Knowledge-Based Systems 76 (2015) 139–147

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Short term load forecasting using a hybrid intelligent method

Adel Abdoos [a], Mohammad Hemmati [a], Ali Akbar Abdoos [b,*]

[a] Department of Management, Islamic Azad University, Semnan Branch, Semnan, Iran
[b] Faculty of Electrical & Computer Engineering, Babol Noshirvani University of Technology, Babol, Mazandaran, Iran

ARTICLE INFO

Article history:
Received 31 May 2014
Received in revised form 25 October 2014
Accepted 7 December 2014
Available online 15 December 2014

Keywords:
Short term load forecasting
Artificial intelligent
Support vector machine
Gram–Schmidt feature selection
Pattern recognition
Wavelet transform

ABSTRACT

Due to the regulation of electrical power systems, electricity market players need precise information of electrical energy consumption and generation in order to maximize their benefit based on appropriate decisions. In this paper a new hybrid intelligent method is proposed for short term load forecasting. In this method, load and temperature of previous days are used for prediction of the next hour electrical load consumption. Since electrical load signals are non-stationary, Wavelet Transform (WT) as a powerful signal analyzer is applied for the signal decomposing. For elimination of redundant data from input matrices, the Feature Selection (FS) method based on Gram–Schmidt (GS) is used for selection of more valuable features. The elimination of redundant data can speed up learning process and improve the generalization capability of the prediction scheme. Support Vector Machine (SVM) with simple structure and few tuning parameters is applied as a powerful regression tool. Two separate structures are considered for prediction of weekday and weekend electrical load consumption. Besides, in order to increase the forecasting accuracy, indices are determined for each day. The simulation results reveal that the Coiflet wavelet function with 2 decomposition levels lead to the best detection accuracy. Moreover, 30 dominant features of previous 50 days should be used to obtain minimum forecasting error. Comparative results show the priority of the proposed method in aspect of prediction accuracy as compared to some reported algorithms.

1. Introduction

By restructuring and deregulation of power system, management of electrical energy has become more important than before. Short term load forecasting can be useful for proper maintenance schedule and coordination of generation units. Moreover midterm and long term electrical load predictions can be applied for both future transmission and generation expansion planning. In competitive electricity market environment, the data of electricity energy consumption help both producers and consumers to make their decision so that they earn maximum revenue [1–3].

Time series-based methods like Auto Regressive Moving Average (ARMA) model and Auto Regressive Integrated Moving Average (ARIMA) model are very popular methods for load prediction. It uses the previous load data for prediction of next hour's loads [4,5]. But future electrical load are complicated and nonlinear functions of previous load and temperature data. Hence, some time series-based methods cannot yield precise prediction accuracy. Intelligent tools that are able to learn the relation between input and output vectors have been used in many researches for detection of electrical consumption load patterns. Feed Forward Artificial Neural Networks (FFANN) [6–8], Fuzzy Neural Network (FNN) [9–11], Support Vector Machines (SVMs) [12–15] and Expert Systems (ES) [16,17] have been applied in the load forecasting area.

Adaptive neural fuzzy inference system (ANFIS) based methods [15,23] need complicated fuzzy rules to give acceptable prediction accuracy. ANN also suffers from two major drawbacks: first, the learning process is a very time-consuming task, and second, there is no exact rule for setting the parameters of NNs.

Since electrical load consumption signals are non-stationary, signal analyzer tools can be applied for time series analysis. Some of these tools like Empirical Mode Decomposition (EMD) [18] analyzes signals only in time domain and some of them like Wavelet Transform (WT) [19] analyze signals in both time and frequency [20–26]. The combination of WT and learning machines has been applied for recognition of different pattern recognition problems such as electrical load forecasting [27]. In some combined algorithms [24,25], heuristic search algorithms have been applied for improvement of classifier operation by precise setting of adjustable parameters.

* Corresponding author. Tel.: +98 912 5314761.
E-mail addresses: adel_abdoos@yahoo.com (A. Abdoos), mo928hem@yahoo.com (M. Hemmati), abdoos_a@yahoo.com (A.A. Abdoos).

http://dx.doi.org/10.1016/j.knosys.2014.12.008
0950-7051/

In the most of proposed methods, historical data of load and temperature have been used for load prediction [8,23]. Since the value of all data is not the same, it is necessary to apply a feature selection method for elimination of less important features. This task improves the generalization capability of learning machine, and reduces both learning time and required memory for saving of data. Moreover, using two separate structures for prediction of weekdays and weekends as well as discriminative indices for different days can yield better prediction accuracy as compared to some papers which use a single predictor structure [8,23,24].

In this paper the electrical load signal is decomposed by WT into two levels. The obtained decomposed signals as well as temperature data create the training input matrices. But before applying learning machine on training matrices, dominant features are selected using Gram–Schmidt method [28,29] to reduce the dimension of input matrix. SVM [30,31] is used as the classifier core for learning of patterns of training matrix. The novelties of the proposed method can be mentioned as follows:

- With analysis of time series of electrical load consumption using WT, the load variation is scrutinized more precisely.
- By using feature selection method, the training matrix dimension decreases and generalization capability of learning machines increases.
- Proposing two separate structures for forecasting of load consumption of weekdays and weekend enhances the performance of predictors.
- Determination of indices for discrimination of days improves the prediction accuracy.
- Using SVM as the predictor core with simple structure and less tuning parameters as compared to ANN [23].

2. Required tools

2.1. Wavelet transform

WT is one the most powerful tools for analysis of non-stationary signals in time–frequency domains. This transform can decompose time series signals into different levels. Approximation level contains low frequency components and detail levels include high frequency components. Fig. 1 shows the decomposition tree of WT [19].

In this transform, different filters namely wavelet function are applied for decomposition process. $\phi(x)$ is wavelet function, if and only if it's Fourier transform $\Psi(\omega)$ satisfies the below condition [19]:

$$\int_{-\infty}^{+\infty} \frac{|\Psi(\omega)|}{|\omega^2|} d\Psi < +\infty \tag{1}$$

The above condition is known as admissibility for $\phi(x)$ that can be written as duality format as follow:

$$\Psi(0) = \int_{-\infty}^{+\infty} (\phi x) dx = 0 \tag{2}$$

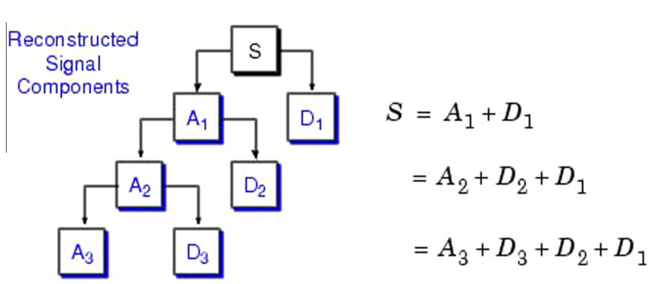

Fig. 1. WT decomposition tree.

Many functions can be found to satisfy above condition such as: Haar, Dabeches, Coiflet, Symlet, and Mexican hat. By using two mathematical operations of translation and dilation, location and magnitude of wavelet function change along the main signal as given below:

$$\phi_{a,b}(x) = \frac{1}{\sqrt{a}} \phi\left(\frac{x-a}{b}\right) \tag{3}$$

At last WT coefficient for each point (b) and for each scale (a) can be calculated according to (4)

$$CWT(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(x)\phi\left(\frac{x-a}{b}\right)dx = \int_{-\infty}^{+\infty} f(x)\phi_{a,b}(x)dx \tag{4}$$

2.2. Gram–Schmidt based feature selection

One of the fundamental subjects in the linear algebra is Gram–Schmidt (GS) orthogonalization [28]. It applies the QR method for decomposition of a matrix into two factorizations $X = QR$. The overview reveals that orthogonal basis present optimal option to perform calculations of the twist vector spaces. To achieve an orthonormal basis for an inner product space, the GS algorithm is applied to build an orthogonal basis. GS method is a formulation for the orthonormalization of a linearly independent set. It is assumed that there are N samples as $x(1), x(2), \ldots, x(N)$, and each sample denotes an n-dimensional vector $x(k) = [x_1(k), x_2(k), \ldots, x_n(k)]^T$. Feature vector x_i and feature matrix X are shown as [29]:

$$X = [x_1, \ldots, x_n] = \begin{bmatrix} x_1(1) & x_2(1) & \ldots & x_n(1) \\ x_1(2) & x_2(2) & \ldots & x_n(2) \\ & & \ddots & \vdots \\ x_1(N) & x_2(N) & & x_n(N) \end{bmatrix}, \quad x_i = [x_i(1), x_i(2), \ldots, x_i(N)]^T \tag{5}$$

The feature matrix X can be decomposed as given below:

$$X = QR \tag{6}$$

One of the important methods for solving linear equations is QR decomposition of matrices. QR decomposition states that if there is an X matrix where $X \in \mathbb{R}^{n*n}$, there exists an orthogonal matrix Q and an upper triangular matrix R that satisfy Eq. (6).

$$R = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \ldots & \alpha_{1n} \\ & \alpha_{22} & \ldots & \alpha_{2n} \\ & & \ddots & \vdots \\ & & & \alpha_{nn} \end{bmatrix},$$
$$Q = [q_1, q_2, \ldots, q_n] = \begin{bmatrix} q_1(1) & q_2(1) & \ldots & q_n(1) \\ q_1(2) & q_2(2) & \ldots & q_n(2) \\ & & \vdots & \vdots \\ q_1(N) & q_2(N) & \ldots & q_n(N) \end{bmatrix} \tag{7}$$

where q_i is the new feature vector in the orthogonal space. In the GS orthogonal decomposition, the orthogonal matrix Q is built using the following method:

$$q_1 = x_1 \tag{8}$$

$$q_i = x_i - \sum_{j=1}^{i-1} \alpha_{ji} q_j \tag{9}$$

where

$$\alpha_{ji} = \begin{cases} \frac{q_j^T x_i}{q_j^T q_i} & \text{for } j = 1, 2, \ldots, i-1 \\ 1 & \text{for } j = i \end{cases} \tag{10}$$