Electric Power Systems Research 143 (2017) 431–437

Contents lists available at ScienceDirect

Electric Power Systems Research

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

Boosted neural networks for improved short-term electric load forecasting

A.S. Khwaja*, X. Zhang, A. Anpalagan, B. Venkatesh

Department of Electrical and Computer Engineering, Ryerson University, Toronto, Canada

ARTICLE INFO

Article history:
Received 7 July 2016
Received in revised form
25 September 2016
Accepted 21 October 2016
Available online 15 November 2016

Keywords:
Short-term electric load forecasting
Artificial neural networks
Ensemble neural networks
Boosting

ABSTRACT

This paper presents an improved technique for short-term electric load forecasting making use of boosted neural networks (BooNN). The BooNN consist of combining a set of artificial neural networks (ANNs) trained iteratively. At each iteration, the error between the estimated output from the ANN model trained in the previous iteration and the target output is minimized. The final predicted result is the weighted sum of output from all the trained models. This process reduces the magnitude of forecasting errors and their variation compared to a single ANN and bagged neural networks (BNN). It further significantly lowers computational time compared to BNN. Results with real data further confirm that BooNN lead to improved load forecasting performance with respect to other existing techniques.

1. Introduction

Electric load forecasting is used for forecasting of future electric loads, given the information of historical load, weather, time, etc. For the independent system operators, there are many applications of load forecasting, including generation scheduling, generation of system reserve, providing information to the dispatcher, and market operation [1]. Accurate models of electric power load forecasting are necessary for the planning management and operations of a utility. Load forecasting has a significant role in helping an electric utility make important decisions on purchasing and generating electric power, load switching, voltage control, infrastructure development, etc. Load forecasts are also important for energy suppliers, financial institutions, and other actors involved in electric energy generation, transmission, distribution and marketing [2].

Load forecasting is categorized as long-term forecasting and short-term forecasting, based on the forecasting duration. Long-term load forecasting is for a span of several months to several years, whereas the forecasting carried out for a single day to a week ahead is usually referred to as short-term load forecasting. It plays an important part in power systems as increasing forecasting accuracy results in a decrease of operating costs and increased reliability of power system operations [3]. This paper concentrates on short-term electric load forecasting as it is required for making daily optimal decisions relating to the economic and secure operations of a power system [4].

Since the economy and reliability of operations of a power system are greatly affected by electric load, cost savings mainly depend on load forecasting accuracy. The load dispatcher at the main dispatch center is responsible for maintaining and controlling the flow of electricity through purchase, sale and scheduling of electricity. The dispatcher is required to have an estimate of the load pattern in advance so that sufficient generation can be allocated to meet the customers' requirements. Overestimating future load can cause starting of unnecessary generating units that will lead to increased reserve and operating costs. Underestimating future load will lead to a failure for providing operating reserve and stability to the system, which can cause a breakdown of the power system network [5].

Load forecasting consists of different methods such as conventional and artificial intelligence based methods. The former includes time series [6,7], multi-variate regression [8] and state estimation approaches, while the latter includes fuzzy logic [9,10], support vector machines (SVM) [11–13], artificial neural networks (ANN) methods [15,17,18] and methods that preprocess training data and then use multiple ANN for load forecasting using the preprocessed data [14,16]. Conventional methods have the advantage of simplicity, while artificial intelligence based methods have higher forecasting accuracy as they can accurately model the highly non-linear relationship between the observed load and the variables on which the observed load is dependent.

* Corresponding author at: Faculty of Engineering and Natural Sciences, Sabanci University, Tuzla, 34956 Istanbul, Turkey.
E-mail address: kashaharyar@gmail.com (A.S. Khwaja).

http://dx.doi.org/10.1016/j.epsr.2016.10.067
0378-7796/

In this paper, forecasting accuracy is improved making use of ensemble learning. In ensemble learning, a group of predictive models are combined to improve the prediction accuracy [19]. Theoretical and experimental studies have demonstrated that when the ANN in the group are accurate and each network's error is negatively correlated with errors for the rest of the group, an improved accurate generalization can be obtained by voting or averaging the output of ANN [20]. Two of the most popular algorithms in ensemble learning are bagging [21] and boosting [22,23]. Both algorithms aggregate the outputs of individual predictive models to improve the overall prediction accuracy. It has been demonstrated that bagging and boosting techniques can be more accurate than using individual predictive models. Bagging applied to ANN has been shown to improve load forecasting by reducing variance of load forecasting error compared to using a single ANN [24]. Boosting has also become popular for classification problems because many empirical studies show that it can yield lower classification error rates and is more robust to overfitting [25].

An improved short-term load forecasting technique using boosted ANN (BooNN) is presented in this paper. The BooNN technique is based on generating a number of ANN models iteratively, where at each iteration the resulting model reduces the error between expected output and that obtained from the trained models in the previous iterations. A forward stage-wise additive model for boosting regression [26] is used that updates the target output at each iteration by subtracting a weighted estimate from the target output at the previous iteration. This work's significance lies in the fact that, according to the best of authors' knowledge, boosting techniques have not been used for load forecasting using ANN. It is further shown with numerical results that this technique leads to a reduction in forecasting errors and variation in forecasting accuracy compared to using a single ANN, bagged ANN (BNN) and other techniques, as well as a reduction in computation time compared to BNN.

In this paper, an uppercase and bold letter represents a matrix, a lowercase and bold letter represents a vector, and a lower case and plain letter represents a scalar. The symbol ' represents transpose of a vector or a matrix. This paper is organized as follows: Section 2 presents an overview of load forecasting using ANN. Section 3 presents the load forecasting problem and boosting applied to neural networks. Section 4 presents results using new England data set [27], and conclusion is given in Section 5.

2. Artificial neural networks

Load patterns are dependant on several factors, such as weather, time, economy, electricity prices, random disturbances and geographical conditions. Many load forecasting techniques have been proposed and applied for predicting the load patterns accurately. Load patterns are often affected by several factors, and relationships between load patterns and affecting factors are nonlinear. Artificial intelligence-based algorithms have a great advantage in coping with nonlinear relationships, such as fuzzy logic, support vector machines and ANN. A significant amount of research has been carried out on the application of artificial intelligence-based algorithms to the load forecasting problem as they can perform better than conventional methods for short-term load forecasting, e.g., SVM [11] and ANN [15,17,18], including similar day-based wavelet neural networks [14] and hybrid ANN models [16], etc. Among these algorithms, ANN based forecasting has been used considerably because of its ease of implementation and good performance.

An ANN is a learning system composed of artificial neurons that try to simulate the function of human brain. In general, the ANN architecture consists of three layers: input layer, hidden layer and output layer. The input to the first layer consists of different factors or features that are deemed to have a significant impact on the output that we want to predict. The hidden layer/layers uses/use these features to calculate intermediate values and the output of the last layer consists of the predicted values.

Each hidden layer of an ANN consists of N neurons. The input to each layer l is multiplied by weights $\boldsymbol{W}^l$ and then added to the biases $\boldsymbol{b}^l$. The weight matrix scales the output from $(l-1)$th layer to the lth layer. As an example, if an input at time t consists of F different feature samples denoted as:

$$\boldsymbol{x}_t = [x_t^1 x_t^2 \cdots x_t^F]', \tag{1}$$

the output of the first layer $\boldsymbol{y}_t^1$ is:

$$\boldsymbol{y}_t^1 = \rho^1(\boldsymbol{W}^1\boldsymbol{x}_t + \boldsymbol{b}^1), \tag{2}$$

where $\boldsymbol{W}^1$ is a weight matrix, $\boldsymbol{b}^1$ is a bias vector and ρ^1 is an activation function. Note that the output of each layer is the input of the next layer, therefore, the final output effectively consists of the weighed input and added biases at each layer. The final output from the last layer L is:

$$y_t^L = \rho^L(\boldsymbol{W}^L\boldsymbol{y}_t^{L-1} + \boldsymbol{b}^L), \tag{3}$$

where y_t^{L-1} is the output from the second-last $(L-1)$th layer, and ρ^L is the activation function for the last layer.

At a layer l, the dimensions of $\boldsymbol{y}_t^l$ are $N \times 1$, while at the last layer L, y_t^L is a scalar. The dimensions of $\boldsymbol{W}^l$ and $\boldsymbol{b}^l$ are $N \times F$ and $N \times 1$, respectively for a layer l immediately following the input layer. For the following layers, the dimensions of $\boldsymbol{W}^l$ and $\boldsymbol{b}^l$ are $N \times N$ and $N \times 1$, respectively, while the dimensions of $\boldsymbol{W}^L$ are $1 \times N$ and $\boldsymbol{b}^L$ is a scalar.

The ANN is trained by changing weights and biases, and comparing the predicted output obtained with the input training data due to the changed weights and biases, and a target output. The training is carried out until difference between the predicted output and target output is minimized. This is expressed mathematically as:

$$\min_{\boldsymbol{W},\boldsymbol{b}} \sum_t (y - y_t^L)^2 \tag{4}$$

where $\boldsymbol{W} = [\boldsymbol{W}^{\boldsymbol{1}}, \boldsymbol{W}^{\boldsymbol{2}}, \ldots, \boldsymbol{W}^{\boldsymbol{L}}]$ and $\boldsymbol{b} = [\boldsymbol{b}^{\boldsymbol{1}}, \boldsymbol{b}^{\boldsymbol{2}}, \ldots, \boldsymbol{b}^{\boldsymbol{L}}]$.

The weights and biases are updated using stochastic gradient descent algorithm: The objective function given above is obtained using a part of the training samples, and the gradient of the objective function is calculated with respect to the weights and biases. Then the weights and biases are updated by subtracting from them the calculated gradients weighed by a factor μ. Each time all the training samples have been used completely, an epoch is said to be finished and a new epoch is started.

3. Improved short-term electric load forecasting using boosting

In short-term electric load forecasting, different independent parameters that can influence electric load are considered. Give past values for these parameters $\boldsymbol{X}$ over discrete time instants $\boldsymbol{t} = 1, 2, 3, \ldots, T$ and dependent observed load values $\boldsymbol{y}$ for these time instants, a model $\hat{f}$ that can describe the relation between $\boldsymbol{X}$ and $\boldsymbol{y}$ is first calculated, i.e.,

$$\hat{f} : \boldsymbol{X} \rightarrow \boldsymbol{y} \tag{5}$$

This function $\hat{f}$ should be such that [28]:

1. The estimated values have a lower bias, i.e., the difference between observed load $\boldsymbol{y}$ and load $\hat{\boldsymbol{y}}$ calculated using the model should be small.
2. The predicted values should also have a lower variance, i.e., there should not be large difference between predictions causes by