EASR

Engineering and Applied Science Research

https://www.tci-thaijo.org/index.php/easr/index

Published by the Faculty of Engineering, Khon Kaen University, Thailand

Electricity load forecasting using a deep neural network

Pyae Pyae Phyo and Chawalit Jeenanunta*

School of Management Technology, Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani 12120, Thailand

Received 18 March 2018
Revised 11 June 2018
Accepted 26 June 2018

Abstract

Forecasting the daily load demand of an electric utility provider is a complex problem as it is nonlinear and influenced by external factors. Deep learning, machine learning and artificial intelligence techniques have been successfully employed in electric consumption load, financial market, and reliability predictions. In this paper, we propose the use of a deep neural network (DNN) for short-term load forecasting (STLF) to overcome nonlinearity problems and to achieve higher forecasting accuracy. Historical data was collected every 30 minutes for 24 hour periods from the Electricity Generating Authority of Thailand (EGAT). The proposed techniques were tested with cleaned data from 2012 to 2013, where holidays, bridging holidays, and outliers were replaced. The forecasting accuracy is indicated by the mean absolute percentage error (MAPE). In this paper, there are two different training datasets, everyday training dataset which is arranged by day sequentially and same day training dataset which is separated seven groups of day (for e.g., only Monday training is used to forecast Monday). The outcomes of a deep neural network (DNN) are compared with an artificial neural network (ANN) and support vector machines (SVM) with an everyday training dataset. The empirical results reveal that the proposed DNN model outperforms the ANN and SVM models. Moreover, the DNN model trained with same day training datasets provides better performance than a DNN trained with an everyday training dataset for weekends, bridging holidays, and Mondays. Additionally, the DNN using a same day training datasets provides higher accuracies for December and January.

Keywords: Deep neural network, Artificial neural network, Support vector machines, Forecasting, Short-term load forecasting

1. Introduction

Load forecasting plays an important role in supplying electric utilities so that they can make correct decisions on electrical power generation, transmission and distribution. Electricity load forecasting can be classified into long-term, medium-term and short-term needs. These forecast durations can be up to one year, one month and one day, respectively. Long-term and medium-term forecasting are usually used in the generating stations and for transmission lines in the power system. However, short-term load forecasting plays an imperative role in the security of the security of the power conducting system and in predicting operating costs. Therefore, short-term load forecasts are more focused due to the growth of competitive in energy markets.

Forecasting techniques that are applied for load forecasting can be classified into two sub-groups, traditional statistical models and artificially intelligent models. Traditional statistical models include regression analysis [1], moving averages [2], exponential smoothing [3], and stochastic time series models [4]. Artificial intelligence techniques include support vector machines [5], artificial neural networks [6], and fuzzy time series [7]. Neural networks are the most popular artificial models for nonlinear time series problems [8]. However, if there are multiple hidden layers, neural networks do not work very well because of backpropagation [9]. These take a very long time and sometimes converge an incorrect local minimum and have slow convergence. To overcome these complex problems, deep learning was introduced.

In this research, a deep neural network (DNN) is proposed to achieve higher performance and accuracy for daily load forecasting. This paper is organized as follows. First, the paper reviews related work and the literature. Then, three models, methodology and data arrangement are presented. Finally, outcomes are discussed to compare forecasting results of the three different methods.

2. Literature review

Warren McCulloch and Walter Pitts previously considered an artificial neural network (ANN) in 1943 [10]. Many researchers have shown that an ANN is an excellent tool for use in several areas including medicine, business, communications, and industrial process control. ANNs are very popular computational models. They are influenced by the structure and functional aspects of biological neural networks. A neural network is composed of an

*Corresponding author. Tel.: +6689 113 3494
Email address: chawalit@siit.tu.ac.th
doi: 10.14456/easr.2019.2

interconnected group of artificial neurons. It computes the targets by processing information using a connection. During the learning process, an ANN is an adaptive system that can change its structure depending on external information flowing through the network. Modern neural networks are usually used to model complex relationships between inputs and outputs to find patterns in data.

In the late 1980’s and early 1990’s, ANNs were applied to forecast load demand in electric power systems [6, 11-12]. Jeenanunta and Abeyrathna adjusted the parameters of an ANN to enhance the forecasting accuracy using a transfer function [13]. A feed-forward neural network with the Levenberg-Marquardt algorithm performed well yielding better forecasting accuracy for electricity load demand [14]. ANNs rely on several parameters such as the number of invisible layers, backpropagation algorithms and selected input variables to improve the accuracy for STLF [15]. They have become very popular among machine leaning techniques, however, there are drawbacks to train the model and obtain better forecasting results due to the weaknesses of the backpropagation algorithm [16].

Deep neural networks (DNN) are commonly based on ANNs with multiple hidden layers and an approximation error that can be reduced by adding many hidden layers between the input and output layers. Deep architectures are used to achieve better representation and capture higher level abstractions. Quan et al. applied an ensemble deep belief network with one artificial dataset and three regression datasets to execute time series and regression predictions [17].

DNN has been successfully applied in the last few years. Several researchers have produced outstanding results in the fields of regression, image classification, automatic speech and face recognition, natural language processing, and bioinformatics. There are various DNN architectures such as long short-term memory [18], recursive neural networks [19], convolutional neural networks [19], and deep belief networks [20]. Each of them uses computational methods that are comprised of numerous hidden layers to learn representations of data at various abstraction levels. They also can detect complex structures in large data sets using backpropagation to overcome the drawbacks of machine learning.

The feed-forward multilayer perceptron, which is one of deep neural network models, was introduced for supervised learning algorithms [21]. A deep belief network was also applied to forecast load demand for hourly electricity consumption data in Macedonia [22]. Moreover, El-Sharkh and Rahman presented a multilayer perceptron, radial basis and recurrent neural network (RNN) with a parallel structure ANN that gave better results compared to standard time series methods [23]. Rashid et al. proposed a RNN with an internal feedback structure for electricity load prediction that showed reliable and robust results [24]. Moreover, a nonlinear auto-regressive RNN provided smoothly forecasted results, in contrast with previous studies for hourly predictions of high resolution wave power [25].

In addition to ANN and DNN, support vector machines (SVM) were introduced for STLF. This approach outperformed an autoregressive model [5]. Moreover, Mohandes found that improvement of performance for a SVM depends on increasing the size of the training dataset. Chen et al. showed that an SVM with seasonal factors could enhance the forecasting performance and that the temperature factor could not influence mid-term load forecasting [26]. Moreover, Support Vector Regression (SVR) outperformed other nonlinear models to solve nonlinear, non-stationary and a-priori undefined problems [8]. A recurrent support vector machine combined with genetic algorithms was developed, which determined the parameters of a SVM. It was applied to forecast a regional electricity load [27]. The forecasting performance using a hybrid model obtained higher forecasting accuracies than regression, SVM and ANN models. Researchers applied several hybrid models with SVR to improve dynamic high-performance accuracy for STLF [28-29].

3. Methodology

In this research, we propose three forecasting techniques to predict the electricity load demand and compare the forecasted results. The first model is an artificial neural network which can be applied to learn complex nonlinear problems. In the training process of an ANN model, only one hidden layer and a sigmoid activation function is used. A simple backpropagation algorithm is employed for training the model. Second, we use a deep neural network (DNN) that is able to learn end-to-end without manually adding features. This DNN model used 50 hidden layers and 50 hidden nodes in its network. Moreover, the activation function is selected as a rectified linear unit (RELU) function instead of a sigmoid function to solve vanishing problems. It is the most popular non-linear function. It can learn much faster in networks with many layers by allowing training of deeply supervised networks without unsupervised pre-training. In the backpropagation algorithm, the model is optimized using a stochastic gradient descent (SGD) before updating parameters at every step. It tends to converge to a global minimum much faster than an ordinary gradient descent. Finally, the last model is a support vector machine (SVM) using a kernel function and quadratic programming to find a large solution space with better performance. RapidMiner tool is used as software platform for integrating all proposed models .The detailed procedures of each forecasting model is discussed below.

3.1 Artificial neural network

An artificial neural network model is used as simple feed-forward neural network trained using a backpropagation algorithm. Figure 1 represents an artificial neural network structure. In the feed-forward process, the information is only moving in the forward direction from the input nodes *x* passing through the hidden nodes to the output nodes *y* with no cycling or looping in the network. The model uses a simple sigmoid activation function (*f*) to produce output values (*y*). Afterwards, the network updates its weights (*w*) using a backpropagation algorithm [15].

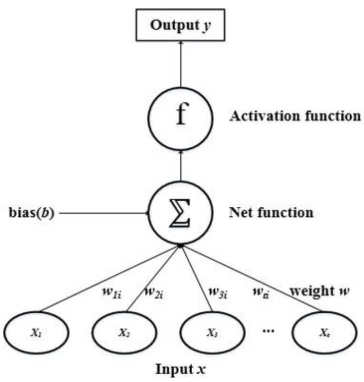

Figure 1 Artificial neural network structure

In the backpropagation algorithm, the predicted output values are compared with the target values to calculate the value of some predefined error function. By feeding the error back through the network, the algorithm adjusts the weights of each connection to minimize error by some amount depending on the specified learning rate. This training process is repeated until the performance of the network is sufficient and the network converges to a small error.

3.2 Deep neural network

Deep learning is an approach to train a DNN model consisting of a large number of processing layers. Deep neural networks are a class of neural network models that have an input layer, an output layer, and a large number of hidden layers. In Figure 2, the input layer *x* and the output layer *y* are referred by the bottom and the top layers respectively. The layers between *x* and *y* represent hidden layers (*h*) of the network which perform as a black box.

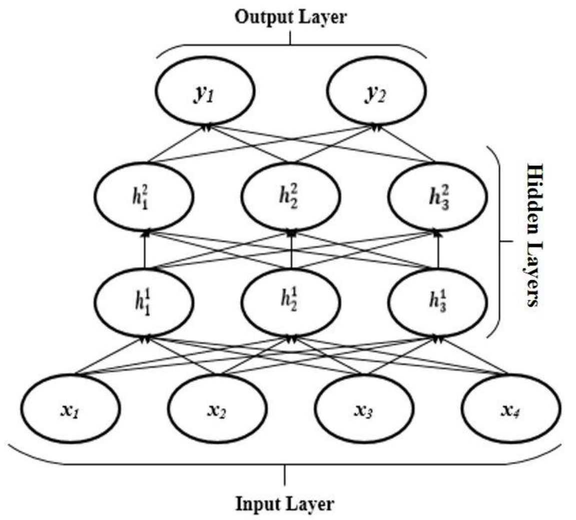

Figure 2 Deep neural network structure

The DNN model trains a feed-forward network to generate the corresponding output values through all of the hidden layers and neuron nodes in a forward propagation. In this research, the model uses rectified linear unit (RELU) as an activation function while similar research problems have been solved using only a sigmoid activation function. The RELU uses a tanh function that is not bounded or continuously differentiable. This function is piece-wise linear and saturates at 0 whenever the input x is less than 0. Equation 1 indicates that a nonlinear activation function *f* takes a weighted sum of input *x* values and returns a value for h_j^l [21].

$$h_j^l = f\left(\sum_{i=1}^{k} w_{ij}^l x_i\right) \tag{1}$$

where,

h_j^l = the j^{th} hidden node in layer *l*,
x_i = the i^{th} input,
w_{ij}^l = the weight between node *i* in layer (*l*-1) and node *j* in layer *l*,
f = activation function.

The main objective is to minimize the error of a calculated output value. Therefore, the network compares predicted output values with actual values. Next, the proposed model minimizes error using a stochastic gradient descent (SGD) algorithm before updating the weights [9].

The objective of using SGD is to overcome speed convergence obstacles and avoiding local minima. At first, the training data are shuffled at each iteration of the training network during the SGD process. Next, all of the weights are updated using only one sample or a few training samples. To reach a global minimum, a SGD updates weights frequently in the direction of the gradient of the loss function, which is the error between target and output at every iteration. Unlike an ordinary gradient descent, a SGD selects a single dataset instead of all datasets to compute the gradient at each iteration. Equation 2 describes updating the weights in the SGD process [9].

$$w^{new} = w^{old} - \eta \nabla_w J(w^{old}; x^{(k)}, y^{(k)}) \tag{2}$$

where,

w^{new} = updated weight value,
w^{old} = old weight value,
J = gradient value,
η = learning rate,
($x^{(k)}$, $y^{(k)}$) = a pair of a training sample at the k^{th} iteration.

In general, most gradient optimization methods converge effectively when using the full training set. A SGD can converge much faster than ordinary gradient descent methods because it is less memory intensive since it uses one dataset at a time [9]. Furthermore, SGD has the ability to obtain a meaningful update without iterating over the entire dataset to overcome redundancy in the datasets. Moreover, if the loss function is convex, using SGD is guaranteed to find a global minimum. A SGD can obtain better performance for big DNN models and large data sets.

3.3 Support vector machine

Support vector machines (SVM) are computational or mathematical models to solve complex problems [5]. A SVM can be used for regression, classification and other tasks. It also can learn very fast and provide good results for many tasks. In the training process, a SVM uses linear, quadratic and asymmetric loss functions. A support vector machine usually sets up a hyperplane or set of hyperplanes in a high or infinite-dimensional space. The hyperplane might have the largest distance to the nearest training data points of a functional margin to achieve good separation. A sets of hyperplanes separate nonlinear groups in a finite dimensional space.

The original finite-dimensional space approaches map into a higher-dimensional space by making discrimination easier in that space. Therefore, in this study, we use two essential factors for the implementation of SVM. The first factor is the kernel model to get a large amount of solution space. The other is the use of a quadratic function to adjust SVM parameters in the entire training process. In Figure 3, there is an input variable *x* and an output variable *y*. The objective of a SVM is to map the input data into a higher-dimensional space using a non-linear mapping and conduct a linear regression.

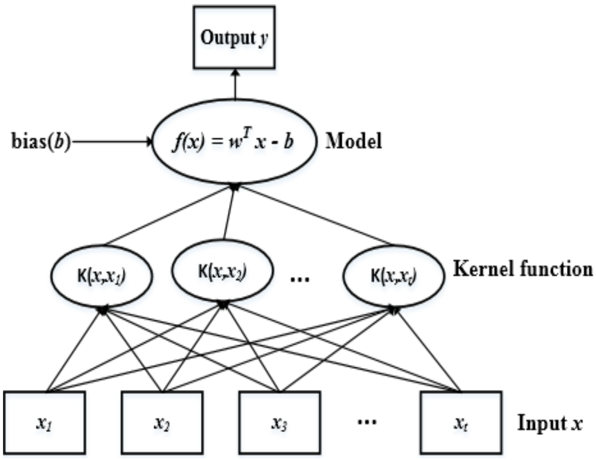

Figure 3 Support vector machine structure

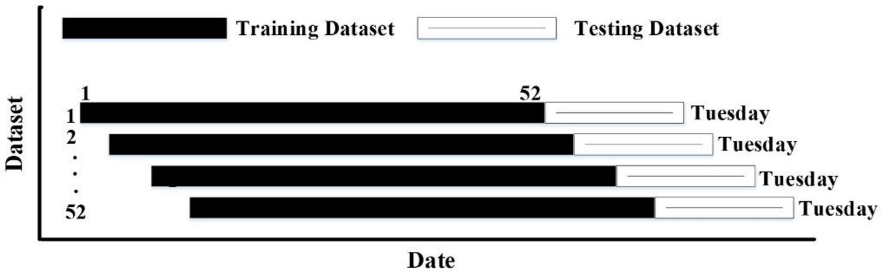

Figure 4 Sample dataset for a walk-forward testing routine

4. Training data and testing data

A collection load data every 30 minutes from 1 March 2009 to 31 December 2013 was supplied by the Electricity Generating Authority of Thailand (EGAT). This data was collected from five different regions, i.e., Metropolitan Bangkok, the Central, South, North and Northeast regions of Thailand. There was low electricity demand with only one peak load curve at night in three regions, the South, North, and Northeast. Conversely, there was high electricity demand with three peak load curves in two regions, i.e., Metropolitan Bangkok and the Central region. In the current study, only Metropolitan Bangkok was considered because of its large load demand and high variations.

In previous studies, two approaches were used for developing machine learning models. The first is to divide the whole dataset into two parts, training and testing datasets. The other approach is to divide the dataset into three parts, training, validation and testing datasets [11, 30-32]. This research was conducted using the first approach since the proposed models are fitted without tuning any parameter in the training process. In other words, simple models do not over-fit the training data.

In this research, the data is separated into two datasets, i.e., training and testing datasets. Both datasets are arranged as paired inputs and targets to train and test the models. We select a one-year dataset from 7 May 2012 to 30 May 2013 to train the models. Thus, there are 388 days in our training dataset in the ANN, SVM, DNN_1 models. As a result, all models have to be trained with 388 datasets to test a one-day forecast. In the DNN_2 model, there are 52 training datasets for each test dataset. Once the model is applied to the test dataset, the forecasting performance is determined. Then the data is slid to consider the next 52 training dataset and perform the same procedure. This procedure is called a "walk-forward testing routine" as depicted in Figure 4.

The original historical data must be cleaned because there are many holidays, missing values, and outliers that affect the results. If these outliers are included in the training data, the accuracy of load predictions would be lower. We categorize load patterns into five categories, Mondays, weekdays, weekends, holidays and bridging holidays as shown in Figure 5. If Thursday and Saturday are holidays, we consider Friday as a bridging holiday. For example, since there are eight days for weekends in a month, we take the average load in each period from all weekends in January 2013 to calculate average load for each category. It can be clearly seen that holiday and bridging holiday load patterns are very different from other load patterns. Consequently, we apply a weighted moving average to replace the holidays and bridging holidays.

Furthermore, we detect outliers using a time-window based filtering band as there is a similar pattern in the same time period and the same day of week. We arrange the dataset consisting of the same weekday and same time period to construct the filtering band of each weekday and each time period. We construct a time-window based filtering band using a four week moving average and standard deviation with same time period and day of the week. After that, all of the data outside the filtering band are regarded as outliers and replaced by a two weeks moving average.

Selection of input variables is one of the main steps for forecasting. There are many external factors affecting the load such as temperature, time (hour of day, day of week, and month of year), and weather conditions among others. Among these, temperature is the most influential factor. It is associated with meteorological situations. In this study,

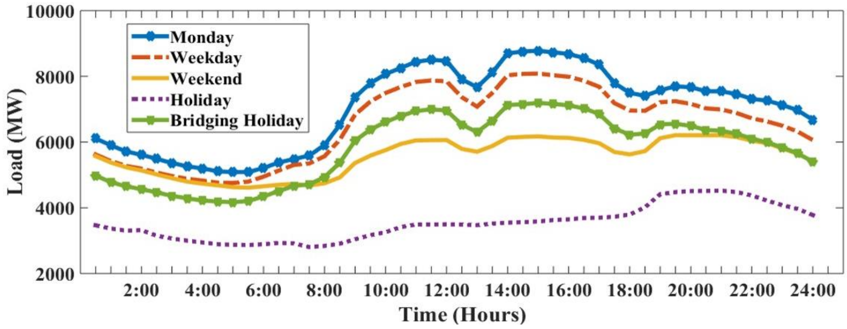

Figure 5 Load patterns for five categories

Table 1 Everyday training data arrangement for testing target 1 June 2013

				Input				Target
	No.	***L_t(d-7)***	***L_t(d-1)***	***T_t(d-1)***	***T_t(d)***	***DoW***	***MoY***	***F_t(d)***
Training	1	01/05/12 (Tue)	07/05/12 (Mon)	07/05/12 (Mon)	08/05/12 (Tue)	1	5	08/09/12 (Tue)
	.	.	.	.	.	.	.	.
	.	.	.	.	.	.	.	.
	.	.	.	.	.	.	.	.
	388	23/05/13 (Fri)	29/5/13 (Thurs)	29/5/13 (Thurs)	30/05/13 (Fri)	4	5	30/05/13 (Fri)
				Input				Output
Testing	**No.**	***L_t(d-7)***	***L_t(d-1)***	***T_t(d-1)***	***T_t(d)***		***MoY***	***F_t(d)***
	1	25/05/13 (Sat)	31/05/13 (Fri)	31/05/13 (Fri)	01/06/13 (Sat)		1	01/06/13 (Sat)

data selected as input that is directly proportional to the load demand. In DNN, ANN, and SVM models, there are six input variables, the previous day's load, previous week's same day load, previous day's temperature, forecasted day's temperature, day of week (*DoW*) and month of year (*MoY*), as in Equation (3). The basic forecasting equation is given by:

$$F_t(d) = b_1 L_t(d-1) + b_2 L_t(d-7) + b_3 T_t(d-1) + b_4 T_t(d) + DoW + MoY \quad (3)$$

where,

- $F_t(d)$ = Forecast load at period *t* for day *d*,
- $L_t(d\text{-}1)$ = Load at period *t* for day *d*-1 (yesterday's load)
- $L_t(d\text{-}7)$ = Load at period *t* for *d*-7 (previous week same day),
- $T_t(d\text{-}1)$ = Temperature at period *t* for *d*-1 (yesterday's temperature),
- $T_t(d)$ = Forecast day's temperature at period *t* for day *d*,
- *DoW* = Day of Week (1, 2, …, 7),
- *MoY* = Month of Year (1, 2, …, 12),
- *t* = 1, 2, 3, …, 48 periods,
- $b_1, \ldots, b_5$ = coefficients of load and temperature.

According to the above equation, all models take six inputs to forecast the next day's load**.** Table 1 shows that there are 388 pairs data in the training dataset from May 1 2012 to May 30 2013 to forecast the load of June 1, 2013. This data arrangement is noted as an everyday training dataset. After completing the training, each model will be tested using one testing dataset. In this case, the goal is to forecast June 1, 2013. Once we complete the testing dataset, the models are trained with 388 new rolling pairs of data and they are tested for the next forecasting day, June 2, 2013.

Additionally, we propose another data arrangement with five input variables. This data arrangement is called the same day training dataset and it is used with DNN. For this DNN model, we selected five input variables, i.e., yesterday's load, previous week same day load, yesterday's temperature, forecasted day's temperature, and month of the year (*MoY*) as in Equation (4). The basic forecasting equation is written as:

$$F_t(d) = b_1 L_t(d-1) + b_2 L_t(d-7) + b_3 T_t(d-1) + b_4 T_t(d) + MoY \quad (4)$$

where,

- $F_t(d)$ = Forecast load at period *t* for day *d*,
- $L_t(d\text{-}1)$ = Load at period *t* for day *d*-1 (yesterday's load),
- $L_t(d\text{-}7)$ = Load at period *t* for *d*-7 (previous week same day)
- $T_t(d\text{-}1)$ = Temperature at period *t* for *d*-1 (yesterday's temperature),
- $T_t(d)$ = Forecast day's temperature at period *t* for day d,
- *MoY* = Month of Year (1, 2, …, 12),
- *t* = 1, 2, 3, …, 48 periods,
- $b_1, \ldots, b_5$ = coefficients of load and temperature.

This second training dataset is arranged to use the same day for the target as shown in Table 2. In this table, the target day is always on Saturday. The training dataset consists of 52 datasets. It also includes Friday's load as yesterday's input when the model predicts Saturday's load, as shown in Table 2.

Table 2 Same day training data arrangement for the testing target on 12 Jan 2013

		Input					Target
	No.	$L_t(d\text{-}7)$	$L_t(d\text{-}1)$	$T_t(d\text{-}1)$	$T_t(d)$	*MoY*	$F_t(d)$
Training	1	07/01/12 (Sat)	13/01/12 (Fri)	13/01/12 (Fri)	14/01/12 (Sat)	1	14/01/12 (Sat)
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	52	22/12/12 (Sat)	28/12/12 (Fri)	28/12/12 (Fri)	29/12/12 (Sat)	12	29/12/12 (Sat)
		Input					Output
	No.	$L_t(d\text{-}7)$	$L_t(d\text{-}1)$	$T_t(d\text{-}1)$	$T_t(d)$	*MoY*	$F_t(d)$
Testing	1	05/01/13 (Sat)	11/01/13 (Fri)	11/01/13 (Fri)	12/01/13 (Sat)	1	12/01/13 (Sat)

Table 3 Monthly MAPE' in 2013 for ANN, SVM, DNN_1, and DNN_2.

Month	ANN	SVM	DNN_1	DNN_2	Number of holidays
Jan	6.2233	6.8384	4.9290	4.1814	1
Feb	4.8546	4.7640	3.5650	4.9687	1
Mar	4.0913	4.6069	3.0386	3.7373	-
Apr	5.2078	5.8053	4.3674	3.4356	7
May	4.6311	5.3445	4.0751	4.0455	5
Jun	3.6040	4.6400	3.2080	3.8653	-
Jul	4.8488	5.8225	4.4771	4.1947	3
Aug	3.2853	3.8207	2.4050	3.7734	1
Sep	3.1977	3.9151	2.5414	4.0620	-
Oct	3.9449	4.3180	3.1084	4.4549	1
Nov	3.5938	4.1131	2.7328	4.5204	-
Dec	11.9294	13.1251	12.3886	6.8291	4
Total Average	4.9510	5.5928	4.2364	4.3390	

The main objective of all models is to minimize forecasting errors. In most previous studies, the mean absolute percentage error (MAPE) is commonly used as an accuracy measurement. This is how many units the forecast result deviates from the original data. However, in this study, we modify the MAPE calling it MAPE' and using cleaned data instead of the original data since the original data is cleaned by a time-window based method to remove outliers. The mean absolute percentage error (MAPE') is an accuracy measurement comparing the cleaned load and the forecasted load.

The mean absolute percentage error (MAPE') is:

$$MAPE' = \frac{1}{t}\sum_{t=1}^{48}\frac{L_t'(d) - F_t(d)}{L_t'(d)} \times 100\,\%, \tag{5}$$

where,

$F_t(d)$ = Forecast load at period t for day d,
$L'_t(d)$ = Cleaned load at period t for day d,
t = 1, 2, 3, …, 48 periods.

5. Result and discussion

In this research, we use a DNN model with two different data structures to predict the daily load. The first one is referred to as DNN_1 and it is tested using an everyday training dataset. The second one is referred to as DNN_2 which applies the same day training dataset to train the model. All models, including ANN and SVM, use cleaned data to train and test. Additionally, ANN and SVM use an everyday training dataset. Table 3 shows the summarized monthly MAPE' outcomes using four different forecasting models for each month in 2013.

According to the Table 3, it is clear that the performance of DNN_1 is better than the ANN and SVM models as its results produced a smaller MAPE'. The MAPE' of December is significantly higher than the rest of the months due to higher fluctuation in load. The forecast result is especially different from the actual load results due to the Christmas season and an unexpectedly high tourist presence in Thailand. Figure 6 shows the monthly average loads for October, November, and December. The average load in December is lower than the previous two months because of a lower average temperature. The variation still continues in the month of January with more unexpected tourists and New Year celebrations.

Electricity consumption is high from April to July due to high temperatures. This is because peak load demand is proportional to temperature and thus it increases as the weather becomes hotter. However, during the month of June, there are no holidays which results in similar pattern in the load. Fluctuations in the load are low during the months from August to November resulting in better forecast results, thus low MAPE'.

Comparing the two data arrangements, DNN_2 provides better accuracy than DNN_1 for months which there are many holidays, April, May, and July. Consequently, using a same day training dataset is good for predicting loads in months which have many holidays. However, the everyday training dataset gives better performance for other months. Moreover, the MAPE' for DNN_1 has almost twice the error of DNN_2 in December.

We summarize the MAPE' into six categories based on the load patterns. These are weekdays, weekends, Mondays, holidays, bridging holidays, and total average to compare the results. According to Table 4, ANN, SVR and DNN_1 yield

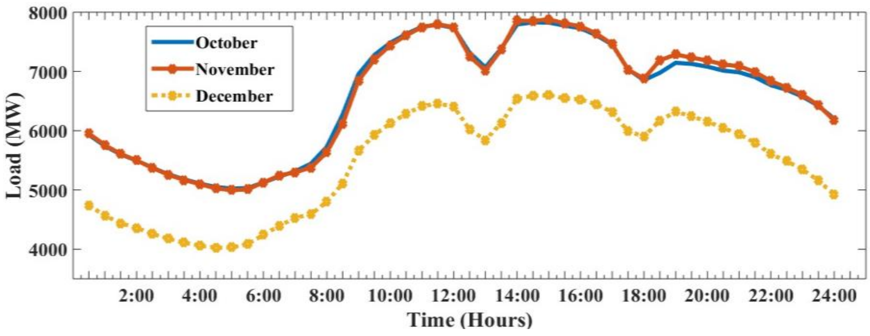

Figure 6 Comparison of average load curves for October, November, and December 2013.

Table 4 Comparing MAPE' in 2013 for day type categories of ANN, SVM, DNN_1 and DNN_2.

	ANN	SVM	DNN_1	DNN_2
Weekdays	4.1471	4.3409	3.3206	5.2315
Weekends	4.8538	5.7487	4.2273	3.1795
Monday	5.6348	7.4658	4.5151	3.7454
Holidays	10.2910	11.6927	10.6844	3.7034
Bridging holidays	6.2243	6.4526	5.9337	3.1571
Total average	4.9649	5.6185	4.2544	4.3378

Table 5 Relationship between computation time taken and techniques used.

	ANN	SVM	DNN_1	DNN_2
Computation time	1 min 22 sec	1 min 41 sec	2 min 46 sec	1 min 44 sec

the highest errors for holidays and bridging holidays because these load patterns are different from those of normal weekdays.

The results of MAPE' on Monday also have lower accuracy since we use Sunday's data as an input to forecast Monday. However, the load on Sunday is normally smooth and lower than on Monday. Additionally, the MAPE' values on the weekends are generally worse than weekdays for ANN, SVM and DNN_1. This occurs because we use Friday as an input to forecast Saturday. Using the same day training dataset improves the forecasting accuracy for Mondays, weekends, and bridging holidays.

The RapidMiner software platform is used in this research study to integrate data preparation, optimization techniques, machine learning and deep learning techniques. The computation time required for each technique is shown in Table 5.

6. Conclusions

In this study, we utilized three powerful forecasting techniques, DNN, ANN and SVM to solve nonlinear problems in STLF. Historical 30 minute load data from 2009 to 2013 was obtained from the Electricity Generating Authority of Thailand (EGAT). All techniques were trained and tested using cleaned load data that included outside temperature, day of week, and month of year to develop daily forecasts in 2013.

There are two training dataset structures, everyday training and same day training. All three models were trained with the everyday training dataset and used to predict each daily load demand for 2013. Moreover, the DNN model was also trained with the same day training dataset to compare performance.

The proposed DNN model with the everyday training dataset obtained better forecasting performance, compared to the ANN and SVM models for every month in 2013 except December. Furthermore, the DNN model also performed better for weekdays, weekends, Mondays, and bridging holidays. This empirical result shows that DNN is a promising model for electricity load forecasting in the electric power industry.

Additionally, we propose a DNN model using the same day training dataset to predict daily load. For this second training dataset, we used five inputs to train the model. After using same day training dataset, the DNN model yielded improved MAPE' results for bridging holidays, Mondays, and weekends. Moreover, it also gave higher accuracy for January and December which normally have the lowest accuracy compared to other months of the year. The forecast for December had the highest MAPE' result since it had the lowest temperature and hence the lowest load demand. In future research, various time series forecasting models and data features may be investigated to improve forecasting performance.

7. Acknowledgements

This research was partially supported by the Logistics and Supply Chain Systems Engineering Research Unit (LogEn), Sirindhorn International Institute of Technology (SIIT), Thammasat University (TU), Thailand. Data used in this research was provided by EGAT, and we acknowledge their support in completing this research.

8. References

[1] Papalexopoulos AD, Hesterberg TC. A regression-based approach to short-term system load forecasting. Power Industry Computer Application Conference; 1989 May 1-5; Seattle, USA. USA: IEEE; 1989. p. 414-23.

[2] Singh AK, Khatoon IS, Muazzam M. An overview of electricity demand forecasting techniques. Network and Complex Systems. 2013;3(3):38-48.

[3] Christiaanse WR. Short-term load forecasting using general exponential smoothing. IEEE Trans Power App Syst. 1971PAS-90(2):900-11.

[4] Liu K, Subbarayan S, Shoults RR, Manry MT, Kwan C, Lewis FI, et al. Comparison of very short-term load forecasting techniques. IEEE Trans Power Syst. 1996; 11(2):877-82.

[5] Mohandes M. Support vector machines for short-term electrical load forecasting. Int J Energ Res. 2002;26(4):335-45.

[6] Hippert HS, Pedreira CE, Souza RC. Neural networks for short-term load forecasting: a review and evaluation. IEEE Trans Power Syst. 2001;16(1):44-55.

[7] Azadeh A, Saberi M, Gitiforouz A. An integrated simulation-based fuzzy regression-time series algorithm for electricity consumption estimation with non-stationary data. J Chin Inst Eng. 2011;34(8):1047-66.

[8] Sapankevych NI, Sankar R. Time series prediction using support vector machines: A survey. IEEE Comput Intell Mag. 2009;4(2):24-38.

[9] Bottou L. Large-scale machine learning with stochastic gradient descent. In: Lechevallier Y, Saporta G, editors. 19th International Conference on Computational Statistics; 2010 Aug 22-27; Paris, France. Heidelberg: Physica-Verlag; 2010. p. 177-86.

[10] McCulloch WS, Pitts W. A logical calculus of the ideas immanent in nervous activity. Bull Math Biol. 1990;52(l-2):99-115. Corrected and republish from: Bull Math Biol. 1943;5:115-33.

[11] Park DC, El-Sharkawi MA, Marks RJ, Atlas LE, Damborg MJ. Electric load forecasting using an artificial neural network. IEEE Trans Power Syst. 1991;6(2):442-9.

[12] Bala A, Yadav NK, Hooda N. Implementation of artificial neural network for short term load forecasting. Curr Trends Tech Sci. 2014;3(4):247-51.

[13] Jeenanunta C, Abeyrathna KD. The study of artificial neural network parameters for electricity forecasting. International Conference on Applied Statistics; 2016 Jul 13-15; Phuket, Thailand. Thailand: Thammasat University; 2016. p. 1-7.

[14] Rodrigues F, Cardeira C, Calado JMF. The daily and hourly energy consumption and load forecasting using artificial neural network method: a case study using a set of 93 households in Portugal. Energy Procedia. 2014;62:220-9.

[15] Rui Y, El-Keib AA. A review of ann-based short-term load forecasting models. Proceedings of 27th Southeast Symponium; 1995 Mar 12-14; Starkville, USA. USA: IEEE; 1995. p. 78-82.

[16] Krunic S, Krcmar I, Rajakovic N. An improved neural network application for short-term load forecasting in power systems. Elec Mach Power Syst. 2000;28(8):703-21.

[17] Qiu X, Zhang L, Ren Y, Suganthan PN, Amaratunga G. Ensemble deep learning for regression and time series forecasting. 2014 IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL); 2014 Dec 9-12; Orlando, USA. USA: IEEE; 2014. p. 1-6.

[18] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735-80.

[19] Urban G, Geras KJ, Kahou SE, Aslan O, Wang S, Mohamed A, et al. Do deep convolutional nets really need to be deep and convolutional?. arXiv:1603.05691. 2016:1-13.

[20] Hinton GE, Osindero S, The YW. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18(7):1527-54.

[21] Liu W, Wang Z, Liu X, Zeng N, Liu Y, Alsaadi FE. A survey of deep neural network architectures and their applications. Neurocomputing. 2017;234:11-26.

[22] Dedinec A, Filiposka S, Dedinec A, Kocarev L. Deep belief network based electricity load forecasting: An analysis of Macedonian case. Energy. 2016;115(Part 3):1688-700.

[23] El-Sharkh MY, Rahman M. Forecasting electricity demand using dynamic artificial neural network model. 2012 International Conference on Industrial Engineering and Operations Management; 2012 Jul 3-6; Istanbul, Turkey. p. 1691-4.

[24] Rashid TA, Huang BQ, Kechadi MT, Gleeson B. Auto-regressive recurrent neural network approach for electricity load forecasting. Int J Comput Intell. 2006;3(1):36-44.

[25] Hatalis K, Pradhan P, Kishore S, Blum RS, Lamadrid AJ. Multi-step forecasting of wave power using a nonlinear recurrent neural network. 2014 IEEE PES General Meeting Conference & Exposition; 2014 Jul 27-31; National Harbor, USA. USA: IEEE; 2014. p. 1-5.

[26] Chen BJ, Chang MW, lin CJ. Load forecasting using support vector machines: a study on eunite competition 2001. IEEE Trans Power Syst. 2004;19(4):1821-30.

[27] Pai PF, Hong WC. Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms. Elec Power Syst Res. 2005;74(3):417-25.

[28] Fard AK, Akbari-Zadeh AR. A hybrid method based on wavelet, ANN and ARIMA model for short-term load forecasting. J Exp Theor Artif Intell. 2014;26(2):167-82.

[29] Ao L, Wang Y, Zhang Q. Application of a hybrid model on short-term load forecasting based on support vector machines (SVM). New Zeal J Agr Res. 2007;50(5):567-72.

[30] Moghram I, Rahman S. Analysis and evaluation of five short-term load forecasting techniques. IEEE Trans Power Syst. 1989;4(4):1484-91.

[31] Lamedica R, Prudenzi A, Sforna M, Caciotta M, Cencellli VO. A neural network based technique for short-term forecasting of anomalous load periods. IEEE Trans Power Syst. 1996;11(4):1749-56.

[32] Wettayaprasit W, Laosen N, Chevakidagarn S. Data filtering technique for neural networks forecasting. Proceedings of the 7th WSEAS International Conference on Simulation, Modelling and Optimization; 2007 Sep 15-17; Beijing, China. p. 225-30.