ELECTROENCEPHALOGRAPHY (EEG) SIGNAL PROCESSING AND EMOTION CLASSIFICATION USING DEEP LEARNING TECHNIQUES

By:

Ng Zheng Xun

Nanyang Technological University (NTU)

School of Computer Science and Engineering (SCSE)

2021

Abstract

Electroencephalography (EEG) signals traditionally require great skill and training to decipher and classify. However, with the rise of Machine Learning (ML) methods, and by applying Feature Extraction techniques such as the Fast Fourier Transform (FFT) and Continuous Wavelet Transform (CWT), ML models could aid in the classification of EEG signals. Hence, this project aims to investigate the different ML methods and signal processing techniques to build an ML model to classify EEG signals, as well as an implementation of the ML model for Emotion Classification.

Acknowledgements

I would like to thank my supervisor Associate Professor Wang Lipo from the School of Electrical and Electronic Engineering (EEE) for his mentorship, guidance and advice, as well as the C.N. Yang Scholars Programme (CNYSP) Office for the opportunity and their valuable support.

Table of Contents

Introduction

Motivation for Research

Electroencephalography (EEG) signals are measurements of electrical activity of the brain, by monitoring electrical impulses generated by neurons through electrodes achered to the scalp. EEG is one of the most popular and prevalent means to monitor brain activity, as it is relatively inexpensive [1] and non-invasive [2].

EEG signals are challenging to interpret, as they are individualistic, varying significantly between subjects [3], and contain noise artifacts inherent within the data collection process, from other muscles. As such, EEG signals have traditionally been interpreted by professional neurologists visually, a testament to the deep expertise and skill required for the task [4]. Such processes tend to be time-consuming, and inefficient. [5] A typical set of signals is shown in Figure 1-1.

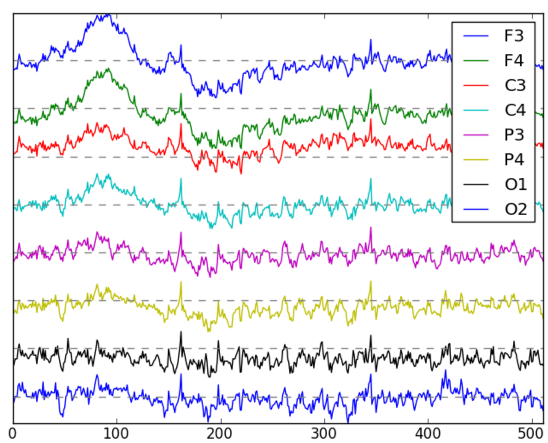


Figure 1-1: A typical set of signals obtained through Electroencephalogram (EEG)

In the recent decade, Machine Learning has been fruitfully applied to several neurophysiological signal classification problems, from Motor Imagery (MI) tasks to Emotion Classification problems [6].

Developments in this domain could pave the way for more efficient EEG analyses and better man-machine communication, where machinery and software could be controlled remotely through an individual's thoughts through Brain-Computer Interfaces (BCIs) [7].

Problem Statement

EEG signals exhibit significant non-stationarity and variability within or between users, as an individual's EEG signals vary significantly between runs. Furthermore, the lack of sufficient training data for Machine Learning (ML) models to learn from poses a significant challenge for ML classification. Hence, this project is an investigation into the feasibility of developing an EEG Emotion Classification model, by using the DEAP dataset for Emotion Classification in 4 classes - Arousal, Valence, Like/Dislike and Dominance.

Objectives

This project aims to investigate the feasibility of developing a Machine Learning model for EEG Emotion Classification, that incorporates deep learning techniques to mitigate the low-resource nature of EEG signals, as well as explore and implement methods to handle and preprocess EEG data.

Scope of Work

As part of the research project, several Machine Learning models were run, to classify 4 different types of emotions: Arousal, Valence, Like/Dislike and Dominance based on the DEAP dataset. Two types of Feature Extraction tools were used: the Fast Fourier Transform (FFT) and the Continuous Wavelet Transform (CWT), and their results on the Emotion Classification Tasks were compared.

Organisation of Report

The report is organised as follows:

Chapter 1, which is this chapter, presents an introduction to the problem statement as well as the context of the research conducted. It includes the research motivation, objectives and scope, as well as the organisation of the report.

Chapter 2 summarises findings from the literature review for related existing literature in the field of EEG, from a medical and neuroscience perspective, followed by a review of Deep Learning methods applied to EEG, for classification and/or feature extraction purposes.

Chapter 3 explains the research methodology used for the research project, including details of the independent and dependent parameters. Explanations for the techniques and algorithms used would also be provided.

Chapter 4 presents the details of the simulation and experimental model setups, along with the hyperparameters used.

Chapter 5 presents and discusses results obtained from the models running the FFT and CWT feature extraction algorithms, as well as a comparison with other State-of-the-Art (SOTA) models.

Chapter 6 concludes the report, and provides discussion for future work in the field of EEG signal processing using Deep Learning techniques to mitigate the non-stationarity of data. Other methods for the processing of EEG signals would also be discussed.

Literature Review

History and Development

One of the most prevalent and popular methods to study the human brain today, Electroencephalography (EEG) was first developed in 1890 by physiologist Adolf Beck as a method to investigate the spontaneous electrical activity of the brain [12]. It was first tested on animals, such as rabbits and dogs, to great success. Eventually, it was implemented on human subjects in 1924, and having been recognised as "one of the most surprising, remarkable, and momentous developments in the history of clinical neurology" [13] , EEG had since taken off as an ubiquitous and valuable tool to study the human brain.

EEG is a versatile tool, having been creatively implemented in numerous industries and applications, from diagnosing epilepsy, to the teleoperation of remote devices. [14] [15]. In recent years, developing emotion recognition systems based on EEG signals have become a popular research topic among cognitive scientists.

Benefits and Shortcomings of EEG

The ubiquity and proliferation of EEG technology is a direct result of the many benefits that it brings. Firstly, due to the great speed that electrical signals propagate, EEG has an excellent time resolution, allowing one to detect activity within cortical areas - even at sub-second timescales [16]. The sampling rate of an EEG setup, defined as the number of recordings taken per second, is impressively broad in scope, up to 20,000Hz if needed. [17] This far exceeds other neuroscience techniques, especially those that are non-invasive in nature. [18]

A significant drawback of monitoring EEG signals is their low signal to noise ratio. [22], as the brain activity measured is often buried under multiple sources of environmental, physiological and activity-specific noise of similar or greater amplitude called 'artifacts'. Furthermore, as alluded to previously, EEG signals exhibit non-stationarity [23], and are highly user and domain specific, and as such, most current BCI systems are calibrated specifically for each user. [22] [7] This poses a challenge for large-scale data analytics applications, where data could be generalised to extract useful insights. [24]

Ultimately, despite its shortcomings, EEG still continues to be an extremely robust and valuable tool for research and signal collection.

Deep Learning Classification of EEG Signals

EEG signals have been applied in a wide variety of tasks. It is common to see researchers use EEG signals to aid in tasks relating to Emotion Recognition, Motor Imagery, and seizure detection. However, this list is far from exhaustive, as researchers have been developing new and innovative methods to better classify EEG signals while mitigating the difficulty of a lack of training data.

In recent years, researchers have utilised several types of ML models to handle EEG data, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The use of CNNs in EEG classification could be traced to their fruitful usage in computer vision tasks, and are hence borrowed for EEG tasks as well. [26]

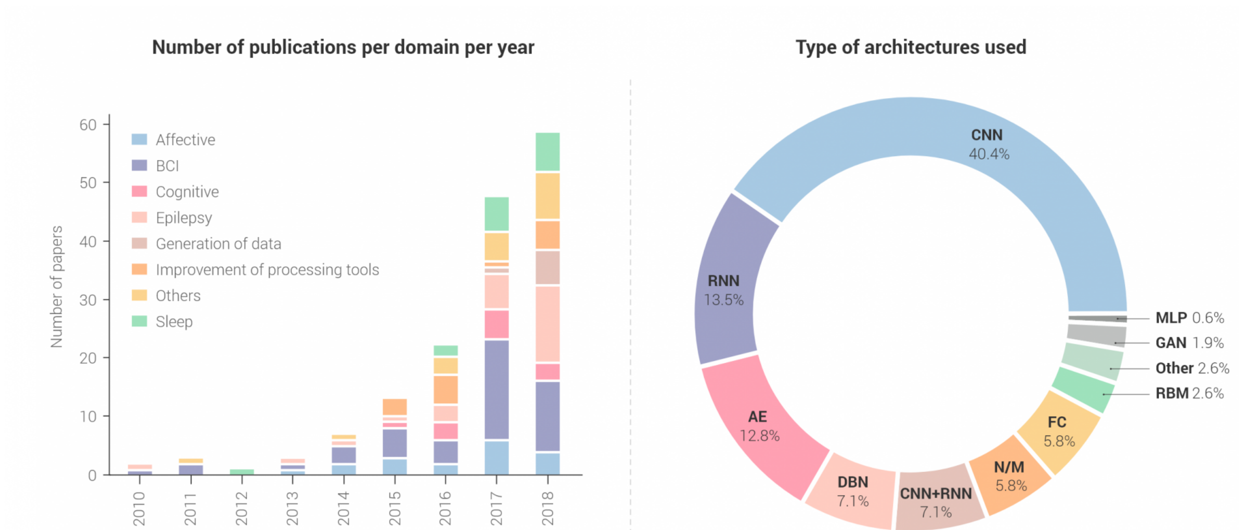


Figure 2-3: A graphical summary of the application of ML techniques to various EEG tasks (left), as well as an overview of the general types of architectures used in the field (right). Source: Roy, 2019

Research Methodology

Electroencephalography (EEG) is one of the most common tools used for monitoring brain activities, through the measurement of electrical signals of neurons in the brain as they fire. Several techniques are applied throughout the entire project pipeline, as shown in Figure 3-1.

Raw EEG Signals → Pre-Processing → Feature Extraction through FFT or CWT→ Neural Network Learning → Neural Network Training → Classification

Figure 3-1: A schematic representation of the data and Machine Learning pipeline

Raw EEG Signal Preprocessing

After raw EEG data has been collected, signal artifacts such as DC offsets and drifts, and electromagnetic noise need to be filtered out. [32] These often involve the application of band pass filters, such as the high-pass or low-pass filters. These filters remove specific components of the signals, based on their frequencies and associated drifts. [33] For example, most EEG signals above 90 Hz are usually irrelevant to data analyses, and can be filtered out using high-pass filters [34]. An implementation of a high-pass filter could be seen in Figure 3-2.

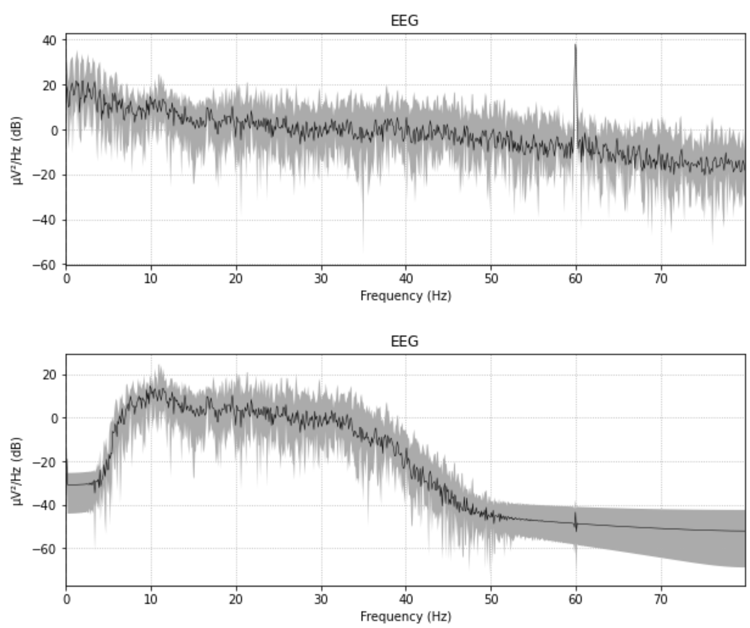


Figure 3-2: An implementation of a high-pass filter on a raw EEG signal (above) into the preprocessed signal (below)

Fourier Transformation and Fast Fourier Transform (FFT)

One of the techniques commonly applied to extract features from EEG signals is the Fourier Transformation, as shown in Fig 3-3. It is an elegant algorithm to transform a signal from its time-domain to its frequency domain. [36]

$$\mathrm{F(x)} = \sum_{n=0}^{N-1} f(n) e^{-j2\pi(x\frac{n}{N})}$$

$$\mathrm{f(n)} = \frac{1}{N} \sum_{n=0}^{N-1} F(x) e^{j2\pi(x\frac{n}{N})}$$

Figure 3-3: The pair of equations for the Fourier Transformation

By applying the Fourier transformation on a waveform such as an EEG signal, the signal could be decomposed and separated into its constituent frequencies, for which it is a superposition of, as the Fourier transform returns the amplitudes and frequencies of the component sine waves, given any periodic function. [37]

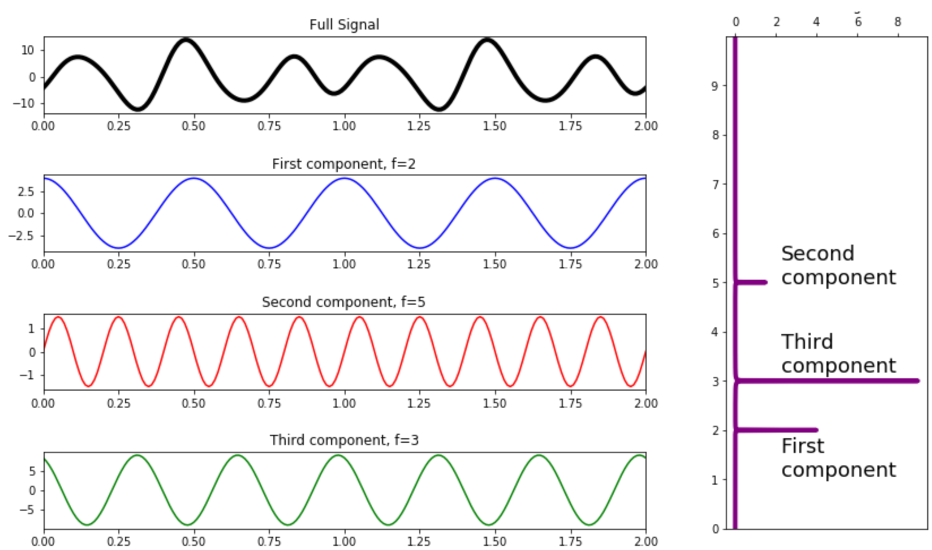


Figure 3-4: An illustration of the Fourier Transform in action - the original signal (in black) was decomposed by the Fourier Transform into its three components (in blue, green and red). A graphical interpretation is also provided (right)

The Fast Fourier Transform (FFT) is a refinement of the Fourier transformation, whereby the Fourier transform of the signal is taken over short-windows of the signal, which, as the name implies, reduces the time needed for the transformation. [37]

However, although FFT has a high resolution in the frequency domain, it has zero resolution in the time domain. So while FFT gives insights into what frequencies are present in a particular signal, it does not shed any light on when these signals occur relative to each other. However, in EEG processing, it can be said that both frequency and time-domain characteristics of EEG are very important.

Continuous Wavelet Transform (CWT)

Another method to analyse EEG signals is through the Continuous Wavelet Transform (CWT). The CWT algorithm centers around the concept that a normal signal could be expressed as linearly combined basis functions of a specific type known as wavelets. These wavelets are arrived at by shifting and expanding a single function known as the mother wavelet. CWT could be used as an estimation technique for expressing the general function given as an infinite set of wavelets, and there are many different types and families of wavelets, as shown in Figure 3-6. On the basis of the input signal *x(t)*, the CWT algorithm could be expressed as

$$cwt(\tau, s) = \frac{1}{\sqrt{|s|}} \int_{-\infty}^{+\infty} x(t)\psi\left(\frac{t-\tau}{s}\right) dt$$

$$\tau = translation, s = scale, \Psi(t) = mother\ wavelet, \left(\frac{t-\tau}{s}\right) = sale\ factor$$

Figure 3-5: The equation for the Continuous Wavelet Transformation

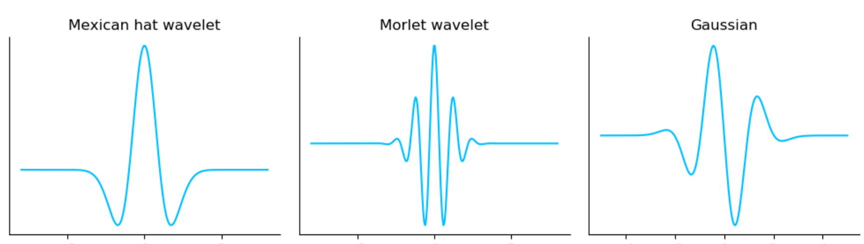


Figure 3-6: Several common types of mother wavelets used for CWT

The results of a CWT on a signal could be visualised in the form of a scalogram, such as the one shown in Figure 3-7. A scalogram indicates where most of the energy (from the color bar right of the scalogram) of the original signal is contained in time and frequency. Furthermore, we can see that the characteristics of the signal are now displayed in highly resolved detail. Each horizontal characteristic in the scalogram can be interpreted as frequencies of the total signal. The fact of not seeing a continuous line in the figure corresponds to the fact that said frequencies are not continuous in time. Such a visualization of the CWT coefficients like the 2D scalogram can be used to improve the distinction between varying types of a signal. Such a scaleogram can not only be used to better understand the dynamical behavior of a system, but it can also be used to distinguish different types of signals produced by a system from each other.

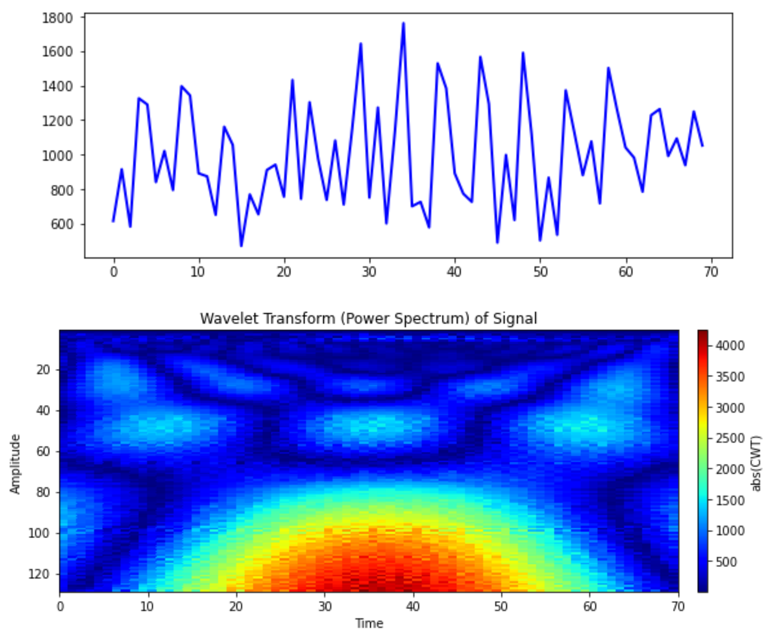


Figure 3-7: A scalogram (below) of the CWT of an EEG signal (above)

K-fold Cross Validation

Cross-validation is the cornerstone of evaluating ML models. When evaluating a machine learning model, it is important to measure how well it generalises to new and unseen data. In order to do this, it is common practice to divide the available data into training and test sets, and cross-validation dictates that there shall be no overlap between the train and test datasets. [43]

K-fold Cross Validation is an improvement upon the usual train-test independence, as the latter might be incomplete and misleading under certain circumstances, such as a lack of train and test data. [44]

Using K-fold, the original dataset is split into k equally-sized sets. The model is then trained k-times. With each time of training, one of the k sets is isolated to be the test set. Finally, the overall performance is computed as the average over the k single performances. This process is summarised in Figure 3-11. This methodology allows ML practitioners to get a more accurate estimate of the performance of a model, and ensures that the model is not performing wildly differently after being trained on different segments of labeled data, which could be the case when there is a lack of data available. [45]

There are several drawbacks with using k-fold cross validation, namely that it is computationally expensive, as there is a need to repeatedly carry out training several times. However, it is also widely acknowledged that cross-validation is essential for evaluating the performance of the learning model. [46]

Figure 3-11: A schematic summary of k-fold cross validation

Methods and Model Setup

The research project was undertaken to explore the different stages of the EEG ML classification pipeline, as well as the various tools and processes available to do so. Numerous libraries were used in the implementation of this project, ranging from Machine Learning purposes to data manipulations and data visualisations. Namely, software such as the MNE library [47] and the PyPI Scaleogram packages [48] were utilised to preprocess the datasets, as well as to generate the visualisations showcased in this chapter and the previous chapter. Effective techniques such as k-fold cross validation are applied too. The default Adam optimizer is used for the baseline model, as well as the default Categorical Cross-Entropy loss function.

DEAP Dataset

In this research project, the DEAP Dataset: a dataset for emotion analysis using EEG, Physiological and video signals was used. [50] It was collected from 32 subjects, and consists of 40 recordings, each with 4 classes - arousal, valence, dominance, and liking. These classes were obtained from the participants after they had been shown a one-minute music video, and were graded on a scale of 1 to 9. Hence, if the grade was greater than 4.5 then the arousal/valence label is high, if the grade is less than 4.5 then the arousal/valence label is low [50]

The 40-channel EEG signals were originally sampled at 512Hz, but were downsampled to 128Hz for data analysis purposes. Any Electrooculogram (EOG) artifacts were manually removed by the original researchers. [50] The data was also band-pass frequency filtered to 4.0-45.0Hz, and averaged to the common reference. [50] For the project, the models would be evaluated on their accuracies in classifying the 4 categories of valence, arousal, like/dislike and dominance.

CNN Model with FFT Feature Extraction

The FFT feature extraction model was based on the Tensorflow's CNN model [51], with the Keras library for One-Hot Encoding [52, p.], while scikit-learn's KFold was used for the implementation of k-fold cross validation [53]. A subset of the DEAP Dataset was used for training and testing, whereby only 3 subjects's raw EEG data (Subjects 01, 02 and 03) are used in the implementation, in light of limited computational resources, as well as due to time considerations. The list of hyperparameters used are listed below. A schematic representation of the FFT CNN is shown in Figure 4-12.

hyperparameter	chosen values
Batch Size	256
Number of Classes	10
Number of Epochs for Training	200
Input Tensor Shape	(70,1)
Number of folds for k-fold cross-validation	5
Loss Function	Categorical Cross Entropy
Optimiser	Adam

Table 4-1: A list of hyperparameters after hyperparameter tuning for FFT CNN Model

As with every ML model, hyperparameter tuning is a necessity to ensure that the final model produces satisfactory results while reducing the likelihood of overfitting or underfitting. The precise nature and approach to tune and adjust the hyperparameters of a model is as much of an art as it is a science.

The raw EEG data were first preprocessed and underwent feature extraction by the FFT algorithm. The 51k pairs of processed data and labels were then split into train-test sets by a 80-20 ratio, and one-hot encoding was applied to the labels. Standard Scalar was applied to normalise the data, for better accuracy. [54]

Max-pooling was implemented for the convolutional sections [42], and the Rectified Linear Unit (ReLU) activation function was used for the Dense layers. [55] Several batch normalisation and dropout layers were inserted to prevent overfitting. For the final classification layer, the softmax activation function was used, to output a probability estimate of each class.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 70, 128)           512
_________________________________________________________________
batch_normalization (BatchNo (None, 70, 128)           512
_________________________________________________________________
max_pooling1d (MaxPooling1D) (None, 35, 128)           0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 35, 128)           49280
_________________________________________________________________
batch_normalization_1 (Batch (None, 35, 128)           512
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 17, 128)           0
_________________________________________________________________
flatten (Flatten)            (None, 2176)              0
_________________________________________________________________
dense (Dense)                (None, 64)                139328
_________________________________________________________________
dropout (Dropout)            (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 32)                2080
_________________________________________________________________
dropout_1 (Dropout)          (None, 32)                0
_________________________________________________________________
dense_2 (Dense)              (None, 16)                528
_________________________________________________________________
dropout_2 (Dropout)          (None, 16)                0
_________________________________________________________________
dense_3 (Dense)              (None, 10)                170
=================================================================
Total params: 192,922
Trainable params: 192,410
Non-trainable params: 512
```

Figure 4-12: A schematic representation of the FFT CNN Model used

CNN Model with CWT Feature Extraction

The CWT Model utilises the CWT algorithm from PyWavelets. [56]. This method takes as input the signal, the mother wavelet to be used and a list of scales in which the signal should be examined. The mother wavelet used for the project is the "Morlet" wavelet, which could be described mathematically in (1) and graphically in Figure 4-13.

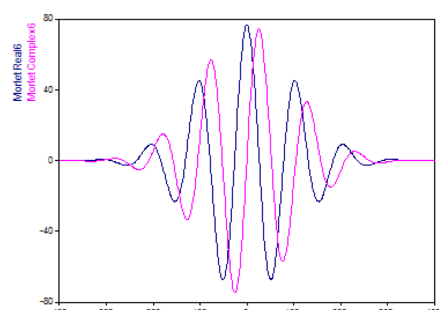


Figure 4-13: A graphical representation of the "Morlet" wavelet used as the mother wavelet for CWT feature extraction

$$\psi(t) = C\sigma\pi-0.25e-0.5t2(ej\sigma t-K\sigma) \quad (1)$$

Similar to the FFT model, the CWT model is implemented with methods such as One-Hot Encoding, Standard Scalar Normalisation, and k-fold cross validation, which are detailed earlier, with slight changes. The hyperparameters were also similar to that of the FFT model, with the exception that the Input Tensor Shape has been changed to (96, 69, 69, 3), by virtue of the CWT algorithm which outputs a scalogram. The batch size was also decreased to 32. A schematic representation of the CWT model architecture implemented is shown in Figure 4-14.

The model architecture has been reworked to better accommodate and to produce better results for the DEAP Dataset. Namely, the number of dropout layers have been reduced, as well as the Batch Normalisation Layer, to prevent large spikes and volatility in the Validation Loss. The implementation of the CWT algorithm was based on an open-sourced Github repository.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 67, 67, 69)        1932
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 33, 33, 69)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 31, 31, 138)       85836
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 15, 15, 138)       0
_________________________________________________________________
flatten (Flatten)            (None, 31050)             0
_________________________________________________________________
dense (Dense)                (None, 138)               4285038
_________________________________________________________________
dense_1 (Dense)              (None, 10)                1390
=================================================================
Total params: 4,374,196
Trainable params: 4,374,196
Non-trainable params: 0
_________________________________________________________________
```

Figure 4-14: A schematic representation of the FFT CNN Model used

Results and Discussion

This section of the report describes the results of both the FFT CNN model and the modified model with the implementation of feature extraction via CWT, on the classification of the Arousal, Valence and Like/Dislike Classes. Due to time constraints, the Dominance Class was unable to be tested.

Analysis of FFT CNN Model

The CNN model with FFT Feature Extraction was trained on 200 epochs with k-fold cross validation (k=5), with confirmation that the model converged. Table 5-2 describes the performance of the model when classifying the Valence class, over 5 folds, and the corresponding confusion matrix for the Valence class is shown in Figure 5-16. Furthermore, a pair of training and testing accuracy and loss curves from the model during one of the folds are shown in Figure 5-15.

The training and loss curves for each fold and class, as well as their corresponding confusion matrices, are supplemented in the various appendixes.

classes	train dataset		test dataset	
	cross-validated accuracy	cross-validated loss	cross-validated accuracy	cross-validated loss
arousal	80.1%	0.595	79.3%	0.635
valence	78.6%	0.635	75.9%	0.700
like/dislike	82.2%	0.506	80.7%	0.597

Table 5-2: Summary of Classification Results for FFT CNN Model

From the results, it could be seen that the FFT model produces decent results, with the accuracy values significantly above chance level, with loss values that are under 1. This indicates that the FFT model generalises pretty well to unseen data too, as the train and test results are comparable in quality. Between the 4 classes, the performance of the FFT model is rather steady, with the Like/Dislike class producing the best test accuracy results of 80.7%.

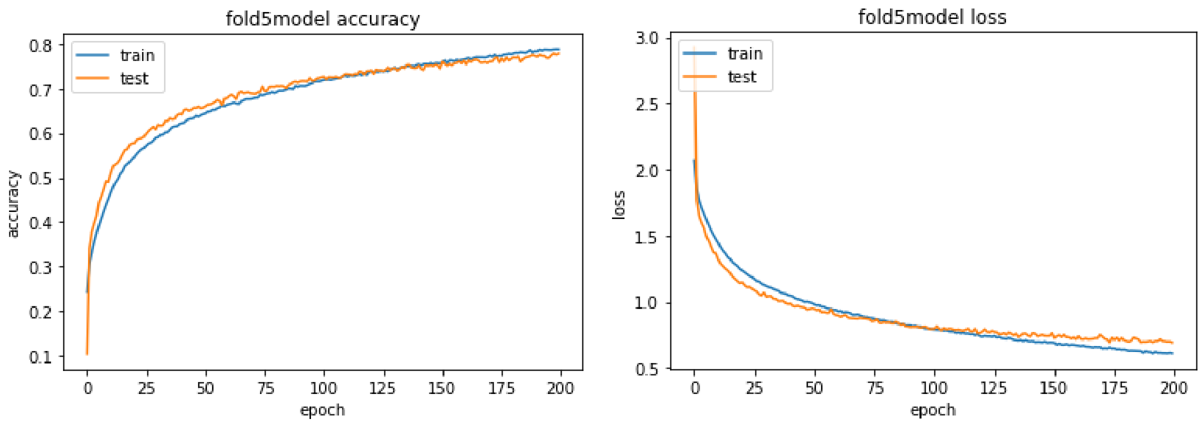


Figure 5-15: Accuracy (left) and Loss (right) Plots of the FFT CNN Model

	train dataset		test dataset	
fold number	accuracy	loss	accuracy	loss
1	78.5%	0.618	65.3%	0.786
2	82.5%	0.590	83.8%	0.573
3	75.8%	0.624	77.1%	0.699
4	77.2%	0.706	76.5%	0.714
5	79.2%	0.695	77.8%	0.693
mean	78.6%	0.635	75.9%	0.700

Table 5-3: The Accuracy and Loss Values of the FFT CNN Model on Valence Datasets (2sf)

It can be seen from Table 2 that the FFT algorithm produces respectable results when applied to the domain of EEG Signal Processing, as the model does fairly well on both the train and test sets, with accuracies significantly above chance level. This shows that the model has generalised well to the data. Furthermore, from the loss plots in Figure 5-15, there is little overfitting in the model as both train and test validation loss curves converge. This could be attributed to the presence of dropout layers in the model setup, as well as the hyperparameter tuning process.

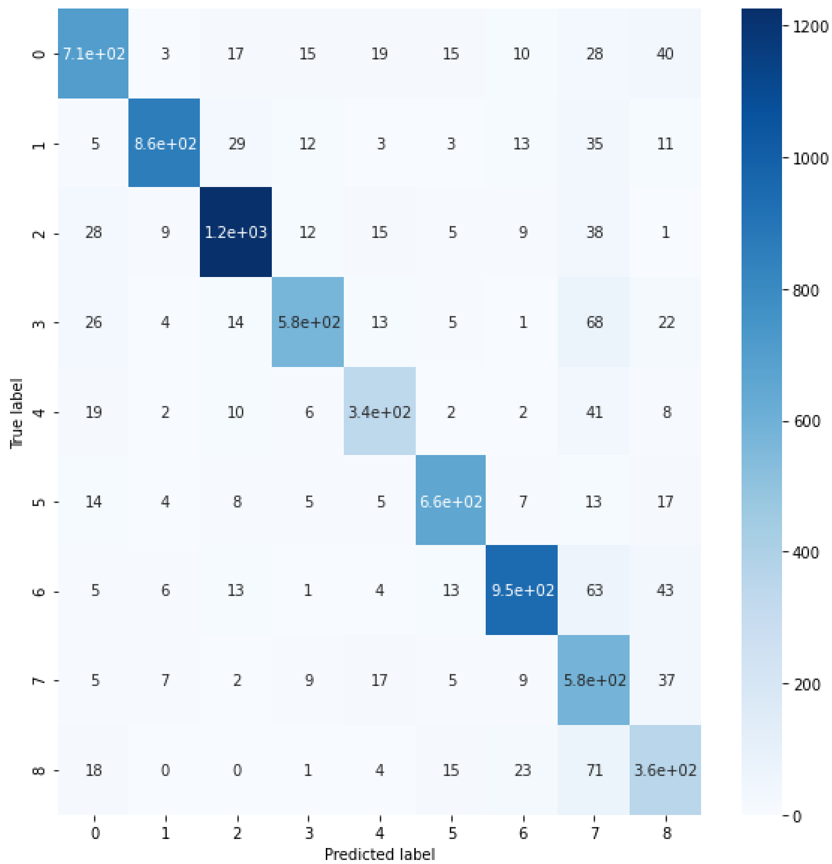


Figure 5-16: The confusion matrix of a fold predicting the Valence class

It could be seen from the confusion matrix that the FFT model performs well on classification too, with the majority of EEG signals being classified correctly, as visualised along the diagonal. There are a few false positives and false negatives, but they are in the minority, which could also be seen by the relatively lighter shades outside of the diagonal.

Analysis of the CWT CNN Model

Similar to the FFT CNN model, the CNN model with CWT Feature Extraction was trained on 200 epochs too. Similarly, Table 5-4 describes the performance of the model, over 5 folds and Figure 5-17 shows a pair of training and testing accuracy and loss curves from the model. Due to time constraints, the CWT model was not able to undergo k-fold cross validation.

The training and loss curves for the CWT model for each class are supplemented in the appendix.

	train dataset		test dataset	
classes	accuracy	loss	accuracy	loss
arousal	97.9%	0.0612	63.8%	9.72
valence	97.2%	0.0551	62.9%	8.38
like/dislike	95.6%	0.265	66.5%	9.62

Table 5-4: Summary of Classification Results for CWT CNN Model (3sf)

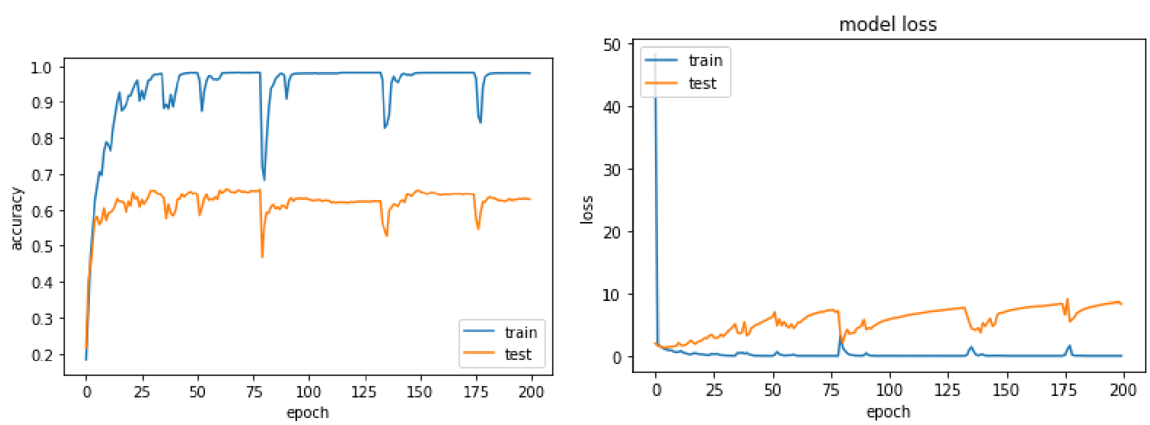


Figure 5-17: Accuracy (left) and Loss (right) Plots of the CWT CNN Model

It can be seen that the CWT model generally produces decent results with both train and test accuracies above chance level, as well as impressive train accuracies and loss. The Like/Dislike class exhibits the best results, with a 66.5% test accuracy, and an impressive 95.6% train accuracy.

However, it is worthy of note that the model exhibits high levels of validation loss, suggesting that the CWT model is overfitted on the training data. The loss graphs corroborate this finding, with the validation loss diverging from the training loss with increasing epochs.

Comparison between FFT and CWT Models

The results from the FFT and CWT models are shown in Table 5-5. As seen, the FFT model performs better than the CWT model on every emotion class of the DEAP dataset, with the average test accuracy being 78%, while the CWT model has an average test accuracy of 65%.

Between the 3 different classes of emotions, it is noteworthy that both the FFT and CWT models have their best results on the Like/Dislike class, followed by the Arousal and Valence classes. This could suggest that there is a higher correlation between the feelings of like and dislike with an individual's EEG signals frequencies, compared to other types of emotions such as arousal.

	test accuracy	
classes	FFT Model	CWT Model
arousal	79.3%	63.8%
valence	75.9%	62.9%
like/dislike	80.7%	66.5%

Table 5-5: Summarised Results from the FFT and CWT Models

Comparison with SOTA Models

A comparison between the FFT and CWT models with SOTA models is done. Namely, with S Alhagry et al.'s LSTM Recurrent Neural Network [57], Choi et al.'s LSTM model [58], as well as Naser et al.'s model which utilised the dual-tree complex wavelet packet transform

(DT-CWPT). [59] All the datasets utilised the DEAP dataset. The results are summarised in Table 5-6.

classes/models	arousal	valence	liking
Choi et al. [58]	74.65%	78%	-
Naser et al. [59]	66.2%	64.3%	70.2%
S Alhagry et al. [57]	85.65%	85.45%	87.9%
FFT CNN Model	79.3%	75.9%	80.7%
CWT CNN Model	63.8%	62.9%	66.5%

Table 5-6: Accuracy Comparison with SOTA Models (3sf)

It can be seen that while the CWT CNN model does not perform as well compared to the other SOTA models, it is still relatively comparable to the LSTM model of Naser et al. The FFT CNN model, on the other hand, compares favourably beside other SOTA models, even exceeding several benchmarks such as the model by Naser et al on classes such as valence. This shows that the FFT model had indeed generalised very well to the EEG data.

Conclusions and Recommendations

Firstly, it can be seen that the CWT CNN model produces poorer results, with an average test accuracy of 65%, compared to the FFT CNN model, which consistently achieves an average accuracy on the test dataset around 70%. However, this is still encouraging on several fronts, namely the efficacy of the usage of Feature Extraction algorithms to process EEG signals for Emotion Classification, as well as the results produced being comparable to other SOTA models, as shown in Table X.

It is noteworthy that by the nature of EEG signals, which vary in time and frequencies, the CWT feature extraction algorithm would be theoretically more suitable for such a task, as we need a tool that has high resolution in the frequency domain and also in the time domain, that allows us to know at which frequencies the signal oscillates, and at which time these oscillations occur, conditions which the CWT fulfils. However, the experimental results from the project suggests that the FFT feature extraction algorithm would be more suited for the task of Emotion Classification, while the CWT algorithm might prove more useful in other domains, such as ECG classification, or Epilepsy Detection.

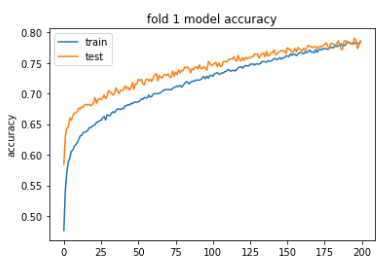
In hindsight, given more datasets and with more hyper parameter tuning, the results and mappings will definitely improve by a good margin. Such models and processes were not implemented and investigated in this project due to the limitations of time and circumstance, but it nonetheless shows potential to have an improvement in results.

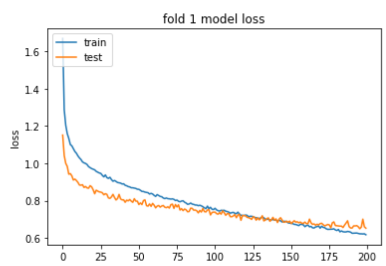
Looking ahead, further work could be done investigating how different signal processing and feature extraction methods such as FFT and CWT could be utilised to mitigate the concerns of non-stationarity of data, as well as the concerns of privacy. Techniques such as Transfer learning and Federated Learning could be implemented and investigated as well.

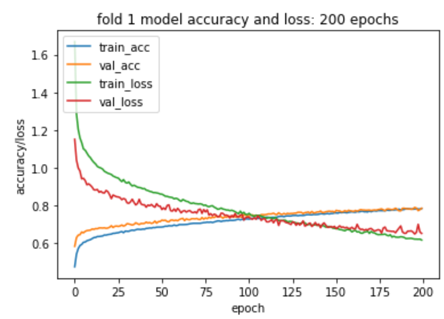
Appendix A: Accuracy and Loss Curves for FFT Model for Valence Class

		train dataset		test dataset	
class	fold number	accuracy	loss	accuracy	loss
valence	1	78.5%	61.8%	65.3%	78.6%
	2	82.5%	59.0%	83.8%	57.3%
	3	75.8%	62.4%	77.1%	69.9%
	4	77.2%	70.6%	76.5%	71.4%
	5	79.2%	69.5%	77.8%	69.3%
	mean	78.6%	63.5%	75.9%	70.0%

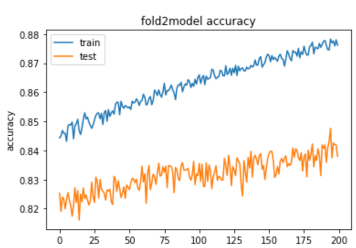
Fold 1:

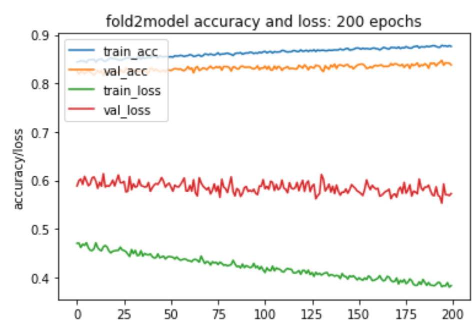




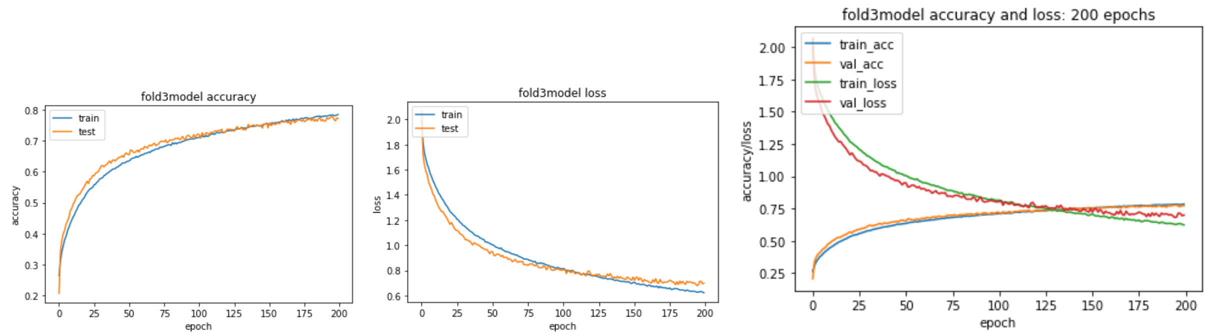


Fold 2:

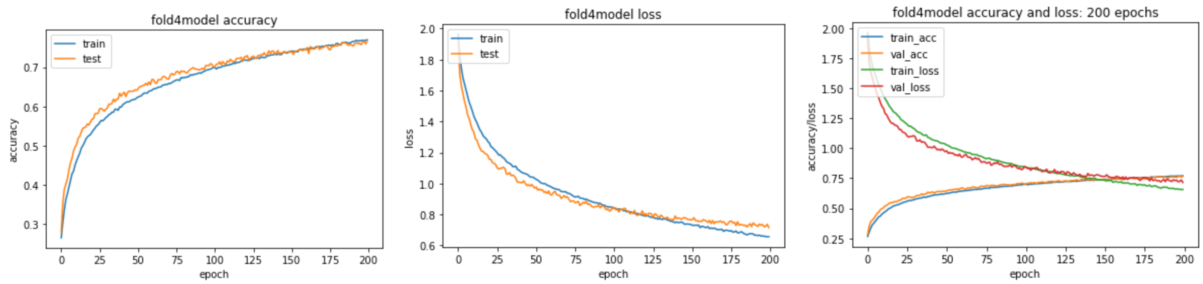




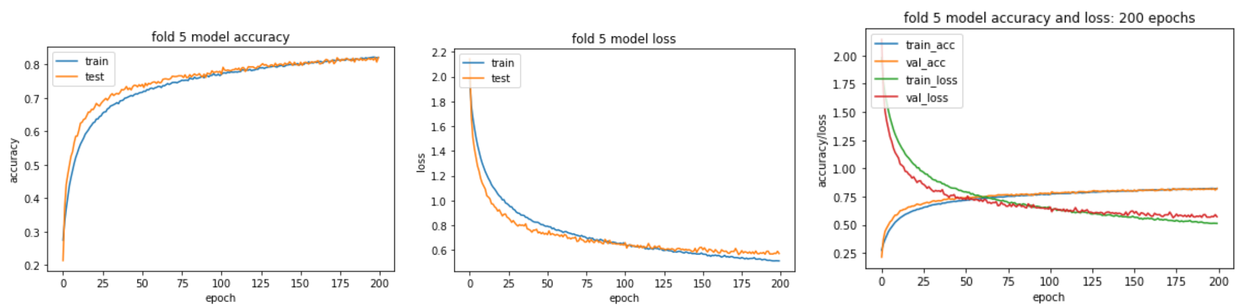
Fold 3:


fold3model accuracy
train
test
accuracy
epoch
fold3model loss
train
test
loss
epoch
fold3model accuracy and loss: 200 epochs
train_acc
val_acc
train_loss
val_loss
accuracy/loss
epoch

Fold 4:

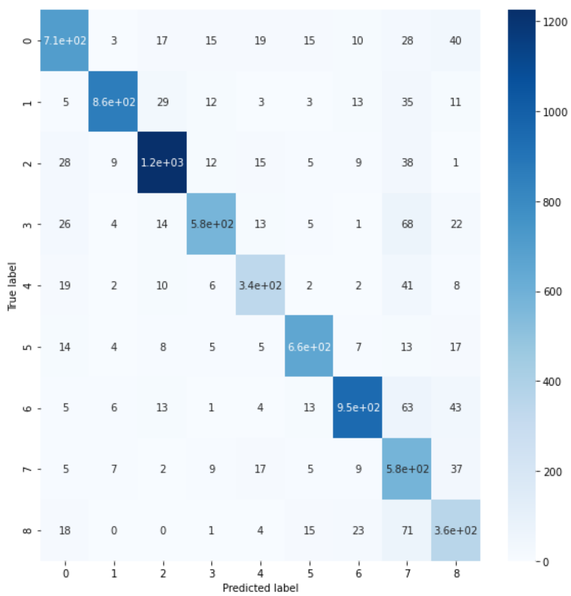

fold4model accuracy
train
test
accuracy
epoch
fold4model loss
train
test
loss
epoch
fold4model accuracy and loss: 200 epochs
train_acc
val_acc
train_loss
val_loss
accuracy/loss
epoch

Fold 5:

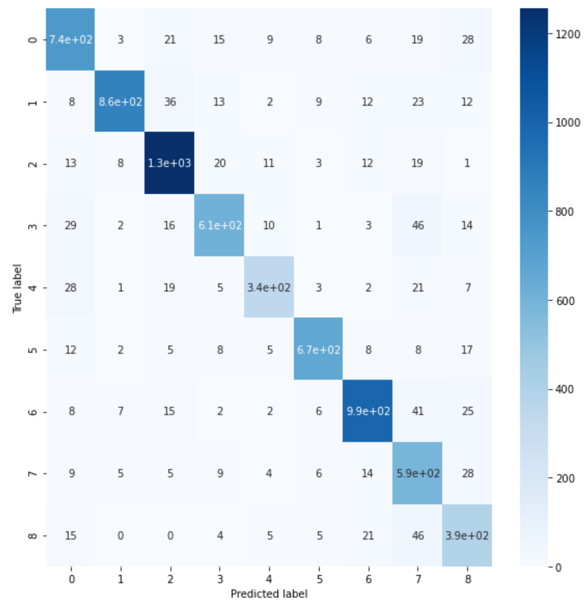

fold 5 model accuracy
train
test
accuracy
epoch
fold 5 model loss
train
test
loss
epoch
fold 5 model accuracy and loss: 200 epochs
train_acc
val_acc
train_loss
val_loss
accuracy/loss
epoch

Appendix B: Selected FFT Confusion Matrices for the Valence Class

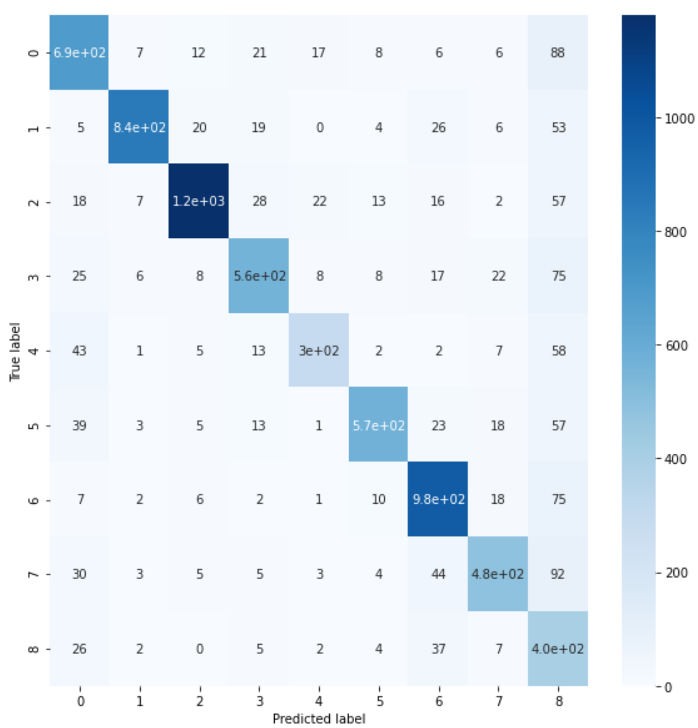
Fold 1:



Fold 2:



Fold 5:



References

[1] "Frontiers | A Systemic Review of Available Low-Cost EEG Headsets Used for Drowsiness Detection | Frontiers in Neuroinformatics." https://www.frontiersin.org/articles/10.3389/fninf.2020.553352/full (accessed Apr. 15, 2021).

[2] "Frontiers | How capable is non-invasive EEG data of predicting the next movement? A mini review | Human Neuroscience." https://www.frontiersin.org/articles/10.3389/fnhum.2013.00124/full (accessed Apr. 15, 2021).

[3] "THE NONLINEAR AND NONSTATIONARY PROPERTIES IN EEG SIGNALS: PROBING THE COMPLEX FLUCTUATIONS BY HILBERT–HUANG TRANSFORM | Advances in Adaptive Data Analysis." https://www.worldscientific.com/doi/abs/10.1142/S1793536909000199 (accessed Apr. 15, 2021).

[4] "2nd Bologna EPIPED-EEG Course: EEG Interpretation in Pediatric Epilepsies // International League Against Epilepsy." https://www.ilae.org/congresses/2nd-bologna-epiped-eeg-course-eeg-interpretation-in-pediatric-epilepsies (accessed Apr. 15, 2021).

[5] "How Deep Learning is changing machine learning AI in EEG data processing," *Bitbrain*, Apr. 23, 2020. https://www.bitbrain.com/blog/ai-eeg-data-processing (accessed Apr. 15, 2021).

[6] "EEG | Papers With Code." https://paperswithcode.com/task/eeg (accessed Apr. 15, 2021).

[7] "A review of artificial intelligence for EEG‐based brain−computer interfaces and applications - Zehong Cao, 2020." https://journals.sagepub.com/doi/full/10.26599/BSA.2020.9050017 (accessed Apr. 15, 2021).

[8] M. Rashid *et al.*, "Current Status, Challenges, and Possible Solutions of EEG-Based Brain-Computer Interface: A Comprehensive Review," *Front. Neurorobotics*, vol. 14, 2020, doi: 10.3389/fnbot.2020.00025.

[9] O. Landau, A. Cohen, S. Gordon, and N. Nissim, "Mind your privacy: Privacy leakage through BCI applications using machine learning methods," *Knowl.-Based Syst.*, vol. 198, p. 105932, Jun. 2020, doi: 10.1016/j.knosys.2020.105932.

[10] M. Hassani and M. R. Karami, "Noise Estimation in Electroencephalogram Signal by Using Volterra Series Coefficients," *J. Med. Signals Sens.*, vol. 5, no. 3, pp. 192–200, 2015.

[11] M. R. Daliri, M. Taghizadeh, and K. S. Niksirat, "EEG Signature of Object Categorization from Event-related Potentials," *J. Med. Signals Sens.*, vol. 3, no. 1, pp. 37–44, 2013.

[12] E. K. S. Louis *et al.*, *Appendix 6. A Brief History of EEG*. American Epilepsy Society, 2016.

[13] "Nature's Intellect - The Research Group." https://www.researchgrp.com/blog/natures-intellect/ (accessed Apr. 15, 2021).

[14] "Workload detection based on EEG device for teleoperation of a mobile robot | IEEE Conference Publication | IEEE Xplore." https://ieeexplore.ieee.org/document/8211636/ (accessed Apr. 15, 2021).

[15] Y. Yuan, Z. Li, and Y. Liu, "Brain Teleoperation of a Mobile Robot Using Deep Learning Technique," in *2018 3rd International Conference on Advanced Robotics and Mechatronics (ICARM)*, Jul. 2018, pp. 54–59, doi: 10.1109/ICARM.2018.8610711.

[16] "iMotions." https://imotions.com/blog/what-is-eeg/ (accessed Apr. 15, 2021).

[17] "Rev. Mod. Phys. 65, 413 (1993) - Magnetoencephalography---theory, instrumentation, and applications to noninvasive studies of the working human brain." https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.65.413 (accessed Apr. 15, 2021).

[18] "Combining complexity measures of EEG data: multiplying measures reveal previously hidden information." https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4648221/ (accessed Apr. 15, 2021).

[19] M. A. Bell and K. Cuevas, "Using EEG to Study Cognitive Development: Issues and Practices," *J. Cogn. Dev. Off. J. Cogn. Dev. Soc.*, vol. 13, no. 3, pp. 281–294, Jul. 2012, doi: 10.1080/15248372.2012.691143.

[20] C. Mulert, “Simultaneous EEG and fMRI: towards the characterization of structure and dynamics of brain networks,” *Dialogues Clin. Neurosci.*, vol. 15, no. 3, pp. 381–386, Sep. 2013.
[21] X. Liu, O. Makeyev, and W. Besio, “Improved Spatial Resolution of Electroencephalogram Using Tripolar Concentric Ring Electrode Sensors,” *J. Sens.*, vol. 2020, p. e6269394, Jun. 2020, doi: 10.1155/2020/6269394.
[22] “Deep learning-based electroencephalography analysis: a systematic review - IOPscience.” https://iopscience.iop.org/article/10.1088/1741-2552/ab260c#jneab260cbib023 (accessed Apr. 15, 2021).
[23] *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Oxford University Press.
[24] R. Srinivasan, “Methods to Improve the Spatial Resolution of EEG,” vol. 1, no. 1, p. 10, 1999.
[25] “The Privacy of EEG Data,” *Neuroethics & Law Blog*. https://kolber.typepad.com/ethics_law_blog/2014/05/the-privacy-of-eeg-data.html (accessed Apr. 20, 2021).
[26] “AI for EEG data processing: Deep Learning | Bitbrain.” https://www.bitbrain.com/blog/ai-eeg-data-processing (accessed Apr. 20, 2021).
[27] “Towards Zero Training for Brain-Computer Interfacing.” https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0002967 (accessed Apr. 20, 2021).
[28] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.
[29] “Google AI Blog: Federated Learning: Collaborative Machine Learning without Centralized Training Data.” https://ai.googleblog.com/2017/04/federated-learning-collaborative.html (accessed Apr. 20, 2021).
[30] “Building decision tree classifier on private data | Proceedings of the IEEE international conference on Privacy, security and data mining - Volume 14.” https://dl.acm.org/doi/10.5555/850782.850784 (accessed Apr. 20, 2021).
[31] C. Ju, D. Gao, R. Mane, B. Tan, Y. Liu, and C. Guan, “Federated Transfer Learning for EEG Signal Classification,” *2020 42nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBC*, pp. 3040–3045, Jul. 2020, doi: 10.1109/EMBC44109.2020.9175344.
[32] “All about EEG Artifacts and filtering tools | Bitbrain.” https://www.bitbrain.com/blog/eeg-artifacts (accessed Apr. 20, 2021).
[33] “EEG Signal Processing for Dummies - Neuroelectrics Neuroelectrics Blog - Latest news about EEG & Brain Stimulation.” https://www.neuroelectrics.com/blog/2014/12/18/eeg-signal-processing-for-dummies/ (accessed Apr. 20, 2021).
[34] “What are the preprocessing methods to enhance EEG data for general purpose?,” *ResearchGate*. https://www.researchgate.net/post/What-are-the-preprocessing-methods-to-enhance-EEG-data-for-general-purpose (accessed Apr. 20, 2021).
[35] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K. Muller, “Optimizing Spatial filters for Robust EEG Single-Trial Analysis,” *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 41–56, 2008, doi: 10.1109/MSP.2008.4408441.
[36] “Comparison of Wavelet Transform and FFT Methods in the Analysis of EEG Signals | SpringerLink.” https://link.springer.com/article/10.1023/A:1015075101937 (accessed Apr. 20, 2021).
[37] “Chapter 3.5: Digital signal processing - Newnes Interfacing Companion.” https://learning.oreilly.com/library/view/newnes-interfacing-companion/9780750657204/xhtml/B9780750657204501212.htm#cesectitle1 (accessed Apr. 20, 2021).
[38] J. Brownlee, “Why One-Hot Encode Data in Machine Learning?,” *Machine Learning Mastery*, Jul. 27, 2017. https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/ (accessed Apr. 20, 2021).
[39] “MIT 6.S191: Convolutional Neural Networks - YouTube.” https://www.youtube.com/watch?v=AjtX1N_VT9E (accessed Apr. 20, 2021).

[40] "A Basic Introduction to Convolutional Neural Network | by Himadri Sankar Chatterjee | Medium." https://medium.com/@himadrisankarchatterjee/a-basic-introduction-to-convolutional-neural-network-8e39019b27c4 (accessed Apr. 20, 2021).
[41] M. S. Researcher PhD, "Simple Introduction to Convolutional Neural Networks," *Medium*, Jul. 29, 2020. https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac (accessed Apr. 20, 2021).
[42] "CNN | Introduction to Pooling Layer - GeeksforGeeks." https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/ (accessed Apr. 20, 2021).
[43] R. Christensen, "Thoughts on prediction and cross-validation.," p. 8.
[44] J. Brownlee, "A Gentle Introduction to k-fold Cross-Validation," *Machine Learning Mastery*, May 22, 2018. https://machinelearningmastery.com/k-fold-cross-validation/ (accessed Apr. 20, 2021).
[45] "K-Fold Cross Validation. Evaluating a Machine Learning model can… | by Krishni | DataDrivenInvestor." https://medium.datadriveninvestor.com/k-fold-cross-validation-6b8518070833 (accessed Apr. 20, 2021).
[46] "Cross-validation (statistics)," *Wikipedia*. Apr. 19, 2021, Accessed: Apr. 20, 2021. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Cross-validation_(statistics)&oldid=1018725954.
[47] "MNE — MNE 0.22.1 documentation." https://mne.tools/stable/index.html (accessed Apr. 20, 2021).
[48] A. Sauve, *scaleogram: User friendly scaleogram plot for Continuous Wavelet Transform*. .
[49] 高大山 Dashan Gao, *DashanGao/Federated-Transfer-Learning-for-EEG*. 2021.
[50] S. Koelstra *et al.*, "DEAP: A Database for Emotion Analysis ;Using Physiological Signals," *IEEE Trans. Affect. Comput.*, vol. 3, no. 1, pp. 18–31, Jan. 2012, doi: 10.1109/T-AFFC.2011.15.
[51] "Convolutional Neural Network (CNN) | TensorFlow Core." https://www.tensorflow.org/tutorials/images/cnn#import_tensorflow (accessed Apr. 20, 2021).
[52] "tf.one_hot | TensorFlow Core v2.4.1." https://www.tensorflow.org/api_docs/python/tf/one_hot (accessed Apr. 20, 2021).
[53] "sklearn.model_selection.KFold — scikit-learn 0.24.1 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html (accessed Apr. 20, 2021).
[54] J. Brownlee, "How to Use StandardScaler and MinMaxScaler Transforms in Python," *Machine Learning Mastery*, Jun. 09, 2020. https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/ (accessed Apr. 20, 2021).
[55] R. H. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit," *Nature*, vol. 405, no. 6789, pp. 947–951, Jun. 2000, doi: 10.1038/35016072.
[56] G. Lee, R. Gommers, F. Waselewski, K. Wohlfahrt, and A. O'Leary, "PyWavelets: A Python package for wavelet analysis," *J. Open Source Softw.*, vol. 4, no. 36, p. 1237, Apr. 2019, doi: 10.21105/joss.01237.
[57] S. Alhagry, A. Aly, and R. El-Khoribi, "Emotion Recognition based on EEG using LSTM Recurrent Neural Network," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, Oct. 2017, doi: 10.14569/IJACSA.2017.081046.
[58] "(PDF) Arousal and Valence Classification Model Based on Long Short-Term Memory and DEAP Data for Mental Healthcare Management." https://www.researchgate.net/publication/328806152_Arousal_and_Valence_Classification_Model_Based_on_Long_Short-Term_Memory_and_DEAP_Data_for_Mental_Healthcare_Management (accessed Apr. 19, 2021).
[59] D. S. Naser and G. Saha, "Recognition of emotions induced by music videos using DT-CWPT," Mar. 2013, pp. 53–57, doi: 10.1109/IndianCMIT.2013.6529408.