**EEG Signal Analysis for Emotional Stimulus Decoding**

**Stefan Gligorevic1, Vangelis Metsis2**

**1**Millersville University of Pennsylvania, Millersville, PA

2Department of Computer Science, Texas State University, San Marcos, TX

**1. Abstract**

Emotions play a significant role in our daily lives. They affect how people feel and directly impact their quality of life. If a person’s emotions, or emotional stimuli which they are confronted with could be detected, that information could be used to improve that person’s overall life. Electroencephalography (EEG) is a useful tool which records brain activity in the form of signals from the scalp using sensors called electrodes. Based on the different band frequencies of these signals, a patient can be classified to a specific emotional state. Using this information, we created two different deep learning models to attempt to classify between static and dynamic versions of fear and anger stimuli. The results showed that fear and anger stimuli were too closely related to be accurately classified by deep learning models.

**2. Introduction**

Emotions play a large part in our daily lives. How we feel controls our whole outlook on life. When we are happy, our quality of life is improved, and we are more likely to spread positivity. However, when we are feeling down, our outlook on life is negative, and we are more likely to lash out at others and do other unfavorable actions as a result. If we could detect the emotions a person is viewing or experiencing, we could use that information to improve the quality of their lives.

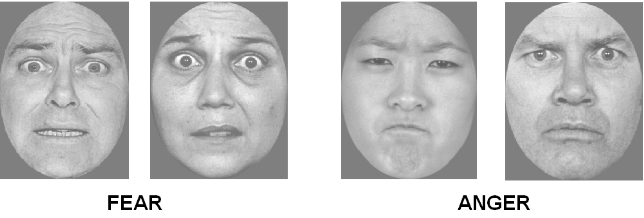
This is particularly useful for those who are mute or those who have a difficult time portraying their emotions and feelings towards things. Many different techniques exist for measuring brain activity, such as recording MEG or EOG. (Maybe include an example and site source)

For us, we felt that EEG was our best bet for collecting brain activity to use for emotional stimulus decoding. An electroencephalogram, or EEG for short, is a test that records the electrical signals of the brain. The electrical signals are recorded using electrodes that are placed on the scalp, which send data in the form of signals to the computer that records the results [1]. Unlike EOG, which is more commonly used to record eye movements, EEG focuses on the electrical activity in the brain. The recorded EEG signals are typically divided into five band frequencies: Delta, Theta, Alpha, Beta, and Gamma [2]. Based on these frequency bands, we can make an accurate guess as to which emotional state someone is experiencing.

Another big benefit to EEG is that it can be fed into machine learning algorithms. Using this capability, my goal is to develop a model which can accurately classify between the different emotional stimuli in our dataset. Using different techniques such as deep learning, classical machine learning, fine-tuning, and feature-extraction, I am hoping to achieve high accuracies with my final model. A possible limitation to my plan is the noise that comes with the raw EEG dataset that I am working with. I will have to eliminate as much noise as possible to allow the model to find distinct patterns to accurately classify the data.

Speaking of my dataset, I am using the LaBar Facemorph files dataset. The experiments were organized and run by Kevin LaBar, Michael Crupain, James Voyvodic, and Gregory McCarthy at Duke University. 64 channels of raw EEG data were recorded along with two additional EOG channels using Functional Magnetic Resonance Imagery (fMRI). For most of my experiments, the two EOG channels were disregarded because my focus was primarily on the EEG data. The experiments consisted of 24 participants who were each shown images depicting the four different stimuli: static fear, static anger, dynamic fear, and dynamic anger. The static images were simply non-animated photographs of a person depicting one of the two emotions. The dynamic images on the other hand were animated showing the person starting from a neutral expression, moving to an expression depicting one of the two emotions.

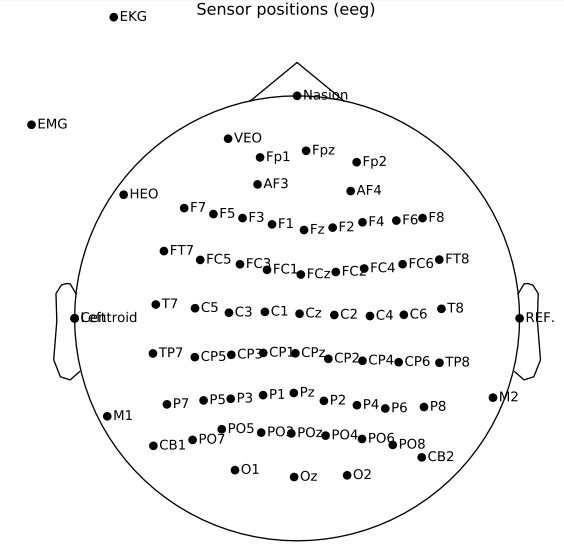
**Figure 1:** Examples of different static fear and static anger stimuli shown



There were two separate versions of this dataset that I worked with. Firstly, there is the large version which was known as the single trial files. In the single trial files, for each of the stimuli, up to 72 trials were recorded with the user, some of them being rejected for reasons such as external noise or errors that interfered with the data. Each of the trials contained 1,101 rows of EEG data by 66 columns. Each column represented one of the channels.

The other version was smaller and averaged the data for the trials. Instead of having up to 72 trials for each stimulus for each user, these trials were taken and the data was averaged to make one trial of 1,101 rows by 66 columns.

**Figure 2:** Positioning of EEG channels on the scalp of the participants.



We developed a 1D-CNN model along with an LSTM model to attempt to classify between the stimuli. The experiments that were run included classifying between fear and anger for both the static and dynamic files, as well as classifying between static and dynamic files for both the fear and anger stimuli. The main focus was on classifying between fear and anger because the project was focused on emotional stimulus decoding. Techniques such as scaling, fine-tuning, and transfer learning were applied to both of the models in attempt to achieve the highest possible accuracies. In the end, we were not able to achieve above a 60% overall validation accuracy for either of the classification experiments. Performing some visualizations on the data, we spotted that the EEG signals for the four categories were very similar for each of the users. Using this information, the conclusion was made that fear and anger stimuli were too closely related to be able to accurately classify between them using machine learning. The deep learning models were not able to find distinct patterns to distinguish between the stimuli and accurately classify them.

**3. Related Work**

EEG is a relatively new concept in the world of data science. Almost all of the articles on EEG related works have been published within the last five years. One of the earliest papers on the topic of EEG is “Classifying Different Emotional States by Means of EEG-Based Functional Connectivity Patterns” which was published by You-Yun Lee and Shulan Hsieh in April of 2014 [3]. It is considered to be one of the first works in the field of EEG, as well as possibly the first big discovery regarding using EEG for emotion recognition.

Their experiments included film clips which depicted one of three of the following emotional states: positive, neutral, or negative. These clips were shown to forty participants who were in their low twentys. The electrical brain activity of these participants was recorded using 64-channel Nueroscan equipment, with EEG and EOG signals being amplified using a multichannel biosignal amplifier.

The experimental procedure included ensuring the participant was in a neutral emotional state with a minute long go/nogo task. Baseline EEGs were then recorded for two 90 second trials, one with the participant’s eyes open, and the other with them closed. Finally, a brief five second countdown was shown to grab the attention of the participant before showing the clip, which was anywhere from thirty seconds to five minutes long.

Finishing with their experiments, the results showed that there was a significant difference in the EEG-based functional connectivity for each of the three stimuli. These connectivity patterns were detected using Quadratic Discriminant Analysis. They concluded that EEG was a useful tool for studying the relationship between brain activity and emotional states.

After the discoveries made by Dr. Lee and Dr. Hsieh, EEG was truly introduced to the world of data science. Many experiments, articles, and research papers came in quick succession following Dr. Lee and Dr. Hsieh’s discoveries. One of these research papers, “Emotion Classification using EEG Signals,” was written by Harsh Dabas, Chaitanya Sethi, Chirag Dua, Mohit Dalawat, and Divyashikha Sethia, and was published in December of 2018 [4]. In their paper, the authors proposed a new model for classifying emotion. The typical model up to this point in time was a 2D Valence-Arousal model. However, the authors’ version was a 3D model which included dominance along with valence and arousal.

The experiments were run using a standard DEAP (Database for Emotional Analysis using Physiological Signals) dataset. DEAP is a standard, widely-known dataset for EEG emotion based signal analysis. The stimuli included different musical videos that were a minute long in length. After preprocessing of the data, a 3D model was constructed. Using the 3D emotional model, they were able to identify eight different emotional states associated with the stimuli. These states included relaxed, bored, peaceful, disgust, nervous, sad, surprised, and excited. Using the Naive Bayes machine learning algorithm, the authors achieved an accuracy just shy of 80%.

On the other hand, one downfall to their research was that they found it difficult to locate the part of the brain where the electrical activity was originating from. The signals coming from the neural activity inside the brain were too weak for the EEG headset to detect. Furthermore, there was added noise to the data due to the non-invasive nature of the electrodes.

Despite this, they were able to conclude that adding another emotional feature (being dominance in this experiment) led to better classification of emotions. The authors infer that adding more emotional features in the future could lead to higher accuracies and better classifications.

Finally, one of the most recent works in emotion-based use of EEG is a research paper titled “Valence-Arousal Model based Emotion Recognition using EEG, peripheral physiological signals and Facial Expression,” which was written by Qingyang Zhu, Guanming Lu, and Jingjie Yan, and was published in January of 2020 [5].

In this paper, the authors mention how most emotional recognition of EEG up to this point has been unimodal or bimodal. The typical bimodal based classifications have been made using EEG and peripheral physiological signals. However, the authors propose adding facial expressions to the mix, making a multimodal fusion for emotional classification.

Experiments used data collected from 18 participants in the standard DEAP dataset to classify each part of the multimodal fusion, which included EEG, facial expressions, and peripheral physiological signals. Using deep learning with convolutional neural networks, the authors ran experiments on each of the unimodal, bimodal, and multimodal fusions. The results showed that the unimodal fusions (EEG by itself, peripheral physiological signals by themselves, or facial expressions by themselves) achieved an accuracy between 68 - 71% for arousal and 71 - 77% for valence. The bimodal fusions (any of the three put together as a pair) achieved accuracies between 70 - 71.52% for arousal and 76.38 - 77.78% for valence.. Finally, the multimodal fusion (all three together) achieved an accuracy of 72.2% for arousal, and 78.47% for valence. Looking at the results, the multimodal fusion achieved the highest accuracy for both valence and arousal. Using this information, the authors concluded that a multimodal fusion was capable of making up for the defects of the unimodal and bimodal fusions, and achieve better results.

**4. Methodology**

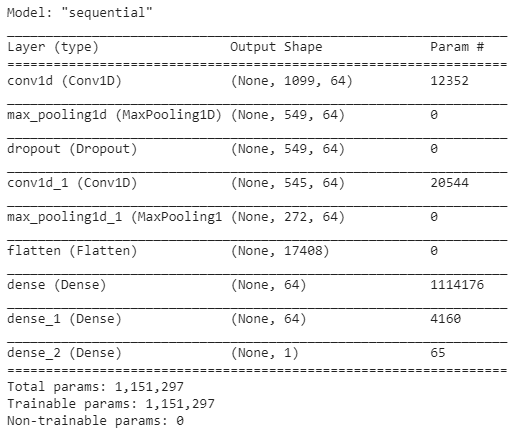
**4.1 Tools**

Now that we have established an idea of what we want to do, it is important to find the right tools to do them. Firstly, the experiments and code was written in Google Collab. Google Collab is an open source notebook provided by Google which offers the ability to separate code into different cells and run them separately, as well as text boxes to clearly label sections and make comments. The language the code was written in is Python. Python is great because it is an open source language that has a lot of support from the community. It contains great libraries and tools for the experiments we are going to run. One of which being another tool we used, TensorFlow. TensorFlow is a library that offers machine learning algorithms as well as modules for creating deep learning neural networks. This will come in handy for the machine learning portion of the experiments. SciPy is another great library, used particularly in data science. For these experiments, its main use is to read and transform the EEG data to make it compatible with the deep learning models. Finally, MNE is possibly the best library for visualizing EEG data. The visualizations created in this paper used MNE to do so.

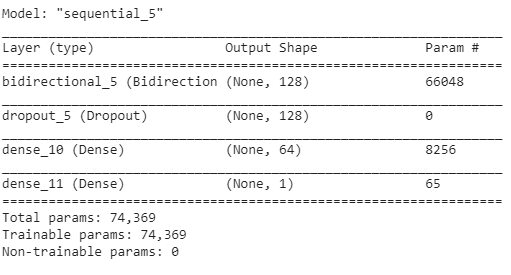
**4.2 Models and Techniques**

The deep learning models used in the experiments were a 1D-Convolution Neural Network (or 1D-CNN) and an LSTM (Long Short-Term Memory) Model. Different techniques were applied to the models to help them achieve the best possible accuracies. These included techniques such as fine-tuning and transfer learning. Fine-tuning included different strategies such as adding or removing layers, adjusting parameters in the model, like the amount of neurons in a dense layer, or adjusting variables that were a part of the training, such as the number of epochs that the data was trained for, or the batch size. Other methods were used to help achieve higher accuracies. For instance, normalization was performed on the data in the form of scaling using a standard scaler, as well as cross validation in the forms of K Fold and Group K Fold, to help prevent overfitting the model and to give each part of the data a chance to be the test set. The number of splits for the cross validation varied to see how the training and validation accuracies changed between each split. However, the number of splits was usually kept at four or five, because that is typical practice.

**Figure 3:** Layout of one of the 1D-CNN models used during the experiments



**Figure 4:** Layout of one of the LSTM models used during the experiments



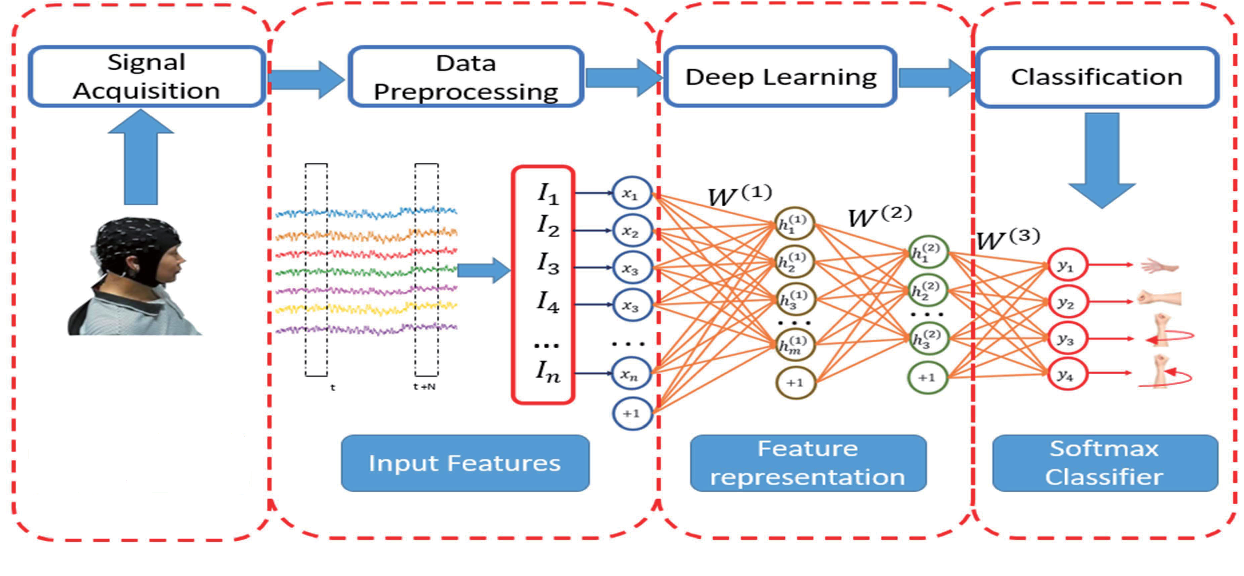
**4.3 Procedure**

Different classification tasks were performed on the datasets. Fear vs anger was the primary focus, since the project was based around emotional stimulus decoding. However, static vs dynamic classification was also performed to see which of the two classification tasks would yield a higher accuracy with our models.

Regarding the large dataset, when performing fear vs anger classifications, the data was split up into arrays containing the static and dynamic files. To go with the arrays containing the files, there were two arrays containing the labels for each file in the static and dynamic files arrays. Files containing fear stimuli were labeled with a 0, and those containing anger were labeled with a 1. Finally, we kept track of which files belonged to which users with groups arrays for each the static and dynamic files. They contained a number between one and 24, representing which user the file belonged to. The files were then read and returned in the form of a 3D array with the shape: (trials, rows, cols). The rows and columns were a constant of 1,101 and 64 since that was how each trial was recorded. The number of trials varied based on the number of trials that were recorded for that user that were accepted to be used for data analysis. Finally, all of the trials for the static and dynamic files were concatenated to make one large 3D array for the static trials and dynamic files, still with the shape: (trials, rows, cols). The same was done to prepare the data for static vs dynamic classification, except the files were split into those containing fear and anger stimuli, and were labeled as dynamic with a 0, and static with a 1. These 3D arrays, called static\_trials and dynamic\_trials or fear\_trials and anger\_trials depending on the experiment, were fed to the deep learning models for training and evaluation.

When working with the smaller dataset, there was no need to concatenate the trials since each file had only one trial. Because there was only one trial and no other rejected or accepted trials, we did not have to create a function to manually loop through and check if a trial was accepted, then record its data if it was. With the smaller dataset, we used the read\_csv function from pandas to read the data from our files. The main trouble was making sure that the lines at the top of the files were ignored, because those contained text and information about the trial that were not actual data. Fortunately, read\_csv has a parameter that allows us to skip a specified set of rows and provides a quick fix to this issue. The read\_csv function returns a pandas dataframe, which looks like an excel spreadsheet. It consisted of 1,101 rows by 66 columns, which was the format of our data. Next, we went through these data frames and removed the two non-EEG channels from the columns, changing the shape to 1,101 rows by 64 columns. Finally, using another pandas function called to\_numpy, all of the data frames were then converted to a 3D numpy array of the shape (samples, rows, columns). Two 3D arrays were made, one for static and one for dynamic files. For static vs dynamic classification, these arrays were split into fear and anger files. These 3D arrays were then fed into the neural networks for deep learning and classification.

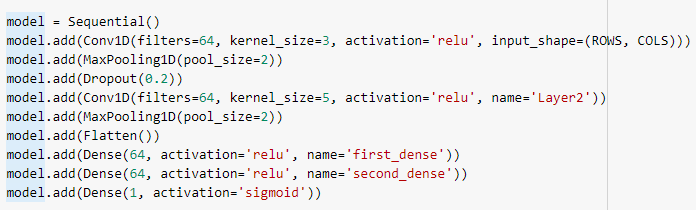
**Figure 5:** A general overview of the process from collecting EEG data to classifying it [6]



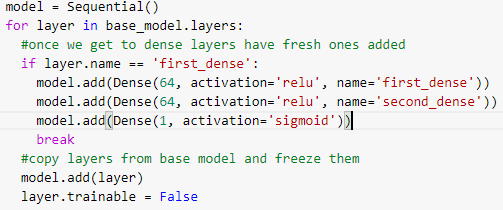
**4.4 Transfer Learning and Scaling**

One of the more intensive experiments performed was transfer learning. Transfer learning involves training all but one user on a base model, then freezing the layers in that model except the last few, then reset the last few layers and use those to train and evaluate on the remaining user. The training with the base model acts as feature extraction, which is a technique used in classical machine learning. The hope using this method is that with the initial training, the model has learned some underlying patterns in the data, and as a result will do well classifying the data using the new model with transfer learning applied.

**Figure 6:** Base 1D-CNN model before transfer learning is applied

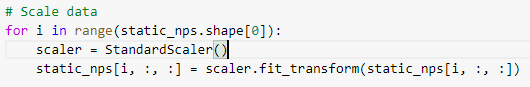


**Figure 7:** 1D-CNN model after transfer learning is applied



Another technique used for achieving higher accuracies was scaling. Scaling is a form of normalization that aims to remove outliers and unwanted noise in the data to help prevent machine learning models from overfitting to the data. This can be done simply by initializing a standard scaler and using it to fit and transform the data.

**Figure 8:** Scaling being performed on data stored in a 3D array called static\_nps



**5. Results**

**5.1 Fear vs Anger Classification**

The primary goal of the project was to classify between the two different emotional stimuli, fear and anger. Experiments were run on both the 1D-CNN models and the LSTM models after they had been fine-tuned. The models were tested on the unchanged data and scaled data. Moreover, transfer learning was applied to the models and tested on unchanged data as well as scaled data. The table below shows the results for each of the experiments with each of the models.

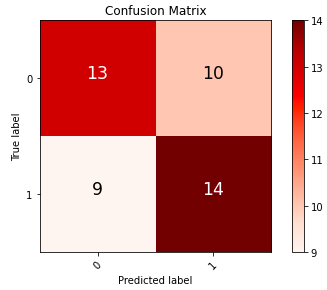
**Table 1:** Results of fear vs anger classification experiments with different models and techniques

|  |  |  |
| --- | --- | --- |
|  | 1D-CNN | LSTM |
| Raw Data | 51.95% accuracy | 52.54% accuracy |
| Transfer Learning | 47.46% accuracy | 48.12% accuracy |
| Scaling | 59% accuracy | 54% accuracy |
| Transfer Learning with Scaling | 46.72% accuracy | 48.13% accuracy |

The experiments showed that scaling provided the biggest improvement, achieving almost 60% overall accuracy with the 1D-CNN model. This is likely due to scaling removing the outliers and unwanted noise, allowing the model to work with the important parts of the data and prevent overfitting. Another takeaway was that in most cases, the 1D-CNN model performed better than the LSTM model, which was a plus considering the CNN model trains much faster than the LSTM model. Unfortunately, for these experiments transfer learning didn’t seem to help. Feature extraction might not be the best approach for our dataset.

Below is the confusion matrix that was printed from one of the classification experiments run using the 1D-CNN model. Remember, zero is labeled for fear, and one for anger. The predicted and true labels matched up more often than not, but there were too many incorrect classifications for our liking. Because the models were accurately predicting around 50% for the raw data, a possible scenario could be that our models cannot learn enough underlying crucial patterns to be able to properly identify each of the classifications, and as a result is almost blindly guessing at the answer. Scaling seems to have helped the models make more than just a random guess, but not enough to make accurate predictions with consistency.

**Figure 9:** Confusion Matrix for the classification using 1D-CNN after scaling was performed



**5.2 Static vs Dynamic Classification**

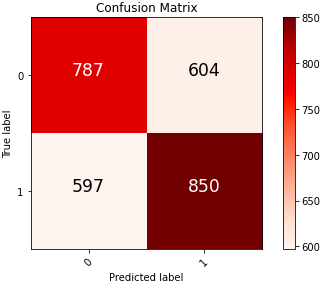
The other main classification task performed was static vs dynamic classification. This was not the primary objective of the project, but it was an informative experiment to see if the models could classify these two better than they could with anger and fear. Using the CNN and LSTM models, along with the same different methods as before, experiments were run and the results are shown on the table below.

**Table 2:** Results of static vs dynamic classification experiments with different models and techniques

|  |  |  |
| --- | --- | --- |
|  | 1D-CNN | LSTM |
| Raw Data | 53.77% accuracy | 50% accuracy |
| Transfer Learning | 52.78% accuracy | 49.94% accuracy |
| Scaling | 57.68% accuracy | 52.36% accuracy |
| Transfer Learning with Scaling | 51.11% accuracy | 48.89% accuracy |

The results showed that the models were not able to perform this classification better, but in fact a little worse. However, similar to before, scaling was the biggest help in achieving a greater accuracy. Furthermore, transfer learning, and in particular combined with scaling, did not help improve accuracies just like the previous classification experiments. The models yet again seem to be almost guessing at which group to classify the data. Shown below is the confusion matrix for one of these experiments, like before more than half were predicted correctly, but also a lot of misclassifications. Remember, zero is the label for dynamic files, and one is for static files.

**Figure 10:** Confusion Matrix for classification using 1D-CNN after scaling was performed



**5.3 Army Research Lab EEG Models**

Seeing that our models did not perform particularly well, we decided to find other existing models to see if they would perform better. After doing some research, we found the Army Research Laboratory (ARL) EEGModels project [7]. This project consisted of a collection of CNN networks built for classifying and processing EEG signals, and was written in TensorFlow. This was perfect for us because we had been building our CNN networks in TensorFlow, meaning their code was compatible and easy to use with our dataset.

The biggest difference was that their project used 2D-CNN models compared to our 1D. To make them compatible with our dataset, we had to add an extra dimension on the last axis to our 3D arrays of EEG data to make them 4D so they could be fed into the 2D-CNN networks. The ARL project consisted of five different 2D-CNN models, each varying in the number of convolution network layers, hyperparameters, and other layers such as dropout or dense layers. We ran experiments with each of these models on our dataset to see if they could perform better than our current models. The results are shown below.

**Table 3:** Results of the experiments with each of the five ARL 2D-CNN models

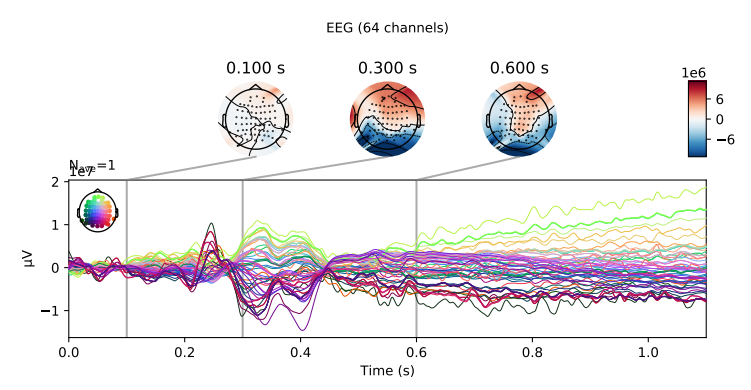
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Old EEGNet Model | New EEGNet Model | EEGNet SSVEP | Shallow ConvNet | Deep ConvNet |
| 49.97% accuracy | 51.93% accuracy | 48.09% accuracy | 51.57% accuracy | 50.52% accuracy |

The results show that the models from the ARL project were unable to do better than our models. This brought up the idea that our models might not be the cause for poor classification, but rather something within our dataset.

**5.4 Visualizations**

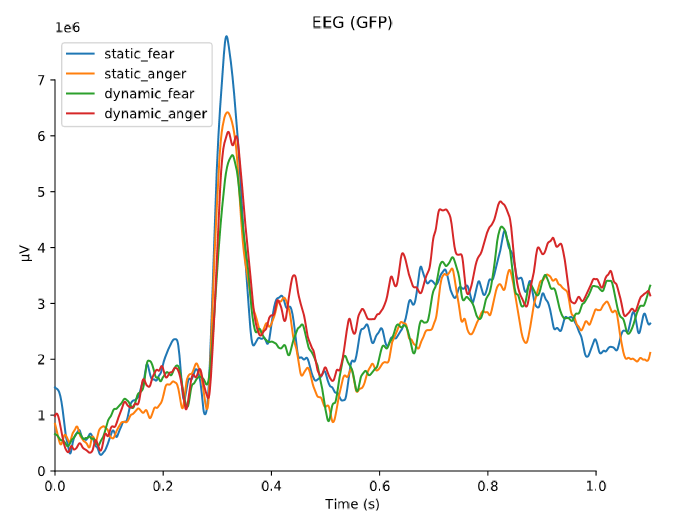
In order to see why our models were having trouble classifying the data, we went through and performed some visualizations to spot any patterns in the data manually. The first visualization performed was a joint plot, which shows both the EEG signals and the scalp topography together for a user overtime. With the scalp topography, it showed which parts of the brain were activated, with red representing high activity levels, and blue representing low activity levels. The times chosen included the 100, 300, and 600 millisecond marks. These exact times were chosen because 100 milliseconds is before the stimulus was shown, 300 milliseconds is around the exact time the stimulus was shown, and 600 milliseconds was after it was shown. In this way, we could see the differences between each of these three time intervals and how the brain activity changes throughout.

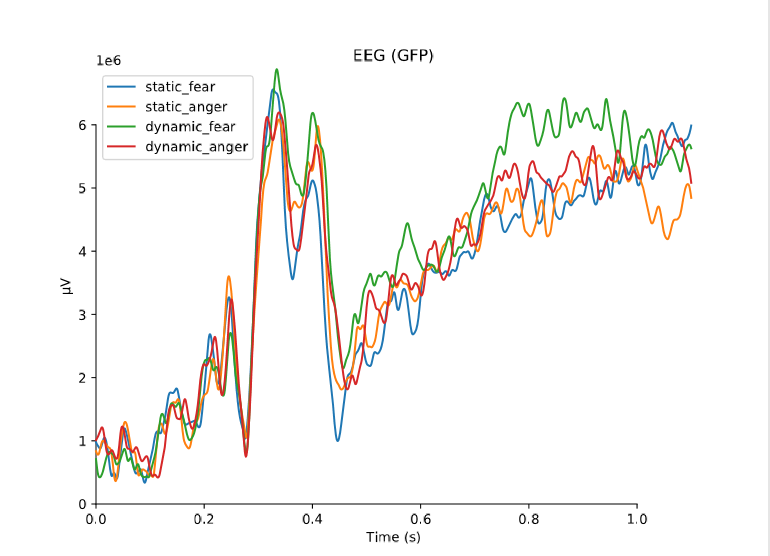
**Figure 11:** Scalp topography combined with a butterfly plot of EEG signals for a particular user, showing brain activity at different time periods throughout the stimulus



Moreover, we plotted the EEG signals of all four stimuli together for each particular user, and it led to our biggest discovery. While the signals differed slightly from person to person, the EEG for each stimulus was very similar for the same user. The graphs showed that there was nothing distinct between the responses to the different stimuli for each of the participants, possibly hinting that the models could not pick up on anything to use to classify between each of them. An example of two of these graphs are shown below.

**Figure 12:** Two different graphs comparing the EEG signals of all four classification types for each user, showing how they differ but yet the four categories are very similar for the same user





**6. Conclusion**

Looking at the EEG signals for each user, it can be seen that for each of the stimuli for that same user, the signals are very similar. Based on this, a conclusion is made that not enough distinction exists between each of the stimuli in order for the deep learning models to be able to distinguish between them. The lack of significant difference between the EEG signals resulted in the deep learning networks not being able to find any patterns to learn and accurately classify between the stimuli. Consequently, the overall validation accuracy was never able to reach above 60% for any of the classification tasks.

A final conclusion is made that fear and anger stimuli, both static and dynamic, are too closely related to be accurately classified by deep learning models. The hope is that with a different dataset that contains largely differing stimuli, such as sadness and happiness, these deep learning models would perform better and achieve much better accuracies. That being said, the next steps in this research include finding another dataset similar to the one that was used, except with different stimuli. Using this new dataset, the goal would be to accurately classify between the stimuli using the models created in the experiments.

**Acknowledgements**

This material is based upon work supported by the National Science Foundation under REU grant #1757893. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

**References**

[1] Blocka, Karla. “EEG (Electroencephalogram): Purpose, Procedure, and Risks.” *Healthline*, Healthline Media, 19 Mar. 2013, www.healthline.com/health/eeg.

[2] “Reading Your Brainwaves: Understanding the Basics of EEG.” *NeuroSky*, 29 Apr. 2015, neurosky.com/2015/04/reading-your-brainwaves-understanding-the-basics-of-eeg.

[3] Lee, You-Yun, and Shulan Hsieh. “Classifying Different Emotional States by Means of EEG-Based Functional Connectivity Patterns.” *PLOS ONE*, Public Library of Science, 17 Apr. 2014, journals.plos.org/plosone/article?id=10.1371%2Fjournal.pone.0095415.

[4] *Harsh Dabas, Chaitanya Sethi, Chirag Dua, Mohit Dalawat, and Divyashikha Sethia. 2018. Emotion Classification Using EEG Signals. In Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence (CSAI ’18). Association for Computing Machinery, New York, NY, USA, 380–384. DOI:https://doi.org/10.1145/3297156.3297177*

[5] *Qingyang Zhu, Guanming Lu, and Jingjie Yan. 2020. Valence-Arousal Model based Emotion Recognition using EEG, peripheral physiological signals and Facial Expression. In Proceedings of the 4th International Conference on Machine Learning and Soft Computing (ICMLSC 2020). Association for Computing Machinery, New York, NY, USA, 81–85. DOI:https://doi.org/10.1145/3380688.3380694*

[6] Idowu, Oluwagbenga Paul, et al. “Towards Control of EEG-Based Robotic Arm Using Deep Learning via Stacked Sparse Autoencoder: Semantic Scholar.” *Undefined*, 1 Jan. 1970, [www.semanticscholar.org/paper/Towards-Control-of-EEG-Based-Robotic-Arm-Using-Deep-Idowu-Fang/2d1320b19f8e9fce80d0efac6d06619ddb8d89f6](http://www.semanticscholar.org/paper/Towards-Control-of-EEG-Based-Robotic-Arm-Using-Deep-Idowu-Fang/2d1320b19f8e9fce80d0efac6d06619ddb8d89f6).

[7] Ibagon, Gabriel. “GabrielIbagon/Arl-Eegmodels.” *GitHub*, 13 May 2019, github.com/gabrielibagon/arl-eegmodels.