Improving EEG-based Driver Fatigue Classification using Sparse-Deep Belief Networks

Rifai Chai[1*], Sai Ho Ling[1], Phyo Phyo San[2], Ganesh R. Naik[1], Tuan N. Nguyen[1], Yvonne Tran[1,3], Ashley Craig[3], Hung T. Nguyen[1]

[1]Centre for Health Technologies, Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW 2007, Australia

[2]Data Analytic Department, Institute for Infocomm Research, A*STAR, Singapore

[3]Kolling Institute of Medical Research, Sydney Medical School, The University of Sydney

*** Correspondence:**
Dr. Rifai Chai
Rifai.Chai@uts.edu.au

Keywords: electroencephalography, driver fatigue, autoregressive model, deep belief networks, sparse-deep belief networks

Abstract

This paper presents an improvement of classification performance for electroencephalography (EEG)-based driver fatigue classification between fatigue and alert states with the data collected from 43 participants. The system employs autoregressive (AR) modeling as the features extraction algorithm, and sparse-deep belief networks (sparse-DBN) as the classification algorithm. Compared to other classifiers, sparse-DBN is a semi supervised learning method which combines unsupervised learning for modeling features in the pre-training layer and supervised learning for classification in the following layer. The sparsity in sparse-DBN is achieved with a regularization term that penalizes a deviation of the expected activation of hidden units from a fixed low-level prevents the network from overfitting and is able to learn low-level structures as well as high-level structures. For comparison, the artificial neural networks (ANN), Bayesian neural networks (BNN) and original deep belief networks (DBN) classifiers are used. The classification results show that using AR feature extractor and DBN classifiers, the classification performance achieves an improved classification performance with a of sensitivity of 90.8%, a specificity of 90.4%, an accuracy of 90.6% and an area under the receiver operating curve (AUROC) of 0.94 compared to ANN (sensitivity at 80.8%, specificity at 77.8%, accuracy at 79.3% with AUC-ROC of 0.83) and BNN classifiers (sensitivity at 84.3%, specificity at 83%, accuracy at 83.6% with AUROC of 0.87). Using the sparse-DBN classifier, the classification performance improved further with sensitivity of 93.9%, a specificity of 92.3% and an accuracy of 93.1% with AUROC of 0.96. Overall, the sparse-DBN classifier improved accuracy by 13.8%, 9.5% and 2.5% over ANN, BNN and DBN classifiers respectively.

1 Introduction

Fatigue during driving is a major cause of road accidents in transportation, and therefore poses a significant risk of injury and fatality, not only to the drivers themselves but also to other road users such as passengers, motorbike users, other drivers and pedestrians (Desmond et al., 2012). Driver fatigue reduces the ability to perform essential driving skills such as vehicle steering control, tracking vehicle speed, visual awareness and sufficient selective attention during a monotonous driving condition for a long period of time (Lal and Craig, 2001; Wijesuriya et al., 2007; Craig et al., 2012; Jurecki and Stańczyk, 2014). As a result an automated countermeasure for a driver fatigue system with reliable and improved fatigue classification/detection accuracy is needed to overcome the risk of driver fatigue in transportation (Lal et al., 2003; Vanlaar et al., 2008; Touryan et al., 2013; Touryan et al., 2014; Chai et al., 2016).

In the digital age, machine learning can be used to provide automated prediction of driver fatigue. Two approaches can be used in machine learning, which are the regression and classification methods. The goal of regression algorithms is the prediction of continuous values to estimate driving performance (Lin et al., 2005; Touryan et al., 2013; Touryan et al., 2014). The outcome of classification algorithms is to predict the target class, such as the classification between fatigue and non-fatigue/alert states (Lin et al., 2010; Zhang et al., 2014; Chai et al., 2016; Xiong et al., 2016). The aim of this study is to improve the accuracy of the prediction of fatigue and non-fatigue states. As a result, this study focuses on using an advanced classification method for enhancing the accuracy of a fatigue classification system previously studied (Chai et al., 2016).

As described in a previous paper (Chai et al., 2016), possible driver fatigue assessment includes psychological and physiological measurements (Lal and Craig, 2001; Borghini et al., 2014). For instance, psychological measurement of driver fatigue involves the need for frequent self-report of fatigue status via brief psychometric questionnaires (Lai et al., 2011). Such an approach would be difficult to implement and may well be biased given its subjective nature (Craig et al., 2006). Physiological measurement of the driver fatigue includes video measurement of the face (Lee and Chung, 2012), brain signal measurement using electroencephalography (EEG) (Lal et al., 2003; Lin et al., 2005; Craig et al., 2012; Chai et al., 2016), eye movement tracking system using camera and electrooculography (EOG) (Hsieh and Tai, 2013) and heart rate measurement using electrocardiography (ECG) (Tran et al., 2009; Jung et al., 2014).

Physiological assessment of facial or eye changes using video recording of the driver's face may lead to privacy issues. Physiological measurement strategies like monitoring eye blink rates using EOG and heart rate variability (HRV) using ECG have been shown to reliably detect fatigue (Tran et al., 2009; Hsieh and Tai, 2013). EEG has also been shown to be a reliable method of detecting fatigue, as it directly measures neurophysiological signals that are correlated with mental fatigue (Wijesuriya et al., 2007; Craig et al., 2012; Zhang et al., 2014; Chuang et al., 2015; He et al., 2015; Xiong et al., 2016). Recently, we have shown a classification of EEG-based driver fatigue with the inclusion of new ICA based pre-processing with a promising classification result (Chai et al., 2016), however, it was concluded the classification accuracy needs to be improved. As a result, this paper will extend the work on a potential EEG-based countermeasure driver fatigue system with an improved classification of fatigue vs. alert states.

An EEG-based classification countermeasure system requires several components including EEG signal measurement, signal pre-processing, feature extraction, and classification modules. For feature extraction in EEG analysis, frequency domain data has been widely explored (Lal and Craig, 2001; Craig et al., 2012). Power spectral density (PSD) methods are popular for converting the time domain of EEG signal into the frequency domain (Demandt et al., 2012; Lin et al., 2014). Alternatively, an autoregressive (AR) modelling parametric approach can also be used for feature extraction in an EEG classification system (McFarland and Wolpaw, 2008; Chai et al., 2016; Wang et al., 2016). The advantage of AR modelling is its inherent capacity to model the peak spectra that are characteristic of the EEG signals and it is an all-pole model making it efficient for resolving sharp changes in the spectra. In our previous finding, an AR modelling feature extractor provided superior classification results compared to PSD for EEG-based driver fatigue classification (Chai et al., 2016). Therefore, in this paper, we present the results of applying AR for modeling feature extraction in order to improve the accuracy the classification algorithm. The PSD method is also included for comparison. For the classification, non-linear methods, such as artificial neural networks (ANN), have been used widely in a variety of applications involving EEG (Nguyen, 2008; Casson, 2014). Bayesian neural networks (BNN) have also been used in EEG-based driver fatigue classification (Chai et al., 2016). The Bayesian regularization framework is able to enhance the generalization of neural networks training regardless of finite and/or noisy data.

Recent attention has been focused on improvement of an ANN approach called deep belief networks (DBN) (Hinton et al., 2006; Hinton and Salakhutdinov, 2006; Bengio, 2009; LeCun et al., 2015), which involves a fast, unsupervised learning algorithm for the deep generative model and supervised learning for a discriminative model. The key advantage of this algorithm is the layer-by-layer training for learning a deep hierarchical probabilistic model efficiently as well as a discriminative fine tuning algorithm to optimize performance on the classification problems (Bengio, 2009; LeCun et al., 2015). A DBN classifier is a promising strategy for improving classification of problems including hand-writing character classification (Hinton et al., 2006), speech recognition (Mohamed et al., 2010; Hinton et al., 2012), visual object recognition (Krizhevsky et al., 2012) and other biomedical applications (O'Connor et al., 2013; Stromatias et al., 2015). The training of the DBN is based on the restricted Boltzmann machine (RBM) with layers-wise training of the network per layer at a time from the bottom up (Hinton et al., 2006). Furthermore, the original RBM approach tended to learn a distributed non-sparse representation. A modified version of the RBM using sparse-RBM to form a sparse-deep belief network (sparse-DBN) has shown promising results for modelling low-order features as well as higher-order features for the application of image classification with improved accuracy (Lee et al., 2008; Ji et al., 2014). As a result of this promising advance in classification of complex features, this paper further investigates the classification of EEG signals associated with driver fatigue using the sparse-DBN. For comparison purposes, the results from several different classifiers are included to determine which algorithms are superior with the highest classification performance.

The main contribution of this paper is the combination of the AR modelling feature extractor and sparse-DBN classifier which have not been explored previously for EEG-based driver fatigue classification, with the objective of enhancing the classification performance over past attempts (Chai et al., 2016). The motivation to utilize the sparse-DBN classifier was to investigate its potential superiority for classifying fatigue, in comparison to other classifiers. Sparse-DBN is a semi supervised learning method that combines unsupervised learning for modelling the feature in the pre-training layer and supervised learning for discriminating the feature in the following layer. Incorporating the sparsity in sparse-DBN, achieved with a regularization term that penalizes a deviation of the expected activation of hidden units from a fixed low-level, prevents the network from overfitting and is able to learn low-level structures as well as high-level structures (Ji et al., 2014). The structure of this paper is as follows: section II covers the background and methodology including general structure, EEG experiment and pre-processing, feature extraction and classification. Section III describes results, followed by section IV for discussion and section V for the conclusions.

2 Background and Methodology

2.1 General Structure

The general structure for the EEG-based driver fatigue classification used in this paper is shown in **FIGURE 1** which is divided into four components: (i) the first component involves EEG data collection in a simulated driver fatigue environment; (ii) the second component involves data pre-processing for removing EEG artifact and the moving window segmentation; (iii) the third component involves the features extraction module that converts the signals into useful features; (iv) the fourth component involves the classification module to process the feature and which translates into output via training and classification procedures. The output of the classification comprises two states: fatigue state and alert (non-fatigue) state.

FIGURE 1 | General structure EEG-based driver fatigue classification in this study

2.2 EEG Data Collection

The EEG data collection has been described in a previous paper (Chai et al., 2016). The study was approved by the Human Research Ethics Committee of the University of Technology Sydney (UTS) obtained from previous experiments of driver fatigue study (Craig et al., 2006; Wijesuriya et al., 2007; Craig et al., 2012). The dataset contains electrophysiological data from 43 healthy participants aged between 18 and 55 years who had a current driver's licence. The study involved continuous measurement taken during a monotonous simulated driving task followed by post EEG measures and post-subjective self-report of fatigue. For the simulated driving task, the divided attention steering simulator (DASS) from Stowood scientific instruments was used (Craig et al., 2012). Participants were asked to keep driving at the centre of the road in the simulation task. The participants were also required to respond to a target number that appeared in any of the four corners of the computer screen in front of the participants when they were driving in the experiment, so as to record reaction time.

FIGURE 2 | Moving window segmentation for driver fatigue study

The simulation driving task was terminated if the participant drove off the simulated road for greater than 15 seconds, or if they showed consistent facial signs of fatigue such as head nodding and extended eyes closure, both determined by analysis of participants' faces that occurred throughout the experiment. Three methods were used to validate fatigue occurrence: (i) using video monitoring for consistent physiological signs of fatigue such as tired eyes, head nodding and extended eye closure, verified further by EOG analysis of blink rate and eye closure; (ii) using performance decrements such as deviation off the road, and (iii) using validated psychometrics such as the Chalder Fatigue Scale and the Stanford Sleepiness Scale. Two participants who did not meet the criterion of becoming fatigued were excluded from the dataset. The validation of fatigue versus non-fatigue in these participants has been reported in prior studies (Craig et al., 2006; Craig et al., 2012). The EEG signals were recorded using a 32-channel EEG system, the Active-Two system (Biosemi) with electrode positions at: FP1, AF3, F7, F3, FC1, FC5, T7, C3, CP1, CP5, P7, P3, PZ, PO3, O1, OZ, O2, PO4, P4, P8, CP6, CP2, C4, T8, FC6, FC2, F4, F8, AF4, FP2, FZ and CZ. The recorded EEG data was down sampled from 2048Hz to 256Hz.

2.3 Data Pre-processing and Segmentation

For the alert status, the first 5 mins of EEG data was selected when the driving simulation task began. For the fatigue status, the data was selected from the last 5 mins of EEG data before the task was terminated, after consistent signs of fatigue were identified and verified. Then in each group of data (alert and fatigue), 20s segments were taken with the segment that was chosen being the first 20 seconds where EEG signals were preserved. For the sample this was all within the first 1 minute of the 5 minutes selected. Further artifact removal using an ICA-based method was used to remove blinks, heart and muscle artifact. As a result, 20s of the alert state and 20s of the fatigue state data were available from each participant.

In the pre-processing module before feature extraction, the second-order blind identification (SOBI) and canonical correlation analysis (CCA) were utilized to remove artifacts of the eyes, muscle and heart signals. The pre-processed data were segmented by applying a moving window of 2s with overlapping 1.75s to the 20s EEG data which provided 73 overlapping segments for each state (fatigue and alert states) as shown in **FIGURE 2**. The pre-processing segments were used in the feature extraction module as described in next section.

2.4 Feature Extraction

For comparison purposes and validity of previous work, a feature extractor using the power spectral density (PSD), a widely used spectral analysis of feature extractor in fatigue studies, is provided in this paper.

An autoregressive (AR) model was also applied as a features extraction algorithm in this study. AR modelling has been used in EEG studies as an alternative to Fourier-based methods, and has been reported to have improved classification accuracy in previous studies compared to spectral analysis of the feature extractor (Brunner et al., 2011; Chai et al., 2016). The advantage of AR modelling is its inherent capacity to model the peak spectra that are characteristic of the EEG signals and it is an all-pole model making it efficient for resolving sharp changes in the spectra. The fast Fourier transform (FFT) is a widely used nonparametric approach that can provide accurate and efficient results, but it does not have acceptable spectral resolution for short data segments (Anderson et al., 2009). AR modelling requires the selection of the model order number. The best AR order number requires consideration of both the signal complexity and the sampling rate. If the AR model order is too low, the whole signal cannot be captured in the model. On the other hand, if the model order is too high, then more noise is captured. In a previous study, the AR order number of five provided the best classification accuracy (Chai et al., 2016). The calculation of the AR modelling was as follows:

$$\boldsymbol{\hat{x}(t) = \sum_{k=1}^{P} a(k)\hat{x}(t-k) + e(t)} \qquad (1)$$

where $\boldsymbol{\hat{x}(t)}$ denotes EEG data at time (t), P denotes the AR order number, $e(t)$ denotes the white noise with zero means error and finite variance, and $a(k)$ denotes the AR coefficients.

2.5 Classification Algorithm

The key feature of DBN is the greedy layer-by-layer training to learn a deep, hierarchical model (Hinton et al., 2006). The main structure of the DBN learning is the restricted Boltzmann machine (RBM). A RBM is a type of Markov random field (MRF) which is a graphical model that has a two-layer architecture in which the observed data variables as visible neurons are connected to hidden neurons. A RBM is as shown in which m visible neuron ($v=(v_1, v_2, v_3,\ldots,v_m)$) and n hidden neurons ($h=(h_1, h_2,\ldots, h_n)$) are fully connected via symmetric undirected weights and there is no intra-layer connections within either the visible or the hidden layer.

The connections weights and the biases define a probability over the joint states of visible and hidden neurons through energy function $E(v,h)$, defined as follows:

$$\boldsymbol{E(v,h;\theta) = -\sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} v_i h_j - \sum_{i=1}^{m} a_i v_i - \sum_{j=1}^{n} b_j h_i} \qquad (2)$$

where w_{ij} denotes the weight between v_i and h_j for all $i \in \{1,\ldots, m\}$ and $j \in \{1,\ldots, n\}$; a_i and b_j are the bias term associated with the i^{th} and j^{th} visible and hidden neurons; $\theta = \{W,b,a\}$ is the model parameter with symmetric weight parameters W_{nm}.

For RBM training, the gradient of log probability of a visible vector (v) over the weight w_{ij} with the updated rule calculated by constructive divergence (CD) method is as follows:

$$\Delta \boldsymbol{w_{ij}} = \eta \left(\langle \boldsymbol{v_i h_j} \rangle_{\boldsymbol{data}} - \langle \boldsymbol{v_i h_j} \rangle_{\boldsymbol{recon}} \right) \qquad (3)$$

where η is a learning rate, $\langle \boldsymbol{v_i h_j} \rangle_{\boldsymbol{recon}}$ is the reconstruction of original visible units which is calculated by setting the visible unit to a random training vector. The updating of the hidden and visible states is considered as follows:

$$\boldsymbol{p\left(h_j = 1 \mid v\right) = \sigma\left(b_j + \sum_i v_i w_{ij}\right)} \qquad (4)$$

$$\boldsymbol{p(v_i = 1 \mid h) = \sigma\left(a_i + \sum_i h_j w_{ij}\right)} \qquad (5)$$

where σ is the logistic sigmoid function.

The original RBM tended to learn a distributed, non-sparse representation of the data, however sparse-RBM is able to play an important role in learning algorithms. In an information-theoretic sense, sparse representations are more efficient than the non-sparse ones, allowing for varying of the effective number of bits per example and able to learn useful low-level and high-level feature representations for unlabeled data (ie. unsupervised learning) (Lee et al., 2008; Ji et al., 2014).

This paper uses the sparse-RBM to form the sparse-DBN for EEG-based driver fatigue classification. The sparsity in sparse-DBN is achieved with a regularization term that penalizes a deviation of the expected activation of hidden units from a fixed low-level, which prevents the network from overfitting, as well as allowing it to learn low-level structures as well as high-level structures (Ji et al., 2014). The sparse-RBM is obtained by adding a regularization term to the full data negative log likelihood with the following optimization:

$$\min_{\{w_{ij} a_i b_j\}} \boldsymbol{E(v,h,\theta) - \sum_{l=1}^{m} \log g \sum_{h} P\left(v^{(l)}, h^{(l)}\right) + \lambda \sum_{j=1}^{n} \left| p - \frac{1}{m} \sum_{l=1}^{m} \mathbb{E}\left[h_j^{(l)} \middle| v^{(l)}\right] \right|^2} \qquad (6)$$

where $\mathbb{E}[.]$ is the conditional expectation given the data, λ is a regularization constant and p is a constant controlling the sparseness of the hidden neurons h_j. The DBN is constructed by stacking a predefined number of RBMs to allow each RBM model in the sequence to receive a different representation of the EEG data. The modelling between visible input (v) and N hidden layer h_k is as follows:

$$\boldsymbol{P(v, h^1, \ldots, h^l) = \left(\prod_{k=0}^{l-2} P([h^{(k)} | h^{(k+1)}])\right) P(h^{l-1}, h^l)} \quad (7)$$

where $v = h^0$, $P(h^k|h^{k+1})$ is a conditional distribution for the visible units conditioned on the hidden units of the RBM at level k and $P(h^{l-1},h^l)$ is the visible-hidden joint distribution at the top-level RBM. Two training types of the RBM can be used: generative and discriminative. The generative training of RBM is used as pre-training with un-supervised learning rule. After greedy layer-wise unsupervised learning, the DBN can be used for discriminative ability using the supervised learning. This paper uses a sparse variant of DBN with 2 layers of semi supervised sparse-DBN as shown in **FIGURE 3** with the first layer using the sparse-RBM for generative mode (un-supervised learning) and the second layer using the sparse-RBM in discriminative mode (supervised learning). After layer-by-layer training in DBN, an ANN with back-propagation method is used through the whole classifier to fine-tune the weights for optimal classification.

FIGURE 3 | Structure of sparse-DBN for driver fatigue classification: (A) Greedy learning stack of sparse-RBM; (B) the corresponding sparse-DBN.

The performance indicators, including, sensitivity or true positive rate ($TPR= TP/(TP+FN)$), specificity or true negative rate ($TNR=TN/(TN+FP)$) and accuracy (TP+TN)/(TP+TN+FP+FN), were used for the performance measurement. *TP* (true positive) denotes the number of the fatigue data correctly classified as fatigue state. *FP* (false positive) is the number of alert datasets classified as a fatigue state. *TN* (true negative) is number of alert datasets correctly classified as an alert state. *FN* (false negative) is the fatigue datasets classified as an alert state.

For network learning generalization, we presented the results based on two cross-validation techniques: an early stopping technique and *k*-fold cross-validation. The early stopping technique used the 'hold-out cross validation' – one of the widely used cross validations techniques. Basically, it divided the dataset into three subsets (training, validation and testing sets). The model is trained using the training set while the validation set is periodically used to evaluate the model performance to avoid over-fitting/over-training. The accuracy of the testing set is used as the result of the model's performance. Another cross validation technique is known as *k*-fold cross-validation (k=3). In k-fold cross-validation (k=3), the dataset is divided into three equal (or near equal) sized folds. The training of the network uses 2 folds and the testing the network uses the remaining fold. The process of training and testing is repeated for three possible choices of the subset omitted from the training. The average performance on the three omitted subsets is then used as an estimate of the generalization performance.

Furthermore, a receiver operating characteristic (ROC) graph is used to evaluate further the performance of the proposed method with the compared method for this study. The areas under the curve of the ROC (AUROC) were also computed to evaluate quantitatively the classification performance.

3 Results

From the 32-EEG channel dataset for the 43 participants (2 participants who did not meet the criterion of becoming fatigued were excluded from original 45 participants), 20s of alert state and 20s of fatigue state data were available from each participant. This was fed to the pre-processing module including artifact removal and a 2s moving window segmentation with overlapping 1.75s to the 20s EEG data, providing 73 overlapping segments for each state. As a result, from the 43 participants, a total 6278 units of datasets were formed for the alert and fatigue states (each state having 3139 units).

The segmented datasets were fed to the feature extraction module. AR modelling with the order number of 5 was used for the feature extractor as it provided an optimum result from the previous study (Chai et al., 2016). The size of the AR features equaled the AR order number multiplied with 32 units of EEG channels, thus the AR order number of 5 resulted in 160 units of the AR features. For comparison and validity purposes, this paper includes the PSD, a popular feature extractor in the EEG classification for driver fatigue classification. The spectrum of EEG bands consisted of: delta (0.5-3Hz), theta (3.5-7.5Hz), alpha (8-13Hz) and beta activity (13.5-30Hz). The total power for each EEG activity band was used for the features that were calculated using the numerical integration trapezoidal method, providing 4 units of power values. This resulted in 128 units of total power of PSD for the 32 EEG channels used.

The variant of standard DBN algorithm, sparse-DBN with semi supervised learning used in this paper, comprised of one layer of sparse-RBM with the generative type learning and the second layer of sparse-RBM with discriminative type of learning. The training of the sparse-DBN is done layer-by-layer. The ANN with back-propagation method was used to fine-tune the weights for optimal classification.

TABLE 1 | Testing several values of regularization constant (λ) and the constant controlling the sparseness (p) in order to select values with the lowest MSE (trial-and-error method)

For the discriminative learning of sparse-DBN, the total 6278 datasets were divided into three subsets with similar amounts of number sets: training (2093 sets) validation (2093 sets) and testing sets (2092 sets). The generative learning of sparse-DBN uses unlabeled data from the training sets. For the training of the sparse-DBN using the learning rate (η) of 0.01, the maximum epoch is set to 200, with a regularization constant (λ) of 1, and the constant controlling the sparseness (p) of 0.02. The selection of these training parameters was chosen by trial-and-error, with the chosen values achieving the best training result. **TABLE 1** shows the selection of the regularization constant (λ), with the chosen value of 1 and the constant controlling the sparseness (p) with the chosen value of 0.02, providing lowest the mean square error (MSE) values of 0.00119 (training set) and 0.0521 (validation set) with the iteration number of 69. The average of the MSE values was 0.0046±0.0018 (training set), and 0.0760±0.0124.

FIGURE 4 |Plot of the training and validation MSE for early stopping of classifiers: (A) MSE training and validation of ANN. (B) MSE training of BNN. (C) MSE training of DBN in hidden layer 1 (Generative mode). (D) MSE training of sparse-DBN in hidden layer 1 (Generative mode). (E) MSE training and validation of DBN in hidden layer 2 (Discriminative mode). (F) MSE training and validation of DBN in hidden layer 2 (Discriminative mode).

TABLE 2 | The best MSE and iteration numbers from the training of the classifiers (ANN, BNN, DBN and Sparse-DBN)

In order to prevent overfitting/over-training in the network, a validation-based early stopping method was used for the proposed classifier of sparse-DBN. The plot of the mean square error (MSE) training set and validation set are shown in **FIGURE 4** for classification using AR and sparse-DBN. **TABLE 2** shows the best performance of the training in term of the MSE values and iteration numbers. For comparison, the results for ANN, BNN and DBN classifier are also included.

ANN, DBN and sparse-DBN classifiers utilized the early stopping framework (with the dataset divided into training validation and test sets) to prevent the overfitting problem, except for BNN (where the dataset was divided into training and testing). The BNN used a different framework for preventing the overfitting problem utilizing adaptive hyper-parameters in the cost function to prevent the neural network weight from being too large, which would have resulted in poor generalization. As a result, the validation set is not required for the BNN. A detailed analysis of BNN for EEG based driver fatigue classification has been addressed in our previous study (Chai et al., 2016). The core parameters for the training classifiers (ANN, BNN, DBN and sparse-DBN) are the ANN-based classifier which includes the number of hidden nodes, an activation function and learning rate. In the BNN classifier, an additional hyper-parameter is introduced to fine tune the optimal structure of the ANN. Further, in the sparse-DBN classifier, the regulation constant and constant controlling of sparseness were introduced for the training the DBN classifier. The DBN and sparse-DBN used two hidden layers: the first hidden layer as generative mode (un-supervised learning) and second hidden layer as discriminative mode (supervised learning).

The mean square error (MSE) of the training set decreased smoothly. Using ANN classifier, the training network stopped after 100 iterations as the MSE validation set reached a maximum fail of 10 times the increment value to ensure no over-training happened with the best validation MSE at 0.115. Using a BNN classifier, the training network stopped after 77 iterations as the conditions are met with the BNN parameters with the best validation MSE at 0.0979. Using a DBN classifier in the first hidden layer (generative mode), the training network stopped after 200 iterations with best MSE at 0.434. Using a DBN classifier in the second hidden layer (discriminative mode), the training network stopped after 68 iterations as the MSE validation set reached maximum fail of 10 times increment value to ensure no over-training happened with the best validation MSE at 0.0649. Using the proposed method of sparse-DBN classifier in the first hidden layer (generative mode), the training network stopped after 200 iterations with the best of MSE at 0.388. Using the proposed method of sparse-DBN classifier in the second hidden layer (discriminative mode), the training network stopped after 69 iterations as the MSE validation set reached maximum fail of 10 times increment value to ensure no over-training happened, with the best validation MSE at 0.0520.

Using the classification results from the validation set, the optimal number of hidden neurons of the sparse-DBN is shown in **FIGURE 5**. For the PSD feature extraction, using 10 hidden nodes resulted in the best classification performance. For the AR feature extraction, using 15 hidden nodes produced the best classification performance. These optimal hidden nodes were then used for the training of the network to classify the test set. Also, the results using a different number of layers (two layers, three layers, five layers and ten layers) are also provided in **FIGURE 5**, with the 2 layers (generative mode for the first layer and discriminative mode for second layer) providing the optimal number of layers in this study. This figure shows that using a higher number of layers (three layers, five layers and ten layers) results in a lower accuracy compared to results of using only two layers. Therefore, the two layers sparse-DBN was the chosen architecture providing the higher accuracy. The optimal size of sparse-DBN to classify the PSD features of the EEG-based driver fatigue is [128-10-10-2] and the optimal size of sparse DBN to classify the AR feature is [160-15-15-2]. **TABLE 3** shows the results for the classification of the fatigue state vs. alert state using AR feature extractor and sparse-DBN classifier. For a feature extractor comparison and validity of previous result, the result of the classification using PSD feature extractor method is included. Also for classifier comparison, the classification results using original DBN, BNN and ANN are given.

FIGURE 5 | Plot of the optimal number hidden nodes and layers

First, for the artificial neural network (ANN) classifier: (i) ANN with PSD, for the fatigue data, of a total with 1046 units of actual fatigue dataset, 782 units were correctly classified as fatigue states (true positive: TP), resulting in a sensitivity of 74.8%. For the alert group, of a total of 1046 units of actual alert dataset, 731 units of alert data were correctly classified as alert state (true negative: TN), resulting in a specificity of 69.9%. The combination of ANN and PSD resulted in an accuracy of 72.3%. (ii) ANN with AR, for the fatigue group, of a total of 1046 units of actual fatigue dataset, 845 units of fatigue data were correctly classified as fatigue states (TP), resulting in a sensitivity of 80.8%. For the alert group, of a total of 1046 units of actual alert dataset, 814 units of alert data were correctly classified as alert states (TN), resulting in a specificity of 77.8%, while the combination of ANN with AR resulted in an improved accuracy of 79.3% compared to ANN with PSD.

Second, for the Bayesian neural networks (BNN) classifier: (i) BNN with PSD achieved an improvement compared to ANN with PSD, and for the fatigue group, of a total of 1046 units of actual fatigue dataset, 808 units of fatigue data were correctly classified as fatigue states (TP), resulting in a sensitivity of 77.2%. For the alert state, of a total of 1046 units of actual alert dataset, 791 units of alert data were correctly classified as alert state (TN), resulting in a specificity of 75.6%. The combination BNN with PSD resulted in an accuracy of 76.4%. (ii) BNN with AR achieved an improvement compared to ANN with AR, and ANN with PSD. BNN with PSD, for the fatigue state, of a total of 1046 units of actual fatigue data, 882 units were correctly classified as fatigue states (TP), resulting in a sensitivity of 84.3%. For the alert state, of a total of 1046 units of actual alert data, 868 units of alert data were correctly classified as alert states (TN), resulting in a specificity of 83%. The combination BNN with AR resulted in an accuracy of 83.6%.

TABLE 3 | Results classification fatigue state versus alert state for the test set on different feature extractors and classifiers

Third, when using the deep belief network (DBN) classifier: (i) DBN with PSD achieved a further improvement compared to BNN with PSD, ANN with PSD and ANN with AR; for the fatigue state, of a total of 1046 units of actual fatigue data, 873 units of fatigue data were correctly classified as fatigue states (TP), resulting in a sensitivity of 83.5%. For the alert state, of a total of 1046 units of actual alert data, 833 units of alert data were correctly classified as alert state (TN), resulting in a specificity of 79.6%. The combination DBN with PSD resulted in an accuracy of 81.5%. (ii) DBN with AR achieved further improvement compared to BNN with AR, ANN with AR, DBN with PSD, BNN with PSD and ANN with PSD, for the fatigue state, of a total of 1046 units of actual fatigue data, 950 units of fatigue data were correctly classified as fatigue states (TP), resulting in a sensitivity of 90.8%. For the alert state, of a total of 1046 units of actual alert data, 946 units of alert data were correctly classified as alert states (TN), resulting in a specificity of 90.4%. The combination of DBN with AR resulted in an accuracy of 90.6%.

Fourth, using sparse deep belief networks (sparse-DBN): (i) sparse-DBN with PSD achieved additional improvements compared to DBN with PSD, BNN with PSD, ANN with PSD, BNN with AR and ANN with AR; for the fatigue state, of a total of 1046 units of actual fatigue data, 919 units of fatigue data were correctly classified as fatigue states (TP), resulting in a sensitivity of 87.9%. For the alert state, of a total of 1046 units of actual alert dataset, 855 units of alert data were correctly classified as alert state (TN), resulting in a specificity of 81.7%. The combination sparse-DBN with PSD resulted in an accuracy of 84.8%. (ii) sparse-DBN with AR achieved the most superior result to the other classifier and feature extractor combination with the fatigue state, of a total of 1046 units of actual fatigue data, 982 units of fatigue data were correctly classified as fatigue states (TP), resulting in a sensitivity of 93.9%. For the alert state, of a total of 1046 units of actual alert data, 965 units of alert data were correctly classified as alert states (TN), resulting in a specificity of 92.3%. The combination sparse-DBN with AR resulted in best accuracy of 93.1% compared to the other classifier and feature extractor combinations.

4 Discussion

In summary, using the PSD feature extractor: (i) compared to the ANN classifier, the sparse-DBN classifier improved the classification performance with sensitivity by 13.1% (from 74.8% to 87.9%), specificity by 11.8% (from 69.9% to 81.7%) and accuracy by 12.5% (from 72.3% to 84.8%); (ii) compared to the BNN classifier, the sparse-DBN resulted in improved performance indicators for sensitivity by 10.7% (from 77.2% to 87.9%), specificity by 6.1% (from 75.6% to 81.7%) and accuracy by 8.4% (from 76.4% to 84.8%); (iii) compared to the DBN classifier, the sparse-DBN resulted in improved performance indicators for sensitivity by 4.4% (from 83.5% to 87.9%), specificity by 2.1% (from 79.6% to 81.7%) and accuracy by 3.3% (from 81.5% to 84.8%).

Further, using the AR feature extractor: (i) compared to the ANN classifier, the sparse-DBN classifier improved the classification performance with sensitivity by 13.1% (from 80.8% to 93.9%), specificity by 14.5% (from 77.8% to 92.3%) and accuracy by 13.8% (from 79.3% to 93.1%); (ii) compared to the BNN classifier, the sparse-DBN resulted in improved performance indicators for sensitivity by 9.6% (from 84.3% to 93.9%), specificity by 9.3% (from 83.0% to 92.3%) and accuracy by 9.5% (from 83.6% to 93.1%); (iii) compared to the DBN classifier, the sparse-DBN resulted in improved performance indicators for sensitivity by 3.1% (from 90.8% to 93.9%), specificity by 1.9% (from 90.4% to 92.3%) and accuracy by 2.5% (from 90.6% to 93.1%).

The result of sensitivity (TPR) and specificity (TNR) analyses can also be viewed as the false positive rate (FPR=1−specificity) and false negative rate (FNR = 1−sensitivity). The FPR is the rate of the non-fatigue (alert) state being incorrectly classified as fatigue state. The FNR is the rate of fatigue state being incorrectly classified as an alert state. As a result, the proposed classifier (sparse-DBN) with the AR feature extractor resulted in a sensitivity (TPR) of 93.9%, FNR of 6.1%, specificity (TNR) of 92.3% and FPR of 7.7%. For a real-time implementation, an additional debounce algorithm could be implemented. By adding a debounce component, it masks multiple consecutive false positive detection that may decrease the FPR (Bashashati et al., 2006). The real-time implementation with a debounce algorithm will be a future direction in this area of our study.

TABLE 4 | Results of classification accuracy fatigue state versus alert state with chosen AR feature extractors and different classifiers – *k*-fold cross validation (3 folds) approach

For the early stopping classifier comparison, a *k*-fold cross-validation, a popular method for EEG machine learning, is evaluated as well (Billinger et al., 2012). As a result, this study used k-fold cross-validation (k = 3) with the mean value of ten results of accuracies on each fold. A total of 6278 datasets were divided into three folds (first fold=2093 sets, second fold=2093 sets and third fold= 2092 sets). Overall, the mean value accuracy of three folds was reported. **TABLE 4** shows results using k-fold cross validation approach with the chosen AR feature extraction and different classifiers. The result shows that the mean accuracy using the k-fold cross validation approach is comparable to the early stopping approach with the proposed classifier of sparse-DBN as the best classifier (94.8%±0.011 of sensitivity, 93.3%±0.012 of specificity and 94.1%±0.011 of accuracy) and followed by DBN (90.9%±0.005 of sensitivity, 90.5%±0.005 of specificity and 90.7%±0.005 of accuracy), BNN (84.8%±0.012 of sensitivity, 83.6%±0.015 of specificity and 84.2%±0.014 of accuracy) and ANN (81.4%±0.010 of sensitivity, 78.4%±0.012 of specificity and 79.9%±0.011 of accuracy).

TABLE 5 | Result of Statistical significance of Tukey−Kramer HSD in pairwise comparison

One-way ANOVA was used to compare the four classifiers (ANN, BNN, DBN and sparse-DBN) and the resultant p-value was 9.3666e-07. This p-value corresponding to the F-statistic of one-way ANOVA is much lower than 0.05, suggesting that one or more classifiers are significantly different for which Tukey's HSD test (Tukey−Kramer method) was used to detect where the differences were. The critical value of the Tukey−Kramer HSD Q statistic based on the four classifiers and v = 8 degree of freedom for the error term, were significance levels of α = 0.01 and 0.05 (p-value). The critical value for Q, for α of 0.01 ($Q^{\alpha=0.01}$) is 6.2044 and the critical value for Q for α of 0.05 ($Q^{\alpha=0.05}$) is 4.5293. The Tukey HSD Q-statistic ($Q_{i,j}$) values were calculated for pairwise comparison of the classifiers. In each pair, the statistical significance is found when $Q_{i,j}$ is more than the critical value of Q. **TABLE 5** presents the Tukey HSD Q-statistic ($Q_{i,j}$) and Tukey HSD p-value and Tukey HSD inference of the pairwise comparisons. The results in **TABLE 5** show all six pairwise combinations reached statistical significance (*p<0.05 and **p<0.01). In addition, to compare the proposed classifier (sparse-DBN) and other classifiers (DBN, BNN, ANN), the sparse-DBN vs. DBN resulted in a p-value of 0.021 (*p<0.05), while sparse-DBN vs. BNN and sparse-DBN vs. ANN resulted in a p-value of 0.001 (**p<0.01).

Overall, the combination of the AR feature extractor and sparse-DBN achieved the best result with improved sensitivity, specificity and accuracy for the classification fatigue vs. alert states in a simulated driving scenario.

FIGURE 6 | ROC plot with AUC values for AR feature extractor and ANN, BNN, DBN and sparse-DBN classifiers of early stopping (hold-out cross-validation) technique.

FIGURE 7 | ROC plot with AUC values for AR feature extractor and ANN, BNN, DBN and sparse-DBN classifiers of *k*-fold cross validation (*k*=3) technique.

FIGURE 6 shows the results displayed in the receiver operating characteristic (ROC) curve analyses with AR feature extractor and ANN, BNN, DBN and sparse-DBN classifiers of early stopping (hold-out cross-validation) techniques. The ROC graph is a plot of true positive rate or sensitivity (TPR) on the Y axis and false positive rate (FPR) or 1− specificity on the X axis by varying different threshold ratios as the sweeping variable. A random performance of a classifier would have a straight line connecting (0, 0) to (1, 1). A ROC curve of the classifier appearing in the lower right triangle suggest it performs worse than random guessing and if the ROC curve appears in the upper left, the classifier is believed to have a superior performance classification (Huang and Ling, 2005; Castanho et al., 2007). All ROC curves in **FIGURE 6** for ANN, BNN, DBN and sparse-DBN classifier shows the curves plotted in the upper left or above random guess classification. The result also shows that the ROC curve for sparse-DBN classifier achieved the best upper left curve compared to DBN, BNN and ANN.

The areas under the curve of ROC (AUROC) were also computed to evaluate quantitatively the classification performance. AUROC represents the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative example, and it exhibits several interesting properties compared to accuracy measurement (Huang and Ling, 2005). The AUROC value lies between 0 and 1 with a higher AUROC value indicating a better classification performance. Fig. 6 shows that the classifier using sparse-DBN and AR feature extractor achieved the best performance result with the highest AUROC of 0.9624 compared to original DBN classifier with AUROC of 0.9428, BNN classifier with AUROC 0.8725 and conventional ANN with AUROC of 0.8306.

FIGURE 7 shows the results displayed in the receiver operating characteristic (ROC) curve analyses with AR feature extractor and ANN, BNN, DBN and sparse-DBN classifiers of *k*-fold cross-validation (3 folds) technique with three subplots for each fold. Similar with the ROC plot from the hold-out cross validation technique, all ROC curves in **FIGURE 7** for ANN, BNN, DBN and sparse-DBN classifier shows the curves plotted in the upper left or above random guess classification, and the ROC curve for the sparse-DBN classifier again had best upper left curve compared to DBN, BNN and ANN. For the area under the curve analysis, in first fold (k=1), sparse-DBN achieved the best AUROC of 0.9643 compared to DBN classifier with AUROC of 0.9484, BNN classifier with AUROC of 0.8879 and ANN classifier with AUROC of 0.8419. For second fold (k=2), the sparse-DBN achieved the best AUROC of 0.9673 compared to DBN classifier with AUROC of 0.9520, BNN classifier with AUROC of 0.8968 and ANN classifier with AUROC of 0.8458. For third fold (k=3), the sparse-DBN achieved the best AUROC of 0.9627 compared to DBN classifier with AUROC of 0.9434, BNN classifier with AUROC of 0.8858 and ANN classifier with AUROC of 0.8372.

Our previous work in (Chai et al., 2016) showed a promising result with the inclusion of an additional pre-processing component using a recent independent component analysis (ICA) algorithm, AR feature extractor and BNN classifier. However, it was concluded that the performance of the classification needed to be improved. The findings presented in this paper, strongly suggests that the use of an AR feature extractor provides superior results compared to PSD method, and also extends further the study by improving the reliability including the sensitivity, specificity and accuracy using sparse-DBN classifier in combination with the AR feature extractor, even without the need to include the ICA pre-processing component.

TABLE 6 | Comparison of the training time and testing time for different classifiers

Using chosen classifier parameters, **TABLE 6** shows the comparison of computation times between the proposed classifier (sparse-DBN) and other classifiers (ANN, BNN and DBN). The computational time is estimated using the MATLAB *tic*/*toc* function, where the *tic* function was called before the program and the *toc* function afterward on the computer (Intel Core i5−4570 processor 3.20 GHz, 8-GB RAM). The result shows that for the training time, the sparse-DBN required 169.23±0.93s which was slower compared to other classifiers (86.79±0.24s for DBN, 55.82±2.77s for BNN and 24.02±1.04 for ANN). In terms of the testing (classification) time, all classifiers required the same amount of time of 0.03s or less than a second to complete the task. Although the proposed sparse-DBN required more time to complete the training process, the classifier was able to perform as fast as other classifiers during the testing process. The reason that the testing times of the classifier are comparable to each other was because, after the training process, the final weights were used as constants and in the classification process all classifiers used the same ANN feed-forward classification routine. For the operation of real-time classification, there is no necessity to perform the classifier training again. The classifier just needs to compute the feed forward ANN routine with the saved weight parameters. Thus, sparse-DBN classification time in the runtime mode (execution) is fast, taking less than a second.

The potential future direction of this research includes: (i) real-time driver fatigue with the active transfer learning approach for new user adaptation (Wu et al., 2014; Marathe et al., 2016; Wu, 2016), (ii) improvement of the classification result through an intelligent fusion algorithm, and (iii) testing the efficacy of hybrid driver fatigue detection systems using a combination of physiological measurement strategies known to be related to fatigue status, such as brain signal measurement using electroencephalography (EEG), eye movement and facial tracking systems using camera and electrooculography (EOG) and heart rate variability measurement using electrocardiography (ECG).

5 Conclusions

In this paper, the EEG-based classification of fatigue vs. alert states during a simulated driving task was applied with 43 participants. The AR was used for feature extractor and the sparse-DBN was used as a classifier. For comparison, the PSD feature extractor and ANN, BNN, original DBN were included.

Using the early stopping (hold-out cross validation) evaluation, the results showed that for a PSD feature extractor, the sparse-DBN classifier achieved a superior classification result (sensitivity at 87.9%, specificity at 81.7% and accuracy at 84.8%) compared to the DBN classifier (sensitivity at 83.5%, specificity at 79.6% and accuracy at 81.6%), BNN classifier (sensitivity at 77.2%, specificity at 75.6% and accuracy at 76.4%) and ANN classifier (sensitivity at 74.8%, specificity at 69.9% and accuracy at 72.3%). Further, using an AR feature extractor and the sparse-DBN achieves a significantly superior classification result (sensitivity at 93.9%, specificity at 92.3% and accuracy at 93.1% with AUROC at 0.96) compared to DBN classifier (sensitivity at 90.8%, specificity at 90.4% and accuracy at 90.6% with AUROC at 0.94), BNN classifier (sensitivity at 84.3%, specificity at 83% and accuracy at 83.6% with AUROC at 0.87) and ANN classifier (sensitivity at 80.8%, specificity at 77.8% and accuracy at 79.3% with AUROC at 0.83).

Overall the findings strongly suggest that a combination of the AR feature extractor and sparse-DBN provides a superior performance of fatigue classification, especially in terms of overall sensitivity, specificity and accuracy for classifying the fatigue vs. alert states. The k-fold cross-validation (k=3) also validated that the sparse-DBN with the AR features extractor is the best algorithm compared to the other classifiers (ANN, BNN and DBN), confirmed by a significance of a p-value < 0.05.

It is hoped these results provide a foundation for the development of real-time sensitive fatigue countermeasure algorithms that can be applied in on-road settings where fatigue is a major contributor to traffic injury and mortality (Craig et al., 2006; Wijesuriya et al., 2007). The challenge for this type of technology to be implemented will involve valid assessment of EEG and fatigue based on classification strategies discussed in this paper, while using an optimal number of EEG channels (that is, the minimum number that will result in valid EEG signals from relevant cortical sites) that can be easily applied. These remain the challenges for detecting fatigue using brain signal classification.

6 Author Contributions

RC performed all data analysis and wrote the manuscript. SL, PS, GN and TN advised the analysis and edited the manuscript. YT and AC conceptualized the experiment and edit the manuscript, HN supervised the study, advised the analysis and edited the manuscript.

7 Funding

This study is funded by ‘non-invasive prediction of adverse neural events using brain wave activity’ from Australian Research Council (DP150102493).

8 Acknowledgments

The authors would like to thank Dr Nirupama Wijesuriya for her contribution to the work for collecting the data in this study.

9 References

Anderson, N.R., Wisneski, K., Eisenman, L., Moran, D.W., Leuthardt, E.C., Krusienski, D.J., et al. (2009). An Offline Evaluation of the Autoregressive Spectrum for Electrocorticography. *IEEE Trans. Biomed. Eng.* 56(3)**,** 913-916. doi: 10.1109/tbme.2009.2009767.

Bashashati, A., Fatourechi, M., Ward, R.K., and Birch, G.E. (2006). User customization of the feature generator of an asynchronous brain interface. *Annals of Biomedical Engineering* 34(6)**,** 1051-1060. doi: 10.1007/s10439-006-9097-5.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and trends® in Machine Learning* 2(1)**,** 1-127. doi: 10.1561/2200000006.

Billinger, M., Daly, I., Kaiser, V., Jin, J., Allison, B.Z., Müller-Putz, G.R., et al. (2012). "Is it significant? Guidelines for reporting BCI performance," in *Towards Practical Brain-Computer Interfaces*. Springer), 333-354.

Borghini, G., Astolfi, L., Vecchiato, G., Mattia, D., and Babiloni, F. (2014). Measuring neurophysiological signals in aircraft pilots and car drivers for the assessment of mental workload, fatigue and drowsiness. *Neurosci. Biobehav. Rev.* 44**,** 58-75. doi: 10.1016/j.neubiorev.2012.10.003.

Brunner, C., Billinger, M., Vidaurre, C., and Neuper, C. (2011). A comparison of univariate, vector, bilinear autoregressive, and band power features for brain–computer interfaces. *Med. Biol. Eng. Comput.* 49(11)**,** 1337-1346. doi: 10.1007/s11517-011-0828-x.

Casson, A.J. (2014). Artificial Neural Network classification of operator workload with an assessment of time variation and noise-enhancement to increase performance. *Frontiers in Neuroscience* 8(372). doi: 10.3389/fnins.2014.00372.

Castanho, M.J., Barros, L.C., Yamakami, A., and Vendite, L.L. (2007). Fuzzy receiver operating characteristic curve: an option to evaluate diagnostic tests. *IEEE Trans. Inf. Technol. Biomed.* 11(3)**,** 244-250. doi: 10.1109/TITB.2006.879593.

Chai, R., Naik, G.R., Nguyen, T.N., Ling, S.H., Tran, Y., Craig, A., et al. (2016). Driver Fatigue Classification with Independent Component by Entropy Rate Bound Minimization Analysis in an EEG-based System. *IEEE J. Biomed. Health Informat.* PP(99)**,** 1-1. doi: 10.1109/jbhi.2016.2532354.

Chuang, C.-H., Huang, C.-S., Ko, L.-W., and Lin, C.-T. (2015). An EEG-based perceptual function integration network for application to drowsy driving. *Knowledge-Based Systems* 80**,** 143-152. doi: 10.1016/j.knosys.2015.01.007.

Craig, A., Tran, Y., Wijesuriya, N., and Boord, P. (2006). A controlled investigation into the psychological determinants of fatigue. *Biol. Psychol.* 72(1)**,** 78-87. doi: 10.1016/j.biopsycho.2005.07.005.

Craig, A., Tran, Y., Wijesuriya, N., and Nguyen, H. (2012). Regional brain wave activity changes associated with fatigue. *Psychophysiology* 49(4)**,** 574-582. doi: 10.1111/j.1469-8986.2011.01329.x.

Demandt, E., Mehring, C., Vogt, K., Schulze-Bonhage, A., Aertsen, A., and Ball, T. (2012). Reaching Movement Onset- and End-Related Characteristics of EEG Spectral Power Modulations. *Frontiers in Neuroscience* 6(65). doi: 10.3389/fnins.2012.00065.

Desmond, P.A., Neubauer, M.C., Matthews, G., and Hancock, P.A. (2012). *The Handbook of Operator Fatigue.* Ashgate Publishing, Ltd.

He, Q., Li, W., Fan, X., and Fei, Z. (2015). Driver fatigue evaluation model with integration of multi-indicators based on dynamic Bayesian network. *IET Intelligent Transport Systems* 9(5)**,** 547-554. doi: 10.1049/iet-its.2014.0103.

Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.r., Jaitly, N., et al. (2012). Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. *IEEE Signal Process. Mag.* 29(6)**,** 82-97. doi: 10.1109/msp.2012.2205597.

Hinton, G.E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Comput.* 18(7)**,** 1527-1554. doi: 10.1162/neco.2006.18.7.1527.

Hinton, G.E., and Salakhutdinov, R.R. (2006). Reducing the dimensionality of data with neural networks. *Science* 313(5786)**,** 504-507. doi: 10.1126/science.1127647.

Hsieh, C.-S., and Tai, C.-C. (2013). An improved and portable eye-blink duration detection system to warn of driver fatigue. *Instrum. Sci. Technol.* 41(5)**,** 429-444. doi: 10.1080/10739149.2013.796560.

Huang, J., and Ling, C.X. (2005). Using AUC and accuracy in evaluating learning algorithms. *IEEE Trans. Knowl. Data Eng.* 17(3)**,** 299-310. doi: 10.1109/TKDE.2005.50.

Ji, N.-N., Zhang, J.-S., and Zhang, C.-X. (2014). A sparse-response deep belief network based on rate distortion theory. *Pattern Recognit.* 47(9)**,** 3179-3191. doi: 10.1016/j.patcog.2014.03.025.

Jung, S.j., Shin, H.s., and Chung, W.y. (2014). Driver fatigue and drowsiness monitoring system with embedded electrocardiogram sensor on steering wheel. *IET Intell. Transp. Syst.* 8(1), 43-50. doi: 10.1049/iet-its.2012.0032.

Jurecki, R.S., and Stańczyk, T.L. (2014). Driver reaction time to lateral entering pedestrian in a simulated crash traffic situation. *Transportation research part F: traffic psychology and behaviour* 27, 22-36. doi: 10.1016/j.trf.2014.08.006.

Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). "Imagenet classification with deep convolutional neural networks", in: *Adv. Neural Inf. Process. Syst.*, 1097-1105.

Lai, J.-S., Cella, D., Choi, S., Junghaenel, D.U., Christodoulou, C., Gershon, R., et al. (2011). How item banks and their application can influence measurement practice in rehabilitation medicine: a PROMIS fatigue item bank example. *Arch. Phys. Med. Rehabil.* 92(10), S20-S27. doi: 10.1016/j.apmr.2010.08.033.

Lal, S.K., and Craig, A. (2001). A critical review of the psychophysiology of driver fatigue. *Biol. Psychol.* 55(3), 173-194. doi: 10.1016/S0301-0511(00)00085-5.

Lal, S.K., Craig, A., Boord, P., Kirkup, L., and Nguyen, H. (2003). Development of an algorithm for an EEG-based driver fatigue countermeasure. *J. Saf. Res.* 34(3), 321-328. doi: 10.1016/S0022-4375(03)00027-6.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521(7553), 436-444. doi: 10.1038/nature14539.

Lee, B.G., and Chung, W.Y. (2012). Driver Alertness Monitoring Using Fusion of Facial Features and Bio-Signals. *IEEE Sensors J.* 12(7), 2416-2422. doi: 10.1109/jsen.2012.2190505.

Lee, H., Ekanadham, C., and Ng, A.Y. (2008). "Sparse deep belief net model for visual area V2", in: *Adv. Neural Inf. Process. Syst.*, 873-880.

Lin, C.-T., Wu, R.-C., Jung, T.-P., Liang, S.-F., and Huang, T.-Y. (2005). Estimating driving performance based on EEG spectrum analysis. *EURASIP Journal on Applied Signal Processing* 2005, 3165-3174. doi: 10.1155/ASP.2005.3165.

Lin, C.T., Chang, C.J., Lin, B.S., Hung, S.H., Chao, C.F., and Wang, I.J. (2010). A Real-Time Wireless Brain-Computer Interface System for Drowsiness Detection. *IEEE Transactions on Biomedical Circuits and Systems* 4(4), 214-222. doi: 10.1109/TBCAS.2010.2046415.

Lin, C.T., Chuang, C.H., Huang, C.S., Tsai, S.F., Lu, S.W., Chen, Y.H., et al. (2014). Wireless and Wearable EEG System for Evaluating Driver Vigilance. *IEEE Trans. Biomed. Circuits Syst.* 8(2), 165-176. doi: 10.1109/tbcas.2014.2316224.

Marathe, A.R., Lawhern, V.J., Wu, D., Slayback, D., and Lance, B.J. (2016). Improved Neural Signal Classification in a Rapid Serial Visual Presentation Task Using Active Learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 24(3), 333-343. doi: 10.1109/TNSRE.2015.2502323.

McFarland, D.J., and Wolpaw, J.R. (2008). Sensorimotor rhythm-based brain–computer interface (BCI): model order selection for autoregressive spectral analysis. *J. Neural Eng.* 5(2), 155. doi: 10.1088/1741-2560/5/2/006.

Mohamed, A.-r., Yu, D., and Deng, L. (2010). "Investigation of full-sequence training of deep belief networks for speech recognition", in: *INTERSPEECH*, 2846-2849.

Nguyen, H.T. (2008). Intelligent technologies for real-time biomedical engineering applications. *Int. J. Autom. Control* 2(Nos.2/3)**,** 274-285. doi: 10.1504/IJAAC.2008.022181.

O'Connor, P., Neil, D., Liu, S.-C., Delbruck, T., and Pfeiffer, M. (2013). Real-time classification and sensor fusion with a spiking deep belief network. *Frontiers in Neuroscience* 7(178). doi: 10.3389/fnins.2013.00178.

Stromatias, E., Neil, D., Pfeiffer, M., Galluppi, F., Furber, S.B., and Liu, S.-C. (2015). Robustness of spiking Deep Belief Networks to noise and reduced bit precision of neuro-inspired hardware platforms. *Frontiers in Neuroscience* 9(222). doi: 10.3389/fnins.2015.00222.

Touryan, J., Apker, G., Kerick, S., Lance, B., Ries, A.J., and McDowell, K. (2013). "Translation of EEG-based performance prediction models to rapid serial visual presentation tasks", in: *International Conference on Augmented Cognition*, 521-530.

Touryan, J., Apker, G., Lance, B.J., Kerick, S.E., Ries, A.J., and McDowell, K. (2014). Estimating endogenous changes in task performance from EEG. *Frontiers in Neuroscience* 8(155). doi: 10.3389/fnins.2014.00155.

Tran, Y., Wijesuriya, N., Tarvainen, M., Karjalainen, P., and Craig, A. (2009). The relationship between spectral changes in heart rate variability and fatigue. *J. Psychophysiol.* 23(3)**,** 143-151. doi: 10.1027/0269-8803.23.3.143.

Vanlaar, W., Simpson, H., Mayhew, D., and Robertson, R. (2008). Fatigued and drowsy driving: A survey of attitudes, opinions and behaviors. *J. Saf. Res.* 39(3)**,** 303-309. doi: 10.1016/j.jsr.2007.12.007.

Wang, G., Sun, Z., Tao, R., Li, K., Bao, G., and Yan, X. (2016). Epileptic Seizure Detection Based on Partial Directed Coherence Analysis. *IEEE J. Biomed. Health Informat.* 20(3)**,** 873-879. doi: 10.1109/jbhi.2015.2424074.

Wijesuriya, N., Tran, Y., and Craig, A. (2007). The psychophysiological determinants of fatigue. *Int. J. Psychophysiol.* 63(1)**,** 77-86. doi: 10.1016/j.ijpsycho.2006.08.005.

Wu, D. (2016). Online and Offline Domain Adaptation for Reducing BCI Calibration Effort. *IEEE Transactions on Human-Machine Systems* PP(99)**,** 1-14. doi: 10.1109/THMS.2016.2608931.

Wu, D., Lance, B., and Lawhern, V. (2014). "Transfer learning and active transfer learning for reducing calibration data in single-trial classification of visually-evoked potentials", in: *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2801-2807.

Xiong, Y., Gao, J., Yang, Y., Yu, X., and Huang, W. (2016). Classifying Driving Fatigue Based on Combined Entropy Measure Using EEG Signals. *International Journal of Control and Automation* 9(3)**,** 329-338. doi: 10.14257/ijca.2016.9.3.30.

Zhang, C., Wang, H., and Fu, R. (2014). Automated Detection of Driver Fatigue Based on Entropy and Complexity Measures. *IEEE Transactions on Intelligent Transportation Systems* 15(1)**,** 168-177. doi: 10.1109/TITS.2013.2275192.

TABLE 1 | Testing several values of regularization constant (λ) and the constant controlling the sparseness (p) in order to select values with the lowest MSE (trial-and-error method)

Regularization constant(λ)	Sparseness constant (p)	MSE training	MSE validation	Iteration number
0.5	0.1	0.00492	0.06625	90
1	0.1	0.00680	0.06710	82
2	0.1	0.00676	0.07961	64
0.5	0.01	0.00542	0.07365	66
1	0.01	0.00507	0.08360	71
2	0.01	0.00395	0.06831	85
0.5	0.02	0.00288	0.07664	73
<u>1</u>	**<u>0.02</u>**	**<u>0.00119</u>**	**<u>0.05206</u>**	**<u>69</u>**
2	0.02	0.00288	0.07181	66
0.5	0.03	0.00327	0.08289	88
1	0.03	0.00574	0.09207	73
2	0.03	0.00665	0.09825	89
Mean		0.004629	0.07615	76.42
SD		0.001803	0.01269	9.72

TABLE 2 | The best MSE and iteration numbers from the training of the classifiers (ANN, BNN, DBN and Sparse-DBN)

Classifiers	Best MSE	Best Iteration Number
ANN	0.115	110
BNN	0.0979	77
DBN	0.0649	68
Sparse-DBN	0.0520	69

TABLE 3 | Results classification fatigue state versus alert state for the test set on different feature extractors and classifiers – early stopping approach

Feature Extraction Methods:	Classification Results	Classification Methods:			
		ANN	**BNN**	**DBN**	**Sparse-DBN**
PSD	TP	782	808	873	919
	FN	264	238	173	127
	TN	731	791	833	855
	FP	315	255	213	191
	Sensitivity (%)	74.8%	77.2%	83.5%	87.9%
	Specificity (%)	69.9%	75.6%	79.6%	81.7%
	Accuracy (%)	72.3%	76.4%	81.5%	84.8%
AR	TP	845	882	950	982
	FN	201	164	96	64
	TN	814	868	946	965
	FP	232	178	100	81
	Sensitivity (%)	80.8%	84.3%	90.8%	**93.9%**
	Specificity (%)	77.8%	83.0%	90.4%	**92.3%**
	Accuracy (%)	79.3%	83.6%	90.6%	**93.1%**

TABLE 4 | Results of classification accuracy fatigue state versus alert state with chosen AR feature extractors and different classifiers – *k*-fold cross validation (3 folds) approach

Classification Results	Classification Methods:			
	ANN (Mean±SD)	**BNN (Mean±SD)**	**DBN (Mean±SD)**	**Sparse-DBN (Mean±SD)**
TP	852.0±10.583	888.0±13.229	951.3±4.933	992±11.930
FN	194.7±10.408	158.7±13.051	95.3±4.726	54.3±11.719
TN	820.3±13.051	874.7±15.308	947.0±5.292	976.0±12.288
FP	225.7±13.051	171.3±15.308	99.0±5.292	70.0±12.288
Sensitivity	81.4%±0.010	84.8%±0.012	90.9%±0.005	**94.8%**±0.011
Specificity	78.4%±0.012	83.6%±0.015	90.5%±0.005	**93.3%**±0.012
Accuracy	79.9%±0.011	84.2%±0.014	90.7%±0.005	**94.1%**±0.011

TABLE 5 | Result of Statistical significance of Tukey−Kramer HSD in pairwise comparison

Pairwise Comparison	**Tukey HSD *Q* statistic**	**Tukey HSD *p*-value**	**Tukey HSD inference**
Sparse DBN vs. DBN	5.376	0.021	***p*<0.05**
Sparse DBN vs. BNN	15.795	0.001	****p*<0.01**
Sparse DBN vs. ANN	22.733	0.001	****p*<0.01**
DBN vs. BNN	10.419	0.001	***p*<0.01
DBN vs. ANN	17.357	0.001	***p*<0.01
BNN vs. ANN	6.938	0.005	***p*<0.01

TABLE 6 | Comparison of the training time and testing time for different classifiers

Classifiers	**Training time (s) (Mean±SD)**	**Testing time (s) (Mean±SD)**
ANN	24.02±1.04	0.0371±0.0023
BNN	55.82±2.77	0.0381±0.0082
DBN	86.79±0.24	0.0334±0.0016
Sparse-DBN	169.23±0.93	0.0385±0.0043

Figure Legends

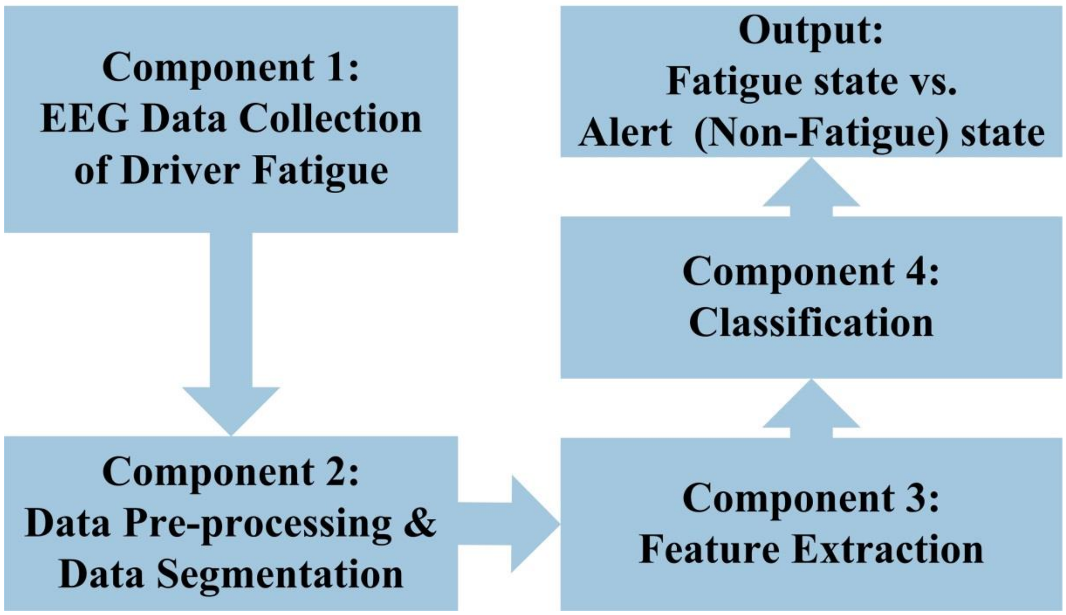

FIGURE 1 | General structure EEG-based driver fatigue classification in this study

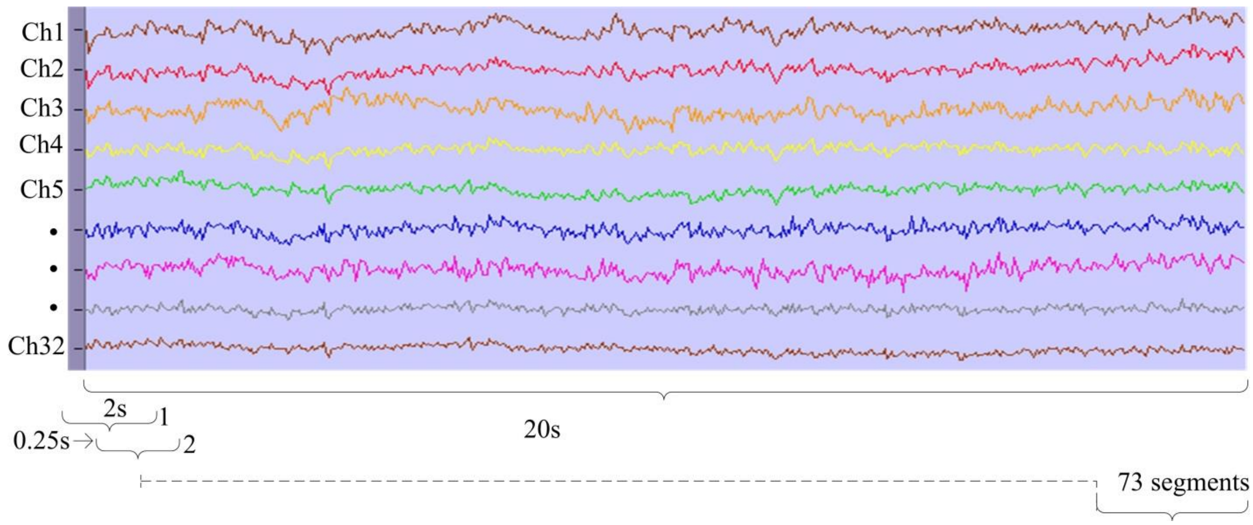

FIGURE 2 | Moving window segmentation for driver fatigue study

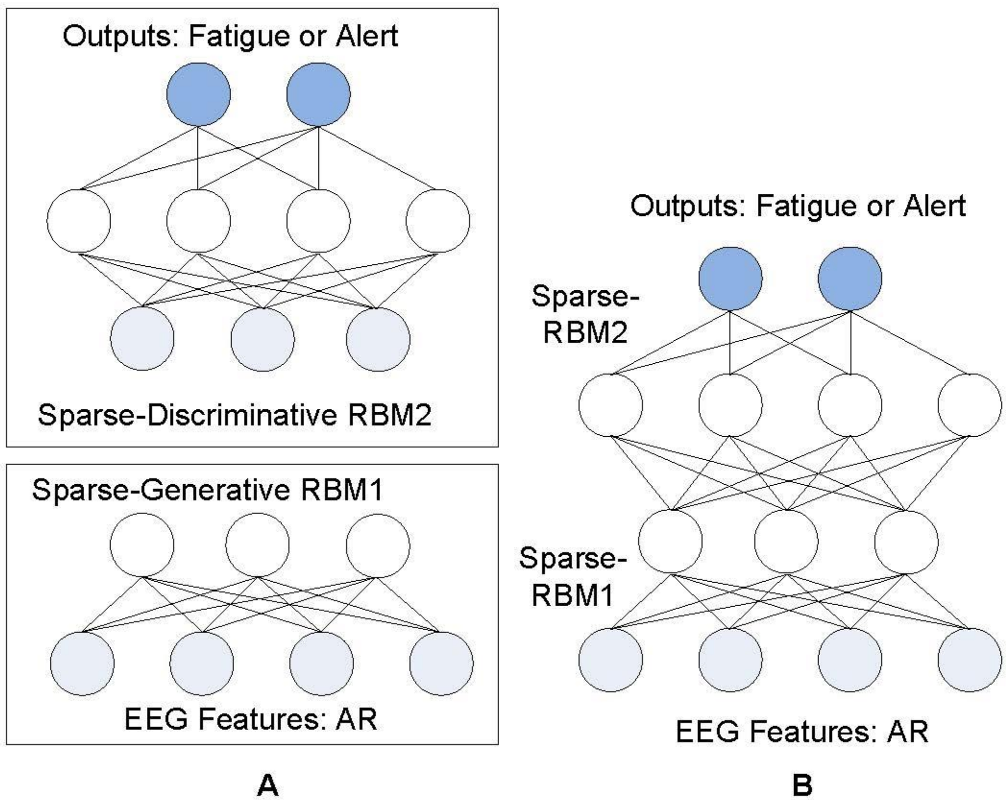

FIGURE 3 | Structure of sparse-DBN for driver fatigue classification: (A) Greedy learning stack of sparse-RBM; (B) the corresponding sparse-DBN.

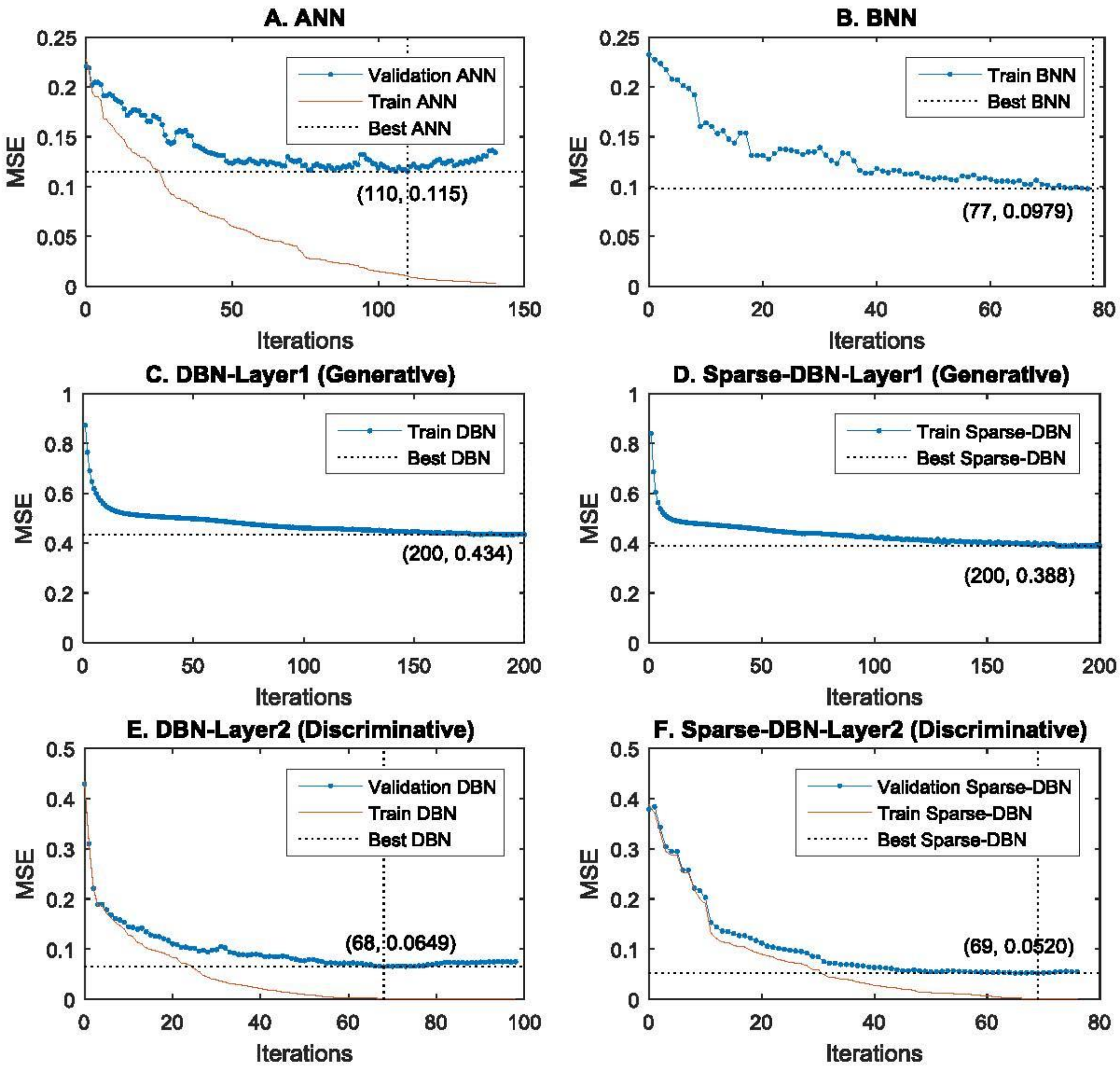

FIGURE 4 |Plot of the training and validation MSE for early stopping of classifiers: (A) MSE training and validation of ANN. (B) MSE training of BNN. (C) MSE training of DBN in hidden layer 1 (Generative mode). (D) MSE training of sparse-DBN in hidden layer 1 (Generative mode). (E) MSE training and validation of DBN in hidden layer 2 (Discriminative mode). (F) MSE training and validation of DBN in hidden layer 2 (Discriminative mode).

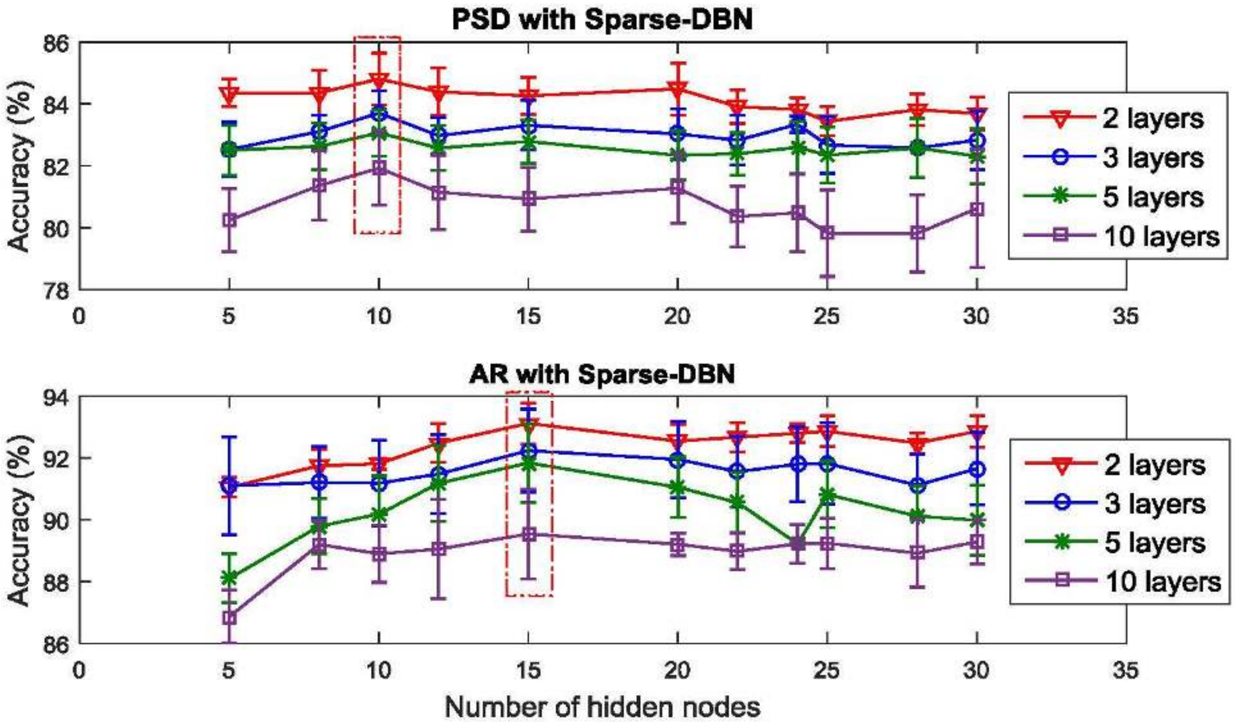

FIGURE 5 | Plot of the optimal number hidden nodes and layers

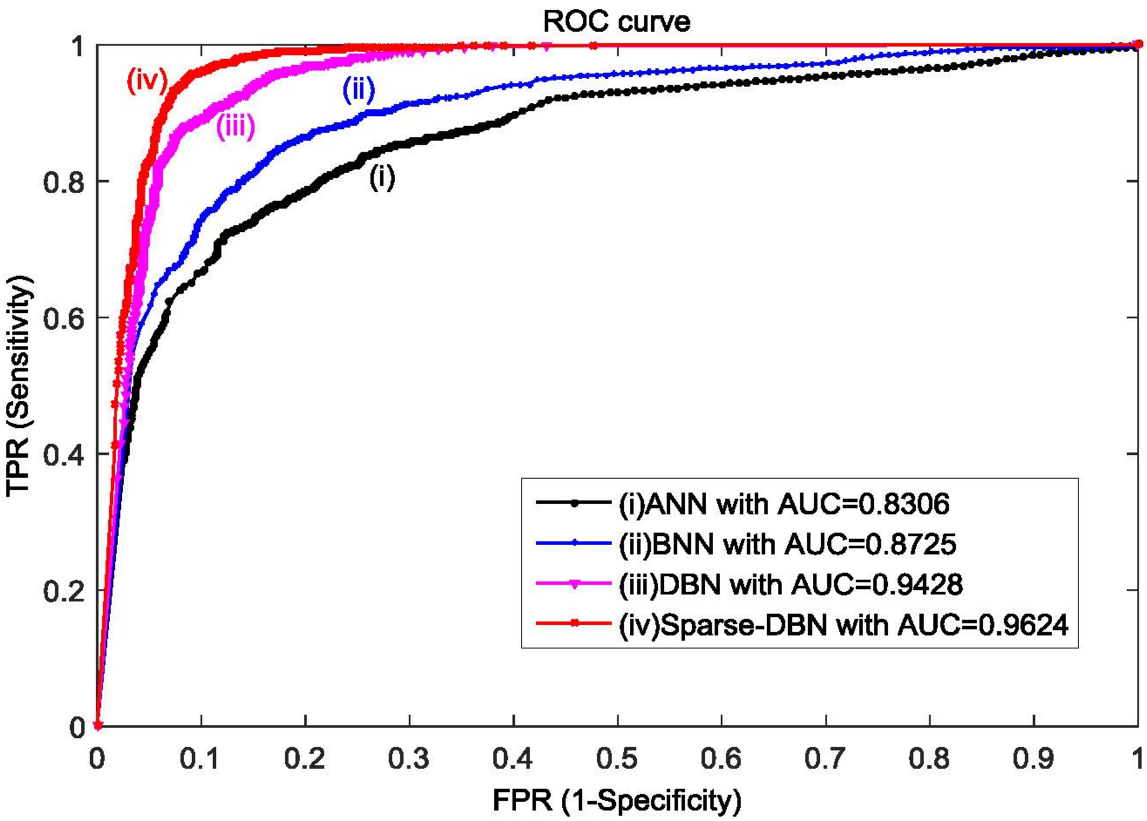

FIGURE 6 | ROC plot with AUC values for AR feature extractor and ANN, BNN, DBN and sparse-DBN classifiers of early stopping (hold-out cross-validation) technique.

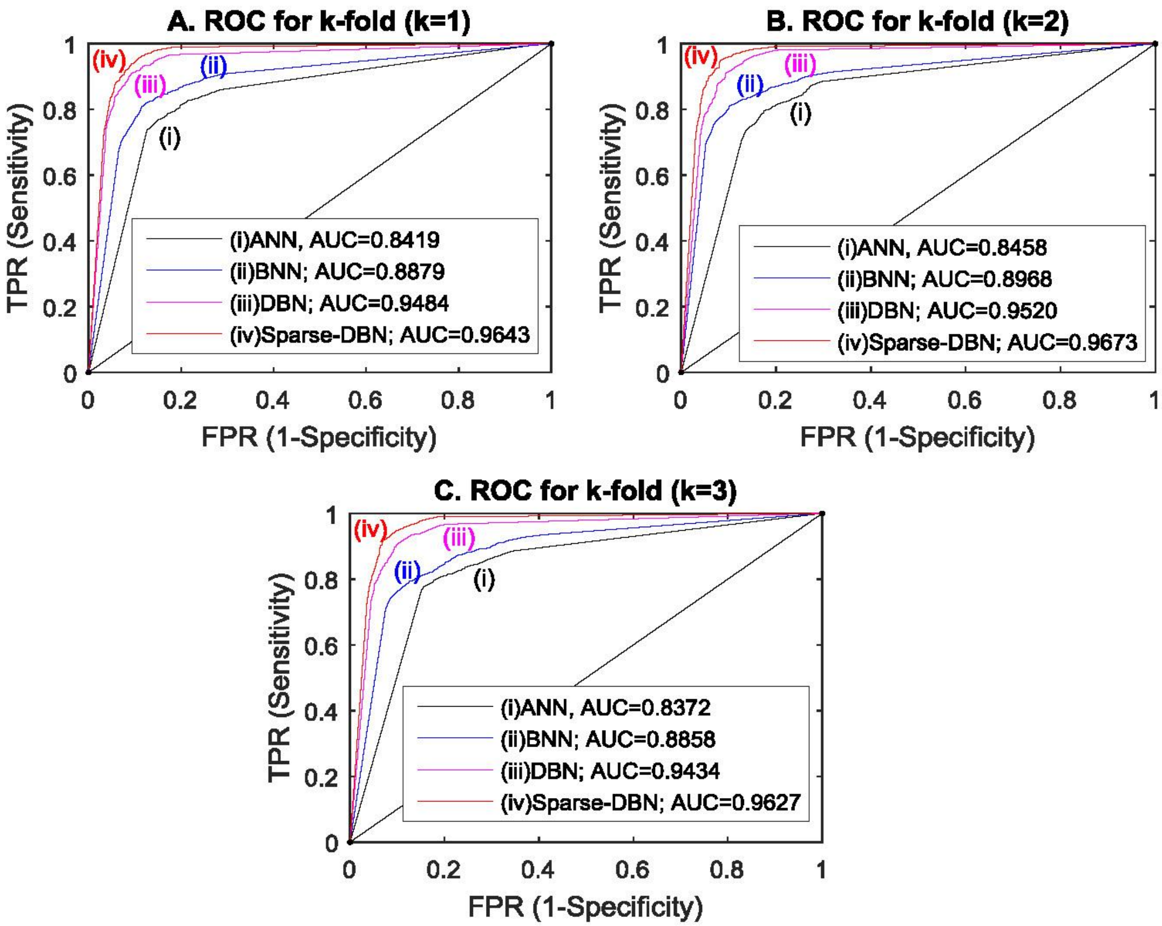

FIGURE 7 | ROC plot with AUC values for AR feature extractor and ANN, BNN, DBN and sparse-DBN classifiers of *k*-fold cross validation (*k*=3) technique.