Pattern Recognition: Mid-term Report

Mental Attention States Classification Using EEG Data

Lecturer: PhD. Ngo Minh Man
TA: MSc. Le Hoang Duc
Group 3

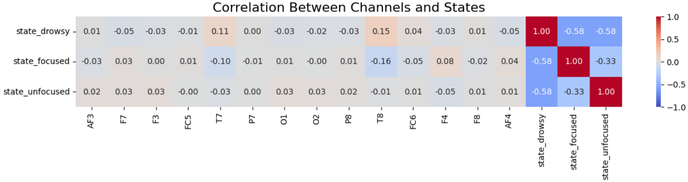
No	Name	ID
1	Tran Nguyen Trung Tuan	22280101
2	Dang Minh Phuc	22280064
3	Le Hong Cat	21110249
4	Truong Minh Hoang	22280034

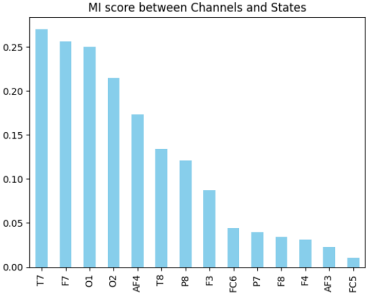
April 29, 2025

Content

Content

The importance of Channels versus States





$\Rightarrow$ **T8** and **T7** channels are important for detecting **drowsy** and **focused**.

The importance of Channels versus States

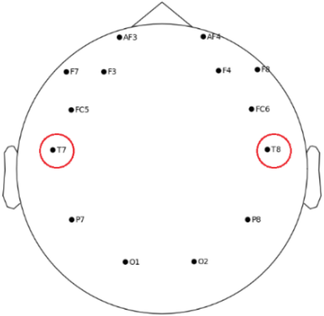


Figure: Position

T7 and **T8** are located on both sides of the temples.
⇒ Therefore, it can be said that the temple position is important for **classifying states** in general.

Pipeline

We used the following pipeline for the mental attention state detector:

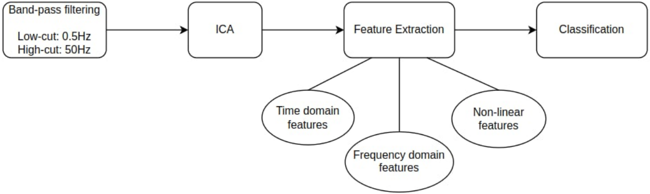


Figure: Pipeline for the mental attention state detector

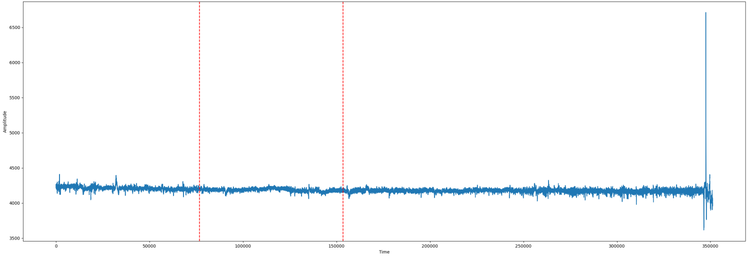
Content

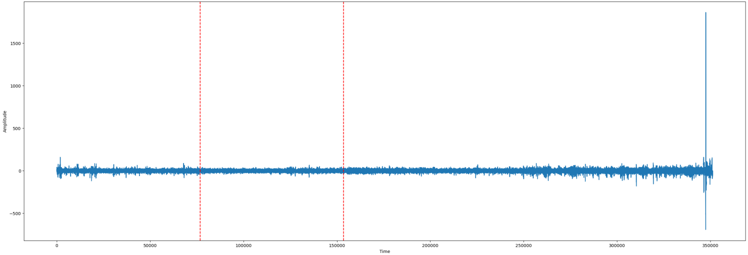
Bandpass Filtering

- Low cut value: 0.5 Hz; high cut value: 50 Hz

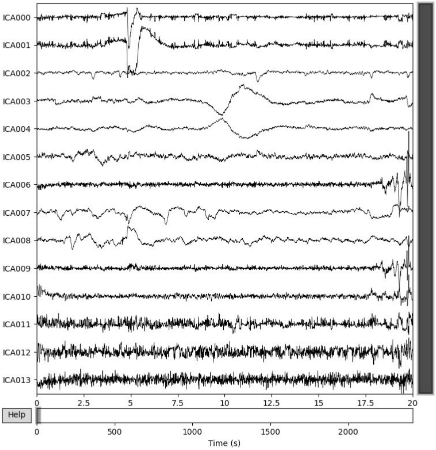
Bandpass Filtering

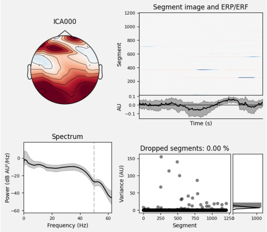
- Low cut value: 0.5 Hz; high cut value: 50 Hz
- Before and after filtering on channel P8, eeg_record5.mat



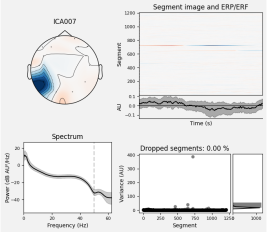


Independent Component Analysis



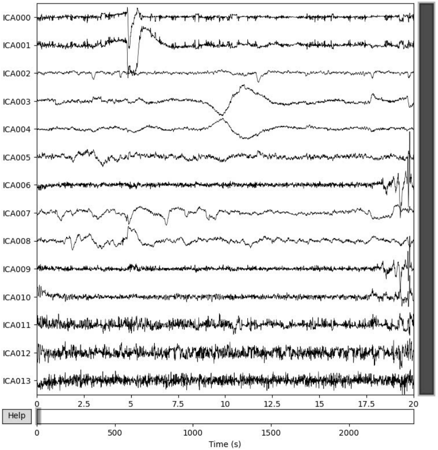


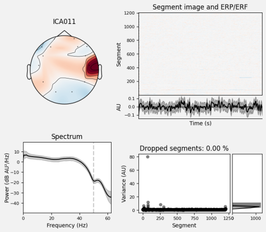
Noise from **eye activities**



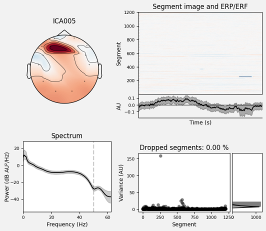
Noise from **heart rhythm**

Independent Component Analysis





Noise from **electrical line interference**

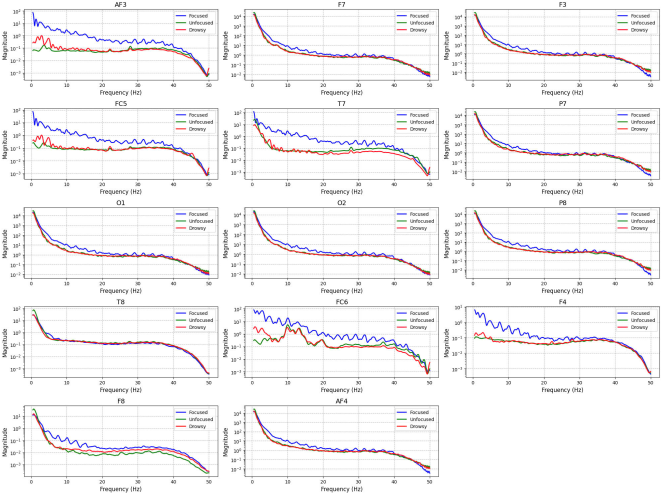


Noise-free

Content

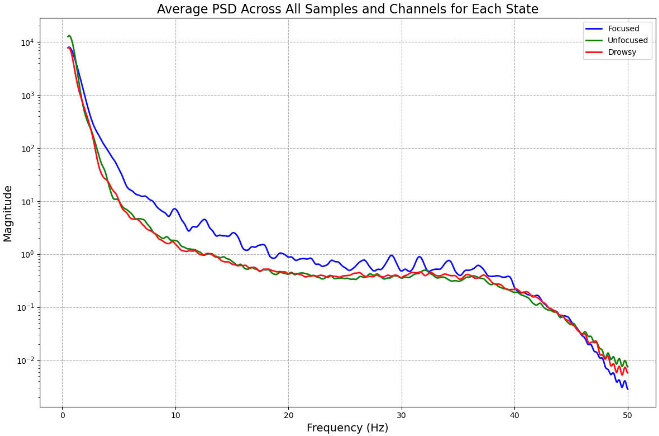
Feature Extraction - Frequency-domain features

Power spectrum of five different bands included δ: [0.5, 4) *Hz*, θ: [4, 8) *Hz*, α: [8, 12) *Hz*, β: [12, 30) *Hz*, and γ: [30, ∞) *Hz*.



Feature Extraction - Frequency-domain features

Power spectrum of five different bands included δ: [0.5, 4) Hz, θ: [4, 8) Hz, α: [8, 12) Hz, β: [12, 30) Hz, and γ: [30, ∞) Hz.



Feature Extraction - Time-domain features

- Zero-crossing rate
- Waveform Length
- Statistical Features: Mean, Standard Deviation, Skewness, Kurtoisis.

Feature Extraction - Non-linear features

Combination of power in different frequency bands.

Index	Formula	Description
1	δ	Delta band frequency power
2	θ	Theta band frequency power
3	α	Alpha band frequency power
4	β	Beta band frequency power
5	γ	Gamma band frequency power
6	$\alpha/(\delta+\theta+\alpha)$	A feature for attention detection (Olbrich et al., 2009; [67])
7	$(\theta+\beta)/(\alpha+\gamma)$	A feature for attention detection [67]
8	$(\alpha+\beta)/(\delta+\theta+\gamma)$	A feature for attention detection [67]
9	β/δ	A feature for attention detection (Morillas-Romero et al., 2015; [67])
10	θ/β	A feature for attention detection (Angelidis et al., 2018; Morillas-Romero et al., 2015; [67])
11	$\beta/(\theta+\alpha)$	A feature for attention and fatigue detection (Eoh et al., 2005; [67])
12	α/β	A feature for attention and drowsiness detection (Jap et al., 2009; Liu et al., 2013; [67])
13	β/θ	A feature for drowsiness and attention detection (Jap et al., 2009; [67])
14	$\alpha/(\beta+\gamma)$	A feature for attention detection [67]
15	$(\alpha+\beta)/\gamma$	A feature for attention detection [67]
16	β/α	A feature for attention and fatigue detection (Eoh et al., 2005; [67])
17	$(\alpha+\beta)/(\alpha+\theta)$	A feature for drowsiness and attention detection (Jap et al., 2009; [67])

Feature Extraction - Non-linear features

- **Entropy-based**:

$$\text{Spectral Entropy} = -\sum_{i=1}^{N} p_i(w) \log_2 p_i(w)$$

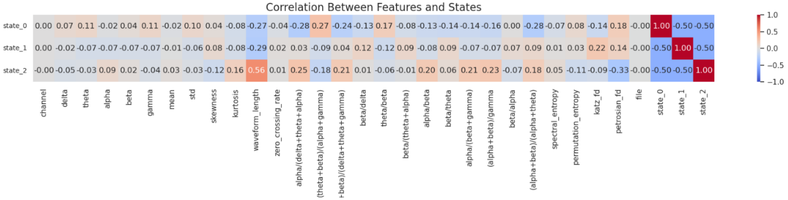
$$\text{Permutation Entropy} = \sum_{i=1}^{N} p_i \log_2 p_i$$

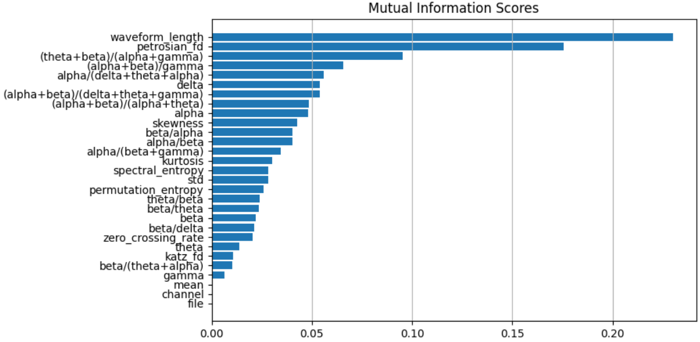
- **Fractal Dimension**:

$$\text{Katz FD} = \frac{\log L - \log s}{\log d - \log s}$$

$$\text{Petrosian FD} = \frac{log_{10} n}{log_{10} n + log_{10}(\frac{n}{n+0.4N_{\Delta}})}$$

Feature Importance - Machine Learning Dataset





Imbalance Handling

- **Machine Learning Dataset:** Computing the power of a channel across three subsets and creating new data based on three states make the proportion of each state in the new dataset is 1:1:1.

Imbalance Handling

- **Machine Learning Dataset:** Computing the power of a channel across three subsets and creating new data based on three states make the proportion of each state in the new dataset is 1:1:1.
- **Deep Learning Dataset:** Using ROS (Random Over Sampling) to improve model and spliting the training/testing/validation set to 0.68:0.2:0.12.

Content

Basic Machine Learning

- We totally used five models. including: Support Vector Machine (SVM), k-Nearest Neighbors (kNN), Logistic Regression, and CatBoost. We decided to use SVM model as a baseline model.

Basic Machine Learning

- We totally used five models. including: Support Vector Machine (SVM), k-Nearest Neighbors (kNN), Logistic Regression, and CatBoost. We decided to use SVM model as a baseline model.
- Before applying model, the dataset is splitted into train set, test set, and validation set, which is used for hyper-parameters tuning.

Basic Machine Learning - Hyper-parameters

- **Logistic Regression** hyper-parameters:

Hyper-parameters	C	Penalty	Solver
Value	10	l1	liblinear

Basic Machine Learning - Hyper-parameters

- **Logistic Regression** hyper-parameters:

Hyper-parameters	C	Penalty	Solver
Value	10	l1	liblinear

- **Support Vector Machine** hyper-parameters:

Hyper-parameters	C	Gamma	Kernel
Value	10	scale	rbf

Basic Machine Learning - Hyper-parameters

- **Logistic Regression** hyper-parameters:

Hyper-parameters	C	Penalty	Solver
Value	10	l1	liblinear

- **Support Vector Machine** hyper-parameters:

Hyper-parameters	C	Gamma	Kernel
Value	10	scale	rbf

- **Random Forest** hyper-parameters:

Hyper-parameters	Max depth	Min s.leaf	Min s.split	N est
Value	30	2	8	16

Basic Machine Learning - Hyper-parameters

- **Logistic Regression** hyper-parameters:

Hyper-parameters	C	Penalty	Solver
Value	10	l1	liblinear

- **Support Vector Machine** hyper-parameters:

Hyper-parameters	C	Gamma	Kernel
Value	10	scale	rbf

- **Random Forest** hyper-parameters:

Hyper-parameters	Max depth	Min s.leaf	Min s.split	N est
Value	30	2	8	16

- **kNN** hyper-parameters:

Hyper-parameters	Metric	N neighbors	Weights
Value	euclidean	3	distance

Basic Machine Learning - Hyper-parameters

- **Logistic Regression** hyper-parameters:

Hyper-parameters	C	Penalty	Solver
Value	10	l1	liblinear

- **Support Vector Machine** hyper-parameters:

Hyper-parameters	C	Gamma	Kernel
Value	10	scale	rbf

- **Random Forest** hyper-parameters:

Hyper-parameters	Max depth	Min s.leaf	Min s.split	N est
Value	30	2	8	16

- **kNN** hyper-parameters:

Hyper-parameters	Metric	N neighbors	Weights
Value	euclidean	3	distance

- **CatBoost** hyper-parameters:

Hyper-parameters	Depth	Iterations	Learning rate
Value	4	200	0.05

Basic Machine Learning - Result

Evaluation score of five models:

	Accuracy	**F1-Score**	**Precision**	**Recall**
Logistic Regression	0.79	0.79	0.79	0.79
SVM	0.84	0.85	0.84	0.85
Random Forest	0.97	0.97	0.97	0.97
CatBoost	0.85	0.85	0.85	0.85
kNN	**1**	**1**	**1**	**1**

Deep Learning

- Deep learning is a powerful tool in machine learning. We use propose the ResNet model for training and evaluation on the raw dataset.

Hyper-parameters	**Value**
Learning rate	1e-4
Batch size	64
Number of epochs	40
Optimizer	Adam
Scheduler	ReduceLROnPlateau (mode=min; patience=3)
Weight Decay	1e-5
Encoder model	ResNet

Deep Learning

- Deep learning is a powerful tool in machine learning. We use propose the ResNet model for training and evaluation on the raw dataset.
- The model consists of an initial convolution layer, three residual layers with blocks for feature learning, then dimensionality reduction using adaptive average pooling, and finally a fully connected layer for classification results.

Hyper-parameters	**Value**
Learning rate	1e-4
Batch size	64
Number of epochs	40
Optimizer	Adam
Scheduler	ReduceLROnPlateau (mode=min; patience=3)
Weight Decay	1e-5
Encoder model	ResNet

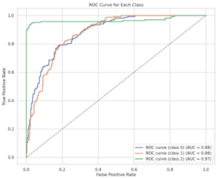
Deep Learning - Result

- We obtain the following result:

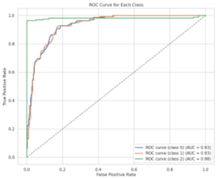
	Accuracy	**F1-Score**	**Precision**	**Recall**
Validation	0.86	-	-	-
Testing	0.85	0.85	0.85	0.85

ROC Curve

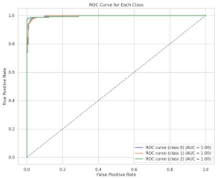
- We obtain the following result:



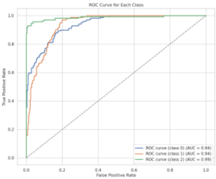
Logistic Regression



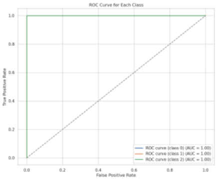
SVM



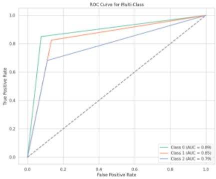
Random Forest



CatBoost



kNN



CNN

Thank you for listening!