Application of Artificial Intelligence to Resting-State EEG data for the detection of sedation state for patients undergoing Anesthesia

Rithik
Saratoga High School
(Dated: October 7, 2025)

This study explores the creation of an model utilizing artificial intelligence to predict patient drowsiness levels under sedative influence.

I. INTRODUCTION

In the realm of medical procedures, patients require adequate dosages of sedatives in order to maintain levels of consciousness. It becomes increasingly critical to determine the amount of dosages to utilize, as well as to understand patient responses towards such sedatives. Anesthesiologists must rely on relevant data, including that of the electroencephalogram (EEG). EEG data involves reading through a patient's brain wave data. EEG data subsequently demonstrate patient states of drowsiness, which refers to a temporary state of fatigue. Typically, drowsiness results in reduced consciousness and functioning ability. However, under sedation, patients are placed in a state of absolute unconsciousness, traversing this period of drowsiness.

EEG data serves as a means to recognize such data, yet there exists a lack of studies completed with solely resting-state data. When a patient is under sedation, it is impossible to deal with data that is not collected at a resting state, and requires an additional data read while the patient is awake. Means of detecting drowsiness typically also involve Electrooculogram data, utilizing eye blinking to determine drowsiness; facial recognition, pertaining to various facial traits to determine fatigue; response times to activity, which show patterns in patient attentiveness and comprehension ability; and more methods. Using solely EEG resting state data serves as a more novel and representative method to collect data regarding patient drowsiness in a more efficient manner. In this study, we explore the utilization of an artificial intelligence model to aid in the recognition of patient drowsiness from raw EEG data, in accordance with levels of sedation. Typically, the prevalence of lower-frequency waves, such as theta and delta waves, is conventionally associated with a state of drowsiness and sleep. Specifically, we investigate the sedative Propofol, which is known for its rapid activation and deactivation, successfully used in medical settings.

II. DATA COLLECTION AND METHODOLOGY

This study was conducted utilizing EEG data on patient sedation levels in response to propofol from the University of Cambridge, focusing on clinical neurosciences research, conducted by Dr. Chennu et. al. Data was collected from twenty neurologically healthy adults, with a gender distribution of nine males to eleven females. EEG data was accumulated through a ninety-one channel system, which collected information based on specifically targeted propofol. A baseline model was achieved at a fully awake state, while a sedative was administered until the patient was rendered conscious again. Mild sedation was accomplished with a propofol concentration of 0.6 mu g/ml, while moderate sedation was determined with a propofol concentration of 1.2 mu g/ml. As the patient was unconscious, data was continuously collected for seven minutes at four major levels: baseline, mild sedation, moderate sedation, and a recovery period. The study also consisted of other biometrics, including propofol concentration and behavioral responsiveness to certain tasks.

All raw EEG recordings, being time-series data, were segmented into epochs and preprocessed to avoid contamination of artifacts. Data was subsequently analyzed in the study and transferred to a frequency domain. Each epoch was spectrally analyzed to estimate a Power Spectral Density (PSD) for each of the ninety-one channels. This model was critical in demonstrating brain activity based on the distribution of power across various frequency bands. A spectral graph was produced, in which synchronized activity was displayed alongside each frequency band, providing information and estimates based on band frequency and appearance. The mean and standard deviation of the spectral power were computed as well, which was collected in Table 1.

TABLE I. Canonical Frequency Bands and Their Corresponding Spectral Ranges.

Frequency Band	**Symbol**	**Frequency Range (Hz)**
Delta	δ	$0.5 - 4$
Theta	θ	$4 - 8$
Alpha	α	$8 - 13$
Beta	β	$13 - 30$

In this study, a model was developed through a Random Forest Classifier (RFC) algorithm, serving as the primary source of modeling. This algorithm, characterized by decision trees that converge towards a final decision, is an efficient method that avoids overfitting through a large collection of decision trees. Each tree consists of varied data and establishes a final prediction upon running through each tree's decisions. The model was configured with specific parameters, including 200 estimators, a minimum sample to split an internal node of 5, a minimum of 2 samples per leaf node, and a maximum tree depth of 20. Each data feature was normalized through a StandardScalar, making all features equally comparable in a range from 0 to 1. Before segmenting the data into epochs, we must process all data stored as MATLAB files. We can utilize an HDF5 format reader to read through the data, which has been applied

with a high-pass filter above 0.5 Hz. A notch filter was also subsequently employed to eliminate any interference from electrical lines.

The prediction of sedation level was approached through an ordinal classification method, under each of the categories from the original data set, such as Baseline and Recovery, which follow a chronological order. In this way, errors of misclassifying the prediction at a different state than reality is avoided. As the scale of misclassification varies, an ordinal loss metric was implemented to determine the distance of the prediction from reality, as shown in Eq. (1).

$$L_{\text{Ordinal}} = \frac{1}{N} \sum_{i=1}^{N} |\text{Index}(y_i) - \text{Index}(\hat{y}_i)| \tag{1}$$

The ordinal loss metric helps address the order of sedation categories. The index function matches each of these categories to a respective integer value and ensures that a larger misclassification between steps is followed by a larger error. The function subsequently maps out the distance between the predicted level, denoted by $\hat{y}_i$, and the actual level, denoted by y_i. Furthermore, the model was refined further by focusing on the time-series aspects of the data. As the brain transitions to each respective state, data follows a temporal pattern, which is divided up by each individual epoch into small segments. To address the challenge of limited EEG labeled data, a synthetic component of the dataset was generated, ensured to mimic the same properties of the EEG features through a technique of feature peturbation. This technique artifically doubles the data set size, with ach feature dimension was covered through Gaussian sampling. A synthetic copy of the data was generated alongside the eight spectral features such as mean and standard deviation power. Noise was inputted in a controlled manner, uniformly sampled between 0.95 and 1.05, w to replicate data variability with a variation of ±5%. This was assigned through the same ordinal thresholds established above to each respective band frequency, to model real clinical data to the high level of accuracy.

Epoching was also further moved through a sliding-window approach, specifically with 30 seconds each with 50 percent overlap. Each window was shifted forward in time with the designated overall of 50 percent. By employing this approach, we assume that the brain state is quasi-stationary within the bounds of the window's duration. The overlap is utilized to prevent changes or major shifts from occurring between each segment. With this large collective of small samples, we are able to augment the overall size of the dataset. This methodology is essential for transforming the long, continuous signal into a set of discrete, overlapping epochs suitable for machine learning classification.

Upon reading all the data, we can compute the mean and standard deviation for each frequency band demonstrated in Table 1. We can utilize Welch's method as a suitable means to estimate the PSD of the data, as seen in Fig. 1. This method, a non-parametric technique, can average out data to generate a smooth estimate for the PSD. The raw EEG data is split into small overlapping segments, and a Fast Fourier Transform (FFT) is utilized to each segment to move the signal to a frequency domain. Each of these PSD estimates, known as periodgrams, are then averaged out to estimate a solid PSD. Upon this estimation, feature vectors can be derived through the power of each frequency band value. The mean power reflects the magnitude of the feature vectors, while the standard deviation reflects the spatial variability of the specific band's power over each of the channels. The entire process generates an eight-dimensional feature vector for each window. Each feature vector covered a specific period of time while remaining continuous over each overlapping segment. Autoregressive features were also opted, in the hope to utilize previous predictions in later windows and inputs. This ensured a certain degree of temporal consistency, creating predictions more closer in distance to the actual sedation category.

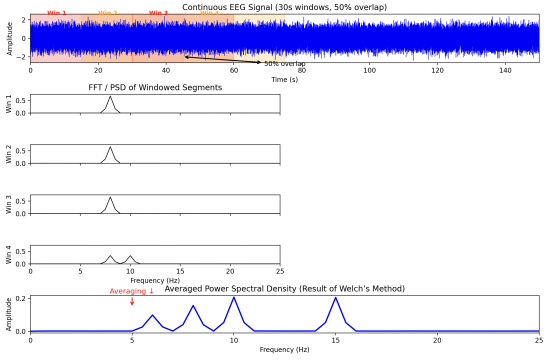

FIG. 1. A visualization of Welch's Method on segmented windows 30 seconds each with a 50 percent.

The dataset was split into a train-test split of 75-25, respectively. The training set was utilized in the development of the model and through hyperparameter tuning, and the test set was kept for a final evaluation. In the training phase, a

5-fold cross validation was applied to test against overfitting. Such a validation metric is used to determine the model's ability to generalize a particular data set. It divides the training data set into five non-overlapping segments of equal sizes, known as a fold. On each iteration, metrics are outputted corresponding to each fold, including accuracy and precision. Based on the prediction target, we can consider two approaches to the Random Forest model: a classical and regressive approach. Both cases can be optimized through the 5-fold cross validation along side a grid search. This grid search method can be used for hyperparameter tuning specifically, by evaluating the best combination of values that optimize model performance. For a classification approach, a Random Forest Classifier can be trained with this grid of parameters, matched with a varied number of trees, restrictions for splitting nodes, number of samples, and maximum depth of trees, in which the best scenario would determine the model's best performance standard. For regressions, a Random Forest Regressor can be used with a grid again, this time basing results on a mean squared error. This mean squared error, visualized in Eq. (2), is a metric that demonstrates the difference between true values and estimated values. n is used to denote the total number of samples, while $\hat{y}_i$ is the predicted value and y_i is the actual category the data is in at a specific point.

$$L_{\text{MSE}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

The mean squared error can be utilized as a criteria on which feature to retain for each split while training a decision tree. The tree with the least reduction of MSE is chosen as the split to minimize any variance in each leaf node of a tree. For both the regression and classification method, the best performing model was determined by a set of five precision factors, which overall can be averaged to produce a final generalization. For this model, we utilized accuracy, precision, recall, and F1-scoring as relevant performance metrics. While accuracy provided a main measure for the correctness of each prediction, we utilized the other metrics to better evaluate the model's overall reliability across each specific sedation level.

III. RESULTS AND DISCUSSION

Upon running the algorithm, we can report an overall accuracy of around 95% for the model. Each performance metric is visualized in Fig. 2, upon averaging each metric to provide a generalization.

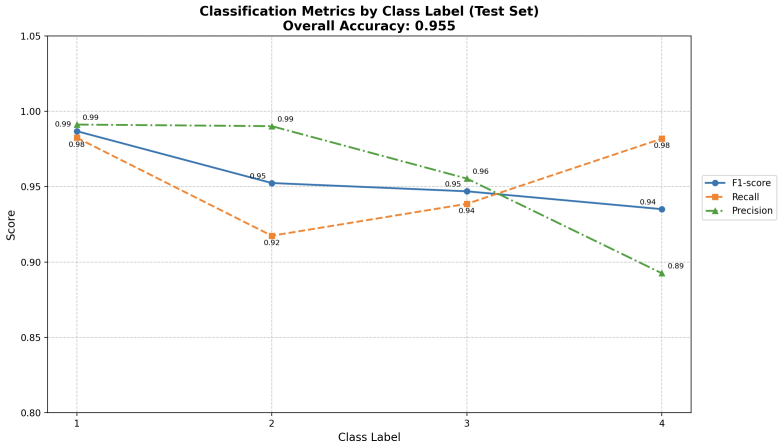

FIG. 2. A plot visualization of all final averaged performance metrics, including accuracy, precision, recall, and F-1 scoring

We can use a confusion matrix to visualize the performance of the model, seen in Fig. 3.

IV. CONCLUSION

V. REFERENCES

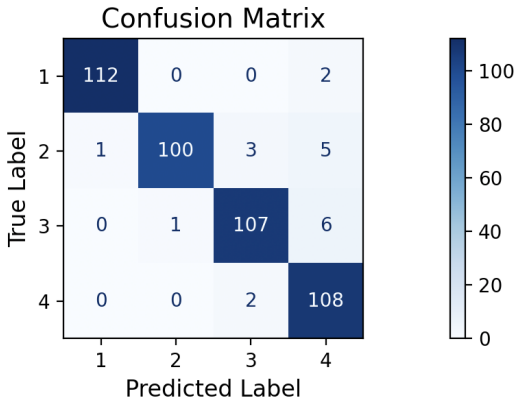

FIG. 3. A confusion matrix illustrating numbers of false positives and negatives in accordance to each respective performance metric