Combine Particle Swarm Optimization with Artificial Neural Networks for Short-Term Load Forecasting

Chawalit Jeenanunta and K. Darshana Abeyrathn

School of Management Technology, Sirindhorn International Institute of Technology,
Thammasat University, Pathumthani, Thailand
E-mail: chawalit@siit.tu.ac.th, a.darshana1991@gmail.com

***Abstract*—Electricity consumption curves are highly non-linear as many external factors affect the electricity consumption. Artificial Neural Networks (ANNs) are popular in electricity load forecasting since its pattern recognizing and learning abilities of data. Training ANNs are important as it directly affects the forecasts. However, backpropagation training algorithm likely to stops at local minima. Therefore, this research uses Particle Swarm Optimization to train an ANN for forecasting short-term load demand in Thailand. One-year training data is used to forecast the days in 2013. Forecast are evaluated in terms of the Mean Absolute Percentage Error (MAPE). Monthly, yearly, weekdays', Mondays', weekends', holidays', and bridging holidays' average MAPEs from PSO are compared with MAPEs from backpropagation training algorithm. Average MSPEs show that PSO outperforms backpropagation for training ANNs in short-term load forecasting.**

***Index Terms*—Short-Term Load Forecasting, Artificial Neural Networks, Backpropagation, Particle Swarm Optimization.**

I. Introduction

Short-Term Load Forecasting (STLF) helps on making decisions related to the utility activities such as fuel purchasing, maintenance scheduling, and generator scheduling [1]. STLF is one of the three categories of electricity forecasting where it targets on forecasting one hour to one-day electricity consumption [2]. However, time series patterns appearing in each of these categories are different [3]. Forecasting techniques are divided into sub-groups considering the usage of data [4]. Time series forecasting techniques use only the historical data to forecast the future electricity consumption. External factors such as temperature variations can be included to forecast the future electricity consumption with causal forecasting techniques. Therefore, causal forecasting techniques are popular among the researchers due to their promising forecasting outcomes. Some of the well-known causal forecasting techniques are Multiple Regression models, Bayesian techniques, Artificial Neural Networks (ANNs), and Support Vector Machine.

ANNs are widely used in the forecasting field due to the easiness of understanding the concept. ANNs can recognize and learn the non-linear patterns in data series. Weights and bias of ANNs are adjusted based on the patterns in the training data sets before it works with the unseen data. The ANN architecture has been evolved over the years where the Feed forward neural network is the based and the most common ANN architecture to use with many applications [5]. Advantages and disadvantages of these ANN based techniques are discussed in [6].

One of the ways to show the forecasting performance of ANN is comparing the forecasting outcomes of ANN with other techniques. ANNs' forecasts are compared with the forecasts from Support Vector Machine (SVM) in [3] to show that ANN performs better in their research. The same conclusion is made in [7] when ANN compared with the Box-Jenkins method. Considering the case of Ireland electricity demand forecasting, [2] identifies that ANN is better to forecast all three time horizons: Short-Term Load Forecasting (STLF), Medium-Term Load Forecasting (MTLF), and Long-Term Load Forecasting (MTLF).

Since, ANN requires a large number of data and time during the training phase, it is considered as a disadvantage. As forecasts depend on the ANN's parameters and initial values of weights and bias, continuous monitoring is required to get accurate forecasting outcomes. Backpropagation is the most common training algorithm for adjusting the weights and bias of ANNs [8-11]. Since it is a gradient search algorithm, Backpropagation has its inherent training drawback that it can trap in local minima during the training process [8, 12,13].

Drawback of using Backpropagation to train ANNs for forecasting short-term load demand is addressed in this research by integrating ANN with Particle Swarm Optimization (PSO). Considering the ability of solving complex non-linear objective functions with PSO [14], several attempts, where it is used to train ANNs for STLF, can be found in [15-17]. The concept of using PSO to train the ANN for SLTF is practically applied for the case of Thailand using the data from Electricity Generating Authority of Thailand (EGAT).

The rest of the paper is arranged as follows: in the next three sections, basics of ANN, Backpropagation, and PSO are explained, respectively. Design of experiment including data cleaning, data arrangement, ANN arrangement, and training ANN using Backpropagation and PSO is explained before discuss the results and make the conclusion.

II. Artificial Neural Networks

Basic components of ANNs are input and output nodes, hidden layers, hidden neurons, weights that connect each node of the adjust layers, and bias in hidden neurons and output nodes. All these components are illustrated in Fig. I.

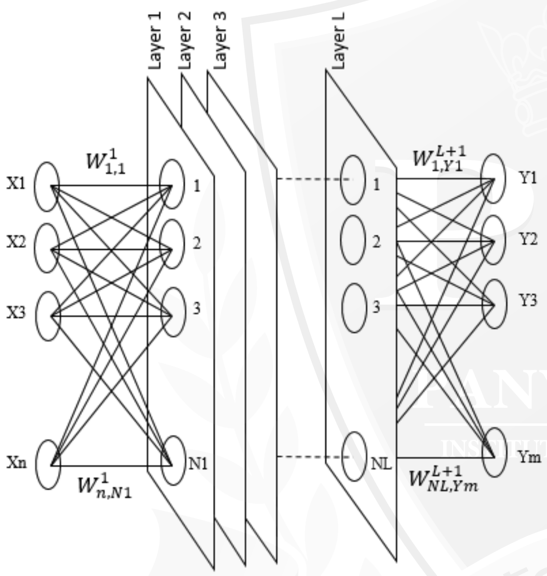


Fig. I. Basic ANN structure

The given basic ANN structure has *X1, X2, X3,…. Xn* input nodes, *Y1, Y2, Y3,….Ym* output nodes, *L* hidden layers, and *N1, N2, N3,….NL* hidden neurons in each hidden layer, respectively. Weights that connect the *Layer* (*L-1*) and *Layer* (*L*) indicate as W^L. As given in the fig. W^1, represents the weights that connect inputs and the hidden nodes in the 1st hidden layer and W^{L+1} represents the weights that connect nodes in *Layer* (*L*) and output nodes. Therefore, $W^1_{1,1}$ is the weight that connects first input node and the first hidden neuron of *Layer 1* and $W^1_{n,N1}$ is the weight that connects n^{th} input nodes and the *N1*th hidden neuron of the *Layer 1*. All these weights are adjusted to get the minimum training error between the target outputs and the network's output for the training inputs. There are different training algorithms to minimize the training error and they take each weight to an optimum value at end of the training iterations called epochs.

III. Backpropagation

Backpropagation is a supervise training algorithm where it uses a target series to adjust the weights and bias. It can be easily found in many ANN related research as it is easier to use than other training algorithms [8-11].

Training inputs move through the randomly initiated weights to get the ANN output. These outputs are compared with the target series to calculate the training error. Training error is propagated back to the input layer for adjusting the weights and complete the first training cycle. This process runs until the training error reaches to the predetermined minimum error or the number of training cycles.

IV. Particle Swarm Optimization

Each particle in PSO is a set of weights that gives different outputs for the training inputs. The fitness of each particle is the Mean Squared Error between the training output for that specific weights and the target series. Weights in each particle are updated to minimize the training error.

After evaluating the fitness of all the particle, PSO updates the personal best (pbest), where the best fitness of each individual particle that has obtained thus far and the global best (gbest), where the best fitness that obtained by any particle for the whole particle set.

Based on the pbest and gbest of the current population, the velocity and potion of each particle are updated for the next generation as given in Eq. (1) and (2), respectively.

$$V^{i}(t+1) = w \times V^{i}(t) + c1 \times rand1 \times (p - X^{i}(t)) + c2 \times rand2 \times (g - X^{i}(t)) \quad (1)$$

$$X^{i}(t+1) = X^{i}(t) + V^{i}(t+1) \quad (2)$$

$V^i(t)$ and $X^i(t)$ are the velocity and position of i^{th} particle at generation *t*, respectively. *p* and *g* represent the position of the pbest and gbest, respectively. *w* is the inertia weight which defines the effect of current velocity for the next generation. rand1, *rand2* are random values between 1 and 0 and c1, c2 are learning factors which have to be predetermined.

V. Design Of Enxperiment

A. Data Cleaning

The data population includes data from March 2009 to December 2013. A sample data set is selected to forecast all the days in 2013 using one year training data. Holidays, bridging holidays, and other outliers due to missing values or measurement errors change the regular patterns appearing in the historical data. Including the abnormal data in the training data set weakens the forecasting accuracy. Therefore, these abnormal data of the data gathered by Electricity Generating Authority of Thailand is removed.

Data cleaning process is started removing the calendar holidays within the sample data period. These holidays' data are replaced with the weighted moving average as given in the following equation.

$$L'_t\,(d) = w_1\,L_t\,(d-7) + w_2\,L_t(d-14) \qquad (3)$$

Where $L_t(d)$ and $L'_t(d)$ are actual and the estimated load data on day d at time period t. w_1 and w_2 are the weight parameters and set to be 0.7 and 0.3, respectively. Then the B' holidays which is a day between two holidays or between a holiday and a weekend are estimated using the same equation that used to estimate the holidays demand.

With the purpose of recognizing other outliers easily and accurately, data are separated into different time-windows considering the similar consumption behavior of similar time periods on similar days. Since there are 7 days per week and 48 time periods for each day, data are divided into 7 48 time-windows as below.

$$V_t(d) \;=\; [L_t(d'), L_t(d'-7), L_t(d'-14), .. L_t(d'-7\times m)] \qquad (4)$$

Where $d \in \{d', d'-7, \ldots, d'-7m\}$, d' is the last 7 days in the data set and m is the number of weeks for each day, d (m = 105 when d = Sunday, Monday, and Tuesday. m = 104 when d = Wednesday, Thursday, Friday, and Saturday). Then the standard deviation of each time-window, and the four-period moving average are used to create the time-window based filtering band, as given in Eq. (5).

$$B_t(d) = \left[\frac{\sum_{i=1}^{4} L_t(d-7\times i)}{4}\right] \pm N \times SD(V_t(d)); t = 1, \ldots 48 \qquad (5)$$

All the data lying outside the time-window based filtering band are identified as the outliers and replaced by the weighted mowing average as given in Eq. (3).

B. Data Arrangement

One year data is prepared to forecast all the days in 2013. Each time period t of each day d is forecasted separately using four inputs to the ANN: previous week's, same time period's load $L'_t(D-7)$, previous day's, same time period's load $L'_t(D-1)$, previous day's, same time period's temperature $T'_t(d)$, and the same day's, same time period's temperature . Since, the target series consists with similar days, only 52 training sets are used to forecast the time periods of Sundays and Mondays. Time periods of all the other days are forecasted using 51 training sets.

C. ANN arrangement

The ANN structure consists with four input nodes and one output node according to the data arrangement. One hidden layer with four hidden neurons is included in the ANN so the total number of weights and bias equal to 25: 16 weights to connect the inputs nodes and the hidden neurons, 4 bias in hidden neurons, 4 weights to connect hidden neurons and the output node, and the bias of the output node. The suggested ANN structure according to the above parameters is illustrated in Fig. II.

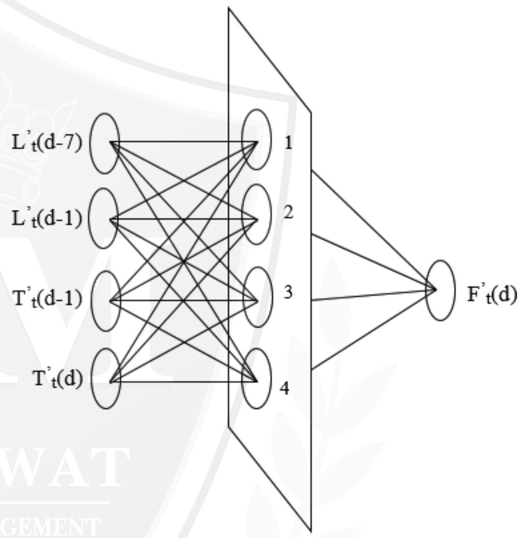


Fig. II. The suggested ANN structure

D. Training ANN with Backpropagation

Once the data and the network are ready, weights and bias of the network can be adjusted by backpropagation training algorithm. The place, where the training algorithm should stop, has to be defined early. The minimum training error (Mean Squared Error between the network outputs and target series) and the maximum number of training cycles are considered as the stopping conditions. These values are set to 0 and 1000, respectively, expecting the maximum training performances.

E. Training ANN with Particle Swarm Optimization

1) Encoding

Weights and bias of the network should have specific places in each particle. The process of placing these weights and bias in particles are called encoding. Since the total number of weights and bias of the suggested ANN is 25, each particle should have the same number of elements in them. Each of these weights and bias are placed in particles as given in Table I.

TABLE I
ENCODING WEIGHTS AND BIAS INTO PARTICLE

Weights/Bias	Elements
$W^1_{1,1}$	1
$W^1_{1,2}$	2
.	.
.	.
$W^1_{1,4}$	16
$b_{(1,1)}$	17
.	.
.	.
$b_{(1,4)}$	20
$W^2_{1,1}$	21
.	.
.	.
$W^2_{4,1}$	24
$b_{(4,1)}$	25

First 16 elements of each particle belong to the weights that connect the input nodes and hidden neurons. Four bias of hidden neurons are placed from 17th element to 20th element. Weights that connect hidden neurons and the output node are placed from 21st element to 24th element. Finally, the bias of the output node is placed at the end.

F. Fitness Values Calculation

During the fitness calculation, elements of each particle are placed in the network at their specific places to get the network output. This output is compared with the target data series and calculated the Mean Squared Error (MSE). Therefore, MSE is the fitness of each particle or the objective function for the PSO. Since the job of the PSO is to minimize the objective or the fitness value, weights are updated at each generation and reach to an optimum set.

1) Setting the Parameters

For updating the pbest and gbest, the fitness of each particle directly involves. However, parameter values have to be defined for updating the velocity and position of each particle. The inertia weight w and the learning factors c1 and c2 are set to 0.6, 2, and 0.5, respectively so that the current velocity and pbest give a higher impact on particles for the next generation. Two random values between 0 and 1 are assigned to rand1 and rand2 each time for updating the velocity and position. The PSO algorithm stops when it reaches to 200 generations, 0 fitness value, or 20 stall generations where there is no or very less improvements given by the fitness values between successive generations.

G. Evaluating the Accuracy

The error between the forecasts and the actual demand is interpreted with Mean Absolute Percentage Error (MAPE). Since the cleaned data is used to train and test the ANN, the error and the MAPE are calculated as given in the following equations.

$$e_t^{'}(d) = L_t^{'}(d) - F_t^{'}(d) \tag{6}$$

$$MAPE_d^{'} = [(\frac{1}{48}) \sum_{t=0}^{48} \left| \frac{e_t^{'}(d)}{L_t^{'}(d)} \right| \times 100 \tag{7}$$

The error $e'_t(d)$ of day d at time t is calculated using the cleaned load $L'_t(d)$ and the forecast $F'_t(d)$ of day d at time t. This error and the cleaned load are used to calculate the MAPE of day d.

VI. RESULTS AND DISCUSSION

Monthly and yearly average $MAPE'$ with backpropagation and PSO training algorithms are given in Table II.

TABLE II
MONTHLY AND YEARLY AVERAGE FOR 2013

Monthly Average $MAPE'$		
Month	BP-ANN	PSO-ANN
January	3.837	3.139
February	3.480	3.215
March	4.253	3.492
April	3.398	2.604
May	3.416	2.938
June	3.591	3.200
July	3.173	2.932
August	2.541	2.574
September	2.653	2.611
October	3.388	3.257
November	2.914	2.554
December	8.470	8.637
Average	3.769	3.439

The yearly average $MAPE'$ with the PSO training algorithm (3.439) is less than the yearly average with backpropagation (3.869). August gives the minimum monthly average $MAPE'$ (2.541) for backpropagation while November gives it for PSO (2.554). The highest monthly average $MAPE'$ for both training algorithm belong to December and these values are equal to 8.470 and 8.637, respectively. Regardless of the training algorithm, the average $MAPE'$ in August, September, and November are lesser than 3 while the average $MAPE'$ in January, February, March, June and October are higher than 4 with the special case of December.

The reason that December has the highest monthly average $MAPE'$ for both backpropagation and PSO is, the electricity consumption in December is lower than the other months even though the holidays and B' holidays are replaced with the estimated data. When the ANN is trained with the consumption values from other months, ANN assumes that December also has the same level of consumption values. Therefore, most of the time, forecasts from ANN for December are higher than the actual consumptions in December. This special case is best illustrated by Fig. 3 with its forecasted and the actual consumption curves for 23rd Monday of December, 2013.

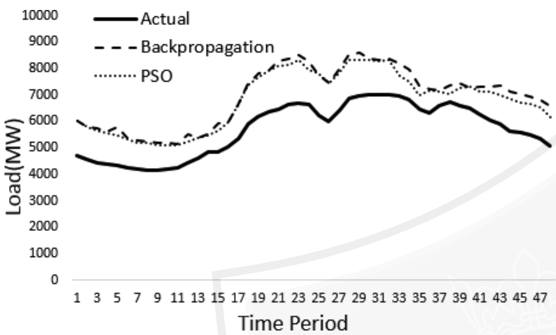


Fig. III. Actual vs Forecasted load for 23rd Monday of December, 2013

Contrast to the above forecasts behavior, Fig. 4 gives accurate forecasting outcomes with both training algorithms.

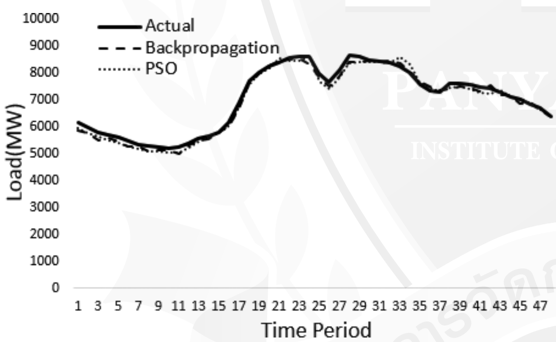


Fig. IV. Actual vs Forecasted load for 25th Wednesday of September, 2013

The day has been selected considering the lower *MAPE'* in August, September, and December. The *MAPE'* on 25th Wednesday of September, 2013 are lower with both training algorithm and it shows by the graph given in Fig. 4.

MAPE' are divided into different categories based on the consumption patterns: weekdays, Mondays, weekends, holidays, and B' holidays. Average *MAPE'* of each category is summarized in Table III. The lowest average *MAPE'* of each category belongs to PSO. From those categories, weekdays' forecasting has lower average *MAPE'* by each training algorithm while holidays' forecasting has the highest average *MAPE'*.

TABLE III
MONTHLY AND YEARLY AVERAGE FOR 2013

	Yearly average *MAPE'*	
Category	BP	PSO
Weekdays	3.317	3.145
Mondays	3.666	3.322
Weekends	4.047	3.491
Holidays	5.262	4.554
B' Holidays	4.795	4.407

VII. CONCLUSION

The objective of the research is to improve the STLF accuracy by reducing the training error of ANN. Backpropagation training algorithm is replaced by PSO algorithm to optimize the weights and bias of the ANN. The ability of optimizing weights and bias of ANN using PSO have been proved for the case of Thailand electricity consumption data for STLF and results are compared with the results from backpropagation training algorithm. Considering the yearly average *MAPE'* for 2013, PSO outperforms the backpropagation training algorithm to train the ANN for STLF. Furthermore, PSO is better for training ANNs to forecast the different categories of days compared to the backpropagation. However, forecasting outcomes for days in December is poor due to its low electricity consumption.

A. Acknowledgment

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

REFERENCES

[1] M. Rana and I. Koprinska. "Forecasting electricity load with advanced wavelet neural networks". *Neurocomputing. 182(c)*. pp. 118-132. Available: https://www.researchgate.net. May. 2016.

[2] J. V. Ringwood, D. Bofelli, and F. T. Murray. Forecasting electricity demand on short, medium and long-time scales using neural networks. *Journal of Intelligent and Robotic Systems. 31(1-3)*, pp. 129-147. Available: https://www.researchgate.net. May.2001.

[3] K. Gajowniczek and T. Ząbkowski . Short term electricity forecasting using individual smart meter data. *Procedia Computer Science. 35*, pp. 589-597. Available: http://www.sciencedirect.com/science/article/pii/S1877050914011053. 2014

[4] S. Nahmais Forecasting. In Production and Operational Analysis. 7th Edition. pp. 51-123. Available: https://www.scribd.com. 2005.

[5] D. Singh and S. Singh . A self-selecting neural network for short-term load forecasting. *Electric Power Components and Systems. 29(2)*, pp. 117-130. Available: http://www.tandfonline.com. 2001.

[6] S. H. Hippert, E. C. Pedreira, and C. R. Souza Neural networks for short-term load forecasting: A review and evaluation. *IEEE Transactions on power system. 16(1)*, pp. 44-55. Available: http://ieeexplore.ieee.org/abstract/document/910780/ Feb. Feb. 2001.

[7] R. Ramakrishna, K. N. Boiroju, and K. M. Reddy . Forecasting Daily Electricity Load Using Neural Networks. *International Journal of Mathematical Archive (IJMA) 2(8)*, pp. 1341-1351. Available: http://ijma.info/index.php/ijma/article/view/461/232. 2001.

[8] H. Shayeghi, A. H. Shayanfar, and G. Azimi. STLF based on optimized neural network using PSO. *International Journal of Electrical and Computer Engineering*. 4(10), pp. 1190-1199. Available: http://www.waset.org/publications/14888.

[9] A. Demiroren and G. Ceylan. Middle anatolian region short-term load forecasting using artificial neural networks. *Electric Power Components and Systems. 34(6))*. pp. 707-724. Available: http://www.tandfonline.com. 2006.

[10] L. Hernández, C. Baladrón, M. J. Aguiar, B. Carro, A. Sánchez-Esguevillas, and J. Lloret . *Artificial neural networks for short-term load forecasting in microgrids environment. Energy*. *75*, pp. 252-264. Available: http://www.sciencedirect.com. Oct. 2014.

[11] T. Senjyu, H. Sakihara, Y. Tamaki, and K. Uezato . Next-day load curve forecasting using neural network based on similarity. *Electric Power Components and Systems. 29(10)*, pp. 939-948. Available: http://www.tandfonline.com.

[12] J. D. Montana and L. Davis Training Feedforward Neural Networks Using Genetic Algorithms. *IJCAI. 89*. pp. 762-767. Available: https://www.ijcai.org Aug. 1989.

[13] K. P. Sarangi, N. Singh, D. Swain, K. R. Chauhan, and R. Singh. Short term load forecasting using neuro genetic hybrid approach: Results analysis with different network architectures. Journal of Theoretical and Applied Information Technology. pp. 109-116. Available:http://jatit.org/volumes/research-papers/Vol10No2/7Vol10No2.pdf. 2009.

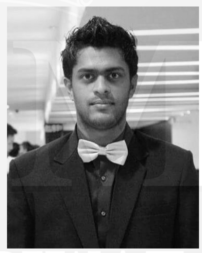
[14] P. Subbaraj and V. Rajasekaran . Evolutionary techniques based combined artificial neural networks for peak load forecasting. *World Acad Sci Eng Technology*. 2(9), pp. 1944-1950. Available: http://www.waset.org/publications/3469. 2008

[15] S. G. Daş (2016). Forecasting the energy demand of Turkey with a NN based on an improved Particle Swarm Optimization. Neural Computing and Applications. pp. 1-11. Available: https://link.springer.com/article/10.1007/s00521-016-2367-8. 2008

[16] R. S. Ul Hassnain and A. Khan. Short term load forecasting using particle swarm optimization based ann approach. Neural Networks. IJCNN. pp. 1476-1481. IEEE. Aug. 2007.

[17] S. Mishra and K. S. Patra . Short term load forecasting using neural network trained with genetic algorithm & particle swarm optimization. In Emerging Trends in Engineering and Technology, First International Conference on. (pp. 606-611). IEEE. Jul. 2008.

Chawalit Jeenanunta is an associate professor of School of Management Technology (MT), Sirindhorn International Institute of Technology, Thammasat University, Thailand. He received a B.S. degree in Mathematics and Computer Science, and M.Sc. in Management Science from University of Maryland and he received his Ph.D. in Industrial and Systems Engineering from Virginia Polytechnic Institute and State University. His Research interests are in area of applications of operations research, simulation, large-scaled optimization and supply chain management.



K.Darshana Abeyrathna is a Master's Student at Sirindhorn International Institute of Technology (SII) Thammasat University. He is currently studying at the School of Management Technology. He received his bachelor's degree from the Asian Institute of Technology (AIT) in Mechatronic Engineering in 2015. He received Partial Fellowship for Excellent Academic Performance during his bachelor's degree for last four consecutive semesters. He also received Excellent Foreign Student (EFS) graduate scholarship at SIIT. His research interests are in Artificial Neural Networks, Data Mining, and Operations Research