Hierarchy of Hybrid Deep Neural Networks for Physical Action Classification by Brain-Computer Interface

Kostiantyn Kostiukevych[1,*,†], Yuri Gordienko[1], Nikita Gordienko[1], Oleksandr Rokovyi[1] and Sergii Stirenko[1]

[1]National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37 Peremohy aveniu, 03056, Kyiv, Ukraine

Abstract

In the fields of bio-monitoring, prosthetic devices, human-computer interaction the use of artificial intelligence methods is usually an integral part. Grasp-and-lift(GAL) has become a popular dataset for the testing of deep learning (DL) models for motor imaginary EEG classification task. In this article various combinations of deep neural network (DNN), such as one and two dimensional convolution layers, separable convolution, time distributed wrapper, recurrent neural networks (RNNs), bidirectional recurrent neural networks (BiRNN), attention based mechanism, additional hidden states, were investigated. Macro-AUC and number of parameters was chosen as a metric of feasibility of models. The hierarchies of the different RNN models development were built. The results showed that using RNN layer with hidden states as input for last fully-connected layer decreased performance, but addition of attention mechanism after output with hidden states allows to solve this problem. Also applying BiRNN with CNN as first layers improves overall macro AUC and reduce number of parameters.

Keywords

Grasp-and-lift, EEG, deep learning, classification, deep learning hybrids

1. Introduction

According to the study, there are nearly 1 in 50 people living with paralysis – approximately 5.4 million people only in United States[1]. Providing to this people ability to interact with computer reduces their suffering and extends their capabilities. There are two types of brain computer interface (BCI) based on the electrodes used for measuring the brain activity: non-invasive BCI where the electrodes are placed on the scalp (e.g., electroencephalography (EEG) based BCI), and invasive Brain computer interface where the electrodes are directly attached on human brain [e.g., BCI based on electrocorticography (ECoG), or intracranial electroencephalography (iEEG)][2]. The EEG employs non-invasive electrodes placed on participants' scalps to measure signals produced by local field potentials with active cortex neurons, having high temporal but

MoMLeT+DS 2022: 4th International Workshop on Modern Machine Learning Technologies and Data Science, November, 25-26, 2022, Leiden-Lviv, The Netherlands-Ukraine.

jjwpey@gmail.com (K. Kostiukevych); yuri.gordienko@gmail.com (Y. Gordienko); nik.gordiienko@gmail.com (N. Gordienko); rokovoy@comsys.kpi.ua (O. Rokovyi); sergii.stirenko@gmail.com (S. Stirenko)

0000-0001-7168-0064 (K. Kostiukevych); 0000-0003-2682-4668 (Y. Gordienko); 0000-0002-6922-4307 (N. Gordienko); 0000-0001-6934-7502 (O. Rokovyi); 0000-0002-9395-8685 (S. Stirenko)

low spatial precision. Basic task in EEG-based BCI is decoding hand movements in order to classify certain movement in moment or classify intention for certain movement.

In order to examine how different combinations of deep neural network (DNN) components are suited for movement classification task, we are interested in identifying and classifying various modifications of DNNs with investigation of their impact on the performance metrics.

The structure of this paper is as follows. Section 2 contains description of the state of the art, section 3 describes dataset, models, and the whole workflow, section 4 contains results and discussions of the results and section 5 resumes the results obtained.

2. Background and Related Works

The different types of DNNs have been used in EEG-research in medical, educational, operational, and other applications [3, 4, 5]. For example, EEGNet DNN, a compact convolutional neural network (CNN), has been developed for EEG-based BCIs [6]. There is similar research which has comparison of different recurrent neural network (RNN) architectures on Grasp-and-lift (GAL) dataset[7]. They observed that dropout regularization improved performance of RNN by average of 4 percentage point. In addition, their findings confirms that smoothing the predictions with moving average helped making consistent predictions, eliminating abrupt and incongruous prediction errors [8]. Another work proposed Discrete Wavelet Transform as part of prepossessing which enhanced Area Under a receiver operating characteristic Curve (AUC) for 7.7 percentage points for CNN-based, and 9.7 and for Long Short Term Memory (LSTM) based networks and discovers that give same or better results but in a much faster, more computationally effective fashion [9].

As it was shown recently, reliable classification of GAL movements can be handled using simple CNNs even (with AUC>0.92 after 1 training epoch) [10]. In our previous work we use Noise Data Augmentation and Detrended Fluctuation Analysis to demonstrate that some physical actions in GAL dataset can be divided in separate groups of actions that can be characterized by complexity and the feasibility of their classification: the easiest (HandStart), medium (LiftOff, Replace, and BothReleased), and hardest (BothStartLoadPhase and FirstDigitTouch) classification [11, 12].

DNNs and their components were intensively researched for analysis of EEG signals in a various applications [3, 4, 5] like air traffic [13, 14, 15], health care [16, 17, 18], education [19, 20, 21], gaming and entertaining [22, 20, 23, 24], and other applications [5]. Different components of convolutional neural network (CNN) [6, 25, 18, 26, 11], recurrent neural networks (RNN) [27, 28, 29, 12], fully-connected networks (FCN) [11, 12], and other DNNs were imvestigated in them. These models combine some methods of EEG feature extraction with use of various filters and show significant improvement of performance in comparison to other models.

3. Methodology

3.1. Dataset

The widely used "grasp-and-lift" (GAL) dataset was used here that contains information about brain activity of 12 persons [7, 30]: more than 3900 trials (monitored and measured by the sampling rate of 500 Hz) in 32 channels of the recorded EEG signals. It contains data from the observed persons who performed 6 types physical activities (Table 1).

Table 1
Grasp-and-lift events

0	HandStart
1	FirstDigitTouch
2	BothStartLoadPhase
3	LiftOff
4	Replace
5	BothRelease

The data preprocessing was used only to cut regions of interests (ROIs) that correspond to the actual HCI physical actions of users. Some action signals overlap and their classification become more complex because they were not presented separate. So classes which intersect with other classes were excluded, namely FirstDigitTouch and Replace.

3.2. Models

Mainly, we compare three Recurrent Neural Network variants: RNN, long short term memory (LSTM) [31] and GRU [32], combining them with (Table 2):

- Hidden states at RNN layer (BiRNN HIDN STATES) (at all rows except row 1 in Table 2),
- Attention after RNN layers with hidden states (BiRNN HIDN ATNT) (rows 3, 5, 9, 12, 14 in Table 2)
- One dimensional Convolution layers (N Conv1D, where N - number of layers) (row 10 in Table 2)
- Two dimensional Convolution layers (N Conv2D, where N - number of layers) (rows 8, 9, 13 in Table 2)
- TimeDistributed wrapper for 1D and 2D Convolution layers(TD(N Conv1D)) (rows 10, 11, 12, 14, 15 in Table 2)
- 2D Separable Convolution layer (N SepConv2D) (rows 4, 5, 7 in Table 2)

All RNN layers was wrapped by bidirectional to have a sequence information in both directions. For each RNN variant, we had 15 following combinations (Table 2).

Table 2
Developed combinations for RNN-family

1	BiRNN
2	BiRNN HIDN STATES
3	BiRNN HIDN ATNT
4	BiRNN HIDN STATES 2SepConv2D
5	BiRNN HIDN ATNT 2SepConv2D
6	BiRNN HIDN STATES BiRNN
7	BiRNN HIDN STATES 2SepConv2D BiRNN
8	3Conv2D BiRNN HIDN STATES
9	3Conv2D BiRNN HIDN ATNT
10	TD(2Conv1D) BiRNN HIDN STATES
11	TD(1Conv2D) BiRNN HIDN STATES
12	TD(1Conv2D) BiRNN HIDN ATNT
13	3Conv2D BiRNN HIDN STATES 1Conv2D
14	TD(1Conv2D) BiRNN HIDN ATNT TD(1Conv2D)
15	TD(1Conv2D) BiRNN HIDN STATES TD(1Conv2D)

3.3. Metrics

Several standard metrics were used like accuracy and loss that were calculated during each run as the minimal value and maximal value of loss and accuracy, respectively. The area under curve (AUC) was measured for receiver operating characteristic (ROC) with their micro and macro versions, and their mean and standard deviation values. To determine the basic statistical properties of the metrics obtained (accuracy, loss, AUC) stratified k-fold cross-validation was applied (k=5) where the folds were created by preserving the percentage of samples for each class.

3.4. Workflow

The number of signal samples (N) in the input EEG time sequence (TS) was equal to 350 timepoints: 150 measurements before the first label, 150 measurements with labels and 50 measurements after the labeled data. At each epoch, the generators take data from each category from a randomly generated sequence. To diversify the data, it was decided to choose a starting point for the sequence to be used for training, validation and testing in a certain range randomly in the range of 10 measurements. The only present label was set as a ground truth (GT). The training, validation, and testing stages were performed for the GAL-dataset that was divided in proportion of 82.4% (3244 examples) / 8.8% (346 examples) / 8.8% (346 examples) for training / validation / testing sets, respectively. Finally, it allowed us to obtain trained models, calculate metrics (including AUC, and its micro and macro versions), and plot metrics versus the model types (see below). All important hyperparameters for used in models provided in Table 3. Hyperparameters was chosen based on previous experience and other related works in

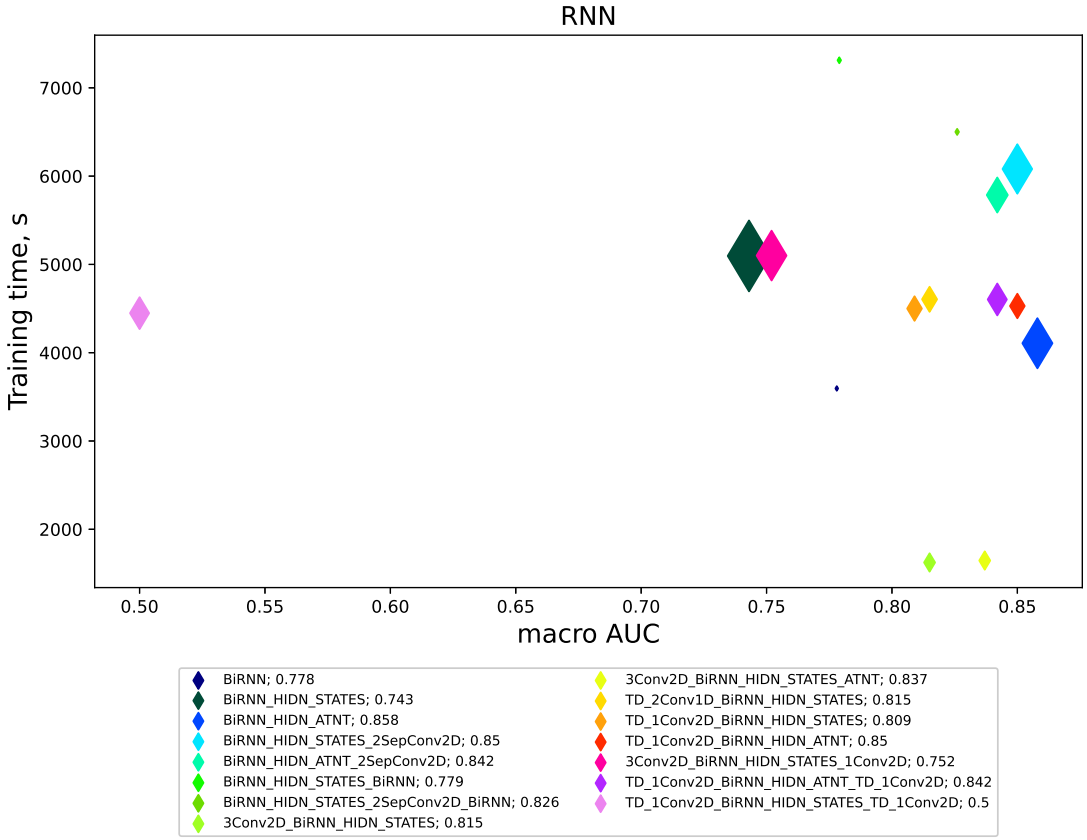

Figure 1: RNN-based models training time and macro AUC. Size of point is relative to number of parameters. Numerical in legend is correspondent to macro AUC values

field.

3.5. Experiment

The 5-fold cross-validation was applied, AUC macro values were determined for each fold, and the mean values were calculated for all AUC macro values determined for all 5 folds.

Then the scatter plots for training time (in seconds per a training part of the dataset) versus macro AUC values with the relative size of the models (given as a symbol size) were prepared (Fig. 1-3).

The performance of several separate families of hybrid combinations was measured and plotted in the correspondent scatter plots:

- RNN-based models (Fig. 1),
- LSTM-based models (Fig. 2),
- GRU-based models (Fig. 3).

These results allowed us to make comparative analysis of the models used with regard to

Table 3
Models hyperparameters

Category	Parameter	Value
Data Preprocesing	Batch size	128
	Epochs	15
	Time Steps	350
	Training/validation/testing sets	82.4% / 8.8% / 8.8%
Conv Layers	Number of conv filters	32|16|8
	Size of conv filters	8|4|2
	Strides	2|1
	Padding	valid
	Conv activation function	tanh
	Pooling size	Maxpooling, 4|2
RNN Layers	Number of RNN units	175
Last 2 Fully Connected Layers	Number of neurons	350, 4
	Activation functions	tanh, sigmoid
Learning	Learning rate	0.0001
	Loss	Categorical Cross Entropy
	Optimizer	Adam
	Validation	k-fold Stratified, k=5

their performance (AUC) and resources needed for model preparation (a training time) and storage (a relative size expressed by a symbol size).

To understand the complex relations between separate components and the whole hierarchy of the models used, the tree-like representation was prepared and given for RNN-based models (Fig. 4), LSTM-based models (Fig. 5), and GRU-based models (Fig. 6).

The nodes in the tree-like plots (Fig. 4-6) denote the models created with the labels containing information about specific details about the components applied (see Table 2).

The edges between nodes denote the hierarchical relationships between them. The sizes of solid symbols (circles) denote the relative sizes of the models.

4. Discussion

The results shown on the scatter plots (Fig. 1-3) allowed us to make the following observations as to training times (in seconds per a training part of the dataset), macro AUC values, and relative sizes of the models (given as a symbol size).

In the RNN-family (Fig. 1), some changes of components can have the very drastic consequences. For example, the significant increase of performance by 11.5% can be obtained by transition from BiRNN HIDN STATES model (AUC=0.743) to BiRNN ATNT model (AUC=0.858). Analogously, the higher increase of performance by 34.2% can be obtained by transition from TD 1Conv2D BiRNN HIDN STATES TD 1Conv2D model (AUC=0.5) to TD 1Conv2D BiRNN HIDN ATNT TD 1Conv2D model (AUC=0.842). And not so high increase by 1.8% can be observed by transition from 3Conv2D BiRNN HIDN STATES model (AUC=0.815) to 3Conv2D BiRNN HIDN

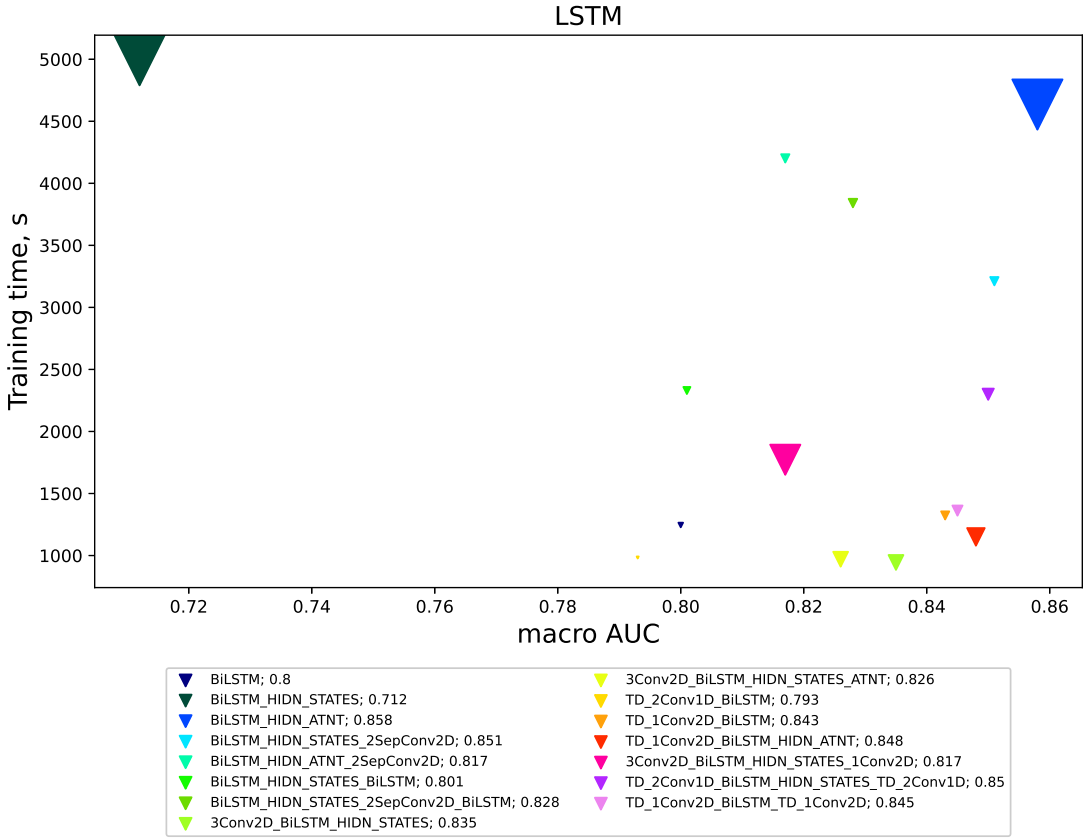

Figure 2: LSTM-based models training time and macro AUC.Size of point is relative to number of parameters. Numerical in legend is correspondent to macro AUC values

STATES ATNT model (AUC=0.837).

In the LSTM-family (Fig. 2), the similar changes of components lead to the following effects. Transition from BiLSTM HIDN STATES (AUC=0.712) to BiLSTM HIDN ATNT (AUC = 0.858) improves performance by 14.6% and reduces training time approximately by 8 minutes. However adding Attention not always yields in performance improvement, like in case with BiLSTM HIDN STATES 2SepConv2D (AUC=0.851) and BiLSTM HIDN ATNT 2SepConv2D (AUC=0.817), but still attention results shorts training time, for this models by 16 minutes. Decrease of AUC after adding attention also happens with models 3Conv2D BiLSTM HIDN STATES (AUC=0.835) and 3Conv2D BiLSTM HIDN ATNT (AUC=0.826) with relative slight training time reduction. If in previous 3Conv2D BiLSTM HIDN STATES (AUC=0.835) model we add another Conv2D layer instead of attention 3Conv2D BiLSTM HIDN 1Conv2D (AUC=0.817), it will worsen not only AUC values but also training time approximately by 10 minutes. Adding attention to models which use TimeDistributed 2D Convolution layers before BiLSTM improves performance by little 0.5% for TD 1Conv2D BiLSTM (AUC=0.843) and TD 1Conv2D BiLSTM HIDN ATNT (AUC=0.843).

In the GRU-family (Fig. 3), the same changes of components cause the following changes

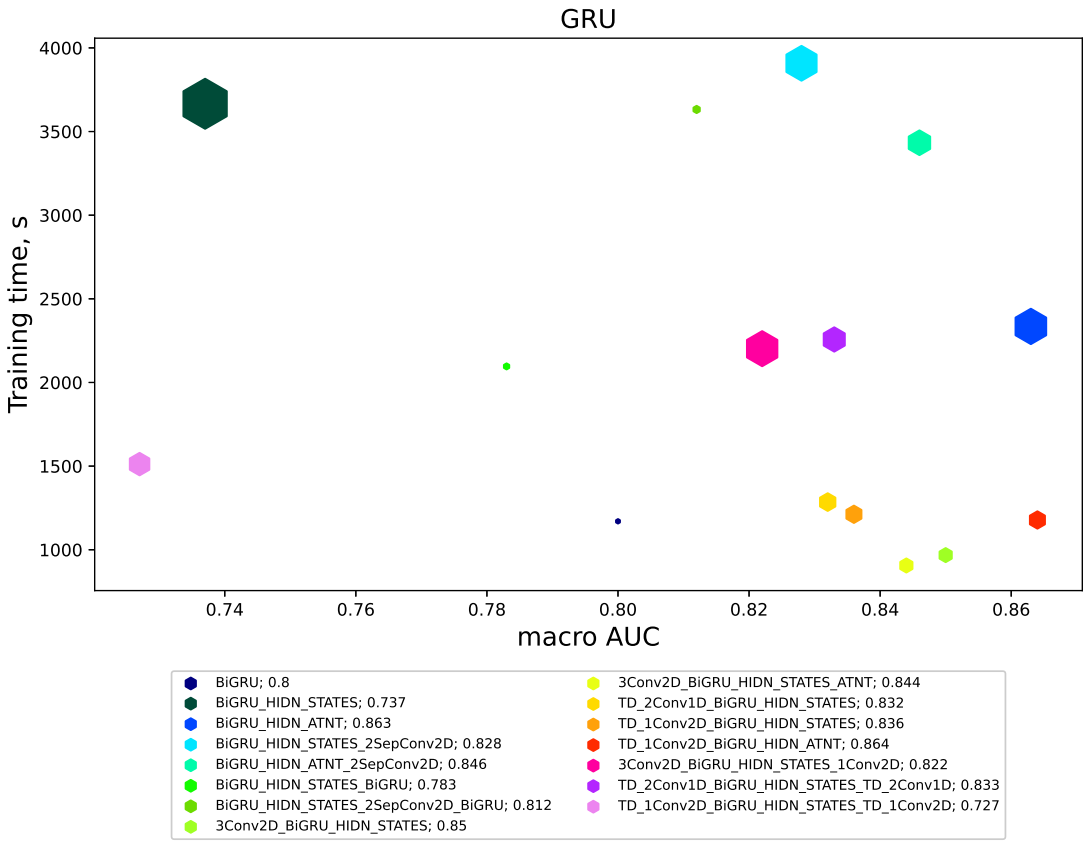

Figure 3: GRU-based models training time and macro AUC. Size of point is relative to number of parameters. Numerical in legend is correspondent to macro AUC values

of performance. Addition to BiGRU HIDN STATES (AUC=0.737) attention mechanism BiGRU HIDN ATNT (AUC=0.863) will increase performance by 12.6%. Dissimilar to LSTM-family adding atention to BiGRU HIDN STATES 2SepConv2D (AUC=0.828) slightly improves perfomance by 1.8%. Transition from TD 1Conv2D BiGRU HIDN STATES (AUC=0.836) to TD 1Conv2D BiGRU HIDN ATNT (AUC=0.864) increases perfomance by 2.8%.

The results demonstrated on the tree-like plots (Fig. 4-6) allowed us to build hierarchy of models and find some relationships between models that can lead to increase or decrease of the performance (macro AUC values).

In the RNN-tree of models (Fig. 4), some branches (several nodes connected by edges and following from the central root node to edge nodes) can lead to:

- increase of performance: BiRNN HIDN STATES (AUC=0.743) → TD 1Conv2D BiRNN HIDN STATES (**AUC=0.809**) → TD 1Conv2D BiRNN HIDN ATNT (**AUC=0.85**),
- or decrease of performance: BiRNN HIDN STATES (AUC=0.743) → TD 1Conv2D BiRNN HIDN STATES (*AUC=0.809*) → TD 1Conv2D BiRNN HIDN STATES TD 1Conv2D (*AUC=0.5*).

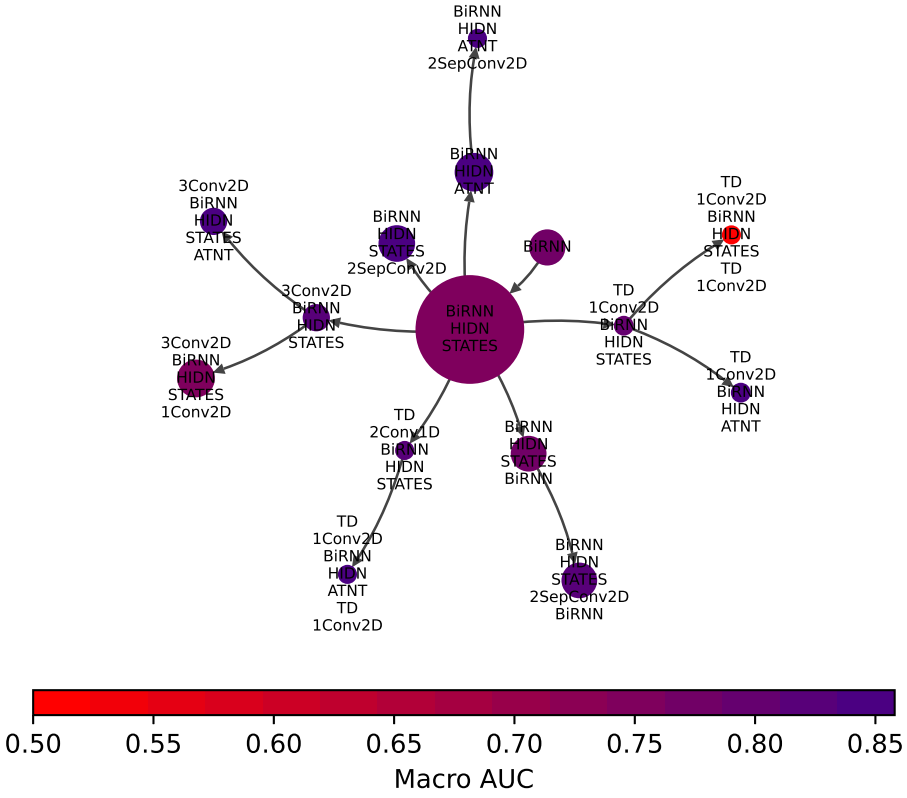

Figure 4: RNN-based models tree

In the LSTM-tree of models (Fig. 5),

- increase of performance: BiLSTM HIDN STATES (AUC=0.712) → BiLSTM HIDN ATNT (**AUC=0.858**) → BiLSTM HIDN ATNT 2SepConv2D (**AUC=0.817**),
- increase of performance: BiLSTM HIDN STATES (AUC=0.712) → TD 1Conv2D BiLSTM (**AUC=0.843**) → TD 1Conv2D BiLSTM TD 1Conv2D (**AUC=0.845**),

In the GRU-tree of models (Fig. 6),

- increase of performance: BiGRU HIDN STATES (AUC=0.737) → TD 1Conv2D BiGRU HIDN STATES (**AUC=0.836**) → TD 1Conv2D BiGRU HIDN ATNT (**AUC=0.864**),
- or decrease of performance: BiGRU HIDN STATES (AUC=0.737) → TD 1Conv2D BiGRU HIDN STATES (**AUC=0.836**) → TD 1Conv2D BiGRU HIDN TD 1Conv2D (**AUC=0.727**)

As one can see the attention mechanism (ATNT) after RNN layer with hidden states (HIDN) in all RNN variants increases macro AUC from 34% to 2% and slightly decrease training time.

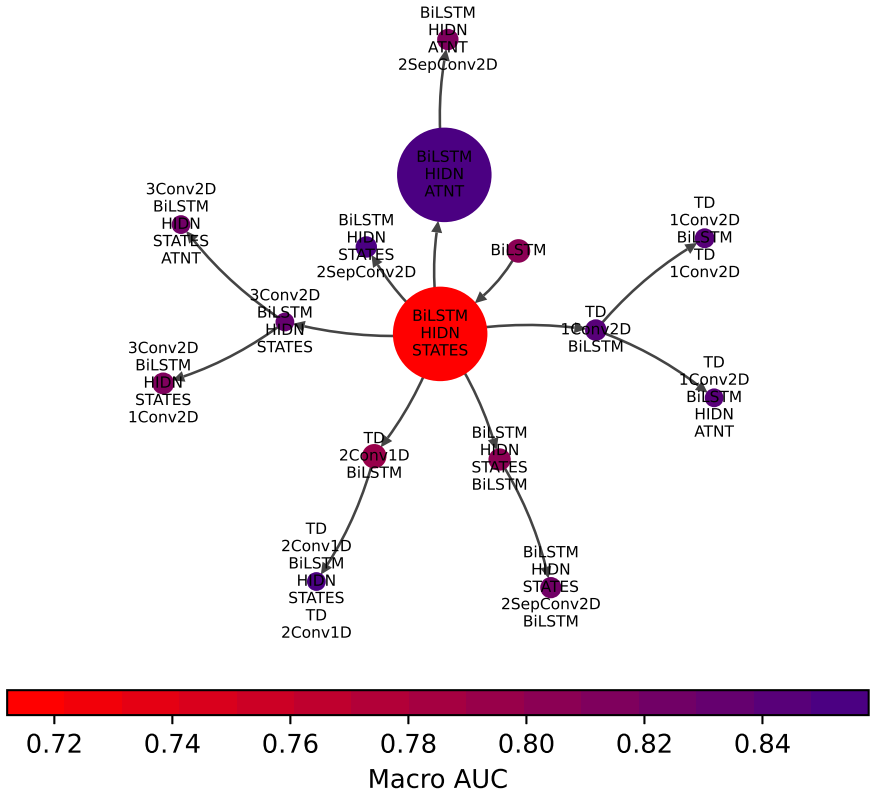

Figure 5: LSTM-based models tree

But using RNN layer with hidden states as input for last fully-connected layer always give worse results.

Using convolution layers as first layers can significantly decrease training time with preserving macro AUC. But using convolution layers after RNN layers with hidden states or hidden states and attention, just decrease number of parameters, without boosting macro AUC and training time.

5. Conclusions

In this article several hybrid combinations of DNN were applied to GAL dataset for hand movement classification. Macro AUC was chosen as a performance metric and compared to , training time and model size (the number of parameters). The results shown that using RNN layer with hidden states as input for last fully-connected layer decreased performance, but adding attention mechanism after output with hidden states solves this problem. Also applying

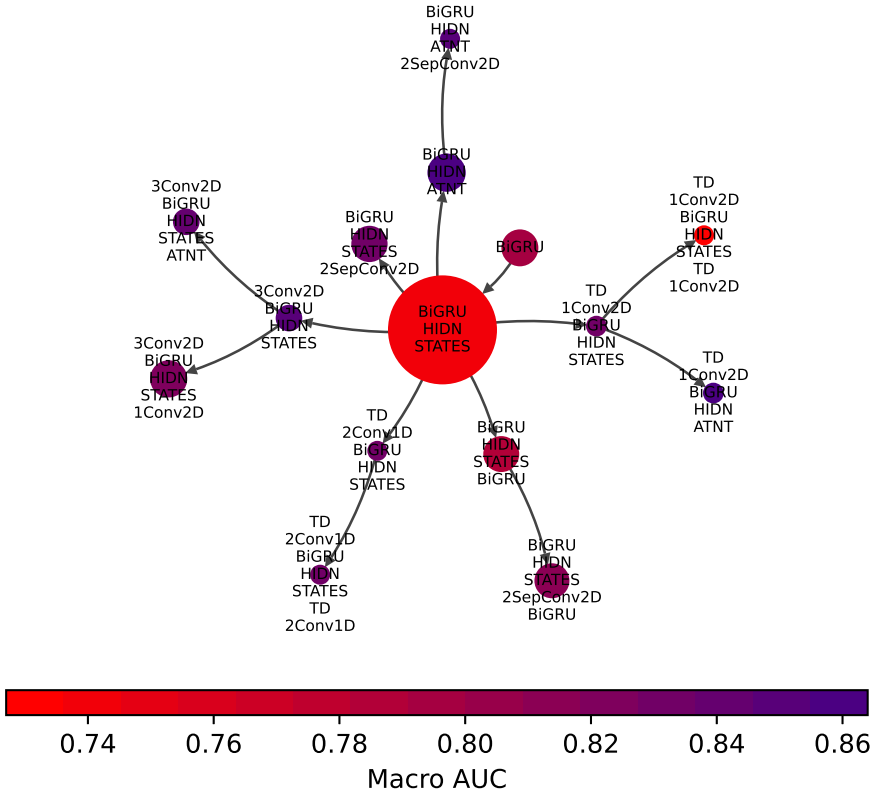

Figure 6: GRU-based models tree

BiRNN with CNN as first layers improves overall macro AUC, reduces number of parameters, and makes training model more computationally effective.

Acknowledgment

The work was supported by "Knowledge At the Tip of Your fingers: Clinical Knowledge for Humanity" (KATY) project funded from the European Union's Horizon 2020 research and innovation program under grant agreement No. 101017453.

References

[1] B. S. Armour, E. A. Courtney-Long, M. H. Fox, H. Fredine, A. Cahill, Prevalence and causes of paralysis—united states, 2013, American Journal of Public Health 106 (2016) 1855–1857. doi:10.2105/AJPH.2016.303270, pMID: 27552260.

[2] A. N. Belkacem, N. Jamil, J. Palmer, S. Ouhbi, C. Chao, Brain computer interfaces for improving the quality of life of older adults and elderly patients, Frontiers in Neuroscience 14 (2020) 692. doi:10.3389/fnins.2020.00692.

[3] G. Li, C. H. Lee, J. J. Jung, Y. C. Youn, D. Camacho, Deep learning for eeg data analytics: A survey, Concurrency and Computation: Practice and Experience 32 (2020) e5199.

[4] S. Aggarwal, N. Chugh, Review of machine learning techniques for eeg based brain computer interface, Archives of Computational Methods in Engineering (2022) 1–20.

[5] M. Zabcikova, Z. Koudelkova, R. Jasek, J. J. Lorenzo Navarro, Recent advances and current trends in brain-computer interface research and their applications, International Journal of Developmental Neuroscience 82 (2022) 107–123.

[6] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, B. J. Lance, Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces, Journal of neural engineering 15 (2018) 056013.

[7] Grasp-and-lift eeg detection dataset, 2022. URL: https://www.kaggle.com/c/grasp-and-lift-eeg-detection/data, accessed on September, 24, 2022.

[8] J. An, S. Cho, Hand motion identification of grasp-and-lift task from electroencephalography recordings using recurrent neural networks, in: 2016 International Conference on Big Data and Smart Computing (BigComp), 2016, pp. 427–429. doi:10.1109/BIGCOMP.2016.7425963.

[9] M. K. Hasan, S. R. Wahid, F. Rahman, S. K. Maliha, S. B. Rahman, Grasp-and-lift detection from eeg signal using convolutional neural network, in: 2022 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE), 2022, pp. 1–6. doi:10.1109/ICAEEE54957.2022.9836375.

[10] Y. Gordienko, K. Kostiukevych, N. Gordienko, O. Rokovyi, O. Alienin, S. Stirenko, Deep learning for grasp-and-lift movement forecasting based on electroencephalography by brain-computer interface, in: Z. Hu, Q. Zhang, S. Petoukhov, M. He (Eds.), Advances in Artificial Systems for Logistics Engineering, Springer International Publishing, Cham, 2021, pp. 3–12.

[11] Y. Gordienko, K. Kostiukevych, N. Gordienko, O. Rokovyi, O. Alienin, S. Stirenko, Deep learning with noise data augmentation and detrended fluctuation analysis for physical action classification by brain-computer interface, in: 2021 8th International Conference on Soft Computing & Machine Intelligence (ISCMI), IEEE, 2021, pp. 176–180.

[12] K. Kostiukevych, Y. Gordienko, N. Gordienko, O. Rokovyi, O. Alienin, S. Stirenko, Convolutional and recurrent neural networks for physical action forecasting by brain-computer interface, in: 11th IEEE Int. Conf. on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IEEE, 2021.

[13] P. Aricò, G. Borghini, G. Di Flumeri, A. Colosimo, S. Pozzi, F. Babiloni, A passive brain–computer interface application for the mental workload assessment on professional air traffic controllers during realistic air traffic control tasks, Progress in brain research 228 (2016) 295–328.

[14] P. Aricò, G. Borghini, G. Di Flumeri, A. Colosimo, S. Bonelli, A. Golfetti, S. Pozzi, J.-P. Imbert, G. Granger, R. Benhacene, et al., Adaptive automation triggered by eeg-based mental workload index: a passive brain-computer interface application in realistic air traffic control environment, Frontiers in human neuroscience 10 (2016) 539.

[15] G. Di Flumeri, F. De Crescenzio, B. Berberian, O. Ohneiser, J. Kramer, P. Aricò, G. Borghini, F. Babiloni, S. Bagassi, S. Piastra, Brain–computer interface-based adaptive automation to prevent out-of-the-loop phenomenon in air traffic controllers dealing with highly automated systems, Frontiers in human neuroscience 13 (2019) 296.

[16] X. Chen, C. Li, A. Liu, M. J. McKeown, R. Qian, Z. J. Wang, Toward open-world electroencephalogram decoding via deep learning: a comprehensive survey, IEEE Signal Processing Magazine 39 (2022) 117–134.

[17] X. Wan, K. Zhang, S. Ramkumar, J. Deny, G. Emayavaramban, M. S. Ramkumar, A. F. Hussein, A review on electroencephalogram based brain computer interface for elderly disabled, IEEE Access 7 (2019) 36380–36387.

[18] X. Gu, Z. Cao, A. Jolfaei, P. Xu, D. Wu, T.-P. Jung, C.-T. Lin, Eeg-based brain-computer interfaces (bcis): A survey of recent studies on signal sensing technologies and computational intelligence approaches and their applications, IEEE/ACM transactions on computational biology and bioinformatics (2021). doi:10.1109/TCBB.2021.3052811.

[19] J. Xu, B. Zhong, Review on portable eeg technology in educational research, Computers in Human Behavior 81 (2018) 340–349.

[20] P. Gang, J. Hui, S. Stirenko, Y. Gordienko, T. Shemsedinov, O. Alienin, Y. Kochura, N. Gordienko, A. Rojbi, J. L. Benito, et al., User-driven intelligent interface on the basis of multimodal augmented reality and brain-computer interaction for people with functional disabilities, in: Future of Information and Communication Conference, Springer, Cham, 2018, pp. 612–631.

[21] J. Belo, M. Clerc, D. Schön, Eeg-based auditory attention detection and its possible future applications for passive bci, Frontiers in Computer Science 3 (2021) 661178.

[22] B. Kerous, F. Skola, F. Liarokapis, Eeg-based bci and video games: a progress report, Virtual Reality 22 (2018) 119–135.

[23] G. A. M. Vasiljevic, L. C. de Miranda, Brain–computer interface games based on consumer-grade eeg devices: A systematic literature review, International Journal of Human–Computer Interaction 36 (2020) 105–142.

[24] G. Cattan, The use of brain–computer interfaces in games is not ready for the general public, Frontiers in computer science 3 (2021) 628773.

[25] B. Lin, S. Deng, H. Gao, J. Yin, A multi-scale activity transition network for data translation in eeg signals decoding, IEEE/ACM Transactions on Computational Biology and Bioinformatics (2020). doi:10.1109/TCBB.2020.3024228.

[26] R. Gatti, Y. Atum, L. Schiaffino, M. Jochumsen, J. B. Manresa, Prediction of hand movement speed and force from single-trial eeg with convolutional neural networks, bioRxiv (2019) 492660.

[27] J. An, S. Cho, Hand motion identification of grasp-and-lift task from electroencephalography recordings using recurrent neural networks, in: 2016 International Conference on Big Data and Smart Computing (BigComp), IEEE, 2016, pp. 427–429.

[28] N. Wang, A. Farhadi, R. Rao, B. Brunton, Ajile movement prediction: Multimodal deep learning for natural human neural recordings and video, in: Proc. of AAAI Conf. on Artificial Intelligence, volume 32, 2018.

[29] S. Pancholi, A. Giri, A. Jain, L. Kumar, S. Roy, Source aware deep learning framework for hand kinematic reconstruction using eeg signal, arXiv preprint arXiv:2103.13862 (2021).

[30] M. D. Luciw, E. Jarocka, B. B. Edin, Multi-channel eeg recordings during 3,936 grasp and lift trials with varying weight and friction, Scientific data 1 (2014) 1–11.
[31] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.
[32] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, arXiv preprint arXiv:1409.1259 (2014).