAE) is shown in equation (18). The prediction errors and definitions are as follows:
APE \equiv \frac{P_{\text{pre}} - P_{\text{real}}}{P_{\text{real}}}, \quad (16) \\

MAPE = \frac{1}{N} \sum_{i=1}^{N} \frac{|P_{i}^{\text{pre}} - P_{i}^{\text{real}}|}{P_{i}^{\text{real}}}, \quad (17) \\

RMSE = \sqrt{\frac{\sum_{i=1}^{N} (P_{i}^{\text{pre}} - P_{i}^{\text{real}})^2}{N}}, \quad (18) \\

MAE = \frac{\sum_{i=1}^{N} |P_{i}^{\text{pre}} - P_{i}^{\text{real}}|}{N}, \quad (19)

where $N$ is the total number of test datasets, $P_{i}^{\text{pre}}$ is the $i$-th demand prediction, and $P_{i}^{\text{real}}$ is the $i$-th demand prediction.

### 3. Results and Discussion

#### 3.1. Experimental Settings

The hardware environment was Intel(R) Core(TM) i5-7200U CPU @2.70 GHz, 64 bit, 8GB DDR4 RAM memory, NVIDIA GeForce 940MX. The software development environment was Python 3.7, Tensorflow2.2.

The electricity spot market hourly dataset was obtained from ISO-NE, in the Connecticut region [27]. The meteorological data were derived from [28], and the names of weather stations were Windsor Locks and Bridgeport. The dataset spans January 1, 2016, to March 31, 2021, at an hourly frequency of 46009 moments.

The real-time demand in 2019 is shown in Figure 5. The four series of forecast dates are represented in order from top to bottom: original series, trend series, seasonal series, and residual series.

The maximum real-time demand at different temperatures is shown in Figure 6. The maximum real-time demand at different wind speeds is shown in Figure 7.

In this paper, the comparison method parameter settings are shown in Table 3.

#### 3.2. Experiment 1: Real-Time Demand Forecast for the Next Day

In this experiment, the real-time power demand was predicted for a day on March 31, 2021, at an hourly frequency of 24 moments. The training dataset includes data from January 1, 2016, to March 30, 2021. The test set has data from March 31, 2020.

Real-time demand projections from March 1, 2021, to March 31, 2021, are shown in Table 4. The result of the comparison method is shown in Table 5. Figure 8 shows the

![Figure 9: Real-time demand forecast on March 31, 2021.](image-url)
results of the KSVM-based real-time demand classification, considering the forecast results for temperature and wind speed for March 31, 2021. Figure 9 shows the comparison between the predicted and actual demand power for each model from March 30, 2021. It can be seen that the proposed models were able to match the actual demand power in their forecasts. In particular, the proposed models could capture occasional fluctuations.

3.3. Experiment 2: Real-Time Demand Forecast for the Next Week. In this experiment, the real-time power demand was predicted for a week, from March 25, 2021, to March 31, 2021, with 168 moments. The training dataset includes data from January 1, 2016, to March 24, 2021. The test set includes data from March 25, 2021, to March 31, 2021.

The result of comparison method is shown in Table 6. Figure 10 shows the results of the KSVM-based real-time demand classification, considering the forecasted temperature and wind speed from March 25, 2021, to March 31, 2021. Figure 11 shows the comparison between the predicted and actual demand power for each model from March 31, 2021. It also can be seen that the proposed models were able to match the actual demand power better than other forecasts.

4. Conclusions

This paper proposed a novel method for short-term demand forecasting in power markets based on KSVM–TCN–GBRT. The advantages of this method over previous methods are as follows:

1. A data-driven method for short-term demand forecasting based on KSVM–TCN–GBRT was designed and improved, and the temperature and the wind speed were proposed for measuring the real-time demand to improve accuracy in forecasting market demand.

2. We adopted a model structure consisting of data classification, a time-convolutional network, and an integrated forecasting model for daily and weekly
forecasts. Our proposed model can do multistep forecasting and improve the accuracy by focusing on each feature differently.

(3) CNN-LSTM, LSTM with the attention mechanism, bidirectional LSTM, and TCN were used for forecasting and comparative analysis, and the operational results indicated that the proposed prediction method can reduce the prediction error and improve the prediction accuracy.

Data Availability

The data of the models and algorithms used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by National Key R&D Program of China (no. 2016YFB0900100).

References


12 Computational Intelligence and Neuroscience


[27] https://www.iso-ne.com/.
