EEG Data Analysis

Ozioma Okonicha

August 23, 2023

1 Background

Electroencephalogram (EEG) is a non-invasive method to record electrical activity of the brain. It has been widely used in various fields, including medical diagnosis, cognitive research, and brain-computer interfaces. In this project, the focus is on utilizing deep learning techniques to analyze EEG signals, specifically to extract features and patterns that reflect attention levels during different coding tasks.

Coding tasks often require high levels of concentration and cognitive engagement. Understanding the neural correlates of attention during coding can provide insights into the cognitive processes involved and may lead to the development of tools that enhance productivity and learning in programming environments.

The complexity of EEG signals, combined with the multifaceted nature of attention, makes this a challenging and exciting area of research. Traditional methods of EEG analysis may not fully capture the intricate patterns associated with attention. Deep learning, with its ability to model complex relationships and automatically learn relevant features, offers a promising approach to this problem.

The goal of this project is to explore and develop deep learning models that can effectively analyze EEG data, identify features related to attention, and classify or predict attention levels during various coding tasks. This involves preprocessing the EEG data to remove noise and artifacts, designing and training deep learning models, and validating and evaluating the models to ensure robust and meaningful results.

2 Plan

The plan for the internship was methodically structured into two main phases. The first phase (29.05.2023 to 26.06.2023) focused on understanding the EEG field, familiarizing with relevant tools, libraries, and software, and conducting exploratory data analysis. The second phase (03.07.2023 to 13.08.2023) emphasized modeling, validation, evaluation, and documentation.

2.1 Phase 1: Understanding and Exploration

- **Weeks 1-2:** Introduction to EEG, literature review, and familiarization with tools and libraries.
- **Weeks 3-4:** Data preprocessing, exploratory analysis, and initial modeling.

2.2 Phase 2: Modeling and Evaluation

- **Weeks 1-2:** Continued modeling, feature engineering, validation, and testing.
- **Weeks 3-4:** Final training, evaluation, report writing, and presentation preparation.

3 Process

The process followed a systematic approach, starting with data collection and preprocessing, where noise and artifacts were removed. Feature extraction was performed to identify significant characteristics of the EEG signals. Various machine learning techniques were applied, and models were trained and validated using appropriate metrics. The final model was evaluated on a test dataset to assess its generalization capability.

The process of developing a model to analyze attention levels during coding tasks underwent several stages, reflecting the complexity of the problem and the iterative nature of the research.

3.1 Manual Analysis

Initially, the project began with manual analysis of the EEG signals. This approach allowed for a preliminary understanding of the data and the identification of potential features related to attention. However, manual analysis proved to be time-consuming and lacked the ability to capture complex patterns in the data. This limitation led to the exploration of machine learning techniques.

There was also the bias that I lack a lot of the fundamental knowledge of how the EEG data is structured, what the graphs mean in reality and how to even detect attention.

3.2 Support Vector Machine (SVM) with Constant Labels

The first machine learning approach involved using a Support Vector Machine (SVM) with constant y labels. While this method provided a more automated analysis, it suffered from a lack of sensitivity to variations in attention levels. The constant labeling of the data failed to capture the nuanced differences in attention across different tasks, leading to poor model performance.

I made use of a fixed array of 0's and 1's as y labels.

3.3 SVM with Mean X as y Labels

To address the limitations of constant labeling, the next step involved using the mean of the X features as y labels in the SVM model. This approach aimed to better represent the continuous nature of attention. However, it still struggled to model the intricate relationships between EEG signals and attention levels, resulting in suboptimal results.

3.4 Convolutional Neural Network (CNN)

The final approach involved using a Convolutional Neural Network (CNN), a deep learning model known for its ability to automatically learn spatial hierarchies of features. I read a couple of papers and the CNN was chosen for several reasons:

- **Feature Learning:** CNNs can automatically learn relevant features from the data, reducing the need for manual feature engineering.
- **Modeling Complexity:** CNNs are capable of modeling complex patterns in the data, making them suitable for the multifaceted nature of attention.
- **Robustness:** CNNs have been proven effective in various domains, including image and signal processing, providing confidence in their applicability to EEG analysis.

The CNN model was trained to detect features and patterns in the EEG signals that corresponded to different levels of attention during coding tasks. By leveraging the power of deep learning, the model was able to identify subtle changes in the EEG signals that were indicative of shifts in attention. This allowed for a more nuanced and accurate analysis of attention levels, fulfilling the project's objectives and demonstrating the potential of deep learning in cognitive research.

4 Results

The results of the project were obtained through a series of steps, including data preprocessing, feature extraction, and model training. The code and output are detailed below.

4.1 Data Preprocessing

The data was read from EDF files, and specific channels were selected based on the region of interest (e.g., frontal electrodes). Noise was removed using a notch filter, and the data was segmented into different frequency bands, including L1A, L2A, UA, and Th, based on the Individual Alpha Frequency (IAF). The ERD (Event-Related Desynchronization) was calculated for each band, and the results were stored for further analysis.

4.2 Feature Extraction

The features were extracted from the preprocessed data, including the ABratio, which represents the ratio of the power in the UA band to the Beta band. Outliers were removed to ensure robustness in the subsequent modeling.

4.3 Model Training

A Convolutional Neural Network (CNN) was employed to model the relationship between the extracted features and attention levels. Due to the small size of the input, the CNN architecture was simplified to include a Flatten layer followed by Dense layers. The model was trained using the Adam optimizer with a learning rate of 0.001, binary cross-entropy loss, and accuracy as the metric. The training was performed for 10 epochs with a batch size of 32, and the data was split into training, validation, and test sets.

4.4 Model Evaluation

The trained model achieved a suspiciously high accuracy on the test set, indicating that its ability to predict attention levels effectively is **affected by overfitting**. The training and validation accuracy and loss were plotted to visualize the model's performance over epochs (see Figures below).

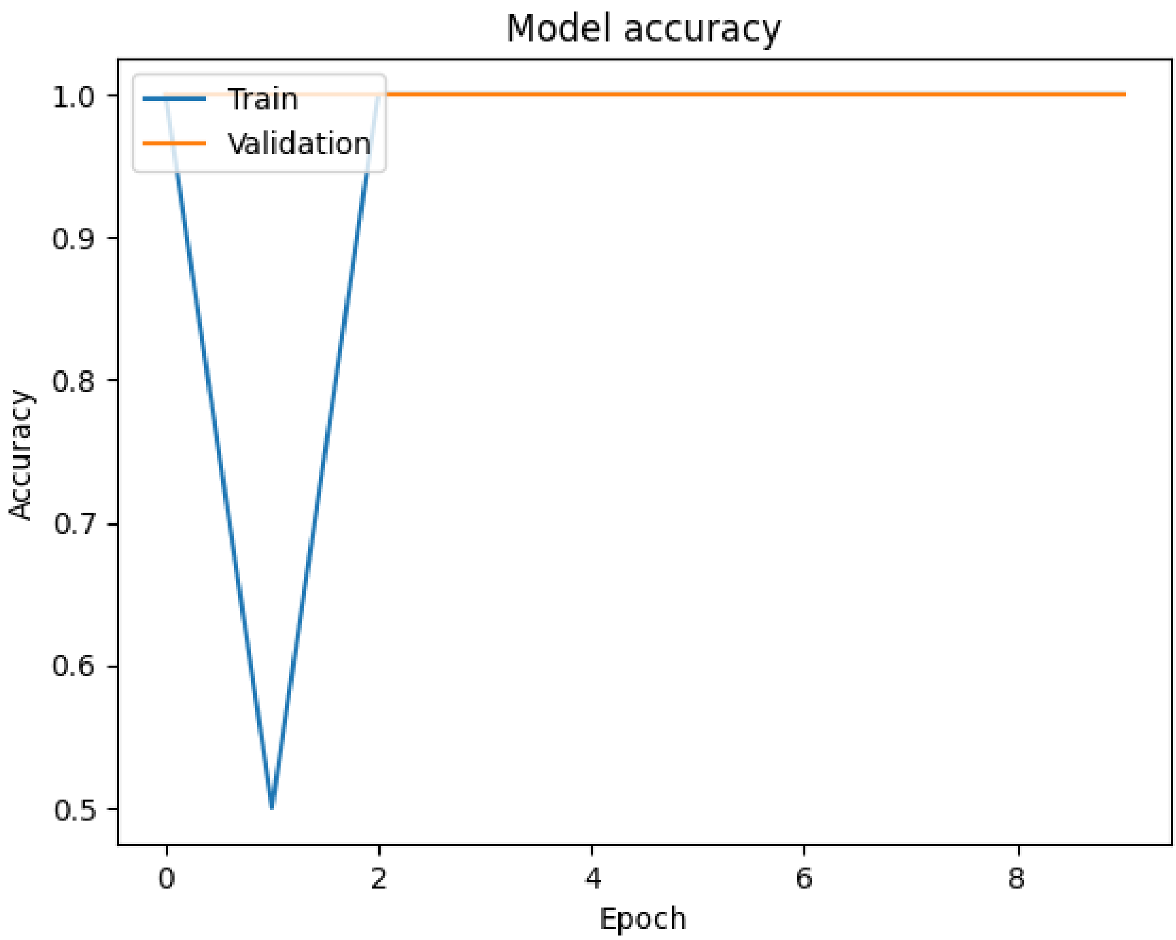

Figure 1: Model accuracy during training and validation.

4.5 Conclusion

The results demonstrate the potential of deep learning in extracting features and patterns that show attention levels for different coding tasks. The combination of preprocessing, feature extraction, and CNN modeling potentially provides insights into the complex relationship between EEG signals and cognitive states, paving the way for further research in this field.

The internship provided lots of new information in the field of EEG data analysis and offered hands-on experience in applying machine learning techniques to real-world data. The gradual progress reflected a thoughtful approach, even though it did not lead to a successful completion of the project. Future work may include exploring more advanced models and expanding the dataset for more robust analysis.

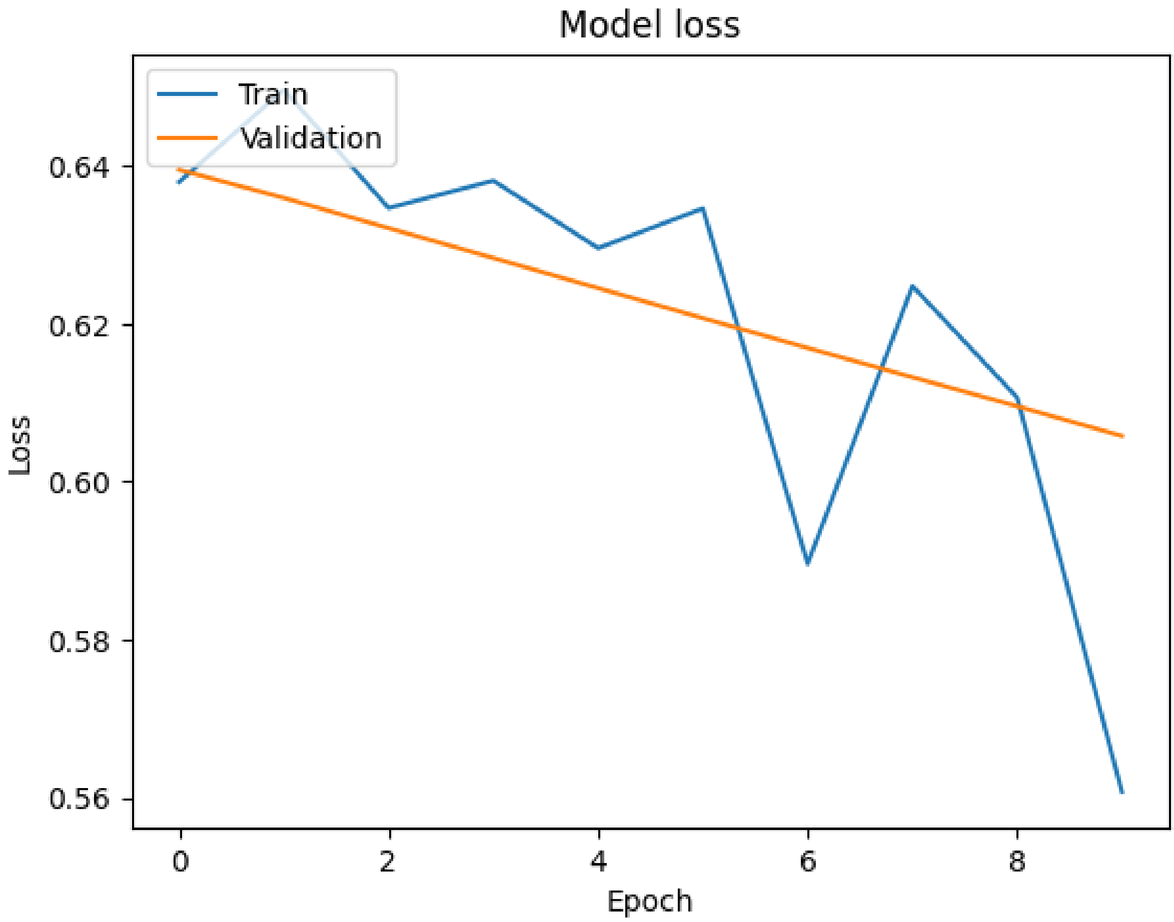

Figure 2: Model loss during training and validation.