EEG Classification Model

1. Introduction

The electroencephalogram (EEG) is a potent diagnostic instrument that records brain electrical activity and offers important insights into neurological disorders and cognitive processes. Different states of consciousness and cognitive processes are reflected by brain waves, which appear as wavy lines on EEG recordings. This project explores the relationship between deep learning and neuroscience with the goal of developing a classification model for EEG data analysis. The main focus is on diagnosing epilepsy, a neurological condition marked by uncontrollable seizures. In addition to being essential for diagnosing epilepsy, EEG data helps to explain a variety of brain conditions and states. The unique patterns that can be seen during seizures, like epileptiform discharges, highlight how useful EEG recordings are for diagnosis.
Tasks related to data preprocessing, feature extraction, data splitting, model selection, training, evaluation, testing, and visualization are all included in the project's progression, which ends with an in-depth investigation of EEG classification for medical applications.

2. Background Information

An EEG is a crucial diagnostic tool for epilepsy and can also aid in diagnosing other brain disorders. Brain waves, reflecting various states of consciousness and cognitive functions, have specific frequency ranges. Delta waves (0.5-4 Hz) are prominent during deep sleep, contributing to restorative processes, while theta waves (4-8 Hz) appear in light sleep, drowsiness, and meditation. Alpha waves (8-13 Hz) signify relaxation, and beta waves (13-30 Hz) dominate wakefulness and cognitive tasks. Gamma waves (30-100 Hz and beyond) are associated with advanced cognitive functions. In epilepsy, EEG captures epileptiform discharges, revealing abnormal neuronal activity for seizure diagnosis. Seizures, abrupt bursts of irregular brain activity impacting consciousness, come in focal and generalized types. Epilepsy, marked by frequent, spontaneous seizures, may result from genetic factors or brain damage, presenting symptoms like convulsions. EEG recordings during seizures unveil distinctive patterns, providing valuable diagnostic insights such as irregular spikes, abrupt waves, and rhythmic activity.

The Bonn dataset consists of 100 single-channel EEG recordings with a spectral bandwidth of 0.5 Hz to 85 Hz. The recordings last 23.6 seconds and are sampled at 173.61 Hz. Multi-channel EEG recordings were extracted into five distinct sets (A, B, C, D, and E) from an original 128-channel acquisition system. Sets A and B show surface EEG recordings from healthy patients with their eyes closed and open, respectively. While set E consists of intracranial EEG during epileptic seizures, sets C and D record intracranial EEG during seizure-free intervals from both inside and outside of seizure-generating areas of epileptic patients. Each set consists of 100 text files containing 4097 ASCII-coded samples of a single EEG time series. Bandpass filtering was applied to the data, with cutoff frequencies set at 40 Hz and 0.53 Hz. Surprisingly, since strong eye movement artifacts have been removed, this artifact-free dataset.

3. Data Preprocessing

The below 3.1 fig, shows the amplitude of the voltage changes in microvolts (mV) over time in seconds. The x-axis goes from 0 to 4000 seconds, while the y-axis goes from -300 to 400 mV.

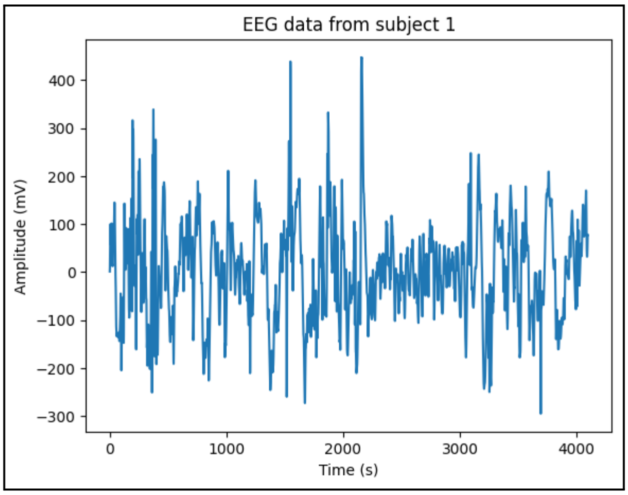


Figure 3.1: Before data preprocessing

A bandpass filter with the Butterworth filter is used to preprocess EEG data, focusing on frequencies between 0.5 and 45 Hz. The script then shows the filtered information for the first subject. It's crucial to remember that thorough analysis and preprocessing of EEG data usually require extra actions. Depending on the precise goals of the analysis, these could involve looking into the features of the data, dealing with missing values, and considering additional techniques for noise reduction.

In fig 3.2, it shows the amplitude of the voltage changes in microvolts (mV) over time in seconds. The x-axis goes from 0 to 4000 seconds, while the y-axis goes from -300 to 400 mV

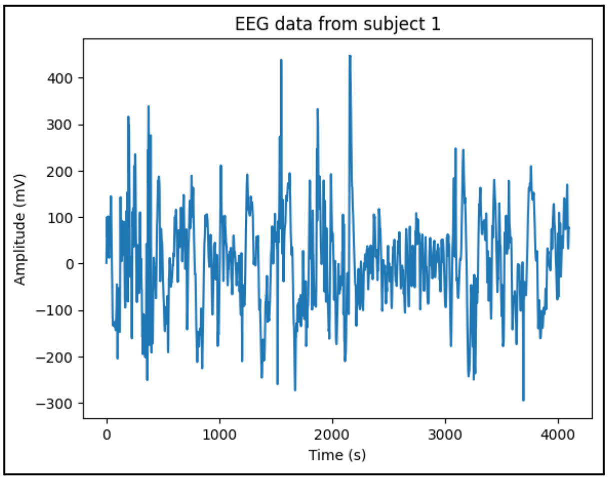


Fig 3.2 After Data Preprocessing

The below Figure 3.3 guarantees consistent feature scaling, it uses `StandardScaler to normalize EEG data. For algorithms that are sensitive to feature magnitudes, this is essential. In order to fully prepare the data for further analysis or deep learning applications, a comprehensive preprocessing approach entails investigating the characteristics of the raw data and possibly addressing missing values, noise, and augmentation.

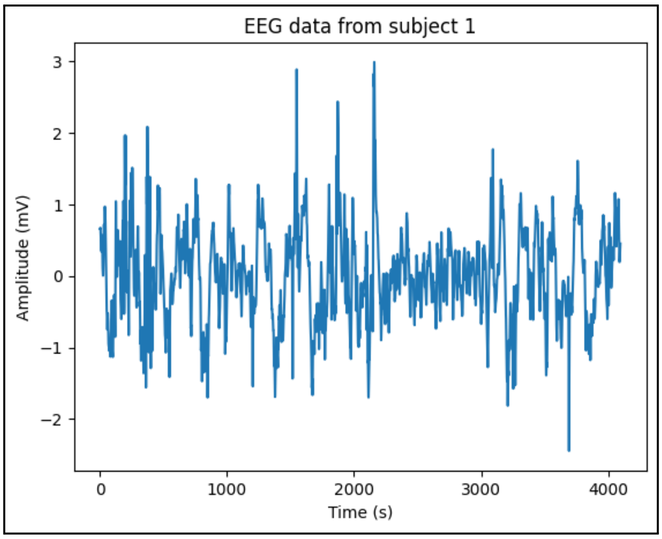
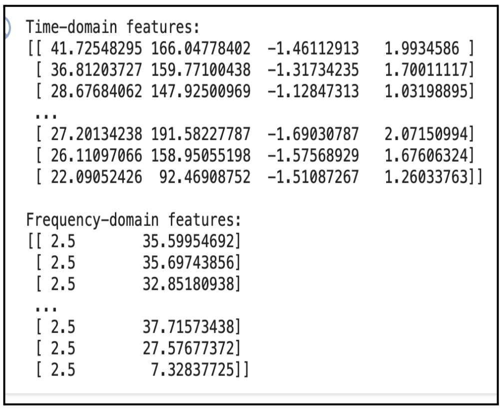


Fig 3.3 After Data Normalization

4. Feature Extraction

Utilizing Welch's method, the feature extraction process yields both frequency-domain (peak frequency, average power) and time-domain features (mean, std, skewness, kurtosis) from EEG signals. The resulting NumPy arrays offer valuable insights into signal characteristics, forming a basis for deeper analyses, particularly in EEG classification tasks.



```
Time-domain features:
[[ 41.72548295 166.04778402  -1.46112913   1.9934586 ]
 [ 36.81203727 159.77100438  -1.31734235   1.70011117]
 [ 28.67684062 147.92500969  -1.12847313   1.03198895]
 ...
 [ 27.20134238 191.58227787  -1.69030787   2.07150994]
 [ 26.11097066 158.95055198  -1.57568929   1.67606324]
 [ 22.09052426  92.46908752  -1.51087267   1.26033763]]

Frequency-domain features:
[[ 2.5         35.59954692]
 [ 2.5         35.69743856]
 [ 2.5         32.85180938]
 ...
 [ 2.5         37.71573438]
 [ 2.5         27.57677372]
 [ 2.5          7.32837725]]
```

Fig 4.1 Feature from EEG signals

The NumPy arrays that are displayed in the above fig 4.1 highlights key characteristics that were taken from EEG signals. Time-domain features that shed light on signal shape and variability include kurtosis, skewness, mean, and standard deviation. On the other hand, signal frequency distribution can be inferred from frequency-domain characteristics like average power and peak frequency. Together, these characteristics create the foundation for a thorough examination, particularly when it comes to classification tasks involving neuroscience.

5. Model architecture

5.1. Data Splitting

The dataset is divided into training (300), validation (75), and testing (125) sets. This partitioning enables effective model training, hyperparameter tuning, and assessment of generalization performance on distinct subsets.

```
Training set size: 300
Validation set size: 75
Testing set size: 125
```

Figure 5.1 Data Splitting

5.2. Model Selection

Optuna is an open-source hyperparameter optimization framework used to find the best set of hyperparameters for an algorithm to improve its performance.

The objective value of the best trial (negative accuracy) and the associated hyperparameters are found. The optimal hyperparameters across the whole training dataset are used to train the final model.

The best trial was trial 6 out of 10 trials which produced a model with a negative accuracy value of -0.96 through hyperparameter optimization using Optuna.

```
[I 2023-12-15 23:54:36,977] A new study created in memory with name: no-name-2a055029-23b8-4337-b9fc-
21695472e707
[I 2023-12-16 00:00:06,790] Trial 0 finished with value: -0.5733333230018616 and parameters: {'filters': 64,
'kernel_size': 5, 'pool_size': 4, 'lstm_units': 123}. Best is trial 0 with value: -0.5733333230018616.
[I 2023-12-16 00:05:49,338] Trial 1 finished with value: -0.8133333325386047 and parameters: {'filters': 114,
'kernel_size': 4, 'pool_size': 2, 'lstm_units': 71}. Best is trial 1 with value: -0.8133333325386047.
[I 2023-12-16 00:14:35,649] Trial 2 finished with value: -0.6666666865348816 and parameters: {'filters': 115,
'kernel_size': 2, 'pool_size': 4, 'lstm_units': 158}. Best is trial 1 with value: -0.8133333325386047.
[I 2023-12-16 00:21:04,422] Trial 3 finished with value: -0.5733333230018616 and parameters: {'filters': 65,
'kernel_size': 5, 'pool_size': 2, 'lstm_units': 74}. Best is trial 1 with value: -0.8133333325386047.
[I 2023-12-16 00:33:42,245] Trial 4 finished with value: -0.5733333230018616 and parameters: {'filters': 101,
'kernel_size': 4, 'pool_size': 4, 'lstm_units': 240}. Best is trial 1 with value: -0.8133333325386047.
[I 2023-12-16 00:44:49,976] Trial 5 finished with value: -0.7333333492279053 and parameters: {'filters': 60,
'kernel_size': 4, 'pool_size': 3, 'lstm_units': 183}. Best is trial 1 with value: -0.8133333325386047.
[I 2023-12-16 00:58:19,537] Trial 6 finished with value: -0.9599999785423279 and parameters: {'filters': 125,
'kernel_size': 3, 'pool_size': 3, 'lstm_units': 198}. Best is trial 6 with value: -0.9599999785423279.
[I 2023-12-16 01:06:41,741] Trial 7 finished with value: -0.5600000023841858 and parameters: {'filters': 83,
'kernel_size': 2, 'pool_size': 4, 'lstm_units': 174}. Best is trial 6 with value: -0.9599999785423279.
[I 2023-12-16 01:17:13,151] Trial 8 finished with value: -0.6666666865348816 and parameters: {'filters': 44,
'kernel_size': 3, 'pool_size': 3, 'lstm_units': 157}. Best is trial 6 with value: -0.9599999785423279.
[I 2023-12-16 01:22:33,796] Trial 9 finished with value: -0.8666666746139526 and parameters: {'filters': 97,
'kernel_size': 3, 'pool_size': 4, 'lstm_units': 116}. Best is trial 6 with value: -0.9599999785423279.

Best trial:
  Value: -0.96000
  Params:
    filters: 125
    kernel_size: 3
    pool_size: 3
    lstm_units: 198
```

Figure 5.2: Hyperparameter Selection using Optuna

The following formula yields the ideal hyperparameters and the model configuration has ideal hyperparameters as follows: There are 125 filters in the convolutional layer, each with a kernel and pooling size of 3. 198 units make up the configuration of the LSTM layers. (Fig 5.3)

During the optimization process, this set of hyperparameters represents the configuration that produced the best accuracy on the validation set. Since the objective of Optuna's optimization is to minimize the objective function, the accuracy value has a negative sign in front of it. In this instance, maximizing accuracy equates to minimizing negative accuracy.

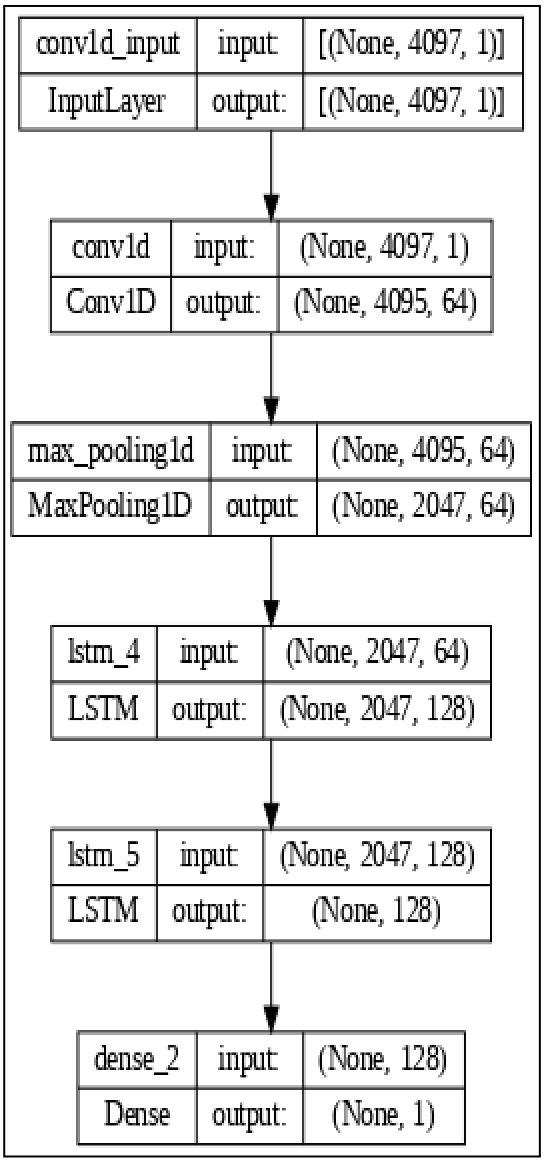


Figure 5.3: Model architecture

6. Model Training

A thorough diagram of the neural network's architecture is shown in Fig. 6.2. The Sequential model, "*sequential_2*" is composed of multiple layers: a MaxPooling1D layer sits after a Convolutional layer with 64 filters and a kernel size of 3. Two LSTM layers are then used, the first with 128 units and the second with an additional 128 units. The model is intended for binary classification tasks, as evidenced by the final layer, a Dense layer with a sigmoid activation function and one output unit. The model has 230,785 total trainable parameters and uses 901.50 kilobytes of memory.

```
Epoch 1/10
8/8 [==============================] - 83s 10s/step - loss: 0.6515 - accuracy: 0.5375 - val_loss: 0.6287 -
val_accuracy: 0.5667
Epoch 2/10
8/8 [==============================] - 78s 10s/step - loss: 0.6294 - accuracy: 0.6250 - val_loss: 0.6199 -
val_accuracy: 0.5667
Epoch 3/10
8/8 [==============================] - 78s 9s/step - loss: 0.5941 - accuracy: 0.6458 - val_loss: 0.6024 -
val_accuracy: 0.6667
Epoch 4/10
8/8 [==============================] - 80s 10s/step - loss: 0.5902 - accuracy: 0.6708 - val_loss: 0.6553 -
val_accuracy: 0.5167
Epoch 5/10
8/8 [==============================] - 79s 10s/step - loss: 0.5923 - accuracy: 0.6125 - val_loss: 0.6106 -
val_accuracy: 0.5333
Epoch 6/10
8/8 [==============================] - 80s 10s/step - loss: 0.5719 - accuracy: 0.6833 - val_loss: 0.5868 -
val_accuracy: 0.6500
Epoch 7/10
8/8 [==============================] - 79s 10s/step - loss: 0.5328 - accuracy: 0.6917 - val_loss: 0.5715 -
val_accuracy: 0.7167
Epoch 8/10
8/8 [==============================] - 80s 10s/step - loss: 0.6081 - accuracy: 0.6333 - val_loss: 0.6404 -
val_accuracy: 0.5333
Epoch 9/10
8/8 [==============================] - 80s 10s/step - loss: 0.5709 - accuracy: 0.6917 - val_loss: 0.6202 -
val_accuracy: 0.5833
Epoch 10/10
8/8 [==============================] - 82s 10s/step - loss: 0.5433 - accuracy: 0.6875 - val_loss: 0.6117 -
val_accuracy: 0.6000

<keras.src.callbacks.History at 0x7820e3080ee0>
```

Figure 6.1: Model training

Given the combination of Convolutional and LSTM layers in this architectural configuration, it is implied that the model can capture both spatial and temporal patterns in the input data. The output shape information provides information about the dimensions at each stage of the network's forward pass, and the comparatively large number of parameters points to a complex model. Overall, the architecture described here is appropriate for applications such as signal processing and specific time series where hierarchical feature extraction from spatial and sequential data is necessary.

```
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d (Conv1D)             (None, 4095, 64)          256

 max_pooling1d (MaxPooling1  (None, 2047, 64)          0
 D)

 lstm_4 (LSTM)               (None, 2047, 128)         98816

 lstm_5 (LSTM)               (None, 128)               131584

 dense_2 (Dense)             (None, 1)                 129

=================================================================
Total params: 230785 (901.50 KB)
Trainable params: 230785 (901.50 KB)
Non-trainable params: 0 (0.00 Byte)
```

Figure 6.2: Model Summary

7. Evaluation Results

In the below Fig 7.1, The model's loss and accuracy evolution over training epochs are shown by the plotted curves. Effective learning is indicated by a steady decrease in the initial high loss. Nevertheless, the accuracy and loss curves level off at epochs 6 and 8, respectively, indicating that there isn't much more training data improvement possible.

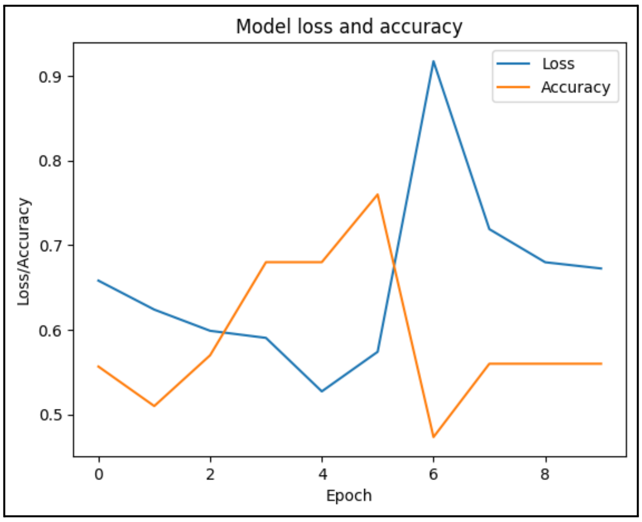


Figure 7.1: Model Evaluation

Below Fig 7.2 is a deep learning model's precision-recall curve, with recall represented by the x-axis and precision by the y-axis. The curve's initial high precision and low recall indicate that the model may be accurate overall, but it may be missing true positives. Precision decreases with recall, suggesting a trade-off. With an AUC of 0.78, the model performs well overall in generating predictions across a range of recall thresholds.

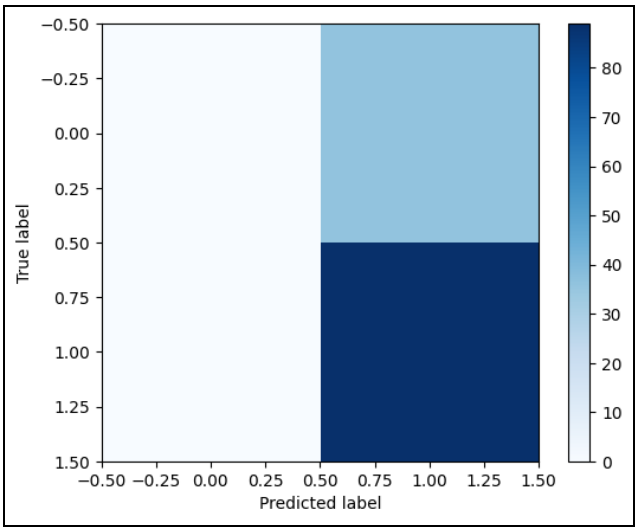


Figure 7.2: Heatmap of the Confusion Matrix

This section is crucial to comprehend the model's performance on the test dataset. The combination of the classification report, confusion matrix, and heatmap visualization provides valuable insights into accuracy, precision, recall, and potential areas for enhancement. Visualizing these metrics aids in interpreting the model's behavior and assessing its effectiveness in classifying instances.

8. Discussion

Significant differences in the two classes' performance are shown by the evaluation results above. With a precision of 71%, recall of 100%, and an F1-score of 83%, Class 1 displays robust classification metrics, demonstrating the model's efficacy in detecting and accurately classifying instances of this class. However, class 0 performance is not at all optimal, with F1-score, recall, and precision all at 0%, suggesting that it is difficult to predict class 0 instances.

The model is successful in classifying instances correctly, as evidenced by the overall accuracy of 71%; however, the macro averages for precision, recall, and F1-score are relatively low, at 36%, 50%, and 42%, respectively. Taking into account the class disparity, the weighted average is 51%, 71%, and 59%. These metrics highlight the need for additional research and possible model improvement, especially addressing the difficulties in class 0 prediction in order to obtain a more balanced and trustworthy overall classification performance.

9. Conclusion

In this study, we developed and trained a neural network model for the classification of EEG data, aiming to distinguish between two classes (0 and 1). The model architecture included Conv1D, MaxPooling1D, and LSTM layers, and the hyperparameters were optimized using the Optuna framework.

The final model achieved an overall accuracy of 71% on the test set, with a precision of 0.71 and a recall of 1.00 for class 1. However, the performance for class 0 was suboptimal, with a precision, recall, and F1 score of 0.00. This suggests that the model struggled to correctly identify instances of class 0, possibly due to class imbalance or other factors.

The macro averages for precision, recall, and F1-score are comparatively low at 36%, 50%, and 42%, respectively, despite the model's overall accuracy of 71%. When class differences are taken into account, the weighted averages rise to 51%, 71%, and 59%. These metrics highlight the need for additional study and possible model improvement, especially when it comes to resolving issues with Class 0 instance prediction.

10. Future Work

In advancing the accuracy of epileptic seizure classification, a multifaceted approach to future research is recommended. Algorithmic exploration should be undertaken, delving into a spectrum of deep learning techniques and model designs beyond the current scope to unearth hidden patterns within EEG signals that could enhance discriminatory capabilities. Feature enhancement is a pivotal avenue, involving a deeper investigation into additional EEG signal characteristics and advanced feature engineering methods to further elevate the model's discriminatory capacity.

Augmenting the dataset through sophisticated data augmentation techniques stands as another critical strategy to improve generalization capabilities. This involves diversifying the dataset using methods such as signal perturbation or synthetic sample generation to expose the model to a more extensive range of scenarios and enhance its robustness.

Ensemble learning strategies offer a promising avenue for future exploration. Integrating predictions from diverse models through ensemble techniques can potentially boost overall classification performance by leveraging the strengths of multiple models.

The effectiveness of transfer learning can be evaluated to enhance the model's ability to identify seizures. Leveraging pre-trained models in relevant tasks with similarities to seizure classification may expedite learning and improve overall performance.

Additionally, incorporating interpretable models into the classification pipeline is crucial for enhancing the explainability of categorization choices. Models with transparent decision-making processes contribute to a deeper understanding of the significance of extracted features, thereby improving interpretability.

Collectively, these proposed directions not only offer opportunities to refine the current model but also hold the potential to advance the broader field of EEG-based seizure classification, ultimately contributing to improved outcomes in healthcare and related domains.