Examining the Performance of Neural Networks for EEG Classification

Sarthak Vora
006299150
sarthakvora@ucla.edu

Vignesh Nagarajan
606185377
vignesh@g.ucla.edu

Aryaman Gokarn
506303588
aryamangokarn09@ucla.edu

Akshat Mehta
906182141
akshatmehta@ucla.edu

Abstract

This study conducts a thorough examination of EEG data classification sourced from the BCI Competition [6] employing a range of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). Our analysis begins with foundational architectures such as Deep Convolutional Networks (Deep CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU), progressively advancing to hybrid models like CNN-LSTM and CNN-GRU. Additionally, we assess the efficacy of cutting-edge architectures, namely TSCeption [4], EEGNet [5], and ATCNet [2], leveraging their spatio-temporal convolutions and attention mechanisms to enhance classification performance. Through comprehensive performance evaluation and ablation studies, we analyze trends in classification performance across diverse model architectures and provide valuable insights into their comparative efficacy.

1. Introduction

The Electroencephalography (EEG) data provided within the Brain-Computer Interface (BCI) competition [6] consists of four distinct motor-imagery tasks, namely the movement of the left hand, right hand, both, feet and tongue. We aim to classify the EEG data collected from nine subjects as they perform these four tasks.

Each EEG dataset analyzed in this study comprises of 1000 timesteps of EEG time-series signals recorded from 22 electrodes placed on each of the nine subjects. Consequently, for every trial, we acquire 22 correlated time-series data points per subject. This data structure motivates the exploration of time-series or sequence-based neural networks.

Recurrent neural networks (RNNs), especially long short-term memory (LSTM) and Gated Recurrent Units (GRU) are well-suited to model sequential data and extract temporal dependencies. We hypothesize that combining CNNs and RNNs into a hybrid architecture could leverage their complementary strengths - using CNNs as encoders to derive spatial features and feed into RNNs (LSTM and GRU) as decoders to learn the temporal frequency content.

Moreover, we evaluated TSCeption [4], EEGNet [5] and ATCNet [2], compact but advanced CNN-based architectures specifically designed for EEG-based motor imagery classification. The dynamic temporal layer in TSCeption, piece-wise temporal and depthwise convolution in EEGNet, and robust temporal attention convolutions in ATCNet naturally set the base for complex spatio-temporal feature decoding by effectively learning frequency filters and finding relevance across temporal depth. By exploring these diverse set of models - CNNs, LSTMs, GRUs, CNN-LSTMs, CNN-GRUs, TSCeption, ATCNet, and EEGNet - we aim to assess their relative potential for fusing spatial and temporal modeling on EEG data.

Furthermore, we experimented with the preprocessing method of subsampling, noise averaging, and max-pooling, leading to notable performance boost. Additionally, we introduced Random Frequency Shifting to facilitate neural network computations with randomized frequency cues, aiming to improve generalizability.

Motivating these architectures from different angles, CNN extracts hierarchical spatial representations, LSTMs and GRUs specialize in learning sequential patterns, CNN-LSTMs and CNN-GRUs jointly map spatial and temporal factors, and TSCeption, EEGNet, and ATCNet architectures demonstrate an EEG-specific architectural inductive bias.

2. Method

2.1. Data Pre-Preprocessing

We explore two preprocessing techniques. Initially, we truncate the data across timesteps and apply a max-pooling layer. Subsequently, we average the data, introduce noise, and then subsample it.

In our second method, termed Random Frequency Shift,

we subject the truncated data to a transformation that randomly shifts frequencies, with minimum and maximum shifts set to -2 and +2, respectively.

2.2. Model Selection

We start with evaluating fundamental RNN and CNN variants, progress to hybrid CNN-RNN architectures, and then implement specialized EEG architectures such as EEGNet, attention-based ATCNet, and spatiotemporal models like TSCeption.

2.3. Training and Hyperparameters

All models were trained using cross-entropy loss with weight decay (L2 Regularization). Early stopping was conducted on a validation set to address overfitting. After extensive fine-tuning and iterative experimentation, we finalized these hyperparameters:

Batch Size	64
Learning Rate	10^{-4}
Weight Decay	10^{-4}
Epochs	200
Dropout	0.5
Loss	Cross-Entropy Loss
Optimizer	Adam

Table 1. Summary of Hyperparameters

3. Results

Table 2 summarizes the performance of all models. We provide performance evaluations for both, preprocessed and unprocessed data, setting on the best-performing time window of *initial 500 timesteps*. Across all models, the traditional data preprocessing technique consistently delivers superior performance. Additionally, our Random Frequency-Shift approach yields slight improvements across all models, contributing to improved variability.

3.1. Deep Convolutional Network (CNN)

We employed a 4-layer Convolutional Neural Network (CNN) with three activation functions—ELU, ReLU, and LeakyReLU—to evaluate their effectiveness in classification tasks with and without preprocessing.

Refer to Table 3 for performance metrics. Notably, ELU (Figure 4) consistently outperforms the other activation functions, attaining the highest accuracy.

3.2. RNN - LSTM and GRU

Vanilla RNN and LSTM models demonstrate subpar performance, achieving only 25-27% accuracy, albeit showing notable improvement with data preprocessing.

3.3. Hybrid CNN-RNN

With data preprocessing, a hybrid CNN-LSTM architecture outperforms naive LSTM or GRU models. Nevertheless, its performance remains suboptimal to that of CNN.

With data preprocessing, CNN-GRU demonstrates approximately a 5% improvement over CNN-LSTM and approximately 2% without preprocessing. CNN-GRU stands out as the better hybrid architecture for classification task.

3.4. TSCeption

With preprocessing, the TSCeption [4] model demonstrated a commendable accuracy of 51.46%. Interestingly, without preprocessing, the accuracy only slightly decreased to 48.98%. These results indicate the robustness of the TSCeption architecture.

3.5. ATCNet

ATCNet [2] shows a notable performance improvement compared to both hybrid and basic CNN and RNN models. Achieving an accuracy above 60% underscores the effectiveness of the robust attention and temporal convolution components integrated within the model.

3.6. EEGNet

EEGNet [5] achieves the highest accuracy of 74.26%, establishing itself as our top-performing model. Its novel spatial and temporal depthwise convolution architecture sets it apart, resulting in a vast improvement in accuracy compared to other models tested.

3.6.1 Ablation across different subjects

For this best-performing model, we conducted an ablation study by training on individual subjects and testing on each of them and training on the whole and testing on each subject. The histogram plot comparison between the two experiments is shown in Fig. 3.

3.6.2 Ablation across different timesteps

We also conducted an ablation across the time duration, incrementally increasing the number of timeframes with a timestep of 100 as shown in Fig. 2.

4. Discussion

4.1. Effect of Data Augmentation and Preprocessing

The conventional data preprocessing technique improves results by enhancing model robustness through data subsampling, averaging, and noise addition, providing a diverse dataset for training. It also helps to capture informative features from EEG data and mitigate variability, leading to better generalization. Regularization techniques within the

function and the increased data volume also lead to the observed performance improvement.

The ***RandomFrequencyShift*** transformation introduces random shifts to the frequency components of the EEG signal, modifying its spectral characteristics while preserving its temporal structure. Random time shifts could potentially distort the temporal sequence of events, thereby disrupting the inherent temporal relationships encoded within the EEG signal. By augmenting the dataset with these frequency shifts, the model acquires robust features that better capture the underlying patterns in the data.

4.2. Deep Convolutional Network (CNN)

ELU activation's superior performance may lie in its more effective handling of negative inputs. Unlike ReLU, which suffers from saturation issues where neurons become inactive during training, ELU permits negative values to pass through. This characteristic enables ELU to potentially capture nuanced data variations, thereby enhancing classification performance.

4.3. RNN - LSTM and GRU

The improvement in accuracy in GRU and LSTM on preprocessing may stem from improved data filtering and robustness. GRU demonstrates slightly greater performance, attributed to its fewer training parameters, facilitating faster learning with fewer samples. Nonetheless, the relatively lower accuracy in these basic models may imply limitations in capturing temporal dependencies.

4.4. Hybrid CNN-RNN

Preprocessing significantly enhances CNN-LSTM accuracy by extracting relevant features, but without it, accuracy decreases due to noise vulnerability. This could be due to LSTM's overfitting from its higher number of parameters and potentially redundant use in decoding EEG data with potentially weak temporal dependencies.

The suboptimal performance of the CNN-GRU model may be attributed to the domain gap between the features computed by CNN and GRU layers. Although GRU networks are proficient in capturing temporal dependencies, their integration with CNN layers at the feature level may not seamlessly combine spatial and temporal data, primarily due to a learning-domain gap between both feature representations.

4.5. TSCeption

The TSCeption model's (see Fig. 11) robust performance, even in the absence of pre-processing, is due to its ability to manage the non-stationary and noisy characteristics of EEG signals. This capability allows it to extract more informative features for classification tasks.

Furthermore, TSCeption combines temporal frequency filters in the spatial layer at a lower level, proving it effective in combining both local and global temporal dependencies across space within time series data.

4.6. ATCNet

ATCNet (Fig. 12) encodes low-level spatiotemporal information through a convolutional block. Subsequently, the attention module selectively passes relevant temporal sequences for classification to the fully connected layer. This approach enables the model to discard redundant temporal cues and focus solely on important features, facilitating improved learning from the noisy and redundant EEG signal. However, the attention blocks may also introduce unnecessary complexity to the task, potentially resulting in overfitting and a slightly suboptimal accuracy.

4.7. EEGNet

EEGNet (Fig. 10) integrates depth-wise separable convolutions and temporal convolutional layers, meticulously crafted to capture both temporal and spatial patterns. Its simplicity and effectiveness make it well-suited for the task at hand. By dividing a 3D convolutional filter into spatial and temporal operations, EEGNet sequentially extracts spatial features followed by temporal convolutions, effectively combining the two representations. This simple design enables EEGNet to extract essential discriminative features without overfitting thereby enhancing its accuracy.

4.7.1 Ablation across different subjects

Training the EEGNet model on the entire dataset and testing it on individual subjects consistently exhibits superior or comparable performance as compared to training and testing on the same subject. This observation suggests that the model benefits from the broader spectrum of data during training, enhancing its ability to generalize across subjects. Additionally, some subjects, like Subject 9 (illustrated in Fig. 1), present EEG signals with indistinct patterns among individual class signals, posing challenges for effective learning.

4.7.2 Ablation across different timesteps

We found that the time duration ranging from 300 to 500 timesteps resulted in the highest accuracy for EEGNet, with peak performance around 500 timesteps. This observation suggests that the initial half of the time-series data is more discriminative and beneficial for classification. Conversely, the latter half appears less informative, possibly due to increased noise or reduced variance across class signals.

References

[1] Imran Ahmed, Gwanggil Jeon, and Francesco Piccialli. From artificial intelligence to explainable artificial intelligence in industry 4.0: a survey on what, how, and where. *IEEE Transactions on Industrial Informatics*, 18(8):5031–5042, 2022. 4

[2] Hamdi Altaheri, Ghulam Muhammad, and Mansour Alsulaiman. Physics-informed attention temporal convolutional network for eeg-based motor imagery classification. *IEEE Transactions on Industrial Informatics*, 19(2):2249–2258, 2023. 1, 2

[3] Hamdi Altaheri, Ghulam Muhammad, Mansour Alsulaiman, Syed Umar Amin, Ghadir Ali Altuwaijri, Wadood Abdul, Mohamed A Bencherif, and Mohammed Faisal. "deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: A review". *Neural Computing and Applications*, 35(20):14681–14722, 2023. 4

[4] Yi Ding, Neethu Robinson, Su Zhang, Qiuhao Zeng, and Cuntai Guan. Tsception: Capturing temporal dynamics and spatial asymmetry from eeg for emotion recognition. *IEEE Transactions on Affective Computing*, 14(3):2238–2250, 2023. 1, 2

[5] Vernon J. Lawhern, Amelia J. Solon, Nicholas R. Waytowich, Stephen M. Gordon, Chou P. Hung, and Brent J. Lance. Eegnet: A compact convolutional network for eeg-based brain-computer interfaces. *CoRR*, abs/1611.08024, 2016. 1, 2

[6] Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Gernot Mueller-Putz, Guido Nolte, et al. Review of the bci competition iv. *Frontiers in neuroscience*, 6:21084, 2012. 1

Table 2. Comparison of Model Performances

Model	Accuracy (500 timesteps)		
	With Preprocess	W/o Preprocess	Random Frequency Shift
DeepCNN	65.69%	60.27%	64.55%
ATCNet	68.62%	61.63%	66.59%
EEGNet*	**74.26%**	**69.75%**	**71.55%**
GRU	34.09%	25.96%	28.21%
LSTM	33.41%	27.09%	30.69%
CNNGRU	51.47%	26.19%	27.99%
CNNLSTM	46.05%	22.80%	28.44%
TSCeption	51.46%	48.98%	48.79%

Table 3. Comparison of Deep Convolutional Network Performance with Different Activation Functions

Activation Function	With Preprocessing	Without Preprocessing
ReLU	60.50%	51.92%
LeakyReLU	62.01%	57.69%
ELU	65.69%	60.27%

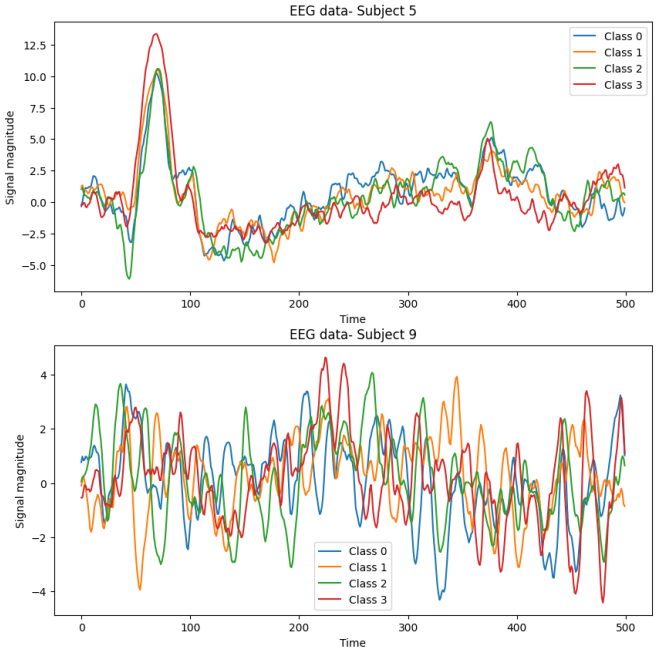

Figure 1. Visualization of EEG data for four classes of Subject 5 and Subject 9. A notable distinction is observed in the class-specific signals of Subject 9 as compared to Subject 5.

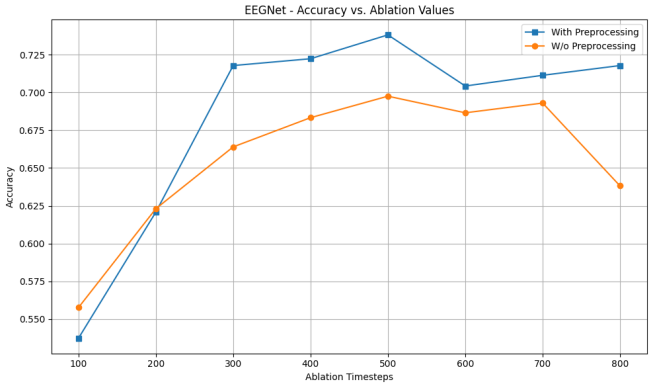

Figure 2. EEGNet accuracy variation across timesteps varied from 100 to 800 with step-size of 100

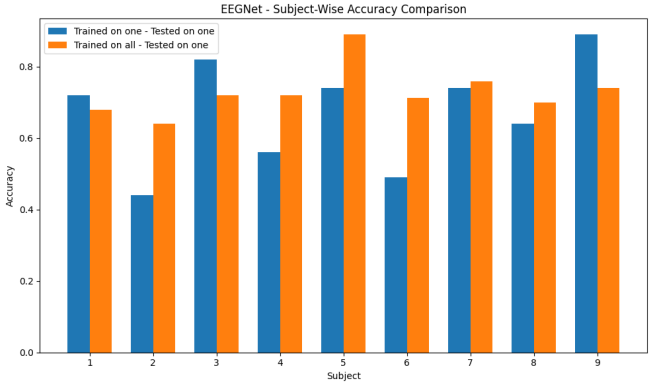

Figure 3. EEGNet accuracy across subjects for timestep=500, for two cases, 'Trained on one - Tested on one' and 'Trained on all - Tested on one'

Layer (type)	Output Shape
Conv2d-1	[-1, 25, 22, 996]
Conv2d-2	[-1, 25, 21, 996]
BatchNorm2d-3	[-1, 25, 21, 996]
ELU-4	[-1, 25, 21, 996]
MaxPool2d-5	[-1, 25, 21, 498]
Dropout-6	[-1, 25, 21, 498]
Conv2d-7	[-1, 50, 21, 494]
BatchNorm2d-8	[-1, 50, 21, 494]
ELU-9	[-1, 50, 21, 494]
MaxPool2d-10	[-1, 50, 21, 247]
Dropout-11	[-1, 50, 21, 247]
Conv2d-12	[-1, 100, 21, 243]
BatchNorm2d-13	[-1, 100, 21, 243]
ELU-14	[-1, 100, 21, 243]
MaxPool2d-15	[-1, 100, 21, 121]
Dropout-16	[-1, 100, 21, 121]
Conv2d-17	[-1, 200, 21, 117]
BatchNorm2d-18	[-1, 200, 21, 117]
ELU-19	[-1, 200, 21, 117]
MaxPool2d-20	[-1, 200, 21, 58]
Dropout-21	[-1, 200, 21, 58]
Flatten-22	[-1, 243600]
Linear-23	[-1, 4]

Figure 4. Deep Convolution Network ELU architecture

Layer (type)	Output Shape
GRU-1	[[-1, 1000, 64], [-1, 2, 64]]
Linear-2	[-1, 4]

Figure 5. GRU architecture

```
LSTM(
  (gru_layer): LSTM(22, 64, num_layers=2, batch_first=True)
  (out): Linear(in_features=64, out_features=4, bias=True)
)
```

Figure 6. LSTM architecture

Layer (type)	Output Shape
Conv1d-1	[-1, 40, 496]
BatchNorm1d-2	[-1, 40, 496]
ReLU-3	[-1, 40, 496]
GRU-4	[[-1, 496, 50], [-1, 496, 50]]
Linear-5	[-1, 4]

Figure 7. CNN GRU architecture

Layer	Kernel Shape	Output Shape
0_cnn.Conv1d_0	[1, 32, 10]	[1, 32, 46]
1_cnn.BatchNorm1d_1	[32]	[1, 32, 46]
2_cnn.ReLU_2	–	[1, 32, 46]
3_rnn	–	[46, 1, 64]
4_fc	[64, 10]	[1, 10]

Figure 8. CNN LSTM architecture

Layer (type)	Output Shape
Conv2d-1	[-1, 16, 22, 1000]
BatchNorm2d-2	[-1, 16, 22, 1000]
Conv2d-3	[-1, 32, 21, 1000]
BatchNorm2d-4	[-1, 32, 21, 1000]
ELU-5	[-1, 32, 21, 1000]
AvgPool2d-6	[-1, 32, 21, 250]
Dropout-7	[-1, 32, 21, 250]
Conv2d-8	[-1, 32, 21, 250]
BatchNorm2d-9	[-1, 32, 21, 250]
ELU-10	[-1, 32, 21, 250]
AvgPool2d-11	[-1, 32, 21, 31]
Dropout-12	[-1, 32, 21, 31]
Flatten-13	[-1, 20832]
Linear-14	[-1, 4]

Figure 9. EEGNet architecture

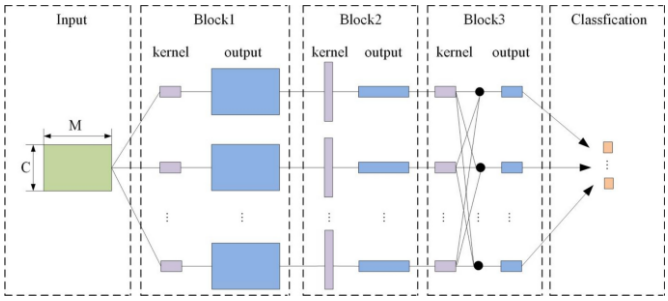

Figure 10. EEGNet architecture - Visual representation

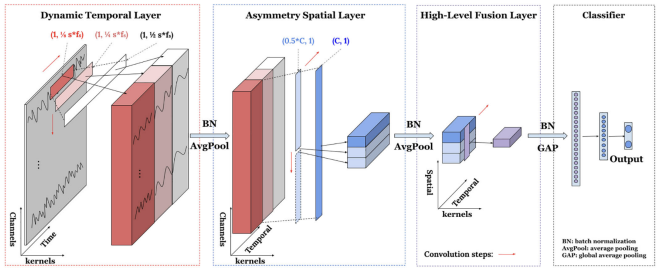

Figure 11. TSCeption architecture - Visual representation

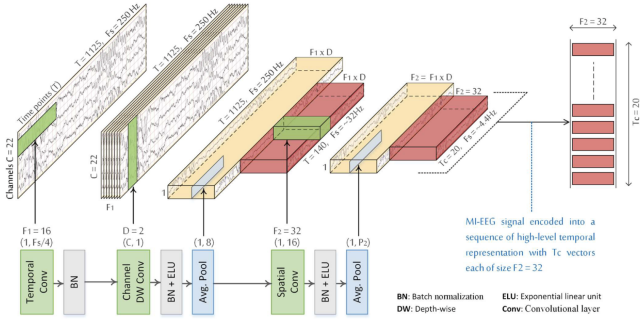

Figure 12. ATCNet architecture - Visual representation