A Deep Learning of Time Series for Efficient Analysis

Tippaya Thinsungnoen, Kittisak Kerdprasop, and Nittaya Kerdprasop

Abstract—The main problem for analyzing time series data with machine learning techniques such as classification and clustering is that a high-dimensional nature of this kind of data can cause computational difficulty in finding optimal solution. Currently, advanced learning strategy such as deep learning has been used extensively and effectively to improve learning performance. In this research, we propose a method to optimize time series analysis by adding a pre-training and fine-tuning process of deep learning based on Deep Belief Networks and Restricted Boltzmann Machines. On evaluating performance of the proposed method, we use electroencephalographic, electrocardiogram, and synthetic time series data to analyze with classification task. The induced classification models are assessed with the four several metrics including cluster evaluation, purity, mean squared error, and processing time. We comparatively compare the three learning schemes: traditional neural networks, deep learning networks, and deep learning networks with added a pre-training and fine-tuning process. The results showed that all three schemes show the same performance on predicting time series data when assessed with mean squared error. For the processing time comparison, neural networks technique is slightly faster than others. But when assessed with cluster formation and purity metrics, we found that deep learning based on the concept of Deep Belief Networks and Restricted Boltzmann Machines that adds a pre-training and fine-tuning process outperforms other learning techniques.

Index Terms—Deep belief networks, deep learning, restricted Boltzmann machines, time series analysis.

I. INTRODUCTION

Time series data are a set of values that had been stored consecutively in terms of occurring time and kept for a long period. Data in many applications are being stored as time series such as sales records, stock prices, weather data, and biomedical measurements [1]. An efficient analysis of time series data is a challenging problem in machine learning. The difficulty is due to the fact that time-series data is a type of temporal data, which is naturally high dimensional and large in data size. Researchers have tried to improve efficiency by reducing dimensionality to low-dimensions or introducing new representation method for time-series. The proposed time series representation appeared in the literature includes a symbolic [2], [3] and a grid [4] representation of time series for similarity search.

Besides representation, efficient learning techniques that are appropriate for time series analysis are also in the main focus of my research teams. One potential learning technique is deep learning, which is a complex learning architectural style composing of sub-layers, where each layer contains multiple linear and non-linear transformations attempting to model high level abstractions in data [5]. Researchers offer several learning architectures based on principles of a deep learning such as deep neural networks and deep belief networks. These learning styles have been successfully applied in computer vision, speech recognition, natural language processing, bioinformatics, and others. In the research of [6], they proposed a fast learning algorithm for deep belief nets. In the work of [7], they proposed to reduce the dimensionality of data with neural networks based on the deep autoencoder method. In the same year, the research team [8] proposed a novel and efficient algorithm to sparse feature learning for deep belief networks to capture high-order dependencies between the input observed variables. Furthermore, in the work of [9], they proposed learning algorithm called Deep Boltzmann Machines (DBM's) such that efficiency can be achieved through the pre-training process. Lately, in research work of [10], they also applied pre-training to the ECG assessment based on Restricted Boltzmann Machines (RBM's).

The works mentioned above are a small review of deep learning applied for efficient data analysis, with a particularly interest in high-dimensional data forming themselves as sequences and important knowledge is hidden in these series. This paper thus proposes a study of deep learning of time series for efficient analysis by using a Deep Architecture and Restricted Boltzmann Machines. The data domains include electroencephalographic (EEG), electrocardiogram (ECG), and synthetic time series data. Learning efficiency is evaluated by comparing the results of classification between the three solutions of learning. We use four evaluation measures, that are, cluster evaluation to assess group formation of data, purity in each data group, mean squared error for series prediction, and time used for processing.

II. BACKGROUND

A. Artificial Neural Network

Artificial neural networks (ANN) are a computational approach, based on a large collection of neural units for information processing with connectionist theory. This machine learning approach simulates the pattern recognition function of neural networks in human brain. The bioelectric network in the human brain consists of neurons and synapses, and the interoperation to connect neurons [11]–[13]. Basically, the neural network consists of three layers including input layers, hidden layer and output layer.

B. Deep Learning

Deep learning (or deep structured learning, hierarchical learning, deep machine learning) is an advanced learning technique relying on a set of algorithms that attempt to model high-level abstractions in data. Deep learning is part of a broader family of machine learning methods based on learning representations of data. The concepts include the processing among multiple layers such that each layer is derived from both the linear and nonlinear conversions [5], [13], [14].

C. Deep Belief Networks

Deep Belief Networks (DBNs) is a kind of networks that use probability in a modeling process including multiple layers of stochastic and latent variables. The latent variables typically have binary values called hidden units or feature detectors. The top two layers are undirected graph between itself and memory. The layer below have been directed by the above layer, as shown on the left network of Fig. 1. This network concept has been used as a simple element of unsupervised learning such as RBMs and autoencoder [15].
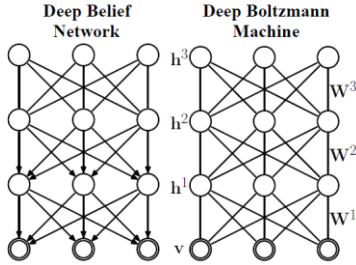

Fig. 1. Model of DBNs and DBMs [9].

D. Deep Boltzmann Machines

A *Boltzmann Machine* is a network of symmetrically coupled stochastic binary units [14]. It consists of a set of visible units $v \in \{0, 1\}$ and a set of hidden units $h \in \{0, 1\}$, as shown on the right of Fig. 1 [9]. The energy of the state $\{v, h\}$ can be defined as:

$$E(v, h; \theta) = -\frac{1}{2} v^T L v - \frac{1}{2} h^T J h - v^T W h, \qquad (1)$$

where $\theta = \{W, L, J\}$ is a set of the model parameters: W, L and J represent visible-hidden, visible-visible, and hidden-hidden, respectively. The diagonal elements of L and J are set to 0.

Deep *Boltzmann Machines* (DBMs) are generally used for learning a deep multilayer Boltzmann Machine [9]. Consider a two-layer Boltzmann Machine with no within-layer connections, the energy of the state $\{v, h^1, h^2\}$ is defined as [9]:

$$E(v, h^1, h^2; \theta) = -v^T W^1 h^1 - h^{1T} W^2 h^2, \qquad (2)$$

where $\theta = \{W^1, W^2\}$ is a pair of model parameters representing visible-hidden and hidden-hidden symmetric interaction terms, respectively.

E. Restricted Boltzmann Machines

A general Boltzmann Machines consist of the top layer to represent a hidden connection and the bottom layer to represent a visible connection, as shown in Fig. 2 (left). But RBMs is a network of symmetrically coupled stochastic binary units, with no hidden-to-hidden and no visible-to-visible connections [9], [16]-[18], as shown in Fig. 2 (right).

RBMs are energy-based probabilistic model. In these models a probability distribution is defined from energy function as [10]:

$$P(x, h) = \frac{e^{-Energy(x,h)}}{Z}, \qquad (3)$$

where x is a set of input variables, and h corresponds to the hidden variables introduced to increase the expressive power of the model. The normalization factor Z is called the partition function defined as [10]:

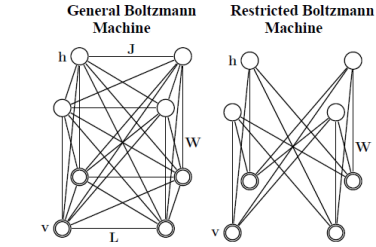$$Z = \sum_{x,h} e^{-Energy(x,h)}, \qquad (4)$$

Fig. 2. Model of general BMs and RBMs [9].

F. Performance Evaluation

There are various measures for evaluating performance if we know previous label of data [18]. This research, we use the three measures: Cluster Evaluation, Purity, and Mean Squared Error (MSE).

Cluster Evaluation is often used for clustering evaluation criteria based on the known label. It is a calculation of an index that measures the amount of agreement between the true cluster partition $G = \{G_1, \ldots, G_k\}$ (the "ground-truth"), and the experimental cluster solution $A = \{A_1, \ldots, A_k\}$ obtained by a machine learning method. The similarity index Sim (G, A) is defined by [19]:

$$Sim(g, A) = \frac{1}{k} \sum_{i=1}^{k} \max_{1 \le j \le k} (G_i, A_j) \qquad (5)$$

$$Sim(g, A) = \frac{|G_i \cap A_j|}{|G_i| + |A_j|} \qquad (6)$$

where $|\bullet|$ denote the member of the set.

Purity is an assessment of how members are organized into class or cluster with the innocence of a member of the cluster. The Purity for members of cluster Ci is defined as follows [18]:

$$purity_i = \frac{1}{n_i} \max_{j=i}^{k} \{n_{ij}\} \qquad (7)$$

where i is an order of clusters from i to r,

j is an order of targets from j to k,

n_i is a member of cluster i,

n_{ij} is a member of cluster i that belongs to class j.

Thus, purity of clustering or classification can be defined as:

$$purity = \sum_{i=1}^{r} \frac{n_i}{n} \; purity_i = \frac{1}{n_i} \max_{j=i}^{k} \{n_{ij}\} \tag{8}$$

Mean *Squared Error* (MSE) is the accuracy index for neural networks model and can be defined as follows [19]:

$$MSE = \sum_{t=1}^{N} \frac{(E_t)^2}{N} \tag{9}$$

where N is total number of data for prediction, E_t is difference (or error) between actual and predicted values of object t.

III. MATERIALS AND METHODOLOGY

A. Dataset Description

The datasets used in this study consist of EEG signals, ECG signals, and Time Series Synthetics. Details of datasets are summarized in Table I.

TABLE I: THE DESCRIPTION OF DATASET

Dataset	Time Series Length	#Series	#Class
EEGs	4,096	20	2
ECGs	96	200	2
Synthetic	200	18	6

Electroencephalographic (EEG) data, taken from [20], [21], consist of five sets: A to E. All EEG signals were recorded with 128-channel amplifier system, using an average common reference. Sets A and B are data sensed from volunteers in a relaxing and awake state with eyes-open (A) and eyes-closed (B), Sets C, D, and E are data originated from EEG archive of presurgical diagnosis. In this research, we use only sets A and B of EEGs data comprising time series of 4,096 samples each, and 10 instances of each class.

Electrocardiogram (ECG) are data taken from [22], [23]. This dataset contains measurements of cardiac electrical activity as recorded from electrodes at various locations on the body; each data set in the ECG recorded by one electrode during one heartbeat. The dataset in each database were analyzed by domain experts; from 200 data records, 133 records were identified as normal and 67 were identified as abnormal.

The Synthetic of Time series (Synthetic.tseries) follows the work of [24] by synthesizing data from six different models; three models were set with both linear and non-linear distributions. Each profile model is shown in Table II.
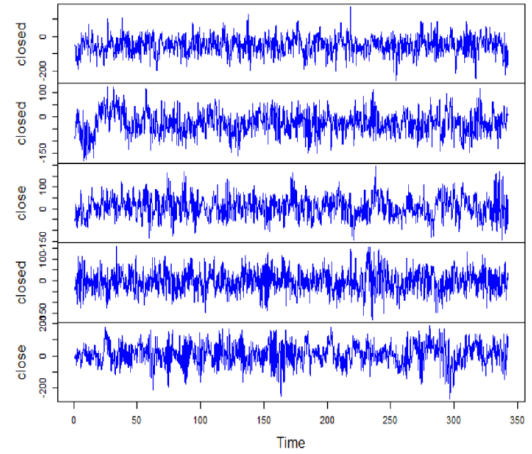
Preview time series of EEG and ECG data, as well as the synthetic data, are shown in Fig. 3 (a), (b), (c), respectively.

In Fig. 3 (a), the preview of EEG signals shows two classes of the whole signal that were very similar. Fig. 3(b) shows previews ECG signals with two classes, that are, classes normal as the S6, S9 and S10 series, and class abnormal as S7 and S8. It can be noticed at time points 40-45 that the two
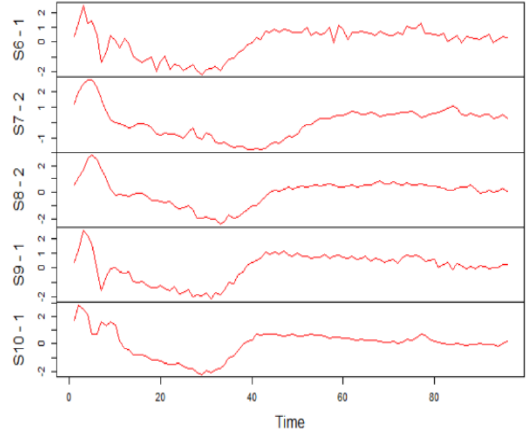
groups have difference changes on the inverse. Fig. 3(c) shows previews of synthesis of time series of complicated shapes derived from different distribution models, thus, resulting in the six different groups that can be clearly differentiated.

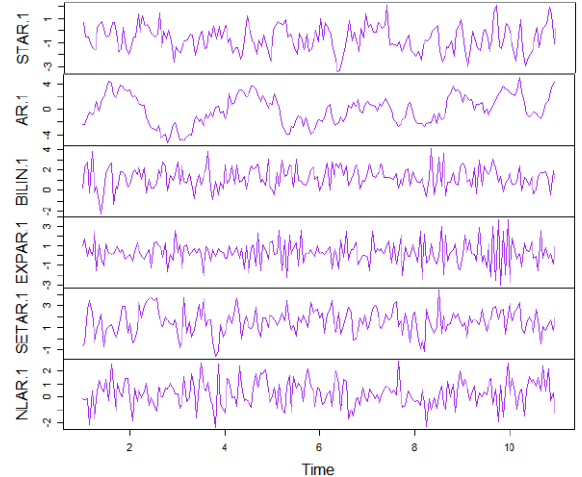TABLE II: THE DESCRIPTION OF SIX SYNTHETICS [24]

Name	Model
AR	$X_t = 0.6 X_{t-1} + \varepsilon_t$
Bilinear	$X_t = (0.3 - 0.2\,\varepsilon_{t-1}) X_{t-1} + 1.0 + \varepsilon_t$
EXPAR	$X_t = (0.9 \exp(-X_{t-1}^2) - 0.6) X_{t-1} + 1.0 + \varepsilon_t$
SETAR	$X_t = (0.3 X_{t-1} + 1)/(X_{t-1} \geq 0.2) - (0.3 X_{t-1} + 1)/(X_{t-1} < 0.2) + \varepsilon_t$
NLAR	$X_t = 0.7 |X_{t-1}| (2 + \lceil X_{t-1} \rceil)^{-1} + \varepsilon_t$
STAR	$X_t = 0.8 X_{t-1} - 0.8 X_{t-1} (1 + \exp(-10 X_{t-1}))^{-1} + \varepsilon_t$

(a) EEG dataset

(b) ECG Dataset

(c) Synthetic time-series

Fig. 3. Preview time series dataset for experimental.

B. Methodology

This research aims at studying a deep learning performance when applied to time series analysis. We employ Deep Architecture and RBMs to find out solution for efficient analysis. The framework of our research shows in Fig. 4. The four main steps can be explained as follows:

Input Raw Time series: This research use the whole raw time series as input to the machines.

Experimental Design: At this step, we divide our experimental method as two sections. Section 1 is the use of Artificial Neural Nets to learn patterns in time series. Section 2 is the application of Deep Architecture and RBMs. The parameter setting of both sections are explained as follows.

Networks consist of three layers. The input layer and a number of neurons equal to the length of time series. Hidden layer contains 10 neurons (which is the best setting from our experiment), and output layer contain one neuron (Fig. 5).
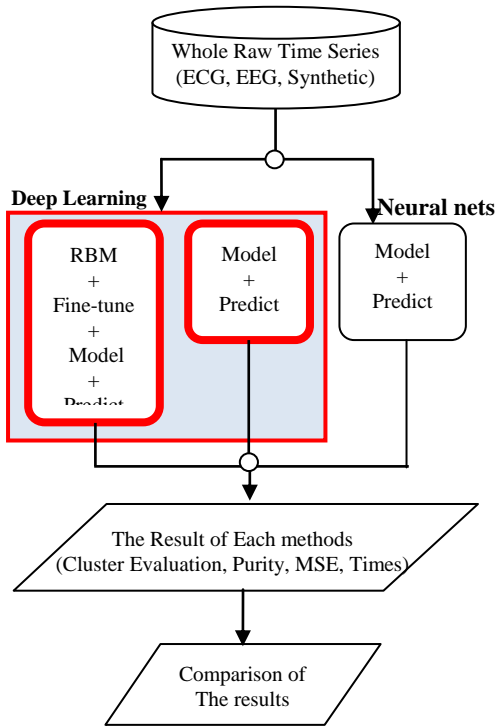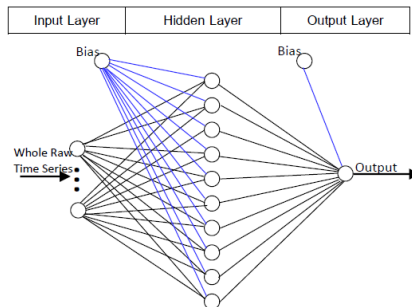

Fig. 4. The framework for this research.

Fig. 5. Model of Neural Networks for this research.

Solutions are based on the setting of 500 epochs for running and stop before the end if small error exists compared to the previous iteration. We design three solutions for analysis. Solution#1 is the analysis with the application of neural nets as a training process for modeling using back propagation. Solution#2 is the use of deep architecture. Solution#3 is a deep learning with pre-training as a startup,

then fine-tune with back propagation and sigmoid function, and end the process with prediction step.

Collection the results: after complete all solutions.

Comparison: comparing of cluster evaluation, purity, MSE, and running time between EEG, ECG and Synthetic datasets.

IV. EXPERIMENTAL RESULTS

This research concerns the study of deep learning based on Deep Architecture and RBMs for effectiveness of time series analysis. The summary of the results is shown in Table III, and the details of the experimental results are follows.

An Evaluated by MSE, the lower is the better. The results show that in Solution#3, Solution#2 and Solution#1, MSE values for EEG, ECG, and synthetic data from all solutions are almost the same at 0.5, 0.335 and 9.17, respectively.

An Evaluated by process time, the lower is the better. The results show that Solution#1 uses the least processing time, but only slightly less than the others. The running time of EEG data for Solution#3, Solution#2 and Solution#1 are 4.78, 3.61, and 2.80 minutes, respectively. The running time of ECG data for Solution#3, Solution#2 and Solution#1 are 4.95, 3.01, and 1.60 minutes, respectively. For synthetic data, the running time from the three solutions are 2.48, 0.95, and 0.76 minutes, respectively.

An Evaluated by cluster evaluation and purity, the higher is the better. These assessment metrics raise an interesting result. In Solution#3, which is the learning based on DBNs and RBMs with the additional pre-training and fine-tuning procedure, the results show the best performance in almost all datasets. The results of cluster evaluation and purity for EEG dataset are 0.949 and 0.95, and for ECG dataset are 0.65 and 0.7, respectively. For the synthetic data, we found that Solution#1 performs slightly better than others with the cluster evaluation and purity showing the same value at 0.39.

TABLE III: THE COMPARING OF THE RESULTS

Dataset	Measures	Solution#1	Solution#2	Solution#3
EEGs	Time (m)	*2.8*	3.611	4.783
	Purity	0.7	0.5	*0.95*
	MSE	*0.5005*	*0.5*	*0.5*
	Cluster Evaluation	0.7	0.67	*0.949*
	Incorrect (%)	6(30%)	10(50%)	*1(5%)*
ECGs	Time (m)	*1.6*	3.01	4.9508
	Purity	*0.7*	*0.7*	*0.7*
	MSE	*0.335*	*0.335*	*0.335*
	Cluster Evaluation	0.55	*0.65*	*0.65*
	Incorrect (%)	88(44%)	*67(33.5%)*	*67(33.5%)*
Synthetic	Time (m)	*0.76*	0.951	2.481
	Purity	*0.39*	0.2	0.2
	MSE	*9.17*	*9.17*	*9.17*
	Cluster Evaluation	*0.39*	0.29	0.29
	Incorrect (%)	*14(77.78%)*	15(83.33%)	15(83.33)

V. CONCLUSION AND FUTURE WORK

A. Conclusion

This research studies the deployment of Deep Learning

method using Deep Architecture and RBMs for analyzing the time series data. The findings for this research are as follows.

Some of Deep architecture such as DBNs and RBMs can be an appropriate method for analyzing the time series data with high-dimensions and a clear sequence pattern. These methods can solve the problem with satisfied accuracy, effectiveness and speed.

An adaptation, such as pre-training and fine-tuning in RBMs, has been experimentally proven to be efficient supporting steps for deep learning more efficiently with a slightly increase in processing time. The study can also conclude that both ANN and RBMs can be used for time series analysis when MSE is the sole concern. But if cluster quality and purity are also additional criteria, RBMs with pre-training and fine-tuning is a better choice for time series analysis.

B. Future Work

Based on the result of synthesis data, created from six different distribution models, we found that ANN performs the best. Deep learning with pre-training and fine-tuning shows remarkably high predictive performance on the EEG dataset. Thus, we plan to extend our research for more studies on the correlation between time series characteristic and deep architectures.

REFERENCES

[1] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering — A decade review," *Information Systems,* vol. 53, pp. 16-38, April 2015.

[2] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *Proc. 8th ACM SIGMOD Workshop on Research Issues Data Mining and Knowledge Discovery – DMKD '03*, 2003, p. 2.

[3] E. Keogh, "Hot sax: Efficiently finding the most unusual time series subsequence," in *Proc. Fifth IEEE International Conference on Data Mining ICDM05*, 2005, pp. 226–233.

[4] G. Duan, Y. Suzuki, and K. Kawagoe, "Grid representation of time series data for similarity search," *The Institute of Electronic, Information, and Communication Engineer*, 2006.

[5] L. Deng and D. Yu, "Deep learning: Methods and applications. foundations and trends," *Signal Processing*, vol. 7, issues 3-4, pp. 197-387, ISSN: 1932-8346, 2014.

[6] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, issues 7, pp. 1527-1554, 2006.

[7] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, issues 5786, pp. 504-507, 2006.

[8] M. A. Ranzato, Y. L. Boureau, and Y. L. Cun, "Sparse feature learning for deep belief networks," *Advances in Neural Information Processing Systems*, pp. 1185-1192, 2008.

[9] R. Salakhutdinov and G. E. Hinton, "Deep Boltzmann machines," *AISTATS*, vol. 1, p. 3, 2009.

[10] V. J. R. Ripoll, A. Wojdel, E. Romero, P. Ramos, and J. Brugada, "ECG assessment based on neural networks with pretraining," *Applied Soft Computing*, vol. 49, pp. 399-406, 2016.

[11] K. Gurney, "An introduction to neural networks," CRC Press, 1997.

[12] H. M. Yao, H. B. Vuthaluru, M. O. Tade, and D. Djukanovic, "Artificial neural network based prediction of hydrogen content of coal in power station boilers," *Fuel*, vol. 84, 2005, pp. 1535–1542.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[14] L. Goodfellow, Y. Bengio, and A. Courville, "Deep learning," MIT Press, 2016.

[15] G. E. Hinton. (2009). Deep belief networks. *Scholarpedia*. [Online]. *4(5)*, p. 5947. Available: http: //www.scholarpedia.org/article/ Deep_belief_networks

[16] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," *Parallel Distributed Processing*, vol. 1, pp. 194-281, 1986.

[17] G. E. Hinton, "Boltzmann machine," *Scholarpedia*, vol. 2, no. 5, 1668, 2007.

[18] J. Z. Mohammed and M. Wagner, "Data mining and analysis: fundamental concepts and algorithms," Cambridge University Press, 2014, USA.

[19] T. T. Nguyen, "Neural network optimal-power-flow," *APSCOM-97,* Hong Kong, November 1997, pp. 266-271.

[20] R. Andrzejak. (2014). Data files. [Online]. Available: http://ntsa.upf.edu/downloads/andrzejak-rg-et-al-2001-indications-no nlinear-deterministic-and-finite-dimensional

[21] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state," *Physical Review E*, vol. 64, no. 6, p. 061907, 2001.

[22] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista. (2015). The UCR time series classification archive. [Online]. Available: www.cs.ucr.edu/~eamonn/time_series_data/

[23] R. T. Olszewski, "Generalized feature extraction for structural pattern recognition in time-series data," School of Computer Science, Carnegie-Mellon University, Pittsburgh, U.S.A., 2001.

[24] P. Montero and A. J. Vilar, "TSclust: An r package for time series clustering," *Journal of Statistical Software*, vol. 62, issue. 1, pp. 1-43, November 2014.

T. Thinsungnoen is currently a doctoral student with the School of Computer Engineering, Suranaree University of Technology (SUT), Thailand. She received her bachelor degree in computer science from Nakhon Ratchasima Rajabhat Institute, Thailand, in 1999, the master degree in computer engineering from SUT in 2007. Her current research of interest includes data mining, deep Boltzmann machines, deep learning of time series data.

K. Kerdprasop is an associate professor and chair of the School of Computer Engineering, SUT. He received his bachelor degree in mathematics from Srinakarinwirot University, Thailand, in 1986, the master degree in computer science from the Prince of Songkla University, Thailand, in 1991 and doctoral degree in Computer Science from Nova Southeastern University, U.S.A., in 1999. His current research includes data mining, artificial intelligence, computational statistics.

N. Kerdprasop is an associate professor with Computer Engineering School, SUT. She received her bachelor degree in radiation techniques from Mahidol University, Thailand, in 1985, the master degree in computer science from the Prince of Songkla University, Thailand, in 1991 and the doctoral degree in computer science from Nova Southeastern University, U.S.A, in 1999. Her research of interest includes data mining, artificial intelligence, and intelligent databases.