Single-Trial EEG Classification with EEGNet and Neural Structured Learning for Improving BCI Performance

Haider Raza*, Anirban Chowdhury*, Saugat Bhattacharyya†, Spyros Samothrakis*

*School of Computer Science and Electronics Engineering, University of Essex, Colchester, England, UK

{h.raza, a.chowdhury, ssamot}@essex.ac.uk

†School of Computing, Eng & Intel. Sys, Ulster University, Derry Londonderry, Northern Ireland, UK

{s.bhattacharyya}@ulster.ac.uk

Abstract—Research and development of new machine learning techniques to augment the performance of Brain-computer Interfaces (BCI) have always been an open area of interest among researchers. The need to develop robust and generalised classifiers has been one of the vital requirements in BCI for real-world application. *EEGNet* is a compact CNN model that had been reported to be generalised for different BCI paradigms. In this paper, we have aimed at further improving the EEGNet architecture by employing Neural Structured Learning (NSL) that taps into the relational information within the data to regularise the training of the neural network. This would allow the EEGNet to make better predictions while maintaining the structural similarity of the input. In addition to better performance, the combination of EEGNet and NSL is more robust, works well with smaller training samples and requires on separate feature engineering, thus saving the computational cost. The proposed approach had been tested on two standard motor imagery datasets: the first being a two-class motor imagery dataset from Graz University and the second is the 4-class Dataset 2a from BCI competition 2008. The accuracy has shown that our combined EEGNet an NSL approach is superior to the sole EEGNet model.

I. INTRODUCTION

Advancements in the field of machine learning has led to wide improvement in Brain-computer Interface (BCI) technologies [1]. BCI aims at providing a direct communication pathway between the human brain and control assistive applications, such as stroke rehabilitation [2], robot and wheelchair control [3], [4], gaming and enhancing user experience [5], [6] and Military [7]. BCI technologies follows the principle that the intent of any action and its subsequent planning originates from the brain, which can be extracted, decoded and analyzed using advanced signal processing, machine learning and statistical algorithms [8]–[10].

Various invasive and non-invasive recording modalities exists to record the neural signals. Of these, Eletroencephalography (EEG) is the most commonly used recording device among BCI researchers. EEG being a non-linear and non-stationary signal requires further processing to distinguish between brain patterns corresponding to different tasks [11], [12]. Hence, EEG signals undergo various forms of filtering

(spatial and temporal) [8], feature extraction, and selection [1] before being used as inputs to a classifier [13].

Traditionally, standard machine learning algorithms such as Linear Discriminant Analysis, Support Vector Machines [13] were frequently used to decode mental states by BCI [14]. The main challenges faced by these classifiers were the low signal-to-noise ratio [15], high variance within and across users and within and across experimental sessions, small and limited amount of training data [16]. Therefore, in recent years, extensive research was undertaken to address these challenges. Emphasis on designing adaptive classifiers [17], [18] and transfer learning techniques [19], [20] were widely adopted to deal with the non-stationarity of the EEG signal and limited training data [14]. Methods involving Riemannian geometry [21], [22] and tensors [23], [24] were also adopted to compensate for the low signal-to-noise ratio of EEG signals. Such techniques narrowed down the feature extraction, feature selection and classification to a single step to achieve high performance and reliability.

In recent years, deep learning (DL) has achieved state-of-the-art performance in various fields of research including image, video and speech classification [25], [26]. The success of DL has motivated researchers to apply them for EEG classification. Cecotti et al [27] was the first BCI group to apply DL in a P300-based BCI. In recent years, several studies had implemented DL for motor imagery (MI) EEG classification. Yang et al. [28] employed convolutional neural network (CNN) to classify among left hand, right hand, foot and tongue labels. An end to end-based CNN model were successful in decoding between two MI signal (right hand and feet), mental rotation and word generation [29]. A Generative DL technique using deep belief network and stacked sparse auto-encoder was employed to deal with noisy MI EEG signals by including minimal objective-free pre-processing [30]. To adapt to the non-stationary changes in the EEG, last year, we have applied unsupervised domain adaptation propagating the negative of generative-adversarial loss [31] for domain adaptation in EEG-based BCI system.

A CNN architecture named *EEGNet* [32] was reported to be generalised across different BCI paradigms including sensori-
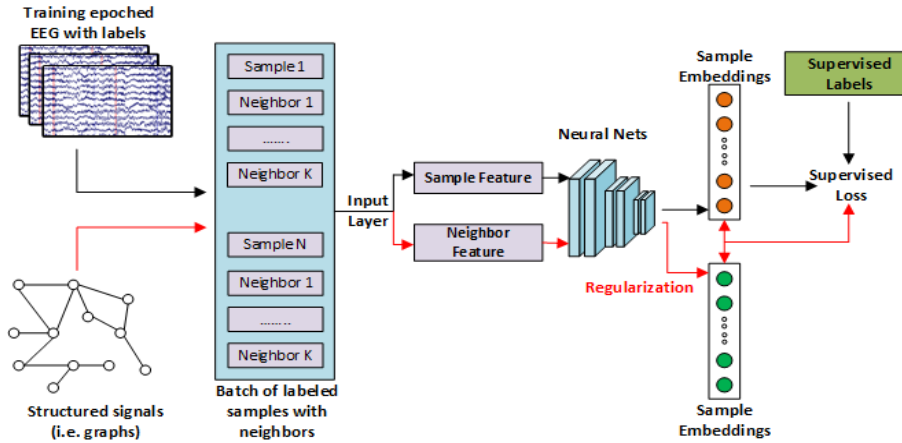
Fig. 1: Neural structured learning block diagram.

motor rthythm (SMR). EEGNet-8,2 had similar performance to the state-of-the-art filter bank Common Spatial Pattern method [33] for within-subject classification of MI data. EEGNet does not requires any form of feature engineering (i.e. feature extraction and feature selection) and can directly learn from the signal data itself. This is a clear advantage over other existing architecture. In this paper, we will have proposed a new classification approach that combines EEGNet architecture with neural structured learning (NSL) [34]. NSL trains neural networks includes structured signals with the signal features as inputs. Structured signals represent the similarity between samples which in turn improves the capability of the neural network to deal with small data while also improving its performance. The proposed approach is capable of performing both binary and multi-class labels. We evaluate on two real-world datasets corresponding to MI detection one for two-class dataset [35] and other is four-class dataset [36]. The method produces state of the art results.

The rest of the paper is organized as follows. Section II introduces Neural Structured Learning and gives a brief description of the EEGNet architecture. This section also briefly describes the standard datasets used in this paper. The results of the binary and multi-class datasets are reported in section III. Section IV discusses the advantages and disadvantages of our proposed approach while also providing insight on the future direction of this research.

II. METHODS

A. Neural Structured Learning

Neural Structured Learning (NSL) is a novel learning paradigm made available as part of Tensorflow in 2019 to train neural networks by borrowing structured signals (when available) in addition to feature inputs [34]. The borrowed structured signals are used to penalize the training on the neural network through regularization, while at the same time maintaining the input structural similarity by minimizing the neighbour loss. The thinking behind NSL can be applied to different types of neural networks that includes feed-forward

NN, convolution NN, recurrent NN, or a combination of all. Notably, the generalized neighbour loss is adjustable and can be selected based on the task in hand. The NSL can be generalised into two categories: 1) *Neural Graph Learning*: if the neighbours are explicitly represented by a graph; 2) *Adversarial Learning*: if neighbours are implicitly persuaded by adversarial perturbation.

The overall workflow for NSL is illustrated in Fig. 1. The black arrow represents the labelled EEG data as the traditional workflow and red arrow indicates the newly introduced workflow in NSL framework to leverage structured signals. First, the training samples (i.e. the epoched EEG data) are augmented to include structured signals. Next, the embeddings are prepared by feeding the augmented training samples (including both labelled samples and their corresponding neighbours) to the neural network (i.e base model). In our case, the base model is an EEGNet architecture (cf II.B).

To add the induced neighbour-based regularization to final loss, we have calculated the neighbour loss as the distance between the output prediction of the induced adversarial neighbour and the ground truth label. This NSL model will reuse the layers and variables as defined in the base model (i.e. EEGNet). Thus, training the NSL model also updates the variables in the base model. The regularization term will be added into training objective and will be minimized during training together with neighbour loss and categorical cross-entropy loss.

B. EEGNet Architecture

EEGNet[1] is a recently realised CNN architecture, which can be used to different BCI paradigms such as sensory-motor rhythm (SMR), P300 visual evoked potentials, error-related negativity responses (ERN), and movement-related cortical potentials (MRCP). In this architecture, two convolutional process were combined (i.e. depth-wise and separable convolutions) to build an EEG-specific network. This network is capable of summarizing a few well-known EEG feature

---

[1]https://github.com/vlawhern/arl-eegmodels

extraction methods such as optimal spatial filter bank. One of it's advantages of this architecture is that it is trainable on a finite amount of data acquired during the training phase and capable of producing separable features of the decoder. We have slightly modified the EEGNet model-based on the requirements for the multi-class MI-related EEG dataset.

The architecture of the model is comprising of three blocks and are defined as follows:

1) Block 1: Combination of Conv2D and DepthwiseConv2D: This block starts with an input layer there has two convolution steps: first step comprises of a 2D convolution filter while the second one has a depth-wise convolution. Both steps undergo batch normalisation at the end. Depth-wise convolution reduces the number of trainable parameters to fit a deep predictive model. Notably, depth-wise convolution is not fully-connected to all previous feature maps, which makes lesser parameters to fit. For EEG, combined Conv2D and depthwiseConv2D provides a direct way to learn spatial filters from each temporal filter. A depth parameter controls the number of spatial filters that are learnt from each feature map. This combination is inspired by the filter-bank common spatial pattern (FBCSP) algorithm, where the spatio-temporal features were prepared during the learning process.

2) Block 2: Separable convolution: After receiving inputs from block 1, a depth-wise convolution was used followed by point-wise convolution. There are two main advantages of using separable convolutions. Firstly, it reduces the number of parameters to fit and secondly, principally separating the relationship with and across feature map by learning a kernel and summarising each feature map individually by optimally merging the output. In other words, this method separates learning on how to summarise individual feature maps in time using depth-wise convolution and learns how to optimally combine feature maps using point-wise convolution. This method represents different feature maps at different time-scales and combines the output afterwards.

3) Block 3: Classification: Finally, the retrieved features are then passed to softmax function. Softmax function is used here because EEGNet is a multi-class classification model but it is also possible to do binary classification using sigmoid function [37].

In this study, we combined the above-mentioned NSL framework and EEGNet as a base model in NSL framework and named it as NSL-EEGNet.

C. Datasets

In this study, we have our tested our algorithm in two standard BCI datasets: one having four-classes while the other having two-classes.

1) Two-class MI Dataset: This dataset [35] comprised of EEG recording from thirteen participants (with no known medical or neurological disease) between the age of 20 and 30 years. Eight of these participants were naive to BCI experiments. The cue guided Graz-BCI training paradigm [38] was employed to undertake sustained (5 seconds) kinaesthetic MI of the right hand (class 1) and of the feet (class 2). Each

session comprised of eight runs of which five were used for training and three for validation via feedback. The participants received the feedback in the form of a white coloured bar. Each participant undertook 100 trials during training run and 60 trials during the test run. The neural signals in the form of EEG was recorded at sampling rate of 512 Hz. Data were collected from fifteen EEG channels located in the vicinity of C3, Cz and C4 (referenced on the left mastoid and the grounded on the right mastoid).

2) Four-class MI Dataset: We have used dataset 2a from BCI competition IV [36] which comprises of four-classes MI: left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). The dataset is comprised of EEG data recorded from nine participants (A01-A09) from two experimental sessions over two separate days. Each session is made of 6 runs with short breaks in between. Each run consists of 48 trials (12 for each class). Hence, each participant across each session yields a total of 288 trials. Data were collected from twenty-two EEG channels located in the frontal, central and parietal region of the brain and three monopolar EOG at a sampling rate of 250 Hz. The signals were bandpass filtered between 0.5 to 100 Hz along with a notch filter of 50 Hz to remove line noise.

D. Parameters setting

A description of the setting parameters is given as follows: a) EEG data is stored in a 3-dimensional format (N, C, T), where N is the number of trials, C is the number of channels, and T is the time samples). b) The EEG data were band-pass filtered from 8 Hz as a lower cut-off to a variable limit as a parameter for upper cut-off with the values 24, 30, and 40 Hz. c) Model parameters: In block 1, the size of the 2D convolution filter is $(1, flt\_size)$ (i.e., the filter length is equal to half of the sampling rate) because it captures the information from 2 Hz and above. In our study, we have evaluated the results on three different kernel lengths (flt_size = [32, 64,128]) (i.e. temporal filter) for two-class MI dataset and (flt_size = [15, 30,60]) (i.e. temporal filter) for dataset 2a; depth-wise convolution of size (C,1), where C is number of channels (i.e. C = 16 for two-class dataset and C = 22 for dataset 2a) to learn the spatial filter with depth parameter D controls the number of spatial filters to learn for each feature map. In block 2, separable convolution is used (i.e. depth-wise convolution) of size (1,16). d) Fitting parameters: the models was fitted using 'adam' optimizer and minimized the 'categorical_crossentropy' function with number of epochs (i.e. epochs = [100, 300, 500]). The codes were executed on Google Colab, where Tesla K80 GPU is freely available. Tensorflow [39] and Keras API [40] were used to create the build the learning model. In two-class dataset: Session I data has 288 trials, where 240 trials were used for training and 48 trials were used for validation and Session II data were used for evaluation. In two-class dataset: Session I data has 100 trials, where 80 trials were used for training and 20 trials were used for validation and whole Session II data were used for evaluation, which has 60 trials. In four-class dataset 2a:
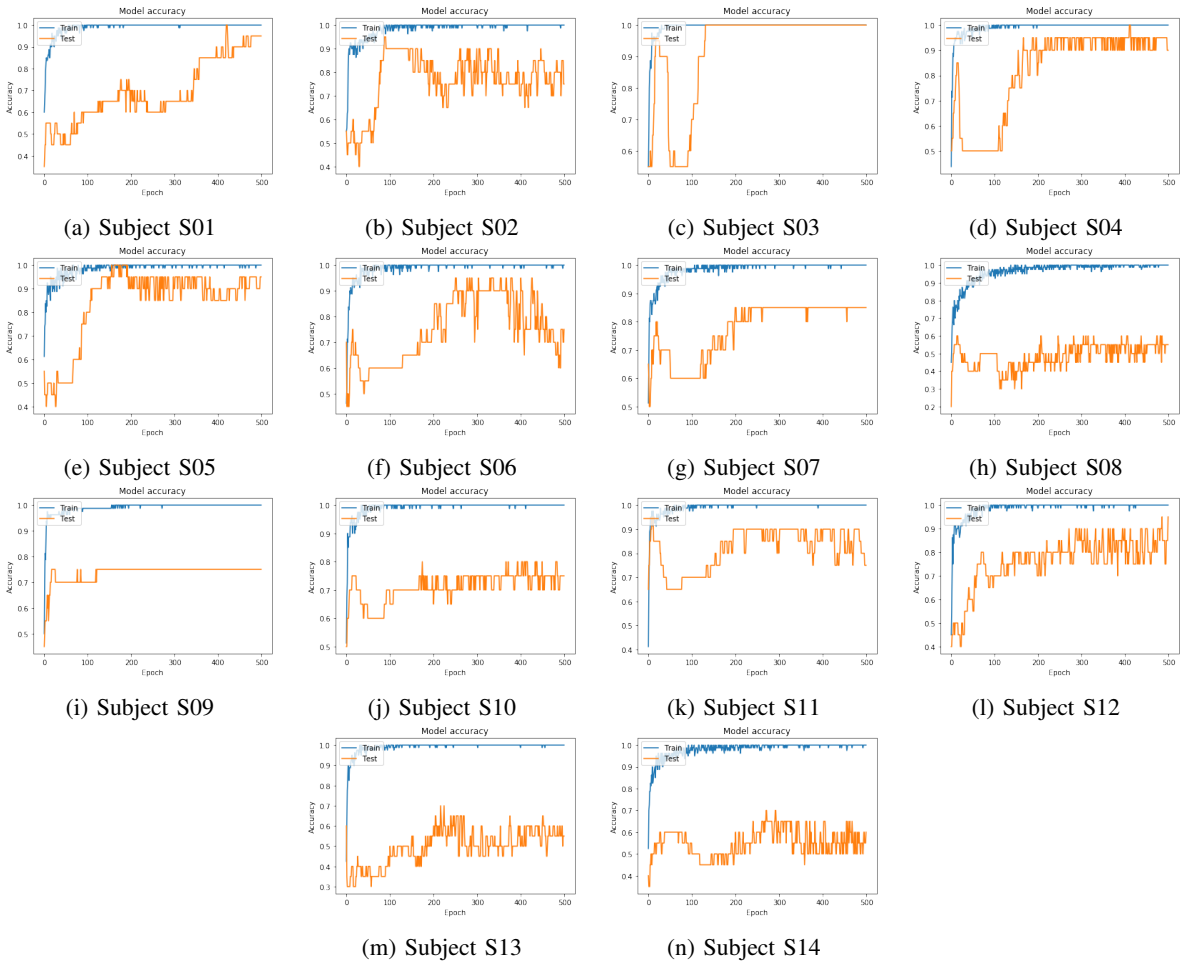
Fig. 2: Training and validation accuracy of NSL-EEGNet on two-class MI dataset with 500 epochs, where the hyper-parameters are: Frequency: [8-30] Hz, dropout: 0.25, Kernel Length = 64.

Session I data has 288 trials, where 240 trials were used for training and 48 trials were used for validation and whole Session II data were used for evaluation, which has 288 trials.

III. RESULTS

A. Two-class MI Dataset

Fig 2 illustrates the training and validation performance with 500 epochs, where the hyper-parameters are: Frequency: [8-30] Hz, dropout: 0.25, Kernel Length = 64. Table I compares the performance of EEGNet with NSL-EEGNet for two-class MI dataset. The EEGNet was trained for a different number of epochs during the training and interestingly, the testing classification accuracy gradually improved with an increase in the number of training epochs. We performed grid search method and obtained a set of best parameters (i.e. freq [8-30], dropout = 0.25, and flt_size (i.e. kernel length = 64)). The average test classification accuracy for EEGNet under different number of training epochs is given as follows: 1) 100 epochs of training: test accuracy 72.14±17.93%; 2) 300 epochs of training: test accuracy 71.19±16.63%; and 3) 500 epochs of training: test accuracy 71.07±17.30%. The performance of

NSL-EEGNet is superior to EEGNet after training for more than 100 epochs. We performed grid search for NSL-EEGNet and found the same set of best parameter (i.e. freq [8-30], dropout = 0.25, and flt_size (i.e. kernel length = 64)). The average test classification accuracy for NSL-EEGNet under different number of training epochs is given as follows: 1) 100 epochs of training: test accuracy 65.95±17.45%, which less than in comparison to EEGNet; 2) 300 epochs of training: test accuracy 75.71±17.92%, which is 4% better than EEGNet; and 3) 500 epochs of training: test accuracy 76.07±17.92%, which is 5% better than EEGNet. Interestingly, the performance of NSL-EEGNet model increased with the increasing the number of training epochs. We have evaluated the both EEGNet and NSL-EEGNet models with a set of parameters and to summaries the average classification accuracy, heatmaps are illustrated in Fig. 4 for EEGNet and Fig. 5 for NSL-EEGNet under the different number of training epochs, where the x-axis is for flt_size (i.e. kernel length) and the y-axis is for dropout. Fig 4.(a-c) and Fig 5.(a-c) shows the average test accuracy with 100, 300 and 500 epochs of training, respectively.

(a) Subject A01 (b) Subject A02 (c) Subject A03 (d) Subject A04

(e) Subject A05 (f) Subject A06 (g) Subject A07 (h) Subject A08
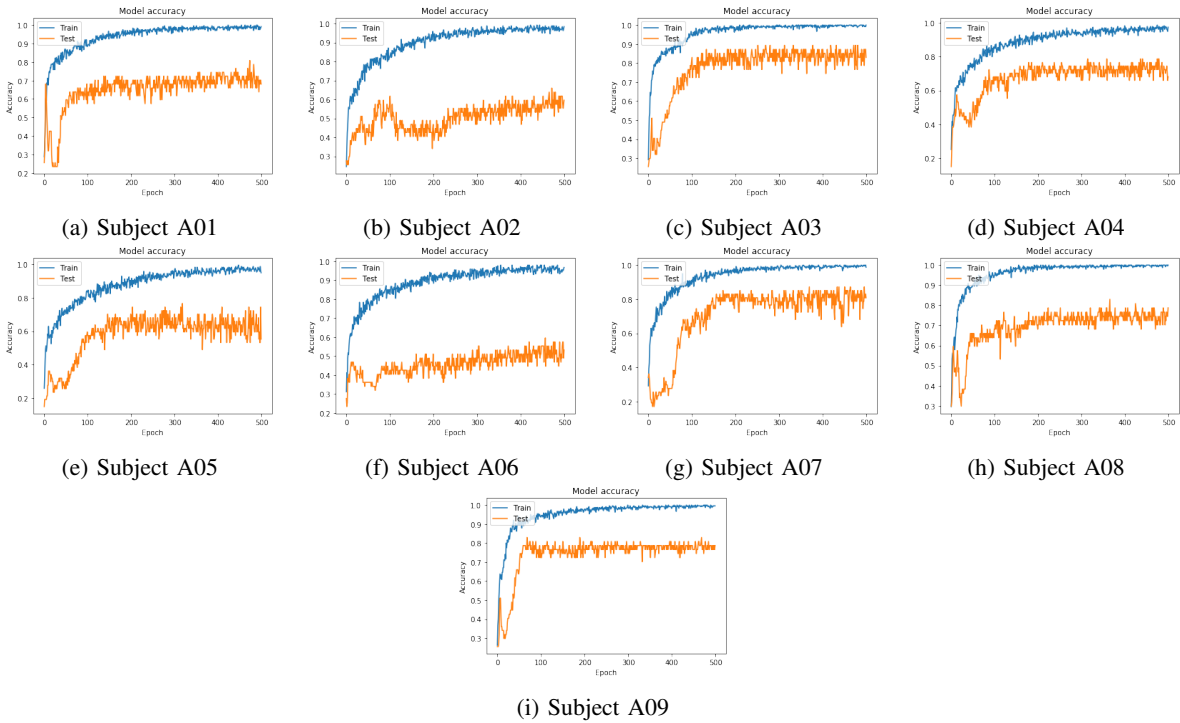
(i) Subject A09

Fig. 3: Training and validation accuracy of NSL-EEGNet on four-class dataset 2a with 500 epochs, where the hyper-parameters are: Frequency: [8-30] Hz, dropout: 0.25, Kernel Length = 64.

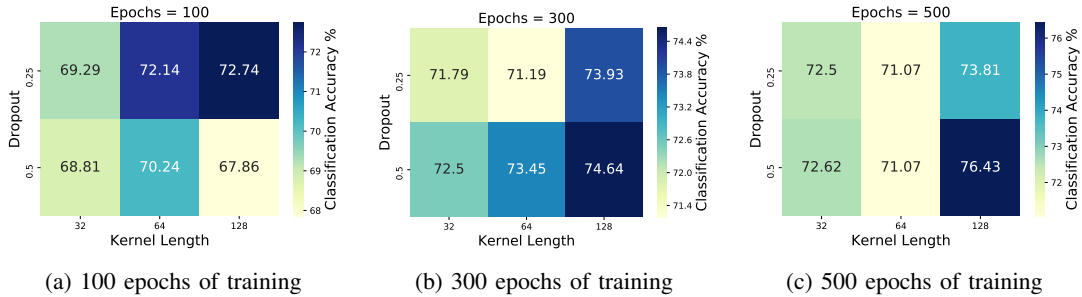(a) 100 epochs of training (b) 300 epochs of training (c) 500 epochs of training

Fig. 4: Two-class MI dataset comparison of the average classification accuracy for EEGNet model under different parameters: epochs = [100, 300, 500]; dropout = [025, 0.5], and kernel length = [32, 64, 128].

B. Four-class MI Dataset 2a

Fig 3 illustrates the training and validation performance with 500 epochs, where the hyper-parameters are: Frequency: [8-30] Hz, dropout: 0.25, Kernel Length = 64. Table II compares the performance of EEGNet with EEGNet plus NSL for four-class MI dataset. The EEGNet was trained for a different number of epochs during the training and interestingly, the testing classification accuracy gradually improved with an increase in the number of training epochs. We performed grid search method and obtained a set of best parameters (i.e. freq [8-30], dropout = 0.25, and flt_size (i.e. kernel length = 30)). The average test classification accuracy for EEGNet under different number of training epochs is given as follows: 1) 100 epochs of training: test accuracy $65.74\pm16.22\%$; 2) 300 epochs of training: test accuracy $66.98\pm15.52\%$; and 3)

500 epochs of training: test accuracy $68.98\pm14.14\%$. Similar to the binary classification results, the performance of NSL-EEGNet are superior to EEGNet only after training with more than 100 epochs. We performed grid search for NSL-EEGNet and found the same set of best parameter (i.e. freq [8-30], dropout = 0.25, and flt_size (i.e. kernel length = 30)). The average test classification accuracy for NSL-EEGNet under different number of training epochs is given as follows: 1) 100 epochs of training: test accuracy $64.27\pm15.55\%$, which is slightly less than in comparison to EEGNet; 2) 300 epochs of training: test accuracy $69.21\pm16.21\%$, which is 2.23% higher than EEGNet; and 3) 500 epochs of training: test accuracy $70.68\pm12.84\%$, which is 1.7% better than EEGNet. Interestingly, the performance of NSL-EEGNet model increased with the increasing number of training epochs. We have evaluated the both EEGNet and NSL-EEGNet models with a

TABLE I: Test classification Accuracy for two-class MI data for two methods EEGNet and NSL-EEGNet with the following parameter: Frequency: [8-30] Hz, dropout: 0.25, Kernel Length = 64

Subject	EEGNet			NSL-EEGNet		
Subject	Test Acc (%)	Test Acc (%)	Test Acc (%)	Test Acc (%)	Test Acc (%)	Test Acc (%)
Epochs	100	300	500	100	300	500
S01	51.67	53.33	66.67	68.33	**98.33**	96.67
S02	**83.33**	76.67	76.67	68.33	78.33	81.67
S03	**100.00**	**100.00**	**100.00**	98.33	**100.00**	**100.00**
S04	86.67	90.00	**91.67**	50.00	86.67	86.67
S05	63.33	61.67	63.33	51.67	65.00	**75.00**
S06	**83.33**	70.00	68.33	81.67	68.33	75.00
S07	81.67	81.67	**90.00**	75.00	**90.00**	86.67
S08	41.67	**50.00**	45.00	45.00	45.00	48.33
S09	98.33	96.67	96.67	98.33	98.33	**100.00**
S10	58.33	61.67	65.00	50.00	65.00	**66.67**
S11	81.67	81.67	61.67	70.00	**83.33**	**83.33**
S12	58.33	55.00	58.33	53.33	**73.33**	63.33
S13	**63.33**	61.67	61.67	60.00	51.67	50.00
S14	**58.33**	56.67	50.00	53.33	56.67	51.67
Mean	72.14	71.19	71.07	65.95	75.71	**76.07**
Std	17.93	16.63	17.30	17.45	17.92	17.92

TABLE II: Test classification Accuracy for BCI competition IV dataset 2a for two methods EEGNet and NSL-EEGNet with the following parameter: Frequency: [8-30] Hz, dropout: 0.25, Kernel Length = 30

Subject	EEGNet			NSL-EEGNet		
Subject	Test Acc (%)	Test Acc (%)	Test Acc (%)	Test Acc (%)	Test Acc (%)	Test Acc (%)
Epochs	100	300	500	100	300	500
A01	71.53	78.82	78.47	68.75	81.25	**82.29**
A02	43.75	48.61	**58.68**	45.14	53.47	51.39
A03	82.99	**86.81**	84.72	79.86	85.76	85.07
A04	61.81	59.03	62.85	61.46	62.15	**67.01**
A05	45.14	46.18	47.22	43.75	45.49	**58.33**
A06	48.61	51.74	52.43	46.53	51.04	**56.25**
A07	78.47	74.31	**84.38**	77.08	81.25	83.33
A08	**78.13**	77.43	71.88	74.65	**78.13**	73.96
A09	81.25	79.86	80.21	81.25	**84.38**	78.47
Mean	65.74	66.98	68.98	64.27	69.21	**70.68**
Std	16.22	15.52	14.14	15.55	16.06	12.84

set of parameters and to summaries the average classification accuracy, heatmaps are illustrated in Fig. 6 for EEGNet and Fig. 7 for NSL-EEGNet under the different number of training epochs, where the x-axis is for flt_size (i.e. kernel length) and the y-axis is for dropout. Fig 6.(a-c) and Fig 7.(a-c) shows the average test accuracy with 100, 300 and 500 epochs of training, respectively.

IV. DISCUSSION

EEGNet uses both depth-wise and separable convolutions which can automatically extract characteristic signature features from raw EEG signal. Results from the original EEGNet paper [32] has shown that it has achieved higher performance across different BCI paradigms and have vindicated the obtained superior performance using feature analysis technique

that their architecture were not influence by artifact in the data. NSL, on the other hand, uses Neural Graph Learning to train neural networks by employing graphs and structured data. Rich relational information are expected to be found among structured data, which in turn could aid a machine learning model to gain higher model accuracy. They aim at regularising the training of neural networks by helping the based model to make better predictions (adversarial learning) while maintaining its structural integrity. This, in turn, makes the model more robust. In this paper, we have combined both EEGNet and NSL to develop a novel DL approach to improve the performance of MI BCI. Both EEGNet and NSL has a combined advantage of being robust models with the ability to work with small training datasets. The generalised structural

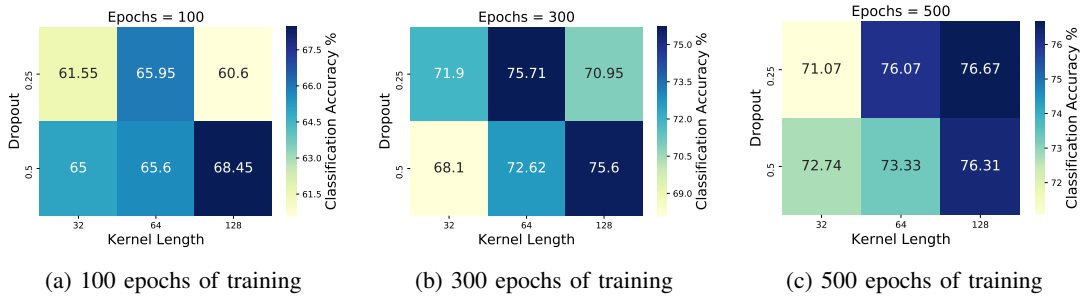| (a) 100 epochs of training | (b) 300 epochs of training | (c) 500 epochs of training |

Fig. 5: Two-class MI dataset comparison of the average classification accuracy for NSL-EEGNet model under different parameters: epochs = [100, 300, 500]; dropout = [025, 0.5], and kernel length = [32, 64, 128].

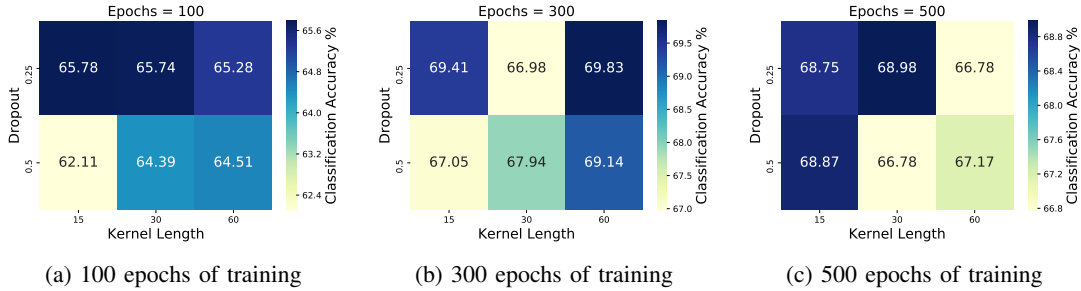| (a) 100 epochs of training | (b) 300 epochs of training | (c) 500 epochs of training |

Fig. 6: Dataset 2a comparison of the average classification accuracy for EEGNet model under different parameters: epochs = [100, 300, 500]; dropout = [025, 0.5], and kernel length = [15, 30, 60]

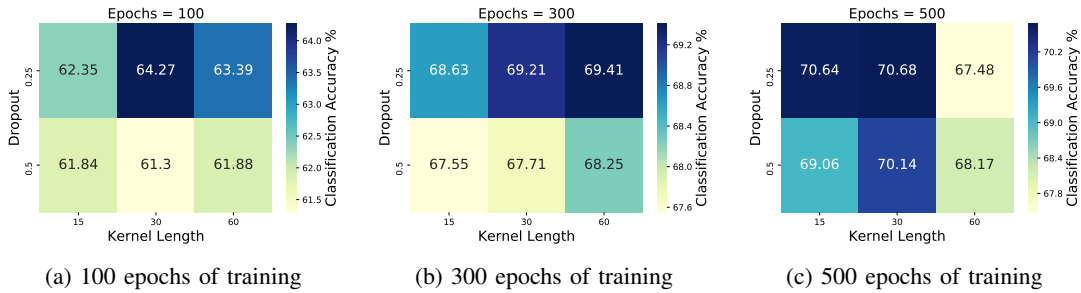| (a) 100 epochs of training | (b) 300 epochs of training | (c) 500 epochs of training |

Fig. 7: EEGNet and NSL for Dataset 2a comparison of the average classification accuracy for NSL-EEGNet model under different parameters: epochs = [100, 300, 500]; dropout = [025, 0.5], and kernel length = [15, 30, 60]

information is further expected to augment the model accuracy. To validate our claim, we have tested our proposed approach on two standard MI datasets: a two-class MI dataset from Graz University [35] and a four-class MI BCI competition dataset [36]. The datasets encompasses both binary and multi-class problems and hence, aims at testing the performance of our proposed approach for both cases. The results from Table I and Table II shows that the combined NSL-EEGNet approach shows improvement over the base model EEGNet performance but mostly for epochs of size greater than 100 at a kernel length of 30. The mean best accuracy of 76.07% obtained our model for binary classification is comparable to the original best average performance of 77.15% (as reported in [35]). Though it is to be noted that in the original paper the EEG had undergone feature engineering through the use of common spatial patterns while in our case the EEG itself was used as a feature. With regards to the multi-class performance, current state-of-the-art has higher reported accuracy of 83% by Zhang et al. [41] and 75.77% by Amin et al. [42] as compared to our best accuracy of 70.68% on the same dataset. This being the first use of neural structured learning for EEG classification it is expected that with further parameter tuning and better implementation of structured learning that complements well with the base model, the model accuracy can be further be improved. If successful, this approach has the potential to augment the performance of BCI for real world applications, especially in neuro-rehabilitation by tapping into the rich structural information embedded within the EEG samples.

REFERENCES

[1] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rako-tomamonjy, and F. Yger, "A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, apr 2018.

[2] K. Ang and C. Guan, "Brain-computer interface in stroke rehabilitation," *Journal of Computer Science and Engineering*, vol. 7, pp. 139–146, 06 2013.

[3] S. Bhattacharyya, S. Shimoda, and M. Hayashibe, "A synergetic brain-machine interfacing paradigm for multi-dof robot control," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 7, pp. 957–968, July 2016.

[4] T. Kaufmann, A. Herweg, and A. Kübler, "Toward brain-computer interface based wheelchair control utilizing tactually-evoked event-related potentials," *Journal of neuroengineering and rehabilitation*, vol. 11, p. 7, 01 2014.

[5] T. Zander and C. Kothe, "Towards passive brain–computer interfaces: applying brain–computer interface technology to human-machine systems in general," *Journal of neural engineering*, vol. 8, p. 025005, 03 2011.

[6] D. Coyle, J. Principe, F. Lotte, and A. Nijholt, "Guest editorial: Brain/neuronal-computer game interfaces and interaction," *Computational Intelligence and AI in Games, IEEE Transactions on*, vol. 5, pp. 77–81, 06 2013.

[7] C. N. Munyon, "Neuroethics of non-primary brain computer interface: Focus on potential military applications," *Frontiers in Neuroscience*, vol. 12, p. 696, 2018.

[8] M. Clerc, *Electroencephalography Data Preprocessing*. John Wiley & Sons, Ltd, 2016, ch. 6, pp. 101–125.

[9] F. Lotte and M. Congedo, *EEG Feature Extraction*. John Wiley & Sons, Ltd, 2016, ch. 7, pp. 127–143.

[10] R. Flamary, A. Rakotomamonjy, and M. Sebag, *Statistical Learning for BCIs*. John Wiley & Sons, Ltd, 2016, ch. 9, pp. 185–205.

[11] H. Raza, D. Rathee, S.-M. Zhou, H. Cecotti, and G. Prasad, "Covariate shift estimation based adaptive ensemble learning for handling non-stationarity in motor imagery related eeg-based brain-computer interface," *Neurocomputing*, vol. 343, pp. 154–166, 2019.

[12] H. Raza, H. Cecotti, and G. Prasad, "Optimising frequency band selection with forward-addition and backward-elimination algorithms in eeg-based brain-computer interfaces," in *2015 international joint conference on neural networks (IJCNN)*. IEEE, 2015, pp. 1–7.

[13] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 4, no. 2, pp. R1–R13, jan 2007.

[14] H. Raza, H. Cecotti, and G. Prasad, "A combination of transductive and inductive learning for handling non-stationarities in motor imagery classification," in *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016, pp. 763–770.

[15] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, no. 6, pp. 767 – 791, 2002.

[16] M. Krauledat, M. Tangermann, B. Blankertz, and K.-R. Müller, "Reducing calibration time for brain-computer interfaces: A clustering approach." vol. 19, 01 2006, pp. 753–760.

[17] H. Raza, H. Cecotti, Y. Li, and G. Prasad, "Adaptive learning with covariate shift-detection for motor imagery-based brain–computer interface," *Soft Computing*, vol. 20, no. 8, pp. 3085–3096, 2016.

[18] A. Chowdhury, H. Raza, Y. K. Meena, A. Dutta, and G. Prasad, "Online covariate shift detection-based adaptive brain–computer interface to trigger hand exoskeleton feedback for neuro-rehabilitation," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 4, pp. 1070–1080, 2017.

[19] O. Özdenizci, Y. Wang, T. Koike-Akino, and D. Erdoğmuş, "Transfer learning in brain-computer interfaces with adversarial variational autoencoders," in *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, March 2019, pp. 207–210.

[20] A. M. Azab, L. Mihaylova, K. K. Ang, and M. Arvaneh, "Weighted transfer learning for improving motor imagery-based brain–computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 7, pp. 1352–1359, July 2019.

[21] M. Congedo, A. Barachant, and R. Bhatia, "Riemannian geometry for eeg-based brain-computer interfaces; a primer and a review," *Brain-Computer Interfaces*, vol. 4, pp. 1–20, 03 2017.

[22] F. Yger, M. Berar, and F. Lotte, "Riemannian approaches in brain-computer interfaces: A review," *IEEE Transactions on Neural System and Rehabilitation Engineering*, vol. PP, 10 2017.

[23] Y. Washizawa, H. Higashi, T. Rutkowski, T. Tanaka, and A. Cichocki, "Tensor based simultaneous feature extraction and sample weighting for eeg classification," 01 2010, pp. 26–33.

[24] Y. Zhang, G. Zhou, J. Jin, X. Wang, and A. Cichocki, "Frequency recognition in ssvep-based bci using multiset canonical correlation analysis," *International journal of neural systems*, vol. 24, p. 1450013, 06 2014.

[25] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.

[26] M. S. Hossain and G. Muhammad, "Emotion recognition using deep learning approach from audio–visual emotional big data," *Information Fusion*, vol. 49, pp. 69 – 78, 2019.

[27] H. Cecotti and A. Graser, "Convolutional neural networks for p300 detection with application to brain-computer interfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 433–445, March 2011.

[28] H. Yang, S. Sakhavi, K. K. Ang, and C. Guan, "On the use of convolutional neural networks and augmented csp features for multi-class motor imagery of eeg signals classification," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug 2015, pp. 2620–2623.

[29] R. Schirrmeister, J. Springenberg, L. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for eeg decoding and visualization: Convolutional neural networks in eeg analysis," *Human Brain Mapping*, vol. 38, 08 2017.

[30] A. Hassanpour, M. Moradikia, H. Adeli, R. Khayami, and P. Shamsinjead Babaki, "A novel end-to-end deep learning scheme for classifying multi-class motor imagery electroencephalography signals," *Expert Systems*, 12 2019.

[31] H. Raza and S. Samothrakis, "Bagging adversarial neural networks for domain adaptation in non-stationary eeg," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–7.

[32] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces," vol. 15, no. 5, p. 056013, jul 2018.

[33] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, and Cuntai Guan, "Filter bank common spatial pattern (fbcsp) in brain-computer interface," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, June 2008, pp. 2390–2397.

[34] T. D. Bui, S. Ravi, and V. Ramavajjala, "Neural graph learning: Training neural networks using graphs," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, ser. WSDM '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 64–71.

[35] C. Vidaurre, A. Schlögl, R. Cabeza, R. Scherer, and G. Pfurtscheller, "Study of on-line adaptive discriminant analysis for eeg-based brain computer interfaces," *IEEE transactions on bio-medical engineering*, vol. 54, pp. 550–6, 04 2007.

[36] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. Miller, G. Mueller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert, and B. Blankertz, "Review of the bci competition iv," *Frontiers in Neuroscience*, vol. 6, p. 55, 2012.

[37] G. Bouchard, "Efficient bounds for the softmax function and applications to approximate inference in hybrid models," in *NIPS 2007 workshop for approximate Bayesian inference in continuous/hybrid systems*, 2007.

[38] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proceedings of the IEEE*, vol. 89, no. 7, pp. 1123–1134, July 2001.

[39] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[40] F. Chollet *et al.*, "keras. github (2015)," 2015.

[41] R. Zhang, Q. Zong, L. Dou, and X. Zhao, "A novel hybrid deep learning scheme for four-class motor imagery classification," *Journal of Neural Engineering*, vol. 16, no. 6, p. 066004, oct 2019.

[42] "Deep learning for eeg motor imagery classification based on multi-layer cnns feature fusion," *Future Generation Computer Systems*, vol. 101, pp. 542 – 554, 2019.