

Research Article

Very Short-Term Load Forecasting Using Hybrid Algebraic Prediction and Support Vector Regression

Marlon Capuno, Jung-Su Kim, and Hwachang Song

Department of Electrical and Information Engineering, Seoul National University of Science & Technology, Seoul 01811, Republic of Korea

Correspondence should be addressed to Hwachang Song; hcsong@seoultech.ac.kr

Received 12 December 2016; Revised 6 March 2017; Accepted 23 March 2017; Published 19 April 2017

Academic Editor: Alessandro Mauro

Copyright © 2017 Marlon Capuno 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.

This paper presents a model for very short-term load forecasting (VSTLF) based on algebraic prediction (AP) using a modified concept of the Hankel rank of a sequence. Moreover, AP is coupled with support vector regression (SVR) to accommodate weather forecast parameters for improved accuracy of a longer prediction horizon; thus, a hybrid model is also proposed. To increase system reliability during peak hours, this prediction model also aims to provide more accurate peak-loading conditions when considerable changes in temperature and humidity happen. The objective of going hybrid is to estimate an increase or decrease on the expected peak load demand by presenting the total MW per Celsius degree change (MW/C°) as criterion for providing a warning signal to system operators to prepare necessary storage facilities and sufficient reserve capacities if urgently needed by the system. The prediction model is applied using actual 2014 load demand of mainland South Korea during the summer months of July to September to demonstrate the performance of the proposed prediction model.

1. Introduction

Load forecasting plays an important role in an efficient operation and planning of power system to maintain system stability, security, reliability, and economics. Operational decisions in power systems, such as economic dispatch, unit commitment, maintenance scheduling, reducing spinning reserve, automatic generation control, and reliability analysis, depend on the future behavior of loads. Therefore, accurate load forecasting helps the electric utility to make these operation decisions appropriately [1].

Substantial deviation from actual load demand may lead to allocation of insufficient reserve capacity resulting in limited supply of electricity at the consumer end, which leads to energy quality reduction in system reliability and load shedding such as the rolling blackout that happened last September 15, 2011, around the neighborhoods of Seoul, Busan, and other major cities [2]. On the other hand, an overestimation may cause unnecessary investments or establishments which run under capacity and therefore result in uneconomic operating conditions. One of the major objectives of this paper is

to help prevent the former problem from happening by providing a warning signal to the system operators if a large mismatch between the actual load and forecasted day-ahead load demand would take place and give them enough time to prepare necessary storage facilities and sufficient reserves to satisfy system security and reliability issues that might occur. This is the very purpose of very short-term load forecasting (ranging from minutes to hours), to support real-time control and security evaluation [3].

Thus, the application of the very short-term load prediction concept serves not only for advanced security evaluation of the system but also for assisting in real-time scheduling of large storage facilities. Figure 1 depicts an overview of very short-term load prediction application for real-time control. Actual load demand and weather-related factors such as ambient temperature and humidity are important input parameters to produce a more accurate forecast for the next few hours. The updated load forecast will then be easily compared with the offline day-ahead load prediction to assist in real-time control and update the scheduling of reserve and storage facilities as needed.

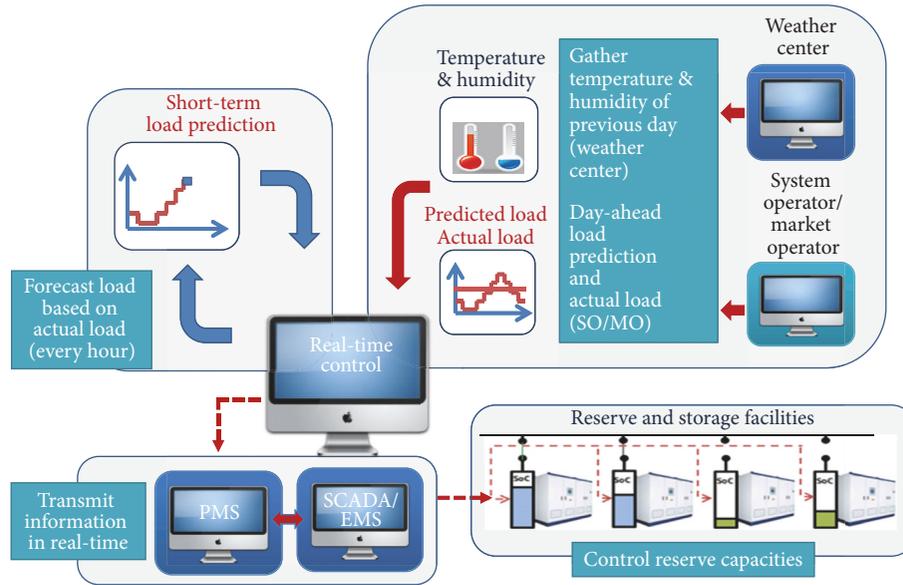


FIGURE 1: Overview of very short-term load prediction for online application.

An exhaustive review of recent studies and previous forecasting methodologies can be found in [4] pertaining to short-term load forecasting (STLF), that is, hours to weeks ahead prediction. But the focus of this paper is mainly on very short-term load forecasting (VSTLF), that is, minutes to hours ahead horizon focusing on peak-loading conditions. Different VSTLF methods reported in literature include methods of persistence, extrapolation, time series, fuzzy logic, Kalman filtering, neural networks, and support vector regression. Persistence forecasting is the simplest method where it assumes that the forecast data will be the same as the last measured values [5]. This is not practical for VSTLF because very short-term load series change in real-time. Extrapolation predicts the load based on the past by using a least-square algorithm [6] or by using a curve fitting algorithm based on a shape similarity criterion [7]. Similar to the extrapolation method, the autoregression method uses a simple linear combination of the previous load series for prediction. Its coefficients were tuned online using the least mean square algorithm in [8]. The method was extended to autoregressive integrated moving average (ARIMA) for load forecasting, and parameters were updated via a recursive least-square algorithm with a forgetting factor in [9]. ARIMA was extended to seasonal autoregressive integrated moving average to capture the seasonal load feature in [10]. Kalman filter was also applied to VSTLF where loads were separated into deterministic and stochastic components, and both were predicted via Kalman filters in [11]. Fuzzy logic methods convert input data to fuzzy values which are then compared with patterns extracted from the training process. The most similar fuzzy value was chosen and then mapped to the prediction in [8].

Finally, support vector regression (SVR) was developed for VSTLF, which was used with kernel functions to create

complex nonlinear decision boundaries in [12]. This paper exploits a different application of SVR as a compensator for the main load prediction, thus coming up with a hybrid prediction model. The hybrid model utilizes the proposed algebraic prediction (AP) as the main forecasting algorithm for an iterative hour-ahead prediction. The main objective is to come up with the most accurate prediction of the peak-loading conditions for online application. The system then identifies if a significant difference between previous day weather data and current weather forecast is found; then it calls upon the implementation of SVR to adjust the initial peak load output from the AP and compensate for the expected change or behavior of the load.

2. Methodology

This section breaks down the proposed hybrid solution method for very short-term load forecasting system model.

Problem formulation and implementation for both solution methods will be discussed in detail starting with the AP followed by SVR and how weather-related parameters affect the prediction results.

Figure 2 illustrates a simple flowchart and overview of the hybrid solution method. First, an initial forecast is done using AP. Then the weather conditions between the current day and the previous day are evaluated if a significant change is found; if yes, the SVR is implemented to update the initial prediction by adjusting the peak-loading conditions; if no, the initial prediction is deemed final.

2.1. Algebraic Prediction. Short-term time series forecasting has been introduced in [13] based on the identification of skeleton algebraic sequences. It has the objective of building a

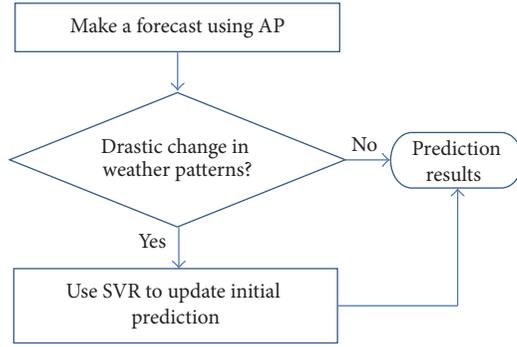


FIGURE 2: Overview of hybrid AP-SVR prediction method.

model of the process and then projecting this model into the future. This algebraic prediction technique is developed based on the rank of the Hankel matrix to identify the skeleton of the algebraic sequence. The Hankel matrix, named after Hermann Hankel, is widely used for system identification when given sequence of output data when a realization of an underlying state-space model is desired. It has been the key tool for solving the state-space realization problem since 1965 as provided in [14]. The concept of Hankel rank of a sequence is proposed in [15] for the identification of a numerical sequence. But it is important to note that the Hankel rank of a sequence is a concept independent from the state-space realization of the system. The Hankel rank just describes algebraic relationships between elements of the sequence without pretending to approximate the analytical model of an underlying dynamical system [13]. Also, these algebraic relationships are exact as illustrated in the following Hankel matrix construction and formulation.

For a given number of observations with $2n$ elements as expressed in (1), one could build a model of the process to extrapolate the past behavior into the future.

$$p_{k-2n}, p_{k-(2n-1)}, p_{k-(2n-2)}, \dots, p_{k-1}. \quad (1)$$

In (1), p_{k-1} is the value of the observation at the present moment. Assuming that the sequence is an algebraic progression and its Hankel rank is equal to n , it is possible to determine the next element of the sequence p_k in the form of a Hankel matrix having an ascending skew-diagonal from left to right as shown in

$$\det \begin{bmatrix} p_{k-2n} & p_{k-(2n-1)} & \cdots & p_{k-(n+1)} & p_{k-n} \\ p_{k-(2n-1)} & p_{k-(2n-2)} & \cdots & p_{k-n} & p_{k-(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ p_{k-(n+1)} & p_{k-n} & \cdots & p_{k-2} & p_{k-1} \\ p_{k-n} & p_{k-(n-1)} & \cdots & p_{k-1} & p_k \end{bmatrix} = 0. \quad (2)$$

With p_k as the only unknown in the series, (2) or the determinant of the Hankel matrix equated to zero can be

expressed and broken down as the following set of linear algebraic equations to solve for p_k :

$$\begin{bmatrix} p_{k-2n} & p_{k-(2n-1)} & \cdots & p_{k-(n+1)} \\ p_{k-(2n-1)} & p_{k-(2n-2)} & \cdots & p_{k-n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{k-(n+1)} & p_{k-n} & \cdots & p_{k-2} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \quad (3)$$

$$= \begin{bmatrix} p_{k-n} \\ p_{k-(n-1)} \\ \vdots \\ p_{k-1} \end{bmatrix},$$

$$[p_{k-n} \ p_{k-(n-1)} \ \cdots \ p_{k-1}] [a_1 \ a_2 \ \cdots \ a_n]^T = p_k. \quad (4)$$

This is the general formulation for algebraic progressions such as arithmetic progression and geometric progression. Unfortunately, such formulation does not apply to real life time series because of existing disturbance and noise in the actual systems.

2.1.1. Forecasting Strategy. To create a more practical model of prediction using the concept of Hankel rank of a matrix, (3) is improved to identify a better representation of the skeleton algebraic sequence in order to predict a more accurate future value of a time series. Equation (3) is modified as follows:

$$\begin{bmatrix} \vdots \\ p_{k-(2n+1)} & p_{k-2n} & \cdots & p_{k-(n+2)} \\ p_{k-2n} & p_{k-(2n-1)} & \cdots & p_{k-(n+1)} \\ p_{k-(2n-1)} & p_{k-(2n-2)} & \cdots & p_{k-n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{k-(n+1)} & p_{k-n} & \cdots & p_{k-2} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \quad (5)$$

$$= \begin{bmatrix} \vdots \\ \frac{p_{k-(n+1)}}{p_{k-n}} \\ p_{k-(n-1)} \\ \vdots \\ p_{k-1} \end{bmatrix}$$

with (4) still retained to solve for the value of p_k .

Notice that additional rows are augmented above the original matrices to allow for more information to increase the accuracy of the prediction. We can simplify (5) using the following formula representation:

$$\mathbf{P}_{m \times n} \mathbf{a}_{n \times 1} = \mathbf{p}_{m \times 1}. \quad (6)$$

It is critical that the number of measurements (m) exceeds the number of system parameters (n); that is, $m > n$. Thus,

the measurement error can be filtered out in the estimation process and good quality estimates can be obtained. In the method of linear least error squares estimation, the objective is to minimize the sum of the squares of the errors or residuals. Using the customary process of getting the pseudo-inverse, we get the following equation:

$$\hat{\mathbf{a}} = [\mathbf{P}^T \mathbf{P}]^{-1} \mathbf{P}^T \mathbf{p}, \quad (7)$$

where $[\mathbf{P}^T \mathbf{P}]^{-1} \mathbf{P}^T$ is called the left pseudo-inverse of \mathbf{P} and $\hat{\mathbf{a}}$ is the optimal or best least-square estimate of \mathbf{a} .

The \mathbf{P} matrix implementation is very similar with building the Hankel matrix using the series of hourly data from the previous days. But the most critical part of maximizing the proposed methodology is by knowing the optimal value of n -row-vectors which has been proved to be the breakthrough for the implementation of the proposed prediction strategy.

The method of linear least error squares estimation finds the mean value of a set of measurements $\mathbf{a}_{n \times 1}$. The mean value is generally accepted to be the best estimate when the set of measurements has a Gaussian error distribution [4]. It is also adversely affected by the presence of bad data; that is why it is necessary to get as much historical data as possible to establish the best estimate based on the data provided. Hence, it is also important to disregard irrelevant data during data preparation and preprocessing. The proposed method is tested using a set of ten days to forty days of historical data to determine the adequate amount of data to be used.

2.2. Support Vector Regression. Support vector regression (SVR) is an extension of the support vector machine (SVM) algorithm for numeric prediction. SVM is a state-of-the-art machine learning algorithm that is applicable to classification tasks [16]. Using the training data, it finds the maximum margin hyperplane between two classes by applying an optimization method. The decision boundary is defined by a subset of the training data, called support vectors. By using a kernel function, nonlinear decision boundaries can be formed while keeping the computational complexity low.

SVR also produces a decision boundary that can be expressed in terms of a few support vectors and can be used with kernel functions to create complex nonlinear decision boundaries. Similar to linear regression (LR), SVR tries to find a function that best fits the training data. In contrast to LR, SVR defines a tube around the regression line using a user specified parameter ε where the errors are ignored and it also tries to maximize the flatness of the line (in addition to minimizing the error) [17].

The SVR formulation for time series prediction is expressed as follows. Given training data $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ where x_i are input vectors and y_i are the associated output value of x_i , then the SVR is an optimization problem.

$$\begin{aligned} \min \quad & \frac{1}{2} w^T w + C \sum_{i=1}^l (\xi_i + \xi_i^*), \\ \text{subject to} \quad & y_i - (w^T \varphi(x_i) + b) \leq \varepsilon + \zeta_i, \end{aligned}$$

$$\begin{aligned} & (w^T \varphi(x_i) + b) - y_i \leq \varepsilon + \xi_i^*, \\ & \zeta_i, \xi_i^* \geq 0, \quad i = 1, \dots, l, \end{aligned} \quad (8)$$

where x_i is mapped to a higher dimensional space and ζ_i is the upper training error (ξ_i^* is the lower) subject to the ε -insensitive tube $|y_i - (w^T \varphi(x_i) + b)| \leq \varepsilon$. The parameters which control the regression quality are the cost of error C , the width of tube ε , and the mapping function, φ [18].

The constraints of (8) imply that we would like to put most data x_i in the tube $|y_i - (w^T \varphi(x_i) + b)| \leq \varepsilon$. If x_i is not in the tube, there is an error ζ_i or ξ_i^* which we would like to minimize in the objective function. For traditional least-square regression, ε is always zero and data are not mapped onto higher dimensional spaces. Hence, SVR is a more general and flexible treatment on regression problems [19].

The main parameters in SVR are ε and C . As mentioned above, ε defines the error-insensitive tube around the regression function and thus controls how well the function fits the training data [20]. The parameter C controls the trade-off between training error and model complexity; a smaller C increases the number of training errors; a larger C increases the penalty for training errors and results in a behavior similar to that of a hard-margin SVM [21].

The principle of the SVR formulation is with similar respect to SVM and, once trained, the SVR will generate predictions using the following formula:

$$f(x) = \sum_{i=1}^l \theta_i \varphi(x, x_i) + b. \quad (9)$$

For the implementation of SVR in this paper, the input training data x_i 's and y_i 's represent the weather-related parameters and load deviation, respectively. Once the training of the input data pairs is prepared, the trained SVR module can be used to approximate the value of y_n given an input x_n .

2.3. Effect of Weather Variables. The lack of weather input into time series models usually limits their forecasting ability. Effect of weather variables such as temperature and humidity has been examined in [22–25]. A minimum distance measurement is used in [24] to identify the appropriate historical patterns of load and temperature readings to estimate network weights in a neural network approach. This is to overcome the problems of drastic changes in weather patterns. In a similar way, historical patterns of load, temperature, and humidity are identified as a basis for the training of SVR module. It is observed that when a significant change between the previous day and present day weather conditions is found, an increased error in prediction results using AP is also observed.

2.4. Implementation and Testing. Algebraic prediction (AP) using the concept of Hankel matrix has been implemented using the historical data of mainland South Korea for the year 2014. To show AP's performance in comparison with other forecasting models, AR, ARMA, and SVR methods are also

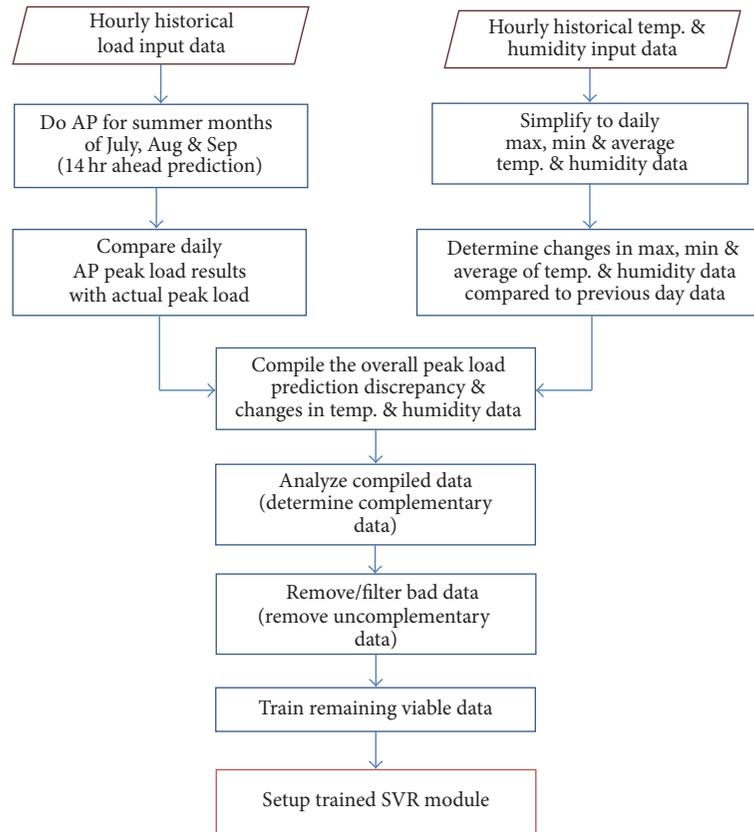


FIGURE 3: Setting up trained SVR module for peak load compensation.

implemented. Accordingly, AP's performance is also tested with increasing input of historical data as well as varying the value of n -row-vectors of the modified Hankel matrix that is surprisingly found to be critical in finding the most accurate prediction.

Finally, an iterative one-step forward prediction is implemented to assess the performance of each forecasting model during the days where significant changes in weather conditions are found.

2.5. Training the SVR Module. The flowchart in Figure 3 maps the process of building the SVR module in relation to the earlier illustration in Figure 2. The process of building the trained SVR module starts with preparing a fair amount of historical data to build a more reliable compensating module. Once the trained SVR module is built, the preliminary peak load output of the algebraic prediction method can be easily adjusted during days of significant change in weather conditions.

Step-by-step procedures are outlined as follows starting from setting up the SVR module.

Step 1. Gather the hourly historical data of load, temperature, and humidity of the past 2-3 years.

Step 2. Do an AP simulation on the summer (or winter) months on the gathered historical data.

Step 3. Based on the simulation results, identify the days where the discrepancy between actual and predicted peak load is significant.

Step 4. Verify the collected data by filtering the ones with corresponding significant deviation of weather parameters (i.e., ΔT_{MAX} , ΔT_{AVE} , and ΔH_{AVE}) between the current day and the previous day.

Step 5. Train the filtered data using SVR with ΔT_{MAX} , ΔT_{AVE} , and ΔH_{AVE} as input training set X and the normalized value of peak load discrepancies as output training set Y . Identify the appropriate kernel function as well as the optimal parameter settings for ϵ and C .

Step 6. Save the trained SVR module.

Consequently, the following procedure is done for real-time operation once the trained SVR module is prepared for peak load compensation.

Step 1. Make a 14-hour-ahead prediction using AP to determine the peak-loading conditions.

Step 2. Determine if the most recent weather forecast has significant deviation with the previous day weather conditions.

Step 3. If significant variations are found, proceed to Step 4. If none, use the initial AP results as the final peak load forecast.

Step 4. Predict the corresponding increase or decrease in the peak load using the trained SVR module with current ΔT_{MAX} , ΔT_{AVE} , and ΔH_{AVE} as input parameters.

Step 5. Update the AP results using the following formula for peak load compensation given as the set $P_1, P_2, \dots, P_i, \dots, P_N$ representing the predicted load for the next N hours:

$$P_{i,\text{new}} = P_0 + \frac{(P_i - P_0)(P_{\text{max}} + \Delta P_{\text{SVR}} - P_0)}{P_{\text{max}} - P_0}, \quad (10)$$

where P_0 is the same as P_1 as the initial prediction for hour 1, P_{max} is the peak load based on the AP results, ΔP_{SVR} is the peak load adjustment from the trained SVR module, and $P_{i,\text{new}}$ is the updated forecast at hour i based on SVR.

Step 6. Save the new set of hourly prediction with updated peak-loading conditions.

For the actual implementation, the only data available are the 2014 hourly historical demand of the mainland South Korea. So the trained SVR modules that can be built for summer season are based only on 3 months of summer data, that is, July, August, and September data. It was found that the optimal settings for the SVR training as kernel function are the radial basis function (RBF) with parameter values of $\varepsilon = 0.001$, $C = 10$.

For the input training sets X and Y used for the SVR building, X is a vector with three components ΔT_{MAX} , ΔT_{AVE} , and ΔH_{AVE} while Y is a normalized value from the set of all Y 's (peak load discrepancies). A pair of X and Y is considered to be a viable data when ΔT_{MAX} , ΔT_{AVE} are both positive along with a positive Y value because an increase in temperature during summer season means a positive increase as well in peak load demand.

While it is more reasonable to consider an hourly temperature and humidity data for the AP's output correction, only the maximum and average values were used as input in the SVR compensating module to make the actual implementation much easier to conduct and set up.

3. Simulation Results

The performance of AP is first tested in response to its sensitivity with increasing amount of historical data and varying the value of n -row-vectors of the input \mathbf{P} matrix. Figure 4 shows a summary of the least MAPE for each number of days of historical data considering different values of n -row-vectors. Similarly, Figure 5 shows how the MAPE changes with respect to the value of n -row-vectors of the \mathbf{P} matrix. When implementing the AP model, it is important to identify the optimal number of needed historical data for a specific system. It is also found by trial approach that n should be at least an integer greater than the period of the underlying sequence to get a decent prediction result (e.g., for an hourly interval, n should be at least 25 because there are 24 hours in a day; for 30-minute intervals, n should be at least 49 because

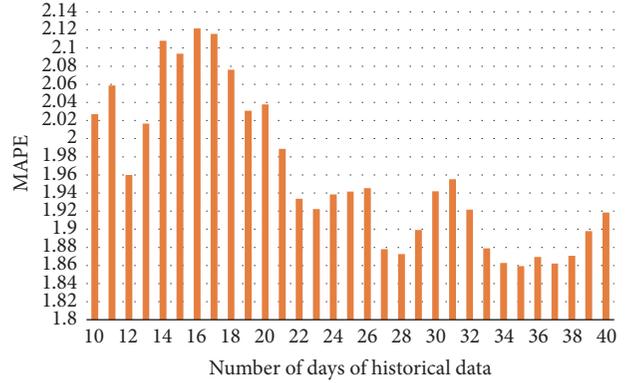


FIGURE 4: MAPE for varying days of input historical data for the month of August.

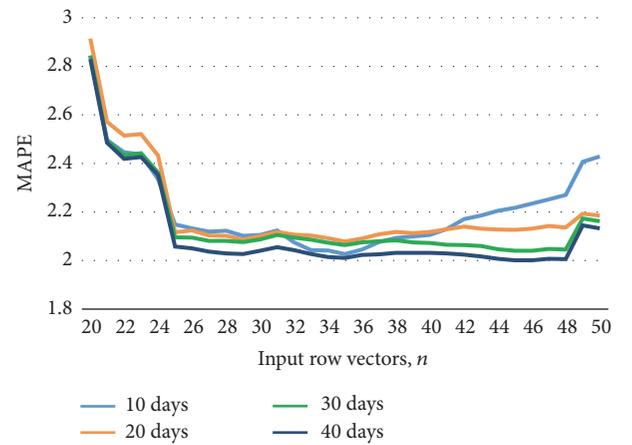


FIGURE 5: MAPE for different values of n -row-vectors of input matrix \mathbf{P} for 10, 20, 30, and 40 days of historical data.

TABLE 1: MAPE and RMSD comparison of different forecasting models using 20-day input during the month of August 2014.

Forecasting model	MAPE (%)	RMSD (MW)	Computation time (sec)
AP	2.1162	1807.0949	0.90
AR	2.3168	1971.6386	16
ARMA	2.3047	1988.9166	21
SVR	2.5917	2172.8847	180

there are 48 30-minute intervals in a day, etc.) Furthermore, using twice the period as the value of n (i.e., $n = 48$ for hourly prediction) is observed to yield a more accurate prediction according to the simulations conducted. These findings could be area specific so it is important to check if the same behavior could be observed in other locations.

Table 1 shows performance of AP in comparison with other forecasting models, that is, AR, ARMA, and SVR models. Initial test is conducted to compare the performance of both prediction methods. The table shows a summary of maximum absolute percentage error (MAPE), root mean square deviation (RMSD), and computation time or the time

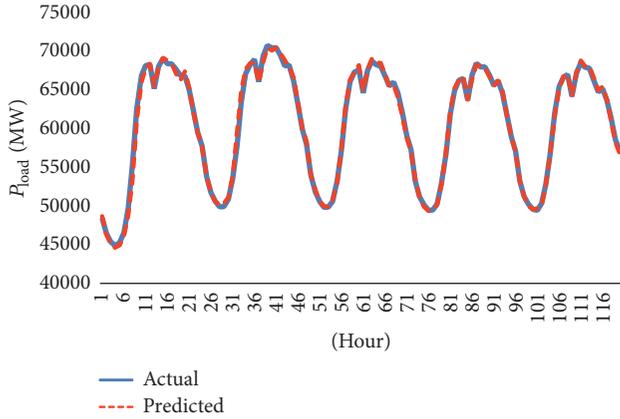


FIGURE 6: One-hour-ahead prediction from August 25–29.

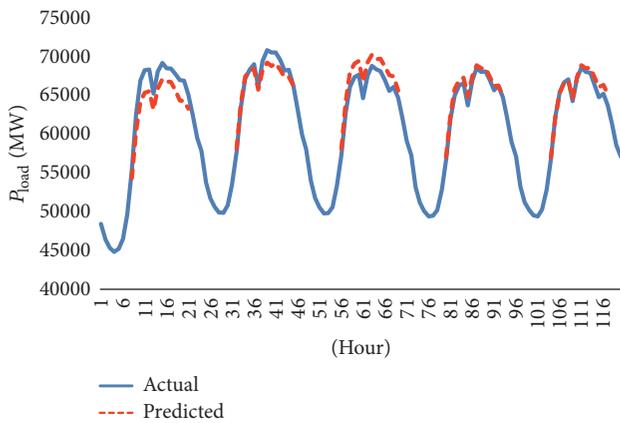


FIGURE 7: Fourteen-hour-ahead prediction from August 25–29.

it takes to finish a set of hourly load prediction. As seen in Table 1, the main advantage of using AP besides lower MAPE and RMSD is that it only takes less than a second in order to yield an output compared to other methods. This is very significant because we want to use as much historical data as possible for better accuracy. Therefore, AP is more practical for use in real-time application.

To better establish the effectiveness of AP, shown in Figure 6 is a sample implementation of one-hour-ahead load prediction for the final week of August. And to demonstrate further its performance, an iterative one-step prediction is implemented for the same week completing a 14-hour-ahead prediction (i.e., from 8 am to 9 pm) to predict the daily peak-loading conditions as shown in Figure 7.

The 14-hour-ahead prediction is performed for the whole month of August and it is found that 60% of the time, the peak load prediction is accurate enough. However, for the remaining 40%, a compensation module is needed to consider weather-related parameters such as forecasted temperature and humidity to obtain a more accurate peak-loading condition. Thus, an SVR compensating module is proposed on the application of 14-hour-ahead prediction to cover for the change in weather conditions.

From the summer months of July, August, and September for the year 2014, seventeen (17) viable pairs for training of

TABLE 2: Training set from the months July, August, and September 2014.

Date	X			Y
	ΔT_{MAX}	ΔT_{AVE}	ΔH_{AVE}	$\Delta P_{K,LOAD}$
July 1	2	0.920	-0.375	1341.633
July 3	-7	-2.710	7.625	-4110.892
July 9	4	1.540	5.167	1642.079
July 16	-1	-0.290	-1.625	-818.390
July 17	-2	-0.330	4.667	-1285.362
July 23	-2	-2.040	4.833	-2228.795
July 24	-1	-1.960	-0.250	-915.093
July 25	9	5.540	-5.500	3170.716
July 28	-4	-2.080	-3.958	-4190.624
Aug 4	-6	-3.790	25.208	-7426.996
Aug 12	2	1.917	0.458	2501.683
Aug 14	-3	-1.670	-3.875	-3083.127
Aug 27	-2	-1.208	-2.542	-1438.378
Sept 1	2	0.580	1.625	2719.843
Sept 2	-2	-0.460	-3.625	-1321.296
Sept 12	-1	-0.500	6.208	-1582.431
Sept 17	-4	-0.708	7.417	-2069.028

TABLE 3: Percentage error when each X from the training set is used as an input to the trained SVR module.

Date	Y_{ACTUAL} $\Delta P_{K,LOAD}$	Y_{TEST} $\Delta P_{K,LOAD}$	MW/C°	% error
July 1	1341.633	1338.462	669.231	0.236
July 3	-4110.892	-4107.722	586.817	0.077
July 9	1642.079	1638.908	409.727	0.193
July 16	-818.390	-821.561	821.561	-0.387
July 17	-1285.362	-1282.191	641.096	0.247
July 23	-2228.795	-2225.624	1112.812	0.142
July 24	-915.093	-918.264	918.263	-0.346
July 25	3170.716	2116.684	351.949	0.1
July 28	-4190.624	-4187.454	1046.863	0.076
Aug 4	-7426.996	-4224.748	1237.304	0.043
Aug 12	2501.683	2116.684	1249.256	0.127
Aug 14	-3083.127	-3079.957	1026.652	0.103
Aug 27	-1438.378	-1435.207	717.603	0.220
Sept 1	2719.843	2116.684	1358.336	0.1175
Sept 2	-1321.296	-1318.126	659.063	0.240
Sept 12	-1582.431	-1579.260	1579.260	0.200
Sept 17	-2069.028	-2065.858	516.464	0.153

the SVR module are identified. These are the days where a significant change in weather conditions is observed. These 17 pairs shown in Table 2 can be tested if suited to be a training set for the building of the SVR compensation.

One way to determine a good training set (given that the optimal SVR parameter settings are already known) is when each member of the set is chosen to be an input for the trained SVR module. All of these are accomplished by trial and error. Shown in Table 3 is the output Y_{TEST} when each X is chosen as an input compared to their output Y_{ACTUAL} .

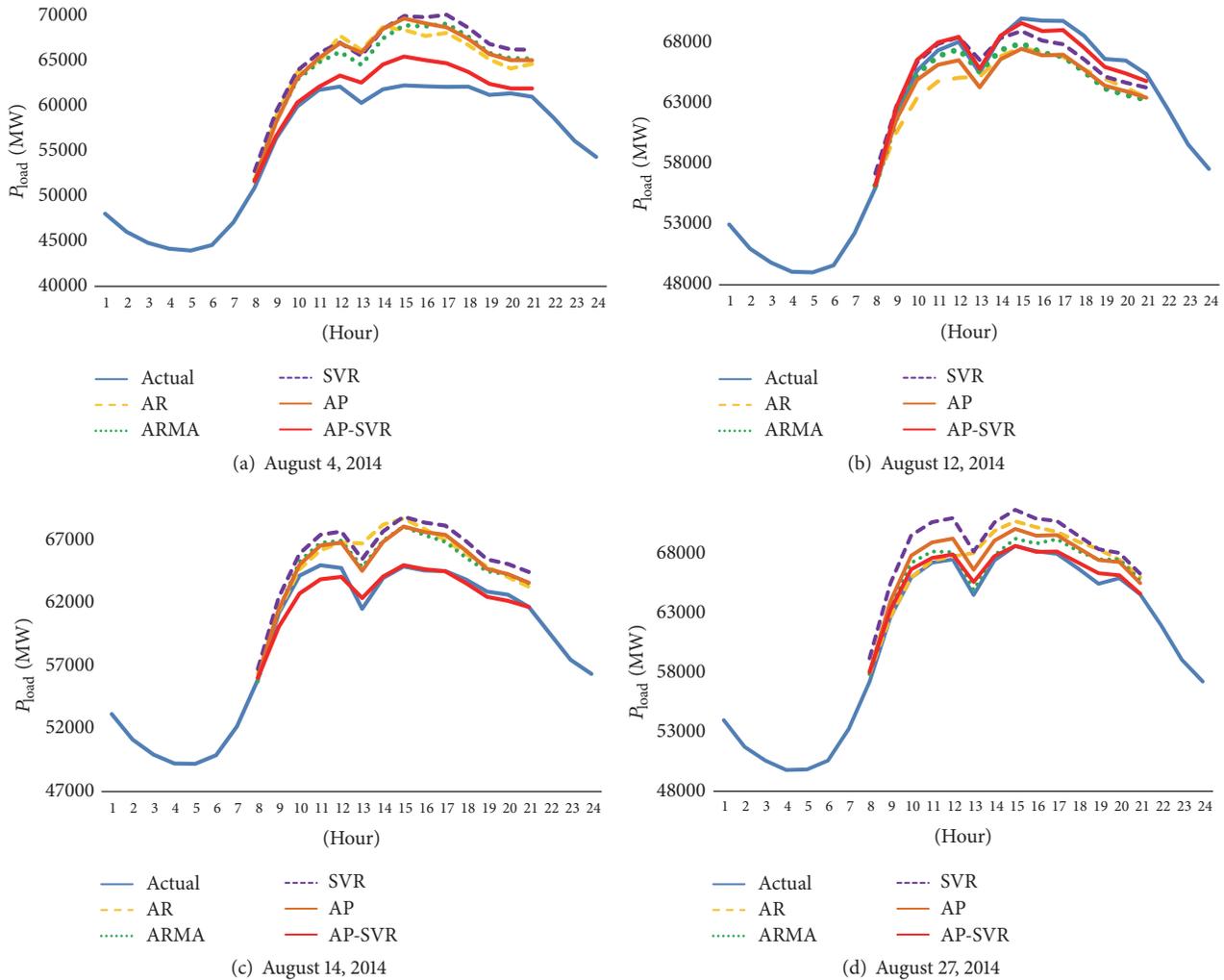


FIGURE 8: Comparison output prediction using AP, AR, ARMA, SVR, and hybrid AP-SVR models during days with significant weather changes.

From Table 3, it can be easily deduced that the error sample during the building of the SVR compensation module is zero, meaning an optimal SVR parameter settings help decrease the MAPE of the training set. Looking at the column “MW/C,” the sensitivity of loading conditions per degree change in temperature is seen to vary from 400 MW to 1.5 GW of load demand.

Using the peak load change based on the output of the SVR compensation module, AP output can be scaled down or up depending on the sign of the SVR output. Shown in Figure 8 are the AP-SVR prediction results as applied on August 4, 12, 14, and 27 in comparison with the other forecasting models.

4. Conclusion

A model for very short-term load forecasting applying a hybrid approach is presented. Algebraic prediction (AP) based on the concept of Hankel rank and linear least error

square estimation is coupled with support vector regression (SVR). The SVR is used to factor in weather-related parameters and improve the accuracy of algebraic prediction output. It is shown that from a set of viable pairs of historical data, an SVR compensating module can be built knowing the optimal SVR setting for the training set. Due to limited amount of historical data, the trained SVR module could not be tested for a different year. Furthermore, several test cases should still be done to expand the number of viable pairs to increase the reliability of the SVR module. Nonetheless, the feasibility of coupling SVR with AP’s output as compensation for changes in temperature and humidity is shown to have promising results given an adequate amount of historical data.

Conflicts of Interest

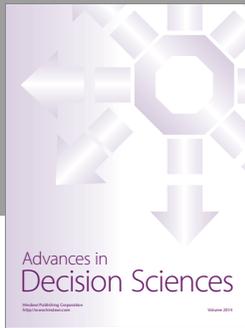
The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the Energy Efficiency & Resources of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) Grant (no. 20142010103010) and by the Human Resources Development of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) Grant (no. 20154030200720), funded by the Korea Government Ministry of Trade, Industry & Energy.

References

- [1] E. Ceperic, V. Ceperic, and A. Baric, "A strategy for short-term load forecasting by support vector regression machines," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4356–4364, 2013.
- [2] South Korea Blackout Risk Rises, <http://blogs.wsj.com/korearealtime/2013/08/12/urgency-for-south-koreans-to-cut-power-use-heightens/>.
- [3] A. Jain and B. Satish, "Integrated approach for short term load forecasting using SVM and ANN," in *Proceedings of the IEEE Region 10 Conference (TENCON '08)*, pp. 1–6, November 2008.
- [4] S. A. Soliman and A. M. Al-Kandari, *Electrical Load Forecasting: Modeling and Model Construction*, Elsevier, 2010.
- [5] B. Fox, D. Flynn, L. Bryans et al., *Wind Power Integration: Connection and System Operational Aspects*, IET Power and Energy Series, IET, 2007.
- [6] F. Wang, K. Xie, E. Yu, G. Liu, and M. Wang, "A simple and effective ultra-short term load forecasting method," *Power Systems Technology*, vol. 20, no. 3, pp. 41–43, 1996.
- [7] D. Luo and H. He, "A shape similarity criterion based curve fitting algorithm and its application in ultra-short-term load forecasting," *Power System Technology*, vol. 31, no. 21, pp. 81–84, 2007.
- [8] K. Liu, S. Subbarayan, R. Shoults et al., "Comparison of very short-term load forecasting techniques," *IEEE Transactions on Power Systems*, vol. 11, no. 2, pp. 877–882, 1996.
- [9] J. Lu, X. Zhang, and W. Sun, "A real-time adaptive forecasting algorithm for electric power load," in *Proceedings of the IEEE/PES Transmission and Distribution Conference and Exhibition: Asia and Pacific*, pp. 1–5, IEEE, Dalian, China, August 2005.
- [10] L. C. M. De Andrade and I. N. Da Silva, "Using intelligent system approach for very short-term load forecasting purposes," in *Proceedings of the IEEE International Energy Conference and Exhibition (EnergyCon '10)*, pp. 694–699, Manama, Bahrain, December 2010.
- [11] D. J. Trudnowski, W. L. McReynolds, and J. M. Johnson, "Real-time very short-term load prediction for power-system automatic generation control," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 2, pp. 254–260, 2001.
- [12] A. Setiawan, I. Koprinska, and V. G. Agelidis, "Very short-term electricity load demand forecasting using support vector regression," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '09)*, pp. 2888–2894, Atlanta, Ga, USA, June 2009.
- [13] M. Ragulskis, K. Lukoseviciute, Z. Navickas, and R. Palivonaite, "Short-term time series forecasting based on the identification of skeleton algebraic sequences," *Neurocomputing*, vol. 74, no. 10, pp. 1735–1747, 2011.
- [14] B. L. Ho and R. E. Kalman, "Effective construction of linear state-variable models from input-output functions," *Regelungstechnik*, vol. 14, no. 12, pp. 545–592, 1966.
- [15] Z. Navickas and L. Bikulciene, "Expressions of solutions of ordinary differential equations by standard functions," *Mathematical Modelling and Analysis*, vol. 11, no. 4, pp. 399–412, 2006.
- [16] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, 1998.
- [17] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2nd edition, 2005.
- [18] M.-W. Chang, B.-J. Chen, and C.-J. Lin, "Eunite Network Competition: Electricity Load Forecasting," Tech. Rep., National Taiwan University, Taipei, Taiwan, 2001.
- [19] A. J. Smola and B. Scholkopf, "A tutorial on support vector regression," Tech. Rep. NC2-TR-1998-030, NeuroCOLT2, 2003.
- [20] T. Joachims, *Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms*, Springer, Boston, Mass, USA, 2002.
- [21] F. Parrella, Online Support Vector Regression, <http://onlinesvr.altervista.org/>.
- [22] M. A. Abu-El-Magd and N. K. Sinha, "Short-term load demand modeling and forecasting: a review," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 12, no. 3, pp. 370–382, 1982.
- [23] K.-L. Ho, Y.-Y. Hsu, C.-F. Chen et al., "Short term load forecasting of taiwan power system using a knowledge-based expert system," *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1214–1221, 1990.
- [24] T. M. Peng, N. F. Hubele, and G. G. Karady, "Advancement in the application of neural networks for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 7, no. 1, pp. 250–257, 1992.
- [25] S.-T. Chen, D. C. Yu, and A. R. Moghaddamjo, "Weather sensitive short-term load forecasting using nonfully connected artificial neural network," *IEEE Transactions on Power Systems*, vol. 7, no. 3, pp. 1098–1105, 1992.




Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

