Motor Imagery Classification From EEG Data

Huijun Hao Syed Nawshad Xinghua Yu Wenhao Yu

[hhao004@ucla.edu](mailto:hhao04@ucla.edu) [snawshad@ucla.edu](mailto:snawshad@ucla.edu) [sungwa@ucla.edu](mailto:sungwa@ucla.edu) [yuwenhao@ucla.ed](mailto:yuwenhao@ucla.edu)u

***Abstract*—In this paper, we explored the classification of human motor imagery based on electroencephalography (EEG) data. The dataset we will be using for this project is BCI competition IV dataset 2a. Our goal for this project is implement 3 network architectures: a Convolutional Neural Network (CNN), a Variational Autoencoder (VAE) with a CNN afterwards to classify, and a CNN with Long-Short-Term Memory (LSTM) for both spatial and temporal features. In addition to these varying network architectures we explored the following preprocessing methodologies to help our networks train: wavelet packet decomposition (WPD) to extract signal information in frequency-time domain, cropped training that subsamples the original data along the time axis, and a method that combines max pooling, averaging and subsampling. Out of these, our best network architecture was the LSTM + CNN architecture with preprocessing for spatial features which achieved about 73% test accuracy followed by a regular CNN with preprocessing which achieved about 72% accuracy. The VAE+CNN with preprocessing was able to achieve varying accuracies from 67-72%.**

***Keywords— EEG; Motor Imaginary; CNN; LSTM; Neuro Engineering***

# Introduction

EEG based motor imagery classification has multiple applications, mainly in brain-computer interfaces (BCI) such as prosthetics. EEG’s are representations of the coordinated activity of millions of neurons near a non-invasive scalp-electrode and are measured by covering a patient’s scalp in these electrodes. The data retrieved via this process, however, has relatively poor spatiotemporal resolution. As a result, this type of data typically requires some preprocessing by either extracting the most useful pieces of information, filtering out the noise, or by simply compressing the data.

The BCI competition IV dataset used was taken from 9 individual subjects (patients). Each subject had 22 electrodes (22 channels) on their scalp that took data while the subject imagined certain movements, particularly with the right hand, left hand, feet, and tongue (4 classes).

In terms of network architecture, the best way, usually, to classify EEG data for motor imagery is with the use of neural networks. This paper will particularly focus on CNNs and the effect of adding LSTM layers, or pre-processing with a VAE, or filter from WPD. Generally, LSTM layers should be expected to boost accuracy of a model. LSTM allows the model to learn valuable spatial features independently from temporal features which lead to higher subject classification accuracy. Furthermore, the preprocessing methods such as VAE and filtering are expected to make it easier for the CNN to learn since the model would be working with filtered or less noisy data, thus yielding better accuracy.

# Implementation

In our work, we tested three different kinds of data preprocessing strategies aiming to get more data by doing augmentation or to decompose signals to improve accuracy of the EEG recognition. As we all know, a large dataset is one of the essentials for machine learning models, and can also prevent overfitting.

## LSTMs

LSTMs are a type of Recurrent Neural Network. Recurrent Neural Networks (RNN) are a special kind of neural network that is meant for time series data or sequences, like the EEG dataset we are using. These neural networks, unlike normal “feedforward” neural networks, take into account the order of a sequence. However, a significant setback of RNNs is that they cannot remember long-term dependencies and as a result have vanishing and exploding gradients causing no learning to happen. LSTMs fix this problem by holding information with long-term dependencies.

* 1. *VAEs*

A VAE is a “generative model”, where real image data is used as inputs and is encoded based on trainable image distribution parameters and then decoded to produce a generated replica as an output. The more accurate the generative model is, the closer the replica outputs are to the original data. This “generative model” can be useful for EEG imagery classification (or any image classification) by generating more data to mimic the features of real EEG image data, which can lead to larger, trainable sets of data and thus more accurate models.

## Cropped training

Lu creates new sample by sliding a window over the time axis[1]. They were inspired by Schirrmeister et al. [2], and proposed the idea called cropped training strategy. [x01] used a window size of 500x22 and they slid the window one step at a time. This created 500 new samples on every existing sample. The dataset ends up getting the dimension of (500\*2558, 500, 22, 1), which our machine cannot handle. We adopt the idea of sliding windows with different window sizes, 10 sliding units and window size of 490 in this case due to the machine memory issue.

## Combination of Max pooling, Averaging, Subsampling

Another augmentation is the combination of max pooling, averaging, subsample with noise, introduced by Tonmoy [3]. With the observation that channel signals at the second half of the time samples (500-1000) do not carry out much useful information, we trim the data on the time axis and only keep the first half (0-500). There are three processing techniques used after trimming. The first one is max pooling on every 2 time step, which gives us a 250 time sample dataset. The second technique is performing averaging on every 2 samples and adding Gaussian distribution noise to it. It also gives us a 250 time sample data. We last used subsampling with Gaussian distribution noise to get two new samples that also have 250 time samples. For example, the input of the processing is (N, 500, 22), the output for the whole processing, three techniques concatenated, will be (N+N+2N, 250, 22) [refer figure]TODO.

## Wavelet Packet Decomposition

If the information is encoded into the neural signal by varying the firing rate, theoretically, the encoded information could be extracted by analyzing the spectrum of the EEG. Wavelet transformation is a popular feature extraction method applied on the EEG. Compared to conventional Fourier Transform (FT) or Short-time Fourier Transform(STFT), it provides a more precise time-localized frequency component extraction which becomes helpful for our model [4]. For motor imagery application, the desired frequency band is 7-30Hz [5]. In our model, we used a low pass filter to remove high-frequency components from the samples and passed the filtered samples through a 3-level Wavelet Packet Decomposition (WPD) filter. We concatenate 0-31 Hz, 31-62 Hz, and 63-94 Hz together for our model. The figure 1 below shows the band decomposition of the WPD.

Diagram

Description automatically generated

Figure 1. 3 level EEG Wavelet Packet Decomposition

Another advantage of using Wavelet Decomposition is that when signals are decomposed into different signal bands, the resulting signal bandwidth of each band bin are only of the bandwidth of the original signal. n is the number of levels of the WPD. The number of samples to represent signals with our interested band could be reduced. The overall computation required by our model will be eased.

* 1. 2D Transformation

For CNN(Spatial) + LSTM(Temporal) model, the data are reshaped to reveal the location relationship between 22 channels [6]. The 22 channels data are mapped to a 6 x 7 2D matrix that is centered at channel 10. The areas that are not covered by the electrodes are filled with 0s. After the transformation, CNN layers are used to extract the spatial feature from the data.

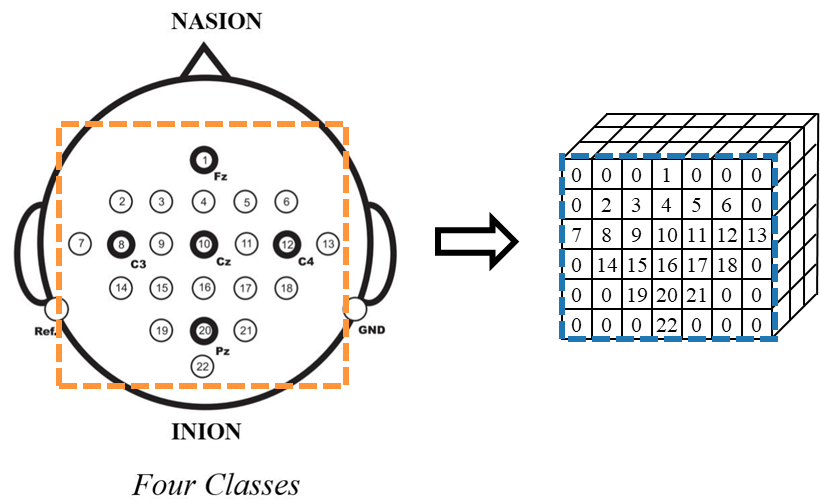


Figure 2 Left: Spatial distribution of electrode, Right: 3D representation of reshaped data (6x7x3)

# Results

For each of our models we had a test/validation split of about 85/15, with the training set having 6960 examples and the validation set having 1500 examples post processing. The test set had 1772 post processing examples. The test results of our implementations are shown below.

|  |  |  |
| --- | --- | --- |
| Architecture/Model | Test Accuracy | |
| With Preprocessing | Without Preprocessing |
| CNN | 72% | 68% |
| CNN+GRU | 73% | 57.8% |
| VAE+CNN | 70% | 53% |
| CNN+LSTM | 31% | N/A |

Table 1 The Classification Accuracy of Different Model

Our goal for each architecture was mostly achieved. Each implementation was learning and achieved either similar or better results than the pure CNN.

For the VAE, while the encoder and decoder were successfully built, certain sections of the VAE collapsed to 0. This is happening because of the way the loss function for the VAE is. The VAE can as a result “cheat” and set the distributions to 0.

figures of the confusion matrices

# Discussion

All preprocessing methods we tried seemed to be helpful for the classification. They have a 2-5% boost compared to the original data. For both [3] and cropped training, we augmented new samples. For WPD, we kept the same sample size but improved the quality of the data to achieve better accuracy. After we visualized the channel signals, we realized sliding a 490 size window sampling every unit for 10 units was similar to having the same sample copied 10 times. However, because we perform it on both training and test sets and achieve better results, it illustrates that the method does help.

We tried with pure Vanilla CNN first which gives us **72%** test accuracy with preprocessing and **68%** without preprocessing. There are 4 convolutional blocks with the same kernel size. Max Pooling on every block helps extract features, batch normalization reduces the variance of each batch and smoother the method of forward and back propagation. Dropout provides a more accurate subject classification result. With preprocessing following three blocks, CNN has better test accuracy than FC.

For the model CNN+LSTM, the input shape is 250×1×22, which means temporal samples is 250. The 2D convolution layer with 3 filters, kernel size (10,1) and activation function RELU also applied in this model RELU function is used as the activation function. Same padding is used in both convolution layer and avg pooling layer. The max-pooling layer is used to preserve the main features and reduce the number of parameters by extracting the maximum value in this region and replacing it with the maximum value. Features observed from the max-pooling layer are put into LSTM for further feature extraction. The architecture is using a LSTM layer with 50 LSTM. The fully connected layer can integrate features together and lower the impact of feature location on classification. A dropout layer is connected after the first dense layer to avoid overfitting; the dropout function can help model abandoned unused features . Finally, a dense layer with 4 neurons using activation function SoftMax to classify four tasks. By using CNN and LSTM, the aim is to make the most of time domain information in EEG data and extract more abstract time domain features by using LSTM [1].

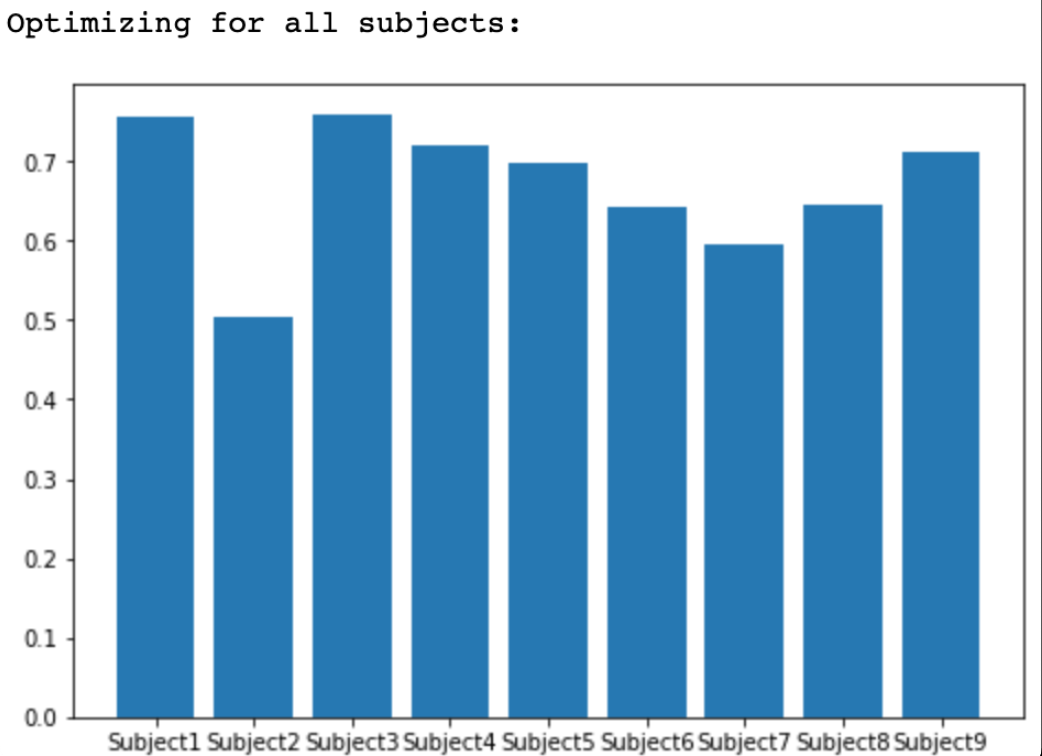
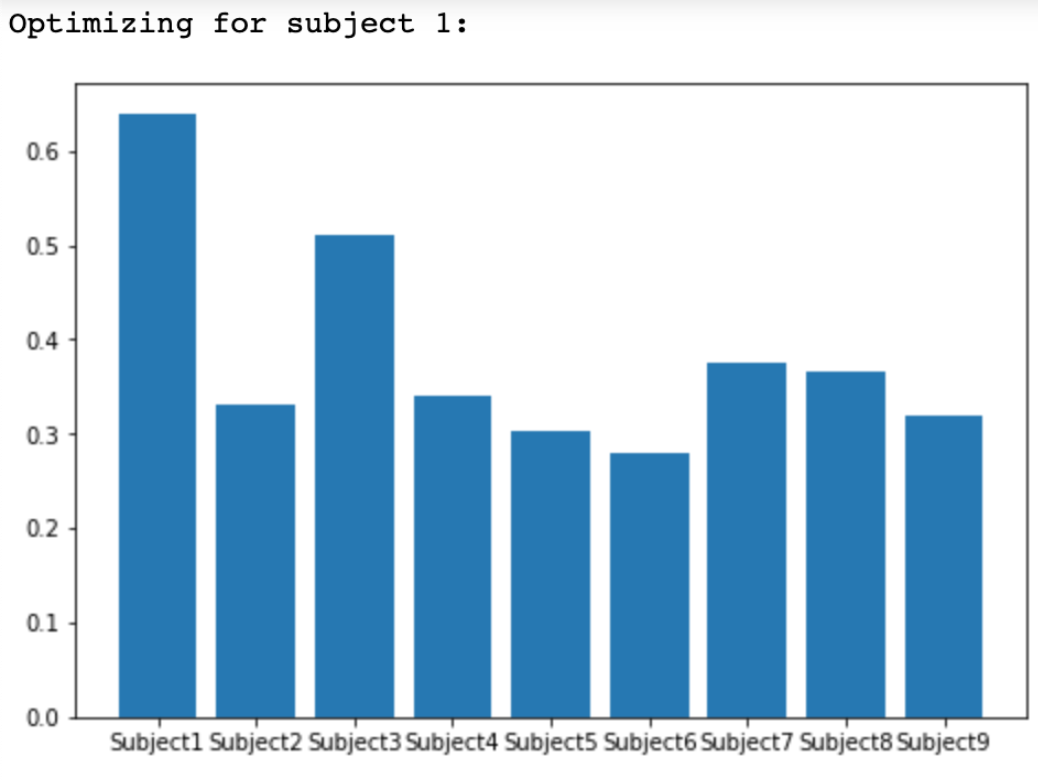
For the CNN and LSTM model the single subject accuracy can be higher than the overall accuracy. For example on subject 7, the overall accuracy for the model is 0.712 but for subject 7 Precision is 0.7159 and Recall equal to 0.7282. Based on the comparison between confusion tables we can tell that overall subjects

Our CNN(Spatial) + LSTM(Temporal) model is composed of 3 CNN blocks, 1 LSTM block and 1 SoftMax block. Each of the CNN blocks contains a conventional CNN layer, a max pooling layer, a normalization Bach layer and a dropout layer. CNN blocks extract the spatial feature from the collected data, and LSTM block extracts the temporal feature from the data. Then the output is generated by a SoftMax layer. We only got accuracy of 30% with the test which is far worse compared to the model using CNN to extract the temporal feature. It is because the motor imaginary data has long-term temporal features which requires more unit in LSTM to capture, whereas when multiple CNN layers are employed to capture those long-term features, each layer of CNN + max pooling would somehow capture localized features and send it to the next layer of CNN. The next layer would collect these features to extract higher level features in a bigger picture. By doing so repeatedly, we could capture the long-term features.

For the VAE-CNN model, approximately 70% was achieved with pre-processing and 53% without pre-processing. This is most likely due to the fact that the VAE generated more noisy data because the preprocessing used was able to remove the noise. As a result, it was more difficult for the network to learn.

While the preprocessing improved the VAE-CNN architecture significantly, it didn’t improve the accuracy of the pure CNN as much. This may be because of the collapse to zero for certain time periods of data mentioned above.

Training for an individual subject, subject 1, seemed to only reduce the accuracy of all the other subjects. The accuracy of subject 1 when optimizing for subject 1 was roughly 68% while the accuracy of subject 1 when optimizing for all subjects was about 75%. This may simply be due to the fact that there is more data available when training for all subjects.



When training for varying time periods

1. References

|  |  |
| --- | --- |
| [1] | P. Lu, N. Gao, Z. Lu, J. Yang, O. Bai and Q. Li, "Combined CNN and LSTM for Motor Imagery Classification," in *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2019. |
| [2] | R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard and T. Ball, "Deep learning with convolutional neural networks for brain mapping and decoding of movement-related information from the human EEG," *CoRR,* vol. abs/1703.05051, 2017. |
| [3] | T. Monsoor, "EE 247, Discussion 10," in *EE 247 Discussion 10, Annotated*, Los Angeles, 2022. |
| [4] | A. Echtioui, A. Mlaouah, W. Zouch, M. Ghorbel, C. Mhiri and H. Hamam, "A Novel Convolutional Neural Network Classification Approach of Motor-Imagery EEG Recording Based on Deep Learning," *Applied Sciences,* vol. 11, 2021. |
| [5] | J. S. Kirar and R. K. Agrawal, "Relevant Frequency Band Selection using Sequential Forward Feature Selection for Motor Imagery Brain Computer Interfaces," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2018. |
| [6] | T. Liu and D. Yang, "A Densely Connected Multi-Branch 3D Convolutional Neural Network for Motor Imagery EEG Decoding.," *Brain sciences,* vol. 11, no. 2, February 2021. |
| [7] | M. Dai, D. Zheng, R. Na, S. Wang and S. Zhang, "EEG Classification of Motor Imagery Using a Novel Deep Learning Framework.," *Sensors (Basel, Switzerland),* vol. 19, no. 3, January 2019. |