Monitoring Attention Span Dynamics in Online Classes Using Electroencephalography

by

Jannatul Ferdaus
21301062
Faizah Binte Mahshudul Huq
21301054
Tangena Islam
21301105
Sakib Hasan Tasin
21301161

A thesis submitted to the
Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
February 2025

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Jannatul Ferdaus	Faizah Binte Mahshudul Huq
21301062	21301054
Tangena Islam	Sakib Hasan Tasin
21301105	21301161

Approval

The thesis titled "Monitoring Attention Span Dynamics in Online Classes Using Electroencephalography" submitted by

1. Jannatul Ferdaus (21301062)
2. Faizah Binte Mahshudul Huq (21301054)
3. Tangena Islam (21301105)
4. Sakib Hasan Tasin (21301161)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on February 07, 2025.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Rafeed Rahman

Senior Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Associate Professor; Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

This study investigates the utilization of electroencephalography (EEG) for monitoring attention dynamics in online classrooms. With the widespread integration of virtual learning due to the impact of COVID-19 pandemic, managing student engagement has become increasingly crucial. Through EEG technology, educators can gain access to real-time data on fluctuations in student attention levels, facilitating timely adjustments in teaching methodologies. By gathering EEG data from students participating in various online learning activities including live lectures, interactive discussions, and multimedia presentations, and analyzing these signals, we can track fluctuation in attention. A multimodal classification method is used and different classifiers such as Bi-LSTM, Bi-GRU, Multi-Head Attention Mechanism, Vision Transformer (ViT), Contrastive Learning etc are used to detect the attention levels efficiently. This allows us to identify periods of sustained focus as well as moments of distraction. Additionally, we investigate how factors such as lecture style, instructor engagement, and external distractions impact attention span dynamics. Consequently, our research provides valuable insights into the cognitive mechanisms governing attention in online learning, offering a promising approach for real-time monitoring and intervention to create a more engaging online educational experience.

Keywords: Electroencephalography (EEG), Attention Span, Monitoring Students, Efficient E-learning, Brainwave signal analysis, Adaptive teaching tools

Acknowledgement

With sincere gratitude, we start by praising and thanking Allah, the Most Merciful for granting us the patience and wisdom for completing this thesis.

We would like to express our deepest gratitude to Dr. Md. Golam Rabiul Alam sir, our thesis supervisor, for his invaluable guidance, support, and constructive feedback throughout this journey. His vast experience and knowledge have been pivotal in guiding this research to its completion and enriching the quality of this work.

We are also profoundly thankful to our co-supervisor, Mr. Rafeed Rahman sir, and the research assistant, Nafiz Imtiaz Rafin sir, for their collaborative efforts, insightful suggestions, and overall support.

Lastly, we are deeply thankful to our family and friends for their unwavering moral support, encouragement, and prayers throughout this journey. Their belief in us has been a constant source of strength and motivation.

Table of Contents

Declaration i

Approval ii

Abstract iv

Acknowledgement v

Table of Contents vi

List of Figures viii

List of Tables ix

Nomenclature ix

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Problem Statement . . . 2
- 1.3 Research Objective . . . 2
- 1.4 Research Contributions . . . 3

2 Literature Review 4

3 Background Study 11
- 3.1 Description of Models . . . 11

4 Methodology 14
- 4.1 Dataset Description . . . 17
- 4.2 Data Preprocessing . . . 18
  - 4.2.1 Butterworth Bandpass Filtering . . . 18
  - 4.2.2 StandardScaler . . . 20
  - 4.2.3 Data transformation and augmentation . . . 20
- 4.3 Feature Extraction . . . 21
- 4.4 Train-Test Split . . . 23

5 Implementation and Result Analysis 25
- 5.1 PC configuration for training models . . . 25
- 5.2 Model Implementation . . . 25
  - 5.2.1 Approach 1 . . . 26

5.2.2 Approach 2 . . . 26
5.2.3 Approach 3 . . . 27
5.3 Performance Evaluation . . . 30
5.4 Results Analysis . . . 31
5.4.1 Approach 1 . . . 31
5.4.2 Approach 2 . . . 34
5.4.3 Approach 3 . . . 37
5.4.4 Result Comparison with Previous Research . . . 51

6 Conclusion **52**
6.1 Conclusion . . . 52
6.2 Limitation and Future Work . . . 52

Bibliography **55**

List of Figures

4.1 Top-level overview of the method 16
4.2 Focused State Signal Before and After Filtering 19
4.3 Unfocused State Signal Before and After Filtering 19
4.4 Drowsy State Signal Before and After Filtering 20
4.5 Original Vs Augmented Image 21
4.6 Frequency Bands of Focused State 22
4.7 Frequency Bands of Unfocused State 23
4.8 Frequency Bands of Drowsy State 23

5.1 Correlation between EEG Channels and Attention State 26
5.2 Confusion Matrices Using AdaBoost 32
5.3 Confusion Matrices Using XGBoost 33
5.4 Accuracy Comparison Between AdaBoost and XGBoost 34
5.5 Accuracy and Loss Graphs for Bi-LSTM Model 35
5.6 Accuracy and Loss Graphs for Bi-GRU Model 36
5.7 Accuracy and Loss Graphs for Bi-LSTM Fusion Model 38
5.8 Confusion Matrix using Bi-LSTM Fusion Model 39
5.9 Accuracy and Loss Graphs for Bi-GRU Fusion Model 40
5.10 Confusion Matrix using Bi-GRU Fusion Model 41
5.11 Accuracy and Loss Graphs for ViT 42
5.12 Confusion Matrix of ViT 43
5.13 Visualization of True Vs Predicted labels using ViT 44
5.14 Accuracy and Loss Graphs for Contrastive Learning 45
5.15 Confusion Matrix of Contrastive Learning 46
5.16 Visualization of True Vs Predicted labels using Contrastive Learning 46
5.17 Confusion Matrices for Bi-LSTM fusion and ViT. 47
5.18 Confusion Matrices for Bi-GRU fusion and ViT. 48
5.19 Confusion Matrices for Bi-LSTM fusion and Contrastive Learning. . . 49
5.20 Confusion Matrices for Bi-GRU fusion and Contrastive Learning. . . 49
5.21 Accuracy Comparison of Multimodal Fusion using Different Combinations 51

List of Tables

4.1 Dataset Summary 18

5.1 PC Configuration 25
5.2 Accuracy Comparisons for AdaBoost and XGBoost on Each Participants 31
5.3 Performance comparison of Bi-LSTM and Bi-GRU 37
5.4 Performance comparison of Bi-LSTM Fusion and Bi-GRU Fusion . . 41
5.5 Performance comparison of ViT and Contrastive Learning models with attention states. 47
5.6 Performance Comparison of Multimodal Fusions using Different Combinations 50

Chapter 1

Introduction

In this section, we discussed the difficulties faced by teachers in monitoring students' attention in online classes. This motivated us to conduct our research, which addresses this issue. This chapter also encompasses our research objectives and contributions.

1.1 Motivation

With the constant advancement of technology, the education system went through a dramatic change in recent years which made it easier for the learners to have access to any educational resources via any digital device. The transition was even more drastic when the world seemed to come to a halt due to Covid-19 and the academic system had to shift to virtual learning, abandoning the traditional in-person education system. Despite many benefits of online learning like flexibility, reduced cost, and time of traveling, both the educators and students seemed to face a few challenges that came with this drastic change. One of those issues was the difficulty in maintaining the attention of students during online learning which was difficult for the instructors to notice in the online setting.

In simple words, attention span is the duration of time in which a person can retain their focus on a task without being distracted. In traditional in-person classrooms, the instructors can easily observe the students and they are able to tell if a student is distracted. Additionally, they can effectively use various strategies including asking questions, having interactive discussions, etc. to gain a student's attention. However, it becomes a difficult task in an online setting as the educators can only see a limited number of students on their screens a limited size where most students are required to turn their microphones off for a better learning environment.

As it becomes difficult to monitor a student's attention level in an online setting, a different approach is used to monitor the attention level of the students. Attention span correlates with brain activity as the brain is the main organ in control of various cognitive processes that include focusing on a task, processing the information, finding its solutions, and remembering it. Hence, one way of monitoring the attention span would be collecting the brain signals using electroencephalography (EEG), which is a process used to record the electrical activities of the brain in real-time, and predict an attention state by processing the data. As a result, analyzing the

data of the fluctuations of the brain signal could help us understand the change in attention levels of students in a class, revealing whether a student is being attentive or not.

1.2 Problem Statement

The new era of E-learning has brought a drastic change in the education system. Students are now able to learn sitting in any place around the world with a single device in their hands without the need to travel from one place to another. Despite the flexibility, there are certain challenges that are required to be addressed and one of them is the maintenance of attention among the students.

In traditional face-to-face classrooms, the teachers or instructors can easily tell whether a student is being attentive or not by interactive sessions and observation of their body language during a class. However, they cannot use these methods in an online environment. This creates an issue as the teachers are now unable to check if a student is being attentive or not and it makes it easier for the student to get away with being distracted, either voluntarily or involuntarily.

Previously, many studies have been carried out, but unfortunately, no consistent method was found to be completely accurate, so it was not entirely effective in helping teachers monitor their students' attention in online classes. As a result, it has been difficult for them to understand whether their classes are being fruitful or not, and if the students are properly learning or not. Brain signals play a vital role in the analysis of the attention state of students. Since the brain controls and helps in logical processes like attention and decision making, electroencephalography (EEG) can be used to measure brain activity, providing real-time data that can help assess the attention span of students. This technology can be added to online classroom facilities, creating a solution for teachers to monitor attentiveness of their students effectively, which will help create a better learning approach.

1.3 Research Objective

The objective of this research is:

- To effectively detect the attention state of a student during online class for the teachers using EEG.
- To provide real-time feedback to the teachers regarding the attention level of the students.
- To promote proper utilization of class time by keeping the students focused.
- To identify the pattern of attention loss among the students for an improved learning environment.
- To promote adaptive learning by the teachers based on the EEG feedback.

1.4 Research Contributions

To achieve the research goals, a handful of approaches were utilized. A combination of some models was implemented to contribute to fulfilling the main objective of the research which are mentioned below:

- Initially, data preprocessing and feature extraction were implemented on EEG signals. Bi-LSTM and Bi-GRU models were employed to classify the signals, and their accuracy levels were recorded. Then, a fusion of multi-head attention with these models was performed to construct an optimized classification framework and enhance the accuracies.
- Subsequently, data transformation and augmentation were performed as part of preprocessing on the image dataset. The ViT and contrastive learning models were used to classify the images and the accuracies were noted.
- Finally, late multimodal fusion was incorporated into the overall architecture, where the results from both data classifications were combined using standard and weighted ensembles to evaluate the overall performance.

Chapter 2

Literature Review

In this section, we have briefly described the relevant research works done in the area of students' attention span dynamics in online classes using electroencephalography. This is to understand how EEG data is used to classify attention states.

In a research [16], a diverse group of males and females of different ages attended a workshop in the Australasian Association for Engineering Education (AAEE) conference in 2019, where they were given an opportunity to gain insights about advancements of technology, especially wearable technology. They were divided into groups to discuss the challenges that students seem to face in classrooms and come up with solutions. After the discussion, it was found that most people were concerned about the students' engagement in the classroom, followed by their attention. Upon further discussion, the participants agreed to focus upon the attention of students.

Attention of students is necessary for their learning process. While it might be easier for teachers to observe whether students are attentive or not, it requires a big portion of their energy. On the other hand, it is difficult for teachers to even monitor the attention of the students. Despite the advancements in technology, usable methods with technology have yet not been established to monitor attention states. Among these technologies, researchers have found EEG, a method of monitoring electrical signals of the brain, and eye-tracking using smart glasses to be an effective method. I. V and A. Kist [16] imply that the use of EEG using Brain Computer Interface (BCI) can be a non-invasive promising method to monitor students' attention in real time. Considering this information, the researcher aims to use a multi-channel EEG BCI device to monitor attention in classrooms and apply machine learning models on the data to propose efficient applications for future usage.

In a study [6], the problem of inattentiveness was tackled during online learning, inspired by students failing to be honest and attentive in classes during COVID-19. A device using a stacked generalization approach was developed, that would automatically detect a student's attention, and was tested on 20 healthy people using a 21 channel Nihon Kohden EEG electrode. The test was done by making the subjects drive a car using the Logitech Driving simulator, since driving and attending online classes both require a proper amount of attention. The test was also done on two states of the subjects. One was where they were properly attentive and the other one was where the subjects were forced to stay awake through the night and naturally, being tired and sleepy, that state was considered inattentive. A training

set of five-fold cross-validation was used. The approach used to classify the data and to calculate mean accuracy were Decision tree, C4.5, SVM, SVM-RBF, LDA, kNN, Bayesian Network, XGBoost and Random forest approach where results were more than satisfactory by outperforming other state-of-the-art with mean accuracy of 92.7%, 95.234%, 98.36%.

Similar to the study mentioned above, an EEG data processing pipeline and a mental state detection algorithm was designed using SVM method, and used that to compare with KNN and Adaptive Neuro-Fuzzy System methods in another research [4]. To do this, EEG recordings were collected from 5 participants, who were asked to participate in experiments to control a simulated train for 35 to 55 minutes. Through the study, three mental states: Focused, Unfocused and Drowsy, were examined using a modified 12-channel EEG headset. The focused state required the participant to supervise the train while being completely attentive. In the unfocused state, the participants were detached but awake, where they did not specifically pay attention to the simulation. In the drowsy state, participants were allowed to drift off, resulting in a complete lack of attention to the simulation. Spectrograms of EEG signals were calculated using STFT and Blackman Window using feature extraction implemented on the data separately collected from 7 channels. After splitting the data into a 8:2 ratio, the experiment included two evaluation paradigms: subject-specific and common-subjects. The accuracy for the classifier was then calculated that ranged from 88.60% to 96.70% and 71.80 to 78.80% for subject-specific paradigm and common-subjects paradigm respectively. The results were then compared with KNN-based classifier and Adaptive Neuro-Fuzzy System (ANFIS) that yielded accuracy of range 73.35% to 83.46% and 78.33% to 85.41% respectively.

Moreover, there are researches that collected EEG data from students through various experiments to explore the attention span dynamics of students.

A research [18] found, aimed to utilize real-time EEG data to classify a group of student's understanding in online classes. The EEG dataset contained 14 features collected from the scalp electrodes which were taken from Kaggle. Fast Fourier Transform (FFT) was used to decompose data, band-pass filtering was used to remove noise and an inverse FFT was used to transform the data in its original state. For feature extraction, they used Short-time Fourier Transform (STFT) for stationary signals, and for non-stationary signals, Continuous Wavelet Transform (CWT) was used. The models used were: VGG-16, VGG-19, ResNet-50, and ResNet-101 where they found that CWT features were better since it helped them to obtain 85.31% using the ResNet-50 model.

EEG data from students were collected in the research by J. Mehta et al. [15], where they focused on students with irregular patterns in brainwave activity. EEG brainwave signals were categorized into five main frequency ranges- delta waves, theta waves, alpha waves, beta waves, and gamma waves which are directly correlated to different mental and emotional conditions. The proposed system involves utilizing a moving average feature to remove noise, and Bi-directional LSTM as the classification model. The results showed that Bi-LSTM outperformed the rest of the

models such as decision trees, random forests, extra trees, and XGBoost, obtaining an accuracy of 98%.

In a different study [11], brain activity across different EEG frequency bands was explored to find out which frequency waves are most affected by attention. MATLAB was used for computing EEG properties, where brain activity is analyzed in two states- attention and relaxation. Ten people, consisting of equal numbers of men and women, with an average age of 21 years, participated in the experiment. During the relaxation state, the participant closed their eyes; whereas during the attention state, the Stroop Color-Word Test was incorporated. Additionally, band-pass filtering was utilized to filter the EEG data. The EEG signal performance was examined across five frequency bands- alpha, beta, gamma, delta, and theta. The results obtained showed that during the attentional state, alpha, beta, and gamma waves increased, while delta waves decreased. Theta, on the other hand, increases on both left and right powers. It was seen that theta activity increased when performing prayers or meditation, while delta wave activity decreased consistently across all four electrodes. The overall results indicated lower concentration levels.

In another small-scaled research [1], an experiment was conducted on 6 males and 4 females during the lecture. Noise was removed by using band filters and 13 features of EEG signals were considered. EEG values were arranged and filed with every participant's self-claimed attention state after feature value extraction using two algorithms: FastICA and ApEn. K-nearest neighbor classifier (KNN) and the naive Bayes classifier were used as classifiers. The set of four features (alpha_FO, theta_H, alpha_power_mean, alpha_Pmax) came out as the most effective in attention recognition for both 3 classes and 5 classes. It was found that the best classification rate for 3 classes is 63.9% and for 5 classes it is 51.9%. In terms of classifier, the best classifier is KNN with K=5 for both 5 classes and 3 classes.

M. Alirezaei and S. Hajipour Sardouie [2] focused on measuring student's attention (attentive and inattentive) using frequency bands by taking 12 participants for their test. Signals were extracted by using an EMOTIV device of 8 channels to group frequency bands into delta, theta, alpha and beta. Preprocessing and features extractions included bandpass filter, Shannon entropy, Fourier-based power analysis, energy computation, peak-to-peak values, mean values, and alpha-to-beta ratios. In total, 168 features were extracted for each subject and the Forward Feature Selection method was used with Fisher criterion to find out the effective feature groups. For this study [2], KNN, Bayesian, Support Vector Machine, and Linear Discriminant Analysis was used with k-fold cross validation for accuracy. Features of beta bands seemed to be most effective from the results and among the four classifiers, c-SVM and LDA gave the most accurate results with an accuracy rate of 92.8% and 92.4% respectively.

Another research [9] proposed an attention classification model called FF-BiALSTM using deep learning methods. The proposed model combines an Attention Mechanism, that captures global features, with Bi-LSTM, that captures time-domain features. The experiment was carried out on the Student EEG and Student Reading datasets, where the data was preprocessed and normalized, before being split

into training and testing sets into an 8:2 ratio. FF-BiALSTM obtained 91.35% accuracy on the Student EEG Dataset, and 89.50% accuracy on the Student Reading Dataset, outperforming all other previous models such as: CNN, BiLSTM, Two BiLSTM, and CNN-BiLSTM.

S. Gopi and N. Dehbozorgi [13] developed an ML pipeline that was used on ADHD and inattention data to detect state of inattention among the students. A public 19-channel EEG dataset of 60 ADHD and 61 non-ADHD participants was used for this purpose and various classification models like KNN, SVM, NN's were used on it. The researchers also developed a Multi-layer Perceptron (MLP) model to train the dataset and achieved an accuracy of 78% initially after preprocessing, filtering, dimensionality reduction, and power spectral feature extraction that improved to 88% using the Leave-one-subject-out (LOSO) method. Upon use of more different classifiers like Decision Tree, Random Forest, XGBoost, and AdaBoost, a stable accuracy of 98% was found which was previously 72% without the LOSO method. The model was then expanded to attention deficiency data. The data usage was expanded using datasets from IEEE dataport and Kaggle. Additionally, S. Gopi and N. Dehbozorgi [13] recruited 15 college students over 18 for another experiment using two visual reasoning tests that included TestMyBrain (TMB) Matrix Reasoning and Flanker Attention tests for assessing their attention levels. Ensemble models like Random Forest, XGBoost, and AdaBoost performed better than neural network models on larger datasets by achieving nearly 95% accuracy and prediction probabilities like Decision Tree, LightGBM, Logistic Regression etc resulted in an increase of accuracy from 88% to 98%.

AttentioNet (a CNN based model) was introduced in a research [17] to classify different complex nature of attention (Selective, Divided, Sustained and Alternating attention) of students using deep neural networks to analyze Electroencephalography, and facial data from 20 students. The experimental tasks included Stroop color test, Reading test, Continuous Performance Test (CPT) and TMT-B test. The EEG data was recorded with a 14 electrode EEG headset and to cancel noise it was bandpass filtered in a range of 0.2Hz to 45Hz. The model categorized attention states using an EDA-based four-block pipeline: Low-Level Feature Extraction, Signal Attention Mechanism, High-Level Feature Extraction and Classification. The model was trained by using 19 people for training and 1 person to test, and this is repeated for every person. Accuracy for AttentioNet was calculated to be 53.75% and compared to EEGNet's accuracy, 43.8%. The model was also personalized by fine-tuning based on every subject to get an even higher average accuracy of 92.3%.

Despite EEG being one of the most reliable ways to detect attention span, combining this method with facial expressions or body movements makes the methods more robust and effective in classifying attention states. The papers discussed below effectively use these methods to monitor the attention states of students.

S. Kim et al. [19] developed a new EEG index to monitor the attention states of students during online learning. The experiment was done on thirty 4th grade elementary school students to get the EEG data consisting of 19 features during

a computerized D2 test where the participants had to quickly distinguish a target symbol from other non-target symbols. For preprocessing the EEG signals, bandpass filtering, notch filtering and independent component analysis (ICA) were used. Then the Pearson correlation coefficients (CC) were calculated for the D2 test to determine the feature that can most accurately predict the attention state of the student. For the main experiment conducted by S. Kim et al. [19], the students watched a 40 minute lecture video on heat waves and volcanoes and were asked to take a test and their upper body video was recorded the entire time. The upper body movements were categorized into nine categories: face, eyes, nose, cheek, mouth, chin, head, torso, and arms and the movements were identified manually and the EEG Index of Attention (EIA) was calculated. As the main goal of this study [19] was to find the characteristics behaviors of students having a low attention state, the concept of attention scores (AS) was developed. The highest attention score was 0.119 for the students opening their mouth, and it was concluded that this was valid as opening the mouth could mean that the participant was yawing which means that he was bored (AS=0.117). Additionally, movement of closed mouth like lip pouting, sticking out of lips could mean a state of confusion in the student and it also showed a high AS value. More movements like movement of head at pitch angle represented the mode of distraction(AS=0.035), leaning back on chair (AS=0.051) meant low state of attention, and rolling movement of head (AS=0.006) could mean state of confusion about any answer.

U. Burnik et al. [3], were aiming to study the pattern attention span by observing common patterns for losing attention using the high resolution Microsoft Kinect 2 camera to record motion of three students at a time. This led them to find out that very few students were able to continuously focus for more than 10-18 minutes and student engagement with note taking was directly correlated to the attention level. The experiment was done by resourcing a field of linear time invariant (LTI) discrete systems response which covers frequency response by analyzing Z-transform. Results were compared for each test, and students scored 75% after participating in the lecture, whereas, before participation, their score was 42%.

To identify effective methods and models for attention detection using facial or body images, we reviewed several studies that explored both supervised and unsupervised approaches to effectively identify attention states.

A new dataset is developed in a research [10] for Vietnamese people named Vietnamese Face Dataset (VFD), which contains 3000 images of 1000 famous Vietnamese people. The study implemented ViT and CNN models as backbones to learn facial features, testing them on multiple datasets (MS1M-RetinaFace, LFW, CALFW, CPLFW, CFPW, AgeDB, VFD) with varying parameters. The images were divided into patches, embedded with positional encodings, and processed through a transformer encoder. REG1K and REG2K are two configurations that were tested by splitting the dataset into two subsets. Results showed that ResNet-101 outperformed ViT on the mid-sized MS1M dataset, but ViT achieved higher accuracy of 89.75% on the VER4K dataset with limited data.

However, the amount of unlabeled data is increasing each day and to make full use

of these data, the researchers are willing to find ways to get high performing models using images because the previous models take large training time and occupy huge memory space due to lack of contrastive pairs and unrepresentative vectors. So a method with location based sampling strategy integrated into the MoCo-3 framework and a memory bank combined into a novel framework with dimension reduction module is proposed in a study [14]. This makes it possible to capture the most comprehensive features and make large numbers of samples available for contrastive learning. The selected model is applied on some representative data sets including CIFAR-100 AND imageNet-100 since these datasets are the baseline in the field of development of self-supervised learning. The proposed method with the chosen MoCo-v1 baseline with integrated dimension reduction and stuck inputs gives 99.4% accuracy for CIFAR10 and 99.5% accuracy for CIFAR100 compared with other methods because proposed method has a better data utilization and increases the number of contrastive pairs due to location based sample strategy.

A semi-supervised learning (SSL) approach was proposed in a study [12] to train CNN classifiers on large-scale facial expression recognition (FER) datasets that contain both labeled and unlabeled data. The method uses contrastive self-supervised learning (SimCLR) to pre-train a network on unlabeled data. Augmentations such as random cropping and Gaussian blur were applied during training and meaningful representations were extracted via a contrastive loss function based on Noise-Contrastive Estimation (InfoNCE). A novel contrastive clustering method was employed to solve distribution mismatch issues between labeled and unlabeled data. A K-means clustering was used to group similar samples together and the Silhouette Coefficient (SC) determined the degree of separation between clusters. The proposed method was tested on four datasets: RAF-DB, FERPlus, AffectNet, MS-Celeb-1M-v1c, which showed improved performance compared to the FixMatch algorithm, especially on RAF-DB and FERPlus datasets. ResNet-18 outperformed WideResNet-28-2 with smaller labeled datasets, and the proposed algorithm achieved an accuracy of 77.41%.

A similar study [21] on FER proposes a method involving the implementation of contrastive learning to improve learning intensities between shallow and deep layer features in convolutional neural networks (CNNs). Facial expression recognition (FER) tasks are often difficult to carry out because of inconsistencies in learning intensities as shallow layers are often subject to weak representations due to the gradient vanishing problem during training. To tackle this, a cross-layer contrastive learning framework was used that aligns latent semantics among the deep and shallow layers. In the proposed framework, the images are initially passed through a feature encoder, and then passed to a Feature Map Contrastive Learning (CLFM) that ensures semantic alignment among deep and shallow layers and a Batch Sample Contrastive Learning (CLBS) that focuses on feature alignment across batch samples for dynamic learning. Additionally, there is a Multi-Scale Representation (MSR) module that combines features from different layers using an attention mechanism, in order to adaptively weight their contributions in improving feature representation. Finally, a joint loss function that combines batch sample contrastive loss, feature map contrastive loss, and classification loss is used. The proposed method achieved state-of-the-art accuracy on the datasets with 92.21% on RAF-DB, 89.50%

on FERPlus, 62.82% on SFEW, and 65.29% on AffectNet datasets, with minimal computational overhead.

As most studies focused only on using a single modality for monitoring attention span of individuals, it motivated us to opt for a more robust approach that includes facial image data for better classification of attention states of students during online classes.

Chapter 3

Background Study

3.1 Description of Models

For this research, we utilized multiple models in order to make a predictive algorithm that is more optimized. In this approach, we selected various ensemble learning models, neural network models, and unsupervised learning to efficiently classify the attention states.

1. AdaBoost

AdaBoost (Adaptive Boosting) is a common boosting algorithm that combines multiple weak classifiers to build a strong predictive classifier model [5]. It uses a Decision Tree as its default classifier, and each weak classifier learns from the previous one's incorrectly classified data. Initially, it assigns equal weights to each of its data samples and trains the weak model on each of these weighted samples. After learning, the algorithm tries to predict the output for that sample, where the predicted and actual outputs are compared to see if the data were classified correctly or not, and these steps are carried out for multiple iterations. After each prediction, more weights are added to the wrongly predicted samples, so that they are corrected in the next iteration. At last, the weighted votes from all the weak models are combined to make a final prediction.

2. XGBoost

XGBoost (Extreme Gradient Boosting) is a boosting algorithm that is popularly used for its ability to work with large datasets and handle missing values efficiently [5]. Compared to AdaBoost, where the weights are slightly changed for training the weak models, XGBoost trains its models based on the residual errors of their previous models. At first, it assigns weights to the individual variables, trains a base model on those variables, and the model predicts the output for each of these variables. Then, the differences between the predicted and actual values are calculated, which are the residual errors. The weights of these misclassified, residual variables are increased, as now, these will be used as labels to train a new weak model. Each new model makes a new prediction and the predictions are updated gradually. The whole process is repeated multiple times, where the algorithm tries to reduce a specific loss function, which is a log-loss for classification, with the aid of gradient descent to

improve the predictions during each iteration. Ultimately, outputs from all models in the sequence are summated to make a final prediction.

3. Bi-LSTM

A bidirectional long short-term memory (Bi-LSTM) network was used, which is a type of recurrent neural network (RNN) that processes sequential input data in both forward and backward directions [20]. It combines LSTM's memory and gating mechanisms (forget, input, and output gates) with bidirectional processing to allow the model to keep track of both upcoming and past data of the input sequence. It comprises two LSTM layers, one working in the forward direction and another in the backward direction, each having its own set of hidden layers and memory cells. The forward LSTM layer allows forward pass, where the input sequence is fed from the first time step to the last time step. Similarly, the backward LSTM layer allows backward pass, where the input sequence is fed in reverse order, from last step to the first. Finally, after the forward and backward passes are completed, the hidden states from both layers are combined at each time step.

4. Bi-GRU

A gated recurrent unit (GRU) is an improved version of RNN that can handle sequential data. There exists a vanishing gradient problem in RNN, which is solved in GRU due to the incorporation of the gating mechanism, comprising of reset and update gates [20]. These gates make it possible for GRU to retain the necessary information for a long time by discarding unwanted or unnecessary information. The reset gate is responsible for controlling the amount of previous information to be forgotten or reset. When the reset gate is close to zero, it retains only the current inputs by ignoring the previous states completely. On the other hand, the update gate is responsible for passing on the necessary portion of the new input to update the hidden state. Bi-GRU is a more efficient process since it extends GRU by the addition of a second GRU layer which processes the input in reverse order. The mechanism is such that the forward GRU is responsible for processing the input sequence from start to finish in the forward direction while the other one is tasked with processing the sequence in the backward direction. Therefore, the model learns from both past and future data, making it ideal for predicting outputs correctly. The outputs from both the GRUs are concatenated to take in the information from both the inputs. Bi-GRU is advantageous since it has fewer gates than LSTM making it more computationally efficient.

5. Multi-head attention mechanism

The multi-head attention mechanism is a key constituent of Transformer architectures. It is designed in such a way, with multiple attention layers placed in parallel, that boosts the model's ability to learn different types of interactions between different tokens in the input sequence [20]. Here, attention is computed multiple times, which are called heads, on different parts of the input sequence, and the outputs from each computation are combined to get richer information about the input. Each head maps out the input into key, query, and value representations. It also acquires context-specific dependencies and evaluates scaled dot-product attention. As the

multi-head attention mechanism works on different parts of the sequence simultaneously, it facilitates the model to show the information more efficiently in multiple contexts. After combining all the outputs from all heads, they are concatenated and passed through a linear layer, which allows a better and more comprehensive understanding of sequential data.

6. ViT

Vision Transformer (ViT) is an adaptation of the transformer model that breaks down input images and processes them as patches, instead of depending on convolutional layers [7]. Initially, it divides an input image into equal, fixed-sized patches and flattens each patch into a one-dimensional vector to form patch embeddings. Then, using a linear transformation, each patch embedding is mapped onto a higher-dimensional space, where the positional information is added to each patch embedding in order to retain spatial context. Moreover, a transformer encoder is used that integrates layer normalization and residual connections for stable training. It is made up of multiple layers of multi-head self-attention, that identify relationships between all patches, and Feed Forward Neural Networks (FFNN), that process the outputs from each attention layer. Furthermore, a Classification Token ([CLS]) is added to the input image sequence, which combines information from all patches for classification tasks. Finally, a fully connected layer is employed to project the [CLS] token outputs to the intended predictions and classes. The ViT model outperforms convolutional neural networks in many vision tasks as it works well on large-scale data, especially when pre-trained on large image datasets.

7. Contrastive Learning

Contrastive learning is a self-supervised learning method to learn distinguishable features in similar and dissimilar parts of a set of data points [22]. In order to improve classification performance, Contrastive Learning pushes the model to learn representations that separate various attention states while grouping comparable states together in the feature space. It works by grouping positive pairs (similar data) and negative pairs (different data). So, images containing the same type of facial expressions are grouped together, while the attentive states images are grouped together, but the attentive and sleepy state images are not paired together and are classified as the negative pairs. By learning robust features, contrastive learning to distinguish and identify different images with the given data improves generalization.

Chapter 4

Methodology

For the purpose of monitoring the attention span of the students in online classes, we first need to collect relevant EEG datasets. We will either use previous datasets from research and studies that were already conducted or we may collect new EEG data from students attending online lectures. For our research, we collected a dataset from a previous research study and we focused on processing and analyzing the EEG data to predict attention states based on the raw EEG signal data. Firstly, we filtered out the necessary files from the entire dataset accordingly and then filtered out relevant EEG channels that would aid us in classifying the attention states. Then we resorted to three different approaches to figure out what works best for classification of EEG signals for attention state detection.

Approach 1: In this approach, no feature extraction was performed and only the relevant EEG channels were used as input after organizing them and labeling the attention states into two different classification models- XGBoost and AdaBoost.

Approach 2: The data were preprocessed using the Butterworth bandpass filter to remove any unnecessary noise in the EEG signals. Then a short-time Fourier transform (STFT) was used to extract relevant features like alpha, beta, gamma, delta and theta for better classification. The data were then organized and attention states were labeled for these to be used as input into two recurrent neural network models- Bi-LSTM and Bi-GRU.

Approach 3: In this approach, we followed the same data pre-processing and feature extraction method as approach 2. The key difference was the use of models. We attempted to use two different fusion models for better classification of attention states. The first one was a multi-head attention mechanism with Bi-LSTM and the second one was a multi-head attention mechanism with Bi-GRU. The data were split for training and testing and classification reports for three attention states: Attentive, Inattentive, and Sleepy, were generated.

We also implemented a multimodal fusion by incorporating an image dataset of facial features for better classification of attention states. We collected a labeled image dataset consisting of three attention states: Attentive, Inattentive and Sleepy, that was divided into train, test and validation sets. We first transformed the training data using data augmentation and followed two approaches: Supervised learning

using ViT and Unsupervised approach using contrastive learning. Both the models were trained with the use of loss functions. The supervised learning efficiently used the labeled data but on the other hand, contrastive learning did not rely on the labels of the data. ViT directly classified the images after training while contrastive learning used the encoded embeddings as an input into a simple classifier and tested the unseen data. The performances were then evaluated.

Then we used a late Multimodal fusion using a standard and a weighted ensemble on the results for classification of the attention states. The overall research methodology is summarized in Figure 4.1.

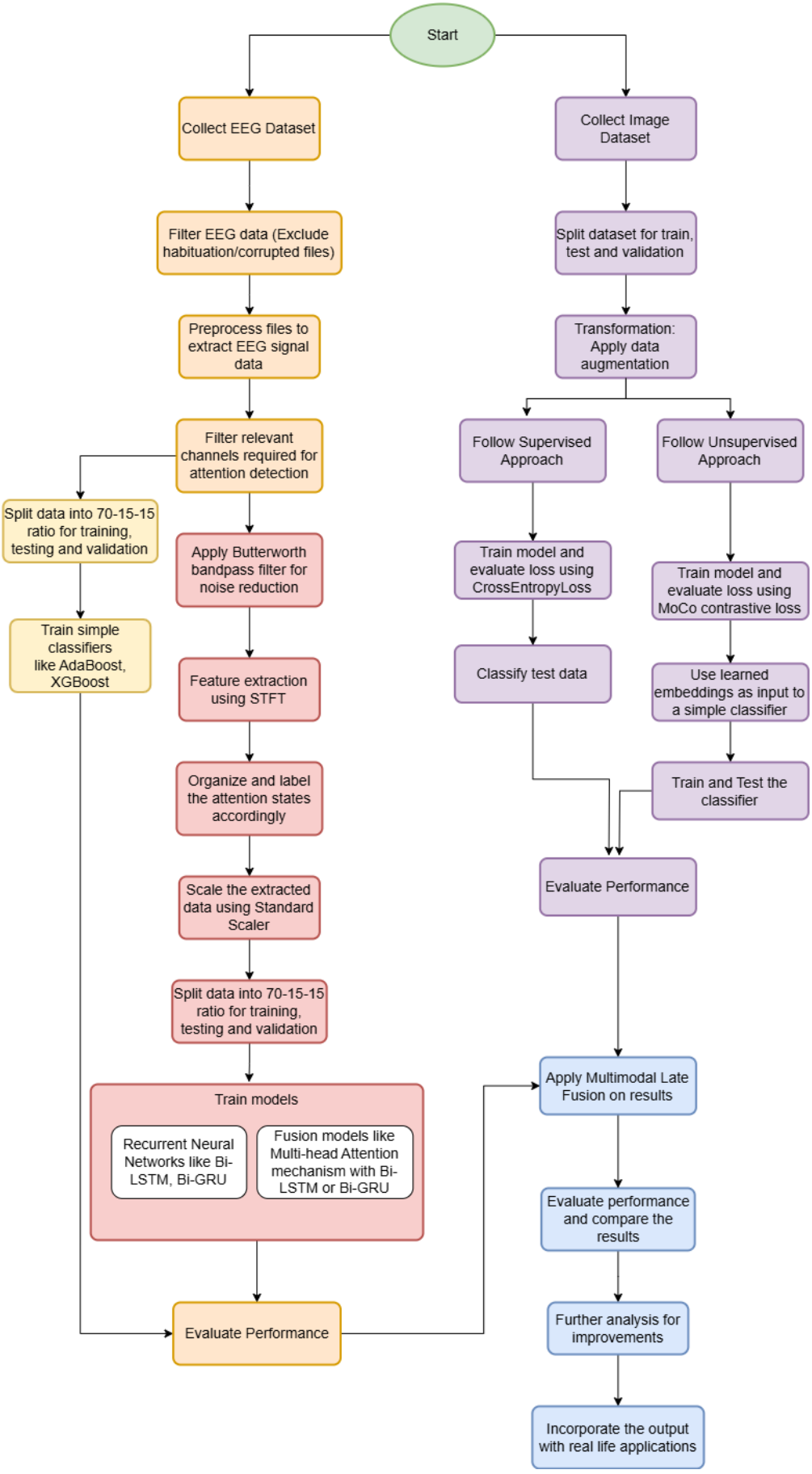


Figure 4.1: Top-level overview of the method

4.1 Dataset Description

We used two different datasets for the purpose of this research. The first dataset that we used was an EEG dataset.

An experiment was carried out by Ç.İ. Acı et al. [4] to create a dataset to study the changing mental state of humans while carrying out a task. 5 participants took part in this experiment where a total of 25 hour EEG recordings were collected. The participants had to take part in controlling a computer-simulated train using the "*Microsoft Train Simulator*" program and they had to control the train in the program for 35 to 55 minutes in total. The intention was to create a dataset consisting of three mental states: Focused, Unfocused, and Drowsy. Each person had to participate in at most one experiment every day for 7 days, from which the data from the first two days were considered to be of less importance as the participants were getting used to the experimental environment. Hence, experimental data of five days from each participant were collected in the form of mat files. However, the final participant took part in 6 experiments in total, resulting in a total of 34 mat files in the entire dataset.

For this experiment, a modified version of the Epoc EEG headset was used to collect the EEG data. The sampling frequency was 128 Hz and each mat file had 25 columns namely: 'ED_COUNTER', 'ED_INTERPOLATED', 'ED_RAW_CQ', 'ED_AF3', 'ED_F7', 'ED_F3', 'ED_FC5', 'ED_T7', 'ED_P7', 'ED_O1', 'ED_O2', 'ED_P8', 'ED_T8', 'ED_FC6', 'ED_F4', 'ED_F8', 'ED_AF4', 'ED_GYROX', 'ED_GYROY', 'ED_TIMESTAMP', 'ED_ES_TIMESTAMP', 'ED_FUNC_ID', 'ED_FUNC_VALUE', 'ED_MARKER', 'ED_SYNC_SIGNAL'. From these columns, the raw EEG data, required for mental state detection, were in the columns starting from 4 to 17. The states "focused", "unfocused", and "drowsy" were grouped by 10 minutes each; a single mat file had to be grouped with a 10-minute mark which was calculated by: 128*60*10. Therefore, the range of the focused state was from 0 to 128*60*10-th row, the data for the unfocused state were from 128*60*10-th row to 128*60*10*2-th row, and lastly, the data for the drowsy state were from the 128*60*10*2-th row until the end. 'ED_COUNTER' is responsible for counting each sample, and 'ED_INTERPOLATED' is a flag determined to indicate if the data point was interpolated. 'ED_RAW_CQ' is responsible for capturing the quality of EEG raw signals. 'ED_GYROX' and 'ED_GYROY' are gyroscope data for the x-axis and y-axis respectively. Two other timestamps were recorded to mark the data collection and timestamp of an event. 'ED_FUNC_ID' was used for identifying functions whereas 'ED_FUNC_VALUE' was used to see the value associated with a specific function. 'ED_MARKER' was used to annotate important points of interest in the dataset. Finally, 'ED_SYNC_SIGNAL' is a synchronization signal, as its name indicates, and is used to align data.

The image dataset that we used was generously provided by one of our distinguished alumni, MD. Asif, who collected and curated the images as part of their previous research. The dataset contained images divided and labeled into three classes: Attentive, Inattentive and Sleepy. The images were in JPG format with a dimension of 100x100 pixels. This dataset was specifically designed to train and test image clas-

sification models and detect the attention states of students during online classes.

The dataset sizes are summarized in Table 4.1 below.

Dataset	Number of Data
EEG Dataset	34 mat files consisting of around 9002.1k EEG signal data
Image Dataset	10798 labeled images

Table 4.1: Dataset Summary

4.2 Data Preprocessing

The data were preprocessed differently for the two datasets.

Firstly, for the EEG dataset [8], we started by separating the mat files that contained relevant EEG data. Each participant took part in the experiment for 7 days, hence we had 7 mat files for each participant. However, as the participants were getting habituated to the experiments, the first two files of each participant had to be omitted. The last participant took part in the experiment for 6 days so only 4 mat files for this participant were used. We then extracted and filtered seven useful columns that were raw EEG signals, required for attention state classification. The channels that were used were 'F7', 'F3', 'P7', 'O1', 'O2', 'P8' and 'AF4'.

As the sampling frequency used was 128Hz, we divided the data of each mat file into three parts using a marker of 128*60*10 as the participants were asked to be in a specific mental state for 10 minutes each except the drowsy part, acting accordingly in the simulation. For each participant's experimental data, attention states were labeled as 0, 1, and 2 for three mental states respectively: focused, unfocused, and drowsy.

After the channels were filtered, we used Butterworth bandpass filtering on the signals.

4.2.1 Butterworth Bandpass Filtering

Bandpass filtering is a filter designed to remove unwanted noise or frequencies. This allows only a specific range of frequencies to be extracted for use, which in our case was between 0.5 Hz and 40 Hz. We first calculated the Nyquist Frequency and normalized the cutoff frequencies by dividing them by the Nyquist value. The signals were then passed through a Butterworth filter in a zero-phase manner, filtering the signal forward and backward, ensuring no phase distortion. Figures 4.2, 4.3, and 4.4 below show the first 5 seconds of channel F7 in three different attention states: Focused, Unfocused, and Drowsy, respectively, showing the comparisons of the signals before and after bandpass filtering.

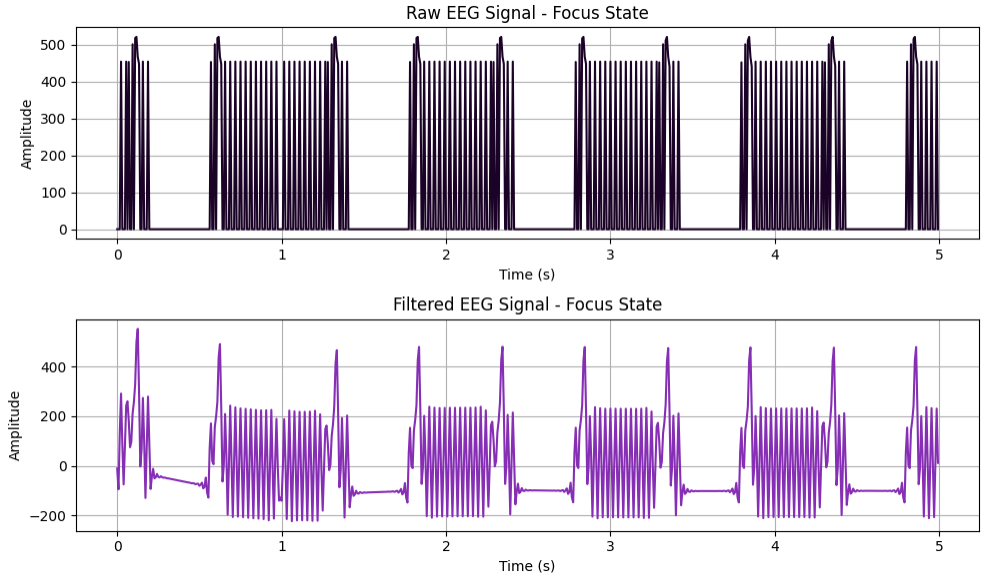


Figure 4.2: Focused State Signal Before and After Filtering

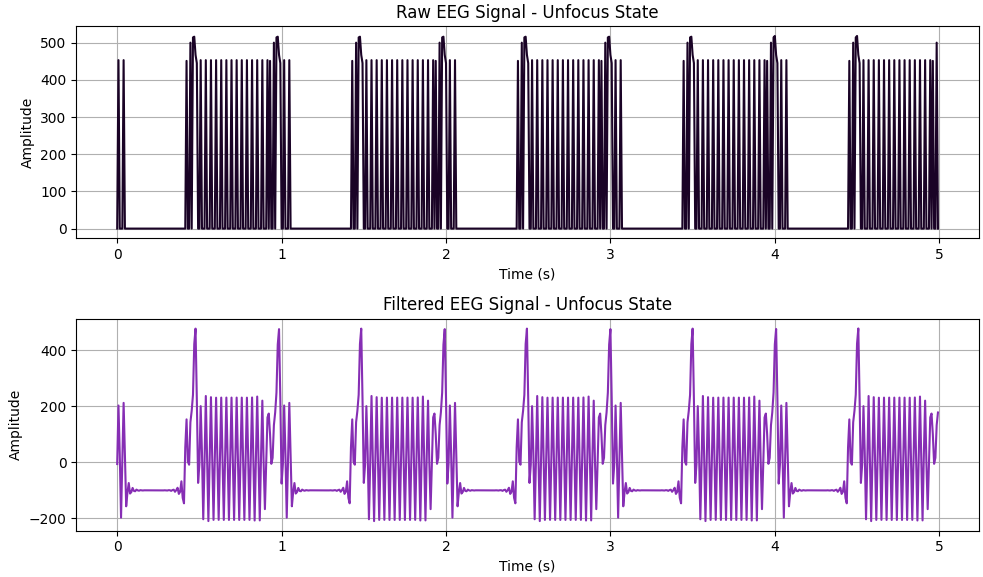


Figure 4.3: Unfocused State Signal Before and After Filtering

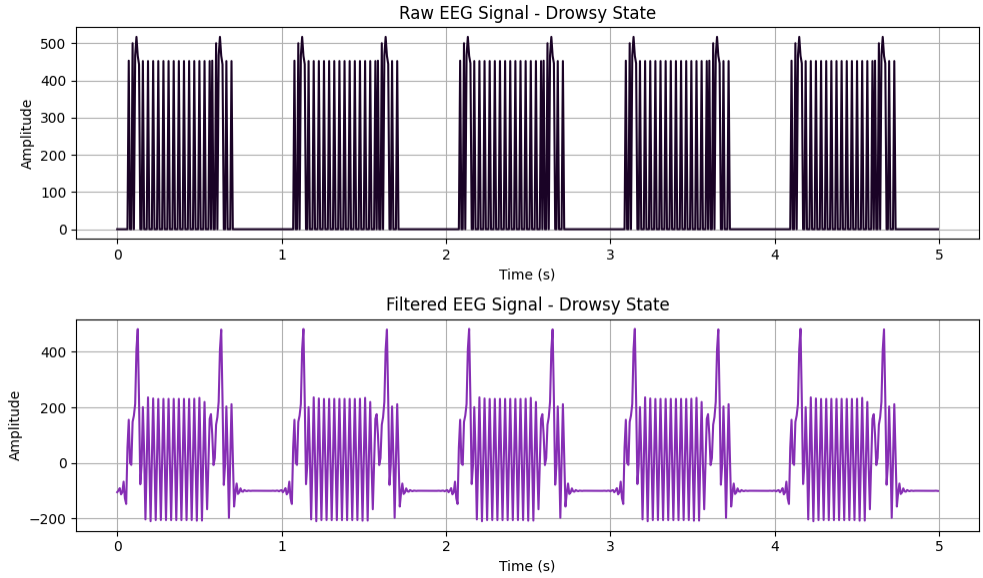


Figure 4.4: Drowsy State Signal Before and After Filtering

We also later standardized the data using StandardScaler.

4.2.2 StandardScaler

Standardization ensures that all features are on a similar scale so that any feature on a larger scale does not end up dominating during the learning process. Otherwise, this would cause the features on a lower scale to be neglected during the training period. The data except the output column was first flattened and then the standardization was achieved using the formula:

$$X' = \frac{X - \mu}{\sigma} \tag{4.1}$$

where μ is the mean and σ is the standard deviation of the feature.
After standardization, the data was reshaped to its original size once again.

On the other hand, we applied data transformation and augmentation to the image dataset as a part of preprocessing.

4.2.3 Data transformation and augmentation

Data transformation and augmentation of images involve modifying the original images to make the input more standard and expand the diversity of the same data. We applied transformations and augmentation on training to improve the generalization of the model. Only data augmentation was applied on testing and validation datasets to ensure consistent evaluation of the data.

The training transform was done in four steps:

- **Resize:** The original images were resized to 224x224 pixels without cropping to ensure consistent size of all the images during training.

- **Random Resized Crop:** This was done for the purpose of slightly zooming the image while ensuring that important facial features of the images were not omitted in any way. The images were randomly cropped between 80% to 100% for creating diversities in the training dataset. The scaling here ensured that the facial features were retained as it was important for attention detection.
- **Tensor conversion:** This was done to convert the image from a PIL format to a PyTorch tensor and scale pixel values to the range [0, 1].
- **Normalization:** The images were normalized by each RGB channel, scaling pixel values to the range [-1, 1]. This improves convergence during training and ensures numerical stability.

The testing and validation transform was kept simpler with three steps:

- Resize
- Tensor conversion
- Normalization

These three steps were performed similar to the train transform to prepare the data to be used for evaluation purposes later. Figure 4.5 shows an example of the image transformation performed to create variations in the training set.

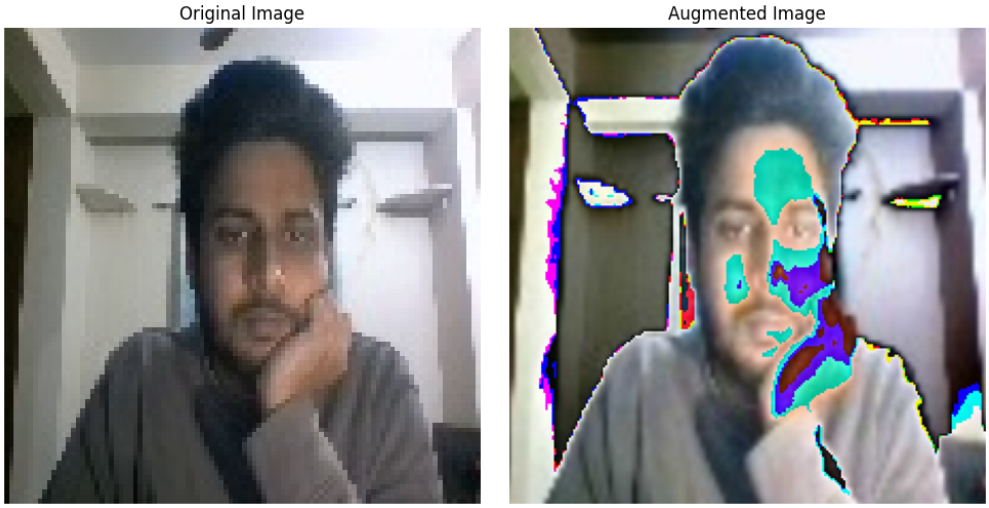


Figure 4.5: Original Vs Augmented Image

4.3 Feature Extraction

Our research involved short-time Fourier transform (STFT) as a part of feature extraction for the non-stationary signals. STFT is used to extract spectral features from multichannel time-series data. We specifically computed power spectral density (PSD) of different frequency bands: Delta (δ), Theta (θ), Alpha (α), Beta (β), and Gamma (γ) for each channel in the input data to obtain meaningful representation of the raw EEG data to help classify three attention states: Focused, Unfocused, and Drowsy.

For each channel in the data, the power spectrum was computed and was averaged across predefined frequency ranges to extract the following bands [2]:

- Delta (0-4 Hz)
- Theta (4-8 Hz)
- Alpha (8-13 Hz)
- Beta (13-30 Hz)
- Gamma (30-50 Hz)

Alpha waves occur when the eyes are closed, whereas beta waves can be seen when they are open during hearing or thinking activities. Gamma waves typically occur in scenarios involving heightened attentional states required for complex information processing. Delta wave activity occurs during deep relaxation states, while theta waves are linked with memories and emotions [11]. The frequency bands play a vital role in the classification of attention states.

The following figures below (Figures 4.6, 4.7, and 4.8) show the differences in different bands in three attention states focused, unfocused and drowsy, respectively for 10 seconds each in the F7 channel.

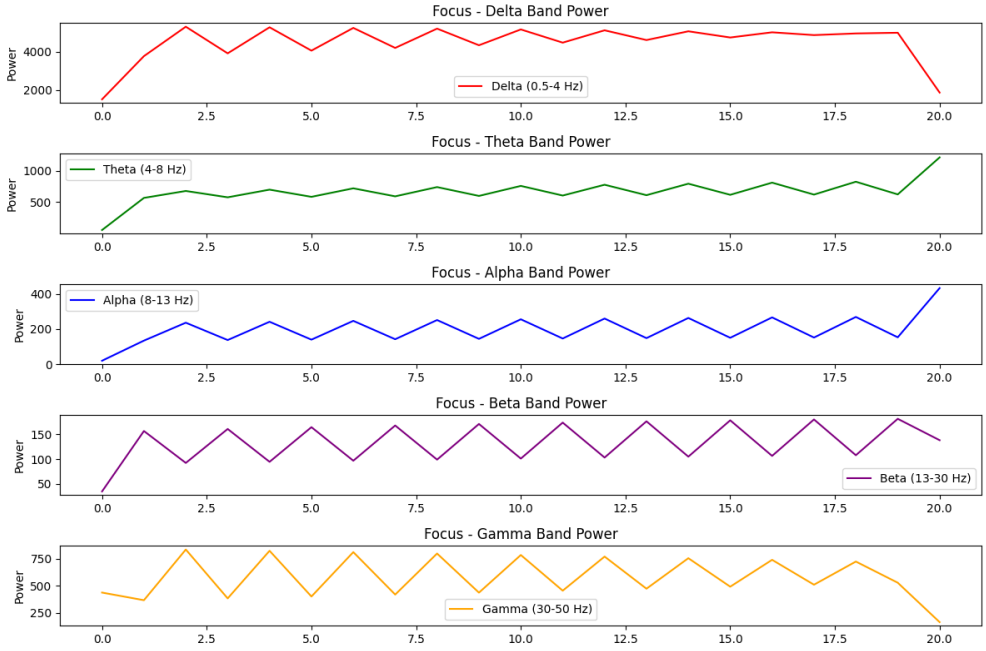


Figure 4.6: Frequency Bands of Focused State

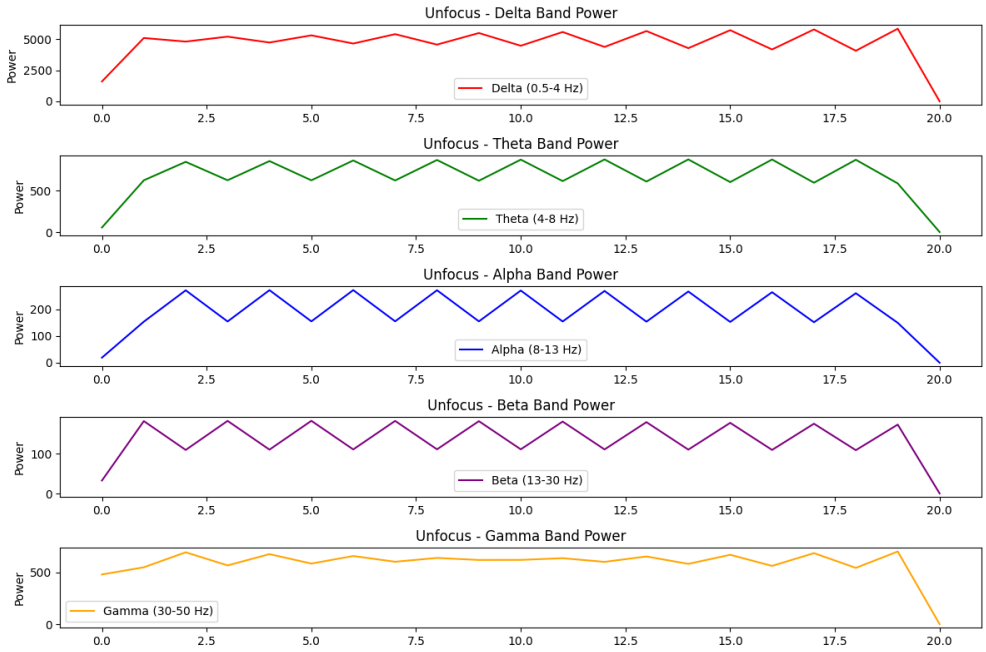


Figure 4.7: Frequency Bands of Unfocused State

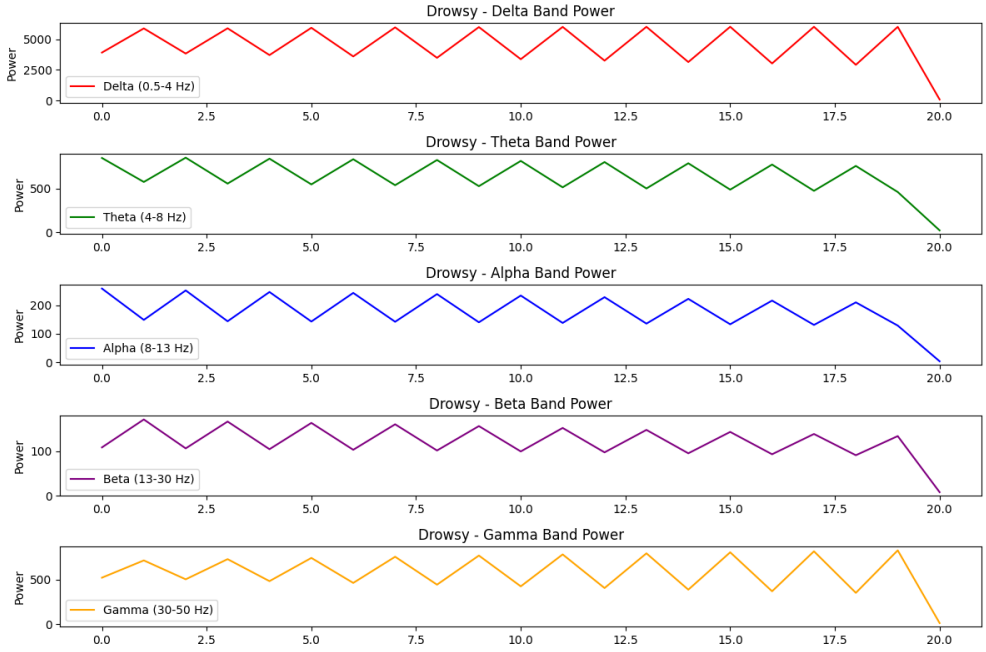


Figure 4.8: Frequency Bands of Drowsy State

4.4 Train-Test Split

Our research utilised the datasets used by splitting them in different ratios for different models. For approach 1 using the models XGBoost and AdaBoost on the EEG dataset, we split the dataset into a 7:3 ratio for training and testing. However, when we moved to approach 2 and 3, we split the dataset into 70:15:15 ratio for train, test, and validation, respectively. This left us with 105602, 22629

and 22929 EEG signal data for train, test and validation purposes. The validation set introduced was essential for the models to be prevented from overfitting. On the other hand, the image dataset was already split into 8199 images for training, 1138 images for testing and 1461 images for validation which is approximately a ratio of 8:1:1.

Chapter 5

Implementation and Result Analysis

5.1 PC configuration for training models

The models were trained and tested on Visual Studio Code and Google Colab using the PC configuration shown in Table 5.1 below.

CPU	Intel Core i7-14700k 3.4 GHz
GPU	Nvidia GeForce RTX 4080 super
RAM	64.0 GB
OS	Windows 11

Table 5.1: PC Configuration

5.2 Model Implementation

Initially, all the necessary libraries needed were imported and the datasets were loaded accordingly.

The EEG dataset [8] contained 34 mat files consisting of around 9002.1k EEG signal data. As the first two files were the experimental data obtained while the participants were getting habituated to the experimental environment, we omitted those files, leaving us with 24 mat files consisting of around 1152k EEG data after the initial pre-processing. No more data had to be excluded as no rows had corrupted data or null values. To analyze the dataset properly and determine the mental state of the participants from the experimental procedure, we had to use the columns 4 to 17 of the dataset [8] of each file that consisted of the raw EEG data. The columns were: 'ED_AF3', 'ED_F7', 'ED_F3', 'ED_FC5', 'ED_T7', 'ED_P7', 'ED_O1', 'ED_O2', 'ED_P8', 'ED_T8', 'ED_FC6', 'ED_F4', 'ED_F8', 'ED_AF4' , respectively. According to Ç.İ. Acı et al. [8], some of the channels were used for current supply and references, so data were not collected from them. As a result, the useful channels used for the actual data collection and analysis were: 'F7', 'F3', 'P7', 'O1',

'O2', 'P8', 'AF4'. The useful channels were reflected in the correlation we found with respect to the attention states that we labeled, shown in Figure 5.1.

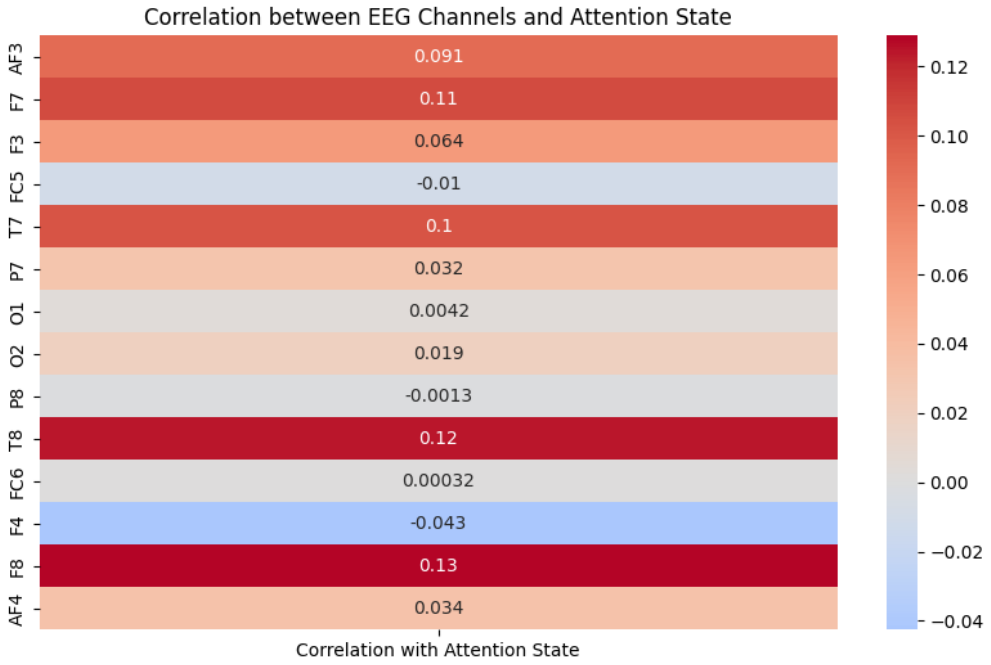


Figure 5.1: Correlation between EEG Channels and Attention State

Each file was divided into three parts according to the marker to prepare the dataset for evaluation. This was the general pre-processing applied to the EEG dataset for every approach covered below.

5.2.1 Approach 1

AdaBoost and XGBoost: For this analysis, we took a subject-specific approach to predict the mental state of one participant at a time. After the dataset was organized for each participant, 70% of the data were used for training and the remaining 30% of the data were tested using AdaBoost and XGBoost models separately with 100 estimators to predict the mental states. However, this approach was not combined with the result of the image dataset for multimodal fusion.

5.2.2 Approach 2

For this approach, after the general preprocessing of the EEG data, we additionally used Butterworth Bandpass Filtering to filter out the noise present in the data. We ensured that only the frequencies ranging from 0.5 Hz and 40 Hz would be kept for our usage. Additionally, we used short-time Fourier transform (STFT) to extract 5 frequency bands Delta (0-4 Hz), Theta (4-8 Hz), Alpha (8-13 Hz), Beta (13-30 Hz), Gamma (30-50 Hz) for better classification of attention states. The extracted signals were then reshaped using a standard scaler to ensure that no extracted feature dominates the training process if it is on a larger scale. The preprocessed data was then split into 70:15:15 ratio for train, test and validation, and the train set was used as input for two different models:

- **Bi-LSTM:** Once the input data was prepared, a Bidirectional LSTM layer with 128 units was used to capture the input that captured both the forward and backward dependencies as the EEG data is sequential. This layer returns the entire sequence rather than just the final state to ensure data is not forgotten soon. The output then is again used as an input to a dense layer consisting of 128 neurons. This dense layer was activated with a tanh function to introduce non-linearity and allow the output to be better. For this model, a dropout layer of 0.4 was added to prevent overfitting by randomly deactivating 40% of the existing neurons. The output from the dense layer was then flattened and a softmax function was used for the output to produce a probability distribution class with the number of neurons in the output classes being equal to the number of categories, which is 3 in our case.

- **Bi-GRU:** This model was utilized similar to Bi-LSTM. A Bidirectional GRU layer with 128 units was used to capture the input data to capture both the forward and backward dependencies. The output then is again used as an input to a dense layer having 128 neurons. This dense layer was activated with the tanh function similarly. A dropout layer of 0.4 was added here as well and the output from the dense layer was then flattened and a softmax function was used for the output.

Both the models were compiled with an Adam Optimizer having a learning rate of 0.001 and categorical cross-entropy was used as a loss function. Early stopping was implemented on both models to avoid overfitting issues. This resulted in the Bi-LSTM and Bi-GRU model to be trained for 114 and 137 epochs, respectively.

However, similar to approach 1, the results of these models were not combined with the results of image data due to comparatively low accuracy discussed later.

5.2.3 Approach 3

For this strategy, we utilized data preprocessing and feature extraction, similar to the second approach. But we used different fusion models for better classification of the attention states. For the EEG signal data, we used two different combinations of fusion models:

- **Multi-head attention with Bi-LSTM:** To leverage the contextual information along with the sequential information, a multi-head attention layer with 16 attention heads, each with a key dimension of 64, was first used to capture the input and enable the model to learn multiple representations. This layer allows the model to learn long-range dependencies in the sequence. Then a layer normalization layer of 1e-6 was added to make the training process stable. The attention layer output was captured now by a Bidirectional LSTM with 128 neurons like the previous approach. The output then was again used as an input to a dense layer having 128 neurons which was activated by the tanh function. A dropout layer of 0.4 was added here as well and the output from the dense layer was then flattened and a softmax function was used for the output.

- **Multi-head attention with Bi-GRU:** This model was utilized similarly as the previous model. But instead of a Bidirectional LSTM layer, a bidirectional GRU layer was used.

Both the models were compiled with an Adam Optimizer having a learning rate of 0.001 and categorical cross-entropy was used as a loss function as before. Early stopping was implemented on both models and an Adaptive Learning Rate Adjustment was introduced here to prevent overfitting during training. This resulted in the Bi-LSTM fusion model and Bi-GRU fusion models to train for 61 and 111 epochs, respectively.

In this approach, we introduced the image dataset and we used two different approaches for it:

- **ViT (Supervised):** This model was implemented to process and classify images using a transfer-based supervised approach. For the image pre-processing, all images were resized to a fixed dimension of 224x224 pixels and were normalized. The training dataset had 32 images per batch while the test and validation dataset each had 16 images per batch. This model involves three steps: patch embedding, transformer encoder, and classification head. The image input was first divided into fixed-size patches (16x16) to be flattened and embedded into vectors. Then a stack of 6 transformer layers, containing multiple Multi-Head Self-Attention layers and Feed-Forward Networks(FFN), processed those embeddings. The classification head is mainly a dense layer that aids in classification of the images, which in our case were: Attentive, Inattentive, and Sleepy.

  We configured the ViT as follows:

  - **Image Size:** 224x224 pixels
  - **Patch Size:** 16x16 pixels
  - **Hidden Size:** 192 (dimension of embeddings)
  - **Attention Heads:** 3
  - **Intermediate Size:** 768 (hidden size of the feed-forward network)
  - **Dropout Rate:** 0.1 to reduce overfitting

  Cross-Entropy Loss was used as loss function and Adam Optimizer with a learning rate of 0.0001 was used for model compilation. Early stopping was employed to stop training so that overfitting is avoided and this model was trained for 54 epochs.

- **Contrastive Learning (Unsupervised):** For this model, we first used a contrastive loss function, Momentum Contrast (MoCo), with temperature = 0.07, memory size = 65536, and embedding dimension = 128. Here, the input embeddings are normalized, the similarity between embeddings is computed using a temperature-scaled dot product, and the loss is computed by maximizing positive pair similarity while minimizing negative pair similarity. The

memory bank is maintained for diverse negative samples, which stores previously computed embeddings. After that, we used a small CNN encoder with two convolutional layers that extracts features from the input image.

Convolutional Layer 1:

- **Input Channels:** 3 (for RGB images)
- **Output Channels:** 64

Convolutional Layer 2:

- **Input Channels:** 64
- **Output Channels:** 128

A max pooling layer was also used and finally, there was a fully connected layer that projected the final feature map into a 128-dimensional embedding space.

The training, test, and validation dataset were augmented accordingly to be used for this model. For each batch of image, two augmented views of each image were generated and these were passed through the encoders to obtain embeddings. The Contrastive MoCo loss was then calculated for these embeddings. After all the embeddings were created, the encoder was frozen and a simple linear classifier was introduced that maps the 128-dimensional embeddings to the number of classes, which was three in our case. The classifier was trained using Adam Optimizer with a learning rate of 0.001 with a cross entropy loss function for 101 epochs as we used the early stopping mechanism to avoid overfitting issue.

The results of each model prediction in this entire approach were then saved as .csv files, and we used two methods of late multimodal fusion on the achieved results. As the number of data varied in the datasets, to ensure proper ensembling, the dataset having prediction with lower numbers of data was replicated to ensure alignment. Subsequently, as the datasets were taken from two different samples, we performed an agreement filtering on the original dataset where the original predictions of the EEG dataset matched with the original predictions of the image dataset.

A standard ensemble was done using the following formula:

$$\text{fusion\_prediction} = \frac{\text{eeg\_prediction} + \text{facial\_prediction}}{2} \tag{5.1}$$

For weighted ensembling, the weights of each modality were calculated first using the following formulas:

$$\text{total\_confidence} = \text{eeg\_confidence} + \text{facial\_confidence} \tag{5.2}$$

$$\text{eeg\_weights} = \frac{\text{eeg\_confidence}}{\text{total\_confidence}} \tag{5.3}$$

$$\text{facial\_weights} = \frac{\text{facial\_confidence}}{\text{total\_confidence}} \tag{5.4}$$

where the eeg_confidence and facial_confidence was the maximum probability of each modality.

Then the weighted ensemble was calculated using the following formula:

$$\begin{aligned}\text{fusion\_prediction} = &(\text{eeg\_weights} \times \text{eeg\_prediction}) \\ &+ (\text{facial\_weights} \times \text{facial\_prediction})\end{aligned} \tag{5.5}$$

5.3 Performance Evaluation

The models were evaluated using four metrics: accuracy, precision, recall, and F1-Score.

1. **Accuracy:** Accuracy is a metric that measures the number of correctly classified data instances over the total number of data instances. Even though it determines the correctness in classification, it does not work well with unbalanced data, often leading to misleading interpretations.

$$\text{Accuracy} = \frac{\text{TN+TP}}{\text{TN+FP+TP+FN}} \tag{5.6}$$

2. **Precision:** Precision gives the positive predictive value. It calculates the proportion of true positive predictions out of all positive predictions. It is ideally high, meaning that the model mostly predicts a positive class correctly. Precision is useful in cases where getting true positives is important and false positives can be prohibitive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{5.7}$$

3. **Recall:** Recall evaluates the proportion of true positives that have been correctly predicted by the model. It is also known as true positive rate or sensitivity. Recall is crucial in situations where not succeeding to find a true positive can be fatal.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{5.8}$$

4. **F1-Score:** F1-Score is the harmonic mean of precision and recall, meaning that it balances out both metrics. This is essential as an optimal classifier requires both precision and recall to be high, implying both false positives and false negatives should be zero. The F1-Score is a much more robust measure than accuracy.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \tag{5.9}$$

Here,
TP = True Positive; TN = True Negative;
FP = False Positive; FN = False Negative;

"True" means correctly predicted, "False" means incorrectly predicted, "Positive" means positive instance of data, "Negative" means negative instance of data.

5.4 Results Analysis

This is the analysis of the results of the models with which we worked on this research.

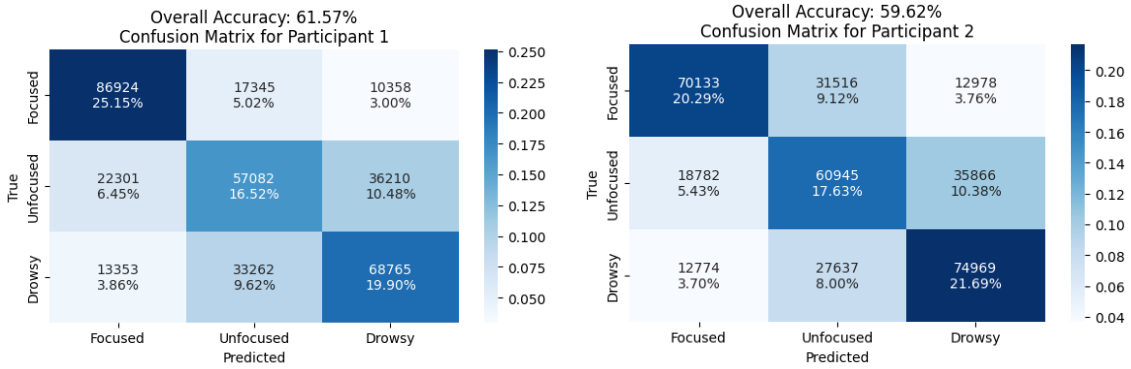
5.4.1 Approach 1

For these models, we resorted to a subject-specific approach. Table 5.2 demonstrates the accuracy comparisons of AdaBoost and XGBoost for each participant.

Model	**Participant 1**	**Participant 2**	**Participant 3**	**Participant 4**	**Participant 5**
AdaBoost	61.57%	59.62%	62.57%	57.74%	59.98%
XGBoost	**76.39%**	**82.95%**	**79.67%**	**74.13%**	**78.53%**

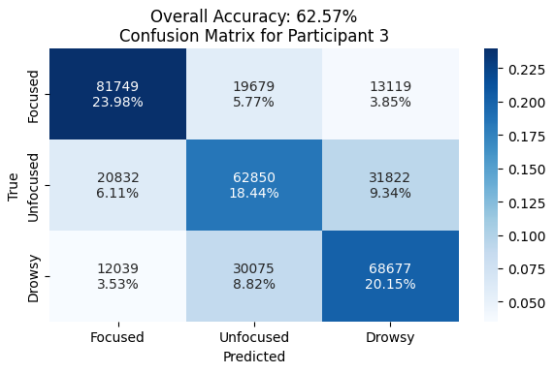
Table 5.2: Accuracy Comparisons for AdaBoost and XGBoost on Each Participants

AdaBoost worked well for some participants but there were some variabilities across the entire dataset. However, XGBoost seemed to perform better for this dataset as it overall yielded a better accuracy. This was done without any type of feature extraction methods and noise filtering, hence, the accuracy is not up to the mark. For each participant, we generated confusion matrices (shown in Figures 5.2 and 5.3) from the results of both models that highlighted how each classifier properly distinguished the mental states. Each matrix compared the true labels with the predicted ones that properly highlighted where the models worked well. Additionally, Figure 5.4 shows the overall accuracy comparison between AdaBoost and XGBoost, indicating that AdaBoost showed similar accuracy across all participants while XGBoost worked better than it in most cases.

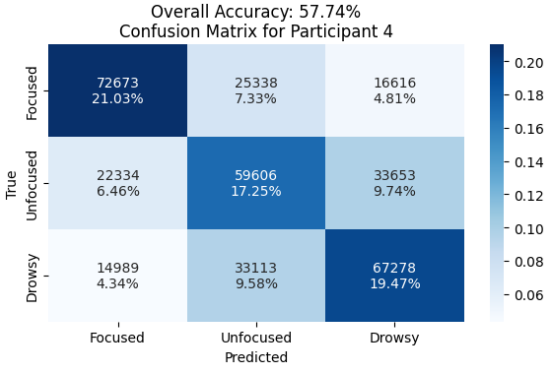


(a) Participant 1

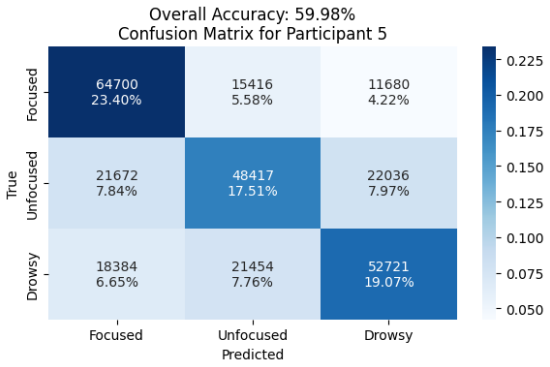
(b) Participant 2



(c) Participant 3



(d) Participant 4



(e) Participant 5

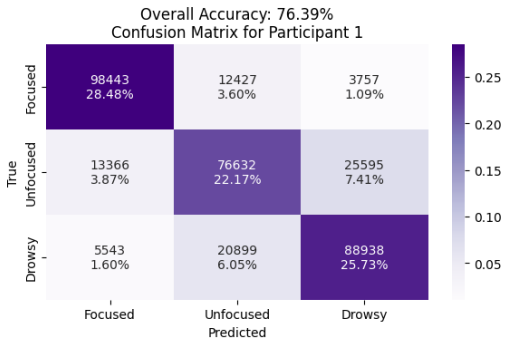
Figure 5.2: Confusion Matrices Using AdaBoost

Figures 5.2a to 5.2e display the confusion matrices of three classes (Focused, Unfocused, and Drowsy) for each participant using AdaBoost. For each participant:

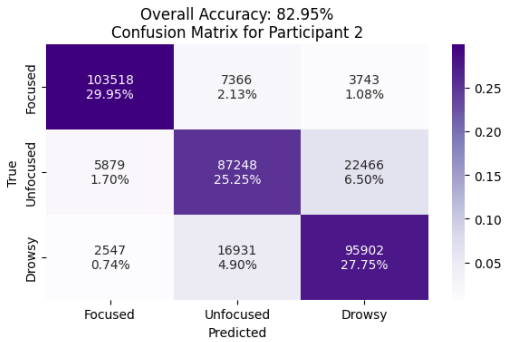
- Participant 1 (Figure 5.2a): True predictions: 86924, 57082, 68765
  False predictions: 17345, 10358, 22301, 36210, 13353, 33262
- Participant 2 (Figure 5.2b): True predictions: 70133, 60945, 74969
  False predictions: 31516, 12978, 18782, 35866, 12774, 27637
- Participant 3 (Figure 5.2c): True predictions: 81749, 62850, 68677
  False predictions: 19679, 13119, 20832, 31822, 12039, 30075

- Participant 4 (Figure 5.2d): True predictions: 72673, 59606, 67278
  False predictions: 25338, 16616, 22334, 33653, 14989, 33113

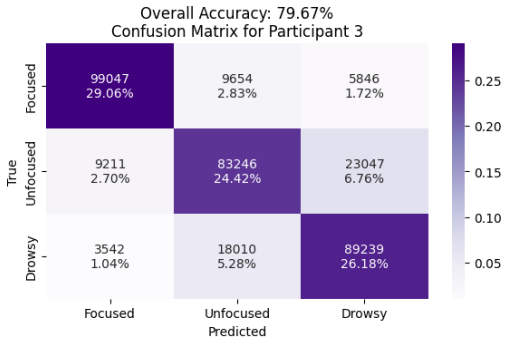
- Participant 5 (Figure 5.2e): True predictions: 64700, 48417, 52721
  False predictions: 15416, 11680, 21672, 22036, 18384, 21454



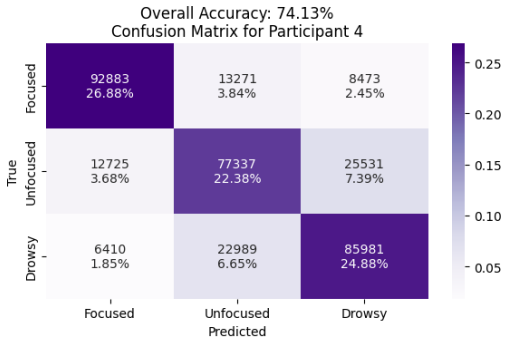
(a) Participant 1



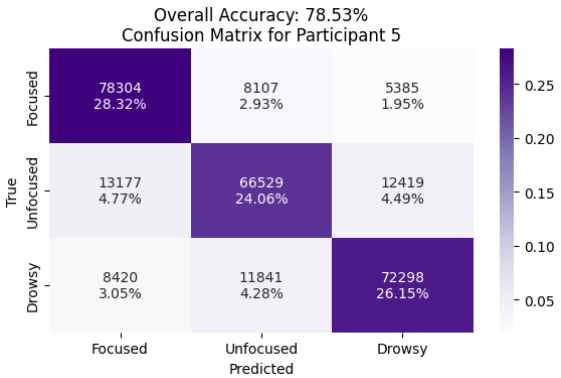
(b) Participant 2



(c) Participant 3



(d) Participant 4



(e) Participant 5

Figure 5.3: Confusion Matrices Using XGBoost

Figures 5.3a to 5.3e display the confusion matrices of three classes (Focused, Unfocused, and Drowsy) for each participant using XGBoost. For each participant:

- Participant 1 (Figure 5.3a): True predictions: 98443, 76632, 88938
  False predictions: 12427, 3757,13366, 25595, 5543, 20899

- Participant 2 (Figure 5.3b): True predictions: 103518, 87248, 95902
  False predictions: 7366, 3743, 5879, 22466, 2547, 16931
- Participant 3 (Figure 5.3c): True predictions: 99047, 83246, 89239
  False predictions: 9654, 5846, 9211, 23047, 3542, 18010
- Participant 4 (Figure 5.3d): True predictions: 92883, 77337, 85981
  False predictions: 13271, 8473, 12725, 25531, 6410, 22989
- Participant 5 (Figure 5.3e): True predictions: 78304, 66529, 72298
  False predictions: 8107, 5385, 13177, 12419, 8420, 11841

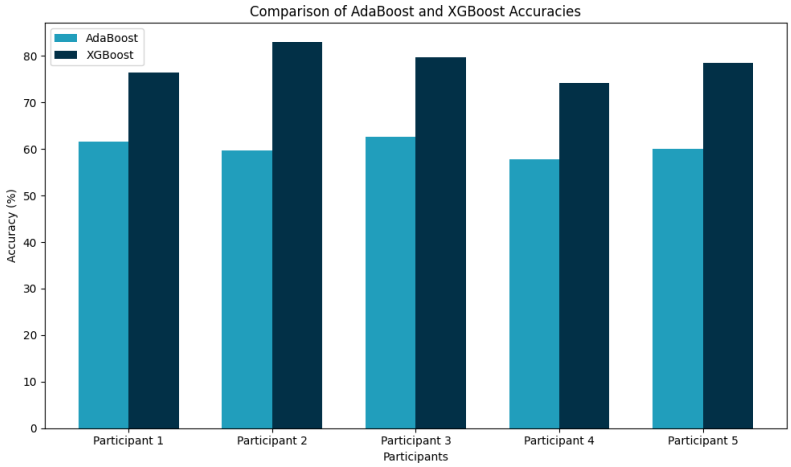
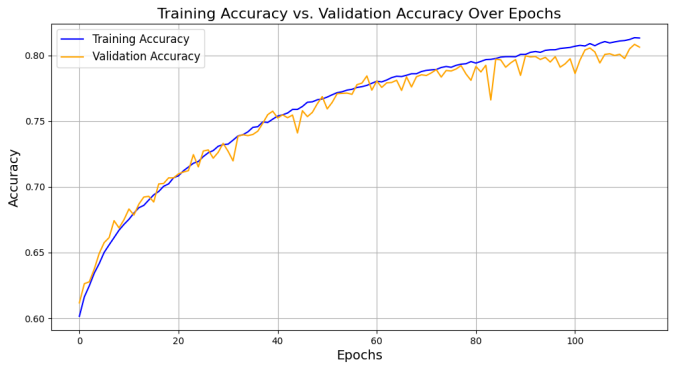


Figure 5.4: Accuracy Comparison Between AdaBoost and XGBoost

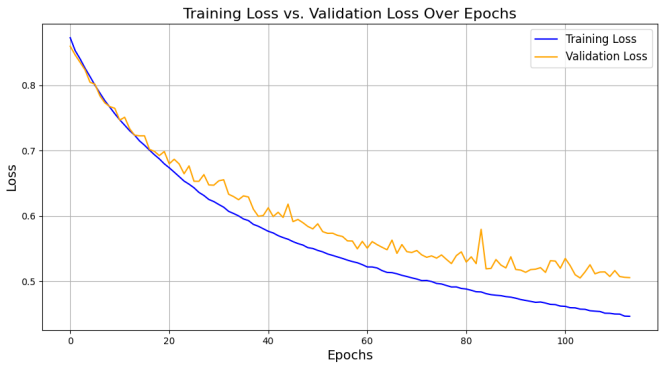
From the results seen, XGBoost consistently achieved higher accuracy across all participants compared to AdaBoost. As XGBoost optimizes decision trees more effectively, it can handle the variations and noise in the EEG data more effectively than AdaBoost. The largest performance gap was seen in Participant 2, where XGBoost had over 20% better accuracy than AdaBoost. Both models commonly misclassified the unfocused and drowsy states the most.

5.4.2 Approach 2

For this approach, we calculated performance metrics using all the combined EEG data for the Bi-LSTM and Bi-GRU models. We also generated graphs (shown in Figure 5.5 and 5.6) that compares the training loss and accuracy with the validation loss and accuracy for each model.



(a) Training and Validation Accuracy

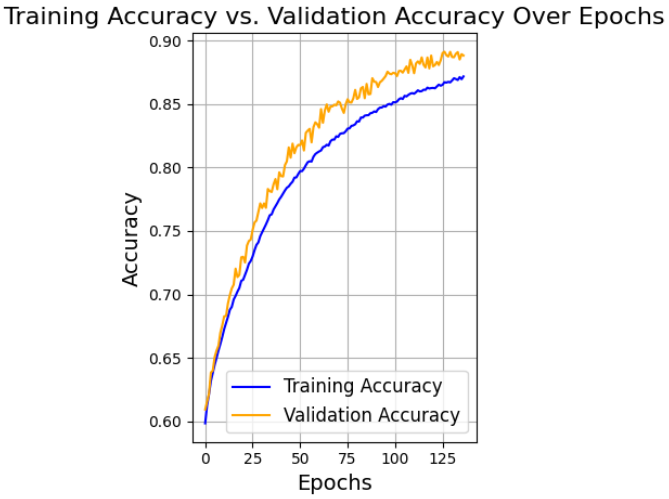


(b) Training and Validation Loss

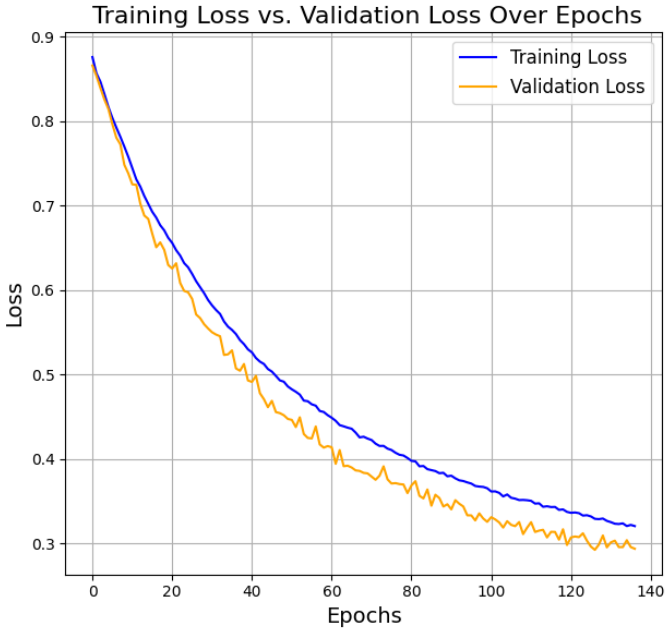
Figure 5.5: Accuracy and Loss Graphs for Bi-LSTM Model

The training and validation accuracy graph in Figure 5.5a shows that both training and validation accuracies increase steadily over the epochs. The validation accuracy is similar to the training accuracy throughout the epochs, with minor fluctuations. It can be seen that both the training and validation accuracies reach over 80%. This suggests that the Bi-LSTM model is learning well and the small gap between the two accuracies indicates that the model is generalizing the unseen data well. However, the slight fluctuations in the validation accuracy emerged due to a small batch size of the validation set.

On the other hand, Figure 5.5b displays a decreasing trend for both training and validation loss graphs. The training loss decreases smoothly while the validation loss experiences fluctuations. This indicates minor variations in the validation set, but as both the loss curves align closely with low diversion, the model exhibits good generalization without significant overfitting.



(a) Training and Validation Accuracy



(b) Training and Validation Loss

Figure 5.6: Accuracy and Loss Graphs for Bi-GRU Model

The training and validation accuracy graph in Figure 5.6a above shows that both training and validation accuracies increase steadily over the epochs. The validation accuracy is similar but slightly higher than the training accuracy most of the time, with slight fluctuations. By the final epochs, the validation accuracy reaches approximately 89%. The overall trend suggests that the Bi-GRU model generalizes well on unseen data without significant overfitting.

On the other hand, Figure 5.6b displays a decreasing trend for both training and validation loss graphs. The training loss decreases smoothly, while the validation loss experiences fluctuations. But, as the curves have low differences, it indicates that the model could generalize the data well. However, the validation loss is lower than the training loss during the epochs due to random neuron deactivation in the dropout layer in the training phase.

Model	Accuracy	Attention state	Precision	Recall	F1-Score
Bi-LSTM	80.19%	Focused	0.74	0.77	0.75
		Unfocused	0.66	0.52	0.58
		Drowsy	0.86	0.91	0.89
Bi-GRU	**88.99%**	Focused	0.87	0.84	0.85
		Unfocused	0.78	0.78	0.78
		Drowsy	0.94	0.95	0.94

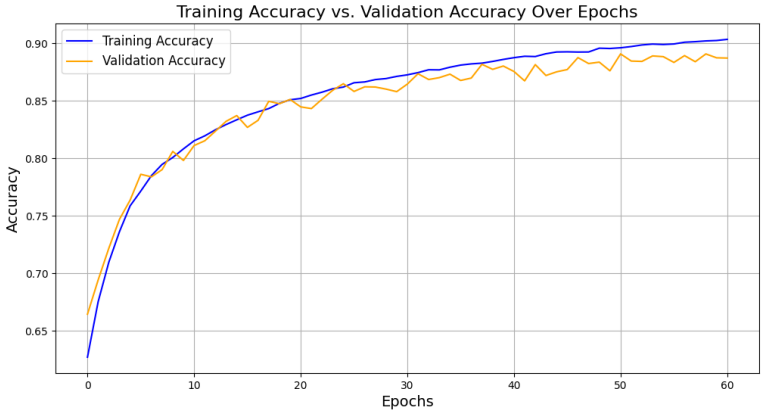
Table 5.3: Performance comparison of Bi-LSTM and Bi-GRU

Table 5.3 compares the other performance metrics of both models for each state. We can see that Bi-GRU worked better in classification of attention states with an accuracy of 88.99%, compared to Bi-LSTM that achieved an accuracy of 80.19%. Since EEG signals have long temporal dependencies, the vanishing gradients can hinder the training process. As Bi-GRU directly passes relevant information through the update gate and avoids unnecessary memory retention, it tackles the vanishing gradients better than Bi-LSTM, leading to more accurate results after a stable training process. Among the three attention states, the F1-Scores of the drowsy state was the highest with a value of 0.89 and 0.94 for the Bi-LSTM and Bi-GRU models, respectively. Bi-LSTM was unable to distinguish the unfocused state properly, leading to the lowest recall and F1-Score of 0.52 and 0.58, respectively.

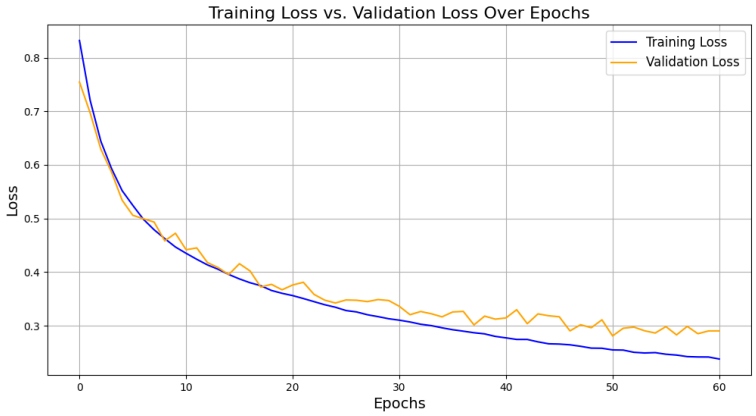
5.4.3 Approach 3

EEG Classification

For this approach as well, we calculated the performance metrics for the combined EEG data, after adding a multi-head attention layer with the previous models. We also generated graphs (shown in Figure 5.7 and 5.9) that compare the training loss and accuracy with the validation loss and accuracy for each fusion model.



(a) Training and Validation Accuracy



(b) Training and Validation Loss

Figure 5.7: Accuracy and Loss Graphs for Bi-LSTM Fusion Model

The training and validation accuracy graph in Figure 5.7a shows steady increase of both training and validation accuracy curves over the epochs. The validation accuracy is similar to the training accuracy throughout the epochs, with slight fluctuations. The training accuracy here, reaches over 90% in the final epochs, meaning that the Bi-LSTM fusion model is learning properly. The small gap between both accuracies indicates that the model is generalizing the data well. However, the slight fluctuations in validation accuracy, again, emerged due to a small batch size of the validation set.

On the contrary, Figure 5.7b displays a decreasing trend for both training and validation loss graphs. The training loss decreases steadily here while the validation loss experiences minor fluctuations with a slight difference compared to the training curve. This indicates some variations in the validation set, but as both the loss curves align closely with low diversion, the model performs good generalization without a significant overfitting trend.

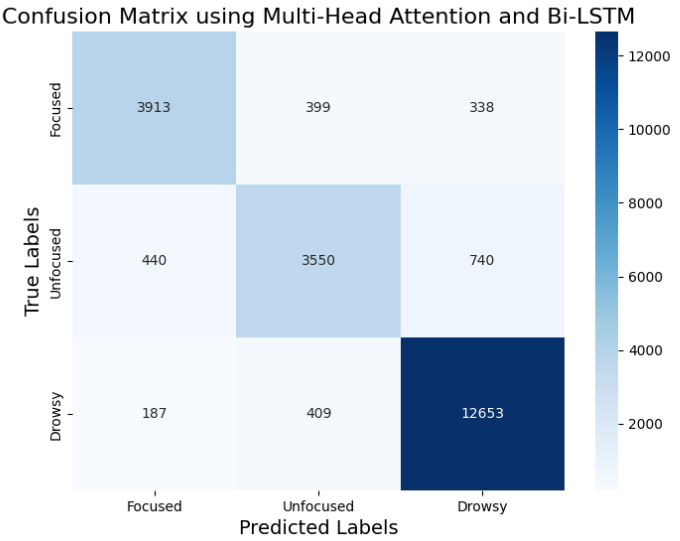
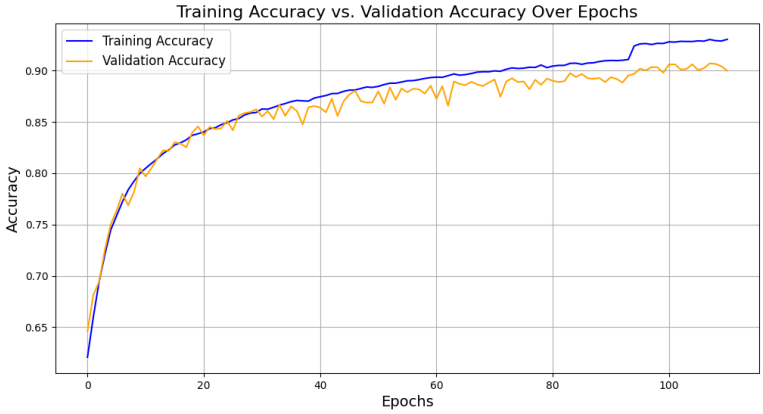


Figure 5.8: Confusion Matrix using Bi-LSTM Fusion Model

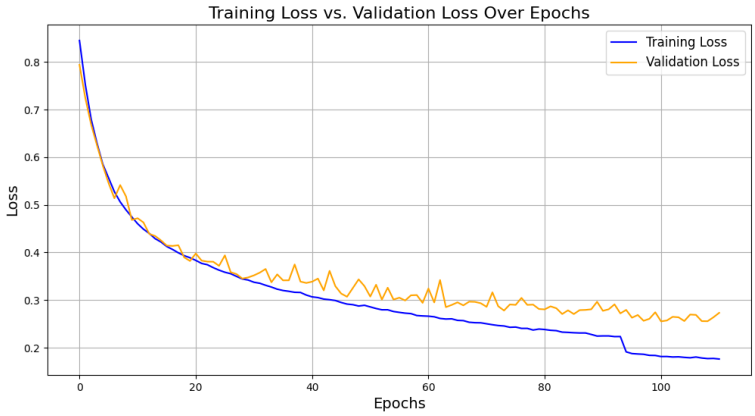
Figure 5.8 displays the confusion matrix of three classes (Focused, Unfocused, and Drowsy) using the Bi-LSTM fusion model:

- True predictions: 3913, 3550, 12653
- False predictions: 399, 338, 440, 740, 187, 409

It can be seen that the Bi-LSTM fusion model mostly struggled in the detection of unfocused labels and considered 740 of those instances as drowsy states and 440 instances as focused states.



(a) Training and Validation Accuracy



(b) Training and Validation Loss

Figure 5.9: Accuracy and Loss Graphs for Bi-GRU Fusion Model

The training and validation accuracy graph in Figure 5.9a shows steady increase of both training and validation accuracy curves with a sharp rise after around 90 epochs due to reaching a learning plateau. Other than that, the validation accuracy is similar to the training accuracy throughout the epochs, with slight fluctuations. The training accuracy here reaches around 94% in the final epochs, meaning that the Bi-GRU fusion model is learning well and generalizing effectively on unseen data.
On the contrary, Figure 5.9b displays a decreasing trend for both training and validation loss graphs. The training loss decreases steadily here, while the validation loss experiences little fluctuations with a slight difference compared to the training curve. This indicates some variations in the validation dataset, but as both loss curves align closely with low diversion, the model performs good generalization without significant overfitting. The sharp drop after around 90 epochs happens due to a reduction of learning rate as it reaches a learning plateau.

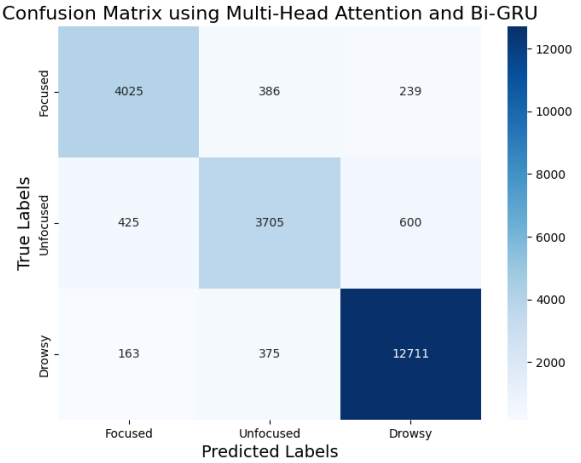


Figure 5.10: Confusion Matrix using Bi-GRU Fusion Model

Figure 5.10 displays the confusion matrix of three classes (Focused, Unfocused, and Drowsy) using the Bi-GRU fusion model:

- True predictions: 4025, 3705, 12711
- False predictions: 386, 239, 425, 600, 163, 375

It can be seen that the Bi-GRU fusion model performed better than the Bi-LSTM fusion model, but mostly struggled in the detection of unfocused labels, considering 425 of those instances as focused states, and 600 instances as drowsy states.

Table 5.4 below compares all the performance metrics of both models for each state: Focused, Unfocused, and Drowsy. The Bi-LSTM fusion model showed an overall accuracy of 88.89% while the Bi-GRU fusion model predicted the attention states better, achieving an overall accuracy of 90.33%. Compared to Approach 2, using only Bi-LSTM or Bi-GRU models, the fusion caused the Bi-LSTM to work better, while the performance of Bi-GRU slightly improved. The addition of the multi-head attention layer caused the Bi-LSTM model to predict the unfocused states better now, yet this state is being misclassified the most by both models, having low F1-Scores of 0.78 and 0.81 for Bi-LSTM and Bi-GRU fusion models, respectively.

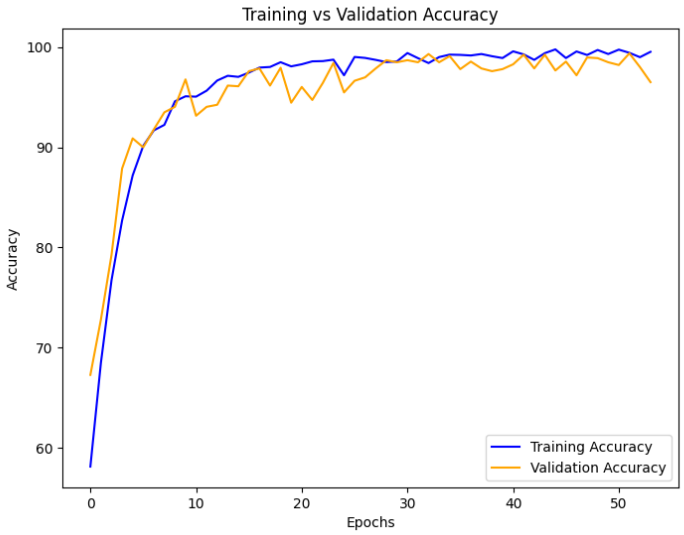
Model	Accuracy	Attention state	Precision	Recall	F1-Score
Bi-LSTM Fusion	88.89%	Focused	0.86	0.84	0.85
		Unfocused	0.81	0.75	0.78
		Drowsy	0.92	0.96	0.94
Bi-GRU Fusion	**90.33%**	Focused	0.87	0.87	0.87
		Unfocused	0.83	0.78	0.81
		Drowsy	0.94	0.96	0.95

Table 5.4: Performance comparison of Bi-LSTM Fusion and Bi-GRU Fusion

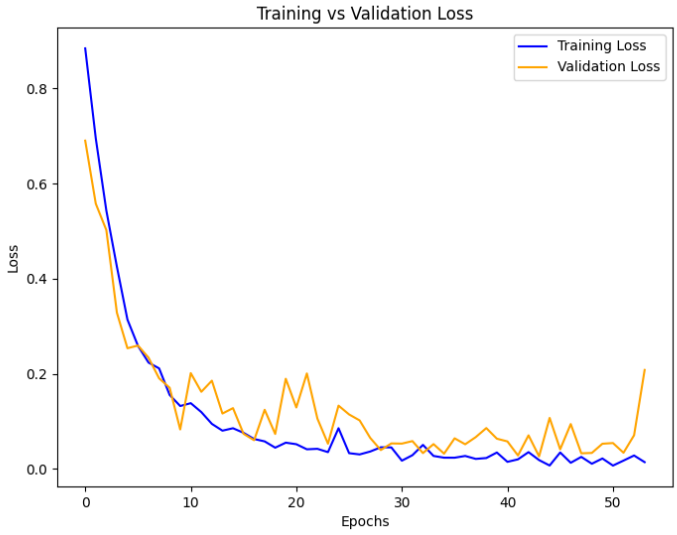
Image Classification

For this dataset, we generated graphs (shown in Figures 5.11 and 5.14) that compared the training loss and accuracy with the validation loss and accuracy for ViT and Contrastive Learning and then, we compared the performance metrics.

ViT:



(a) Training and Validation Accuracy



(b) Training and Validation Loss

Figure 5.11: Accuracy and Loss Graphs for ViT

The training and validation accuracy graph in Figure 5.7a shows an increase of both training and validation accuracy curves over the epochs. The validation accuracy follows closely with the training accuracy throughout the epochs, but it shows some fluctuations overall. The training accuracy here reaches almost 99% in the final

epochs, meaning that the ViT model is learning the training set well and generalizing unseen data effectively. The fluctuations in validation accuracy, here as well, emerged because of the small batch size of the validation set of the image data.
Similarly, Figure 5.7b displays a decreasing trend for both training and validation loss graphs. The training loss decreases almost steadily here, while the validation loss experiences fluctuations, especially around 20 epochs and during the final epochs with differences, compared to the training curve. This indicates some variations in the validation dataset, but as both loss curves align closely with low diversion, ViT shows good generalization overall with low overfitting.

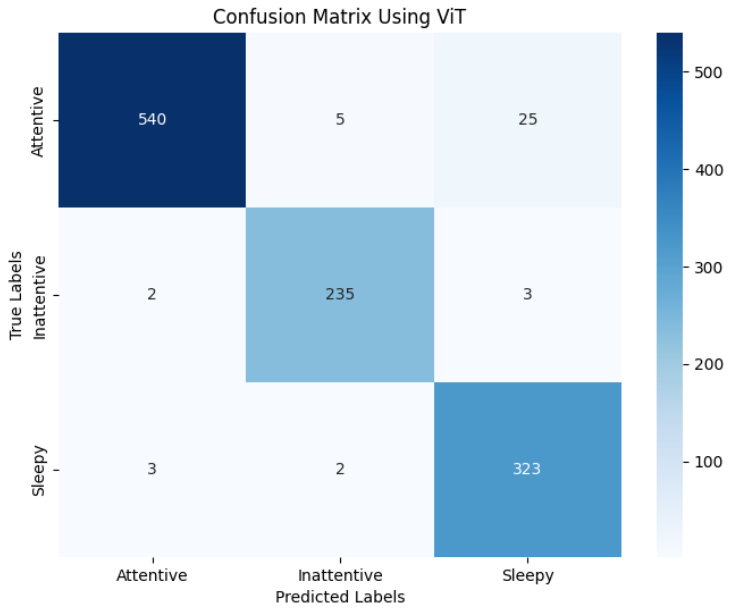


Figure 5.12: Confusion Matrix of ViT

Figure 5.12 displays the confusion matrix of three classes (Attentive, Inattentive, and Sleepy) using ViT:

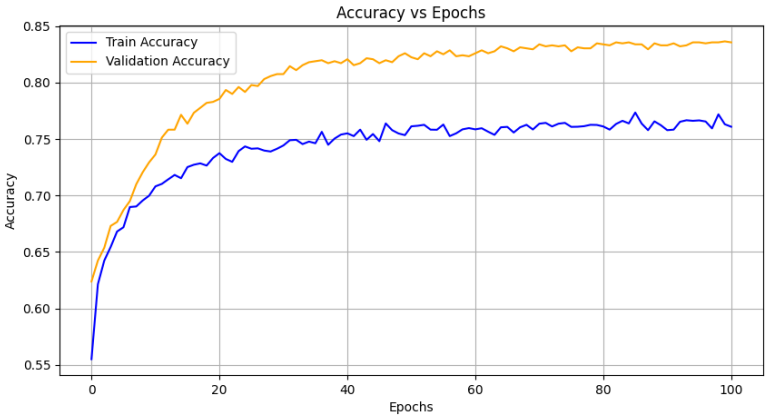
- True predictions: 540, 235, 323
- False predictions: 5, 25, 2, 3, 3, 2

From this confusion matrix, it can be seen that ViT was mostly able to classify all the attention states well, except for 25 attentive states, which were predicted to be sleepy. This was the most misclassified attention state using ViT.

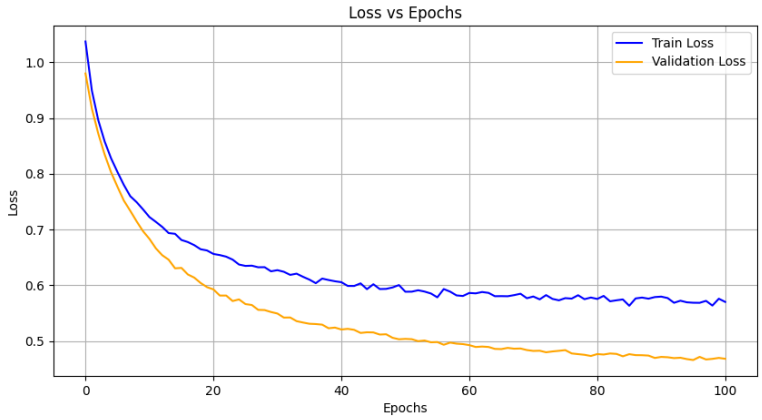
Figure 5.13: Visualization of True Vs Predicted labels using ViT

Figure 5.13 demonstrates a selection of randomly chosen images along with their true and predicted labels. Here, 0 corresponds to Attentive state, 1 to Inattentive state, and 2 to Sleepy state.

Contrastive Learning:



(a) Training and Validation Accuracy



(b) Training and Validation Loss

Figure 5.14: Accuracy and Loss Graphs for Contrastive Learning

The training and validation accuracy graph in Figure 5.14a shows an increase of both training and validation accuracy curves over the epochs. The validation accuracy follows closely with the training accuracy throughout the epochs with some differences in values. The training accuracy here reaches almost 85% in the final epochs, meaning that the contrastive learning model is learning effectively from the training set and generalizing well on new data. However, the fluctuations in the validation accuracy, here, may have been caused due to variations of the image data in the validation set.
On the other hand, Figure 5.14b shows a steady decreasing trend for both training and validation loss curves with low fluctuations. There is a slight difference between both curves' values over the epochs, which indicates some variations in the validation dataset. But, as both loss curves align closely with low diversion, contrastive learning generalizes on the unseen data well, overall, without high overfitting.

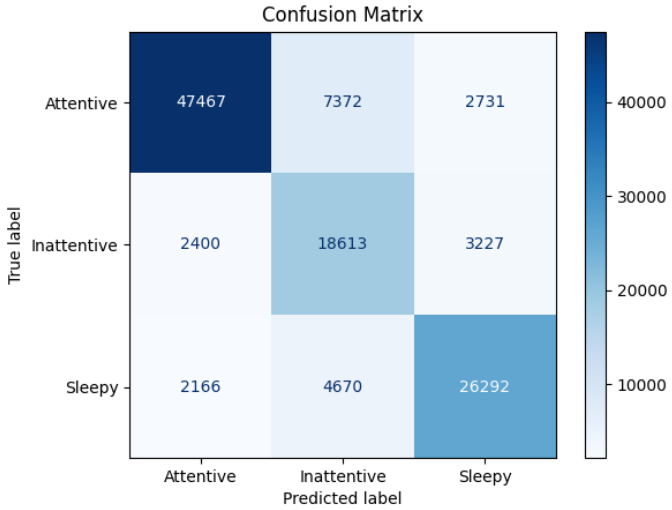


Figure 5.15: Confusion Matrix of Contrastive Learning

Figure 5.15 displays the confusion matrix of three classes (Attentive, Inattentive, and Sleepy) using Contrastive Learning:

- True predictions: 47467, 18613, 26292
- False predictions: 7372, 2731, 2400, 3227, 2166, 4670

From this confusion matrix, it can be seen that contrastive learning was able to classify all the attention states well. However, it struggled the most in differentiating the inattentive states, with the other states, where 7372 attentive instances were misclassified as inattentive states whereas 4670 sleepy states were misclassified as inattentive states. This happened due to the data transformation and augmentation, which made some of the images too dark for the model to be able to differentiate between the attention states, while using the embeddings of the facial expressions during classifications.

Figure 5.16: Visualization of True Vs Predicted labels using Contrastive Learning

Figure 5.16 demonstrates a selection of randomly chosen images along with their true and predicted labels using contrastive learning. Here, 0, 1, and 2 corresponds to Attentive, Inattentive, and Sleepy state, respectively.

Table 5.5 below shows the comparison of the performance metrics of these two models.

Model	Accuracy	Attention state	Precision	Recall	F1-Score
ViT	**96.49%**	Attentive	0.99	0.95	0.97
		Inattentive	0.97	0.98	0.98
		Sleepy	0.92	0.98	0.95
Contrastive Learning	80.37%	Attentive	0.91	0.82	0.87
		Inattentive	0.61	0.77	0.68
		Sleepy	0.82	0.79	0.80

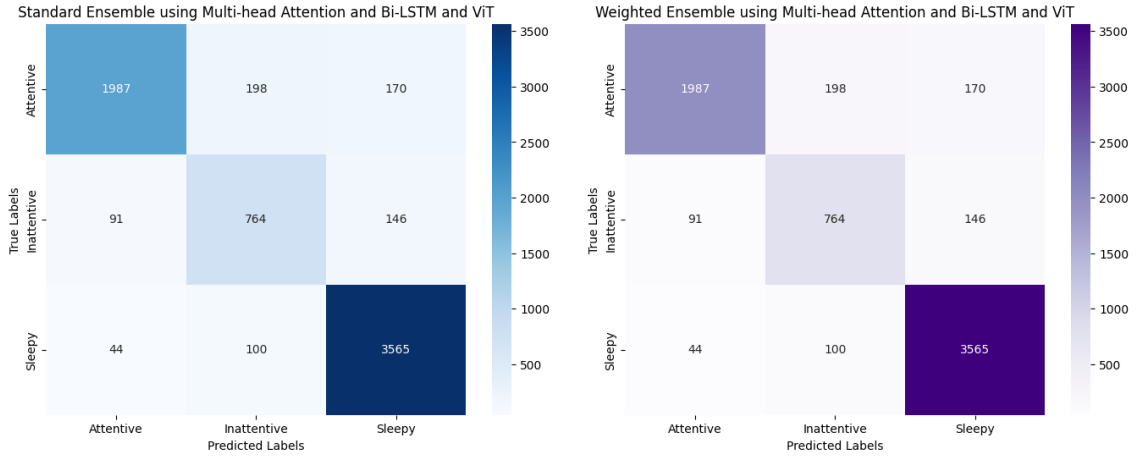
Table 5.5: Performance comparison of ViT and Contrastive Learning models with attention states.

ViT had a higher accuracy of 96.49% compared to contrastive learning, which had an accuracy of 80.37%. ViT and contrastive learning had high precisions of 0.99 and 0.91, respectively, in the classification of attentive states. However, contrastive learning had the lowest F1-Scores of 0.68 and 0.80 in the classification of inattentive and sleepy states, respectively, among the F1-Scores. This indicates that contrastive learning misclassified these two states the most.

The results from both EEG and Image datasets were combined using late multimodal fusion. However, as the datasets were collected from different experiments, it was necessary to combine the results in such a way that the ground true values were the same for more reliability. Therefore, the results of the image dataset were replicated to match the amount of data from the EEG signals. After that, only the data having the same ground true values were used for fusion to avoid discrepancies.

Figures 5.17 to 5.20 below demonstrate the confusion matrices of the combinations of the models using both standard and weighted ensembles.

Bi-LSTM fusion and ViT:

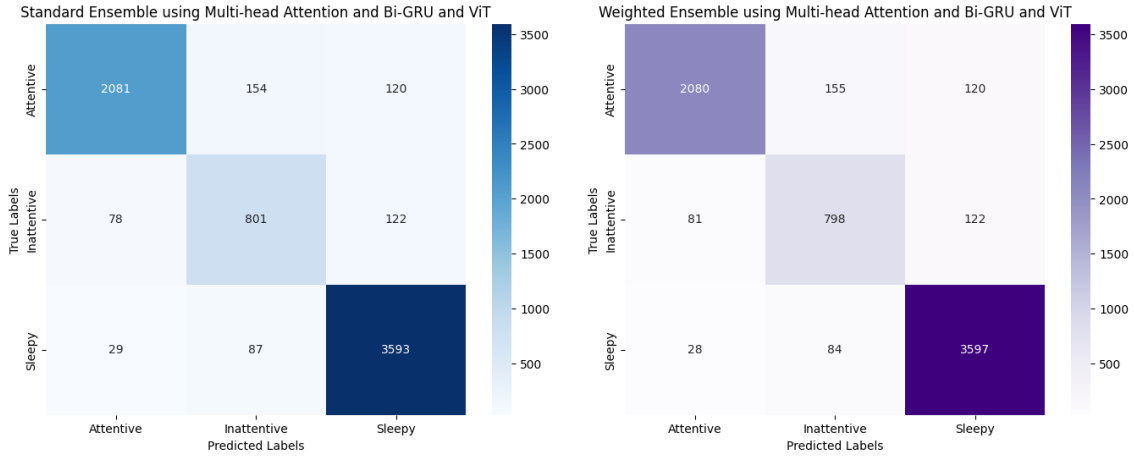


(a) Standard Ensemble (b) Weighted Ensemble

Figure 5.17: Confusion Matrices for Bi-LSTM fusion and ViT.

- Standard Ensemble from Figure 5.17a:
  - True predictions: 1987, 764, 3565
  - False predictions: 198, 170, 91, 146, 44, 100
- Weighted Ensemble from Figure 5.17b:
  - True predictions: 1987, 764, 3565
  - False predictions: 198, 170, 91, 146, 44, 100

Bi-GRU fusion and ViT:

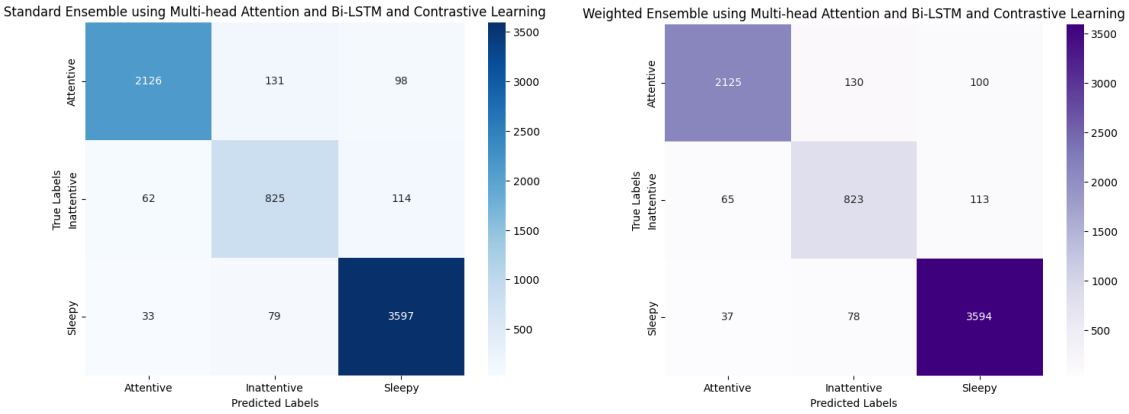


(a) Standard Ensemble (b) Weighted Ensemble

Figure 5.18: Confusion Matrices for Bi-GRU fusion and ViT.

- Standard Ensemble from Figure 5.18a:
  - True predictions: 2081, 801, 3593
  - False predictions: 154, 120, 78, 122, 29, 87
- Weighted Ensemble from Figure 5.18b:
  - True predictions: 2080, 798, 3597
  - False predictions: 155, 120, 81, 122, 28, 84

Bi-LSTM fusion and Contrastive Learning:

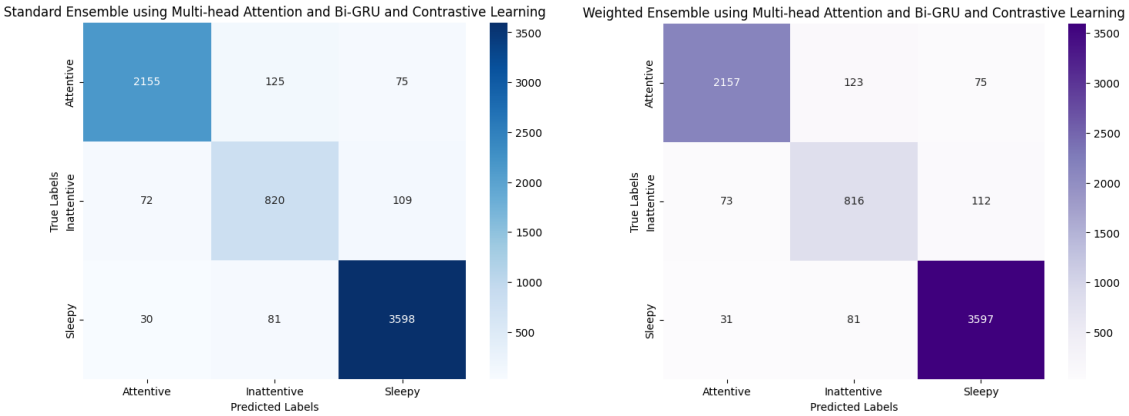


(a) Standard Ensemble (b) Weighted Ensemble

Figure 5.19: Confusion Matrices for Bi-LSTM fusion and Contrastive Learning.

- Standard Ensemble from Figure 5.19a:
  - True predictions: 2126, 825, 3597
  - False predictions: 131, 98, 62, 114, 33, 79
- Weighted Ensemble from Figure 5.19b:
  - True predictions: 2125, 823, 3594
  - False predictions: 130, 100, 65, 113, 37, 78

Bi-GRU fusion and Contrastive Learning:



(a) Standard Ensemble (b) Weighted Ensemble

Figure 5.20: Confusion Matrices for Bi-GRU fusion and Contrastive Learning.

- Standard Ensemble from Figure 5.20a:
  - True predictions: 2155, 820, 3598
  - False predictions: 125, 75, 72, 109, 30, 81

- Weighted Ensemble from Figure 5.20b:
  - True predictions: 2157, 816, 3597
  - False predictions: 123, 75, 73, 112, 31, 81

Table 5.6 below shows the overall comparison of performance metrics, using standard and weighted ensembles on different combinations of models.

Model Combination	Fusion Type	Accuracy	Precision	Recall	F1-Score
Bi-LSTM Fusion Model and ViT	Standard	89.40%	0.90	0.89	0.89
	Weighted	89.40%	0.90	0.89	0.89
Bi-GRU Fusion Model and ViT	Standard	91.65%	0.92	0.92	0.92
	Weighted	91.65%	0.92	0.92	0.92
Bi-LSTM Fusion Model and Contrastive Learning	Standard	92.68%	0.93	0.93	0.93
	Weighted	92.60%	0.93	0.93	0.93
Bi-GRU Fusion Model and Contrastive Learning	Standard	**93.04%**	0.93	0.93	0.93
	Weighted	**92.99%**	0.93	0.93	0.93

Table 5.6: Performance Comparison of Multimodal Fusions using Different Combinations

Among all the combinations of multimodal fusion, although ViT had the best accuracy in image classification alone, the Bi-GRU fusion model with Contrastive learning achieved the highest accuracies of 93.04% and 92.99%, using standard and weighted ensembles, respectively. Rest of the combinations were also similarly effective in proper classification of attention states, ranging from accuracies of 89.40% and 92.68%. Both the fusion models, when combined with contrastive learning, perform better in terms of F1-Scores in both standard and weighted ensembles, which is 0.93. Generally, the accuracies and F1-Scores seem to be the same for both standard and weighted ensembles, when the models are combined. Multimodal fusion with contrastive learning showed a very slight difference in terms of accuracy in those two types of ensembles.

Figure 5.21 below shows the accuracy comparison of the above model combinations using standard and weighted ensemble in a bar chart form, indicating that both the standard and weighted ensembles performed well using different combinations of fusion models.

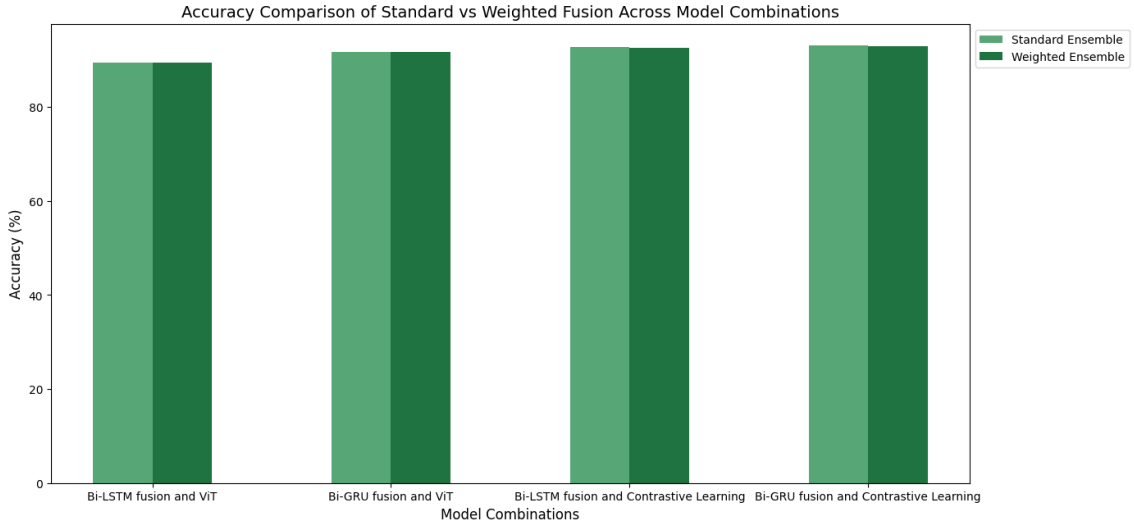


Figure 5.21: Accuracy Comparison of Multimodal Fusion using Different Combinations

5.4.4 Result Comparison with Previous Research

Previous study by Ç.İ. Acı et al. [4] achieved accuracies ranging from 71.80% to 78.80% using a common-subject paradigm by utilizing multiple SVM classifiers on the dataset we used for our research. KNN-based classifier and Adaptive Neuro-Fuzzy System (ANFIS) helped to achieve accuracies ranging from 73.35% to 83.46% and 78.33% to 85.41%, respectively.

However, our proposed method on EEG signal itself outperformed these with accuracies of 88.89% and 90.33% using the Bi-LSTM fusion and Bi-GRU fusion models, respectively, improving the classification accuracy of predicting attention states. Upon further combining with an image dataset, we were able to improve the accuracy upto 93.04%, demonstrating the effectiveness of multimodal fusion.

Chapter 6

Conclusion

6.1 Conclusion

Our research aimed to classify the attention states of students in online classes using a multimodal approach that incorporated EEG signals and facial image data. By effective use of multimodal fusion, we were able to demonstrate the effectiveness of those data for improved classification accuracy of attention states, which would be inefficient if only a single modality was used. Though the EEG signal itself achieved relatively high accuracy, combining it with the facial data boosted the accuracy even more, creating a robust model and providing a more comprehensive and accurate evaluation of student attention in classes compared to single-modality systems.

To conclude, the shift from traditional classroom settings to online learning has given rise to multiple challenges that were not as prevalent before. It has shed light on the need for new methods to address these issues, especially monitoring and maintaining student attention in online classrooms. This research paper uses EEG technology with a multimodal approach to record and analyze brainwave activity, which can provide valuable insights into student attention. The findings reveal the potential of multimodal approaches with EEG to be beneficial for assessing the attention states of students in real-time.

6.2 Limitation and Future Work

Despite achieving a robust performance using a multimodal approach with EEG data, there are some limitations to this work. First of all, the datasets used in this research were collected from two different experiments in two different environments. This required us to duplicate some of the data while incorporating a multimodal fusion. Additionally, as the settings of the experimental data were different, some data had to be excluded to ensure that the true ground labels were the same for both datasets during the late fusion, resulting in a comparatively lower number of data to work with.

However, due to the robust performance across various metrics, it is safe to say that this research has the potential to be implemented in real-life settings. This can be done if a single experiment is carried out from which both the EEG and facial image data are taken into account. This would allow the experiment to be more relevant

and would provide us with a better understanding of this approach in evaluating the attention states of the students. Moreover, as most research is done using a small number of students, it is necessary to implement the experiments on a greater scale for better results that would help this to be implemented in real life to monitor the attention span of students in online classes.

Bibliography

[1] X. Li, B. Hu, D. Qunxi, W. Campbell, P. Moore, and H. Peng, "Eeg-based attention recognition," *Proceedings - 2011 6th International Conference on Pervasive Computing and Applications, ICPCA 2011*, Oct. 2011. DOI: 10.1109/ICPCA.2011.6106504.

[2] M. Alirezaei and S. Hajipour Sardouie, "Detection of human attention using eeg signals," in *2017 24th National and 2nd International Iranian Conference on Biomedical Engineering (ICBME)*, 2017, pp. 1–5. DOI: 10.1109/ICBME.2017.8430244.

[3] U. Burnik, J. Zaletelj, and A. Kosir, "Kinect based system for student engagement monitoring," May 2017, pp. 1229–1232. DOI: 10.1109/UKRCON.2017.8100449.

[4] C. I. Aci, M. Kaya, and Y. Mishchenko, "Distinguishing mental attention states of humans via an eeg-based passive bci using machine learning methods," *Expert Syst. Appl.*, vol. 134, pp. 153–166, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:192533151.

[5] N. H. N. Md Shahri, S. Lai, M. Mohamad, H. Rahman, and A. Rambli, "Comparing the performance of adaboost, xgboost, and logistic regression for imbalanced data," *Mathematics and Statistics*, vol. 9, pp. 379–385, May 2021. DOI: 10.13189/ms.2021.090320.

[6] S. Parui, D. Basu, W. Mansoor, and U. Ghosh, "Artificial intelligence induced multi -level attention states recognition from brain using eeg signal," Nov. 2021, pp. 1–4. DOI: 10.1109/ICSPIS53734.2021.9652419.

[7] M. Bi, M. Wang, Z. Li, and D. Hong, "Vision transformer with contrastive learning for remote sensing image scene classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. PP, pp. 1–13, Jan. 2022. DOI: 10.1109/JSTARS.2022.3230835.

[8] I. Cigem, *Eeg data for mental attention state detection*, Accessed: 2024-10-17, 2022. [Online]. Available: https://www.kaggle.com/datasets/inancigdem/eeg-data-for-mental-attention-state-detection/data.

[9] C. Li, Q. Yang, M. Li, D. Wen, and Y. Wang, "Classification of students' attentional states using attention mechanism and bilstm fusion," in *2022 International Conference on Intelligent Education and Intelligent Research (IEIR)*, 2022, pp. 221–226. DOI: 10.1109/IEIR56323.2022.10050046.

[10] C.-P. Tran, A.-K. Vu, and V.-T. Nguyen, "Baby learning with vision transformer for face recognition," Oct. 2022, pp. 1–6. DOI: 10.1109/MAPR56351.2022.9924795.

[11] E. Attar, "Eeg waves studying intensively to recognize the human attention behavior," Nov. 2023, pp. 1–5. DOI: 10.1109/ICCCIS60361.2023.10425510.

[12] B. Fang, X. Li, G. Han, and J. He, "Rethinking pseudo-labeling for semi-supervised facial expression recognition with contrastive self-supervised learning," *IEEE Access*, vol. PP, pp. 1–1, Jan. 2023. DOI: 10.1109/ACCESS.2023.3274193.

[13] S. Gopi and N. Dehbozorgi, "Eeg spectral analysis for inattention detection in academic domain," in *2023 IEEE Frontiers in Education Conference (FIE)*, 2023, pp. 1–5. DOI: 10.1109/FIE58773.2023.10343261.

[14] Y. Li, Q. Liu, L. Zhou, W. Zhao, Y. Tian, and W. Zhang, "Improved contrastive learning with moco framework," Jun. 2023, pp. 729–732. DOI: 10.1109/ICCECE58074.2023.10135455.

[15] J. Mehta, H. Lakhani, H. Dave, S. Degadwala, and D. Vyas, "Eeg brainwave data classification of a confused student using moving average feature," Jun. 2023, pp. 1461–1466. DOI: 10.1109/ICPCSN58827.2023.00243.

[16] I. V and A. Kist, "Using electroencephalography to determine student attention in the classroom," May 2023, pp. 1–3. DOI: 10.1109/EDUCON54358.2023.10125158.

[17] D. Verma, S. Bhalla, S. V. S. Santosh, S. Yadav, A. Parnami, and J. Shukla, "Attentionet: Monitoring student attention type in learning with eeg-based measurement system," in *2023 11th International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2023, pp. 1–8. DOI: 10.1109/ACII59096.2023.10388212. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ACII59096.2023.10388212.

[18] M. Bhuiyan, R. Bayesh, and S. Das, "Enhancing e-learning: Eeg signal classification to evaluate students' understanding of online lectures," May 2024, pp. 13–18. DOI: 10.1109/ICEEICT62016.2024.10534580.

[19] S. Kim, J.-H. Kim, W. Hyung, *et al.*, "Characteristic behaviors of elementary students in a low attention state during online learning identified using electroencephalography," *IEEE Transactions on Learning Technologies*, vol. 17, pp. 619–628, 2024. DOI: 10.1109/TLT.2023.3289498.

[20] F. Mortezapour Shiri, T. Perumal, and R. Mohamed, "A comprehensive overview and comparative analysis on deep learning models," *Journal on Artificial Intelligence*, vol. 6, pp. 301–360, Nov. 2024. DOI: 10.32604/jai.2024.054314.

[21] W. Xie, Z. Peng, L. Shen, W. Lu, Y. Zhang, and S. Song, "Cross-layer contrastive learning of latent semantics for facial expression recognition," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. PP, Mar. 2024. DOI: 10.1109/TIP.2024.3378459.

[22] J. Liu, "A neural collaborative filtering recommendation algorithm based on attention mechanism and contrastive learning," *Scalable Computing: Practice and Experience*, vol. 26, pp. 191–200, Jan. 2025. DOI: 10.12694/scpe.v26i1.3813.