Sleep Stages Classification Based on Temporal Pattern Recognition in Neural Network Approach

Ary Noviyanto  
Faculty of Computer Science  
Universitas Indonesia  
Depok, Indonesia  
Email: ary.noviyanto@ui.ac.id

Aniati Murni Arymurthy  
Faculty of Computer Science  
Universitas Indonesia  
Depok, Indonesia  
Email: aniati@cs.ui.ac.id

Abstract—Several previous researchers in sleep stages classification often considered that sleep stages were independent events. They have assumed that every epoch in sleep stages is independent. By nature, sleep is a sequence process so that the current sleep stages will affect to the next sleep stages. Ten datasets of single lead ECG signal from healthy people have been collected. Fifteen features can be extracted from raw ECG signal to describe the sleep stages. Smoothing signal using wavelet denoising is done as the preprocessing steps in order to eliminate noise. Data normalization of input value is also used to handle extreme feature values which will be mapped by activation function in neural network approach.

This paper evaluate contribution of temporal pattern in the sleep stages classification result based on fact that sleep stages is a time series data. Multilayer perceptron (MLP) and Time Delay Neural network (TDNN) using standard back propagation algorithm and moment technique are applied to analyze the contribution of the temporal pattern. TDNN is an extended of MLP that the inputs are sequence of current epoch and previous epoch. TDNN as a classifier that can learn temporal pattern has shown better performance than MLP. It shows that temporal pattern takes a part to determine the correct classification result in the sleep stages classification. An appropriate memory long of temporal pattern is required to get the optimal classification result because longer memory cannot guarantee that the classification result is always better.

Keywords; sleep stages classification, temporal pattern, time series, time delay neural network

I. INTRODUCTION

Time series problems are any problems that related to sequence of time. The example of time series problems are predictions of exchange rate on transaction currencies, Arabic recognition [1], speech recognition [2], traffic forecasting in Asynchronous Transfer Mode (ATM) networks [3], etc. Time series problem is often related with prediction problem but the classification problem can also be considered as time series problem because the classification result of some problems are also affected by temporal pattern; it can be called as time series classification. Time series classification is a unique problem because the target class does not only depend on inputs at the current time but also inputs at the previous time. It can be defined that the time series classification consists of two characteristics; the pattern of the current input and pattern of the input through time.

Sleep stages classification is often thought as independent classification problem that temporal pattern is not considered. The several previous works that related with independent sleep stage classification are Lewicke et al. [4] in 2008, Yilmaz et al. [5] in 2010 and Bsoul et al. [6] in 2010. Lewicke et al. have determined sleep and wake only using ECG signal of infants, Yilmaz et al. have investigated features that derived only from single-lead ECG to classify sleep stages and obstructive apneic epochs using quadratic discriminant analysis (QDA) and support vector machines (SVM) methods and Bsoul et al. have done a research in sleep quality assessment based on ECG measurements.

By nature, sleep is a process that runs through time. It means that a sleep stage is influenced by the previous sleep stage and influences next sleep stage. The temporal pattern of sleep stages will be investigated in sleep stages classification using neural network approach. As an extension of multilayer perceptron, time delay neural network has a particular design to solve the time series classification problems. The performance of time delay neural network and multilayer perceptron will be compared as a representation of classifiers which consider temporal pattern or not.

II. SLEEP STAGES DATASET AND FEATURES

A. Sleep Stages Dataset

Only doctor who is licensed as a sleep expert can annotate sleep stages. The sleep expert analyzes at least using three biological signals; i.e. brain activity (electroencephalogram, EEG), eye movements (electroocculogram, EOG), and chin muscle activity (electromyohram, EMG). This is a standard technique to determine the sleep stages. In this research, the only single ECG (Electrocardiogram) signal is used because the previous works [4, 5, 6] the ECG has promising results to be a substitute of the standard technique to determine the sleep stages.

Sleep stages are determined in a particular length which called epoch. Each epoch is 30 seconds length. The previous
works used different sleep stages. In this research, stages of sleep refer to the newest standard that has been published by American Academy of Sleep Medicine (AASM) in 2007. Based on the standard, sleep stages are divided into five classes; i.e. Wake (W), sleep stage 1 (N1), sleep stage 2 (N2), sleep stage 3 (N3) and sleep stage REM (R) [7]. The ECG signals from ten volunteers have been collected with 200Hz sampling rate. This datasets are coded as slp0, slp1, ..., slp9. The distribution of sleep stages of the ten volunteers are shown in Table 1.

The preprocessing step is used to remove the noise from the acquisition process so that the signals become smoother. The smoothing process is applied to the datasets using stationary wavelet transform with coiflet mother wavelet, four level decomposition, and universal hard thresholding method [8]. The example of denoising process is shown in Fig. 1.

### Table I. Distribution of Sleep Stages of Ten Volunteers.

| Dataset | W  | N1 | N2 | N3 | R  |
|---------|----|----|----|----|----|
| slp0    | 234| 27 | 307| 170| 122|
| slp1    | 114| 49 | 396| 164| 128|
| slp2    | 114| 24 | 244| 206| 224|
| slp3    | 138| 27 | 389| 111| 106|
| slp4    | 208| 51 | 249| 121| 152|
| slp5    | 187| 35 | 271| 145|  47|
| slp6    |  66| 49 | 359| 115| 196|
| slp7    | 119| 49 | 413|  63|  84|
| slp8    | 115| 24 | 253|  68| 196|
| slp9    |  94| 37 | 408| 213| 149|

Figure 1. Denoising using wavelet; a is original signal and b is signal after denoising.

### C. Data Normalization

The values of the features in real life data may have different range. The data may also get the extreme value (i.e. extremely low or extremely high). Data normalization will be an important process when using neural network approach. Neural network uses activation function (e.g. sigmoid function) that makes input data is mapped into specific range with a particular pattern. If the input data are very low than all of the data will be mapped into around 0 and if the input data are very high than all of the data will be mapped into around 1.

Equation (1) is the normalization formula. $x$ is the data, $x'$ is a new value of $x$, $\text{min}$ is the minimum value of a set of the data, $\text{max}$ is the maximum value of a set of the data, $\min'$ is a new minimum bound of the data, and $\max'$ is a new maximum bound of the data. $\min'$ and $\max'$ are set to 0 and 1 respectively.

$$x' = \frac{x - \text{min}}{\text{max} - \text{min}} \times (\max' - \min') + \min'$$  \hfill (1)

### III. Methodology

To evaluate the influence of temporal pattern, the performance of multilayer perceptron and time delay neural network are compared. The number of neuron is predefined; they are 15 neurons and 1 bias neuron in the input layer, 20 neurons and 1 bias neuron in the hidden layer and 5 neurons in the output layer. The number of neuron in hidden layer is calculated in straight forward manner as a summation of the number of neuron in input layer and the number of neuron in output layer.

#### A. Multilayer Perceptron (MLP)

Multilayer neural network that uses single hidden layer was introduced by Funahashi in 1989. It has been proved to approximate all nonlinear functions with the desired accuracy [10]. It can be called as multilayer perceptron (MLP). The structure of MLP consists of three layers; input layer, hidden layer and output layer as shown in Fig. 2. The output of each neuron in hidden layer and output layer will be calculated using a particular activation function. The activation function that used is sigmoid function. Sigmoid function maps the input into 0 to 1 range value so that it works well in case of 0/1 final decision in output layer. It also can be derived so that it can be used in gradient descent based training methods. The equation of sigmoid function shown in (2).

$$f(x) = \frac{1}{1 + e^{-x}}$$

The illustration of MLP to classify sleep stages is depicted in Fig. 3. Every epoch is treated independently. The sleep stage in an epoch is influenced by only features in the epoch.
\[ f(x) = \frac{1}{1 + e^{-x}} \]  
\[ (2) \]

B. Time Delay Neural Network

Time Delay Neural Network (also called Tapped Delay Neural Network) (TDNN) is a neural network that considers temporal pattern [11]. It looks like multilayer perceptron with its input is a sequence of time. The structure of time delay neural network is depicted in Fig. 4. User has to determine a number of backward steps to be the input. With these predetermined backward steps, this kind of neural network can capture temporal pattern.

The illustration of TDNN to classify sleep stages is depicted in Fig. 5. The sleep stage in an epoch is influenced by not only features in the epoch but also features in the previous epoch(s). Fig. 5 shows the using of 2 delays.

\[ \begin{align*}
\text{Output Layer} \\
\text{Hidden Layer(s)}
\end{align*} \]

Figure 4. Structure of Time Delay Time Neural Network

C. Learning Algorithm

The standard back propagation algorithm [12, 13] is used as the learning method for multilayer perceptron and time delay neural network. This algorithm consists of two major phases:

1. Feed-forward. The input will be fed into the network and we can get the output value using weighted sum function as in (3). \( net_o \) is the output of the \( o \)-th neuron in a layer, \( f \) is the activation function, \( net_h \) is the output of \( h \)-th neuron in the previous layer, \( w_{ho} \) is weight that connects between \( h \)-th neuron and \( o \)-th neuron.

\[ net_o = f \left( \sum_h net_h \times w_{ho} + b \right) \]  
\[ (3) \]

2. Back propagation. We calculate gradient of error with respect to a particular weight. The gradient of error is used for weight updating. The each weight in the network is updated according to (4). This technique uses the gradient decent approach.

\[ w_i(t) = w_i(t-1) + \Delta w_i(t) \]  
\[ (4) \]

where \( w_i(t) \) is weight in time \( t \), \( w_i(t-1) \) is the weight in time \( t-1 \) or the previous weight and the \( \Delta w_i(t) \) is calculated as a learning rate (\( \eta \)) times the partial derivative of error function as shown in (5). The error function is square error according to (6).

\[ \Delta w_i(t) = \eta \left( -\frac{\partial E}{\partial w_i} \right) \]  
\[ (5) \]

\[ E = \frac{1}{2} (O_k - d_k)^2 \]  
\[ (6) \]
$O_i$ is the output of the network and $d_k$ is the desired output.

The momentum is also used to enhance the gradient decent approach. This technique can reduce possibilities to be trapped in local minimal and also increase the learning speed [14]. The momentum ($\alpha$) is added in the (5) according to (7). $\Delta w_i(t-1)$ is the previous updated weight factor.

$$\Delta w_i(t) = \eta \left( -\frac{\partial E}{\partial w_i} \right) + \alpha \times \Delta w_i(t - 1) \quad (7)$$

Back propagation algorithm can be instantly used as learning algorithm for TDNN. TDNN can be considered as MLP with $n$ set of input layer, where $n$ is the number of delay. It is clear that TDNN is an extended of MLP. The equation to get the output value of hidden layer is according to (8). $net_i$ is the output of $h$-th neuron in hidden layer, $f$ is the activation function, $net_{ih}$ is output of $i$-th neuron in input layer and $d$-th delay, $w_{dih}$ is weight that connects between $h$-th neuron in hidden layer and $i$-th neuron input layer with $d$-th delay.

$$net_h = f \left( \sum_i \sum_d net_{di} \times w_{dih} + b \right) \quad (8)$$

D. The Best Weight Memorizing

Sometimes the learning process goes to wrong direction. This concept will memorize the set of weight which leads the best performance of the neural network while the training process. The performance is indicated by the value of means square error (MSE). The MSE will be calculated and the set of weight that produces the minimum MSE will be save.

E. Performance Evaluations

Percent correct classification and Kappa statistic are used to measure performance comparison between MLP and TDNN. The percent correct (pc) is calculated using (9). CC is the number of sleep stages which are correctly classified and NC is the total number of sleep stages in a dataset.

$$pc = \frac{CC}{NC} \quad (9)$$

Kappa statistic shows the degree of agreement between classification decision and ground truth [15]. The value of Kappa statistic is in range 0 to 0.99. The interpretation of numerical value of Kappa statistic is shown in Table I.

TABLE II. Interpretation of Kappa Statistic [15].

| Kappa    | Agreement                      |
|----------|--------------------------------|
| < 0      | Less than chance agreement     |
| 0.01 - 0.20 | Slight agreement         |
| 0.21 - 0.40 | Fair agreement             |
| 0.41 - 0.60 | Moderate agreement         |
| 0.61 - 0.80 | Substantial agreement       |
| 0.81 - 0.99 | Almost perfect agreement    |

IV. Experiment

A. Experiment Scenario

The performance of MLP and TDNN will be evaluated to learn the sleep stages pattern. For each dataset, percent correct and Kappa statistic of MLP and TDNN are calculated. The bigger value of percent correct and Kappa statistic means better.

In this research, experiment is done in two sets. All sets use 12500 learning iteration and sigmoid function as an activation function. Set one uses 0.3 learning rate and 0.2 momentum. Set two uses 0.1 learning rate and 0.3 momentum. The delay parameters of TDNN are 1 delay, 2 delays and 3 delays. The experiment aims to evaluate the influence of time pattern in the classification result. The performance that compared is the result of neural network to learn individually of each subject.

B. Experiment Result

The percent correct and Kappa statistic value using 0.3 learning rate and 0.2 momentum are showed in Table 3 and Table 4 respectively. Table 3 can be illustrated in Fig. 6 and Table 4 can be illustrated in Fig. 7. The percent correct and kappa statistic show corresponding result. It can be clearly seen that the graphic of Fig. 6 and Fig. 7 are similar.

TABLE III. Comparison of Percent Correct, Using 0.3 Learning Rate and 0.2 Momentum.

| Data Set | MLP 1 delay | MLP 2 delays | MLP 3 delays | TDNN 1 delay | TDNN 2 delays | TDNN 3 delays |
|----------|-------------|--------------|--------------|--------------|--------------|--------------|
| slp0     | 0.8151      | 0.9512       | 0.9453       | 0.9721       |
| slp1     | 0.5711      | 0.7474       | 0.8484       | 0.7203       |
| slp2     | 0.3571      | 0.4877       | 0.6158       | 0.6515       |
| slp3     | 0.6096      | 0.8288       | 0.9403       | 0.9572       |
| slp4     | 0.3931      | 0.7490       | 0.7721       | 0.7286       |
| slp5     | 0.3956      | 0.4219       | 0.4905       | 0.4131       |
| slp6     | 0.6268      | 0.7822       | 0.7822       | 0.8650       |
| slp7     | 0.5330      | 0.7363       | 0.8063       | 0.7129       |
| slp8     | 0.8018      | 0.9299       | 0.9832       | 0.9924       |
| slp9     | 0.6016      | 0.7769       | 0.8801       | 0.8479       |

![Figure 6. Percent correct using 0.3 learning rate and 0.2 momentum.](image-url)
TABLE IV. COMPARISON OF KAPPA STATISTIC, USING 0.3 LEARNING RATE AND 0.2 MOMENTUM.

| Data Set | MLP      | TDNN 1 delay | TDNN 2 delay | TDNN 3 delay |
|----------|----------|--------------|--------------|--------------|
| slp0     | 0.7586   | 0.9343       | 0.9265       | 0.9623       |
| slp1     | 0.3823   | 0.6546       | 0.7823       | 0.6126       |
| slp2     | 0.2611   | 0.3933       | 0.5005       | 0.5533       |
| slp3     | 0.4631   | 0.7515       | 0.9114       | 0.9369       |
| slp4     | 0.2981   | 0.6787       | 0.7127       | 0.6489       |
| slp5     | 0.1968   | 0.3027       | 0.4051       | 0.2653       |
| slp6     | 0.4617   | 0.6810       | 0.6850       | 0.8039       |
| slp7     | 0.3513   | 0.5978       | 0.7084       | 0.5637       |
| slp8     | 0.7365   | 0.9043       | 0.9767       | 0.9894       |
| slp9     | 0.4839   | 0.6980       | 0.8307       | 0.7904       |

Figure 7. Kappa statistic using learning rate 0.3 and momentum 0.2.

The percent correct and kappa statistic of TDNN always show better result than MLP using 0.3 learning rate and 0.2 momentum. The longer the delay parameter in TDNN shows better performance, but it is not always.

The percent correct and Kappa statistic values using 0.1 learning rate and 0.3 momentum are showed in Table 5 and Table 6 respectively. Table 5 can be illustrated in Fig. 8 and Table 6 can be illustrated in Fig. 9. The percent correct and kappa statistic show corresponding result. It can be clearly seen that the graphic of Fig. 8 and Fig. 9 are similar. This inference is the same with the previous result (i.e. Fig. 6 and Fig. 7). Fig. 8 and Fig. 9 show that the longer delay parameter than the better performance but it is does not always apply in Fig. 6 and Fig. 7. The learning parameter and momentum absolutely take a part to the performance of the neural network.

TABLE V. COMPARISON OF PERCENT CORRECT, USING 0.1 LEARNING RATE AND 0.3 MOMENTUM.

| Data Set | MLP      | TDNN 1 delay | TDNN 2 delay | TDNN 3 delay |
|----------|----------|--------------|--------------|--------------|
| slp0     | 0.7686   | 0.9267       | 0.9570       | 0.9698       |
| slp1     | 0.5946   | 0.7250       | 0.9095       | 0.9589       |
| slp2     | 0.3978   | 0.5702       | 0.6182       | 0.6933       |
| slp3     | 0.5940   | 0.8573       | 0.9442       | 0.9598       |
| slp4     | 0.5211   | 0.7631       | 0.8067       | 0.8707       |
| slp5     | 0.3898   | 0.4540       | 0.4774       | 0.5591       |
| slp6     | 0.6548   | 0.8191       | 0.8764       | 0.9376       |
| slp7     | 0.5934   | 0.6813       | 0.7747       | 0.8723       |
| slp8     | 0.6768   | 0.8643       | 0.9771       | 0.9924       |
| slp9     | 0.6104   | 0.7791       | 0.8835       | 0.9301       |

Figure 8. Percent correct using 0.1 learning rate and 0.3 momentum.

TABLE VI. COMPARISON OF KAPPA STATISTIC, USING 0.1 LEARNING RATE AND 0.3 MOMENTUM.

| Data Set | MLP      | TDNN 1 delay | TDNN 2 delay | TDNN 3 delay |
|----------|----------|--------------|--------------|--------------|
| slp0     | 0.6967   | 0.9023       | 0.9420       | 0.9593       |
| slp1     | 0.4407   | 0.6054       | 0.8712       | 0.9418       |
| slp2     | 0.2903   | 0.4423       | 0.5166       | 0.6154       |
| slp3     | 0.4087   | 0.7899       | 0.9175       | 0.9405       |
| slp4     | 0.4243   | 0.6995       | 0.7532       | 0.8332       |
| slp5     | 0.2348   | 0.3462       | 0.3929       | 0.4715       |
| slp6     | 0.5070   | 0.7467       | 0.8248       | 0.9103       |
| slp7     | 0.3990   | 0.5101       | 0.6699       | 0.7996       |
| slp8     | 0.5894   | 0.8206       | 0.9684       | 0.9894       |
| slp9     | 0.4969   | 0.6986       | 0.8362       | 0.9005       |
V. CONCLUSION

The performance of time delay neural network that can learn temporal pattern always shows better accuracy and agreement than standard multilayer perceptron. The longer delay parameter means that TDNN can memorize longer temporal pattern but cannot give guarantee that the performance is always better. The performance of TDNN is also based on the parameters of learning algorithm, such as learning rate and momentum. Based on the experiment result, the sleep stages classification has been proven as a time series problem so that the temporal pattern will significantly influence in the classification result.

REFERENCES

[1] S. Ismail and A. M. b. Ahmad, Recurrent Neural Network with Backpropagation Through Time Aalgorithm for Arabic Recognition, Proceedings 18th European Simulation Multiconference Graham Horton(c) SCS Europe, 2004.

[2] A. M. Ahmad, S. Ismail, and D. F. Samaon, “Recurrent neural network with backpropagation through time for speech recognition,” In Proceedings of the IEEE international symposium on communications and information technology, vol. 1, pp. 98-102, 2004.

[3] T. Edwards, et al., Traffic trends analysis using neural networks, Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications 3, pp. 157—164, 1997.

[4] A. Lewiecke, et al., Sleep versus wake classification from heat rate variability using computational intelligence: consideration of rejection in classification models, IEEE Trans Biomed Eng, Vol. 55, No. 1, 2008, pp. 108–18.

[5] B. Yilmaz et al., Sleep stage and obstructive apneaic epoch classification using single-lead ecg, BioMedical Engineering OnLine, Vol. 9, No. 1, 2010, pp. 39.

[6] M. Boul et al., Real-time sleep quality assessment using single-lead ECG and multi-stage SVM classifier, in the International Conference of IEEE Engineering in Medicine and Biology Society, 2010, Vol 2010, pp 1178–1181.

[7] Iber C, Ancoli-Israel S, Chesson AL, Quan SF. The AASM Manual for the Scoring of Sleep and Associated Events: Rules, Terminology, and Technical Specifications. Westchester, Ill: American Academy of Sleep Medicine; 2007.

[8] S. M. Isa, A. Noviyanto and A. M. Arymurthy,“ Optimal Selection of Wavelet Thresholding Algorithm for ECG Signal Denoising”, the IEEE International Conference on Advance Computer Science and Information System. 2011.

[9] A. Noviyanto, et al., Selecting Features of Single Lead ECG Signal for Automatic Sleep Stages Classification using Correlation-based Feature Subset Selection, International Journal of Computer Science Issues, 2011, Vol. 8, Issue 5, No 1, pp. 139-148.

[10] P. Thomas and A. Thomas. 2011. Multilayer perceptron for simulation models reduction: Application to a sawmill workshop. Eng. Appl. Artif. Intell. 24, 4 (June 2011), 646-657. DOI=10.1016/j.engappai.2011.01.004 http://dx.doi.org/10.1016/j.engappai.2011.01.004

[11] B. de Vries, and J. C., Principe, “A theory for neural networks with time delays,” Proceedings of the 1990 conference on Advances in neural information processing systems 3, pp. 162–168, 1990.

[12] R. Rojas, Neural Networks: A Systematic Introduction, 1st Edition, Springer, 1996.

[13] Andries P. Engelbrecht, Computational Intelligence An Introduction Second Edition John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England, 2007.

[14] L. Fausett, Backpropagation Through Time and Derivative Adaptive Critics: A Common Framework for Comparison, chapter Englewood Cliffs, NJ: Prentice Hall, 1994.

[15] A. J. Viera and J. M. Garrett, “Understanding Interobserver Agreement: The Kappa Statistic”, Family Medicine, Vol. 37, No. 5, 2005, pp. 360–363.