Deep Learning Models for EEG Seizure Detection

Arturo Altamirano, Justin Haysbert, and Pavan Kumar Sanjay

Abstract

Epilepsy is a disease that burdens the afflicted with intermittent and debilitating seizures. Much research has been done into its treatment and toward the mitigation of its symptoms. One such field of study pertains to the analysis of electroencephalogram (EEG) data to identify the afflicted portion of the brain, determine which episodes are truly epileptic in nature, as well as to possibly warn of an oncoming episode. These recordings generate a large amount of data, which if analyzed by a medical professional, may take a large amount of effort and concentration. This excessive manpower requirement may be augmented, or perhaps even replaced, by a sufficiently trained deep learning model. Such a model will enable medical professionals to receive rapid, accurate diagnosis of their patients, and may even open the possibility of models capable of warning those in primary care when an epileptic episode is coming. Thus further assisting medical professionals. While many models have been developed for both detection and prediction, we have developed a unique method which trains to high accuracy with very minimal training time. This method leverages scalogram images of data, with additional denoising and filtration, to improve upon current accuracy levels from leading models. In doing so, providing a superior means of short-length seizure detection.

Intro

The purpose of this project was to detect, or even predict, when a seizure was occurring or would occur. This research builds on several pre-existing studies, aiming to iterate upon their findings and explore new methodologies. By leveraging advancements in machine learning and signal processing, we sought to create effective models capable of both detecting seizures in EEG signals and predicting their onset in time-sensitive scenarios. The long-term implementations of our models would require a protracted effort that spans disciplines, what we produced in the scope of this project was designed to satisfy the requirements of the assignment, but with enthusiastic drive toward solving the problem presented to us.

To achieve these objectives, we developed three distinct models and architectures. Two of these were focused on the Bonn database, which provides a rich, but brief, dataset for seizure detection. These models were designed to classify seizure and non-seizure EEG signals with high accuracy, exploring innovative feature extraction and classification techniques. Our experiments with this database helped refine detection methodologies and provided insights into signal variability across seizure events. We intended to experiment with currently defined architectures, adding elements or implementing them in different ways, with the hope of improving overall performance.

The third model was dedicated to the CHB MIT dataset, with a focus on seizure prediction. Unlike the Bonn database, which is tailored for detection, the CHB MIT dataset allows for temporal analysis on massive amounts of time-series data, making it suitable for forecasting seizures before they occur. This approach required the development of a time-series prediction model capable of capturing long-term dependencies in the data. Our goal was to identify precursors to seizure events, providing a framework for proactive intervention and patient safety. We were ultimately unsuccessful at developing this portion, however our success in the prior 2 experiments was satisfactory for our effort.

Background

There were 3 main literary sources for this project. Primarily we utilized modified implementations of models defined in "Deep Learning Classification for Epilepsy Detection Using a Single Channel Electroencephalography" by Liu and Woodson (2019) of the University of North Texas, "Features extracted by eigenvector methods for detecting variability of EEG signals" by Übeyli and Güler (2006), and finally a highly modified version of the model described by Mallick and Baths (2024) in their paper titled: "Novel deep learning framework for detection of epileptic seizures using EEG signals" for our multichannel model.

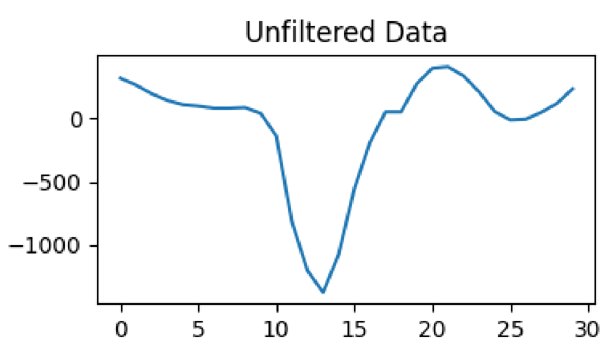
The work provided by Liu and Woodson details the use of Butterworth filters prior to training on a CNN based architecture. Their findings indicate a high degree of accuracy for a binary classification task on single-channel data with short training periods and practical testing results. We decided to implement this approach with the addition of a deeper architecture and an additional bandpass filter, to increase the accuracy they had already achieved.

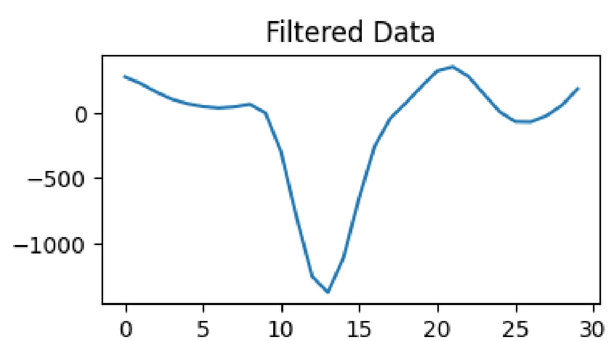
In their research, Übeyli and Güler describe a methodology for analyzing EEG signals by extracting features using eigenvector methods. Their work emphasizes the importance of transforming EEG data into a domain where variations are more apparent, enabling improved detection and classification. We wanted to measure its effectiveness for this task, and to gauge how a CNN would perform when trained on such images.

Finally, Mallick and Baths outlined a general use model fit for detecting epileptic signals. They claimed that it was fit for a 4 to 5 class classification problem, with data needing to be separated into a series of stages in relation to a seizure period. Their model architecture, composed of a combination of 1D Convolutional Neural Networks and Bidirectional Long Short-Term Memory layers, achieved state of the art accuracy on the Bonn dataset. We decided this would be our candidate to attempt a bridge to multichannel classification.

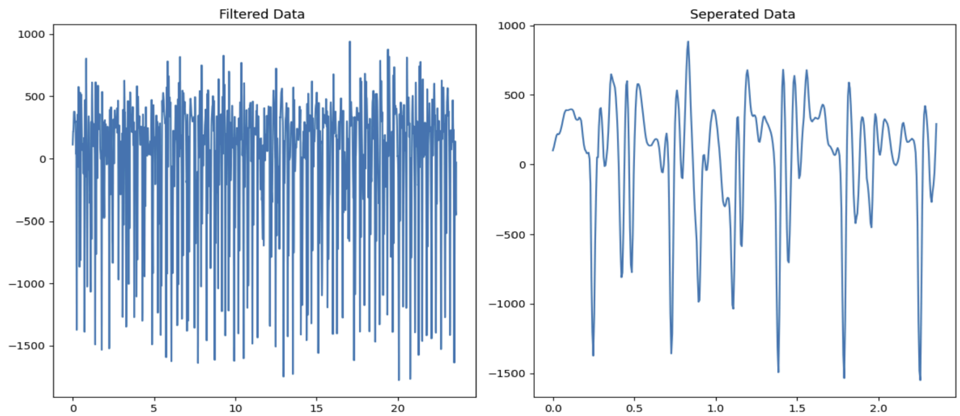
Methods

For the Bonn detection models, we utilized filtration and image abstraction to the visual domain. For this filtration we utilized a set of Butterworth and Bandpass techniques to 'smoothen' and reduce the noise of the data. This enables the rather dense and erratic data to be more easily interpreted by our model architecture.



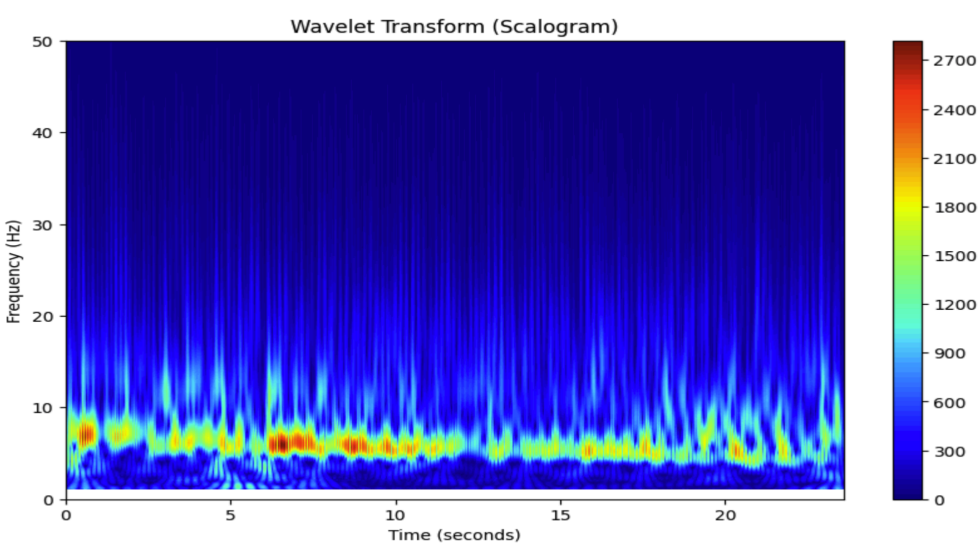


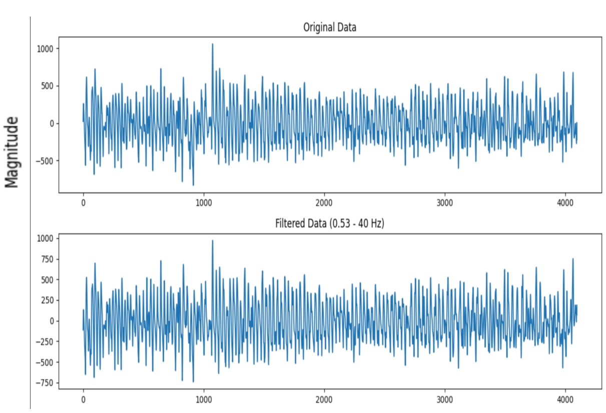
The implementation from the original literature was implemented in MatLab, our project centered around converting this implementation to Python and then incorporating the additional bandpass filtration. Data was also truncated, where appropriate, to ensure meaningful insight for the model to observe. This truncation, and subsequent separation, increased validation and testing accuracy and resulted in more precision. By filtering only the most informative portions of the EEG recordings, we ensured that the model was exposed to data rich in meaningful features. This preprocessing step not only improved computational efficiency but also significantly boosted validation and testing accuracy. The truncation strategy ultimately resulted in a model that was not only more precise but also more robust, effectively capturing the critical patterns required for accurate seizure detection and prediction.



Upon converting the described architecture from the literature to a Python implementation, its performance was not equitable to the original reported levels. We believed this to be a result of the performance trade-off when using Python compared to the native MatLab. We decided another contribution would be to modify the described model to achieve results more closely aligned with the literature. In order to achieve this performance increase, a dense layer was added to increase linear transformation. After several re-instatiations, superior performance was confirmed and we began working on the second architecture.

For the second architecture/approach, we decided to experiment with time vector representation and to operate on the data in the visual domain via scalogram transformations. Inspired by the approach of Uler, we opted to leverage scalogram transformations to convert the time-series EEG data into the frequency domain. By subsequently generating scalograms, we visualize the frequency content over time, capturing both spectral and temporal features. This approach allows us to explore intricate patterns in the EEG data, and for our model to identify variability and period of distinction within an otherwise noisy dataset.





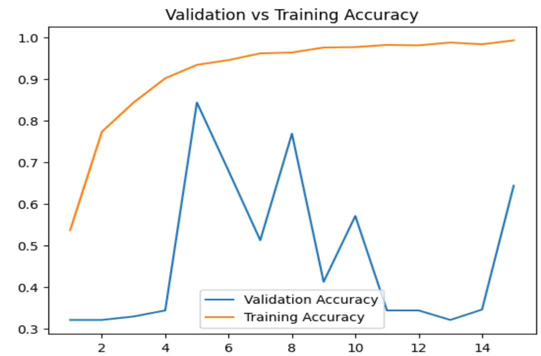
Experiments:

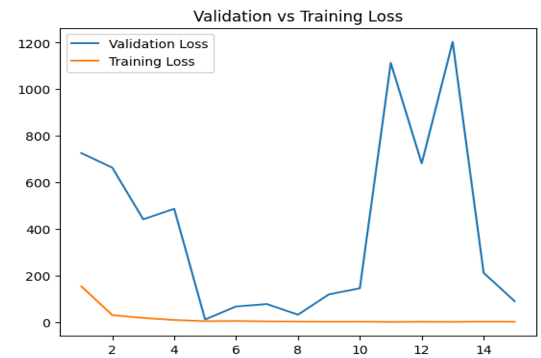
The original data for the Bonn dataset was split into 5 sets (A-E) each containing 100 files. The A-B sets represent healthy signals,sets C-D represent pre-seizure data from different portions of the brain and with eyes open and closed, and E represents pure seizure period data.

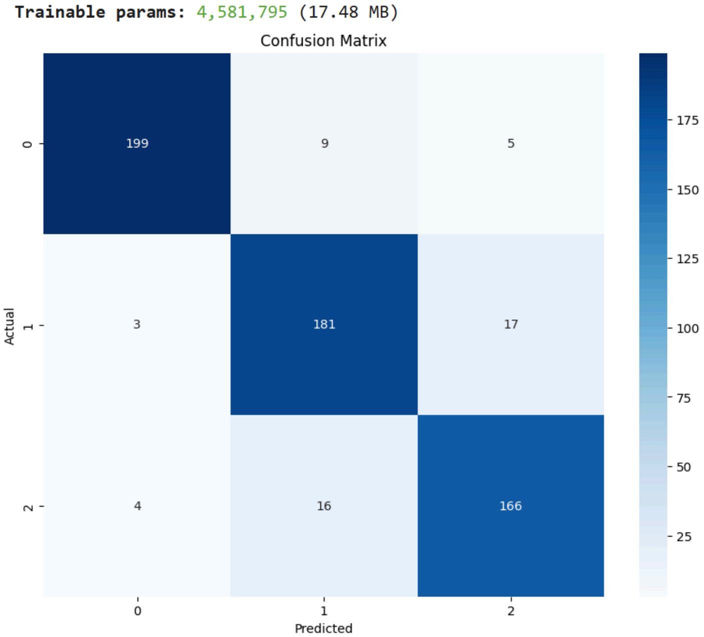
Model: "functional_3"

Layer (type)	Output Shape	Param #
input_layer_3 (InputLayer)	(None, 224, 224, 3)	0
conv2d_9 (Conv2D)	(None, 222, 222, 32)	864
batch_normalization_9 (BatchNormalization)	(None, 222, 222, 32)	128
activation_9 (Activation)	(None, 222, 222, 32)	0
conv2d_10 (Conv2D)	(None, 220, 220, 32)	9,216
batch_normalization_10 (BatchNormalization)	(None, 220, 220, 32)	128
activation_10 (Activation)	(None, 220, 220, 32)	0
conv2d_11 (Conv2D)	(None, 218, 218, 32)	9,216
batch_normalization_11 (BatchNormalization)	(None, 218, 218, 32)	128
activation_11 (Activation)	(None, 218, 218, 32)	0
flatten (Flatten)	(None, 1520768)	0
dense_3 (Dense)	(None, 3)	4,562,307

Total params: 4,581,987 (17.48 MB)
Trainable params: 4,581,795 (17.48 MB)



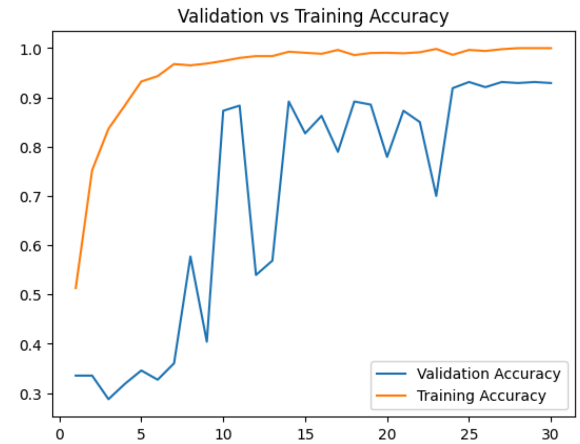


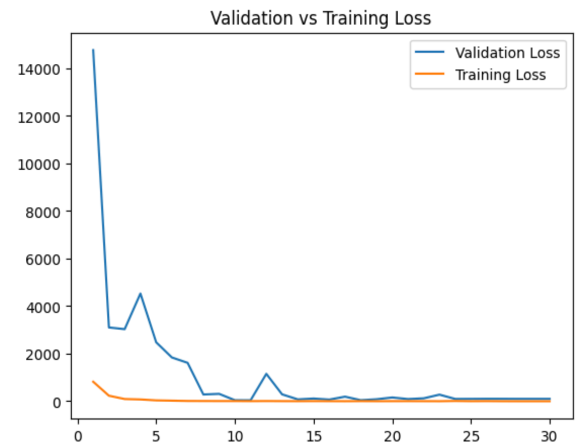


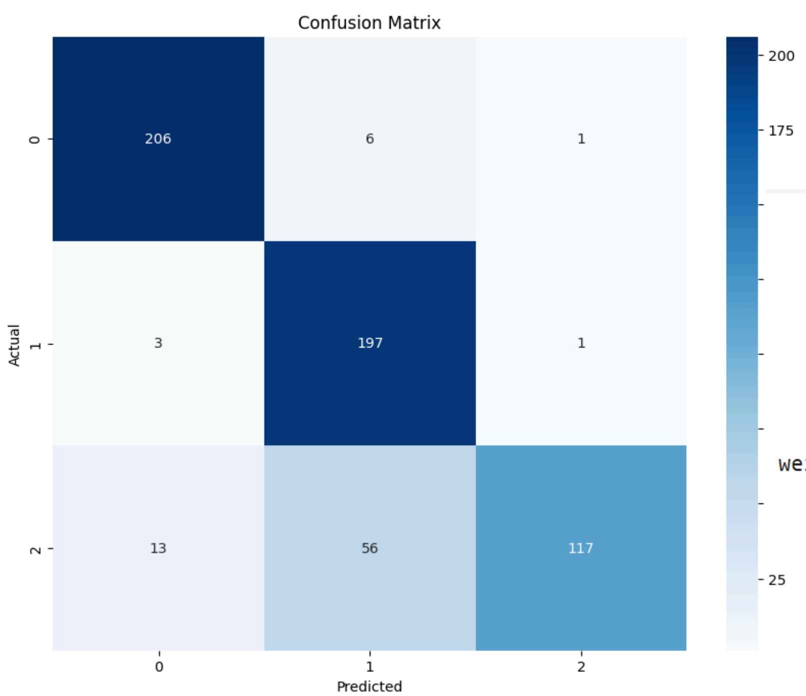
	precision	recall	f1-score	support
0	0.93	0.97	0.95	213
1	0.76	0.98	0.86	201
2	0.98	0.63	0.77	186
accuracy			0.87	600
macro avg	0.89	0.86	0.86	600
weighted avg	0.89	0.87	0.86	600

The above model, based on the literature, can be seen to exhibit high volatility. We experimented with several architectures, eventually determining that a dense layer preserves the quick training time, while enabling better performance. Below you can see the improved performance.

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 224, 224, 3)]     0

 conv2d (Conv2D)             (None, 222, 222, 32)      864

 batch_normalization (Batch  (None, 222, 222, 32)      128
 Normalization)

 activation (Activation)     (None, 222, 222, 32)      0

 conv2d_1 (Conv2D)           (None, 220, 220, 32)      9216

 batch_normalization_1 (Bat  (None, 220, 220, 32)      128
 chNormalization)

 activation_1 (Activation)   (None, 220, 220, 32)      0

 conv2d_2 (Conv2D)           (None, 218, 218, 32)      9216

 batch_normalization_2 (Bat  (None, 218, 218, 32)      128
 chNormalization)

 activation_2 (Activation)   (None, 218, 218, 32)      0

 dense (Dense)               (None, 218, 218, 16)      528

 flatten (Flatten)           (None, 760384)            0

 dense_1 (Dense)             (None, 3)                 2281155

=================================================================
Total params: 2301363 (8.78 MB)
Trainable params: 2301171 (8.78 MB)
Non-trainable params: 192 (768.00 Byte)
```



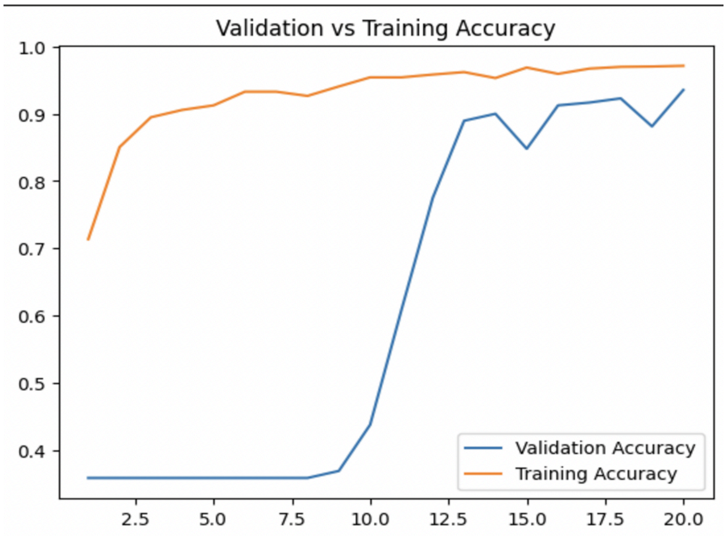


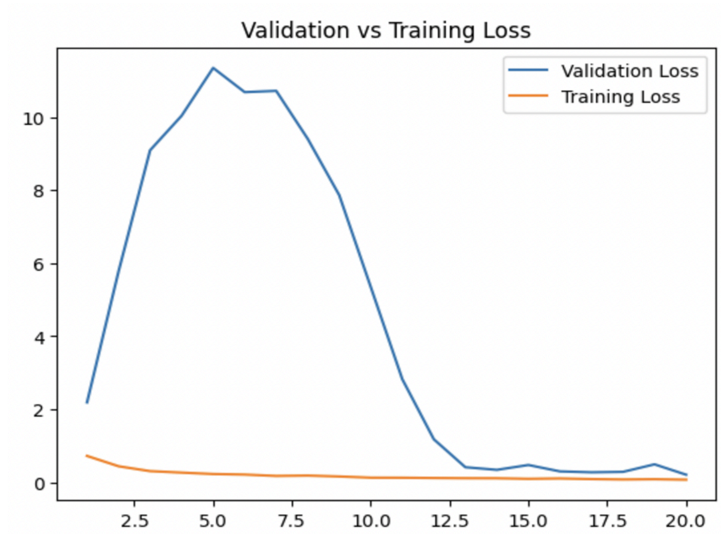


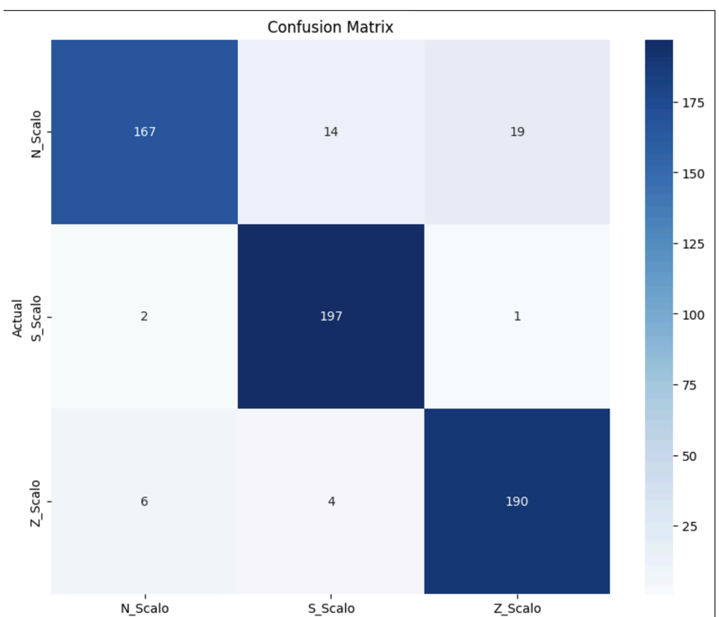

```
              precision    recall  f1-score   support

           0       0.97      0.93      0.95       213
           1       0.88      0.90      0.89       201
           2       0.88      0.89      0.89       186

    accuracy                           0.91       600
   macro avg       0.91      0.91      0.91       600
weighted avg       0.91      0.91      0.91       600
```

In our novel approach to train models using scalogram data representation the same models from the experiments above were used with the only change being the use of dropout layers. Using this model on scalogram data representations we were able to obtain 0.93 test accuracy. We were able to obtain a weighted average precision of 0.93 as well as a recall of 0.92.

Model 1 w/ scalograms:

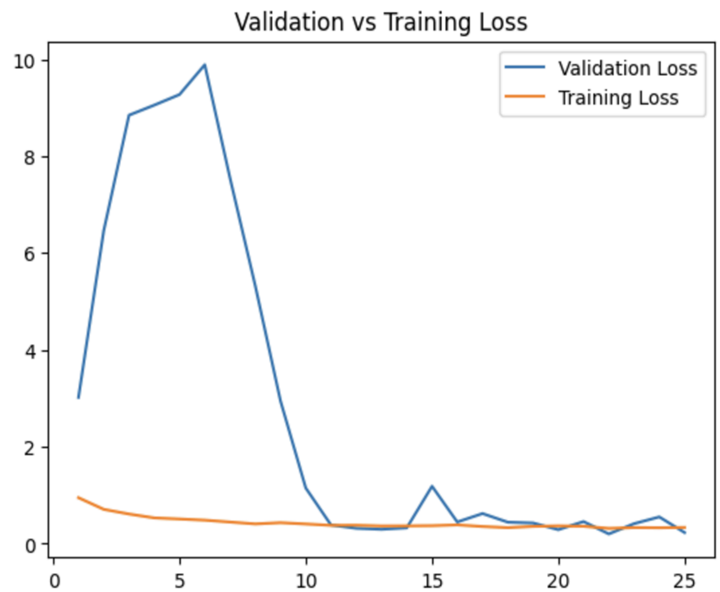


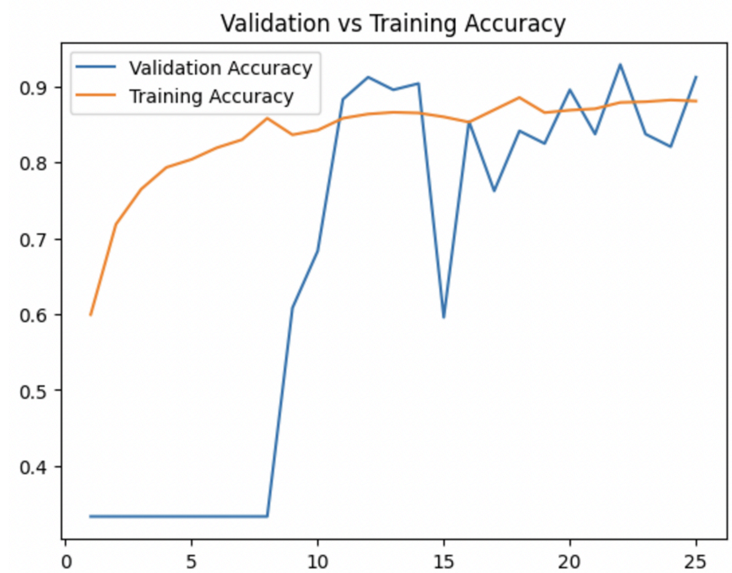


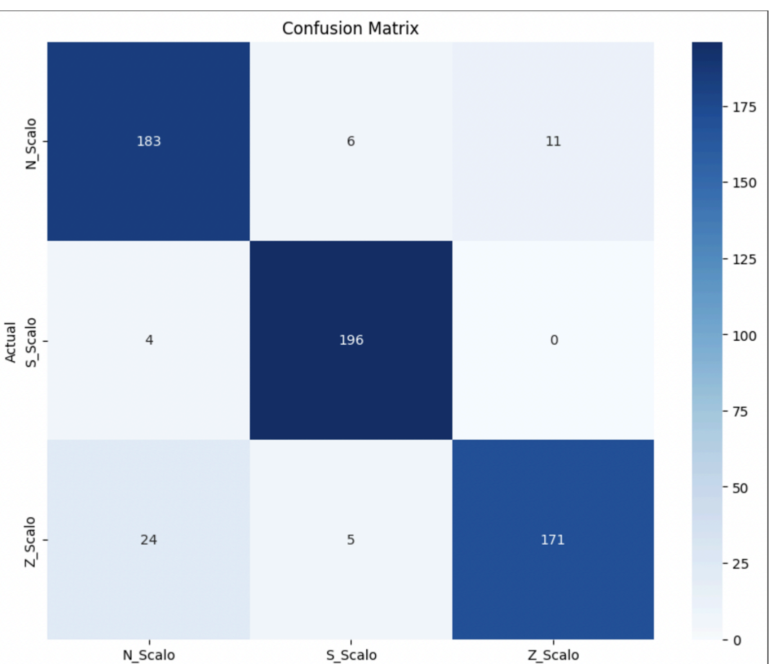


	precision	recall	f1-score	support
N_Scalo	0.95	0.83	0.89	200
S_Scalo	0.92	0.98	0.95	200
Z_Scalo	0.90	0.95	0.93	200
accuracy			0.92	600
macro avg	0.93	0.92	0.92	600
weighted avg	0.93	0.92	0.92	600

In an attempt to obtain a higher accuracy model we attempted to use data augmentation. Our augmentation included rotations of 15 degrees, 10% horizontal and vertical shifts, and image flips. The augmentations had little impact on our model with it producing a similar accuracy of 0.92, 0.92 Precision, and 0.92 recall. This means that our data augmentation methods were ineffective at providing useful training instances. In the future new augmentation methods that are compatible with scalogram image data should be used. The cited paper by Übeyli and Güler , detailed the use of added noise to parts of the scalogram to create new training instances. Using this literature it is possible to implement a viable augmentation of data to make up for the size of the dataset. Another solution would be acquiring more data to skip the need for data augmentation.

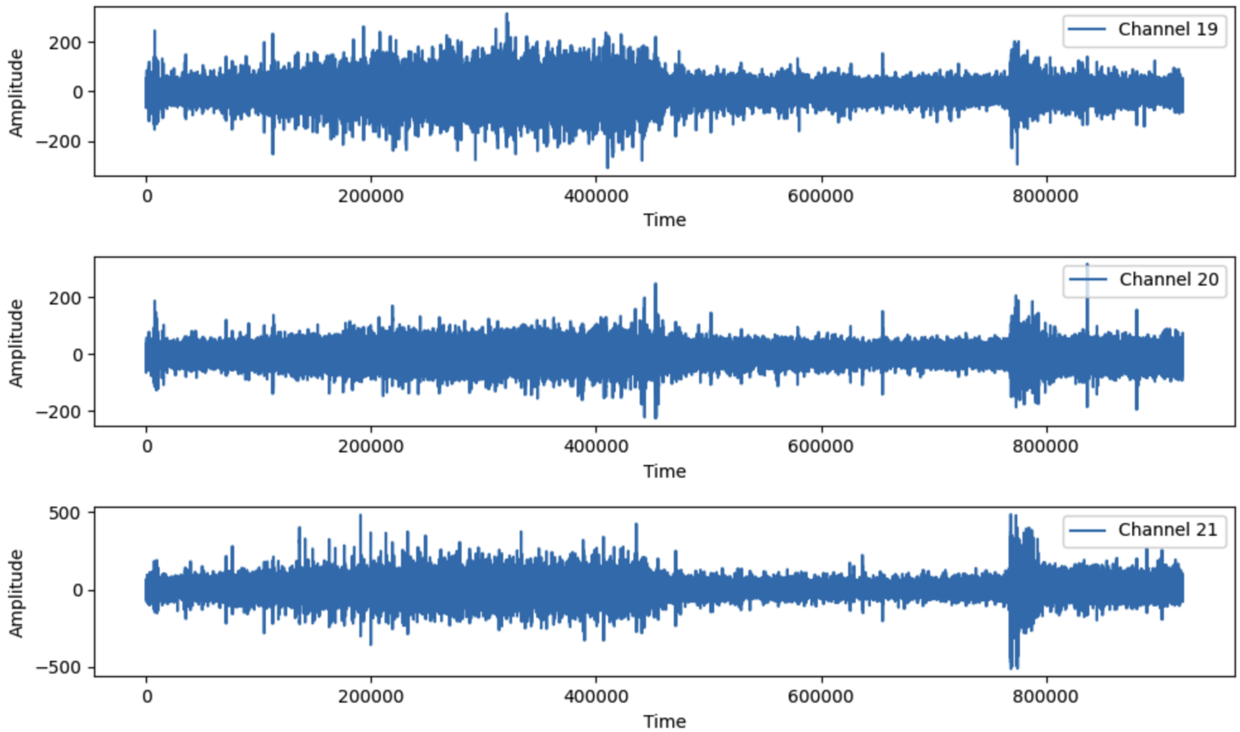




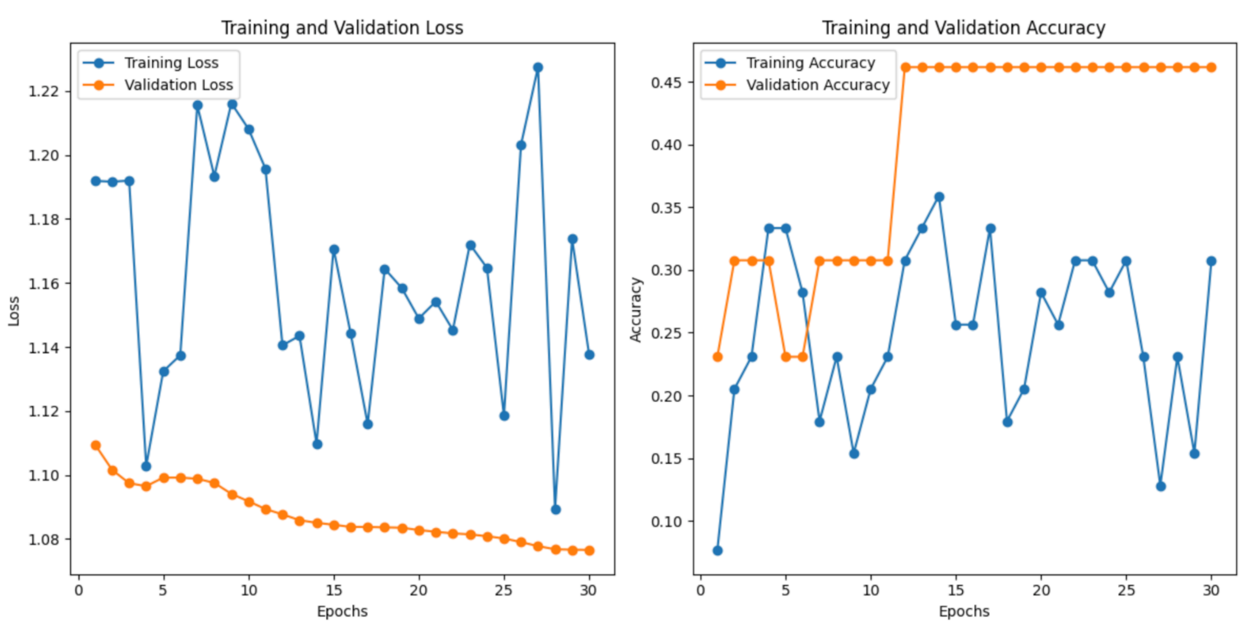


	precision	recall	f1-score	support
N_Scalo	0.87	0.92	0.89	200
S_Scalo	0.95	0.98	0.96	200
Z_Scalo	0.94	0.85	0.90	200
accuracy			0.92	600
macro avg	0.92	0.92	0.92	600
weighted avg	0.92	0.92	0.92	600

For the CHB MIT dataset, raw data is separated between 22 pediatric subjects. Each case contains between 9 and 42 continuous .edf files containing 23 channels of EEG data collected by a headset of electrical nodes. This data was collected from patients undergoing testing to evaluate their candidacy for surgery. For preprocessing, we manually labeled each of the 1-hour segments and passed these data segments through a butterworth and band pass filter, trained on the model described in the literature, and achieved an approximate accuracy of 75% with a relatively unpredictable training period.



Our multichannel model struggled with the inherent lack of seizure data, and tended to overfit to non-seizure data. Ultimately, we decided this portion of the project would not be feasible.



Discussion:

Obviously, the lack of success toward the multichannel model is something we would have liked to remedy, but given the time constraint of the semester, we are pleased with our overall results. The need for more advanced processing, and the immense computational cost, means that the multichannel dataset would be a project fit for a much larger team with a much larger resource pool. We believe the problem lies in the way the multichannel data is preprocessed, and that a more nuanced approach is needed than simply loading it into a numpy format.

Concerning the pair of single channel implementations, a potential alteration to enable multichannel classification would naturally be the next step in improving performance. Experimentation with attention mechanisms was something we considered, as insight into the areas of emphasis within our dataset was of great interest to us.

Conclusion:

We learned a tremendous amount during this project, and enjoyed developing improvements for state of the art deep learning implementations. We would ultimately like to return to the multichannel prediction problem in the future, and will do so if given an opportunity.

References:

Übeyli, E. D., & Güler, İ. (2006). Features extracted by eigenvector methods for detecting variability of EEG signals. *Pattern Recognition Letters*, *28*(5), 592–603. https://doi.org/10.1016/j.patrec.2006.10.004

Liu, J., & Woodson, B. (2019). Deep Learning Classification for Epilepsy Detection Using a Single Channel Electroencephalography (EEG). *University of North Texas Publications*, 23–26. https://doi.org/10.1145/3342999.3343008

Mallick, S., & Baths, V. (2024). Novel deep learning framework for detection of epileptic seizures using EEG signals. *Frontiers in Computational Neuroscience*, *18*. https://doi.org/10.3389/fncom.2024.1340251

Singh, N., & Dehuri, S. (2018). Usage of deep learning in epileptic seizure detection through EEG Signal. In *Lecture notes in electrical engineering* (pp. 219–228). https://doi.org/10.1007/978-981-13-0776-8_20