Daily Load Forecasting with a Fuzzy-input-Neural Network in an Intelligent Home

S.H. Ling, F.H.F Leung and P.K.S. Tam
Centre of Multimedia Signal Processing
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University
Hong Kong

Abstract

Daily load forecasting is essential to improve the reliability of the AC power line data network and provide optimal load scheduling in an intelligent home system. In this paper, a fuzzy-input-neural network forecaster model is proposed. This model combines a fuzzy system and a neural network. It can forecast the daily load accurately with respect to different day types under various variables. In this model, the fuzzy system performs a preprocessing for the neural network, so that the computational demand of the neural network can be reduced. Simulation results on a daily load forecasting will be given. Comparing the proposed algorithm with that of a conventional neural network, it can be shown that the proposed algorithm produces more accurate forecasting results.

I. INTRODUCTION

Modern homes should have smart features to ensure a higher degree of home security, entertainment and comfort. To realize these features, reliable channels for the communication among electrical appliances and users should be present. Appliances should be used in an efficient way to reduce the wastage of energy. This paper is based on an intelligent home system [1]. In this system, the AC power line network is used not only for supplying electrical power, but also serving as the data communication channel for electrical appliances. With this AC power line data network, a short-term load forecasting can be realized. An accurate load forecasting can bring the following benefits to the intelligent home: 1) Increasing the reliability [2-3] of the AC power line data network, and 2) Optimal load scheduling.

Computational intelligence techniques have been proposed for load forecasting. Owing to their capabilities of performing a nonlinear mapping from a set of inputs to a set of outputs, learning from experience and generalizing from previous examples, neural networks were employed for load forecasting [4-8,11]. However, their properties of slow convergence time and poor ability of processing much variable information have added restrictions to the application and development. Fuzzy logic offers a paradigm for representing and processing linguistic information. By processing fuzzy information, reasoning with respect to a linguistic knowledge base can be done. It has been used in load forecasting in order to deal with the variable information [10]. Obviously, fuzzy logic can complement a neural network and a properly integrated fuzzy neural network model can be a good tool for daily load forecasting. In this paper we proposed a fuzzy-input-neural network model. It combines a fuzzy system and a feed forward neural network. The fuzzy system is a preprocessing system that is used to reason the percentage of load change under the presence of input variable information. This fuzzy system is employed to reduce the input dimension of the neural network so that the computational demand of the neural network is reduced. By using this fuzzy-input-neural network, a short-term learning of load consumption pattern in an intelligent home can be realized so as to increase the reliability of the AC power line data network and ensure optimal load scheduling. The proposed fuzzy neural network model will be introduced in Section 2. Simulation results and a comparison with those from a conventional neural network will be given in Section 3. A conclusion will be drawn in Section 4.

II. FUZZY-INPUT-NEURAL NETWORK FORECASTER

Fig. 1 shows the fuzzy-input-neural network model developed for daily load forecasting in an intelligent home. The proposed fuzzy-input-neural network model consists of a fuzzy preprocessing system and a neural network. It forecasts the daily load based on historical daily load data and various variables including the maximum and minimum temperatures, the maximum and minimum humidity, the rainfall and the day type (Monday to Sunday). The fuzzy system preprocesses the various variables and reasons the percentage of load change. With the input dimension reduced by the fuzzy system, the neural network is used for learning the load consumption pattern based on historical patterns and other variable information interpreted by the fuzzy system. Once the network is trained, it is able to forecast the daily load.

The system output \( L_r(w,d,t) \) is the forecasted load as a function of hour of day of week \( w \), where \( w = 1, 2, ..., 52 \) is the week index (52 weeks in a year), \( d = 1, 2, ..., 7 \) is the day type (e.g. \( d=1 \) is Monday, \( d=7 \) is Sunday) and \( t = 1, 2, ..., 24 \) is the hour index. The variable information is preprocessed by a single fuzzy system corresponding to every hour of a week. Therefore the fuzzy preprocessing system is formed by a total of 168 (24x7) fuzzy systems. Referring to Fig. 1, fuzzy system 1, for example, is to forecast the percentage of load change at hour 1 (\( t=1 \)) on Monday (\( d=1 \)).
A. Fuzzy Preprocessing System

The structure of every fuzzy system is identical. Each fuzzy system has m rules. The k-th rule has the following format:

Rule k: IF x1 is A 1 k and x2 is A 2 k and ... and xn is A n k

THEN y is f k

x1 (i = 1, 2, ..., n) is the input variables of the fuzzy system, y is the output variable of the fuzzy system, f k is a singleton output membership function and A 1 k is a linguistic term characterized by a Gaussian membership function µ A 1 k (x i ) given by,

\[ \mu_{A_i}(x_i) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(x_i - \mu_i)^2}{2\sigma_i^2}} \]

where \( \mu_i \) and \( \sigma_i \) are the mean value and the standard deviation of the member function respectively.

The grade of membership of rule k is written as \( \mu_k = \sum \mu_{A_i}(x_i) \) \( \wedge \sum \mu_{A_i}(x_i) \) \( \wedge \sum \mu_{A_i}(x_i) \) where \( \wedge \) is the minimum operator.

The output of the fuzzy system is given by,

\[ y = \sum_{k=1}^{m} \mu_k f_k \]

The fuzzy preprocessing system has 5 inputs, i.e. n=5 and \( x = [x_1, x_2, ..., x_5] \). These inputs are the variable factors of maximum temperature (x1), minimum temperature (x2), maximum humidity (x3), minimum humidity (x4) and rainfall index (x5). Data of actual load pattern are obtained to design the fuzzy neural system. It is founded that the load pattern changes periodically every week. However, the general trend of the load consumption is almost the same everyday. We also find that the load of the previous day has a correlation with the desired load. Therefore we consider the fuzzy preprocessor's output at 8 different sampled times (to construct the inputs to the neural network) with emphasis putting on the weekly changes. The output vector of the fuzzy preprocessing system (Y) is given by:

\[ Y = [y(w, d-1, t), y(w, d-1, t-1), y(w-1, d, t), y(w-1, d, t-1), y(w-2, d, t), y(w-2, d, t-1), y(w-3, d, t), y(w-3, d, t-1)] \]

It should be noted that all elements in Y are corresponding to historical data. Refer to Fig. 1, the output vector Y of the fuzzy preprocessing system will be scaled by some nonlinear function \( f(t) = L_0(t) + L_1(t)Y(t) \), where \( L_0(t) \) is the historical actual load value. The scaled output in vector form is given by:

\[ \hat{Y}_n = f(Y, L_0) = L_0 + L_1Y \]

\[ L_0 = [L_0(w, d-1, t), L_0(w, d-1, t-1), L_0(w-1, d, t), L_0(w-1, d, t-1), L_0(w-2, d, t), L_0(w-2, d, t-1), L_0(w-3, d, t), L_0(w-3, d, t-1)] \]

\[ L_1 = [L_1(w, d-1, t), L_1(w, d-1, t-1), L_1(w-1, d, t), L_1(w-1, d, t-1), L_1(w-2, d, t), L_1(w-2, d, t-1), L_1(w-3, d, t), L_1(w-3, d, t-1)] \]

and

\[ \hat{L}_n = [\hat{L}_0(w, d-1, t), \hat{L}_0(w, d-1, t-1), \hat{L}_0(w-1, d, t), \hat{L}_0(w-1, d, t-1), \hat{L}_0(w-2, d, t), \hat{L}_0(w-2, d, t-1), \hat{L}_0(w-3, d, t), \hat{L}_0(w-3, d, t-1)] \]

The reasoned historical load value. \( \hat{L}_n \) will then be fed to the neural network to perform the learning process.

B. Neural Network

The neural network model is shown in Fig. 2. It is a three-layer feed-forward network with 8 nodes in the input layer and 1 node in the output layer. Through learning, the neural network forecasts the load (L (w, d, t)). The number of nodes in the hidden layer was determined experimentally and the best results are obtained when the hidden layer has 15 nodes. A log-sigmoid transfer function is used in the hidden layer and a linear line transfer function is used in the output layer. The neural network is tuned by the back propagation algorithm. Referring to Fig. 2, the inputs to the neural network is \( z = [x_1, x_2, ..., x_5] = \hat{L}_n \). The node outputs of the hidden layer are defined as,

\[ h_p = \frac{1}{1 + e^{-\sum_{p} w_{pi} x_i}} \]

where \( w_{pi} \) is the weight of the link between node n of the input layer and node p of the hidden layer, and \( l = 15 \) (15 hidden nodes are used). The output \( L(w, d, t) \) in the output layer is defined as,

\[ g = \sum_{l=1}^{15} w_{l1} h_l \]

where \( w_{l1} \) is the weight of the link between node p of the hidden layer and the output nodes, and \( h_p \) is the output of the hidden node p.

\[ L(w, d, t) = h(g, y(w, d, t)) = (1 - y(w, d, t))g \]

Back propagation gradient descent algorithm [9] is employed to train the network by minimizing the squared error between the actual load and the forecasted load.

III. SIMULATION RESULTS

The proposed fuzzy-input-neural network for daily load forecasting will be tested in this section. Load patterns of 10 weeks real data are used to train the neural network. After learning, the daily load forecasting for weekday (Monday) and Sunday load were tested against another set of real data. The daily load forecast by the fuzzy-input-neural network was compared to the actual load and the error calculated. The error of the forecast results are calculated based on the mean absolute percentage error (MAPE) given by

\[ MAPE(\%) = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{act_i - forecast_i}{act_i} \right| \times 100 \]

where N is the total number of hours (in this case 24 for a day), act and forecast are the actual and forecasted load at hour i. For comparison purposes, a conventional feed forward neural network forecaster [11] has been employed to forecast the daily load. Fig. 3 and 4 show the forecast
load obtained by this neural network forecaster and the proposed fuzzy-input-neural network forecaster for a weekday (Monday) and Sunday. Table 1 lists the MAPEs of these two forecasters. We can observe that the MAPEs given by the fuzzy-input-neural network are smaller than those given by the conventional neural network approach. Considering the number of inputs to the neural network, the proposed method has only 8 inputs while the conventional neural network has 40 (8 historical data each with 5 variable factors). The training time based on the proposed system is 163.96 time-unit, while that of the conventional neural network is 627.11 time-unit. We can observe that the proposed method has a much shorter training time.

IV. CONCLUSION

This paper describes the development and implementation of a daily load forecaster in an intelligent home based on a fuzzy-input-neural network approach. A neural network has been used to build the nonlinear mapping between the uncorrelated inputs and output. A fuzzy preprocessing system is used to process various inputs such as the day type and the weather information so that the computational demand on the neural network is significantly reduced. The results indicate that an accurate forecasting of the daily load curves could be achieved by using the proposed fuzzy-input-neural network model.

ACKNOWLEDGEMENTS

The work described in this paper is substantially supported by a Research Grant from The Hong Kong Polytechnic University (Project No. G-V954).

REFERENCES


Fig. 1. Fuzzy-input-neural network forecaster model.

Fig. 2. Structure of Neural Network in the fuzzy-input-neural network forecaster.
Fig. 3. Forecasting results for a Monday.

Fig. 4. Forecasting results for a Sunday.

**TABLE I. COMPARISON OF MAPEs FROM DIFFERENT METHODS**

<table>
<thead>
<tr>
<th>Forecasting Method</th>
<th>Weekday (Monday)</th>
<th>Sunday</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purposed Fuzzy-input-neural network</td>
<td>0.8574%</td>
<td>1.5613%</td>
</tr>
<tr>
<td>Neural network</td>
<td>1.3112%</td>
<td>3.1898%</td>
</tr>
</tbody>
</table>