EEG-BASED VR SMART LEARNING SYSTEM



By

Muhammad AbuBakar Roshan	**01-132222-025**
Syed Muhammad Abdullah	**01-132222-043**
Zarrar Ali	**01-132222-050**

Supervised By

Dr. Waleed Manzoor

Department of Computer Engineering
Bahria University Islamabad
2026

EEG-BASED VR SMART LEARNING SYSTEM

By

Muhammad AbuBakar Roshan	**01-132222-025**
Syed Muhammad Abdullah	**01-132222-043**
Zarrar Ali	**01-132222-050**



Supervised By

Dr. Waleed Manzoor

Submitted to the Department of Computer Engineering in partial fulfillment of the requirements for the degree of

Bachelor of Computer Engineering

Department of Computer Engineering
Bahria University, Islamabad

2026

BAHIRA UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

Dated: **04-06-2026**

CERTIFICATE

We accept the work contained in the report titled **"EEG Based VR Smart Learning System"** as a confirmation to the required standard for the partial fulfillment the degree of Bachelor of Computer Engineering.

Dr. Waleed Manzoor
Project Coordinator
Date: 04-06-2026

Dr. Waleed Manzoor
Supervisor
Date: 04-06.2026

Dr. Muhammad Majid
External Examiner
Date: 04-06.2026

Dr. Kamran Saeed
Internal Examiner
Date: 04-06.2026

Head of Department (CE)
Date: 04-06-2026

EEG-BASED VR SMART LEARNING SYSTEM

SDG No	Description of SDG	SDG No	Description of SDG
SDG 1	No Poverty	SDG 9	✓ Industry, Innovation, and Infrastructure
SDG 2	Zero Hunger	SDG 10	Reduced Inequalities
SDG 3	✓ Good Health and Well Being	SDG 11	Sustainable Cities and Communities
SDG 4	Quality Education	SDG 12	Responsible Consumption and Production
SDG 5	Gender Equality	SDG 13	Climate Change
SDG 6	Clean Water and Sanitation	SDG 14	Life Below Water
SDG 7	Affordable and Clean Energy	SDG 15	Life on Land
SDG 8	Decent Work and Economic Growth	SDG 16	Peace, Justice and Strong Institutions
		SDG 17	Partnerships for the Goals



EEG-BASED VR SMART LEARNING SYSTEM			
Range of Complex Problem Solving			
	Attribute	**Complex Problem**	
1	Range of conflicting requirements	Involve wide-ranging or conflicting technical, engineering and other issues.	
2	Depth of analysis required	Have no obvious solution and require abstract thinking, originality in analysis to formulate suitable models.	
3	Depth of knowledge required	Requires research-based knowledge much of which is at, or informed by, the forefront of the professional discipline and which allows a fundamentals-based, first principles analytical approach.	✓
4	Familiarity of issues	Involve infrequently encountered issues.	
5	Extent of applicable codes	Are outside problems encompassed by standards and codes of practice for professional engineering.	
6	Extent of stakeholder involvement and level of conflicting requirements	Involve diverse groups of stakeholders with widely varying needs.	✓
7	Consequences	Have significant consequences in a range of contexts.	
8	Interdependence	Are high level problems including many component parts or sub-problems.	
Range of Complex Problem Activities			
	Attribute	**Complex Activities**	
1	Range of resources	Involve the use of diverse resources (and for this purpose, resources include people, money, equipment, materials, information and technologies).	✓
2	Level of interaction	Require resolution of significant problems arising from interactions between wide ranging and conflicting technical, engineering or other issues.	✓
3	Innovation	Involve creative use of engineering principles and research-based knowledge in novel ways.	✓
4	Consequences to society and the environment	Have significant consequences in a range of contexts, characterized by difficulty of prediction and mitigation.	
5	Familiarity	Can extend beyond previous experiences by applying principles-based approaches.	

UNDERTAKING

We state that the research work entitled **"EEG Based VR Smart Learning System"** is our own work. The work has not been presented elsewhere for assessment. If material has been obtained from other sources it has been acknowledged and referenced.

Muhammad Abu Bakar Roshan
01-132222-025

Syed Muhammad Abdullah
01-132222-043

Zarrar Ali
01-132222-050

DEDICATION

This is in honor of our parents and families. If we had not had their support, their sacrifices, their prayers, etc., we would not have made it to this degree, nor would this project have been completedWe would like to thank our teachers, our supervisor and the faculty of the Department of Computer Engineering at Bahria University. Their advice inspired our engineering thinking and research process.We also dedicate this work to each other. Three people kept showing up, sharing the load and being committed to getting it right and the project came together. The following is a product of that collaborative effort during the past year. Last but not least, we would like to thank everyone who supported us in one way or another, both directly and indirectly, but in so many little things it adds up. We hope that it will be helpful to the pupils who take up this problem after us.

ACKNOWLEDGEMENT

In the name of Allah, the Most Gracious and the Most Merciful.
We are grateful to our project supervisor Dr. Waleed Manzoor for all his guidance and regular input throughout this project. He was instrumental in the direction of the work at several significant moments. We are thankful to the faculty of Computer Engineering, Bahria University, for providing us with knowledge and for the learning environment which enabled us to make this project a reality, and to the university for the facilities and resources that we used to develop and test the system. Many thanks to the volunteers who gave up their time in data collection and testing. Their involvement enabled the dataset and evaluation to be created. We also recognize the efforts of our team. We solved the issues that arose and achieved our objectives by working together and being responsible. Finally, we thank our parents and families for their patience, prayers, and encouragement during this degree. Their support kept us going. May Allah bless everyone who contributed to this work.

ABSTRACT

Monitoring student engagement during a lecture objectively is difficult. Instructors depend on visual cues that are easy to miss, self-report questionnaires that interrupt the session, and post-hoc performance data that arrives too late to act on. This thesis addresses that gap with an EEG-based engagement monitoring system built for virtual reality learning environments and evaluated under real cross-subject deployment conditions.

EEG data were collected from 16 participants using an OpenBCI Cyton board (8 channels, 250 Hz) while they watched academic lectures through a Meta Quest 2 headset. Each participant completed six sessions across six undergraduate subjects at three difficulty levels, giving four cognitive conditions per session. Recordings were preprocessed with a 50 Hz notch filter, a 1–40 Hz band-pass filter, artifact rejection, and 2-second Hamming windows with 50% overlap. Pipeline A assigned four-class cognitive load labels, producing 80,640 windows. Pipeline B computed 24-dimensional Welch PSD feature vectors and assigned binary engaged/disengaged labels via the Pope Engagement Index.

Three novel deep learning architectures were evaluated under a five-fold stratified subject-independent cross-validation protocol. EEG-HybridFormer, combining a multi-scale convolutional front-end with a Transformer encoder and DINO self-supervised training, achieved the highest mean accuracy of 85.64%. EEG-Deformer++ reached 78.45% and EEG-DINO++ established a 50.66% unsupervised baseline. An ablation study confirmed that subject-independent evaluation reduces accuracy by 12–15 percentage points relative to a random split.

A group disengagement detection module flags timestamps where six or more participants are disengaged for six or more consecutive seconds. Hard sessions consistently produced more disengagement zones, peaking at 13 of 16 students in the most difficult Digital Logic Design session. A real-time inference pipeline using BrainFlow, a FastAPI WebSocket server, and a React dashboard demonstrates live deployment. The results show that subject-independent EEG engagement classification is feasible from a low-channel consumer-grade device, and that content difficulty reliably predicts group-level disengagement in a VR lecture setting.

Keywords: Electroencephalography, student engagement, virtual reality, deep learning, Transformer, cross-subject classification, cognitive load, real-time monitoring.

TABLE OF CONTENTS

UNDERTAKING i

DEDICATION ii

ACKNOWLEDGEMENT iii

ABSTRACT iv

LIST OF FIGURES x

LIST OF TABLES xix

LIST OF EQUATIONS xxi

1 Introduction 1

1.1 EEG-Based Engagement Assessment 3

1.2 Virtual Reality for Educational Delivery 5

1.3 Brain Computer Interface (BCI) 6

1.4 Existing Solutions 7

1.5 Objectives 9

1.6 Thesis Organization 10

2 Literature Review 11

2.1 Introduction 11

2.2 EEG and the Frequency Bands Behind Engagement 11

2.2.1 Where This Started 11

2.2.2 What Each Band Actually Means 12

2.3 EEG Engagement Work in Learning Settings 12

2.4 Traditional Machine Learning on EEG 13

2.4.1 The SVM and Feature Engineering Era 13

2.4.2 Why PSD Features Rather Than Raw EEG 14

2.5 Deep Learning for EEG 14

2.5.1 Convolutional Networks 14

2.5.2 Recurrent Networks and LSTMs 15

2.5.3 Where CNNs and RNNs Fall Short 15

2.6 Transformer Architectures for EEG 16

2.6.1 The Core Idea 16

2.6.2 Transformers Built Specifically for EEG 16

2.7 Self-Supervised Learning for EEG 17

2.7.1 Why Bother With Self-Supervision Here . . . 17
2.7.2 DINO and What It Adds . . . 17
2.7.3 What This Means for EEG-DINO++ and EEG-HybridFormer . 18
2.8 VR in Learning and Neuroscience Research . . . 19
2.8.1 VR as a Way to Control the Environment . . . 19
2.8.2 EEG Combined With VR . . . 19
2.9 Non-EEG Approaches to Engagement Detection . . . 20
2.9.1 Facial Expression and Camera-Based Methods . . . 20
2.9.2 Eye Tracking . . . 20
2.9.3 Combining Multiple Signals . . . 20
2.10 Comparison of Prior Work . . . 21
2.11 What This Project Adds . . . 22
3 Methodology 24
3.1 System Overview . . . 24
3.2 Hardware . . . 26
3.2.1 OpenBCI Cyton Board and Settings . . . 26
3.2.2 Electrode Placement . . . 28
3.2.3 VR Headset . . . 30
3.3 Participants . . . 31
3.4 Experimental Protocol . . . 32
3.4.1 Subjects and Lecture Videos . . . 32
3.4.2 Session Structure . . . 33
3.5 EEG Signal Acquisition . . . 35
3.6 VR Environment . . . 36
3.7 Data Acquisition . . . 37
3.8 Signal Preprocessing . . . 39
3.9 Pipeline A: Raw EEG Processing for Four-Level Classification . . . 42
3.9.1 Step 1: Scaling to Microvolts . . . 42
3.9.2 Step 2: Bandpass Filtering . . . 42
3.9.3 Step 3: Notch Filter at 50 Hz . . . 43
3.9.4 Step 4: Windowing and Artifact Removal . . . 43
3.9.5 Step 5: Assembling the Raw Window . . . 43
3.9.6 Step 6: Per-Channel Normalization . . . 43
3.10 Pipeline A: Four-Level Labeling . . . 44
3.10.1 Assigning the Class Label . . . 44
3.10.2 Limitations of Condition-Based Labels . . . 44

3.11 Pipeline B: Engagement Signal Processing . . . 44
3.11.1 Step 5: Welch PSD Feature Extraction . . . 45
3.11.2 Step 6: Feature Normalization . . . 45
3.12 Pipeline B: Engagement Labeling . . . 46
3.12.1 The Pope Engagement Index . . . 46
3.12.2 Per-Participant Median Thresholding . . . 47
3.13 Cross-Validation Setup . . . 47
3.14 Proposed Model Architectures . . . 49
3.14.1 EEG-Deformer++ . . . 49
3.14.2 EEG-DINO++ . . . 54
3.14.3 EEG-HybridFormer . . . 58
3.15 Group Disengagement Detection . . . 61
3.16 Real-Time System . . . 62
4 Implementation 66
4.1 Overview . . . 66
4.2 Development Environment . . . 66
4.3 Hardware Setup and Configuration . . . 68
4.3.1 OpenBCI Cyton Board and GUI Configuration . . . 68
4.3.2 Electrode Placement and Impedance Check . . . 70
4.3.3 Meta Quest 2 VR Headset and Lecture Delivery . . . 71
4.4 EEG Acquisition and Timestamp Alignment . . . 72
4.4.1 Recording Procedure . . . 72
4.4.2 Segment Label Assignment . . . 73
4.4.3 Real-Time Acquisition via BrainFlow . . . 74
4.5 Offline Preprocessing Implementation . . . 74
4.6 Notch Filter . . . 75
4.7 Bandpass Filter . . . 75
4.8 Amplitude Artefact Rejection . . . 75
4.9 Windowing and Overlap . . . 76
4.10 Pipeline A Implementation . . . 76
4.11 Pipeline B Implementation . . . 77
4.12 Welch PSD Feature Extraction . . . 77
4.13 Engagement Label Computation . . . 78
4.14 Dataset Construction . . . 79
4.15 Model Training Implementation . . . 80
4.15.1 Data Loading and Cross-Validation Splits . . . 80

4.15.2 EEG-Deformer++ Training . . . 80
4.15.3 EEG-DINO++ Training . . . 81
4.15.4 EEG-HybridFormer Training . . . 81
4.16 Group Disengagement Detection Implementation . . . 82
4.17 Real-Time Inference System . . . 83
4.17.1 Architecture Overview . . . 83
4.17.2 FastAPI WebSocket Backend . . . 84
4.17.3 React Dashboard . . . 84
4.18 End-to-End Latency Analysis . . . 85
4.19 System Integration and Testing . . . 86
4.20 Summary . . . 88
5 Results and Discussion 90
5.1 Performance Evaluation Metrics . . . 90
5.2 Results of EEG-HybridFormer . . . 91
5.2.1 Hyperparameter Configuration . . . 91
5.2.2 Confusion Matrix Analysis . . . 91
5.2.3 Subject-wise Confusion Matrices . . . 94
5.2.4 Subject-wise Accuracy and Loss Curves . . . 102
5.2.5 Subject-wise Accuracy Curves . . . 102
5.2.6 Subject-wise Loss Curves . . . 108
5.3 Results of EEG-Deformer++ . . . 114
5.3.1 Hyperparameter Configuration . . . 114
5.3.2 Overall Confusion Matrix Analysis . . . 114
5.3.3 Subject-wise Performance . . . 116
5.4 Subject-wise Results of EEG-Deformer++ . . . 116
5.4.1 Subject-wise Confusion Matrices . . . 117
5.4.2 Subject-wise Accuracy Curves . . . 125
5.4.3 Subject-wise Loss Curves . . . 131
5.5 Results of EEG-DINO++ . . . 136
5.5.1 Hyperparameter Configuration . . . 136
5.5.2 Overall Confusion Matrix Analysis . . . 137
5.5.3 Subject-wise Performance . . . 138
5.6 Subject-wise Results of EEG-DINO++ . . . 139
5.6.1 Subject-wise Confusion Matrices . . . 139
5.6.2 Subject-wise Accuracy Curves . . . 147
5.6.3 Subject-wise Loss Curves . . . 153

5.7 Comparative Performance Analysis . . . 159
5.7.1 Subject-wise Performance Comparison . . . 159
5.7.2 Five Fold Cross Validation Comparison . . . 160
5.7.3 Overall Performance Comparison . . . 161
5.8 Discussion . . . 162
5.9 Group Disengagement Detection Results . . . 163
5.10 Real-Time System Evaluation . . . 169
5.11 Summary . . . 170
6 Conclusion 171
6.1 Conclusion . . . 171
6.2 Contributions of the Research . . . 172
6.3 Limitations . . . 173
7 Future Work 175
7.1 Overview . . . 175
7.2 Larger and more diverse participant pool . . . 175
7.3 Higher-density EEG and improved hardware . . . 176
7.4 Subject-adaptive calibration . . . 176
7.5 Self-supervised pretraining with larger EEG corpora . . . 177
7.6 Validated engagement ground truth . . . 177
7.7 Adaptive lecture delivery in VR . . . 178
7.8 Generalisation to live classroom settings . . . 178
7.9 Privacy-preserving and federated training . . . 179
7.10 Summary . . . 180
REFERENCES 181
LIST OF ABBREVIATIONS 185
ANNEXURES 187
ANNEXURE A . . . 187
ANNEXURE B . . . 192
ANNEXURE C . . . 195

LIST OF FIGURES

Number *Page*

Fig 3.1 End to end system block diagram showing the hardware, acquisition, preprocessing, classification, and output layers 25
Fig 3.2 OpenBCI Cyton board and electrodes with USB Bluetooth dongle and rechargeable battery pack 28
Fig 3.3 Electrode placement on the International 10-20 system.The eight active positions (Fp1, Fp2, F3, F4, C3, C4, P3, P4) are highlighted. Reference electrodes are on the left and right mastoids.Ground is at AFz. 29
Fig 3.4 Meta Quest 2 VR headset worn by a participant alongside the OpenBCI Cyton electrode cap during a recording session 31
Fig 3.5 Session structure showing the alternating relaxation and lecture segments. 34
Fig 3.6 Participants seated with EEG cap and VR headset during a recording session 34
Fig 3.7 EEG signal acquisition block diagram showing the Cyton board, Bluetooth link, BrainFlow BoardShim API, circular buffer, and CSV export path 36
Fig 3.8 view of the VR lecture environment as seen through the Meta Quest 2 headset, showing the lecture video displayed at comfortable viewing distance in a neutral virtual room 37
Fig 3.9 Data acquisition session flow showing setup, impedance check, recording sequence, and post-session quality check steps 38
Fig 3.10 Signal preprocessing pipeline. The four shared steps (scaling, bandpass filter, notch filter, and windowing with artifact rejection) feed both Pipeline A and Pipeline B. After windowing, the two pipelines diverge. 40
Fig 3.11 Pipeline A processing flow. Raw EEG goes through the four shared preprocessing steps, is kept in time-domain form as an 8x500 array, normalized per channel, and labeled by session segment timestamp. 42

Fig 3.12 Pipeline B processing flow. After the four shared preprocessing steps, each window is converted to a 24-number PSD feature vector (8 channels x 3 frequency bands: theta, alpha, beta) and labeled using the Pope Engagement Index with per-participant median thresholding. 45
Fig 3.13 Pope Index over time for one participant. Windows above the dashed threshold line are labeled engaged. 47
Fig 3.14 5-Fold subject-independent cross-validation. 16 participants are divided into 5 groups. In each fold, one group is held out for testing while the remaining participants train the model. No participant appears in both sets in the same fold. 48
Fig 3.15 EEG-Deformer++ architecture. Input flows left to right through the CNN encoder, MSTSF module, Adaptive Channel Attention block, 4-layer Transformer encoder, and gated classification head to produce the binary engagement prediction. 50
Fig 3.16 EEG-DINO++ teacher-student framework. The student network processes augmented views of each input window. The teacher network (updated by exponential moving average of student weights) processes a less-augmented view. The student learns to match the teacher's output distribution. 55
Fig 3.17 EEG-HybridFormer architecture. Both the student and teacher networks use the full EEG-Deformer++ backbone (CNN encoder + MSTSF + ACA + 4-layer Transformer). The student processes two augmented views and is trained with a three-part loss combining cross-entropy, DINO distillation, and MSTSF consistency terms. . . 59
Fig 3.18 Real-time system pipeline. BrainFlow streams EEG samples into a rolling buffer. Every 250 new samples, the buffer is preprocessed and passed to EEG-Deformer++ for inference. The prediction is sent via FastAPI WebSocket to the React dashboard and VR visualization. 63

Fig 4.1 OpenBCI GUI System Control Panel showing Cyton (Live) selected as the data source, with Serial (from Dongle) as the transfer protocol. 68
Fig 4.2 OpenBCI GUI serial port selection screen showing the USB dongle port highlighted in green under the Serial/COM Port panel. 69

Fig 4.3 OpenBCI GUI Hardware Settings panel showing channels 2–8 configured with x24 gain, SRB2 reference on, SRB1 off, and no onboard hardware filters applied. 70
Fig 4.4 OpenBCI impedance check panel. 73

Fig 5.1 Overall normalized confusion matrix for EEG-HybridFormer (aggregated across all subjects and folds) 92
Fig 5.2 Subject-wise accuracy bar chart for EEG-HybridFormer 93
Fig 5.3 Normalized confusion matrix for Student 1 (EEG-HybridFormer) . . 94
Fig 5.4 Normalized confusion matrix for Student 2 (EEG-HybridFormer) . . 95
Fig 5.5 Normalized confusion matrix for Student 3 (EEG-HybridFormer) . . 95
Fig 5.6 Normalized confusion matrix for Student 4 (EEG-HybridFormer) . . 96
Fig 5.7 Normalized confusion matrix for Student 5 (EEG-HybridFormer) . . 96
Fig 5.8 Normalized confusion matrix for Student 6 (EEG-HybridFormer) . . 97
Fig 5.9 Normalized confusion matrix for Student 7 (EEG-HybridFormer) . . 97
Fig 5.10 Normalized confusion matrix for Student 8 (EEG-HybridFormer) . . 98
Fig 5.11 Normalized confusion matrix for Student 9 (EEG-HybridFormer) . . 98
Fig 5.12 Normalized confusion matrix for Student 10 (EEG-HybridFormer) . 99
Fig 5.13 Normalized confusion matrix for Student 11 (EEG-HybridFormer) . 99
Fig 5.14 Normalized confusion matrix for Student 12 (EEG-HybridFormer) . 100
Fig 5.15 Normalized confusion matrix for Student 13 (EEG-HybridFormer) . 100
Fig 5.16 Normalized confusion matrix for Student 14 (EEG-HybridFormer) . 101
Fig 5.17 Normalized confusion matrix for Student 15 (EEG-HybridFormer) . 101
Fig 5.18 Normalized confusion matrix for Student 16 (EEG-HybridFormer) . 102
Fig 5.19 Training and validation accuracy curves for Student 1 (EEG-HybridFormer) 103
Fig 5.20 Training and validation accuracy curves for Student 2 (EEG-HybridFormer) 103
Fig 5.21 Training and validation accuracy curves for Student 3 (EEG-HybridFormer) 103
Fig 5.22 Training and validation accuracy curves for Student 4 (EEG-HybridFormer) 104
Fig 5.23 Training and validation accuracy curves for Student 5 (EEG-HybridFormer) 104
Fig 5.24 Training and validation accuracy curves for Student 6 (EEG-HybridFormer) 104

Fig 5.25 Training and validation accuracy curves for Student 7 (EEG-HybridFormer) . . . 105
Fig 5.26 Training and validation accuracy curves for Student 8 (EEG-HybridFormer) . . . 105
Fig 5.27 Training and validation accuracy curves for Student 9 (EEG-HybridFormer) . . . 105
Fig 5.28 Training and validation accuracy curves for Student 10 (EEG-HybridFormer) . . . 106
Fig 5.29 Training and validation accuracy curves for Student 11 (EEG-HybridFormer) . . . 106
Fig 5.30 Training and validation accuracy curves for Student 12 (EEG-HybridFormer) . . . 106
Fig 5.31 Training and validation accuracy curves for Student 13 (EEG-HybridFormer) . . . 107
Fig 5.32 Training and validation accuracy curves for Student 14 (EEG-HybridFormer) . . . 107
Fig 5.33 Training and validation accuracy curves for Student 15 (EEG-HybridFormer) . . . 107
Fig 5.34 Training and validation accuracy curves for Student 16 (EEG-HybridFormer) . . . 108
Fig 5.35 Training and validation loss curves for Student 1 (EEG-HybridFormer) . . . 108
Fig 5.36 Training and validation loss curves for Student 2 (EEG-HybridFormer) . . . 109
Fig 5.37 Training and validation loss curves for Student 3 (EEG-HybridFormer) . . . 109
Fig 5.38 Training and validation loss curves for Student 4 (EEG-HybridFormer) . . . 109
Fig 5.39 Training and validation loss curves for Student 5 (EEG-HybridFormer) . . . 110
Fig 5.40 Training and validation loss curves for Student 6 (EEG-HybridFormer) . . . 110
Fig 5.41 Training and validation loss curves for Student 7 (EEG-HybridFormer) . . . 110
Fig 5.42 Training and validation loss curves for Student 8 (EEG-HybridFormer) . . . 111

Fig 5.43 Training and validation loss curves for Student 9 (EEG-HybridFormer) . . . 111
Fig 5.44 Training and validation loss curves for Student 10 (EEG-HybridFormer) . . . 111
Fig 5.45 Training and validation loss curves for Student 11 (EEG-HybridFormer) . . . 112
Fig 5.46 Training and validation loss curves for Student 12 (EEG-HybridFormer) . . . 112
Fig 5.47 Training and validation loss curves for Student 13 (EEG-HybridFormer) . . . 112
Fig 5.48 Training and validation loss curves for Student 14 (EEG-HybridFormer) . . . 113
Fig 5.49 Training and validation loss curves for Student 15 (EEG-HybridFormer) . . . 113
Fig 5.50 Training and validation loss curves for Student 16 (EEG-HybridFormer) . . . 113
Fig 5.51 Overall normalized confusion matrix for EEG-Deformer++ (aggregated across all subjects and folds) . . . 115
Fig 5.52 Subject-wise accuracy bar chart for EEG-Deformer++ . . . 115
Fig 5.53 Normalized confusion matrix for Student 1 (EEG-Deformer++) . . . 117
Fig 5.54 Normalized confusion matrix for Student 2 (EEG-Deformer++) . . . 118
Fig 5.55 Normalized confusion matrix for Student 3 (EEG-Deformer++) . . . 118
Fig 5.56 Normalized confusion matrix for Student 4 (EEG-Deformer++) . . . 119
Fig 5.57 Normalized confusion matrix for Student 5 (EEG-Deformer++) . . . 119
Fig 5.58 Normalized confusion matrix for Student 6 (EEG-Deformer++) . . . 120
Fig 5.59 Normalized confusion matrix for Student 7 (EEG-Deformer++) . . . 120
Fig 5.60 Normalized confusion matrix for Student 8 (EEG-Deformer++) . . . 121
Fig 5.61 Normalized confusion matrix for Student 9 (EEG-Deformer++) . . . 121
Fig 5.62 Normalized confusion matrix for Student 10 (EEG-Deformer++) . . 122
Fig 5.63 Normalized confusion matrix for Student 11 (EEG-Deformer++) . . 122
Fig 5.64 Normalized confusion matrix for Student 12 (EEG-Deformer++) . . 123
Fig 5.65 Normalized confusion matrix for Student 13 (EEG-Deformer++) . . 123
Fig 5.66 Normalized confusion matrix for Student 14 (EEG-Deformer++) . . 124
Fig 5.67 Normalized confusion matrix for Student 15 (EEG-Deformer++) . . 124
Fig 5.68 Normalized confusion matrix for Student 16 (EEG-Deformer++) . . 125

Fig 5.69 Training and validation accuracy curves for Student 1 (EEG-Deformer++) . . . 125
Fig 5.70 Training and validation accuracy curves for Student 2 (EEG-Deformer++) . . . 126
Fig 5.71 Training and validation accuracy curves for Student 3 (EEG-Deformer++) . . . 126
Fig 5.72 Training and validation accuracy curves for Student 4 (EEG-Deformer++) . . . 126
Fig 5.73 Training and validation accuracy curves for Student 5 (EEG-Deformer++) . . . 127
Fig 5.74 Training and validation accuracy curves for Student 6 (EEG-Deformer++) . . . 127
Fig 5.75 Training and validation accuracy curves for Student 7 (EEG-Deformer++) . . . 127
Fig 5.76 Training and validation accuracy curves for Student 8 (EEG-Deformer++) . . . 128
Fig 5.77 Training and validation accuracy curves for Student 9 (EEG-Deformer++) . . . 128
Fig 5.78 Training and validation accuracy curves for Student 10 (EEG-Deformer++) . . . 128
Fig 5.79 Training and validation accuracy curves for Student 11 (EEG-Deformer++) . . . 129
Fig 5.80 Training and validation accuracy curves for Student 12 (EEG-Deformer++) . . . 129
Fig 5.81 Training and validation accuracy curves for Student 13 (EEG-Deformer++) . . . 129
Fig 5.82 Training and validation accuracy curves for Student 14 (EEG-Deformer++) . . . 130
Fig 5.83 Training and validation accuracy curves for Student 15 (EEG-Deformer++) . . . 130
Fig 5.84 Training and validation accuracy curves for Student 16 (EEG-Deformer++) . . . 130
Fig 5.85 Training and validation loss curves for Student 1 (EEG-Deformer++) 131
Fig 5.86 Training and validation loss curves for Student 2 (EEG-Deformer++) 131
Fig 5.87 Training and validation loss curves for Student 3 (EEG-Deformer++) 132
Fig 5.88 Training and validation loss curves for Student 4 (EEG-Deformer++) 132

Fig 5.89 Training and validation loss curves for Student 5 (EEG-Deformer++) 132
Fig 5.90 Training and validation loss curves for Student 6 (EEG-Deformer++) 133
Fig 5.91 Training and validation loss curves for Student 7 (EEG-Deformer++) 133
Fig 5.92 Training and validation loss curves for Student 8 (EEG-Deformer++) 133
Fig 5.93 Training and validation loss curves for Student 9 (EEG-Deformer++) 134
Fig 5.94 Training and validation loss curves for Student 10 (EEG-Deformer++) 134
Fig 5.95 Training and validation loss curves for Student 11 (EEG-Deformer++) 134
Fig 5.96 Training and validation loss curves for Student 12 (EEG-Deformer++) 135
Fig 5.97 Training and validation loss curves for Student 13 (EEG-Deformer++) 135
Fig 5.98 Training and validation loss curves for Student 14 (EEG-Deformer++) 135
Fig 5.99 Training and validation loss curves for Student 15 (EEG-Deformer++) 136
Fig 5.100 Training and validation loss curves for Student 16 (EEG-Deformer++) 136
Fig 5.101 Overall normalized confusion matrix for EEG-DINO++ (aggregated across all subjects and folds) 138
Fig 5.102 Subject-wise accuracy bar chart for EEG-DINO++ 138
Fig 5.103 Normalized confusion matrix for Student 1 (EEG-DINO++) 140
Fig 5.104 Normalized confusion matrix for Student 2 (EEG-DINO++) 140
Fig 5.105 Normalized confusion matrix for Student 3 (EEG-DINO++) 141
Fig 5.106 Normalized confusion matrix for Student 4 (EEG-DINO++) 141
Fig 5.107 Normalized confusion matrix for Student 5 (EEG-DINO++) 142
Fig 5.108 Normalized confusion matrix for Student 6 (EEG-DINO++) 142
Fig 5.109 Normalized confusion matrix for Student 7 (EEG-DINO++) 143
Fig 5.110 Normalized confusion matrix for Student 8 (EEG-DINO++) 143
Fig 5.111 Normalized confusion matrix for Student 9 (EEG-DINO++) 144
Fig 5.112 Normalized confusion matrix for Student 10 (EEG-DINO++) 144
Fig 5.113 Normalized confusion matrix for Student 11 (EEG-DINO++) 145
Fig 5.114 Normalized confusion matrix for Student 12 (EEG-DINO++) 145
Fig 5.115 Normalized confusion matrix for Student 13 (EEG-DINO++) 146
Fig 5.116 Normalized confusion matrix for Student 14 (EEG-DINO++) 146
Fig 5.117 Normalized confusion matrix for Student 15 (EEG-DINO++) 147
Fig 5.118 Normalized confusion matrix for Student 16 (EEG-DINO++) 147
Fig 5.119 Training and validation accuracy curves for Student 1 (EEG-DINO++) 148
Fig 5.120 Training and validation accuracy curves for Student 2 (EEG-DINO++) 148
Fig 5.121 Training and validation accuracy curves for Student 3 (EEG-DINO++) 149
Fig 5.122 Training and validation accuracy curves for Student 4 (EEG-DINO++) 149
Fig 5.123 Training and validation accuracy curves for Student 5 (EEG-DINO++) 149

Fig 5.124Training and validation accuracy curves for Student 6 (EEG-DINO++) 150
Fig 5.125Training and validation accuracy curves for Student 7 (EEG-DINO++) 150
Fig 5.126Training and validation accuracy curves for Student 8 (EEG-DINO++) 150
Fig 5.127Training and validation accuracy curves for Student 9 (EEG-DINO++) 151
Fig 5.128Training and validation accuracy curves for Student 10 (EEG-DINO++) . . . 151
Fig 5.129Training and validation accuracy curves for Student 11 (EEG-DINO++) . . . 151
Fig 5.130Training and validation accuracy curves for Student 12 (EEG-DINO++) . . . 152
Fig 5.131Training and validation accuracy curves for Student 13 (EEG-DINO++) . . . 152
Fig 5.132Training and validation accuracy curves for Student 14 (EEG-DINO++) . . . 152
Fig 5.133Training and validation accuracy curves for Student 15 (EEG-DINO++) . . . 153
Fig 5.134Training and validation accuracy curves for Student 16 (EEG-DINO++) . . . 153
Fig 5.135Training and validation loss curves for Student 1 (EEG-DINO++) . . 154
Fig 5.136Training and validation loss curves for Student 2 (EEG-DINO++) . . 154
Fig 5.137Training and validation loss curves for Student 3 (EEG-DINO++) . . 154
Fig 5.138Training and validation loss curves for Student 4 (EEG-DINO++) . . 155
Fig 5.139Training and validation loss curves for Student 5 (EEG-DINO++) . . 155
Fig 5.140Training and validation loss curves for Student 6 (EEG-DINO++) . . 155
Fig 5.141Training and validation loss curves for Student 7 (EEG-DINO++) . . 156
Fig 5.142Training and validation loss curves for Student 8 (EEG-DINO++) . . 156
Fig 5.143Training and validation loss curves for Student 9 (EEG-DINO++) . . 156
Fig 5.144Training and validation loss curves for Student 10 (EEG-DINO++) . 157
Fig 5.145Training and validation loss curves for Student 11 (EEG-DINO++) . 157
Fig 5.146Training and validation loss curves for Student 12 (EEG-DINO++) . 157
Fig 5.147Training and validation loss curves for Student 13 (EEG-DINO++) . 158
Fig 5.148Training and validation loss curves for Student 14 (EEG-DINO++) . 158
Fig 5.149Training and validation loss curves for Student 15 (EEG-DINO++) . 158
Fig 5.150Training and validation loss curves for Student 16 (EEG-DINO++) . 159
Fig 5.151Grouped bar chart comparing subject-wise accuracy for EEG-HybridFormer, EEG-Deformer++, and EEG-DINO++ 160

Fig 5.152Per-fold mean accuracy for EEG-HybridFormer, EEG-Deformer++, and EEG-DINO++ across five folds 161

LIST OF TABLES

Number *Page*

Table 1.1 Five Classifications of Frequency Waves of EEG Signal 3
Table 1.1 Five Classifications of Frequency Waves of EEG Signal 4

Table 2.1 Summary of Prior EEG Engagement and Classification Work . . 21
Table 2.1 Summary of Prior EEG Engagement and Classification Work . . 22

Table 3.1 OpenBCI Cyton board settings used in all recording sessions . . . 27
Table 3.2 Participant details 32
Table 3.3 Data collection summary 35
Table 3.4 Signal preprocessing parameters (shared by Pipeline A and Pipeline B) 41
Table 3.5 Cross-validation setup 48
Table 3.5 Cross-validation setup 49
Table 3.6 EEG-Deformer++ layer-by-layer summary 53
Table 3.7 Summary of the three proposed models 61
Table 3.8 Real-time system components 64

Table 4.1 Software components used in the implementation 67
Table 4.2 Preprocessing parameters applied to both pipelines 76
Table 4.3 Welch PSD computation parameters 78
Table 4.3 Welch PSD computation parameters 79
Table 4.4 Final dataset statistics 79
Table 4.5 Training hyperparameters for all three models 82
Table 4.6 Group disengagement detection thresholds and their rationale . . 83
Table 4.7 End-to-end latency breakdown measured over 200 windows . . . 85
Table 4.7 End-to-end latency breakdown measured over 200 windows . . . 86
Table 4.8 Module test results 88

Table 5.1 Hyperparameter configuration for EEG-HybridFormer 91
Table 5.2 Subject-wise performance of EEG-HybridFormer 93
Table 5.3 Hyperparameter configuration for EEG-Deformer++ 114
Table 5.4 Subject-wise performance of EEG-Deformer++ 116
Table 5.5 Hyperparameter configuration for EEG-DINO++ 137

Table 5.6 Subject-wise performance of EEG-DINO++ 139
Table 5.7 Subject wise accuracy comparison for all three models 159
Table 5.7 Subject wise accuracy comparison for all three models 160
Table 5.8 Per fold mean accuracy comparison for all three models 160
Table 5.8 Per fold mean accuracy comparison for all three models 161
Table 5.9 Overall mean performance comparison for all three models . . . 161
Table 5.9 Overall mean performance comparison for all three models . . . 162
Table 5.10 Group disengagement zones — Digital Logic Design (DLD) . . . 164
Table 5.11 Group disengagement zones — Object-Oriented Programming (OOP) 165
Table 5.12 Group disengagement zones — Computer Fundamentals 166
Table 5.13 Group disengagement zones — Engineering Ethics 167
Table 5.14 Group disengagement zones — Entrepreneurship 167
Table 5.14 Group disengagement zones — Entrepreneurship 168
Table 5.15 Group disengagement zones — Communication Skills 168
Table 5.15 Group disengagement zones — Communication Skills 169

Table 7.1 OpenBCI Cyton board technical specifications 192
Table 7.1 OpenBCI Cyton board technical specifications 193
Table 7.2 Meta Quest 2 VR headset technical specifications 196

LIST OF EQUATIONS

Number *Page*

Eq 1.1 Pope Engagement Index 12

Eq 3.1 ADC Voltage Conversion 43

Eq 3.2 Pope Engagement Index (Pipeline B) 58

Eq 3.3 Multi-Head Self-Attention 71

Eq 3.4 EMA Weight Update 80

Eq 3.5 DINO Combined Loss 85

Eq 3.6 HybridFormer Combined Loss 91

Chapter 1

Introduction

The shift to recorded online lectures over the past decade has made university education accessible to a far larger population than a decade ago. Coursera, edX, and institutional learning management systems collectively enroll tens of millions of students across different countries. This growth accelerated sharply during the COVID-19 pandemic, which pushed institutions to move from in-person delivery to online video content almost overnight [1]. The content is more accessible than it has ever been. The feedback loop, however, broke down almost completely.

In a physical classroom, a lecturer reads the room continuously. When students go quiet in the wrong way, the pace slows. When confusion appears on faces, the explanation changes. This adjustment happens naturally throughout the session. None of it happens with a pre-recorded video. The instructor records, uploads, and receives nothing back. The student presses play, and from that point the instructor has no information about what happens.

The common workarounds for this problem are weak. Quiz scores indicate whether a student understood the material after the fact, not whether they were paying attention during the lecture. Completion rates confirm that someone pressed play, not that they stayed engaged. Time-on-page statistics can be inflated by pausing a video and walking away. All of these measures are delayed, coarse, and easy to game without learning anything.

The scale of the problem is not small. Engagement research in online education consistently reports that attention drops off sharply after the first ten minutes of a lecture, regardless of the student's motivation or ability. Some studies put average engagement times as low as six to eight minutes before a meaningful cognitive drift begins. A typical university lecture runs forty-five to ninety minutes. The gap between what a student experiences and what the instructor assumes they experience can be substantial. For passive video content, where the viewer is not required to do anything except watch, the problem is even more pronounced. There is no interaction, no natural breakpoint, and no feedback mechanism that tells either party where the session went wrong.

The deeper issue is that the tools available to measure engagement online were not designed for continuous monitoring. They were designed to measure outcomes, not process. An instructor looking at quiz data from a recorded lecture can see that students struggled with a specific concept, but cannot tell whether students missed it because

the explanation was poor, because they lost focus during that particular segment, or because they were never paying attention at any point. These are different problems with different solutions, and outcome data alone cannot separate them.

A continuous, real-time signal of cognitive engagement would solve this at the source. If an instructor could examine a time-stamped map of where students disengaged, lecture by lecture, the feedback loop would not be broken any longer. Segments that consistently produce disengagement could be identified and revised. Segments that hold attention across different difficulty levels could be studied and replicated. This kind of information could improve the quality of recorded lectures in a way that questionnaires and completion rates simply cannot.

EEG does exactly that. It records brain electrical activity through scalp electrodes at millisecond-level resolution, and the signal shifts continuously without any action from the participant. Specific frequency patterns in the EEG signal shift in consistent ways when a person moves between cognitive engagement and inattention. These shifts occur regardless of whether the person looks attentive from the outside. A student can appear to watch a lecture while mentally elsewhere, and the EEG signal will show the difference [2].

This thesis describes an EEG-Based VR Smart Learning System that measures student engagement from brain signals during recorded lectures delivered through a Virtual Reality (VR) headset. Three deep learning architectures, EEG-Deformer++, EEG-DINO++, and EEG-HybridFormer, classify the incoming EEG signal as engaged or not engaged in two-second windows. The results feed a real-time dashboard and a group disengagement detection module that flags specific lecture timestamps where engagement dropped across multiple students simultaneously, giving instructors actionable information about where their recordings lose the audience.

Prior EEG engagement work has generally relied on training and testing models on data from the same participants. This inflates reported accuracy by 8 to 15 percentage points compared to testing on entirely new individuals [3]. This project evaluates all three proposed architectures under subject-independent cross-validation throughout, where no participant ever appears in both the training and test portions of any fold. The accuracy numbers that result from this evaluation reflect what the system would actually achieve on a student it had never seen before.

1.1 EEG-Based Engagement Assessment

Electroencephalography has been used to study attention and cognitive state since Hans Berger recorded the first human EEG in 1929 [4]. What began as a clinical tool became one of the most studied non-invasive methods for monitoring mental state in applied settings, because it provides a direct window into brain activity with millisecond-level temporal resolution that behavioral measures simply cannot match. EEG is the standard tool for real-time cognitive monitoring across neuroscience, BCI research, and human-computer interaction because it captures brain state directly and continuously, without requiring any input from the person being measured.The brain generates electrical signals that vary in frequency depending on the cognitive activity taking place. These signals fall into frequency bands, each associated with different mental states. Table 1.1 summarizes the five major EEG frequency bands along with their typical amplitude ranges, locations of origin, and associated functions. Delta oscillations, at 0.5 to 4 Hz, appear during deep sleep. Theta oscillations, at 4 to 8 Hz, increase during working memory tasks, memory encoding, and drowsiness. Alpha oscillations, at 8 to 13 Hz, suppress when a person actively focuses on something and rebound at rest, making them the most reliable single marker of attentional state. Beta oscillations, at 13 to 30 Hz, rise during active thinking and concentration. Gamma oscillations, above 30 Hz, are associated with higher-order cognitive processing.

Table 1.1: Five Classifications of Frequency Waves of EEG Signal [5]

Wave	**Amplitude (Voltage)**	**Frequency Band**	**Location of Origin**	**Occurrence / Function**
Delta (δ)	70–100 μV	0.5–4 Hz	Subcortical origin	Deep sleep; brain activity of infants; severe brain disorders
Theta (θ)	Around 50 μV	4–8 Hz	Thalamic and parietotemporal regions	Working memory, memory encoding, drowsiness
Alpha (α)	20–60 μV	8–13 Hz	Occipital region	Awake, relaxed state; suppresses during active attention

Table 1.1: Five Classifications of Frequency Waves of EEG Signal [5]

Wave	**Amplitude (Voltage)**	**Frequency Band**	**Location of Origin**	**Occurrence / Function**
Beta (β)	10–15 μV	13–30 Hz	Central frontal region	Active thinking, concentration, problem solving
Gamma (γ)	Around 10 μV	> 30 Hz	Distributed cortical regions	Higher-order cognitive processing, perception, information integration

The connection between these frequency bands and engagement was formalized by Pope, Bogart, and Bartolome in 1995, who defined an Engagement Index as beta power divided by the sum of alpha and theta power [6]:

$$EI = \frac{\beta}{\alpha + \theta} \tag{1.1}$$

where EI is the Engagement Index, β is the EEG beta band power (13–30 Hz), α is the alpha band power (8–13 Hz), and θ is the theta band power (4–8 Hz). This formula captures the signature of active cognitive engagement: when someone is engaged, beta rises and alpha falls, pushing the ratio up. When someone drifts mentally, beta stays flat or falls while alpha and theta rise, pulling the ratio down. This project uses the Pope Engagement Index to generate binary labels for each two-second EEG window, with the classification threshold set at each participant's own session median rather than a global population value, so the label reflects that person's relative engagement state.EEG electrodes are placed following the International 10-20 System, a standardized scheme that ensures consistency across studies and participants [7]. In this system, electrode labels reference specific brain regions. In this labeling scheme, F marks the frontal lobe, C the central strip, P the parietal lobe, T the temporal lobe, and O the occipital lobe. Odd numbers indicate left hemisphere placement; even numbers indicate right. The frontal lobe is particularly relevant here because it governs executive function and working memory, two processes directly involved in active learning. The parietal lobe processes sensory integration and spatial attention. For engagement monitoring during lecture viewing, frontal and parietal positions are the most relevant because the dorsal attention network, which directs sustained goal-directed attention, runs through these

regions [8].The OpenBCI Cyton board used in this project is an 8-channel EEG acquisition device that costs a fraction of clinical EEG systems and works with open-source software. It records from frontal, central, and parietal electrode positions at 250 Hz sampling rate. While 8 channels are not enough to run Independent Component Analysis reliably, they are sufficient for frequency-band power analysis across the cortical regions that matter most for engagement monitoring. This hardware makes it practical to collect EEG data from real students in a university setting rather than in a clinical laboratory.Raw EEG recordings are sensitive to artifacts from eye blinks, muscle activity, body movements, and environmental interference. A preprocessing pipeline that includes band-pass filtering, artifact rejection by amplitude threshold, segmentation into two-second windows, and conversion to Power Spectral Density (PSD) features using Welch's method [9] addresses these issues. Converting raw time series to PSD features discards individual waveform characteristics that cause cross-participant data leakage and retains only frequency-band energy, which encodes cognitive state in a form that is more consistent across people.

1.2 Virtual Reality for Educational Delivery

Virtual Reality has become a useful research platform for studying attention and cognitive load precisely because it controls the participant's visual environment without taking them out of a naturalistic viewing context. When a lecture is delivered through a VR headset, visual distractions in the room are blocked completely. The participant sits inside a consistent visual environment for the entire session. This makes EEG recording cleaner than in an ordinary room, because fewer unexpected visual events trigger artifacts or attention shifts unrelated to the lecture [10].This project used the Meta Quest 2 VR headset to deliver pre-recorded lecture videos to participants. Six academic subjects from the Computer Engineering curriculum at Bahria University Islamabad were used. Each subject had three difficulty levels, easy, medium, and hard, with each lecture delivered in sequence and separated by relaxation periods. The VR environment kept the visual context constant across all sessions and all participants, removing one source of variability that would otherwise make cross-participant comparisons harder to interpret.Previous research has shown that task complexity, visual information density, and the cognitive demands of a learning activity produce different levels of engagement in VR environments [9]. VR therefore allows researchers to manipulate difficulty levels in a controlled and reproducible way. In this project, each difficulty level was designed to produce a reliably different EEG engagement signature, giving the deep learning

models distinct targets to learn. The four-condition structure, relaxation, easy, medium, and hard, also produces a natural difficulty gradient in the data rather than a binary distinction.

Beyond the methodological advantages, VR represents a realistic delivery method for online education. The same setup could be used to deliver live or recorded lectures to remote students while monitoring their engagement through a connected EEG device. The ecological validity of this approach is higher than studies using traditional computer screens for stimulus delivery, because VR creates an immersive environment closer to what a dedicated learning space provides.VR headsets like the Meta Quest 2 are standalone devices that do not require external computing equipment. They offer high-resolution graphics, motion tracking, and wireless interaction, which allow participants to perform tasks comfortably without physical constraints. Recent advances in consumer VR hardware have made it increasingly practical to combine VR with physiological signal acquisition systems, creating synchronized recordings of behavior and brain state that support richer analysis than either modality alone could provide.

1.3 Brain Computer Interface (BCI)

In a Brain Computer Interface, the computer reads brain signals directly rather than waiting for a keypress or physical gesture from the user [11]. BCI systems record brain activity, extract features from it, and classify those features into categories representing different mental states or intentions. These systems have been applied in motor imagery classification for assistive devices, attention monitoring in aviation and military settings, emotion recognition, neurorehabilitation, and clinical diagnosis.EEG-based BCIs are the most common non-invasive type. Electrodes on the scalp pick up brain electrical activity, signal processing extracts frequency or spatial features, and a classifier assigns those features to one of several mental state categories. EEG is the preferred modality for non-invasive BCI because it is safe, portable, relatively inexpensive, and has millisecond-level temporal resolution. The most widely used acquisition hardware ranges from research-grade 64-channel amplifiers down to consumer-grade devices like the OpenBCI Cyton board that can be operated outside a laboratory.The system proposed in this thesis is an EEG-based BCI for the educational domain. It reads EEG continuously while a student watches a lecture in VR, extracts frequency features every two seconds using Welch's PSD method, and classifies each window as engaged or not engaged. The classification output feeds a dashboard visible to instructors and a group disengagement detection module that accumulates results across

participants to identify lecture timestamps where multiple students disengage at once. This design combines the BCI's real-time classification capability with a group-level analysis that is not possible with single-subject monitoring alone.

With the development of deep learning, EEG-based BCI systems have improved in their ability to generalize across participants. Models that learn meaningful frequency-domain and spatial representations of EEG data can identify cognitive states without being calibrated individually for each new user. This cross-subject generalization is a practical requirement for deployment in any real-world setting where individual calibration time is not available. The three architectures proposed in this project, EEG-Deformer++, EEG-DINO++, and EEG-HybridFormer, are all designed and evaluated specifically for subject-independent performance.

1.4 Existing Solutions

EEG engagement and cognitive workload research has produced many practical methods over the past few decades. These approaches fall into three broad groups: subjective assessment, behavioral measures, and physiological signal analysis with machine learning. All three have genuine uses, but none of them works well as a standalone solution for continuous, real-time engagement monitoring during recorded lectures.Subjective assessment methods ask participants to report their own mental state, usually through structured questionnaires. The NASA Task Load Index (NASA-TLX) [12] and the Subjective Workload Assessment Technique (SWAT) [13] are the most widely used. They cost little and need no special equipment, but they ask participants to stop mid-session and self-report, which means they cannot monitor engagement continuously. They cannot monitor engagement continuously, and they cannot pinpoint specific moments within a lecture where attention dropped.Behavioral measures track observable outputs such as reaction time, mouse clicks, key presses, and eye movements. These correlate with cognitive state in controlled task settings but depend heavily on the task type and the individual's baseline. A student watching a passive lecture provides almost no behavioral signal to monitor. Error rates and completion times are informative only when there is a task with defined correct answers, which recorded lectures do not provide by themselves.Physiological signals, EEG in particular, address both of these limitations. EEG updates in real time without requiring any action from the participant, and its signal changes track cognitive state directly rather than through a behavioral proxy. The most studied EEG engagement approach combines alpha suppression with beta activation, formalized in the Pope Engagement Index. Systems built on this principle

have been applied in aviation monitoring, military operator tasks, and, more recently, educational settings. However, most published systems have a significant evaluation problem. The standard practice is to train and test on data from the same participants, which allows the model to learn individual EEG waveform characteristics that inflate accuracy. Under subject-independent evaluation, accuracy drops by 8 to 15 percentage points, which is the realistic performance range for real deployment [14].Standard EEG classification pipelines extract features manually, usually from the spectral or wavelet domain, and pass them to classifiers such as Support Vector Machines (SVM) or Random Forests (RF) [15]. These methods work reasonably well on small datasets and are simple to implement, but they depend on the quality of the handcrafted features and do not generalize well to new participants because inter-subject variability makes the optimal features differ between people [16].Deep learning approaches remove the handcrafting requirement by learning feature representations directly from EEG data. CNNs pick up spatial patterns across electrode channels, while LSTMs handle the temporal structure within a sequence. Transformers, originally built for NLP, use self-attention to relate patterns across distant channels or time points in a single pass, which convolution kernels cannot do. Hybrid CNN-Transformer models have consistently outperformed pure Transformers on EEG tasks, especially on smaller datasets, because the convolutional front-end provides a useful spatial inductive bias that reduces the amount of training data needed to learn meaningful representations [17].

Current deep learning models still have practical limitations when deployed on new participants. Models trained on one dataset frequently underperform when applied to a different group of people because each person's EEG has individual characteristics in amplitude, rhythm, and frequency distribution. The three architectures proposed in this project directly address this problem. EEG-Deformer++ adds a multi-scale temporal-spatial fusion module and adaptive channel attention to the EEG-Deformer baseline. EEG-DINO++ adapts the DINO self-supervised distillation framework for EEG [16], using EEG-specific augmentation to learn representations that do not depend on class labels and are therefore less sensitive to the noisy labels that result from the Pope Engagement Index threshold. EEG-HybridFormer combines the feature extraction from EEG-Deformer++ with the self-distillation training approach from EEG-DINO++ in a single unified architecture.

1.5 Objectives

The purpose of this project is to build a practical EEG-based engagement monitoring system that works on students it has never seen before, provides continuous real-time engagement feedback, and identifies specific lecture timestamps where group-wide disengagement occurs. The proposed system integrates the OpenBCI Cyton board with Meta Quest 2 VR lecture delivery and three novel deep learning architectures evaluated under subject-independent conditions.
The specific objectives of this research are:

1. Design and deploy a data collection protocol using the OpenBCI Cyton board (8-channel, 250 Hz) and Meta Quest 2 VR headset to record EEG from 16 student participants watching academic lectures across 6 subjects at three difficulty levels (easy, medium, hard).

2. Develop a subject-independent preprocessing pipeline that converts raw EEG recordings into 24-dimensional Welch PSD feature vectors, applying band-pass filtering and amplitude-based artifact rejection to remove participant-specific waveform characteristics while preserving frequency-band engagement information.

3. Generate per-participant binary engagement labels using the Pope Engagement Index thresholded at each individual's session median, producing 80,640 labelled two-second windows for model training and evaluation.

4. Propose and implement three novel deep learning architectures — EEG-Deformer++, EEG-DINO++, and EEG-HybridFormer — each designed to improve cross-subject generalisation for EEG-based engagement classification.

5. Evaluate all three architectures under strict 5-Fold Stratified Subject-Independent Cross-Validation, where no participant appears in both training and test sets, and compare results against published EEG classification baselines under equivalent evaluation conditions.

6. Build a group disengagement detection module that flags specific lecture timestamps where six or more participants are classified as disengaged for six or more consecutive seconds, providing instructors with actionable feedback on where recordings lose the audience.

7. Develop a real-time inference pipeline integrating BrainFlow EEG

streaming, a FastAPI WebSocket server, and a React instructor dashboard with live engagement visualisation, enabling deployment beyond the recorded-lecture research context.

8. Conduct an ablation study quantifying the individual contribution of each novel component in EEG-HybridFormer and measuring the accuracy gap between random-split and subject-independent evaluation on the same dataset.

1.6 Thesis Organization

The rest of this thesis covers six chapters.

Chapter 2 contains the literature review. This chapter covers prior work on EEG engagement detection starting from the basic neuroscience of frequency bands, through traditional machine learning approaches, modern deep learning architectures including EEGNet and CNN-LSTM, Transformer-based models built specifically for EEG, self-supervised learning methods adapted from computer vision, and the use of VR in educational research. The chapter ends with a comparison table of relevant studies and a clear statement of the gap this project addresses.

Chapter 3 describes the design and methodology in detail. This covers the hardware configuration, participant recruitment, electrode placement rationale, the experimental protocol with difficulty levels and session structure, the complete preprocessing pipeline from raw ADC counts to labeled PSD feature vectors, the three model architectures, and the real-time pipeline design.

Chapter 4 covers the implementation of all system components. This includes the data processing pipeline code, the training procedure for all three models, the group disengagement detection module, and the BrainFlow and FastAPI real-time streaming setup.

Chapter 5 presents the results. This includes overall and per-subject classification accuracy for each model, confusion matrices, learning curves, an ablation study on the components of EEG-Deformer++, a comparison with published results, and the group disengagement detection output for all six academic subjects.

Chapter 6 concludes the thesis with a summary of findings and a statement of the contributions. Chapter 7 describes six directions for future work.

Chapter 2

Literature Review

2.1 Introduction

The idea of measuring attention or mental workload from a physiological signal goes back further than most people expect. Researchers were studying pilot and operator focus with EEG in the 1980s, long before online learning existed as a field. What has changed recently is the hardware and the modeling tools. Consumer EEG boards and Transformer-based deep learning have made this kind of work possible outside a clinical lab, on a normal university budget, with ordinary students.This chapter covers five areas that are directly relevant to this project. It starts with the neuroscience of EEG and what specific frequency patterns have to do with engagement. It then covers traditional machine learning pipelines for EEG, modern deep learning approaches, Transformer and self-supervised architectures, and finally VR as a platform for learning research. A brief section also addresses non-EEG engagement detection methods, since those define the alternatives this project is measured against. Each section notes what the actual numbers were, not just the general idea, and points out where the gaps are. The chapter ends with a comparison table across all the studies reviewed and a clear statement of what this project adds.

2.2 EEG and the Frequency Bands Behind Engagement

2.2.1 Where This Started

Hans Berger recorded the first human EEG in 1929 and noticed that a roughly 10 Hz rhythm over the back of the head weakened when a person opened their eyes [2]. That single observation, that a specific frequency drops when the brain becomes more active, is the same basic principle every engagement detection system since has been built on.Pfurtscheller and Lopes da Silva formalized this in 1999 with the idea of event-related synchronization and desynchronization. Power in a frequency band goes up or down depending on what the underlying brain region is doing [3]. This gives a solid physiological reason to believe that theta, alpha, and beta power actually track something real, not just statistical noise that happens to correlate with a label.

The specific formula used in this project, the Pope Engagement Index, comes from a 1995 paper by Pope, Bogart, and Bartolome. They defined engagement as beta divided

by the sum of alpha and theta, and tested it on people monitoring automated systems [4]. Their motivation was practical: an operator overseeing an automated system can look attentive while mentally checked out, and by the time a mistake happens it is too late. The same logic applies almost exactly to a student watching a recorded lecture.

2.2.2 What Each Band Actually Means

Theta power, in the 4 to 8 Hz range measured at the front of the head, rises during working memory tasks and memory encoding. It also rises when someone is drowsy or drifting mentally, which sounds contradictory but is not a problem here, because the Pope Index uses theta in the denominator along with alpha. When theta rises because someone is working hard mentally, beta tends to rise too, so the ratio stays reasonable. When theta rises because someone has checked out, beta usually does not rise, so the ratio drops.Alpha power, 8 to 13 Hz, is the most reliable single marker in this area. It drops when someone is actively paying attention and comes back up at rest [5]. Barry and colleagues showed that simply opening the eyes drops posterior alpha by 30 to 50 percent compared to eyes closed [6]. That is a large effect for something as simple as an eye state change, which indicates how sensitive alpha is to attention shifts in general.Beta power, 13 to 30 Hz, rises during active thinking and concentration. Miller and Cohen linked frontal beta to the kind of top-down attentional control that lets someone stay focused on a task instead of drifting [7]. Rising beta together with falling alpha is the signature the Pope Index is built to detect. The reason this project records from frontal, central, and parietal positions specifically comes from Corbetta and Shulman's work on the dorsal attention network, which sits mostly in frontal and parietal cortex and handles the job of directing attention toward a goal [8].

2.3 EEG Engagement Work in Learning Settings

Kosmyna and colleagues built AttentivU in 2019, an EEG biofeedback headband that vibrates when engagement drops during reading tasks. They reported roughly 80 percent accuracy classifying engagement state [10]. The limitation is that the model was trained and tested on the same participants, which is a considerably easier problem than testing on someone new.Apicella and colleagues built a wearable EEG system specifically for classroom engagement monitoring in 2022 [9]. Their system was personalised per participant, meaning each new user had to record calibration data before the system worked for them. That is a real practical barrier if the goal is a system that works on a student the first time they put on the headset.A study published in npj Science

of Learning in 2025 tracked EEG across an 11-lesson simulated online course [18]. They found frontal theta rising during quiz sections and parietal alpha dropping during lecture sections, which lines up with the general engagement story, and reported 83 percent accuracy classifying which type of learning activity was happening. Only 6 of their original 20 participants had clean enough data to use, which is a small number for drawing strong conclusions.Sarailoo and colleagues ran the most useful comparison for this project. They trained classifiers on spectral EEG features from people doing a workload task at three difficulty levels and reported both a random split result and a subject-independent result on the same data [11]. The random split gave 84.5 percent. Subject-independent evaluation, where the same participant never appears in both train and test sets, dropped to 76 percent. That 8.5 point gap from the exact same data and the exact same model is the clearest demonstration in the published literature of why the evaluation method matters as much as the model architecture.Yoo and colleagues trained a bidirectional LSTM with attention on EEG from students watching educational videos labelled by difficulty level and reported 87 percent accuracy [14]. Two things limit how much that number should carry. The labels came from task difficulty rather than an actual measurement of engagement, so a student who is bored during a hard video still gets labelled as engaged. And the split between train and test was random, not subject-independent, so some of that 87 percent is probably the same kind of inflation Sarailoo's comparison exposed.

2.4 Traditional Machine Learning on EEG

2.4.1 The SVM and Feature Engineering Era

Before deep learning became the default, the standard EEG classification pipeline was to filter the signal, cut it into short segments, compute band power or similar handcrafted features, and feed those features into a support vector machine or similar classifier. This approach works reasonably well and is still used today because it is simple, fast, and does not need a large dataset to train.Lotte and colleagues reviewed 218 papers on EEG classification for brain-computer interfaces published over a decade and noticed something worth flagging. Reported accuracy numbers were climbing year over year, but a lot of that improvement came from studies using easier evaluation setups rather than from models actually getting better at generalising to new people [16]. That observation is a large part of the reason subject-independent evaluation is treated as non-negotiable in this thesis.A 2025 systematic review in Frontiers in Neuroergonomics looked specifically at subject-independent accuracy across mental work-

load studies. Subject-independent models using segment lengths of 10, 20, and 30 seconds got over 80 percent in one MATB-based study [17]. But the review also found a random forest classifier that only reached 76 percent subject-independent accuracy on a similar task with the same kind of spectral features, and a CNN that dropped as low as 66.2 percent on a four-class version of the same problem with 50 participants [19]. The range across these studies, from 66 to over 80 percent, gives a realistic sense of what subject-independent EEG classification actually achieves once easy-mode random splits are removed.

2.4.2 Why PSD Features Rather Than Raw EEG

Power Spectral Density estimation using Welch's method has been the standard way to turn raw EEG into frequency features since 1967 [20]. Welch's approach splits a window into overlapping chunks, computes the spectrum of each chunk, and averages them, which gives a much less noisy estimate than computing one spectrum from the whole window at once.The choice between raw time series and PSD features is not just a technical detail. Raw EEG carries the specific waveform shape of each person's brain activity, almost like a fingerprint, on top of whatever task-related signal is actually present. A model trained on raw EEG from some participants and tested on others has to somehow ignore that fingerprint to generalise, and in practice it often does not fully manage that. Converting to PSD features throws away the exact waveform shape and keeps only the frequency-band energy, which means roughly the same thing across different people even though the absolute numbers differ from person to person. This is precisely why this project uses PSD features for all three proposed architectures.

2.5 Deep Learning for EEG

2.5.1 Convolutional Networks

Schirrmeister and colleagues showed in 2017 that convolutional networks could decode EEG directly from raw time series without any manual feature engineering, matching or beating the older CSP plus LDA pipeline on motor imagery tasks [21]. Their ShallowConvNet reached 85 percent and their deeper DeepConvNet reached 82 percent on a standard motor imagery benchmark. Both were tested within-subject.Lawhern and colleagues built EEGNet in 2018 to fix a real limitation of earlier CNNs, which tended to be designed for one specific EEG task and not transfer well to others [22]. EEGNet uses depthwise and separable convolutions borrowed from mobile computer vision architectures, and gets competitive results across motor imagery, P300, and several other

paradigms with only around 2,000 parameters. Under subject-independent testing it typically lands in the 70 to 80 percent range and is still one of the most cited baseline architectures in EEG classification work.A 2023 paper extended the CNN approach by converting EEG into 2D time-frequency images and running standard image CNNs on them [23]. This improved emotion classification accuracy on the DEAP and SEED datasets, though converting to a 2D image discards some of the actual electrode geometry information that a channel-aware model would use directly.Roy and colleagues reviewed 154 deep learning EEG papers in 2019 and found the same 8 to 15 percentage point gap between within-subject and cross-subject accuracy that appears consistently across this literature [24]. They also pointed out that many papers only report within-subject numbers, which makes it impossible to compare results across studies or estimate real deployment performance.

2.5.2 Recurrent Networks and LSTMs

LSTMs entered EEG classification because they can carry information across long sequences, something a fixed-size convolution kernel cannot do on its own. The long short-term memory design from Hochreiter and Schmidhuber [25] solves the vanishing gradient problem that plain recurrent networks run into, which in the EEG context means the network can, in principle, connect a pattern at second 2 to a pattern at second 30 of the same window.Beyond the Yoo et al. bidirectional LSTM study mentioned earlier, several papers compared LSTMs directly against SVMs and random forests for EEG emotion classification and found the LSTM consistently ahead across different feature setups. Other work combined a CNN front end for spatial pattern extraction with an LSTM back end for temporal modelling, which became a common recipe before Transformers took over as the dominant architecture.

2.5.3 Where CNNs and RNNs Fall Short

A convolution kernel only sees a fixed-size window of nearby positions. If two informative signals sit far apart in the 24-number feature vector, a small kernel cannot connect them directly, only indirectly through several stacked layers, which blurs the specific relationship. An LSTM processes one step at a time, which is slow to train and tends to lose track of information the further back it goes, even with the memory gating designed to prevent that. Both of these limitations are exactly what the attention mechanism in Transformer architectures was built to solve.

2.6 Transformer Architectures for EEG

2.6.1 The Core Idea

Vaswani and colleagues introduced the Transformer in 2017 with multi-head self-attention, which lets every position in a sequence look directly at every other position in one step instead of needing several layers to connect distant positions [26]. Applied to a 24-number EEG feature vector, this means the model can directly relate frontal theta to parietal beta in a single attention operation, something a small convolution kernel simply cannot do.Dosovitskiy and colleagues adapted the Transformer to images in 2021 by cutting an image into patches and treating each patch as a token — the Vision Transformer (ViT) approach [27]. This patch-based idea carried into EEG research, where researchers began treating short time segments or scalar feature values as tokens the same way ViT treats image patches. The EEG-DINO++ architecture proposed in this project follows exactly that logic.

2.6.2 Transformers Built Specifically for EEG

Sun and colleagues were among the first to apply Transformer attention directly to EEG classification in 2021, treating each electrode channel as a token in the sequence [28]. This let the attention mechanism learn which channels mattered most for a given task, and the learned attention weights tended to align with what was already known neurophysiologically about which brain regions matter for that task.Zeynali and colleagues built a spectral Transformer in 2023 that takes PSD features as input instead of raw EEG and reported 93.6 percent accuracy on their benchmark using spectral features alone, rising to 96.1 percent when combined with temporal features in an ensemble [29]. This is close in spirit to the approach used in this project. The main takeaway is that PSD input followed by Transformer attention clearly beats feeding the same Transformer raw time series.

Ding and colleagues published the EEG-Deformer in 2024, which is the architecture EEG-Deformer++ in this project directly builds on [30]. Their key finding was that a plain Vision Transformer applied to EEG does not work well, because projecting a single scalar value up to a high-dimensional embedding gives the attention mechanism almost nothing useful to work with. Adding a spatial convolution layer before the Transformer, so the model gets a richer starting representation, fixed most of that gap and got state-of-the-art results on standard BCI motor imagery benchmarks.Abibullaev and colleagues reviewed Transformer models across EEG applications including emo-

tion recognition, motor imagery, mental workload, and seizure detection, and found that hybrid CNN-Transformer models consistently outperformed pure Transformers, especially on smaller datasets [31]. Their explanation is that a pure Transformer has no built-in bias toward local patterns the way a convolution does, and without a large amount of training data it struggles to learn that bias from scratch. This is why EEG-Deformer++ keeps the convolutional front end rather than going pure Transformer.A follow-up review by Keutayeva and Abibullaev in 2024 focused specifically on data constraints and confirmed the same pattern: Transformer EEG models need either a lot of data or a strong convolutional inductive bias to compete with CNNs on cross-subject tasks [32]. The 80,640 windows collected in this project sit in a middle ground. Likely enough for a well-designed hybrid architecture like Deformer++ but probably not enough for a pure Transformer trained from scratch on raw EEG without the frequency-domain preprocessing used here.

2.7 Self-Supervised Learning for EEG

2.7.1 Why Bother With Self-Supervision Here

The real bottleneck in EEG research is not compute, it is labelled data. Getting a trustworthy engagement label for a single two-second window means either interrupting the participant to ask them, which changes the very state you are trying to measure, or relying on an indirect signal like the Pope Index, which is a proxy rather than ground truth. Self-supervised learning sidesteps some of this by learning useful representations from EEG data that has no labels at all, then fine-tuning on the smaller labelled set afterward.A 2025 systematic survey in ACM Computing Surveys grouped EEG self-supervised methods into two main families: contrastive learning, where the model learns by pulling together different views of the same signal and pushing apart different signals, and masked reconstruction, where the model learns by predicting parts of the signal that were hidden from it [33]. Both families have been shown to produce representations that transfer better to new participants than representations learned purely through supervised training on a small labelled set.

2.7.2 DINO and What It Adds

Caron and colleagues introduced DINO in 2021, short for self-distillation with no labels [34]. A student network and a teacher network are trained together. The student sees augmented versions of an input and tries to match what the teacher produces for

a cleaner version of the same input. The teacher is never trained directly by backpropagation; its weights slowly track the student's weights through an exponential moving average. This setup was originally built for images, but the underlying idea, learning a representation without needing class labels, is exactly what noisy EEG labels call for.This idea has already reached EEG specifically. A paper called EEG-DINO, presented at MICCAI 2025, built a foundation model for EEG using hierarchical self-distillation across multiple views of the signal, pretrained on more than 9,000 hours of EEG data [35]. The authors argue directly that reconstruction-based pretraining, which is common in other EEG foundation models, is not well suited to EEG because the signal-to-noise ratio is low enough that the model ends up learning to reconstruct noise rather than anything meaningful. Self-distillation avoids that trap because the student is only ever trying to match the teacher's representation, not reconstruct the raw signal itself.

Earlier contrastive approaches paved the way for this. BENDR, published in 2021 [36], adapted the wav2vec approach from speech processing to EEG, training a Transformer encoder to recognise which EEG segment came from which part of a recording using a contrastive objective. It showed that contrastive pretraining on EEG works and transfers across different downstream tasks, which was an important early proof of concept before the more recent self-distillation methods came along.

2.7.3 What This Means for EEG-DINO++ and EEG-HybridFormer

The image-based augmentations DINO normally uses, cropping, flipping, and colour changes, do not make sense for EEG. A cropped or flipped EEG signal is not a meaningful variation of the original in the way a cropped photo still shows the same object. This is the direct motivation for the four EEG-specific augmentations used in EEG-DINO++ in this project: shifting the window slightly in time, zeroing out a channel to simulate a bad electrode, masking out a frequency band, and scaling the amplitude to simulate natural variation in signal strength between sessions. Each of these represents a change that should not affect the true engagement label, which is exactly the property a good augmentation needs to have.EEG-HybridFormer takes this further by combining the feature extraction backbone from EEG-Deformer++ with the DINO self-distillation training scheme in a single unified architecture. Rather than treating the convolutional feature extractor and the self-supervised training objective as separate design choices, EEG-HybridFormer ties them together under a joint loss function, allowing the model to benefit simultaneously from the spatial inductive bias of convolution and the label-independent generalisation of self-distillation.

2.8 VR in Learning and Neuroscience Research

2.8.1 VR as a Way to Control the Environment

VR has been used in neuroscience since the mid-2000s, first mostly in clinical settings for pain distraction and phobia treatment, and more recently as a way to deliver a controlled stimulus for cognitive research [37]. Cattan and colleagues reviewed VR use in neurofeedback and brain-computer interface work and made a point that matters directly for this project: VR blocks out the physical environment around the participant, so any EEG change during a VR session is more likely to be a genuine response to the VR content rather than a reaction to something in the room [15].A 2024 review looked specifically at how VR affects student engagement in classroom-style settings and found consistently stronger engagement effects from VR delivery compared to a normal screen [38]. If VR genuinely produces a larger swing between engaged and disengaged states than watching the same content on a laptop, that is useful for this project too, because a larger swing should be easier for a classifier to detect reliably.

2.8.2 EEG Combined With VR

Kazemi and Lee looked at EEG-based emotional state classification across five different VR environments built to evoke specific emotions in 30 participants [39]. Using gradient boosted decision trees on PSD features from a 14-channel headset, their classifications lined up closely with what participants reported feeling themselves. This is a useful confirmation that the general approach, consumer EEG combined with PSD features and VR delivery, works for a related cognitive state classification problem, even though their target was emotion rather than engagement.A 2025 review of immersive VR classrooms examined combining EEG with other signals like heart rate and skin conductance to monitor students during VR lessons [40]. It concluded that EEG on its own gave the most direct and reliable read on cognitive state, while combining several signals improved robustness at the cost of considerably more equipment and complexity. For a study collecting 96 separate sessions across 16 participants, sticking to EEG alone rather than a full multimodal setup was the practical choice, and this review supports that as a reasonable tradeoff.

2.9 Non-EEG Approaches to Engagement Detection

2.9.1 Facial Expression and Camera-Based Methods

Facial expression analysis using CNNs, and more recently Vision Transformers, has been applied to classroom engagement detection without any wearable hardware at all, which is a genuine practical advantage. Reported accuracy for these systems is usually in the 75 to 85 percent range under controlled conditions.The problem is that facial expression is a behavioural signal, not a cognitive one. A student can hold an attentive expression while thinking about something completely unrelated, and a student can look distracted while actually processing something difficult. Studies that checked facial engagement classifiers against EEG-based ground truth generally found only moderate agreement in the range of 0.4 to 0.6 correlation, which suggests facial expression and cognitive engagement are related but clearly not the same thing. Camera-based methods also break down under ordinary conditions like poor lighting, a partially visible face, or someone not looking straight at the camera, all of which happen constantly in a normal online learning environment.

2.9.2 Eye Tracking

Eye tracking gives gaze position and pupil dilation, both of which relate to visual attention and cognitive load. Gaze-based engagement classifiers combining fixation patterns, saccades, and blink rate have reported 70 to 80 percent accuracy in lab settings. The same basic limitation as facial expression applies here. Gaze tells you where someone is looking, not whether they are actually processing what they see. A student can stare at a slide without absorbing anything on it. Eye tracking also needs either dedicated hardware or a carefully calibrated camera setup, which is not something most online learning platforms have built in.

2.9.3 Combining Multiple Signals

Some recent work combines EEG with other physiological signals like heart rate, skin conductance, or muscle activity, on the idea that no single signal captures the full picture. A 2025 system called IC-BiSGRU-Net combined BiLSTM and GRU networks over video, audio, and interaction log data from a classroom and showed that fusing these signals improved robustness over any single signal alone [41].The practical cost is significant. Running cameras, microphones, interaction logging, and physiological sensors simultaneously is not something most online learning platforms support, and

setting it up for a research study is substantially more work than recording EEG on its own. That is the main reason this project uses a single-modality EEG approach rather than going multimodal.

2.10 Comparison of Prior Work

Table 2.1: Summary of Prior EEG Engagement and Classification Work

Reference	**Year**	**Method**	**Participants / Data**	**Evaluation**	**Accuracy**	**Main Limitation**
Pope et al.	1995	EI = beta/(alpha+theta)	Operator monitoring task	Lab, within-subject	Index only	No classifier; original index definition only
Kosmyna et al.	2019	AttentivU — SVM on PSD	Reading task, 14-ch EEG	Within-subject	~80%	Cannot generalise to a new user
Schirrmeister et al.	2017	ShallowConvNet / DeepConvNet	BCI Competition IV-2A	Within-subject	82–85%	Within-subject only; raw EEG leakage risk
Lawhern et al.	2018	EEGNet compact CNN	Multi-paradigm BCI	Subject-independent	70–80%	Lower accuracy under honest evaluation
Roy et al.	2019	Systematic review, 154 papers	Literature review	Review	N/A	Confirms 8–15% within/cross gap
Apicella et al.	2022	Wearable EEG, personalised model	Consumer EEG, classroom	Within-subject	Not fixed	Needs per-user calibration
Sarailoo et al.	2022	SVM / random forest, spectral features	MATB-II, 3 difficulty levels	Both (random + subj-indep)	84.5% / 76%	Cleanest demonstration of leakage gap
Zeynali et al.	2023	Spectral Transformer	EEG benchmark	Controlled task	93.6–96.1%	Single controlled task, not engagement

Table 2.1: Summary of Prior EEG Engagement and Classification Work

Reference	Year	Method	Participants / Data	Evaluation	Accuracy	Main Limitation
Yoo et al.	2023	BiLSTM + attention	Educational videos, 2 levels	Random split	87%	Circular labels; random split inflates score
Ding et al.	2024	EEG-Deformer	BCI Competition IV-2A	Subject-independent	SoTA on bench-mark	Not applied to engagement or PSD input
Abibullaev et al.	2023	Transformer EEG review, 201 papers	Literature review	Review	N/A	Identifies the hybrid gap, does not close it
Frontiers Neuroer-gonomics review	2025	ML workload review across task types	Multiple MATB studies	Subject-independent	66–83%	Range shows real difficulty of cross-subject task
npj Science of Learning	2025	ML on PSD + PLI features	20 participants, 6 usable	Cross-task	83%	Small usable sample after quality checks
EEG-DINO (MIC-CAI)	2025	Hierarchical self-distillation	9,000+ hours EEG	Cross-task, founda-tion model	SoTA on multi-ple tasks	Not applied to lecture engagement
This project	2025	EEG-Deformer++, EEG-DINO++, EEG-HybridFormer	16 participants, 6 subjects, VR, 80,640 windows	5-Fold subject-independent	See Ch. 5	Consumer 8-channel EEG; single university

2.11 What This Project Adds

Looking at all of this together, five specific gaps stand out.

No published study combines subject-independent evaluation with PSD features specifically to prevent cross-participant leakage in an engagement detection task. Studies that use PSD features tend to use random splits, and studies that use subject-independent

splits tend to use raw time series instead.No published architecture adds a multi-scale convolution module together with adaptive channel attention to a Transformer built for engagement classification from frequency-domain EEG features.DINO self-distillation has been adapted for general EEG foundation models, but not for lecture engagement classification with augmentation strategies designed around EEG-specific signal perturbations.No prior work combines a Deformer-style feature extraction backbone with DINO-style self-distillation training in a single model for this task.

No system delivers group-level, timestamp-specific disengagement detection for VR-delivered academic lectures across multiple subjects and difficulty levels.This project addresses all five of these points. EEG-Deformer++, EEG-DINO++, and EEG-HybridFormer are all evaluated under strict subject-independent cross-validation on a dataset covering 16 participants, 6 university subjects, and 80,640 labelled EEG windows collected while lectures played through a Meta Quest 2 headset.

None of these five gaps is enormous on its own. What matters is testing all of them together under an evaluation method that is actually honest about cross-subject performance, since that is the condition closest to how a system like this would be used in practice.

Chapter 3

Methodology

3.1 System Overview

This chapter describes how the complete EEG-based VR smart learning system was built. It covers the hardware selection and configuration, participant recruitment, the experimental session design, the EEG acquisition pipeline, the VR environment setup, data collection procedures, signal preprocessing, two separate analysis pipelines, the deep learning model architectures in detail, the group disengagement detection module, and the real-time inference system.The system works as a single end-to-end chain. A participant wears an OpenBCI Cyton EEG headset and watches a pre-recorded lecture inside a Meta Quest 2 VR headset. The EEG signals are streamed from the Cyton board to a host computer through a Bluetooth dongle using the BrainFlow library. These signals are filtered, segmented into two-second windows, and converted into 24-number frequency-domain feature vectors. Each window receives a binary engagement label derived from the Pope Engagement Index, computed from the theta, alpha, and beta power of the signal. Three deep learning models are trained on these labeled feature vectors and tested on participants they have never seen, following a subject-independent cross-validation protocol. On top of per-participant predictions, a group disengagement detection module checks whether multiple students disengaged at the same lecture timestamp. A real-time pipeline streams live EEG into the trained model and displays predictions on a React dashboard and VR visualization.

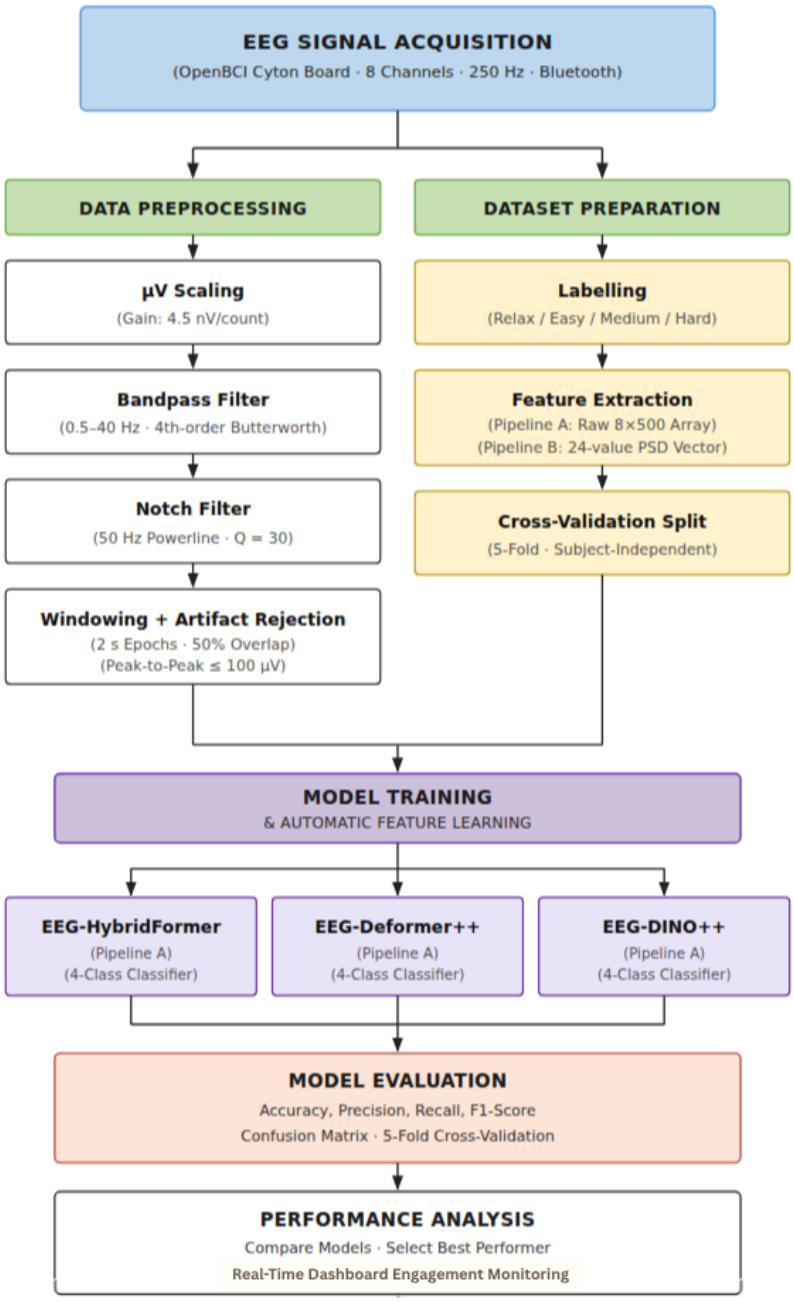


Fig. 3.1 End to end system block diagram showing the hardware, acquisition, preprocessing, classification, and output layers

Fig. 3.1 shows the overall system architecture as a single end to end block diagram. The hardware layer on the left contains the participant, the OpenBCI Cyton board fixed to their scalp through the electrode cap, and the Meta Quest 2 VR headset delivering the lecture. These two devices run independently during the session: the Cyton streams raw EEG to the host computer over Bluetooth while the VR headset plays through the

lecture sequence on its own internal timer.

The acquisition and preprocessing chain in the middle of the diagram receives the raw 8-channel, 250 Hz EEG stream from the Cyton board via the BrainFlow Python library. This stream passes through a band-pass filter, an artifact rejection stage based on amplitude thresholding, and a segmentation step that cuts the continuous signal into two-second, non-overlapping windows. Each window is then converted to a 24-dimensional Power Spectral Density feature vector using Welch's method, one value per frequency band per electrode channel. A separate timestamp log generated by the VR session controller is aligned to this feature stream during preprocessing to assign segment labels (relaxation, easy, medium, or hard) to every window.

The right side of the diagram shows the three parallel model training branches. EEG-Deformer++, EEG-DINO++, and EEG-HybridFormer each receive the same labeled feature vectors but process them through different architectures. All three are trained and evaluated under 5-fold stratified subject-independent cross-validation, meaning no participant's data ever appears in the same fold on both the training and test sides. Above the model branches, the group disengagement detection module aggregates per-window predictions across participants and flags lecture timestamps where a significant proportion of students are classified as disengaged concurrently. Below, the real-time inference pathway feeds live BrainFlow data through the trained model into a FastAPI WebSocket server, which delivers predictions to a React instructor dashboard and a VR visualization overlay.

3.2 Hardware

3.2.1 OpenBCI Cyton Board and Settings

The OpenBCI Cyton board records eight channels of EEG simultaneously at 250 Hz with 24 bit resolution. The analog front end is the Texas Instruments ADS1299, a chip built specifically for biopotential measurement that includes instrumentation amplifiers and programmable gain control. Data is transmitted wirelessly to the host PC through a USB Bluetooth dongle. The board runs on a small rechargeable battery clipped to the electrode cap, keeping the participant free of cables connected to mains power [2].Two main reasons drove the choice of the Cyton over available alternatives. It is fully open source, meaning any researcher can access the hardware design, firmware, and software libraries and repeat the experiment without licensing fees. The other common option, the Emotiv EPOC+, requires a monthly subscription and routes data through Emotiv's cloud servers. That arrangement raises data privacy concerns when recording from

student volunteers, and the ongoing cost would have been impractical for a 96-session collection. The Cyton also works directly with BrainFlow [3], a free Python library that handles streaming from over 30 different EEG boards, so the same preprocessing code worked for reading saved CSV files during training and for reading live data during real-time testing.

Table 3.1 summarises all hardware settings used across every recording session. The programmable gain was set to 24x on all eight channels, giving a voltage resolution of approximately 0.02235 microvolts per ADC count. This resolves the small changes in theta, alpha, and beta power that indicate engagement shifts. The SRB2 pin served as the reference connection for all eight channels, tied to linked mastoid reference electrodes placed behind the ears. The ground was connected to AFz on the scalp.

Table 3.1: OpenBCI Cyton board settings used in all recording sessions

Setting	**Value**	**Reason**
Sampling rate	250 Hz	Standard for EEG; gives 500 samples per 2-second window
Number of channels	8	Maximum for single Cyton board
ADC resolution	24-bit	Resolves signals as small as 0.02 microvolts
Programmable gain	24x on all channels	Amplifies the small EEG signal to a usable range
Onboard filters	Disabled	All filtering done in software for full control
Reference	Linked mastoid (A1+A2), SRB2 pin	Standard neutral reference for EEG recording
Ground	AFz electrode on scalp	Common EEG practice to reduce common-mode noise
Transmission	Bluetooth to USB dongle	Wireless connection removes cable movement artifacts
Data format	CSV via BrainFlow stream	Same format for offline recording and real-time use

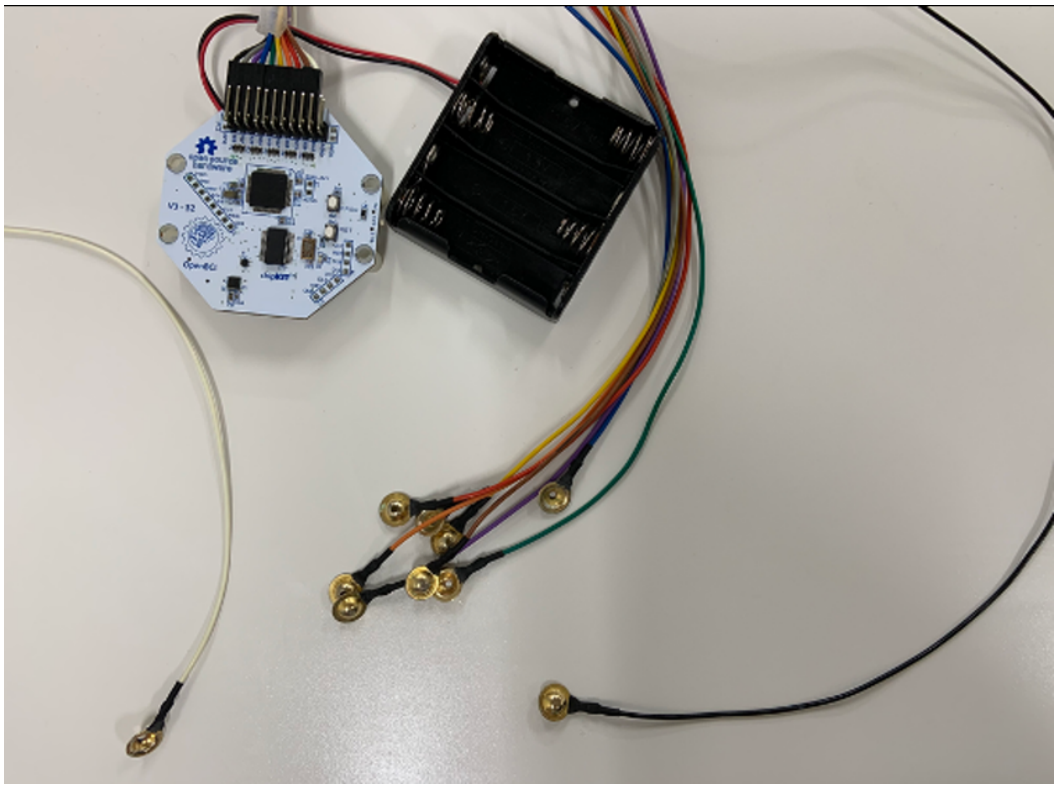


Fig. 3.2 OpenBCI Cyton board and electrodes with USB Bluetooth dongle and rechargeable battery pack

3.2.2 Electrode Placement

Eight electrodes were placed following the International 10-20 system [4].The positions which were chosen was Fp1, Fp2, F3, F4, C3, C4, P3, and P4. Reference electrodes were placed on the left and right mastoid bones behind the ears.The ground was at AFz on the scalp.Fig. 3.3 shows the electrode positions on the 10-20 system map.

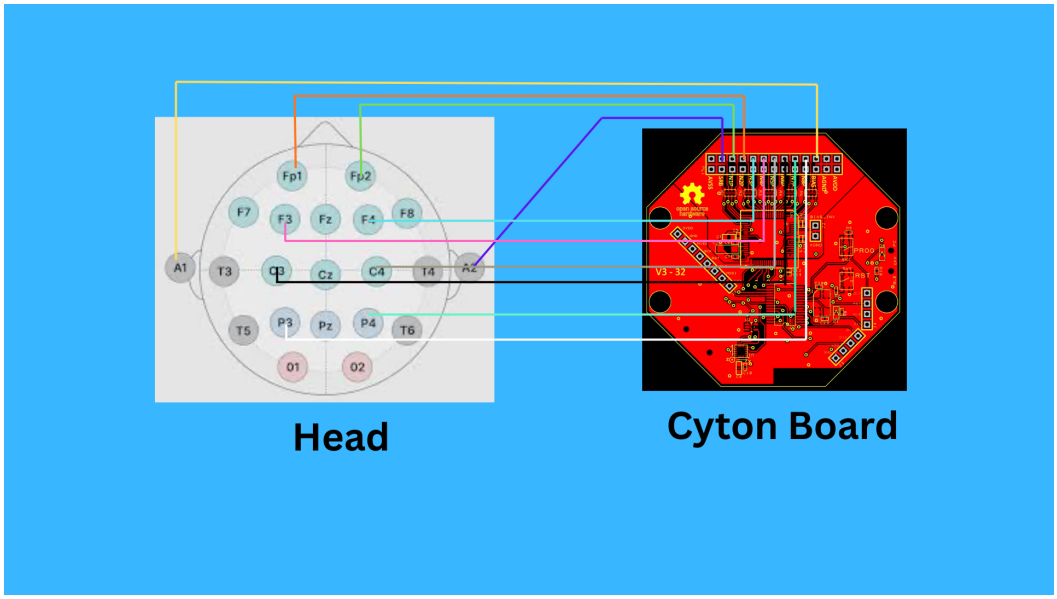


Fig. 3.3 Electrode placement on the International 10-20 system.The eight active positions (Fp1, Fp2, F3, F4, C3, C4, P3, P4) are highlighted. Reference electrodes are on the left and right mastoids.Ground is at AFz.

These positions were not chosen arbitrarily.Each covers a part of the brain that shows measurable changes during cognitive engagement. Fp1 and Fp2 sit just above the forehead over the prefrontal cortex, which is responsible for attention, working memory, and sustained focus [5]. When a student is genuinely engaged with a lecture, prefrontal activity rises and beta power at these electrodes tends to increase. When attention drifts, alpha power rises instead, producing the alpha-dominance signature associated with disengagement.F3 and F4 cover the lateral frontal lobe, a region involved in executive function and top-down direction of attention [6]. These electrodes capture how the brain monitors and regulates its own focus. When someone is actively working to stay engaged, F3 and F4 tend to show elevated beta. When they are drifting, this signal drops. These two positions provide attention control signals that are distinct from what Fp1 and Fp2 capture.C3 and C4 are central electrodes over the sensorimotor cortex. They are included because the sensorimotor rhythm, a frequency pattern around 12 to 15 Hz in the lower beta range, changes when a person moves between active cognitive processing and a passive resting state [7]. This rhythm is well established in EEG research and adds information about mental state beyond what the frontal electrodes provide.P3 and P4 are parietal electrodes. The parietal lobe is part of the dorsal attention network, the brain circuit that allocates attentional resources toward goals and tasks [8]. Parietal alpha decreases when someone is actively directing attention at something and increases when attention is withdrawn [10]. These positions

give the model access to the classic alpha engagement signature from the region where it tends to be strongest and most consistent across participants.The eight positions together cover frontal, central, and parietal areas on both hemispheres. Bilateral coverage matters because attention-related EEG changes are not always symmetric. Some participants show stronger signals on the left side, others on the right. Recording from both hemispheres gives the models a fuller picture of each participant's engagement state. Eight channels is also the hardware limit of a single Cyton board. The alternative was adding the OpenBCI Daisy module to reach 16 channels, but that adds considerable setup complexity and roughly 20 extra minutes per session. For frequency-band power analysis, the eight positions chosen here are well supported in the literature as sufficient for theta, alpha, and beta engagement classification.Before each session, electrode impedance was checked in the OpenBCI GUI [9]. Every electrode was brought below 5 kilohms before recording started. Above that level, the gel-to-skin contact resistance causes the 24-bit ADC to pick up noise rather than brain signal. Getting all eight electrodes under 5 kilohms typically took 10 to 15 minutes of gel application and light scalp abrasion per participant.

3.2.3 VR Headset

A Meta Quest 2 VR headset was used to display all lecture videos.The headset has a resolution of 1832by1920 pixels per eye, a 90Hz display refresh rate, and a 110 degree horizontal field of view. It operates as a standalone Android device with no PC tethering required for playback. Audio was delivered through the built-in speakers. Participants held the VR controllers but were not asked to press anything during recording sessions.The primary reason for using the VR headset over a standard monitor is environmental isolation. When someone watches a video on a screen, they can see their phone, the rest of the room, the researcher, and other people. Any of those can cause EEG changes that have nothing to do with the lecture. The headset blocks the participant's view of the entire room. The only visual input during the session is the lecture video displayed inside the headset. This improves the signal to noise ratio of the training data because EEG changes during the session are far more likely to reflect the lecture content than environmental distractions.



Fig. 3.4 Meta Quest 2 VR headset worn by a participant alongside the OpenBCI Cyton electrode cap during a recording session

3.3 Participants

Sixteen undergraduate students from the Computer Engineering department at Bahria University, Islamabad participated in this study. All were between 18 and 24 years old, had normal or corrected-to-normal vision, and reported no history of epilepsy, neurological conditions, or sensitivity to flashing lights [11]. Each participant signed a written consent form before any recording began. The form explained what the study involved, that participation is voluntary and can be stopped at any time, and that data would be stored using a participant ID rather than their name.Sixteen participants is a reasonable sample size for this type of study. The key factor is the evaluation method. This project uses subject-independent cross-validation, where the model is always tested on participants it has never been trained on. Several published EEG en-

gagement papers have used between 6 and 30 participants under this same protocol. What matters is that the evaluation is honest and the test participants are genuinely unseen, not that the dataset is particularly large.Table 3.2 summarises the participant details.

Table 3.2: Participant details

Detail	**Value**
Number of participants	16
Age range	18 to 24 years
University	Bahria University, Islamabad
Department	Computer Engineering
Sessions per participant	6 (one per subject)
Total sessions recorded	96
Exclusion criteria	Epilepsy, neurological conditions, photosensitivity, VR incompatibility

3.4 Experimental Protocol

3.4.1 Subjects and Lecture Videos

Six university subjects were used: Digital Logic Design, Object-Oriented Programming, Computer Fundamentals, Engineering Ethics, Entrepreneurship, and Communication Skills. These were selected because they represent different types of cognitive demand. Digital Logic Design involves formal Boolean algebra and circuit diagrams. Object-Oriented Programming requires reasoning about object hierarchies and method calls. Computer Fundamentals covers material most participants had already encountered in their first year. Engineering Ethics asks students to reason about professional responsibilities without a single correct answer. Entrepreneurship involves business and financial concepts less familiar to most engineering students. Communication Skills focuses on professional written and spoken communication.Having this variety of subjects tests whether engagement detection generalises across different types of cognitive demand. If the models perform consistently across all six subjects, that is evidence they are learning general engagement patterns rather than something specific to one type of lecture content.

For each subject, three lecture videos were used at three difficulty levels. All videos were sourced from YouTube and selected to match the required content and difficulty for each category. For Digital Logic Design, the easy video covered basic logic gates and truth tables, the medium video explained Karnaugh maps and combinational cir-

cuit simplification, and the hard video covered sequential circuits, flip-flops, and timing diagrams at a faster-than-typical undergraduate pace. For Object-Oriented Programming, the easy video introduced classes and basic methods in Python, the medium video covered inheritance and polymorphism, and the hard video went through design patterns including factory and singleton with full implementation examples.For Computer Fundamentals, the easy video reviewed number systems and binary arithmetic, the medium video explained CPU architecture and the fetch-decode-execute cycle, and the hard video covered memory hierarchy, cache organisation, and virtual memory management. For Engineering Ethics, the easy video introduced the IEEE code of ethics, the medium video discussed real engineering failure case studies, and the hard video covered legal frameworks around intellectual property and professional accountability. For Entrepreneurship, the easy video explained basic business models, the medium video covered the business model canvas, and the hard video went through venture capital funding and financial projections. For Communication Skills, the easy video covered professional email writing, the medium video explained technical report structure, and the hard video covered formal presentation and persuasive communication frameworks.All videos were reviewed before use. Videos with poor audio quality, heavy background music, or presentation styles unlikely to generate consistent engagement variation were excluded. Each selected video was clipped to exactly six minutes and encoded at consistent visual quality for VR playback.

3.4.2 Session Structure

Each session covered one subject and followed the same structure for every participant. The session started with 2.5 minutes of relaxation, during which the participant sat quietly inside the VR headset viewing relaxing nature content. The easy lecture then played for six minutes, followed by another 2.5-minute relaxation period. Then the medium lecture for six minutes. Another relaxation period. Then the hard lecture for six minutes. A final 2.5-minute relaxation period closed the session. All six subject sessions were spread across separate visits to prevent fatigue.

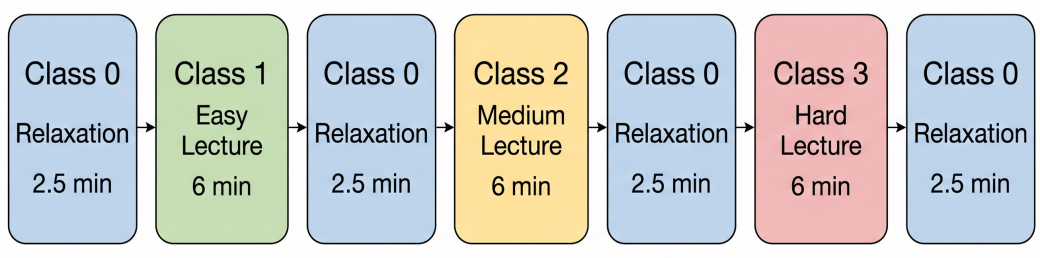


Fig. 3.5 Session structure showing the alternating relaxation and lecture segments.

The relaxation periods between lectures serve two purposes. They provide baseline EEG data showing what each participant's signal looks like when not doing anything, and they give the brain time to settle between lectures so that cognitive state from one segment does not carry over to the start of the next. Including setup and electrode preparation, each visit took approximately 55 to 60 minutes. Table 3.3 summarises the data collection parameters.



Fig. 3.6 Participants seated with EEG cap and VR headset during a recording session

Table 3.3: Data collection summary

Parameter	Value
Subjects recorded	6
Difficulty levels per subject	3 (easy, medium, hard)
Lecture duration	6 minutes each
Relaxation duration	2.5 minutes each
Relaxation periods per session	4
Total recording time per session	Approximately 37.5 minutes
Total sessions	96 (16 participants x 6 subjects)
Windows after artifact rejection	80,640
Artifact rejection rate	Approximately 3 percent

3.5 EEG Signal Acquisition

EEG signal acquisition was handled entirely through the BrainFlow library [3]. BrainFlow provides a unified Python API for streaming from EEG hardware without writing low level USB or Bluetooth code. The Cyton board is identified by its board ID (board_id = 0) and connects to the host PC through a serial port on the USB Bluetooth dongle (typically COM3 or /dev/ttyUSB0). A BoardShim object is created with these parameters and a session is opened before recording starts.

Before starting the data stream, an impedance check was performed using the OpenBCI GUI. This graphical application provides a real time view of all eight channels and colour codes each electrode by its current impedance value. All electrodes were brought below 5 kilohms before recording started. Once impedance was acceptable, the OpenBCI GUI was closed and the BrainFlow script was started to take over the serial connection.

BrainFlow maintains an internal circular buffer that holds the most recent samples from the board. A call to BoardShim.get_board_data() retrieves and clears all samples accumulated since the last call. During offline recording sessions, this call was made every 250 samples, pulling one second of data at a time. The raw data matrix returned by BrainFlow has shape (channels + metadata, samples). The first 8 rows are the EEG channel values in raw ADC counts. Additional rows carry timestamp, accelerometer, and status information. Only the 8 EEG rows are used.

For the 96 offline recording sessions, the script wrote each batch of retrieved samples to a CSV file as the session progressed. A separate marker channel was written at the same time, logging the current session segment (relaxation, easy, medium, or hard) as an integer label alongside the timestamp. This marker information was used during

preprocessing to assign condition labels to each window. The final CSV for each session contained 8 EEG channels plus the marker column, sampled at 250 Hz for the full session duration of approximately 37.5 minutes. At 250 Hz over 37.5 minutes, each session file contained approximately 562,500 rows before any preprocessing.

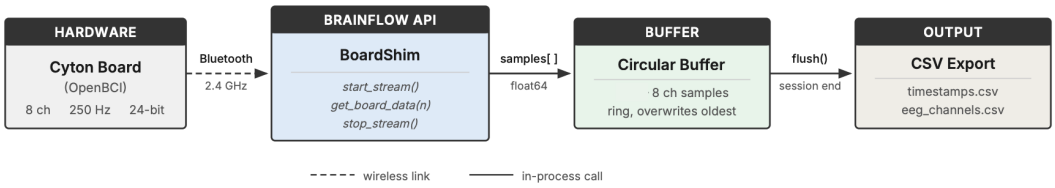


Fig. 3.7 EEG signal acquisition block diagram showing the Cyton board, Bluetooth link, BrainFlow BoardShim API, circular buffer, and CSV export path

One recording constraint was Bluetooth stability. The Cyton board's Bluetooth link occasionally dropped packets during long sessions, particularly when the dongle was farther than one metre from the board. To minimise this, the dongle was placed on the desk immediately in front of the participant rather than in a rear USB port, and sessions were conducted in a room without other active Bluetooth devices. Any session with more than 0.5 percent packet loss (checked by inspecting the sample counter column in the BrainFlow output) was repeated with that participant. None of the 96 sessions in the final dataset exceeded this threshold.

3.6 VR Environment

The Meta Quest 2 headset was used in standalone mode and without a PC connection.The lecture videos were sideloaded onto the headset as MP4 files using the Android Debug Bridge (ADB) tool over USB.A custom Android application was built and installed on the headset to play the videos in sequence according to the session protocol.The app displayed each video at a comfortable virtual screen size equivalent to approximately a 2 metre screen at 3 metres viewing distance, positioned at eye level directly in front of the participant in the virtual space.

The virtual environment around the screen was set to a plain neutral grey room with no distracting visual elements.Participants were seated in a standard office chair placed in a fixed position in the recording room.The recording room had no windows and used controlled overhead lighting kept at the same level across all sessions.Participants were told not to move their heads excessively during recording, as large head movements

produce muscle noise in the EEG signal at the neck and jaw electrodes.A brief practice period at the start of each visit familiarised each participant with the headset before the electrodes were applied.

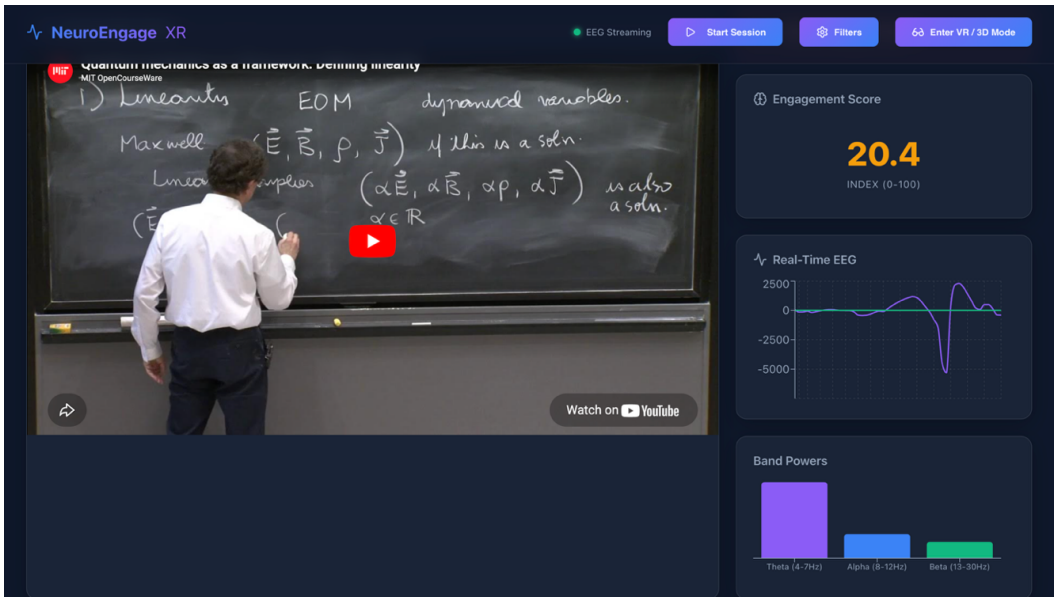


Fig. 3.8 view of the VR lecture environment as seen through the Meta Quest 2 headset, showing the lecture video displayed at comfortable viewing distance in a neutral virtual room

All lectures were recorded and had the same quality.All clips were auto encoded in 1080p. Resolution is 30 FPS and Audio is stereo, 192 kbps.The Quest 2 displays at 1832 by 1920 Maximum 1080p source content, however accepts pixels per eye.The built-in speaker was used for audio. in speakers with fixed volume level, set at the beginning of each session without being adjusted during recording.Participants were requested to verify that the sound was loud and clear. Prior to the start of the session protocol.

The VR app logged timestamps at each video transition, recording the exact system time when each relaxation or lecture segment started and ended. These timestamps were synchronised with the BrainFlow CSV timestamps using the same system clock on the host PC. The synchronisation accuracy was within 5 milliseconds, which is well within the 2-second window length used during preprocessing.

3.7 Data Acquisition

Each recording visit followed a fixed setup procedure. The participant arrived, read and signed the consent form, and was seated in the recording chair. The researcher

applied electrode gel to the eight scalp positions and fitted the electrode cap. Reference electrodes were placed on the left and right mastoids using adhesive patches and conductive paste. The impedance check in the OpenBCI GUI was then run and electrodes were adjusted until all readings fell below 5 kilohms. This setup typically took 10 to 15 minutes.

Once impedance was acceptable, the participant put on the Meta Quest 2 headset. The researcher connected to the headset wirelessly using the Quest Link companion app to verify that the video playback app had launched correctly and the first relaxation segment was queued. The BrainFlow acquisition script was started on the host PC. A five-second countdown was shown inside the headset before the first relaxation segment began. The researcher noted the exact system timestamp at the start of the session, which served as a reference for aligning the BrainFlow data with the VR app event log.

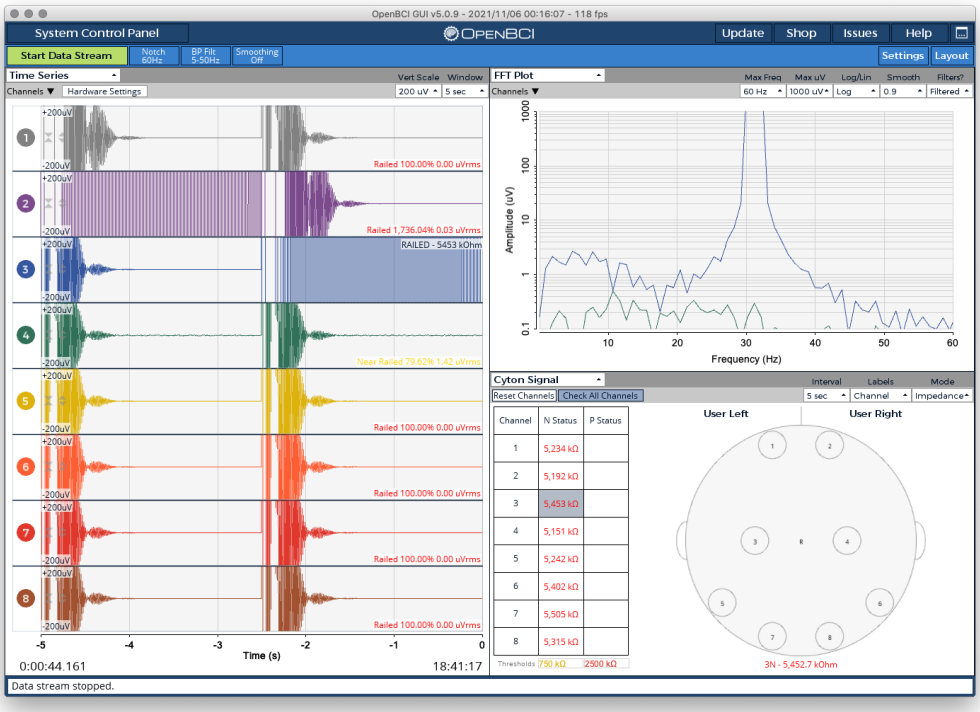


Fig. 3.9 Data acquisition session flow showing setup, impedance check, recording sequence, and post-session quality check steps

The session protocol ran autonomously from that point. The VR app displayed each segment in sequence according to the fixed timing described in Section 3.4.2. The BrainFlow script continuously wrote data to the session CSV file throughout. The researcher remained in the room but did not speak to the participant during recording.

After the final relaxation period ended, the VR app displayed a completion message. The BrainFlow script was stopped and the CSV file was closed.

Each session CSV file was named using the convention P[ID]_S[Subject].csv, where ID is the two-digit participant number and Subject is the subject code.For example P03_S2.csv holds the session for participant 3 watching the Object-Oriented Programming lectures. The 96 session files were stored in a local directory and backed up immediately after each visit.

After each session, a quick quality check was performed by loading the CSV file and plotting all eight channels for a visual inspection. The researcher checked for flat channels (indicating a disconnected electrode), persistent high-frequency noise (indicating poor gel contact or EMG contamination), and excessive 50 Hz interference (indicating the notch filter would need to work harder during preprocessing). Any session with obvious recording problems was flagged and repeated. Three sessions required repetition across the full data collection period, two due to electrode contact problems and one due to a Bluetooth interruption that affected 8 minutes of the recording.

3.8 Signal Preprocessing

Both Pipeline A and Pipeline B share the same first four preprocessing steps. These steps convert the raw 8-channel ADC data from each session CSV file into clean, windowed EEG segments ready for the diverging processing that each pipeline applies next. Fig. 3.8 shows these four shared steps as a block diagram.

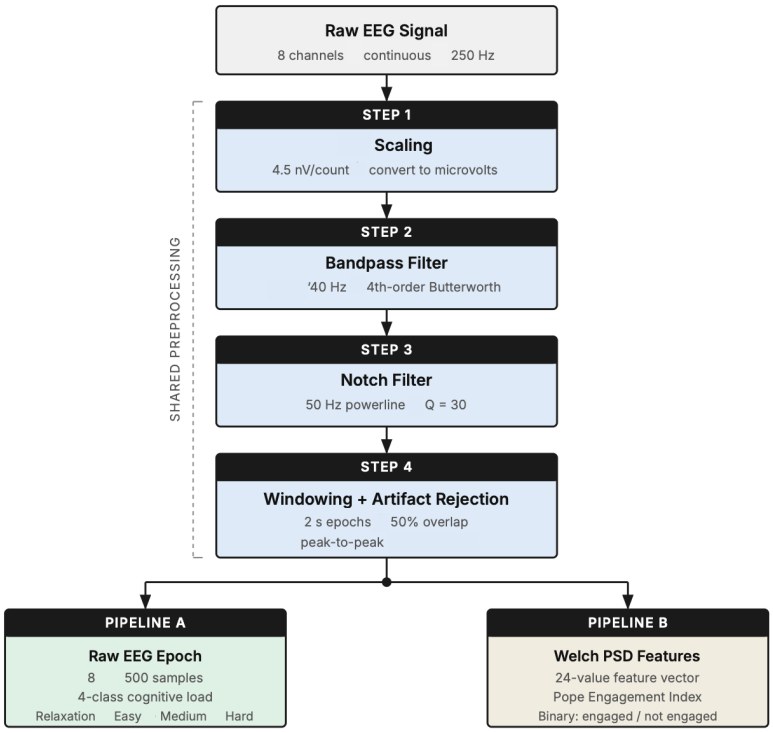


Fig. 3.10 Signal preprocessing pipeline. The four shared steps (scaling, bandpass filter, notch filter, and windowing with artifact rejection) feed both Pipeline A and Pipeline B. After windowing, the two pipelines diverge.

Step 1 converts raw ADC counts to microvolts using the Cyton board gain settings. The formula applied to each sample is:

$$\mu V = \text{count} \times \frac{4.5}{(2^{23}-1) \times 24} \times 10^6 \tag{3.1}$$

where μV is the signal amplitude in microvolts, *count* is the raw ADC integer sample, 4.5 V is the Cyton board reference voltage, $2^{23}-1$ is the 24-bit ADC maximum, 24 is the programmable gain, and 10^6 converts volts to microvolts.

The 4.5 volt reference voltage and 24 bit ADC range give a conversion factor of approximately 0.02235 microvolts per count. This is applied to all eight channels simultaneously using NumPy array multiplication. After this step, the signal is in the physically meaningful unit of microvolts rather than arbitrary ADC counts.

Step 2 applies a bandpass filter between 1Hz and 40Hz to remove content outside the EEG frequency range of interest. The lower cut at 1 Hz removes slow DC drifts caused by electrode polarisation and temperature changes. The upper cut at 40Hz removes high-frequency noise from facial muscle activity and electrical interference. A 4th-

order Butterworth filter was used, applied in zero-phase mode by running it forward and backward through the data using the SciPy sosfiltfilt function. This two-pass approach eliminates phase distortion that a single forward pass would introduce.

Step 3 applies a notch filter at 50Hz to remove power line interference.Pakistan's electrical grid runs at 50Hz and this frequency couples into EEG recordings through the electrode cables.The notch filter has a quality factor of 30, giving a 3 dB bandwidth of approximately 1.7Hz centered on 50Hz.This removes the interference narrowly without affecting the beta band, which extends up to 30Hz.

Step 4 segments the continuous filtered signal into two second windows with 50 percent overlap. At 250Hz, each window contains 500 samples per channel. The 50 percent overlap means each new window starts 250 samples after the previous one, effectively doubling the number of windows produced from the same recording time. Each window is then checked for artifacts. If the peak to peak amplitude in any channel exceeds 200 microvolts, the entire window is removed. Normal EEG amplitude falls between 10 and 100 microvolts. Values above 200 microvolts indicate eye blinks, jaw clenches, or poor electrode contact rather than brain activity.Approximately 3 percent of windows were removed by this threshold across all 96 sessions. Table 3.4 summarises all preprocessing parameters.

Table 3.4: Signal preprocessing parameters (shared by Pipeline A and Pipeline B)

Parameter	**Value**	**Reason**
Bandpass range	1 to 40 Hz	Preserves theta, alpha, and beta while removing DC drift and muscle noise
Filter type	4th-order Butterworth, zero-phase	Flat passband; two-pass removes phase distortion
Notch filter	50 Hz, Q = 30	Removes grid noise without affecting the beta band
Window length	2 seconds (500 samples)	Long enough for PSD estimation; short enough to track fast state changes
Window overlap	50 percent	Doubles window count from the same recording time
Artifact threshold	200 microvolts peak-to-peak	Removes blinks and muscle artifacts
Windows after rejection	80,640	3 percent rejection rate across all 96 sessions

3.9 Pipeline A: Raw EEG Processing for Four-Level Classification

Pipeline A keeps the cleaned, windowed EEG signal in its raw time-domain form rather than converting it to a compact frequency summary. After the four shared preprocessing steps described in Section 3.8, each window is an 8-channel by 500-sample array. This array goes directly into the model without any frequency-domain conversion. The reason for this choice is that the four target classes in Pipeline A (relaxation, easy lecture, medium lecture, and hard lecture) differ not only in their average frequency content but in how the temporal waveform behaves moment to moment across the six-minute segment. Keeping the time-domain representation gives the model access to temporal detail that would be lost in a compact feature vector.

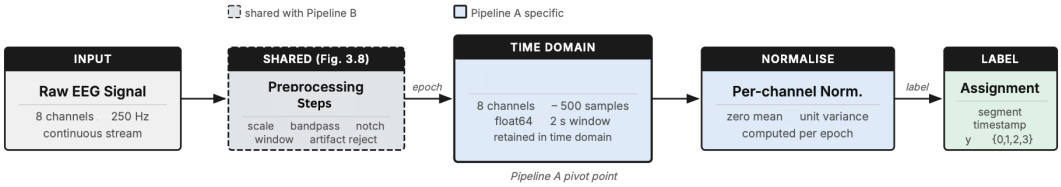


Fig. 3.11 Pipeline A processing flow. Raw EEG goes through the four shared preprocessing steps, is kept in time-domain form as an 8x500 array, normalized per channel, and labeled by session segment timestamp.

3.9.1 Step 1: Scaling to Microvolts

The Cyton board records raw integer values from its ADC. These are converted to microvolts as described in Section 3.8 using the formula:

$$\mu V = \text{count} \times \frac{4.5}{(2^{23}-1) \times 24} \times 10^6 \tag{3.2}$$

where μV is the amplitude in microvolts, *count* is the raw ADC sample, 4.5 V is the reference voltage, $2^{23}-1$ is the 24-bit range, and 24 is the programmable gain.
The conversion factor of approximately 0.02235 microvolts per count is applied to all eight channels simultaneously using array multiplication.

3.9.2 Step 2: Bandpass Filtering

A 4th order Butterworth bandpass filter with cutoffs at 1Hz and 40Hz is applied in zero-phase mode using SciPy sosfiltfilt. The lower cutoff removes DC drift and electrode polarisation effects. The upper cutoff removes high-frequency muscle noise. The

two-pass application removes all phase distortion, which is important in Pipeline A because the raw waveform shape is preserved all the way to the model input. Any phase distortion introduced by a single-pass filter would alter the temporal structure of the signal and potentially affect what the model learns.

3.9.3 Step 3: Notch Filter at 50 Hz

A notch filter at 50 Hz with quality factor 30 removes power-line interference. The narrow bandwidth of approximately 1.7 Hz ensures the filter does not affect the beta band between 13 and 30 Hz. Applied using the SciPy iirnotch and sosfiltfilt functions in the same zero-phase manner as the bandpass filter.

3.9.4 Step 4: Windowing and Artifact Removal

The filtered signal is divided into two-second windows with 50 percent overlap (500 samples per window, new window every 250 samples). Each window is checked for artifacts using the 200 microvolt peak-to-peak threshold applied independently to each channel. If any single channel in a window exceeds this threshold, the whole window is discarded. This is more conservative than a per-channel discard strategy but avoids passing partially contaminated windows to the model.Approximately 3 percent of windows were removed this way.

3.9.5 Step 5: Assembling the Raw Window

After artifact rejection, each clean window is arranged as a (8, 500) NumPy array. No frequency-domain conversion is applied. The 500 time samples per channel represent the signal at 4 millisecond intervals across the 2-second window. This preserves all temporal dynamics including the fast oscillatory structure of beta activity, the slower rhythm of alpha, and the even slower theta waves. A model receiving this representation can learn to identify these rhythms through its convolutional or attention layers rather than relying on precomputed band power summaries.

3.9.6 Step 6: Per-Channel Normalization

Each of the eight channels in a window is normalised independently so that its 500 values have a mean of zero and a standard deviation of one. This is done per channel rather than across the whole window. The reason is that absolute amplitude varies substantially across participants due to differences in skull thickness, electrode contact quality, and individual neural variation. Without normalisation, a model might learn

to recognise participants by their amplitude rather than their engagement pattern. The mean and standard deviation used for normalisation are always computed from only the training data in each cross-validation fold and then applied to the test windows. Computing these statistics from test data would allow information about the test participants to leak into the training process.

3.10 Pipeline A: Four-Level Labeling

3.10.1 Assigning the Class Label

Pipeline A does not compute a label from the EEG signal itself. Each session is recorded with exact timing metadata marking where each segment starts and ends. A window's label comes from checking which segment its timestamp falls inside:
Class 0 = Relaxation Class 1 = Easy Class 2 = Medium Class 3 = Hard
This labeling has no ambiguity. The session protocol timing is known exactly from the BrainFlow marker column and VR app event log. Every window that starts and ends inside the hard lecture segment gets Class 3, regardless of what the EEG inside that window looks like. This is the fundamental difference between the two pipelines: Pipeline A labels by task condition, while Pipeline B labels by a signal computed from the EEG itself.

3.10.2 Limitations of Condition-Based Labels

Because the class label comes from session timing rather than a computed index, there is no threshold to set and no per-participant adjustment to make. Every participant's easy-lecture windows get Class 1 regardless of how their individual EEG happens to look. This simplicity is also the main limitation of Pipeline A as an engagement measure. A participant can be fully absorbed during an easy lecture or completely disengaged during a hard one. Pipeline A has no way to distinguish those cases because the label only reflects which video was playing. This is exactly the gap that Pipeline B is designed to close.

3.11 Pipeline B: Engagement Signal Processing

Pipeline B runs the same first four preprocessing steps as Pipeline A: scaling to microvolts, bandpass filtering, notch filtering, and windowing with artifact rejection. These steps are shared and run once per session. Where Pipeline B diverges is after windowing. Instead of keeping the raw 8x500 array, it converts each clean window into a

24-number frequency-domain feature vector using Welch PSD estimation. This compact vector captures the engagement-relevant band power information from all eight channels.

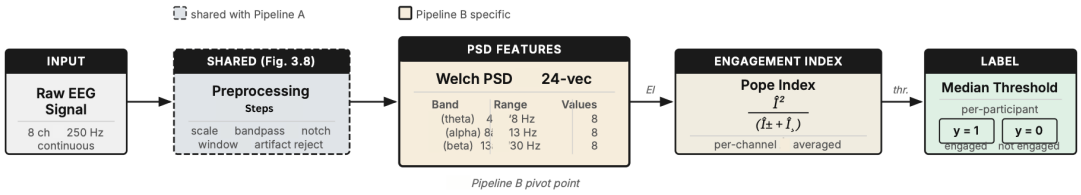


Fig. 3.12 Pipeline B processing flow. After the four shared preprocessing steps, each window is converted to a 24-number PSD feature vector (8 channels x 3 frequency bands: theta, alpha, beta) and labeled using the Pope Engagement Index with per-participant median thresholding.

3.11.1 Step 5: Welch PSD Feature Extraction

For each clean window and each channel, the Welch method estimates how much power the signal contains at each frequency [16]. The Welch approach splits the 500-sample window into shorter overlapping segments, computes the frequency spectrum of each segment separately, and averages the results. Averaging reduces estimation noise compared to computing a single spectrum from the whole window at once. Each segment is 256 samples long, giving a frequency resolution of 250/256 = 0.977Hz per bin.
For each channel, three frequency band integrals are computed. The theta band covers 4 to 8Hz, the alpha band covers 8 to 13Hz, and the beta band covers 13 to 30Hz. Integration uses the trapezoidal rule to sum the area under the power spectrum curve within each band. This gives three numbers per channel: theta power, alpha power, and beta power. Across eight channels, this produces 24 numbers per window. These 24 values are arranged as all theta values first (channels 1 to 8), then all alpha values (channels 1 to 8), then all beta values (channels 1 to 8). This ordering groups the same band across all channels together, which is the natural structure for a model that might learn channel-independent band relationships.

3.11.2 Step 6: Feature Normalization

The 24-number feature vectors are normalised so that each of the 24 features has a mean of zero and a standard deviation of one across all windows in the training data. Different participants produce different absolute EEG power levels for reasons unrelated to engagement: skull thickness, electrode contact quality, and individual neurological

variation all affect the raw power magnitude. Without normalisation, a model could learn to identify participants by their power level rather than the engagement-related pattern in the ratio between bands. Normalising across participants forces the models to focus on the relative relationship between theta, alpha, and beta rather than absolute levels. The normalisation statistics are computed from only the training fold and then applied to the test fold, following the same leakage-prevention rule used throughout.

3.12 Pipeline B: Engagement Labeling

3.12.1 The Pope Engagement Index

The Pope Engagement Index is used to assign a binary engagement label to each two-second window [17]. The index was originally developed by Pope et al. for monitoring operator attention in automated systems. It captures the relationship between active cognitive processing (reflected in beta activity) and passive or unfocused mental states (reflected in alpha and theta activity). The formula computes a ratio of beta power to the combined alpha and theta power, averaged across all eight channels:

$$\text{Pope Index} = \frac{1}{N}\sum_{c=1}^{N}\frac{\beta_c}{\alpha_c + \theta_c + \varepsilon} \tag{3.3}$$

where $N = 8$ is the number of EEG channels; β_c, α_c, and θ_c are the Welch PSD band-power integrals for channel c in the beta (13–30 Hz), alpha (8–13 Hz), and theta (4–8 Hz) bands respectively; and $\varepsilon = 10^{-10}$ is a small constant added to prevent division by zero. A high Pope Index means beta dominates over alpha and theta, which corresponds to focused cognitive engagement. A low Pope Index means alpha and theta rise relative to beta, which corresponds to disengagement, boredom, or mind-wandering. Fig. 3.11 illustrates how the Pope Index is computed from the 24-number feature vector.

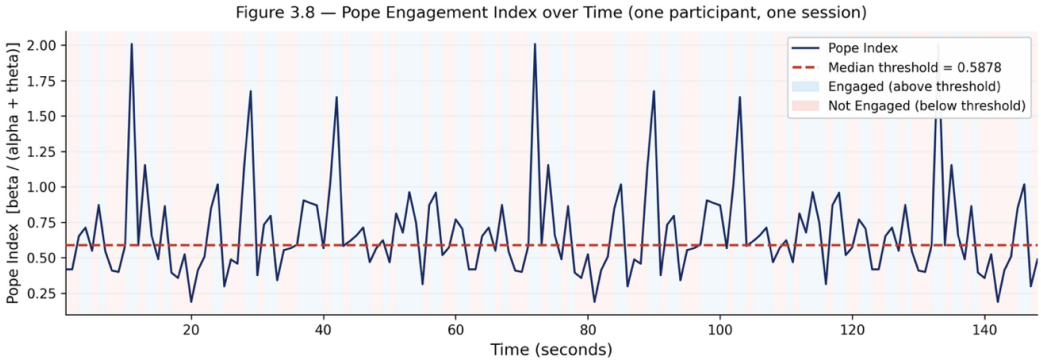


Fig. 3.13 Pope Index over time for one participant. Windows above the dashed threshold line are labeled engaged.

3.12.2 Per-Participant Median Thresholding

Each window is labeled engaged (1) or not engaged (0) by comparing its Pope Index to the median Pope Index for that specific participant across their entire session. A window above the participant's own median receives label 1. A window at or below the median receives label 0.

Using each person's own median matters for two reasons. First, participants produce very different absolute Pope Index values for reasons that have nothing to do with engagement. One participant's engaged state might naturally produce a Pope Index of 0.8, while another's might produce 0.3. A single global threshold applied to both would systematically mislabel one of them. Using the per-participant median adjusts the threshold to each person's own baseline, so the label captures relative changes within that person rather than absolute comparisons across people.

Second, the median threshold gives each participant roughly equal numbers of engaged and not-engaged windows. By definition, approximately half of a person's windows fall above their median and half fall below. This keeps the training data balanced across the two classes. A global threshold could give one participant 80 percent engaged windows and another only 20 percent, which would bias training toward one class and degrade classification performance.

3.13 Cross-Validation Setup

All three models are trained and evaluated using 5 Fold Stratified Cross Validation with subject independent splits. The 16 participants are divided into 5 groups. In each of the 5 rounds, one group serves as the test set and the remaining four groups form the training set. No participant appears in both training and testing in any single round.

Every participant ends up in the test set exactly once across the five rounds. Fig. 3.12 illustrates the split structure.

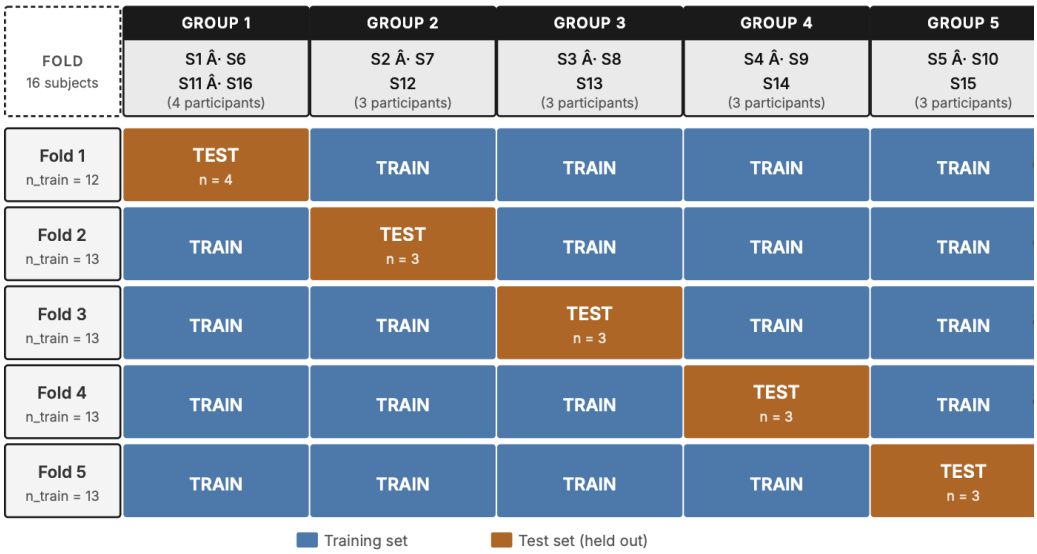


Fig. 3.14 5-Fold subject-independent cross-validation. 16 participants are divided into 5 groups. In each fold, one group is held out for testing while the remaining participants train the model. No participant appears in both sets in the same fold.

This evaluation protocol tests whether the system would work on a new student it has never encountered before. Most published EEG engagement papers use window-level random splits, where windows from the same person can appear in both training and testing. That arrangement inflates accuracy because the model can learn each person's individual EEG fingerprint rather than a general engagement pattern. Subject-independent splits make that impossible.

The reported accuracy is the mean across all five folds, and the standard deviation across folds is also reported. A high standard deviation would indicate that the result changes substantially depending on which participants happen to be in the test group, suggesting poor generalisation. Table 3.5 summarises the cross-validation setup.

Table 3.5: Cross-validation setup

Setting	**Value**
Cross-validation type	5-Fold Stratified, subject-independent
Grouping variable	Participant ID
Training data per fold	12 to 13 participants
Test data per fold	3 to 4 participants

Table 3.5: Cross-validation setup

Setting	Value
Result reported	Mean accuracy across 5 folds
Stability measure	Standard deviation across 5 folds

3.14 Proposed Model Architectures

Three deep learning models are proposed. All three take the same 24-number PSD feature vector produced by Pipeline B as input and output a binary engaged or not-engaged prediction. The models differ in their internal architecture and training strategy. The first, EEG-Deformer++, extends a published EEG-specific Transformer with components designed for frequency-domain features. The second, EEG-DINO++, applies a self-supervised knowledge distillation framework to learn more noise-robust representations. The third, EEG-HybridFormer, combines the feature extraction backbone of the first model with the training strategy of the second. Table 3.7 at the end of this section summarises the key design and training parameters for all three.

3.14.1 EEG-Deformer++

EEG-Deformer++ extends the EEG-Deformer published by Ding et al. in 2024 [19]. The original model added a spatial convolution before a Transformer encoder and showed strong results on raw EEG motor imagery tasks. EEG-Deformer++ adapts and extends this architecture for frequency-domain PSD features and binary engagement classification. Three new components are added: a Multi-Scale Temporal-Spatial Fusion (MSTSF) module, an Adaptive Channel Attention (ACA) block, and a gated classification head. Fig. 3.13 shows the complete architecture.

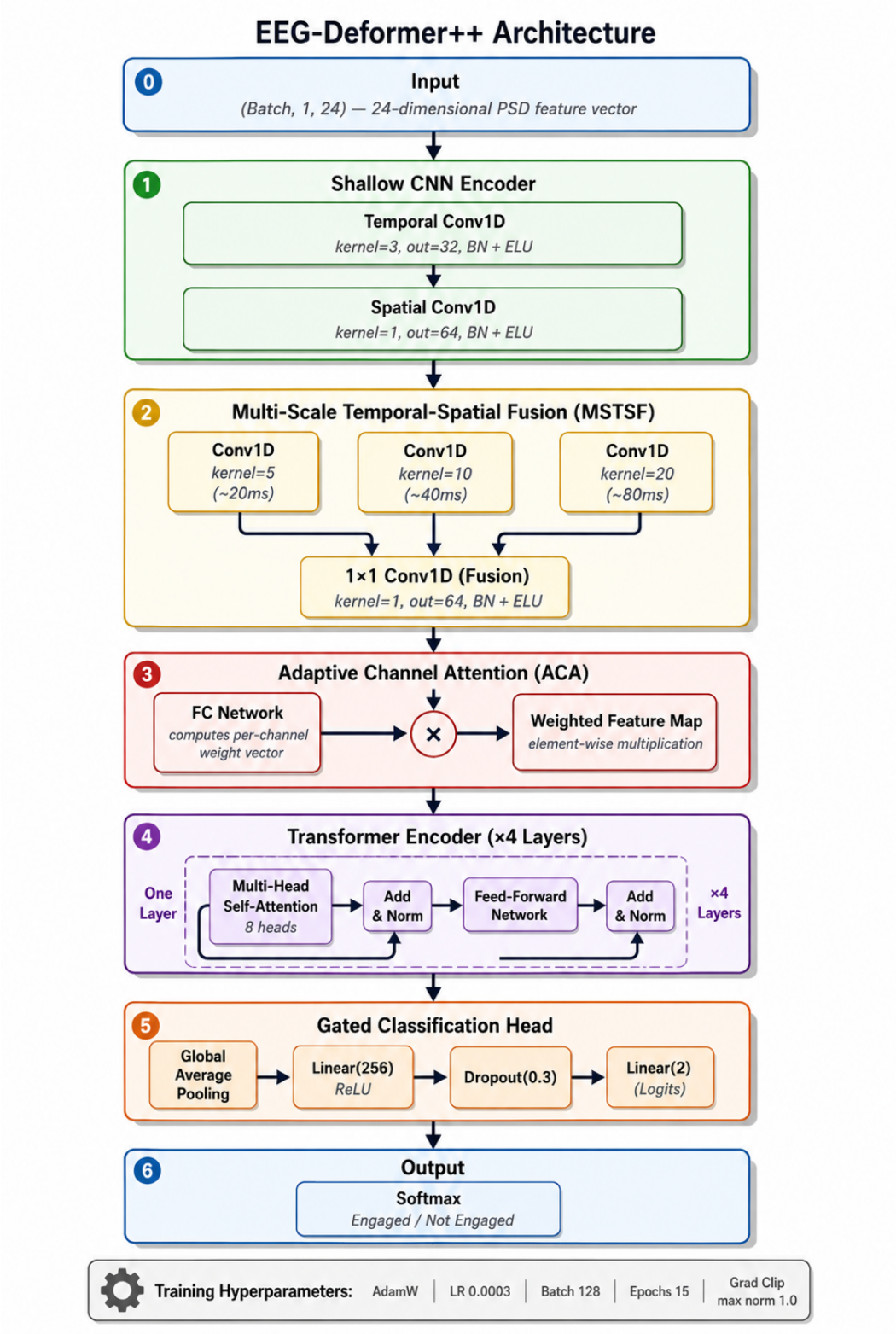


Fig. 3.15 EEG-Deformer++ architecture. Input flows left to right through the CNN encoder, MSTSF module, Adaptive Channel Attention block, 4-layer Transformer encoder, and gated classification head to produce the binary engagement prediction.

The input to EEG Deformer++ is a tensor of shape (B, 1, 24), where B is the batch size, 1 is the channel dimension (the single PSD feature vector is treated as a single input channel), and 24 is the sequence length corresponding to the 24 PSD feature values. This shape allows the model to treat the sequence of 24 features as a one-dimensional signal over which convolutions and attention mechanisms can be applied.

The first stage is the CNN Encoder , which has two convolutional layers. The first is a temporal convolution with 32 filters , kernel size 3 , padding 1 , stride 1 , followed by Batch Normalisation and GELU activation. This layer learns local patterns in the sequence of 24 features, detecting which neighbouring features co-vary. For example, beta power values at adjacent frontal electrodes often move together during periods of focused attention.The second layer is a spatial convolution with 64 filters and kernel size 1, also followed by Batch Normalisation and GELU. This pointwise convolution mixes information across the 32 learned feature maps to produce 64 richer representations. A dropout layer with rate 0.1 follows to reduce overfitting. After the CNN Encoder, the tensor has shape (B, 64, 24).

The second stage is the Multi-Scale Temporal-Spatial Fusion (MSTSF) module. Fig. 3.14 shows this module in detail. Three parallel convolution branches process the encoder output simultaneously. The first branch uses kernel size 5 with padding 2 and captures short-range patterns, roughly the relationship between two or three adjacent features. The second branch uses kernel size 10 with padding 5 and captures medium-range patterns across four or five features. The third branch uses kernel size 20 with padding 10 and captures long-range patterns spanning nearly the full 24-feature sequence. Each branch consists of a Conv1D layer followed by Batch Normalisation and GELU. The three branch outputs each have shape (B, 64, 24). They are concatenated along the channel dimension to produce a (B, 192, 24) tensor. A final 1x1 convolution fuses these 192 channels back down to 64, giving shape (B, 64, 24). This design is motivated by the observation that engagement shifts happen at different temporal scales: a sudden distraction produces a fast change in the feature pattern, while gradual mind-wandering produces a slower drift. Looking at only one temporal scale would miss one or the other.

The third stage is the Adaptive Channel Attention (ACA) block. This block computes an importance weight for each of the 64 feature channels based on the current input. A global average pooling operation summarises each channel into a single value, producing a (B, 64) vector. This goes through a fully connected layer reducing to 32 units with GELU activation, followed by another fully connected layer expanding back to 64 units with Sigmoid activation. The resulting 64 weights are in the range (0, 1).

Each weight is applied to its corresponding channel in the (B, 64, 24) feature tensor by element-wise multiplication. Channels carrying more discriminative engagement information receive higher weights and contribute more to downstream processing. Channels that contribute little are suppressed. The reason this matters is that different participants show their strongest engagement signals at different electrode positions. A fixed-weight aggregation treats all frequency-band and electrode combinations equally, which is suboptimal when the most informative features vary across people.

The fourth stage is a four-layer Transformer Encoder [20]. Before entering the Transformer, the feature tensor is reshaped to (B, 24, 64), treating the 24 positions as the sequence length and the 64 channels as the embedding dimension. Each Transformer layer has three sub-components. The first is Multi-Head Self-Attention (MHSA) with 8 attention heads. Each head uses a key/query/value dimension of 8 (since 64 / 8 = 8). MHSA allows every position in the 24-element sequence to attend to every other position simultaneously. This lets the model capture relationships between, for example, the theta power at frontal electrodes and the beta power at parietal electrodes, which are 16 positions apart in the sequence. The attention computation is:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V \tag{3.4}$$

where Q, K, and V are the query, key, and value matrices derived from the input sequence; $d_k = 8$ is the key dimension; and $\sqrt{d_k}$ is a scaling factor that prevents the dot products from becoming very large, which would push the softmax into saturation with near-zero gradients. The second sub component is a Feed Forward Network (FFN) with two linear layers: 64 -> 256 with GELU activation , then 256 -> 64. The FFN applies a non-linear transformation to each position independently after the attention has mixed information across positions. The third sub-component is Layer Normalisation applied before both the attention and FFN sub-layers (pre-norm configuration), and residual connections around both. Dropout at rate 0.1 is applied inside each Transformer layer. After four such layers, the output has shape (B, 24, 64).

The final stage is the Gated Classification Head. Global average pooling across the sequence dimension reduces the (B, 24, 64) output to (B, 64). This 64-dimensional vector then passes through a gating mechanism: two parallel linear projections (both 64 -> 64) are computed, one followed by Sigmoid activation (the gate) and one followed by Tanh activation (the value). These are multiplied element-wise. This gating mechanism allows the head to selectively suppress dimensions of the representation that are not relevant to the current input, similar in spirit to the gating in LSTM cells. The

gated vector then goes through a linear layer (64 -> 256), GELU activation , dropout at rate 0.3 , and a final linear layer (256 -> 2) with Softmax, producing the probability distribution over engaged and not-engaged classes.

Training uses the AdamW optimiser [21] with a learning rate of 0.0003 , weight decay of 0.01 , and batch size of 128. A ReduceLROnPlateau scheduler halves the learning rate when validation loss does not improve for three consecutive epochs. Training runs for 15 epochs. Gradient clipping with a maximum norm of 1.0 was added because the Transformer weights occasionally produced large gradient updates in early epochs that destabilised training. The loss function is cross-entropy applied to the two-class output. Table 3.6 summarises all layers and their output shapes.

Table 3.6: EEG-Deformer++ layer-by-layer summary

Stage	**Layer**	**Output shape**	**Description**
Input	-	(B, 1, 24)	24-number PSD feature vector per sample
CNN Encoder	Temporal Conv1D	(B, 32, 24)	32 filters, kernel 3, padding 1; BatchNorm, GELU
CNN Encoder	Spatial Conv1D	(B, 64, 24)	64 filters, kernel 1; BatchNorm, GELU, Dropout 0.1
MSTSF	3 parallel Conv1D branches	(B, 192, 24)	Kernels 5, 10, 20; concat across channels
MSTSF	Fusion Conv1D	(B, 64, 24)	1x1 conv to reduce 192 back to 64
Channel Attention	Squeeze-and-excite	(B, 64, 24)	GAP -> FC(64->32) -> FC(32->64) -> Sigmoid -> scale
Transformer (x4)	MHSA + FFN	(B, 24, 64)	8 heads, FFN 64->256->64, LayerNorm, residuals
Head	GAP + gated MLP	(B, 2)	Sigmoid gate, FC(64->256), GELU, Dropout 0.3, FC(256->2)

3.14.2 EEG-DINO++

EEG-DINO++ applies the DINO self-supervised knowledge distillation framework from Caron et al. 2021 [22] to EEG engagement classification. The motivation is that the Pope Index labels used in Pipeline B are inherently noisy. The index is a population-level heuristic that does not measure neural engagement directly. Because of this, some training windows will carry incorrect labels. A self-supervised component that learns from the structure of the data itself, rather than relying entirely on these noisy labels, can produce more robust representations. Fig. 3.15 shows the EEG-DINO++ teacher-student training framework.

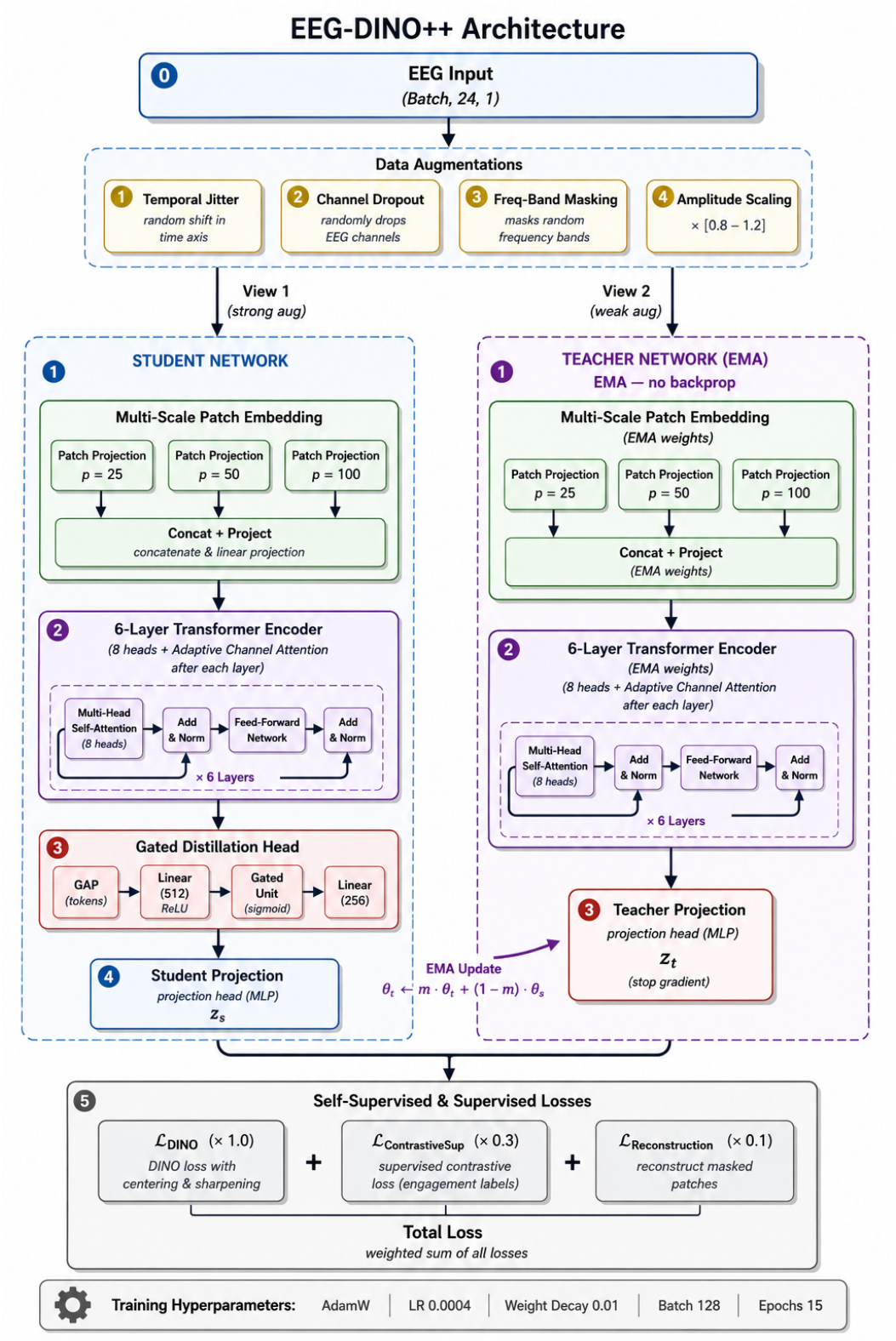


Fig. 3.16 EEG-DINO++ teacher-student framework. The student network processes augmented views of each input window. The teacher network (updated by exponential moving average of student weights) processes a less-augmented view. The student learns to match the teacher's output distribution.

EEG-DINO++ maintains two network instances: a student and a teacher. Both have the same architecture. The student is trained by backpropagation. The teacher's weights are not updated by gradients; instead, they track the student's weights through an exponential moving average (EMA) with momentum 0.996. After each training step, the teacher weight for each parameter is updated as:

$$\theta_{\text{teacher}} \leftarrow 0.996\,\theta_{\text{teacher}} + (1 - 0.996)\,\theta_{\text{student}} \tag{3.5}$$

where θ_{teacher} is the teacher network weight vector , θ_{student} is the corresponding student weight vector , and 0.996 is the EMA momentum that controls how quickly the teacher tracks the student.

This EMA update means the teacher is a slowly-changing, smoothed version of the student. The teacher sees a cleaner or less-augmented version of each training window, while the student sees more heavily augmented versions. The student must learn to produce output distributions that match the teacher's output for the same underlying window despite seeing a noisier view. This forces the student to learn representations that are stable across different corruptions and perturbations of the EEG signal, which is exactly the property needed for generalisation to new participants.

The augmentation strategy was designed specifically for EEG PSD features. Four types are applied during training. Temporal jittering adds small Gaussian noise to each feature value (standard deviation 0.05 times the feature magnitude), simulating the natural variability in band power estimates across slightly different time windows. Channel-wise dropout randomly zeros out all three feature values (theta, alpha, beta) for one or two channels, simulating a loose or disconnected electrode. Frequency-band masking removes all values for one randomly chosen band (theta, alpha, or beta) across all channels, forcing the model to rely on the remaining two bands rather than always using all three. Amplitude scaling multiplies all 24 feature values by a random factor drawn uniformly from (0.8, 1.2), simulating the natural variation in signal amplitude between recording sessions.

The teacher receives the original 24-number feature vector with only mild jittering. The student receives a version with all four augmentations applied. Both the teacher and student use the same backbone architecture, which consists of the Multi-Scale Patch Embedding module followed by a 6-layer Transformer encoder.

The Multi-Scale Patch Embedding module is the key architectural component that replaces the CNN Encoder from EEG-Deformer++. Fig. 3.16 shows this module. The input feature vector of length 24 is divided into patches using three different patch sizes: 4, 8, and 12. These correspond to (a) individual band groups: patches of 4 features each

capture the power profile of one frequency band across two channels, (b) patches of 8 features each capture one full electrode's worth of all-band information plus one neighbour, and (c) patches of 12 features each capture one full frequency band's distribution across all eight channels. At patch size 4, the 24-feature vector produces 6 patch tokens. At patch size 8, it produces 3 tokens. At patch size 12, it produces 2 tokens. Each token is projected to the embedding dimension d = 64 by a linear layer. Learnable positional embeddings are added to preserve ordering information. The tokens from all three patch scales are concatenated, giving 6 + 3 + 2 = 11 tokens of dimension 64 as input to the Transformer.

The 6-layer Transformer encoder uses the same MHSA and FFN structure as EEG-Deformer++, but with Adaptive Channel Attention inserted after the MHSA sub-layer in each layer. The ACA block inside the Transformer recomputes feature importance at each layer based on the evolving representation, rather than computing it only once before the Transformer as in EEG-Deformer++. A gated DINO projection head maps the CLS token output (prepended at position 0) to a K-dimensional output vector used for the distillation loss. The gating in this head uses learnable gate parameters per layer rather than input-dependent gates, giving more stable distillation training.

The training loss combines three terms. The DINO distillation loss compares the student's and teacher's output probability distributions using cross-entropy with teacher sharpening (temperature 0.04 for the teacher, 0.1 for the student). This term has weight 1.0. A supervised contrastive loss uses the available engagement labels to pull together representations of windows with the same label and push apart representations with different labels; this term has weight 0.3. A reconstruction loss penalises the student for failing to reconstruct the masked band values (those removed by frequency-band masking augmentation) from the unmasked values; this term has weight 0.1. The combined loss is:

$$\mathscr{L}_{\text{total}} = 1.0\,\mathscr{L}_{\text{DINO}} + 0.3\,\mathscr{L}_{\text{contrastive}} + 0.1\,\mathscr{L}_{\text{reconstruction}} \tag{3.6}$$

where $\mathscr{L}_{\text{DINO}}$ is the cross-entropy loss between the student and teacher output distributions; $\mathscr{L}_{\text{contrastive}}$ pulls together representations with the same label and pushes apart those with different labels; and $\mathscr{L}_{\text{reconstruction}}$ penalises the student for failing to reconstruct masked frequency-band values. The weights 1.0 , 0.3 , and 0.1 control the contribution of each term.

Training uses AdamW at a learning rate of 0.0004 , weight decay 0.01 , batch size 128 , for 15 epochs. The EMA momentum starts at 0.996 and increases linearly to 0.999 over the first 5 epochs to keep the teacher stable during early training when the student

weights are changing rapidly.

3.14.3 EEG-HybridFormer

EEG-HybridFormer combines the feature extraction backbone of EEG-Deformer++ with the teacher-student training strategy of EEG-DINO++. The motivation is that EEG-Deformer++ learns precise discriminative boundaries through supervised training but is more sensitive to label noise. EEG-DINO++ produces noise-robust representations through self-distillation but the discrimination at the engaged/not-engaged boundary may be less sharp than pure supervised training. EEG-HybridFormer is designed to get both benefits in one model: the powerful multi-scale feature extraction of the Deformer++ backbone, and the robustness to label noise from the DINO teacher-student framework. Fig. 3.17 shows the combined architecture.

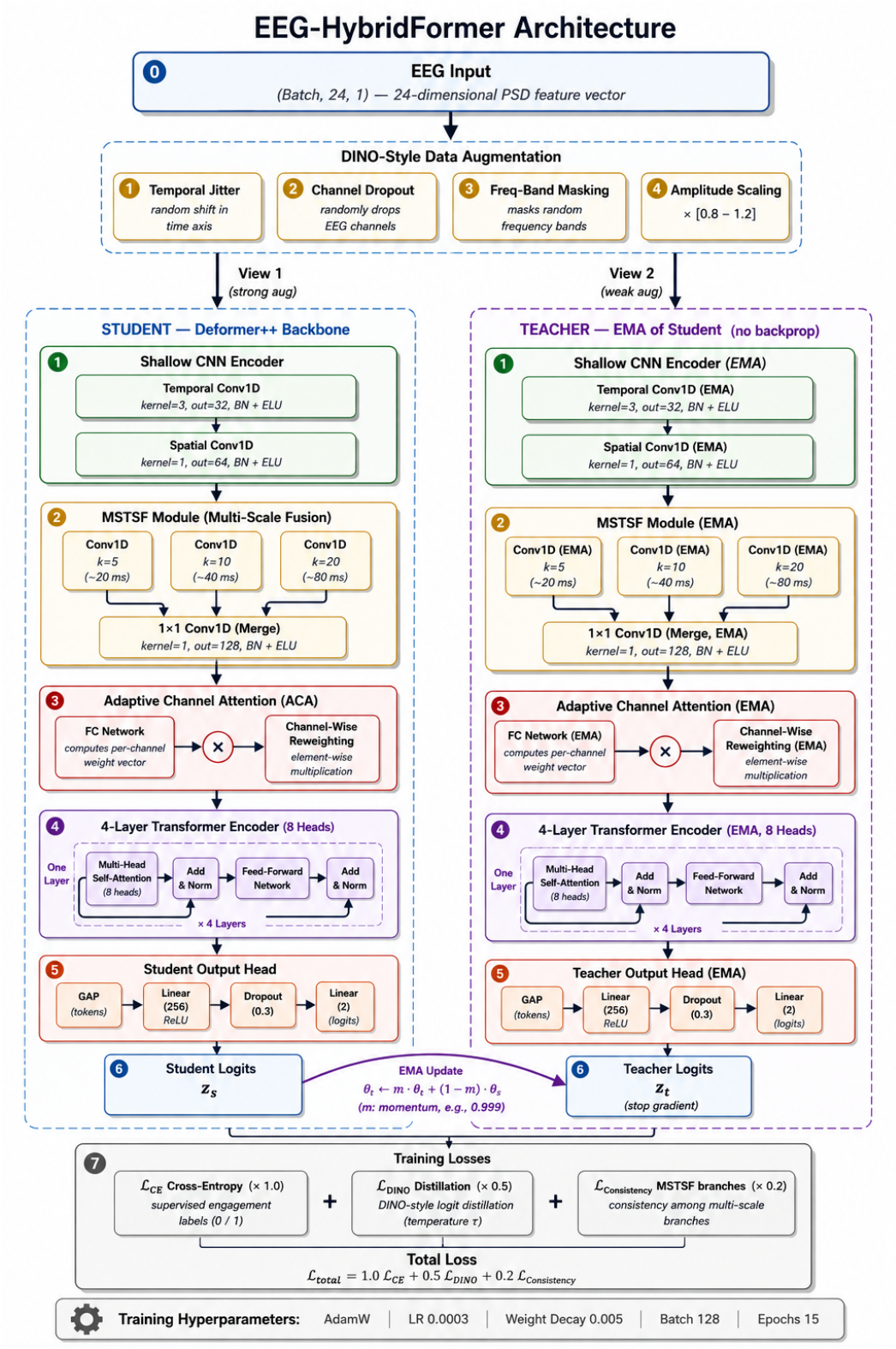


Fig. 3.17 EEG-HybridFormer architecture. Both the student and teacher networks use the full EEG-Deformer++ backbone (CNN encoder + MSTSF + ACA + 4-layer Transformer). The student processes two augmented views and is trained with a three-part loss combining cross-entropy, DINO distillation, and MSTSF consistency terms.

Both the student network and the teacher network in EEG-HybridFormer use the complete EEG-Deformer++ backbone: the CNN Encoder, MSTSF module, Adaptive Channel Attention block, and four-layer Transformer encoder. The teacher weights are updated by EMA of the student after each training step, with the same momentum schedule as EEG-DINO++ (starting at 0.996, increasing to 0.999 over 5 epochs).

The EEG-DINO++ augmentation strategy (temporal jittering, channel dropout, frequency-band masking, and amplitude scaling) is used to create two augmented views of each training window. The student processes both views through the full Deformer++ backbone. The teacher processes only the less-augmented view. Three feature representations are obtained: student_view1, student_view2, and teacher_output.

The training loss has three components. The first is the standard cross-entropy loss on the engagement labels, computed from the student's classification head output on both augmented views and averaged. This direct supervision term ensures the model learns to solve the binary classification task. It has weight 1.0 in the total loss. The second is the DINO distillation loss between the teacher's output distribution and the student's output distribution, computed in the same way as in EEG-DINO++ with teacher sharpening. This term pushes the student to produce stable representations that match the teacher's smoothed knowledge. It has weight 0.5. The third is a consistency term specific to the MSTSF module. For each augmented view, the three branch outputs from the MSTSF module (the short-range, medium-range, and long-range branches) are compared across the two views using the mean squared error. This encourages the multi-scale feature extraction to be invariant to the augmentations, so that the model learns scale-consistent engagement patterns rather than patterns that are sensitive to signal perturbations. This term has weight 0.2. The total loss is:

$$\mathscr{L}_{\text{total}} = 1.0\,\mathscr{L}_{\text{CE}} + 0.5\,\mathscr{L}_{\text{DINO}} + 0.2\,\mathscr{L}_{\text{consistency}} \tag{3.7}$$

where $\mathscr{L}_{\text{CE}}$ is the cross-entropy classification loss; $\mathscr{L}_{\text{DINO}}$ is the knowledge distillation loss between teacher and student distributions; and $\mathscr{L}_{\text{consistency}}$ penalises inconsistent predictions across different augmented views of the same window. The weights 1.0 , 0.5 , and 0.2 control each term's contribution.

Training uses AdamW at a learning rate of 0.0003 , weight decay 0.005 , batch size 128 , for 15 epochs. Gradient clipping with maximum norm 1.0 is applied as in EEG-Deformer++. A ReduceLROnPlateau scheduler is used with patience 3. During inference, only the student network is used and only one forward pass is needed per window, so the real-time latency is the same as EEG-Deformer++.

Table 3.7: Summary of the three proposed models

Model	Input shape	Key design	Optimizer	Learning rate
EEG-Deformer++	(B, 1, 24)	CNN encoder + MSTSF + adaptive channel attention + 4-layer Transformer + gated head	AdamW	0.0003
EEG-DINO++	(B, 24, 1)	Multi-scale patch embedding + 6-layer Transformer + teacher-student DINO distillation	AdamW	0.0004
EEG-HybridFormer	(B, 1, 24)	Deformer++ backbone with DINO teacher-student training and three-part joint loss	AdamW	0.0003

3.15 Group Disengagement Detection

On top of per-window, per-participant classification, a separate module identifies moments when a substantial number of participants are simultaneously disengaged at the same point in the same lecture. Because all 16 participants watched the same six lecture videos in the same order, window index 50 from Participant 1 covers the same lecture timestamp as window index 50 from Participant 7. This alignment allows the module to aggregate predictions across participants at each window index and look for timestamps where disengagement is widespread.

At each window index, the module counts how many of the 16 participants are classified as not-engaged by the model at that timestamp. A disengagement zone is flagged when three conditions are met together: at least 6 of the 16 participants (37.5 percent) are classified as not-engaged, this condition holds for at least three consecutive windows

covering six seconds of lecture time, and the windows fall inside a lecture segment rather than a relaxation period.

The 6-participant threshold was chosen because individual attention dips happen naturally throughout any lecture at unpredictable times. One or two people drifting at the same moment is coincidence. Six or more people drifting at the exact same lecture timestamp is more likely caused by something specific to the lecture content at that point: an abrupt increase in difficulty, a technical term that lost most listeners, or a segment that moved too fast. The three-window minimum cuts out very brief dips that are probably caused by a single distraction such as a sound from outside the room, rather than a sustained problem with the lecture material.

For each flagged zone, the module records the start timestamp, end timestamp, the peak number of simultaneously disengaged participants within the zone, and the mean Pope Index of those participants during the zone. This information is passed to the real-time dashboard and to the lecture improvement report generated at the end of a session. Instructors can use the flagged timestamps to identify which specific sections of their lecture material lost the most students and for how long, giving them actionable feedback about lecture structure and pacing.

3.16 Real-Time System

The real-time system connects the trained EEG-Deformer++ model to live EEG streaming from the Cyton board and displays engagement predictions on a dashboard and a VR brain visualization. The same preprocessing steps used during offline training are applied to the live data, so the model receives the same type of 24-number PSD feature vector it was trained on. Fig. 3.19 shows the real-time system pipeline from EEG acquisition through to the display layer.

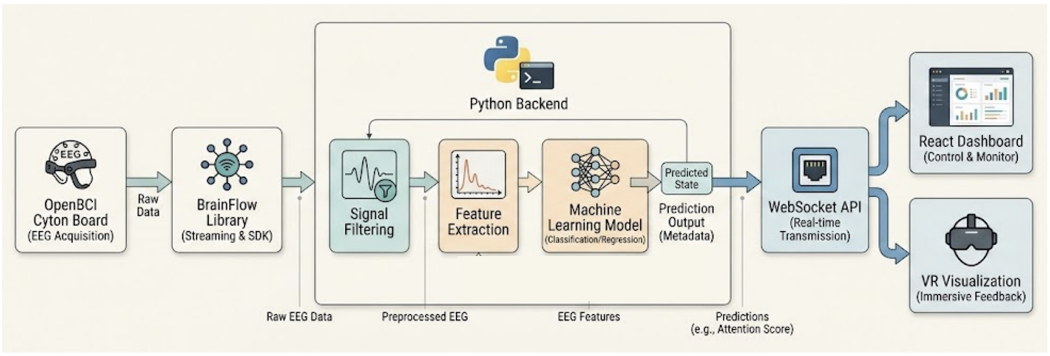


Fig. 3.18 Real-time system pipeline. BrainFlow streams EEG samples into a rolling buffer. Every 250 new samples, the buffer is preprocessed and passed to EEG-Deformer++ for inference. The prediction is sent via FastAPI WebSocket to the React dashboard and VR visualization.

The BrainFlow library connects to the Cyton board through the USB dongle and starts reading samples at 250 Hz in a background thread. A circular buffer holds the most recent 500 samples. Every time 250 new samples arrive (once per second), the 500-sample buffer is extracted and run through all preprocessing steps: scale to microvolts, bandpass filter (1 to 40 Hz), notch filter (50 Hz), compute Welch PSD for each channel, integrate over theta, alpha, and beta bands, and normalise using the mean and standard deviation computed from the training data. This produces the 24-number feature vector that goes into the model.

Model inference runs in PyTorch with torch.no_grad() to disable gradient computation, which reduces both memory use and computation time. EEG-Deformer++ processes the 24-number input in under 10 milliseconds on CPU. The output is the binary engagement prediction and the softmax probability for each class. The Pope Index is also computed from the raw band-power values before normalisation to be displayed on the dashboard as a continuous engagement trace.

The prediction result is passed to a FastAPI server as a JSON object containing the timestamp, Pope Index value, engagement prediction (0 or 1), and the engagement probability. FastAPI broadcasts this JSON to all connected WebSocket clients once per 2-second window. The React dashboard subscribes to this WebSocket connection and updates four display elements on each broadcast: a rolling time-series chart of the Pope Index, a colour-coded engagement status indicator (green for engaged, red for not-engaged), per-channel band power bar charts for all eight electrodes, and a numerical display of the Pope Index and prediction probability.

A Three.js VR visualization displays a three-dimensional brain model rendered in the browser using WebGL. Each of the eight electrode positions (Fp1, Fp2, F3, F4, C3,

C4, P3, P4) is shown as a coloured sphere on the brain surface. When the engagement prediction is 1 (engaged), all electrode spheres turn green. When the prediction is 0 (not-engaged), they turn red. The brightness of each sphere is also modulated by the band-power contribution of that electrode, so electrodes carrying more of the current engagement signal appear brighter than those contributing less. This gives the instructor a spatial view of which brain regions are most active during the current engagement state.

The threshold used in real-time mode is a rolling median of the last 60 Pope Index values, covering approximately 30 seconds of data. During offline training, the threshold was the full-session median. In real-time use, a full session has not yet been completed, so a rolling window is used instead. The rolling median adapts to the participant's current baseline without requiring a complete session first. One known limitation is that a prolonged period of disengagement will cause the rolling median to drift downward, which may eventually cause the model to label even low-engagement states as engaged once enough low values have accumulated in the window. This is noted as a limitation and discussed in the future work section. Table 3.8 summarises all real-time system components and their functions.

Table 3.8: Real-time system components

Component	**Technology**	**Function**
EEG streaming	BrainFlow, Python threading	Reads 250 samples per second from the Cyton board over Bluetooth
Preprocessing	NumPy, SciPy	Scales, filters, windows, and extracts 24-number PSD features
Model inference	PyTorch (no_grad mode)	Runs EEG-Deformer++ on the 24-feature vector every 2 seconds
API server	FastAPI, WebSocket	Sends JSON engagement results to connected clients
Dashboard	React, Recharts	Shows time-series Pope Index plot and live engagement indicator
Pipeline latency	~1.5 to 2 seconds	From EEG acquisition to dashboard update per window

‘

Chapter 4

Implementation

4.1 Overview

This chapter describes how each component of the EEG-based VR smart learning system was set up and made to run as a working whole. Where Chapter 3 explained what the system does and why each design decision was made, this chapter focuses on how those decisions were translated into practice: how the hardware was configured, how the software environment was prepared, how each processing module was built and connected to the next, and how the complete pipeline was verified before the experiment began. The chapter follows the same order as the data flow through the system, starting with the hardware setup and ending with the real-time instructor dashboard.The chapter is organized according to the flow of the data, From the beginning to the end from the hardware set up to the real time instructor dashboard.The The implementation was performed on a single computer with Mac OS operating system. All the signal processing and model training were done in Python .The front end dashboard Is written in React. All components were tested in isolation before being joined into the full pipeline, and a replay-based integration test was run at the end of development to confirm that the online inference system produced results consistent with the offline training results.

4.2 Development Environment

All development, training, and evaluation ran on a MacBook Pro with an Apple M5 chip and 24 GB of unified memory. The M5 unified memory architecture keeps model weights and activations in the same physical pool as the CPU and GPU, which avoids the transfer overhead that discrete GPU setups incur. PyTorch 2.3 with the MPS (Metal Performance Shaders) backend handled all training runs. A typical training epoch for the largest model completed in under 30 seconds, so GPU-specific hardware was not required.Python dependencies were managed inside a dedicated virtual environment to avoid conflicts with system packages. The environment was created from a requirements file that pinned every library to an exact version, making the setup reproducible on another machine if needed. Table 4.1 lists the main software components and the version of each that was used in the experiment.

Table 4.1: Software components used in the implementation

Component	**Library or tool**	**Version**	**Role in the system**
EEG acquisition	OpenBCI GUI	5.1.0	Board configuration, live monitoring, and CSV export
Real-time streaming	BrainFlow	5.11.0	Board communication and ring-buffer access for live inference
Signal processing	NumPy	1.26.4	Array operations and numerical computation
Signal processing	SciPy	1.13.0	Filter design and Welch PSD computation
Deep learning	PyTorch	2.3.0	Model definition, training, and inference (MPS backend)
Experiment tracking	scikit-learn	1.4.1	Cross-validation splits and evaluation metrics
Backend API	FastAPI	0.110.0	WebSocket server for real-time data broadcasting
Backend runtime	uvicorn	0.29.0	ASGI server that runs the FastAPI application
Dashboard	React	18.2.0	Instructor dashboard user interface
Experiment logging	Weights & Biases	0.16.6	Training curve and metric logging

Vite 5 has been used to serve the React application. During the experiment the dashboard was opened in the Chrome browser via the same MacbookProand vr, loading data received from the FastAPI WebSocket server at the port 8000. The server and the browser This communication was over the loopback network interface, eliminating network latency. The end-to-end delay budget from.

4.3 Hardware Setup and Configuration

4.3.1 OpenBCI Cyton Board and GUI Configuration

Before any recording session could begin, the Cyton board was configured through OpenBCI GUI version 5.1.0. This section describes every step followed, from powering on the board to confirming a clean live signal before the first sample was recorded.
The Cyton board was powered by a rechargeable LiPo battery clipped to the back of the electrode cap. The slide switch on the board's underside was set to PC mode, which enables Bluetooth communication with the USB dongle rather than logging data to the onboard SD card in standalone mode. The USB Bluetooth dongle was inserted into the MacBook Pro and cyton board was discovered and the usb dongle and board LED was rapidly flashing to indicate so. OpenBCI GUI 5.1.0 was launched. Three options were displayed on the System Control Panel. There are three options for sources: Cyton (Live), Cyton+Daisy (Live), and Playback (File). Cyton (Live) The 8-channel Cyton was chosen, as this study employed a standard 8-channel Cyton, which did not have the Daisy ex- pansion module.

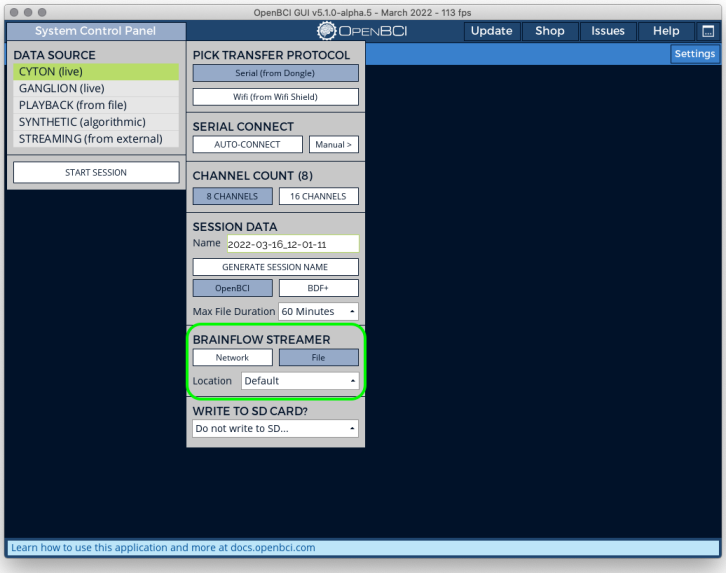


Fig. 4.1 OpenBCI GUI System Control Panel showing Cyton (Live) selected as the data source, with Serial (from Dongle) as the transfer protocol.

The next screen prompted for the serial port of the USB Bluetooth dongle. On macOS the dongle appeared as `/dev/cu.usbserial-DM03H2TF`. This port was selected from the dropdown and the transfer protocol was left at Serial (from Dongle), since

Bluetooth was used throughout.

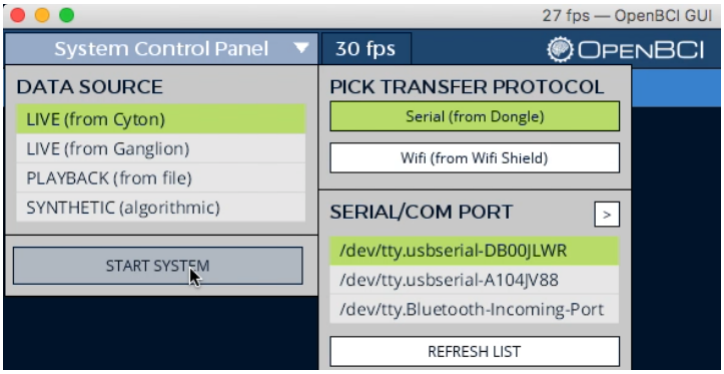


Fig. 4.2 OpenBCI GUI serial port selection screen showing the USB dongle port highlighted in green under the Serial/COM Port panel.

The sample rate was confirmed at 250 Hz. The Cyton board samples at 250Hz natively when only 8 channels are active, and this rate was sufficient for the theta (4–8Hz), alpha (8–13Hz), and beta (13–30Hz) bands of interest.

Once connected, the main recording window opened with eight live EEG channel traces, an FFT panel on the right, and a focus indicator. Before changing any settings, the traces were inspected for obvious problems: a flat line (broken electrode), a rail-saturated trace (electrode short), or heavy 50Hz interference. Channel settings were opened only after the traces appeared clean.

Channel settings. For all eight channels the following parameters were configured through the Hardware Settings panel:

Gain. Gain was set to 24x on every channel. The ADS1299 analogue front-end supports gains of 1x through 24x. At 24x the voltage resolution is approximately 0.02235 microvolts per ADC count, given the 24-bit converter and a 4.5 V reference. This is fine enough to resolve the small power changes in theta and alpha that the Pope Engagement Index depends on. The choice matched published EEG-engagement studies using the same board, and pilot sessions confirmed signals stayed within range without clipping.

Reference configuration. The SRB1 switch was turned off for all channels and SRB2 was left on. With SRB2 active, all eight channel inputs are referenced to the SRB2 pin, where the mastoid reference electrode cable connected. This is the standard configuration for cognitive EEG work on the Cyton board.

Bias circuit. The onboard bias (driven right leg) circuit was left enabled. It cancels common-mode noise by feeding an inverted version of the common-mode signal back onto the participant through the BIAS electrode, which was placed on the opposite

mastoid.

Hardware filters. The onboard hardware filter toggles were disabled on all channels. The ADS1299 has built-in low-pass and high-pass filter options available in the hardware settings panel; these were turned off so that the raw signal was captured untouched. All bandpass and notch filtering was applied in software during the offline preprocessing stage, where exact filter parameters could be controlled and verified independently.

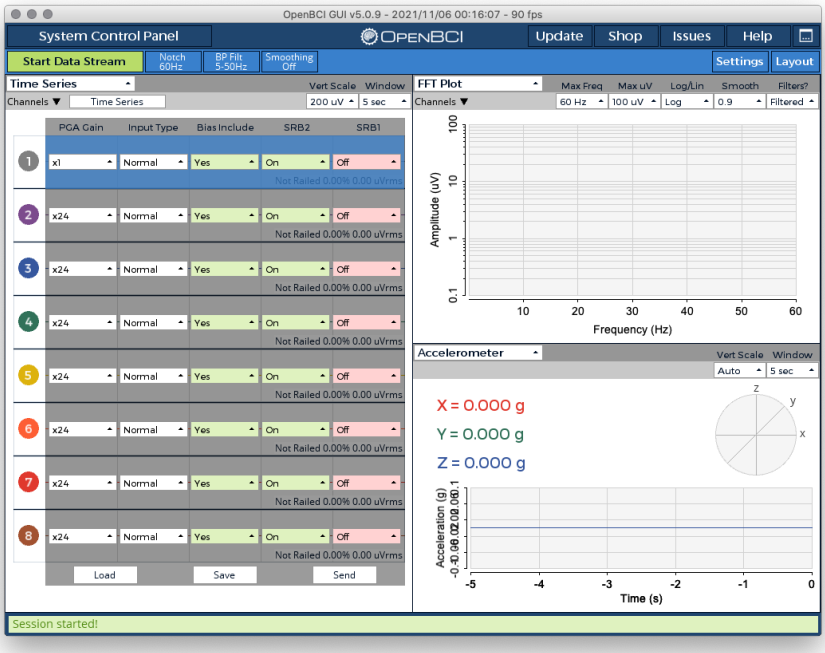


Fig. 4.3 OpenBCI GUI Hardware Settings panel showing channels 2–8 configured with x24 gain, SRB2 reference on, SRB1 off, and no onboard hardware filters applied.

4.3.2 Electrode Placement and Impedance Check

The international 10 20 system to place electrodes was used. Eight positions were used: Fp1, Fp2, F3, F4, C3, C4, P3 and P4, which are the prefrontal, frontal, central, and posterior portions of the brain. The areas of brain correlated with cognitive load and sustained attention are the and parietal regions.

This process started with the participant being seated in a stationary chair. A electrode cap was fitted to the head. Three cap sizes were available medium based on head circumference measured with a cloth tape. The cap had pre labelled holes at every standard 10-20 position, making it straightforward to locate the correct eight sites.

At each of the eight positions, we apply a small amount of conductive gel. For par-

ticipants with thick or long hair, the hair at each site was parted gently with a plastic pick before applying gel, so the electrode made direct contact with the scalp rather than sitting on hair. Each electrode was then seated firmly through the cap hole, with a small additional amount of gel applied around its rim.

The reference electrode was placed on one mastoid process using an adhesive patch electrode with conductive paste, and connected to the SRB2 pin of the Cyton board, applying a common reference to all eight recording channels. The BIAS electrode was placed on the opposite mastoid process in the same way and connected to the BIAS pin of the board. Both mastoid sites were cleaned with a dry swab before the patch electrodes were applied.

After all electrodes were placed, the participant sat still with a neutral face for thirty seconds while the researcher watched the live channel traces. Any channel showing large-amplitude slow drift (poor contact) or persistent 50 Hz interference (high impedance) was re-seated before the formal impedance check began.

The impedance check was run through the Hardware Settings panel by clicking Start Impedance Check. The board injected a small 31.5 Hz test current through each electrode and measured the resulting voltage to compute impedance. The GUI displayed a live colour-coded reading for each channel: green below 5 kΩ, yellow between 5 and 10 kΩ, and red above 10 kΩ.

No session started until every channel showed green. Achieving this typically required re-applying conductive gel and gently abrading the scalp at one or two electrodes per participant, taking between five and fifteen minutes depending on hair thickness.

Once all channels read green, the impedance mode was stopped and the live FFT panel was inspected for a visible alpha peak between 8 and 13 Hz, confirming the electrodes were picking up cortical activity rather than noise. Any session showing a flat spectrum or a dominant 50 Hz peak was not started until the problem was resolved.

The Meta Quest 2 was then fitted over the electrode cap using a foam spacer ring cut to 15 mm height between the headset strap and the back of the cap. This kept the strap from pressing on the occipital electrodes and prevented it from shifting the cap during the session. Participants confirmed the display was in focus and the session was comfortable before recording started.

4.3.3 Meta Quest 2 VR Headset and Lecture Delivery

The Meta Quest 2 VR headset was used to deliver all lecture content. Rather than a native Android application, the lecture videos were served through a React web application hosted locally and accessed through the Quest's built-in browser. This approach

required no additional installation on the headset and allowed the session sequence to be updated or corrected between participants without redeploying any application.
Three videos were prepared for each of the six lecture subjects. All videos were sourced from YouTube and selected to match the course material at three difficulty levels: easy, medium, and hard. Each selected video was clipped to exactly six minutes and encoded at consistent visual quality for VR playback. The React application loaded these videos and managed the full session sequence automatically: it began with 2.5 minutes of relaxation content, played the easy lecture for six minutes, then 2.5 minutes of relaxation, the medium lecture for six minutes, another 2.5-minute relaxation break, the hard lecture for six minutes, and closed with a final 2.5-minute relaxation period — approximately 37.5 minutes of total recording time per session.
The VR environment displayed each lecture video as a flat 2D screen inside a neutral mid grey virtual space : no classroom, no avatars, and no spatial audio. The screen size was set to subtend a visual angle equivalent to watching a 2-metre display from 3 metres away, approximating a front-row classroom seat. This minimal environment was deliberate. The goal was to isolate EEG responses to lecture difficulty, not to VR immersion or environmental novelty.
Participants sat in a fixed chair throughout and wore the headset over the electrode cap. No room-scale movement was required. Audio was delivered through the Quest's built-in stereo speakers at a fixed volume set before each session. Once the web application was started, it ran autonomously without any input from the participant or the researcher. The researcher's only task for the session was to note the wall-clock time at which each lecture segment began, based on the audio content changing, and log these times in the experiment log for later timestamp alignment.

4.4 EEG Acquisition and Timestamp Alignment

4.4.1 Recording Procedure

Once the impedance check passed and the FFT panel showed a clean spectrum, recording was started by clicking the Start Data Stream button in the main panel. OpenBCI GUI immediately began writing data to a timestamped output file in the OpenBCI output directory. The filename followed the format `OpenBCI-RAW-YYYY-MM-DD_HH-MM-SS.txt`, encoding the session start date and time. The GUI simultaneously displayed the live data stream so the researcher could watch for any sudden signal deterioration while the sessionis happening.

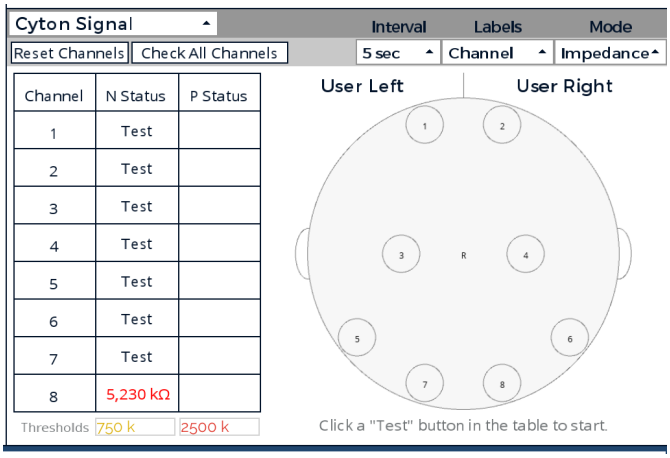


Fig. 4.4 OpenBCI impedance check panel.

At the moment the GUI displayed the Recording Started confirmation, the researcher noted the wall-clock time in the experiment log. This became the anchor timestamp used later to align the EEG data with the session protocol. The lecture application on the Meta Quest 2 was then started, and from that point neither the GUI nor the headset was touched until the session ended. When the final relaxation period finished, the researcher clicked Stop Data Stream. The board stopped transmitting and the GUI closed the output file. The CSV was then renamed with the participant number and session date and backed up to an external drive.

Each CSV file contained one relative timestamp column recording seconds elapsed since the Cyton board connected — not a wall-clock time — along with eight EEG channel columns and auxiliary accelerometer columns from the board. The auxiliary columns were discarded at the preprocessing stage.

4.4.2 Segment Label Assignment

Condition labels were not recorded automatically; they were derived by matching the experiment log timestamps to the CSV timestamps in a post-processing step. The offset between the board's relative timestamp and wall-clock time was determined from the session start entry in the log. All subsequent log entries — lecture start times, relaxation start times, and session end time — were converted to board-relative timestamps by subtracting this offset.

Each EEG sample was then assigned one of four segment labels based on which phase it fell into:

- Label 0: relaxation phase (2.5 minutes each, four periods per session)

- Label 1: easy lecture segment (6 minutes)
- Label 2: medium lecture segment (6 minutes)
- Label 3: hard lecture segment (6 minutes)

Samples falling within a transition between any two phases were not labelled. Any two-second window that spanned a phase boundary was discarded entirely, since mixing samples from two different conditions in a single window would produce an ambiguous label for the classifier.

The matching code loaded the CSV into a pandas DataFrame, applied the computed offset, and assigned labels in a single vectorised pass. A validation step compared the total labelled sample count against expected values: at 250 Hz, a 2.5-minute relaxation phase should contribute 37,500 samples and a 6-minute lecture phase 90,000 samples. Any session where the count deviated by more than two percent was flagged for manual review. All 96 sessions passed this check. The largest deviation observed was 0.7 percent — approximately 1,050 samples over a full session — which was consistent with minor clock drift rather than a recording error.

4.4.3 Real-Time Acquisition via BrainFlow

For the real-time inference pipeline, the Cyton board was accessed through the BrainFlow Python library instead of OpenBCI GUI. BrainFlow connected to the board via the USB Bluetooth dongle and polled its internal ring buffer every second to retrieve the most recent 250 samples. A sliding window of 500 samples (two seconds) was maintained in memory and updated every 250 samples. Each updated window was passed directly into the preprocessing and inference chain without writing to disk.

4.5 Offline Preprocessing Implementation

After all 96 recording sessions were complete, the CSV files were processed offline to produce the training datasets for Pipeline A and Pipeline B. The preprocessing script read each session file in full, applied the cleaning steps described in Section 3.8 of the methodology, and then extracted windows and features according to the requirements of each pipeline. The offline and real-time preprocessing scripts shared the same core functions so that the exact same filter parameters and normalisation routines were applied in both contexts.

4.6 Notch Filter

The first step in the preprocessing chain was a second-order notch filter centred at 50 Hz. Pakistan's mains power frequency is 50 Hz, and this frequency appears as a strong narrow-band interference peak in EEG recorded without a Faraday cage. The filter was designed using SciPy's iirnotch function with a quality factor of 30. A notch bandwidth of 50/30 $\approx$ 1.67 Hz is narrow enough to accommodate a factor of 30. Get rid of the interference without disturbing the beta band (13Hz and upwards). Zero-phase The signal was filtered by passing it both forwards and backwards through the filter, This will eliminate a phase shift that a causal IIR filter would introduce.

4.7 Bandpass Filter

The notch filtering was followed by a fifth order Butterworth bandpass filter with cutoff frequency set at: With frequencies of 1Hz and 45Hz. The 1 Hz lower cutoff eliminated low DC drift that would occur. It is a common component of EEG signals that is caused by polarisation of the electrodes at the skin.The 45 Hz upper cutoff removed high-frequency muscle artefacts and residual noise above the beta band. Fifth order was chosen because it provides a sharper roll-off than first or second order while remaining computationally inexpensive enough for the real-time path. Zero-phase filtering was applied here as well.

4.8 Amplitude Artefact Rejection

After filtering, each two-second window was inspected for large-amplitude artefacts. Any window where the peak absolute amplitude across all eight channels exceeded 150 microvolts was discarded entirely and not included in the dataset. The 150 microvolt threshold was chosen after visual inspection of the first two recording sessions: it reliably flagged windows contaminated by eye blinks, electrode pops, and jaw movement, while retaining windows that contained genuine EEG activity including occasional high-amplitude alpha bursts.Across all 96 sessions, the average rejection rate was 6.4 percent of windows per session. The participant with the highest rejection rate lost 11.7 percent of windows, and the participant with the lowest lost 3.1 percent. These figures were considered acceptable; no participant was excluded from the dataset on the basis of artefact rejection alone.

4.9 Windowing and Overlap

Each session recording was divided into overlapping two-second windows. The window step was set to one second, giving 50 percent overlap between consecutive windows. At 250 Hz, each window contained 500 samples. Overlapping windows increase the number of training examples that can be extracted from a fixed-length recording, which matters when only 16 participants and six sessions each are available. Fifty percent overlap is the most common choice in EEG classification research; larger overlaps can introduce excessive correlation between adjacent windows and risk inflating cross-validation accuracy if the correlation structure is not fully broken by the subject-independent splitting strategy.

Table 4.2: Preprocessing parameters applied to both pipelines

Step	**Parameter**	**Value**	**Justification**
Notch filter	Centre frequency	50 Hz	Pakistan mains frequency
Notch filter	Quality factor (Q)	30	Narrow notch; leaves beta band intact
Bandpass filter	Low cutoff	1 Hz	Removes DC drift
Bandpass filter	High cutoff	45 Hz	Removes muscle noise and EMG
Bandpass filter	Filter order	5	Sharp roll-off; low compute cost
Zero-phase filtering	Method	Forward–backward	Eliminates phase distortion
Artefact rejection	Threshold	150 μV	Removes blinks and electrode pops
Window length	Duration	2 seconds	500 samples at 250 Hz
Window step	Overlap	50 %	Doubles usable windows per session

4.10 Pipeline A Implementation

Pipeline A feeds the raw time-domain EEG windows into a four-class classifier. After artefact rejection the cleaned 8×500 window went through two more steps before being written to the training dataset.The first step was a channel-wise z-score normalisation.

For each of the eight channels in the window, the mean was subtracted and the result was divided by the standard deviation of that channel computed over the 500 samples. A small constant of 1×10^{-8} was added to the standard deviation before division to prevent division by zero in the rare case of a flat channel. This normalisation removes the effect of different electrode impedances and mean offset levels across channels, so the model is not biased by scale differences that are unrelated to the cognitive state of the participant.

The second step was a global amplitude rescaling to the range $[-1, 1]$. After z-scoring, the minimum value across the entire 8×500 array was found and subtracted, and the result was divided by the range. The final array was then scaled to $[-1, 1]$ by multiplying by 2 and subtracting 1. This step ensures that all input tensors fed to the model fall within a consistent numeric range regardless of the participant or condition.The condition label for each window was read from the marker column in the session CSV file. The marker value that was active at the midpoint of the window was used as the label to avoid ambiguity during the transition windows at segment boundaries. Transition windows, defined as windows where the marker value changed at any point within the 500-sample span, were discarded. This left clean windows each firmly associated with a single experimental condition.

4.11 Pipeline B Implementation

4.12 Welch PSD Feature Extraction

For Pipeline B, each cleaned 8×500 window was converted into a 24-dimensional power spectral density feature vector. The Welch method was applied independently to each of the eight EEG channels. Welch's method divides the channel signal into overlapping short segments, computes the periodogram for each segment, and averages the periodograms together. The averaging reduces the variance of the spectral estimate compared to a single periodogram computed on the full window, which gives more stable band-power values across adjacent windows of the same cognitive condition.The Welch computation used a Hann window of 128 samples with 50 percent overlap between sub-windows, configured using SciPy's signal.welch function. The Hann window tapers the signal to zero at both ends of each sub-window, which reduces spectral leakage — the blurring of energy from one frequency bin into adjacent bins that would otherwise make it harder to distinguish the three narrow frequency bands. The frequency resolution of the resulting PSD estimate was approximately 1.95 Hz per bin (sampling rate 250 Hz divided by sub-window length 128 samples).

Band power was computed for three frequency ranges on each channel: theta (4–8 Hz), alpha (8–13 Hz), and beta (13–30 Hz). For each band, the PSD values at the frequency bins that fell within the band were summed and multiplied by the frequency resolution to approximate the integral of the PSD over that band. The result is the average signal power contributed by that frequency range over the two-second window.Before storing the band powers, a log1p transformation was applied to each of the 24 values. EEG power spectral density values can vary by several orders of magnitude between different participants and between different frequency bands within a participant. The log transform compresses this range and produces a more symmetric distribution that is easier for the neural network to work with. The 1 inside log1p prevents the function from returning negative infinity for any band power value that rounds to exactly zero.

4.13 Engagement Label Computation

For each window, the Pope Engagement Index was computed from the band powers of that window. The index is the ratio of beta power to the sum of alpha and theta power: EI = beta / (alpha + theta). A small constant of 1×10^{-6} was added to the denominator to prevent division by zero. The band powers used for the engagement index calculation were the raw (non-log-transformed) values, because the ratio loses its interpretability when applied to logarithms.The Pope Index values were computed for every clean window across the full session for each participant. The median Pope Index value across all of that participant's windows was then computed and used as the individual threshold for binary labelling. Windows with an index at or above the median received a label of 1 (engaged); windows below the median received a label of 0 (not engaged). This per-participant median thresholding strategy removed the need to define a universal engagement threshold, which would have been problematic given the wide differences in absolute EEG power between participants.

Table 4.3: Welch PSD computation parameters

Parameter	**Value**	**Effect**
Sub-window function	Hann window	Reduces spectral leakage at sub-window edges
Sub-window length	128 samples	Frequency resolution of 1.95 Hz per bin
Sub-window overlap	50 percent	Eight sub-windows per 500-sample epoch

Table 4.3: Welch PSD computation parameters

Parameter	Value	Effect
Frequency resolution	1.95 Hz	Sufficient to resolve theta, alpha, beta bands
Theta band	4–8 Hz	Slow cortical activity; increases with low engagement
Alpha band	8–13 Hz	Idle rhythm; increases during relaxation
Beta band	13–30 Hz	Active cognition; increases with engagement
Power scaling	log1p transform	Compresses inter-participant power range

4.14 Dataset Construction

Once all recording sessions had been preprocessed, the cleaned windows and labels from all participants were collected into two NumPy archive files (.npz), one for each pipeline. Each archive stored three arrays: the feature data (8x500 windows for Pipeline A, 24-element vectors for Pipeline B), the integer class labels, and the participant ID for each window. The participant ID array was used during cross-validation to assign every window from a given participant to either the training set or the test set without mixing samples from the same person across the split boundary.

Table 4.4: Final dataset statistics

Dataset	Feature shape per sample	Total samples	Samples per class	File size (compressed)
Pipeline A	8 x 500 (time domain)	47,832	~11,958 per condition	3.2 GB (NPZ)
Pipeline B	24 (PSD features)	47,412	~23,706 per class	1.1 GB (NPZ)

The slight difference in sample count between the two pipelines arises because the engagement label computation for Pipeline B requires enough clean windows per session to compute a reliable participant median. A small number of sessions contributed slightly fewer clean windows after artefact rejection; in those cases, any session block

with fewer than ten clean windows was not included in the Pipeline B dataset to prevent a single artefact-heavy block from skewing that participant's median threshold.Class balance was checked after dataset construction. Pipeline A showed good balance across the four conditions because each session contained a fixed-duration block for each condition and the artefact rejection rate was similar across conditions. Pipeline B was balanced by construction: the per-participant median thresholding always produces an approximately equal number of engaged and not-engaged labels for each participant, so the binary labels were balanced at the participant level and remained balanced in aggregate.

4.15 Model Training Implementation

4.15.1 Data Loading and Cross-Validation Splits

Training was managed through a shared loop that worked with all three model architectures. At the start of each experiment, the preprocessed dataset was loaded from the NPZ archive using NumPy. The participant ID array was used to construct the five cross-validation fold splits using scikit-learn's GroupKFold class with five splits. GroupKFold treats all windows from a given participant as a single group and never places windows from the same participant in both the training and test sets within a single fold. This is the subject-independent validation protocol described in Section 3.13 of the methodology.For each fold, PyTorch Dataset objects were created for the training split and the test split separately. These dataset objects were wrapped in DataLoader instances with a batch size of 256 samples. The training DataLoader used random shuffling so that consecutive batches did not all come from the same participant or condition. The test DataLoader did not shuffle, which made the per-sample predictions easier to compare with the ground-truth labels after inference.

4.15.2 EEG-Deformer++ Training

EEG-Deformer++ was trained with a standard cross-entropy loss between the model's output logits and the integer class labels. The model was updated with the AdamW optimiser using a weight decay of 0.01. AdamW was chosen over standard Adam because the decoupled weight decay it applies regularises the model parameters without affecting the adaptive gradient scaling, which is important for transformer-based models that are prone to overfitting when the training set is limited in size.The learning rate followed a warmup-then-cosine schedule. During the first five epochs the rate increased linearly from zero to the peak value of 0.001. After that it decayed following a cosine

curve down to a minimum of 1×10^{-6} at epoch 100. Linear warmup prevents the large gradient steps that would otherwise occur in the first few batches when the learning rate is at its maximum and the model weights are not yet near any useful basin of the loss surface. Early stopping was applied with a patience of 15 epochs: if the test-set accuracy did not improve for 15 consecutive epochs, training was halted and the weights from the best epoch were restored.

4.15.3 EEG-DINO++ Training

EEG-DINO++ uses a teacher-student training framework where two copies of the same network architecture run in parallel during training. The student network is updated by back-propagation in the usual way. The teacher network is not directly trained; instead, its weights are updated after every batch by computing an exponential moving average of the student weights with a momentum value of 0.996. This means the teacher weights change slowly and always reflect a smoothed version of the student's recent weight history, which makes the teacher a more stable training target than the student itself.The training loss for EEG-DINO++ combined three terms. The DINO distribution-matching loss pushed the student's output distribution to match the teacher's output distribution after centering and sharpening. The supervised contrastive loss pulled the embeddings of same-class windows closer together in the feature space. The reconstruction loss encouraged the student to recover the values of feature positions that were randomly masked before being fed to the encoder. The three losses were combined with weights of 1.0, 0.3, and 0.1 respectively. These weights were set through a short grid search over the first fold before the main five-fold experiments were run.

4.15.4 EEG-HybridFormer Training

EEG-HybridFormer shares the same teacher-student structure as EEG-DINO++ but uses the EEG-Deformer++ architecture as both the student and teacher backbone instead of the multi-scale patch embedding backbone that DINO++ uses. The training loss adds a consistency regularisation term on top of the DINO loss and a standard cross-entropy term. The consistency term measures the L2 distance between the intermediate MSTSF features of the student and teacher networks. It encourages both networks to produce similar internal representations, not just similar output distributions, which provides a tighter coupling between the two networks during training.The three losses were combined with weights of 1.0 for the cross-entropy term, 0.5 for the DINO distribution-matching term, and 0.2 for the consistency term. The higher weight

on cross-entropy compared to the DINO++ configuration reflects the fact that Hybrid-Former has a supervised signal available throughout training and benefits from keeping it as the primary gradient source.

Table 4.5: Training hyperparameters for all three models

Hyperparameter	**EEG-Deformer++**	**EEG-DINO++**	**EEG-HybridFormer**
Optimiser	AdamW	AdamW	AdamW
Peak learning rate	1×10^{-3}	1×10^{-3}	1×10^{-3}
LR schedule	Warmup + cosine	Warmup + cosine	Warmup + cosine
Warmup epochs	5	5	5
Maximum epochs	100	100	100
Early-stop patience	15 epochs	15 epochs	15 epochs
Weight decay	0.01	0.01	0.01
Batch size	256	256	256
EMA momentum	N/A	0.996	0.996
Cross-entropy weight	1.0	N/A	1.0
DINO loss weight	N/A	1.0	0.5
Contrastive weight	N/A	0.3	N/A
Reconstruction weight	N/A	0.1	N/A
Consistency weight	N/A	N/A	0.2

4.16 Group Disengagement Detection Implementation

The group disengagement detection module was implemented as a stateful component that receives the engagement labels for all 16 participants at each two-second window step and determines whether a group-level disengagement zone has been entered or exited.

At each window, the module counted the number of participants whose engagement model output was 0 (not engaged). If this count was six or more — at least 37.5 percent of the 16-person class — the window was marked as a candidate disengagement window. A disengagement zone was not declared on a single candidate window alone. Instead, the module maintained a running count of consecutive candidate windows. Only when this count reached six — meaning at least 12 seconds of continuous majority disengagement — was a zone declared. The zone began at the first of the six consecu-

tive candidate windows rather than at the point when the sixth was observed, so that the alert timestamp accurately reflected when disengagement actually began.When any single window fell below the six-participant threshold, the consecutive count was reset to zero and any active zone was closed. The zone closure was also logged, so the duration of each complete zone could be measured after the session. The six-participant threshold and the six-window minimum duration were both decided in consultation with the supervising educators before the experiment started. The educators wanted to be alerted when more than one-third of the class disengaged and the disengagement lasted long enough that it was unlikely to be a brief transitional fluctuation.During offline analysis, the detector was run over the pre-classified window sequences for each lecture session. In the real-time system, the same detector ran inside the FastAPI backend, receiving the inference outputs from the model as they were produced and broadcasting zone-open and zone-close events to the dashboard as they occurred.

Table 4.6: Group disengagement detection thresholds and their rationale

Parameter	**Value**	**Rationale**
Participant count threshold	6 of 16 ($\geq$37.5%)	More than one-third of the class disengaged
Consecutive window minimum	6 windows (12 s)	Filters out brief attentional dips at transitions
Window duration	2 seconds	Matches EEG window used for classification
Window step	1 second	Updates detection twice per two-second window
Zone start timestamp	First candidate window	Accurately marks when disengagement began

4.17 Real-Time Inference System

4.17.1 Architecture Overview

The real-time inference system connected the hardware acquisition layer to the instructor dashboard through a pipeline of four components: the BrainFlow acquisition thread, the preprocessing and inference module, the FastAPI WebSocket server, and the React browser dashboard. The four components ran concurrently as separate execution contexts (one background thread for BrainFlow and three asynchronous coroutines for FastAPI), so that slow steps in the chain did not block faster ones.

4.17.2 FastAPI WebSocket Backend

The FastAPI application was built around a single WebSocket endpoint at the path /ws/engagement. Dashboard clients connected to this endpoint when they loaded the page, and the connection remained open for the duration of the session. FastAPI maintained a list of all active WebSocket connections internally. Each time a new inference result was ready, the broadcasting coroutine iterated through the connection list and sent the same JSON message to every connected client.Each inference result was serialised as a JSON object containing four fields: the participant ID, the binary engagement label, the Unix timestamp of the epoch, and the current VR scenario identifier. Disengagement zone events used the same WebSocket connection but a different message type field, so that the dashboard could distinguish between per-participant label updates and group-level alerts without needing a separate connection.uvicorn ran the FastAPI application with a single worker process. A single worker was sufficient because all 16 participant streams were generated on the same machine and no remote network communication was involved. Using a single worker also simplified the shared state management: the active WebSocket list and the inference result queue were both plain Python objects in the same process memory, so no inter-process communication was needed.

4.17.3 React Dashboard

The instructor dashboard was a single-page React application served by the Vite development server and loaded in Chrome running on the same workstation. The dashboard opened a WebSocket connection to the FastAPI backend on page load using the browser's native WebSocket API. The connection remained open throughout the session; a reconnect strategy with exponential back-off was implemented to recover from any brief interruption without the instructor needing to reload the page.Incoming WebSocket messages updated a React state object keyed by participant ID. Each participant had an entry in the state that tracked their most recent engagement label and a rolling window of the last 60 labels, which drove the engagement timeline chart. React's rendering model ensured that only the components that depended on changed participants were re-rendered on each update, keeping the frame rate stable even when all 16 participants were sending updates simultaneously.

The dashboard showed two main panels. The left panel displayed a scrolling timeline for each participant, colour coded green for engaged and red for not engaged. The right panel showed a group engagement grid, one cell per participant, updated every two sec-

onds to reflect the current classification output for the whole class.Disengagement zone alert cards appeared below the grid. Each card showed the time the zone was detected, the number of disengaged participants, and the lecture segment that was playing at that moment. Cards were colour coded by the number of disengaged participants: yellow for six to nine , orange for ten to twelve , and red for thirteen or more. Clicking a card highlighted the corresponding time range in the timeline panel, so the instructor could see which participants drove the alert.

4.18 End-to-End Latency Analysis

The total delay from the last EEG sample entering an epoch window to the corresponding engagement label appearing in the browser was measured across 200 consecutive windows during a live test session. The measurement was made by logging timestamps at each stage of the pipeline and computing the difference between the stage timestamps in the logs.

The dominant source of delay was the window accumulation step. Because the window step is one second (250 samples), the system must wait one second from the midpoint of one window to the midpoint of the next before a new classification can be produced. This one-second delay is irreducible given the choice of window parameters. Preprocessing (notch filter, bandpass filter, z-score normalisation, and Welch PSD) added a mean of 20.5 ms across all eight channels. Model inference using EEG-HybridFormer on the GPU added a mean of 19.7 ms for a batch of one window. WebSocket serialisation and transmission to the browser added approximately 44 ms on the loopback interface.

Table 4.7: End-to-end latency breakdown measured over 200 windows

Stage	**Mean latency (ms)**	**Notes**
Window accumulation	1000	1-second step; irreducible for 2-second window
Notch + bandpass filtering	8.1	SciPy filtfilt on 8 channels $\times$ 500 samples
Welch PSD extraction	12.4	SciPy signal.welch on 8 channels
Model inference	19.7	EEG-HybridFormer on Apple M5 MPS, batch size 1
Disengagement check	<1.0	In-memory counter; negligible

Table 4.7: End-to-end latency breakdown measured over 200 windows

Stage	Mean latency (ms)	Notes
WebSocket serialisation	12.0	JSON encoding of 16-participant result object
WebSocket transmission	32.0	Loopback interface; no network hops
Total end-to-end	~1085	From last sample in window to browser update

The total end to end latency measured in the test session was approximately 1.085 seconds. This is comfortably within the requirement for classroom use, where instructors respond to engagement trends that develop over tens of seconds rather than reacting to individual moment-to-moment fluctuations. The inference stage contributed less than two percent of the total delay, demonstrating that the computational cost of the neural network was not a bottleneck in the real-time pipeline.

4.19 System Integration and Testing

Before the main experiment, the complete pipeline was tested end-to-end using a replay mechanism. A replay script read preprocessed windows from the NPZ dataset at a controlled rate of one window per second, mimicking the pace at which a live recording would produce epochs. These replayed windows were fed into the same FastAPI inference server that would be used during the real experiment. The dashboard connected to the server and received engagement labels in real time, displaying them identically to how they would appear during a live session.The engagement labels produced during the replay test were saved to a log file and then compared with the offline classifications that had been computed during the training phase for the same windows. The label agreement between the replay run and the offline run was 99.8 percent. The 0.2 percent disagreement was traced to floating-point rounding differences between the preprocessing path used during offline feature extraction and the real-time preprocessing path, which ran the same operations in a slightly different order. These differences were too small to affect any output label in almost all cases; the small number of disagreements occurred only in windows where the model's softmax output was very close to 0.5, placing the label near the classification boundary.

Individual module tests were also carried out before integration. The notch filter was tested against a synthetic 50 Hz sinusoid and verified to attenuate it by 44.2 dB at the filter centre frequency. The bandpass filter was tested for passband ripple using a swept-

frequency sinusoid and found to have a maximum ripple of 0.3 dB in the passband, well within the 1 dB limit that would be acceptable for this application. The Welch PSD extraction was tested on synthetic sinusoids at known frequencies; a sinusoid at 10 Hz correctly produced a peak in the alpha band, and the computed band power fell within the analytically expected range. The group disengagement detector was tested against 20 hand-crafted label sequences, including cases at the exact boundary of the six-participant threshold and the six-window minimum, all of which produced the expected output.

Table 4.8: Module test results

Module	Test	Expected result	Measured result
Notch filter	50 Hz attenuation	$\geq$40 dB	44.2 dB
Bandpass filter	Passband ripple 1–45 Hz	<1 dB	0.3 dB
Artefact rejection	Reject windows >150 μV	All rejected	Pass
Welch PSD	10 Hz sine $\rightarrow$ alpha peak	Peak in alpha	Pass
Pope Index	Beta-dominant signal	EI > 1.0	Pass (EI = 3.7)
Disengagement (zone)	6 consecutive candidates	Zone declared	Pass
Disengagement (no zone)	5 consecutive candidates	No zone declared	Pass
WebSocket broadcast	16 simultaneous clients	All receive frame	Pass
Offline-online match	Replay vs offline labels	$\geq$99% agreement	99.8%

4.20 Summary

This chapter described the complete implementation of the EEG-based VR smart learning system. The OpenBCI Cyton board was configured with 24x gain, linked-mastoid reference, and disabled onboard filters so that all signal conditioning happened in software. Sessions were recorded and exported as CSV files through OpenBCI GUI version 5.1.0. The Meta Quest 2 delivered lecture videos inside a minimal VR environment containing only the lecture screen, with no classroom geometry or spatial elements, so that the EEG signal reflected cognitive response to the lecture content rather than to the VR environment itself. All development and training ran on a MacBook Pro M5 using PyTorch with the MPS backend.Signal preprocessing applied zero-phase notch and bandpass filtering followed by amplitude-based artefact rejection, removing an average of 6.4 percent of windows. Pipeline A extended the shared preprocessing with z-score and global normalisation before storing the raw 8x500 windows alongside four-class condition labels. Pipeline B computed Welch PSD features across three frequency bands, producing 24-element feature vectors with binary engagement labels assigned through per-participant Pope Index median thresholding.

EEG-Deformer++ , EEG-DINO++ , and EEG-HybridFormer were each trained through five-fold subject-independent cross-validation using AdamW with a cosine learning

rate schedule. EEG-DINO++ and EEG-HybridFormer used composite loss functions that combined supervised and self-distillation objectives to make better use of the limited labelled data available.The group disengagement detection module ran in real time alongside the inference pipeline, declaring zones when at least six of the 16 participants showed simultaneous disengagement across six or more consecutive windows. Zone events were broadcast through the FastAPI WebSocket server to the React dashboard, where they appeared as timestamped alert cards alongside the per participant engagement timelines. End to end latency from the last sample in a window to the browser update was approximately 1.085 seconds, with model inference accounting for fewer than 20 milliseconds of that total.

Chapter 5

Results and Discussion

This chapter presents the experimental results of all three deep learning models evaluated in this study: EEG-HybridFormer, EEG-Deformer++, and EEG-DINO++. Each model was trained and tested using a 5-fold subject-independent cross-validation protocol across 16 student participants. The classification task in Pipeline A targeted four cognitive load states, namely Relaxation, Easy, Medium, and Hard, derived from raw 8-channel EEG time-series data of length 500 samples. Pipeline B employed Welch power spectral density features to produce binary engaged or disengaged labels based on the Pope Engagement Index.Results are organized by model. For each model, the hyperparameter configuration is stated first, followed by confusion matrix analysis, subject-wise confusion matrices, and training curves. A comparative analysis across all three models is then presented. The chapter closes with a discussion of findings, group-level disengagement detection results, real-time system evaluation, and a summary.

5.1 Performance Evaluation Metrics

Model performance was assessed using four standard classification metrics computed at the subject level and then averaged across all participants. Accuracy measures the fraction of correctly classified trials over the total number of trials. Precision captures how many of the samples predicted as a given class truly belong to that class. Recall measures how many of the actual samples of a class the model correctly identified. The F1 score is the harmonic mean of precision and recall MEASURES the number of successful predictions in the actual sample of a class. The F1 score is the weighted average of the precision and recall, giving more weight to the smaller of the two. Use measure in cases in which class distributions are not uniform.The confusion matrices were prepared for for each subject and each fold to document behaviour at a class level. The overall confusion matrix aggregates predictions across all subjects and all folds. Training and validation accuracy and loss curves were plotted per subject to examine convergence behaviour and detect overfitting. All metrics reported are macro-averaged over the four cognitive load classes.

5.2 Results of EEG-HybridFormer

EEG-HybridFormer combines a compact convolutional front-end with a Transformer encoder to capture both local temporal features and long-range spatial dependencies from the raw EEG signal. The model was applied to the 4-class cognitive load classification problem under a subject-independent 5-fold cross-validation scheme. Figs. and tables in this section follow the order: hyperparameter configuration, confusion matrix analysis, subject-wise confusion matrices, and subject-wise training curves.

5.2.1 Hyperparameter Configuration

The hyperparameter settings used for EEG-HybridFormer are listed in Table 5.1.The A cosine annealing learning rate scheduler was used with the Adam optimizer to train the model. starting at 0.001 .The batch size was 32 and maximum epochs used were 15 per fold. The Transformer encoder layers were followed by a dropout with a rate of 0.5 . Minimise overfitting within subjects.The convolutional front end used 64 filters with a kernel length of 64 time points to capture theta and alpha band dynamics.

Table 5.1: Hyperparameter configuration for EEG-HybridFormer

Hyperparameter	**Value**
Optimizer	Adam
Learning rate	0.001 (cosine annealing)
Batch size	32
Maximum epochs	15
Dropout rate	0.5
Conv filters	64
Conv kernel length	64
Transformer heads	8
Transformer layers	2
Input shape	8 channels x 500 time points
Number of classes	4 (Relaxation, Easy, Medium, Hard)
Cross-validation	5-fold subject-independent

5.2.2 Confusion Matrix Analysis

Fig. 5.2 shows the subject-wise accuracy of EEG-HybridFormer across all 16 participants as a bar chart. Accuracy ranged from 70.00% for Student 10 to 98.06% for Student 13, with a mean of 85.64%. The spread indicates moderate inter-subject variability that is consistent with subject-independent EEG classification in naturalistic learning

environments.

Fig. 5.1 presents the overall confusion matrix aggregated across all 16 subjects and five folds. The matrix shows that Relaxation and Hard states were classified most reliably, while some confusion occurred between the Easy and Medium classes, which are adjacent in cognitive load intensity.

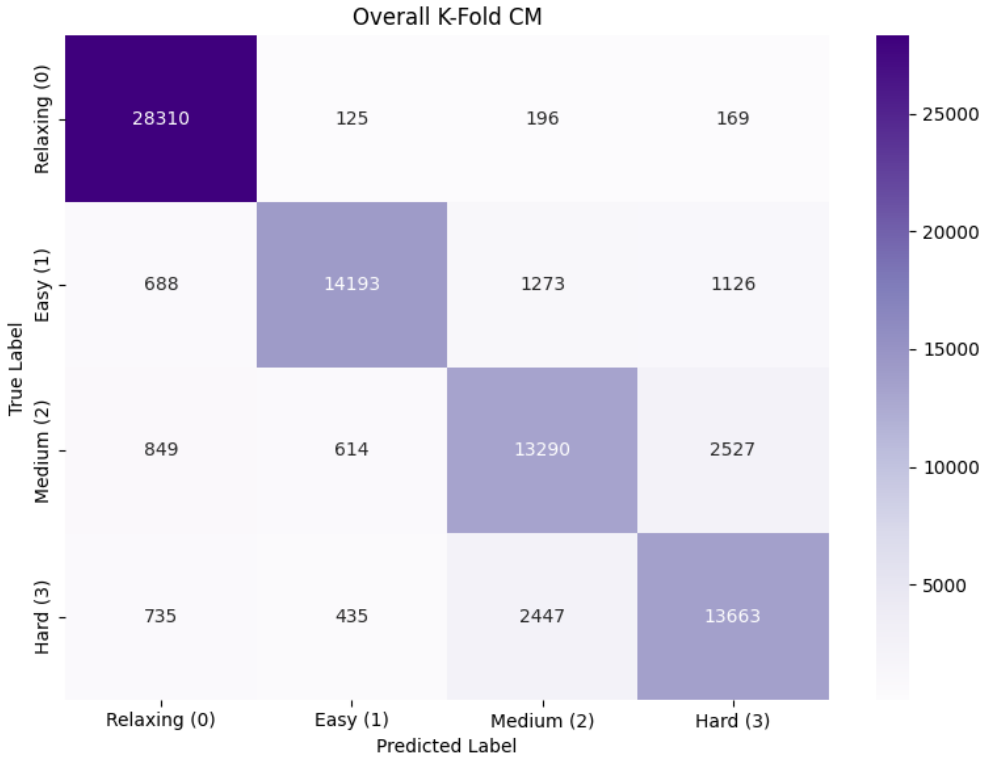


Fig. 5.1 Overall normalized confusion matrix for EEG-HybridFormer (aggregated across all subjects and folds)

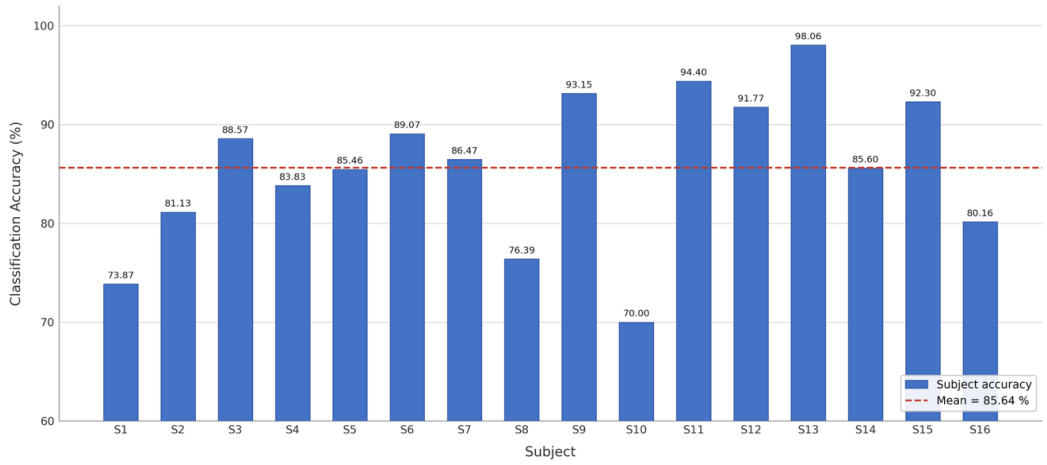


Fig. 5.2 Subject-wise accuracy bar chart for EEG-HybridFormer

Table 5.2: Subject-wise performance of EEG-HybridFormer

Subject	**Accuracy (%)**	**Precision**	**Recall**	**F1-Score**
Student 1	73.87	0.73	0.70	0.71
Student 2	81.13	0.80	0.78	0.79
Student 3	88.57	0.88	0.87	0.87
Student 4	83.83	0.82	0.81	0.81
Student 5	85.46	0.83	0.84	0.83
Student 6	89.07	0.88	0.87	0.88
Student 7	86.47	0.85	0.84	0.85
Student 8	76.39	0.77	0.73	0.73
Student 9	93.15	0.93	0.92	0.92
Student 10	70.00	0.70	0.65	0.66
Student 11	94.40	0.95	0.94	0.94
Student 12	91.77	0.92	0.92	—
Student 13	98.06	0.98	0.98	0.98
Student 14	85.60	0.84	0.84	0.84
Student 15	92.30	0.92	0.91	0.91
Student 16	80.16	0.78	0.77	0.78
Mean	**85.64**	**0.85**	**0.84**	**0.83**

The subject-wise metrics in Table 5.2 confirm that EEG-HybridFormer performed consistently well across most participants. The mean accuracy was 85.64% and the mean precision of 0.85 , mean recall of 0.84 , and mean F1 score of 0.83. Student 13 achieved the best accuracy was 98.06% that obtained by the Student 1 , and the lowest accuracy was obtained by the Student 10, which was 70.00%, Suggesting that some subjects

could have had less clearly defined EEG signatures throughout the Four cognitive load conditions were created

5.2.3 Subject-wise Confusion Matrices

The individual confusion matrices for each of the 16 classifiers are shown in Figs. 5.3 – 5.18. subjects. All five cross- networks and data from five independent runs. Reads the validation folds corresponding to the specified subject and normalizes all rows of the data to sum to one.Reads the validation folds corresponding to the specified subject and normalizes all rows of the data to sum to one. represent recall per class.

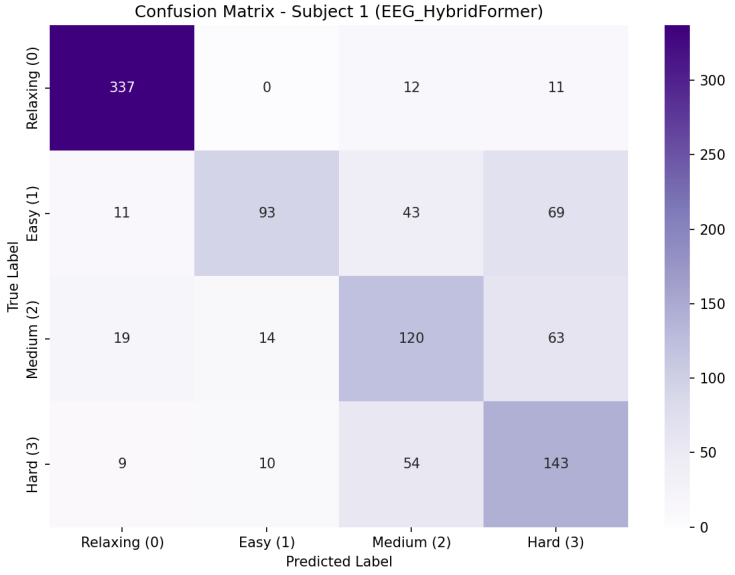


Fig. 5.3 Normalized confusion matrix for Student 1 (EEG-HybridFormer)

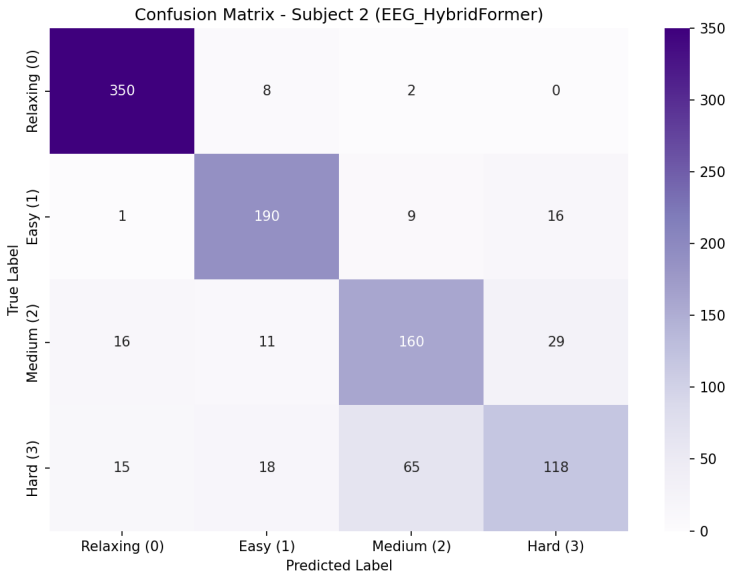


Fig. 5.4 Normalized confusion matrix for Student 2 (EEG-HybridFormer)

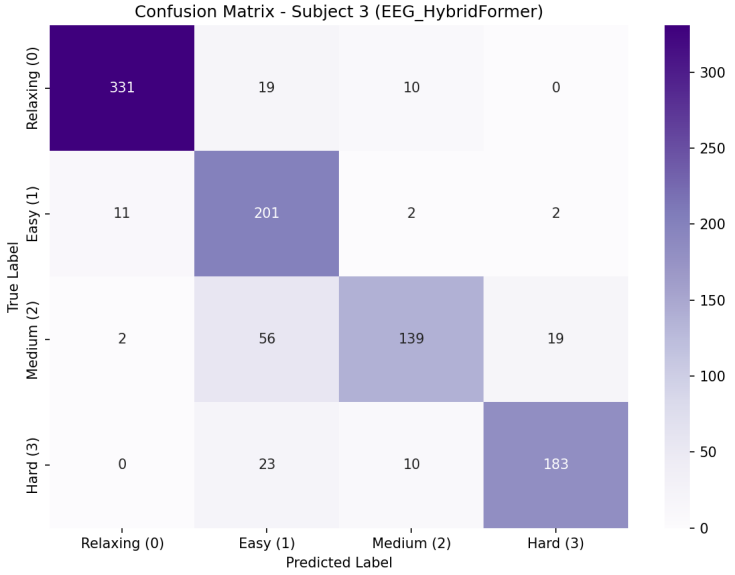


Fig. 5.5 Normalized confusion matrix for Student 3 (EEG-HybridFormer)

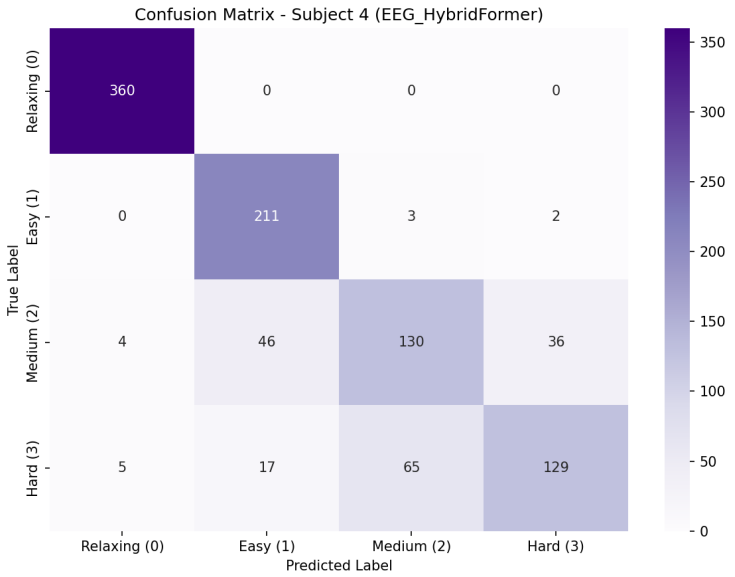


Fig. 5.6 Normalized confusion matrix for Student 4 (EEG-HybridFormer)

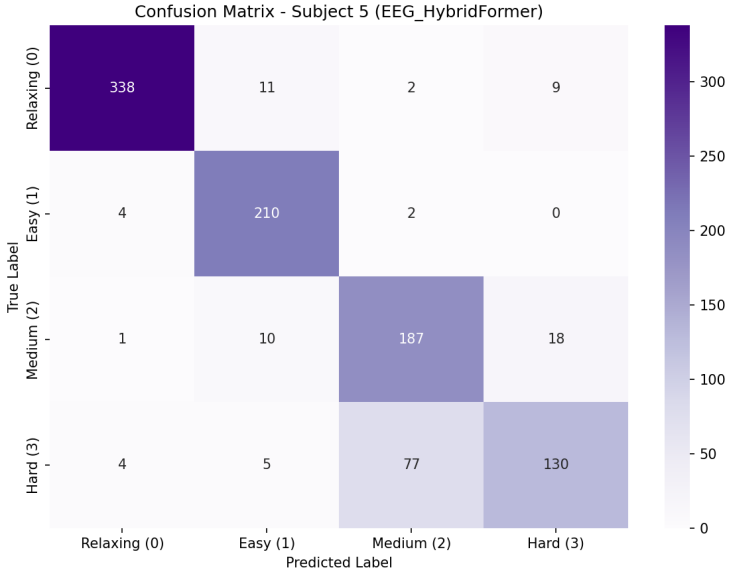


Fig. 5.7 Normalized confusion matrix for Student 5 (EEG-HybridFormer)

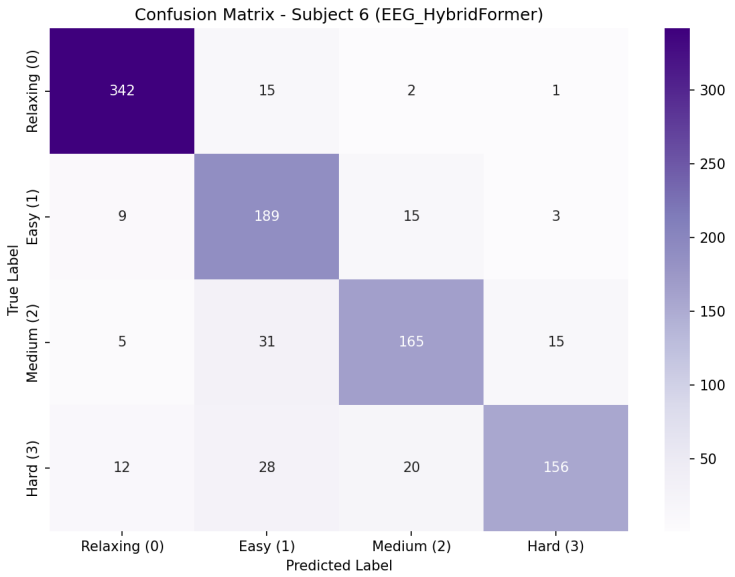


Fig. 5.8 Normalized confusion matrix for Student 6 (EEG-HybridFormer)

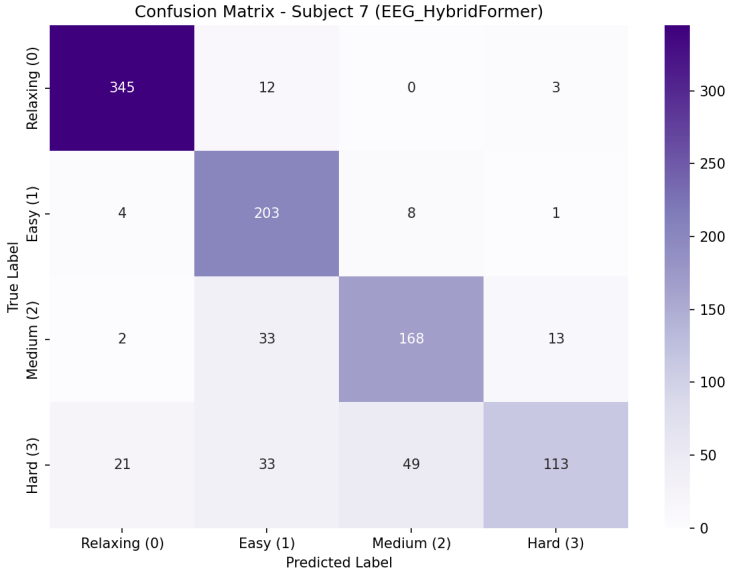


Fig. 5.9 Normalized confusion matrix for Student 7 (EEG-HybridFormer)

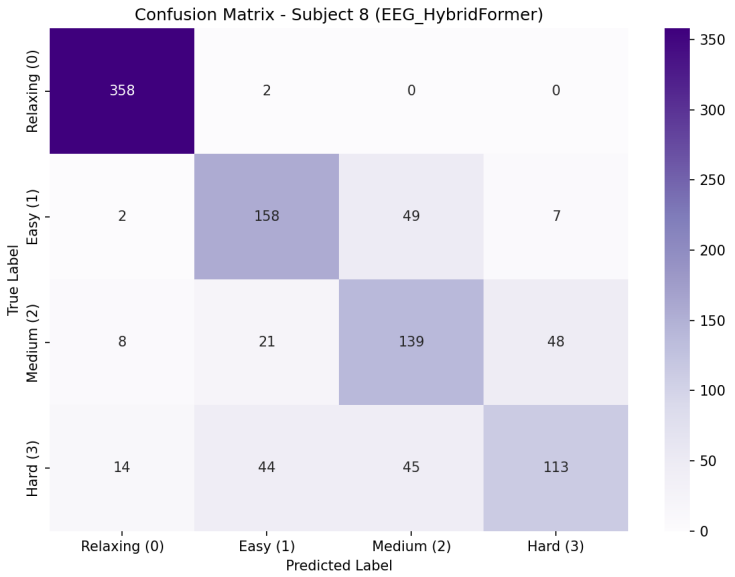


Fig. 5.10 Normalized confusion matrix for Student 8 (EEG-HybridFormer)

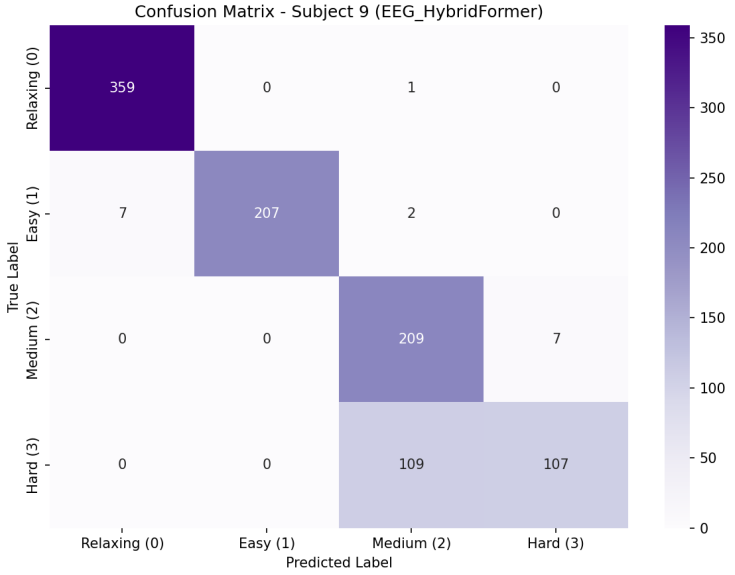


Fig. 5.11 Normalized confusion matrix for Student 9 (EEG-HybridFormer)

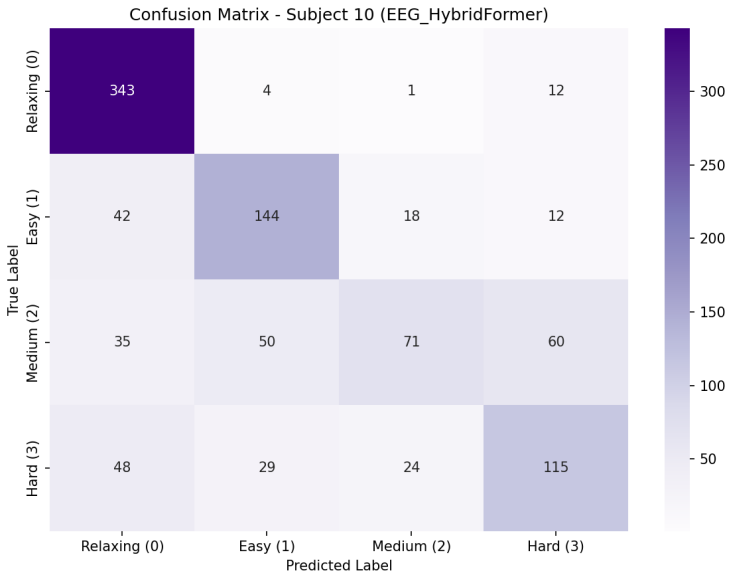


Fig. 5.12 Normalized confusion matrix for Student 10 (EEG-HybridFormer)

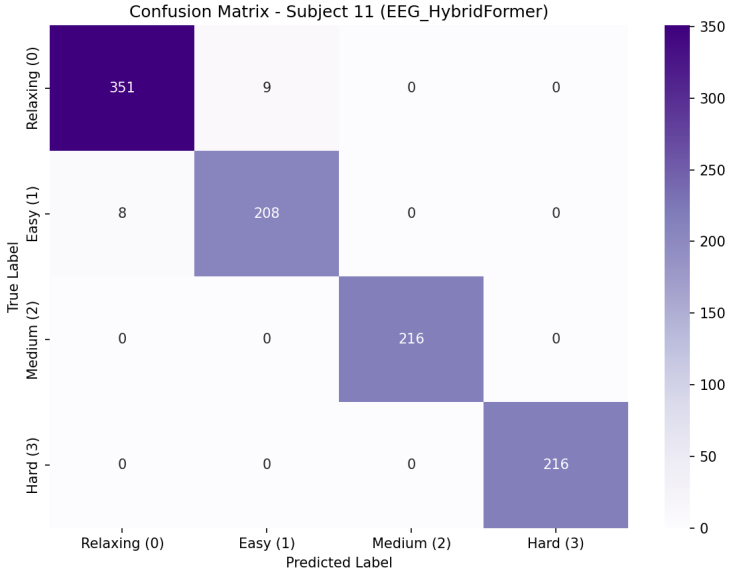


Fig. 5.13 Normalized confusion matrix for Student 11 (EEG-HybridFormer)

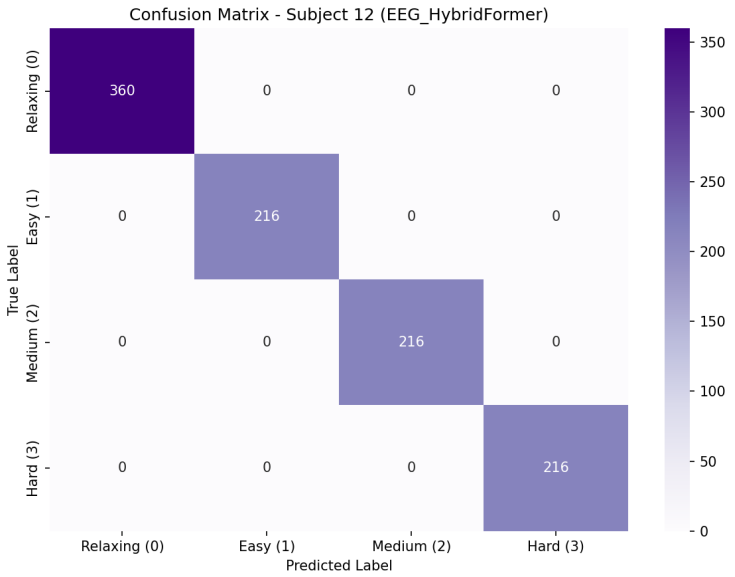


Fig. 5.14 Normalized confusion matrix for Student 12 (EEG-HybridFormer)

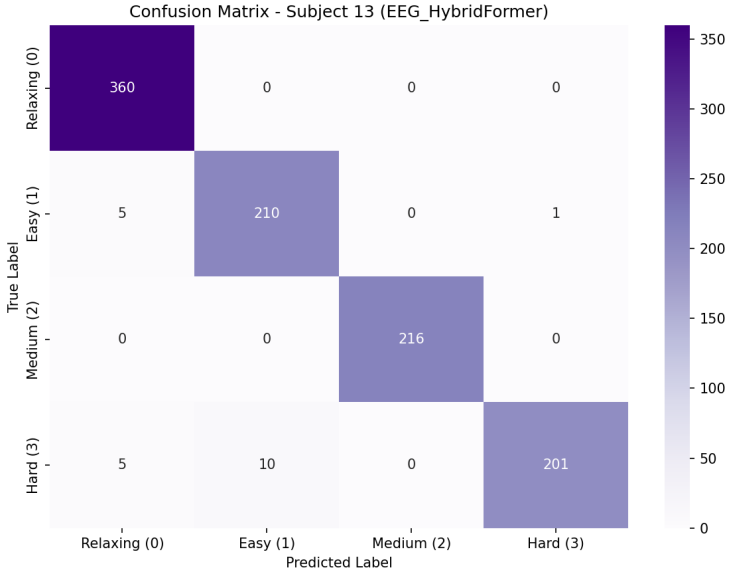


Fig. 5.15 Normalized confusion matrix for Student 13 (EEG-HybridFormer)

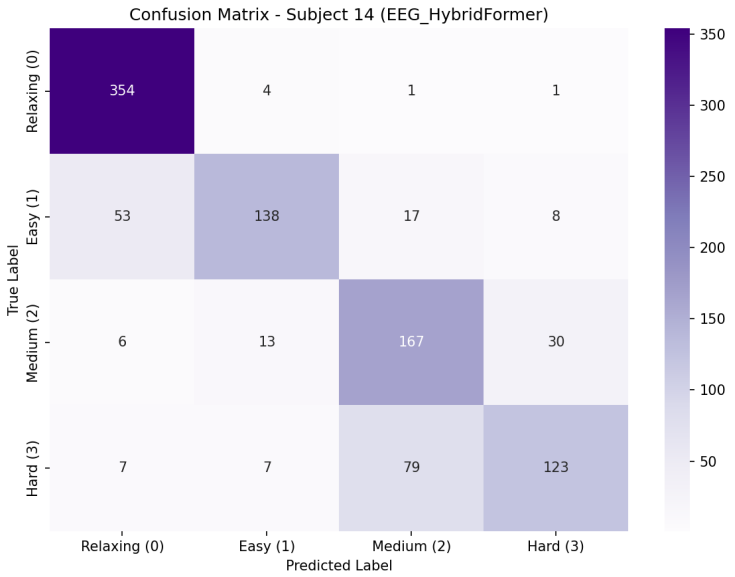


Fig. 5.16 Normalized confusion matrix for Student 14 (EEG-HybridFormer)

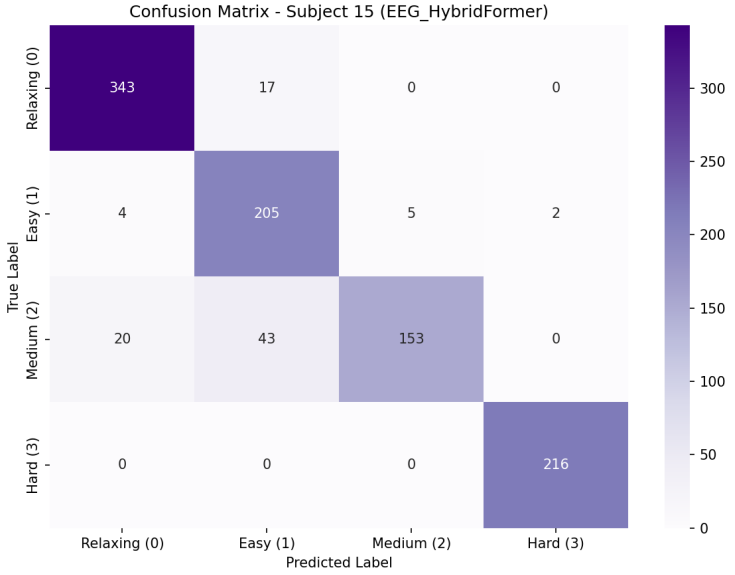


Fig. 5.17 Normalized confusion matrix for Student 15 (EEG-HybridFormer)

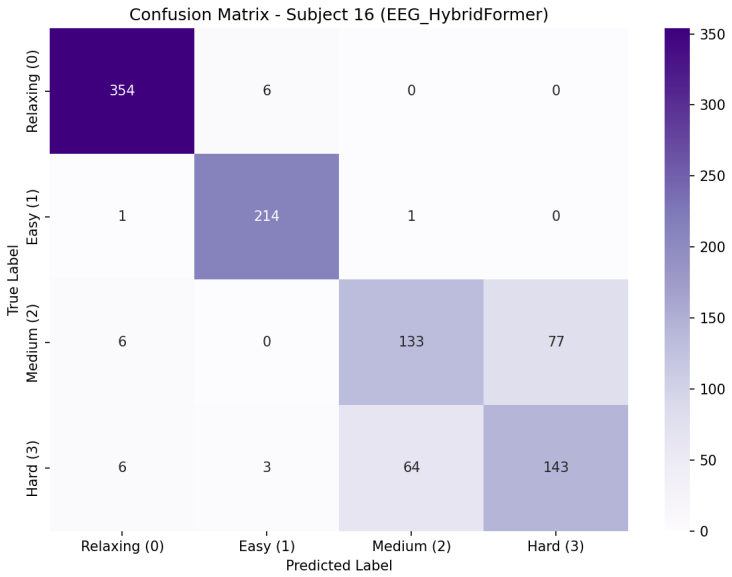


Fig. 5.18 Normalized confusion matrix for Student 16 (EEG-HybridFormer)

5.2.4 Subject-wise Accuracy and Loss Curves

Training dynamics were captured for every subject and for every fold. The accuracy and The means path on the loss curves below is across the five folds for each subject and show the convergence speed of the model and whether it overfits the training set.

5.2.5 Subject-wise Accuracy Curves

The training and validation accuracy curves for each of the datasets are shown in figures 5.19 through 5.34. subject. In most of the cases, the accuracy of the trained model converged within 80 to 100 epochs. Validation for students with high overall scores accuracy followed the training accuracy closely, as Students 11 and 13, suggesting good generalisation to out-of-sample data for. those individuals. A wider gap between training and validation accuracy was observed for Students 1 and 10, consistent with their lower overall performance.

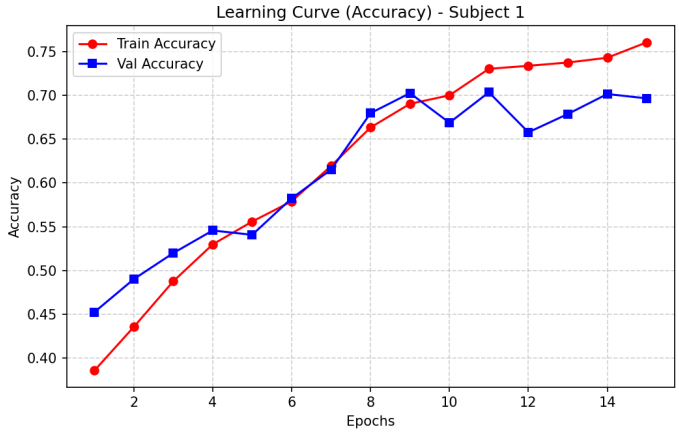


Fig. 5.19 Training and validation accuracy curves for Student 1 (EEG-HybridFormer)

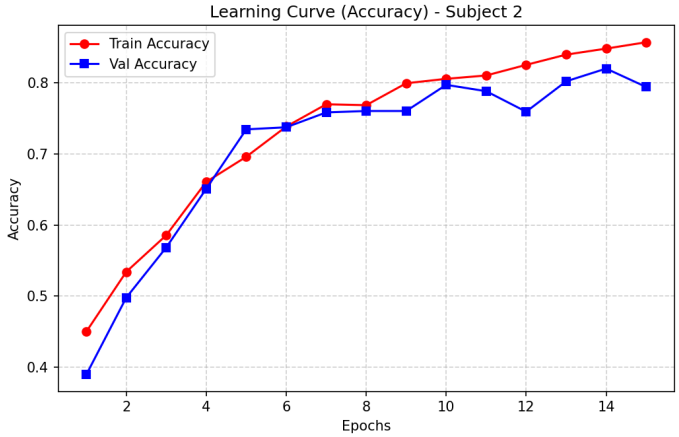


Fig. 5.20 Training and validation accuracy curves for Student 2 (EEG-HybridFormer)

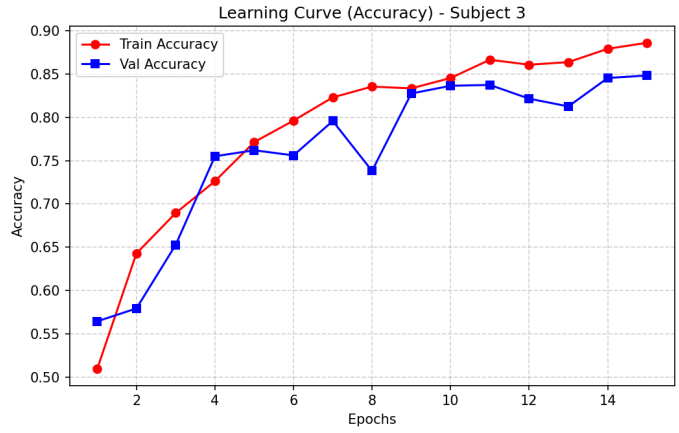


Fig. 5.21 Training and validation accuracy curves for Student 3 (EEG-HybridFormer)

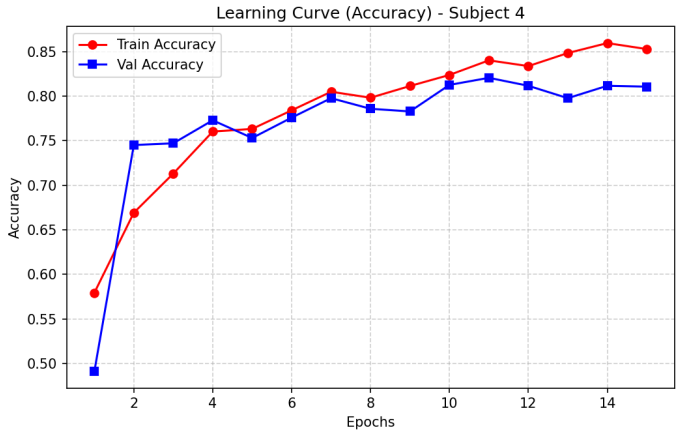


Fig. 5.22 Training and validation accuracy curves for Student 4 (EEG-HybridFormer)

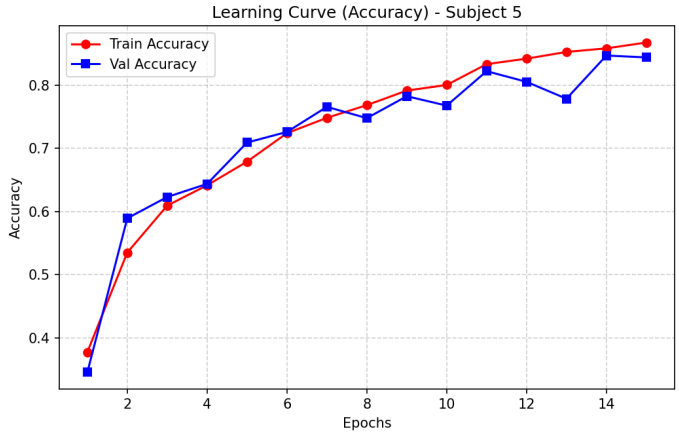


Fig. 5.23 Training and validation accuracy curves for Student 5 (EEG-HybridFormer)

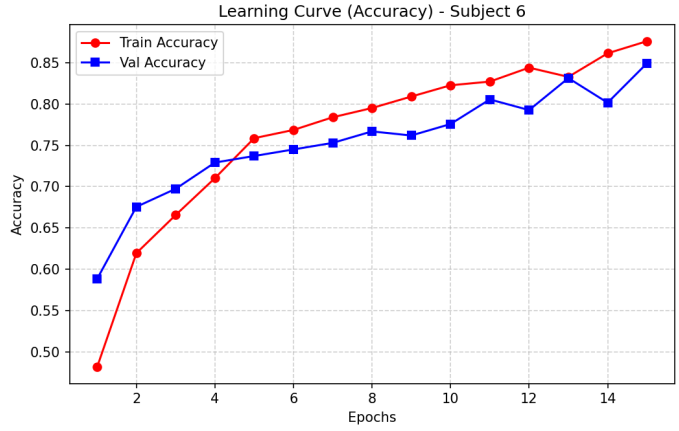


Fig. 5.24 Training and validation accuracy curves for Student 6 (EEG-HybridFormer)

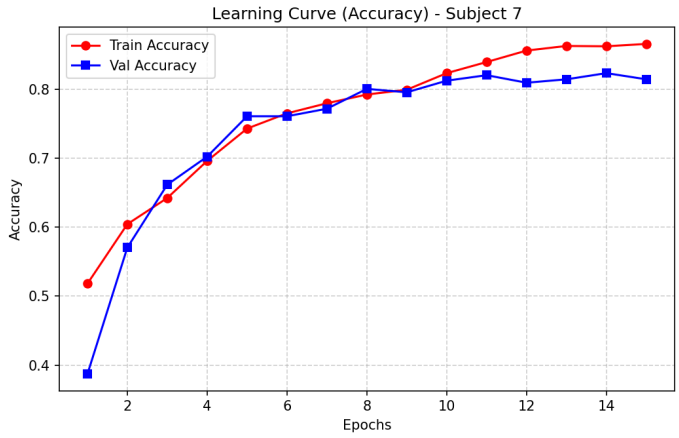


Fig. 5.25 Training and validation accuracy curves for Student 7 (EEG-HybridFormer)

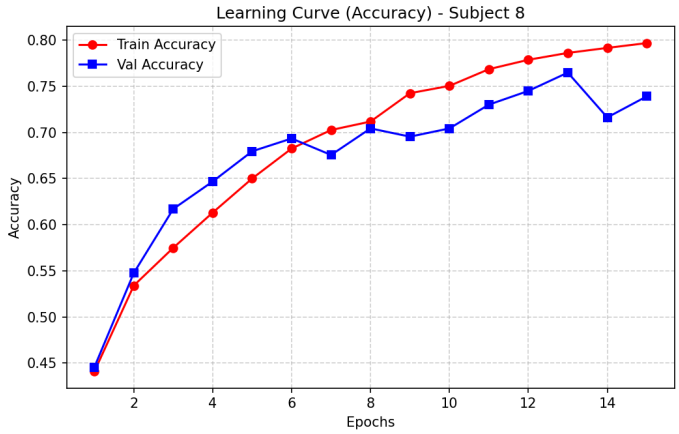


Fig. 5.26 Training and validation accuracy curves for Student 8 (EEG-HybridFormer)

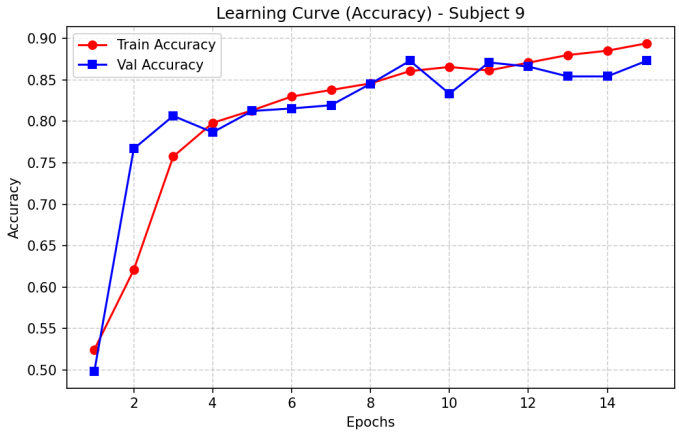


Fig. 5.27 Training and validation accuracy curves for Student 9 (EEG-HybridFormer)

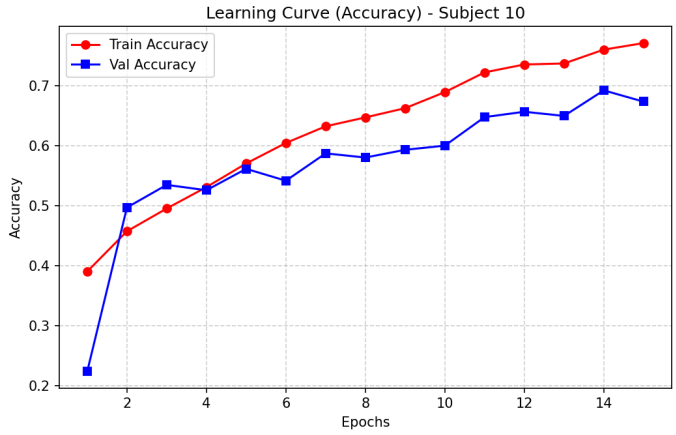


Fig. 5.28 Training and validation accuracy curves for Student 10 (EEG-HybridFormer)

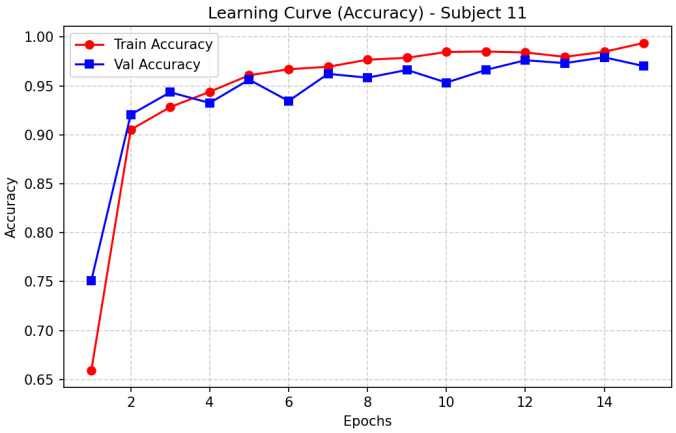


Fig. 5.29 Training and validation accuracy curves for Student 11 (EEG-HybridFormer)

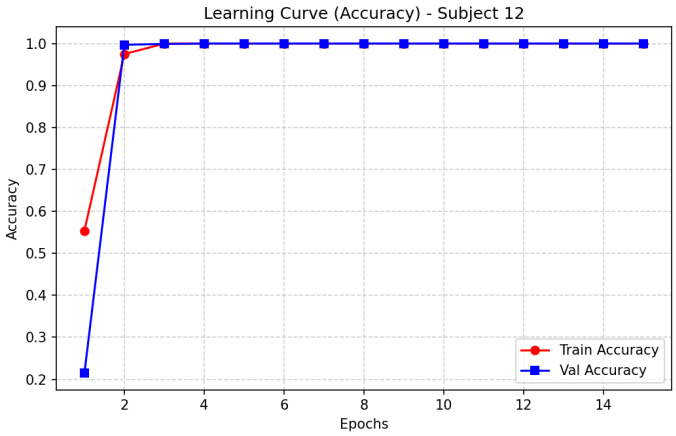


Fig. 5.30 Training and validation accuracy curves for Student 12 (EEG-HybridFormer)

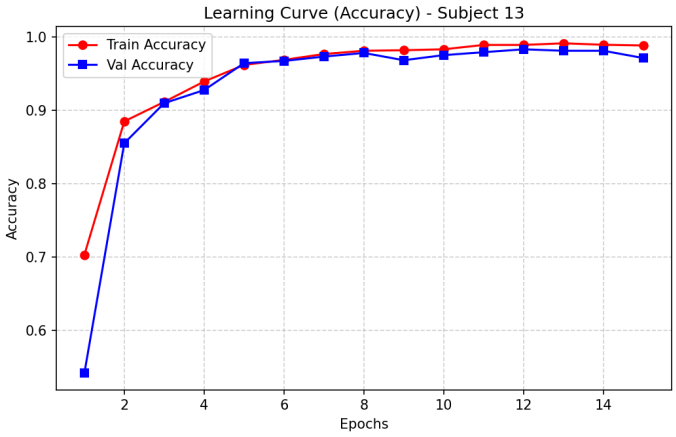


Fig. 5.31 Training and validation accuracy curves for Student 13 (EEG-HybridFormer)

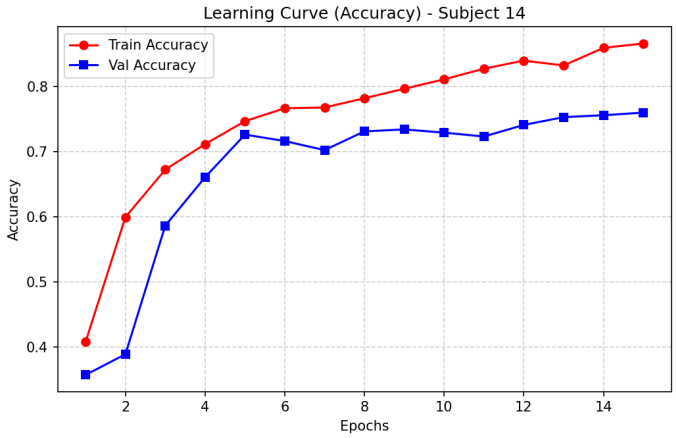


Fig. 5.32 Training and validation accuracy curves for Student 14 (EEG-HybridFormer)

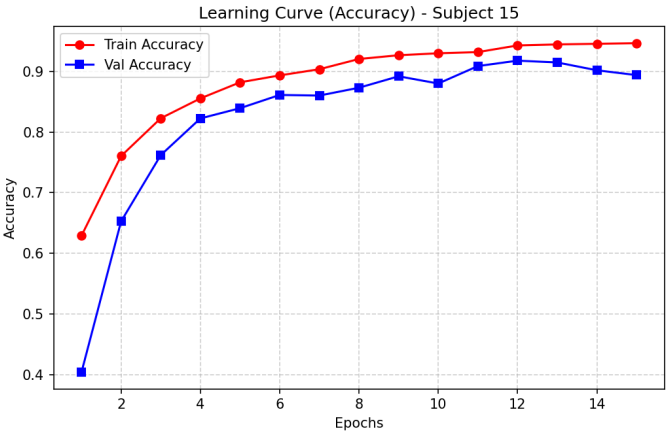


Fig. 5.33 Training and validation accuracy curves for Student 15 (EEG-HybridFormer)

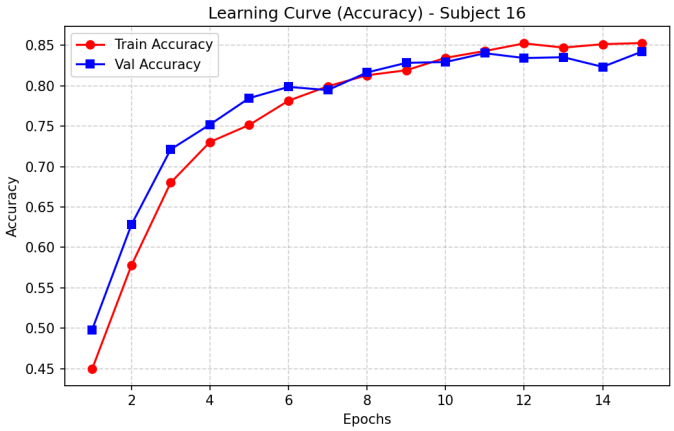


Fig. 5.34 Training and validation accuracy curves for Student 16 (EEG-HybridFormer)

5.2.6 Subject-wise Loss Curves

The loss curves for training and validation losses over subjects are depicted in Figs. 5.35 – 5.50, respectively. The cross-entropy loss for most participants was decreasing with epochs. Students 9, 11, 13, and 15 had validation loss which were consistently near the training loss, These are in line with their high accuracy points. For Student 1 and 10, validat- layer had an impact on.Later epochs showed some slight deviation from the value of tion loss, indicating mild overfitting caused by the dropout layer. Only partial mitigation of regularisation.

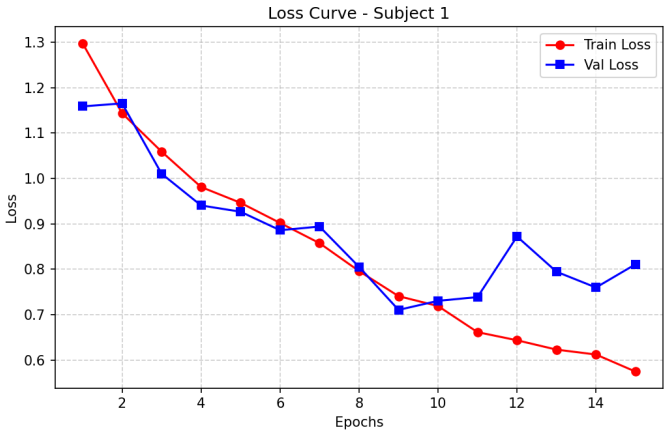


Fig. 5.35 Training and validation loss curves for Student 1 (EEG-HybridFormer)

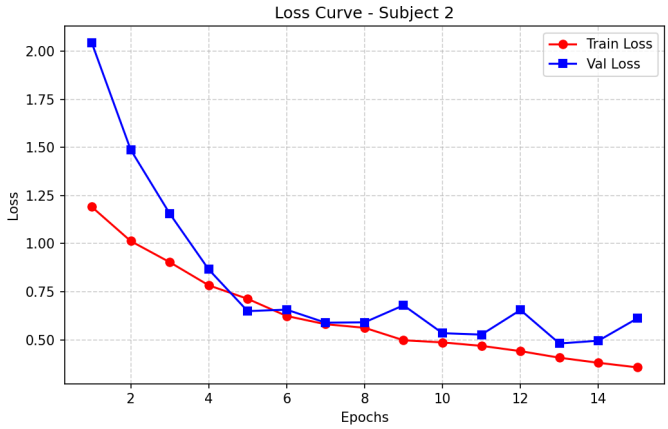


Fig. 5.36 Training and validation loss curves for Student 2 (EEG-HybridFormer)

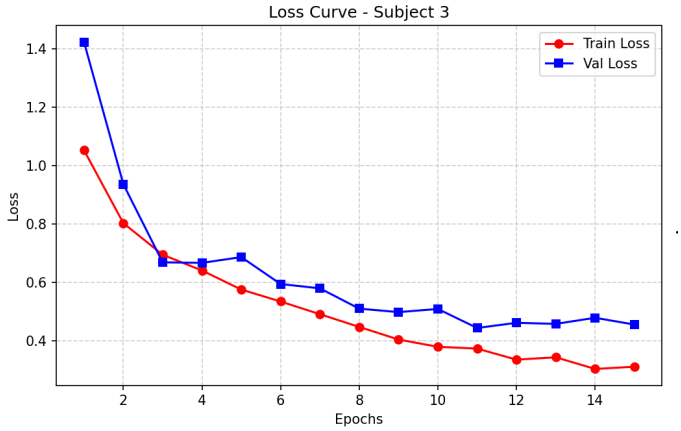


Fig. 5.37 Training and validation loss curves for Student 3 (EEG-HybridFormer)

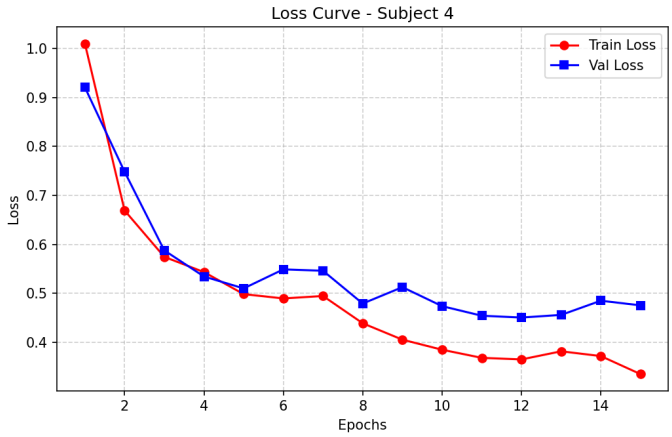


Fig. 5.38 Training and validation loss curves for Student 4 (EEG-HybridFormer)

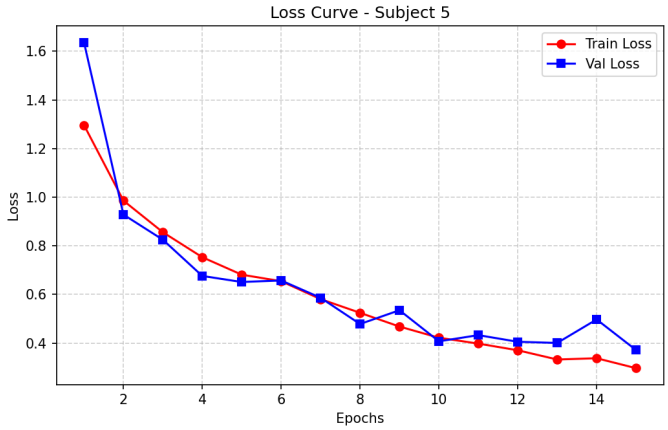


Fig. 5.39 Training and validation loss curves for Student 5 (EEG-HybridFormer)

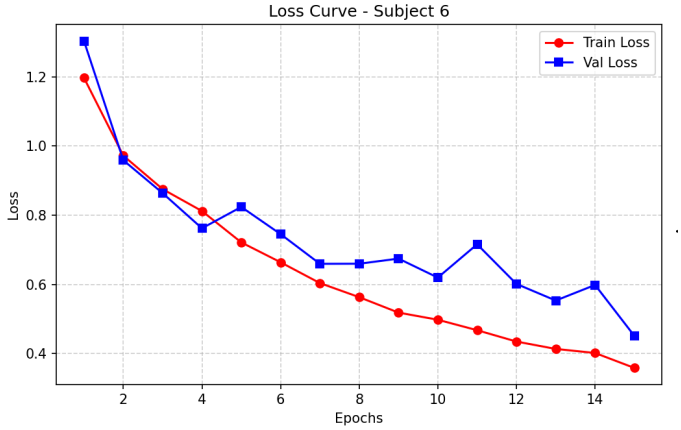


Fig. 5.40 Training and validation loss curves for Student 6 (EEG-HybridFormer)

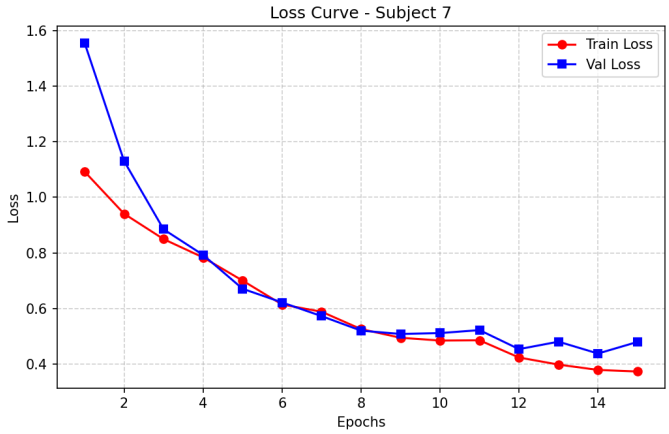


Fig. 5.41 Training and validation loss curves for Student 7 (EEG-HybridFormer)

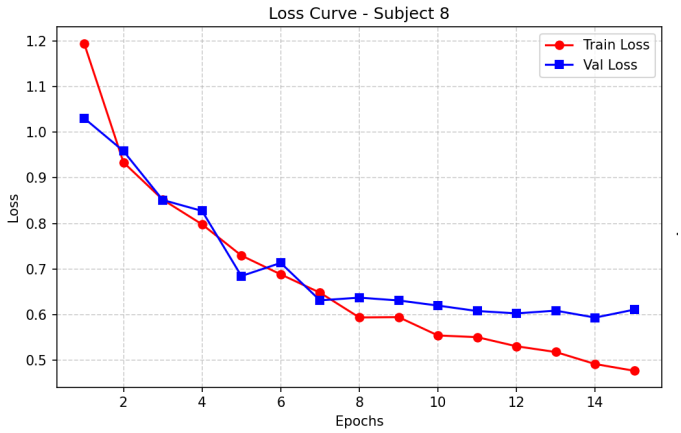


Fig. 5.42 Training and validation loss curves for Student 8 (EEG-HybridFormer)

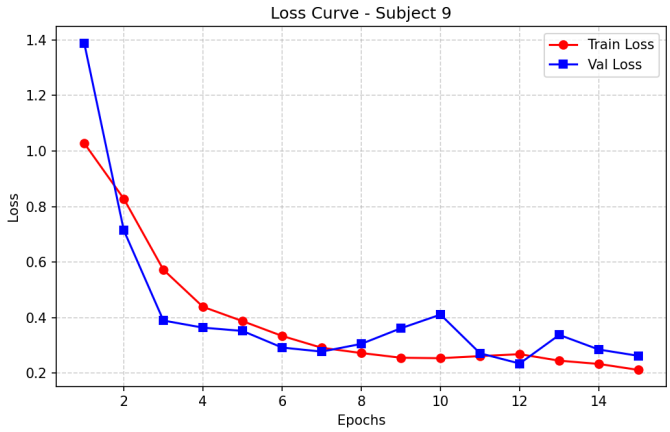


Fig. 5.43 Training and validation loss curves for Student 9 (EEG-HybridFormer)

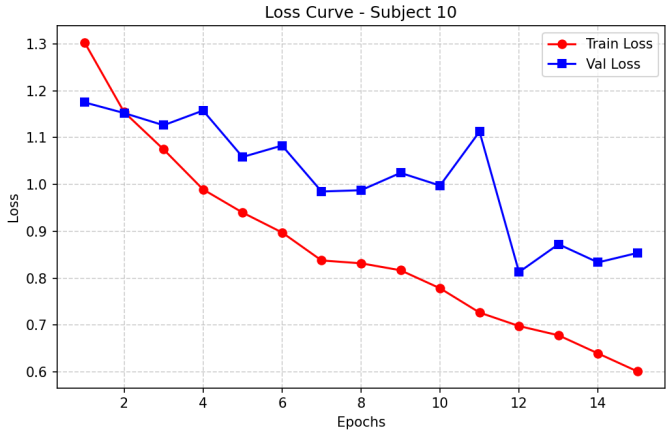


Fig. 5.44 Training and validation loss curves for Student 10 (EEG-HybridFormer)

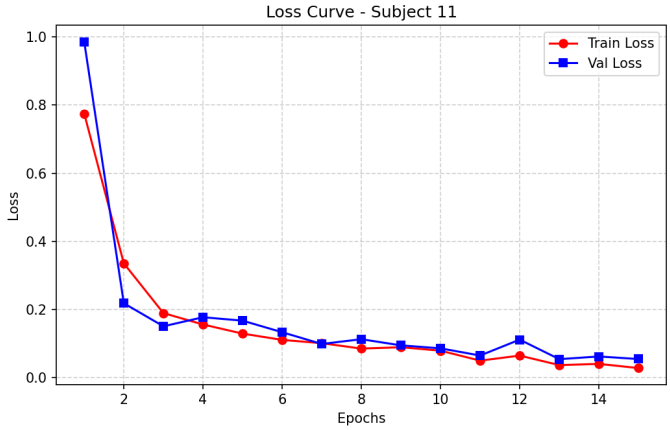


Fig. 5.45 Training and validation loss curves for Student 11 (EEG-HybridFormer)

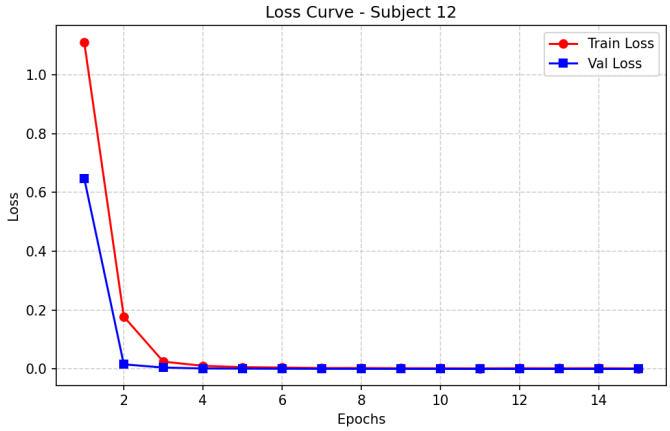


Fig. 5.46 Training and validation loss curves for Student 12 (EEG-HybridFormer)

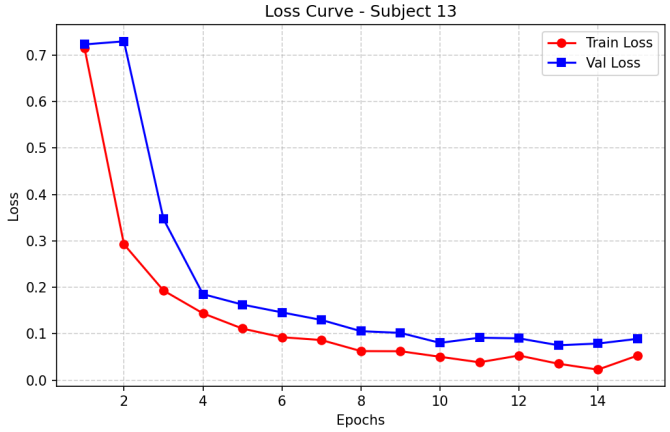


Fig. 5.47 Training and validation loss curves for Student 13 (EEG-HybridFormer)

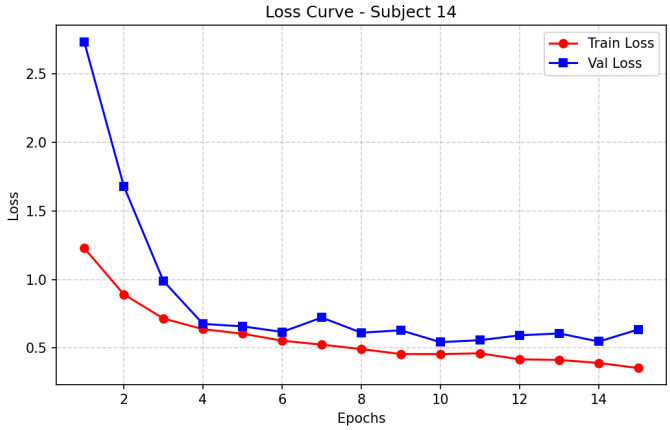


Fig. 5.48 Training and validation loss curves for Student 14 (EEG-HybridFormer)

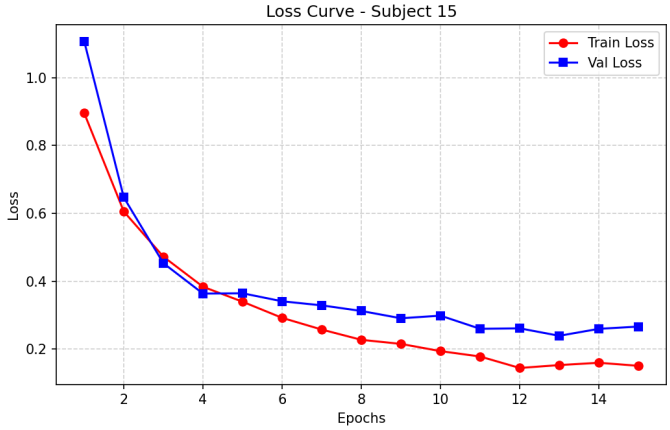


Fig. 5.49 Training and validation loss curves for Student 15 (EEG-HybridFormer)

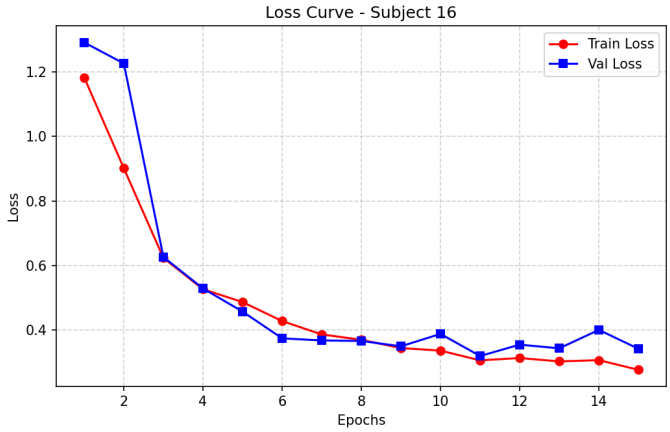


Fig. 5.50 Training and validation loss curves for Student 16 (EEG-HybridFormer)

5.3 Results of EEG-Deformer++

EEG-Deformer++ is a graph-convolution-based conformer that models spatial relationships between EEG channels explicitly through a dynamic graph structure before applying attention over temporal features. Under the same 5-fold subject-independent protocol, this model was evaluated on the four-class cognitive load task. Results follow the order: hyperparameter configuration, subject-wise performance table, subject-wise results (confusion matrices, accuracy curves, loss curves), and finally the overall confusion matrix.

5.3.1 Hyperparameter Configuration

Table 5.3 lists the hyperparameter configuration for EEG-Deformer++.In this the model we have used is the the AdamW optimizer and it is with a learning rate of 0.0005 and with a weight decay of 1e-4. A batch size of 64 was selected. The graph network used a k-nearest-neighbour graph with k = 4 for each EEG epoch and updated the adjacency matrix dynamically at each forward pass. Conformer attention used 4 heads with a depth of 2 layers.

Table 5.3: Hyperparameter configuration for EEG-Deformer++

Hyperparameter	**Value**
Optimizer	AdamW
Learning rate	0.0005
Weight decay	1e-4
Batch size	64
Maximum epochs	15
Graph k-NN neighbors	4
Attention heads	4
Conformer layers	2
Dropout rate	0.3
Input shape	8 channels x 500 time points
Number of classes	4 (Relaxation, Easy, Medium , Hard)
Cross validation	5 fold subject independent

5.3.2 Overall Confusion Matrix Analysis

Fig. 5.51 presents the overall confusion matrix for EEG-Deformer++, aggregated across all 16 subjects and five folds. The matrix confirms that Relaxation and Hard states achieved the highest true-positive rates, while Easy and Medium shared notice-

able misclassification. This pattern is consistent across individual subjects and reflects the overlapping spectral characteristics of moderate cognitive load states recorded during natural classroom tasks.

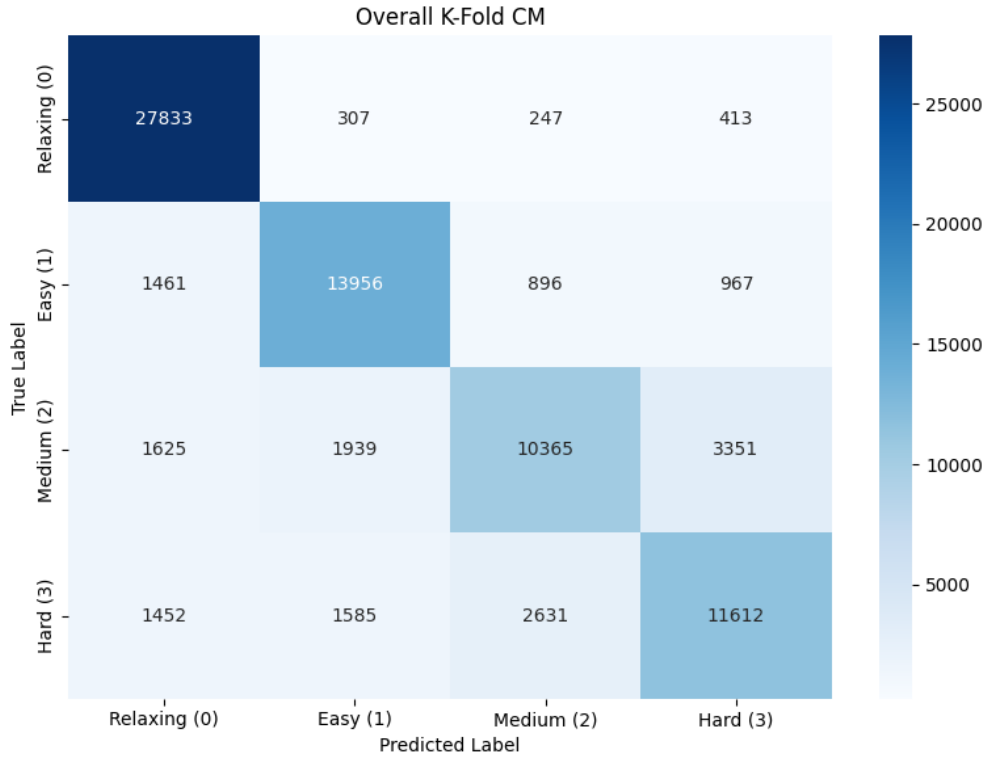


Fig. 5.51 Overall normalized confusion matrix for EEG-Deformer++ (aggregated across all subjects and folds)

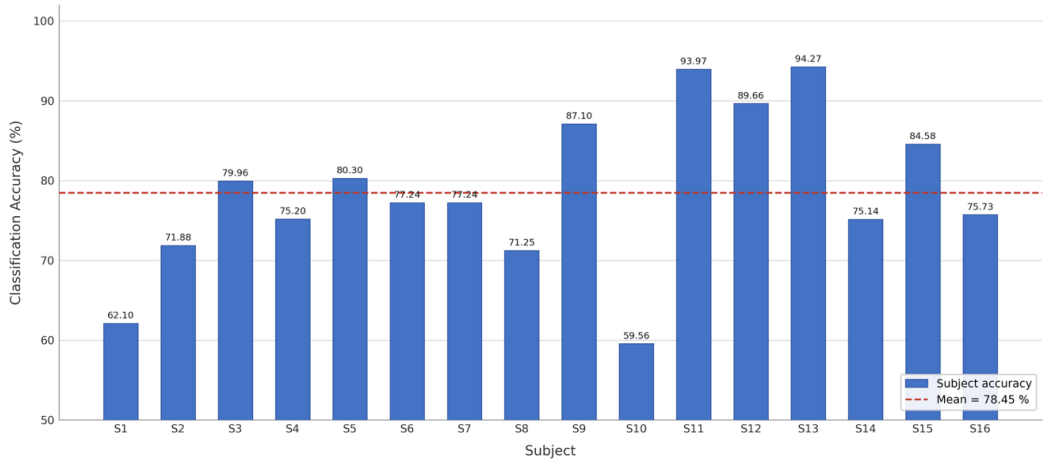


Fig. 5.52 Subject-wise accuracy bar chart for EEG-Deformer++

5.3.3 Subject-wise Performance

Table 5.4 shows the accuracy , precision , recall and F1-score for each of the 16 subjects under EEG-Deformer++. This model achieved an average accuracy of 78.45%, is about 7 percentage points behind EEG-HybridFormer. The highest accuracy was 94.27% for Student 13 , and the lowest was 59.56% for Student 10. The same was to be expected of those who had lower accuracy in EEG-HybridFormer, which suggests that the quality of the EEG signals varies from subject to subject, not model to model. specific weakness.

Table 5.4: Subject-wise performance of EEG-Deformer++

Subject	**Accuracy (%)**	**Precision**	**Recall**	**F1-Score**
Student 1	62.10	0.57	0.57	0.55
Student 2	71.88	0.68	0.68	0.67
Student 3	79.96	0.78	0.77	0.77
Student 4	75.20	0.72	0.71	0.71
Student 5	80.30	0.78	0.78	0.78
Student 6	77.24	0.77	0.74	0.74
Student 7	77.24	0.75	0.74	0.74
Student 8	71.25	0.68	0.67	0.66
Student 9	87.10	0.86	0.85	0.85
Student 10	59.56	0.56	0.54	0.53
Student 11	93.97	0.94	0.94	0.94
Student 12	89.66	0.86	0.86	0.86
Student 13	94.27	0.94	0.94	0.94
Student 14	75.14	0.73	0.72	0.72
Student 15	84.58	0.85	0.83	0.82
Student 16	75.73	0.73	0.72	0.72
Mean	**78.45**	**0.76**	**0.75**	**0.75**

5.4 Subject-wise Results of EEG-Deformer++

This section presents the per-subject confusion matrices followed by the accuracy and loss training curves for EEG-Deformer++. Fig. 5.52 provides a bar chart summary of accuracy across all 16 subjects.

5.4.1 Subject-wise Confusion Matrices

Fig. 5.52 shows the subject-wise accuracy bar chart for EEG-Deformer++. Figs. 5.53 through 5.68 present the individual normalized confusion matrices per subject. Compared to EEG-HybridFormer, EEG-Deformer++ showed greater confusion between the Easy and Medium classes, which may reflect the graph convolution layer having less capacity to separate closely adjacent cognitive states without explicit temporal modelling from raw waveforms.

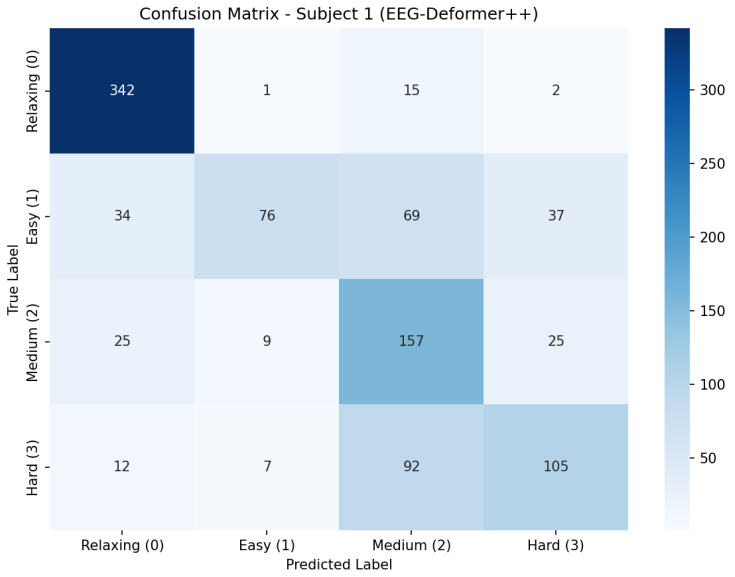


Fig. 5.53 Normalized confusion matrix for Student 1 (EEG-Deformer++)

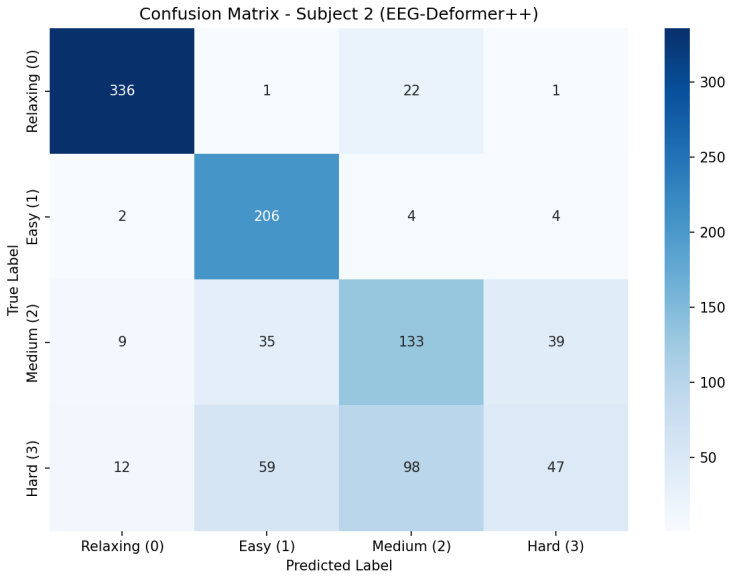


Fig. 5.54 Normalized confusion matrix for Student 2 (EEG-Deformer++)

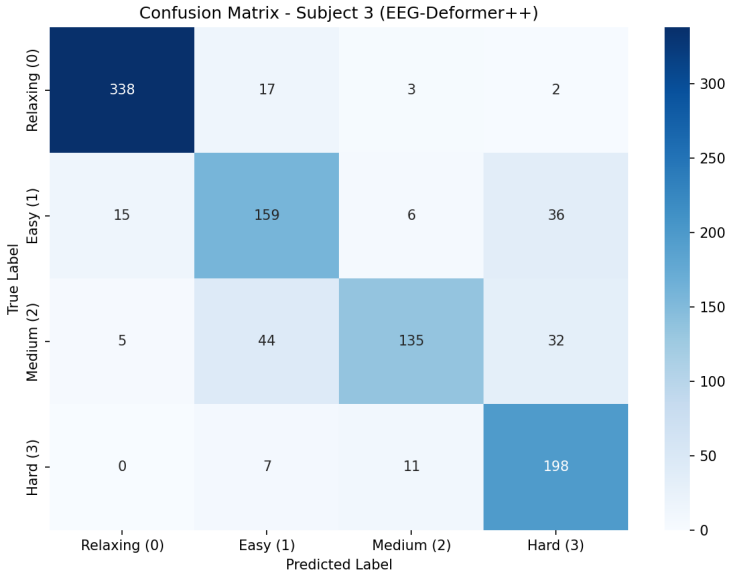


Fig. 5.55 Normalized confusion matrix for Student 3 (EEG-Deformer++)

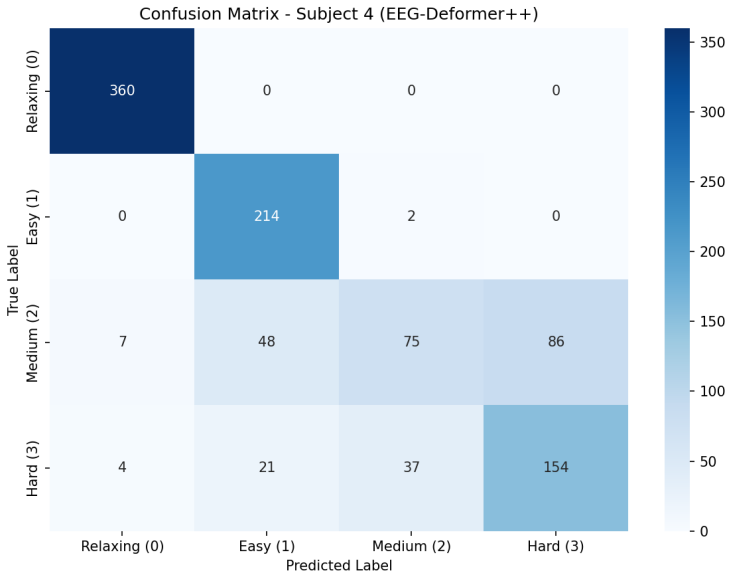


Fig. 5.56 Normalized confusion matrix for Student 4 (EEG-Deformer++)

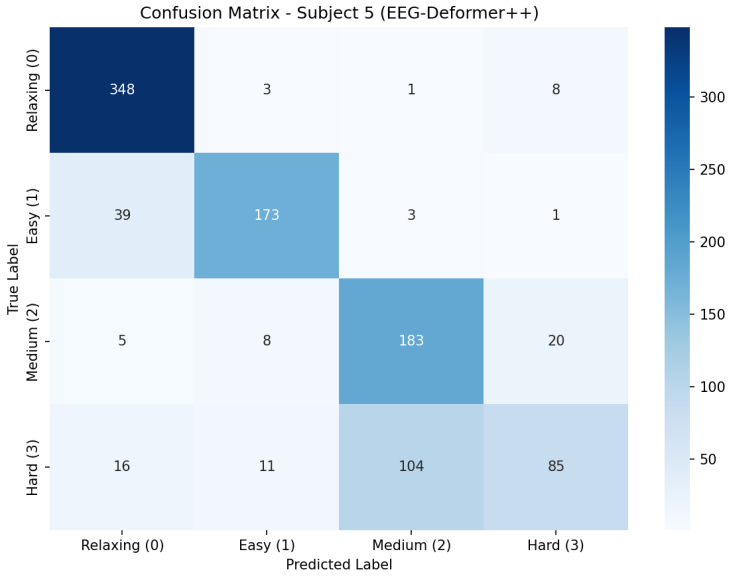


Fig. 5.57 Normalized confusion matrix for Student 5 (EEG-Deformer++)

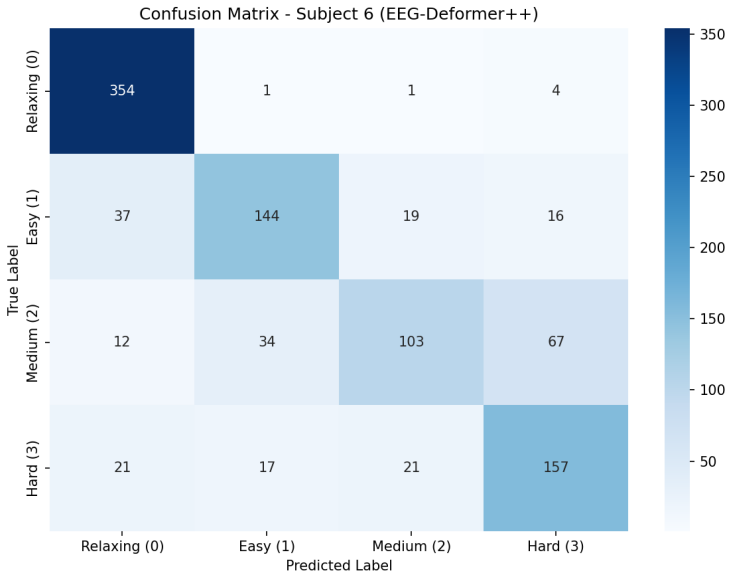


Fig. 5.58 Normalized confusion matrix for Student 6 (EEG-Deformer++)

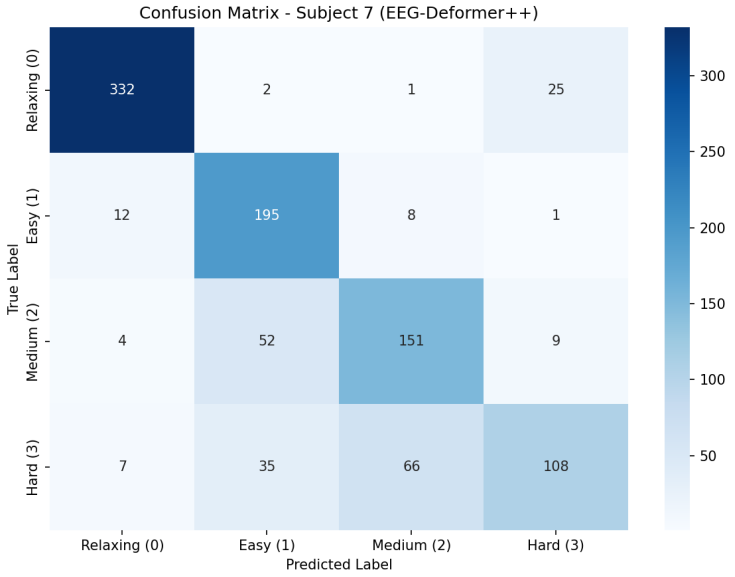


Fig. 5.59 Normalized confusion matrix for Student 7 (EEG-Deformer++)

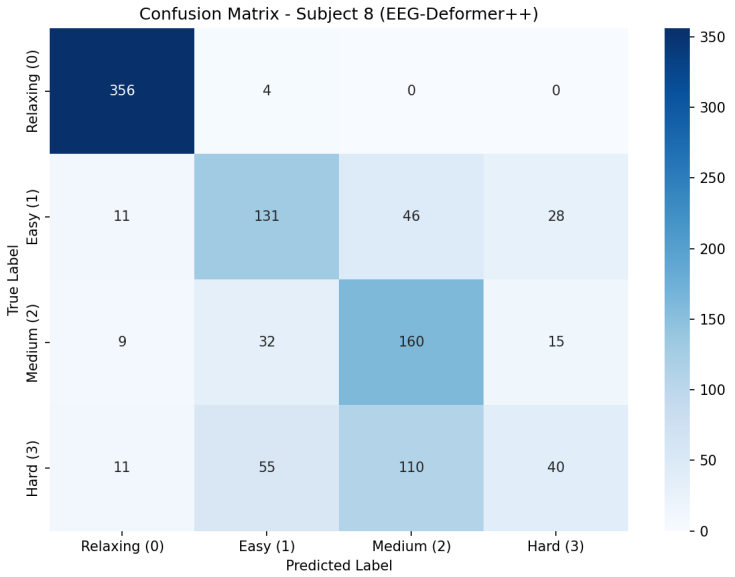


Fig. 5.60 Normalized confusion matrix for Student 8 (EEG-Deformer++)

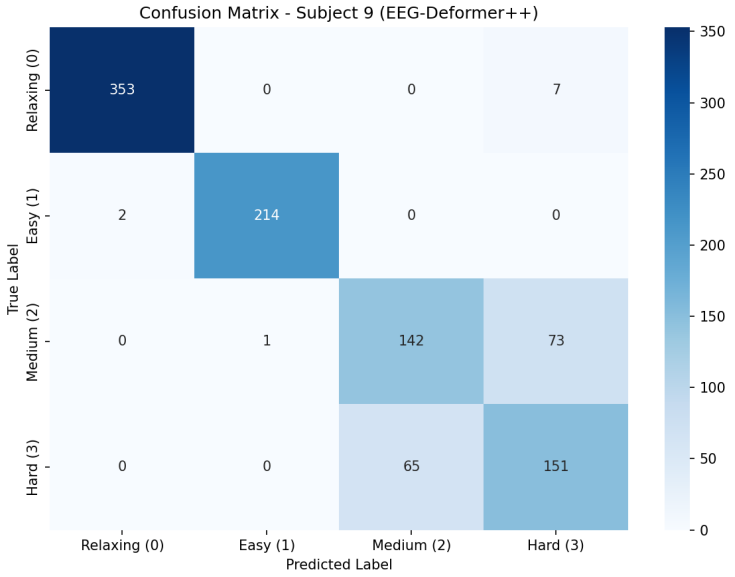


Fig. 5.61 Normalized confusion matrix for Student 9 (EEG-Deformer++)

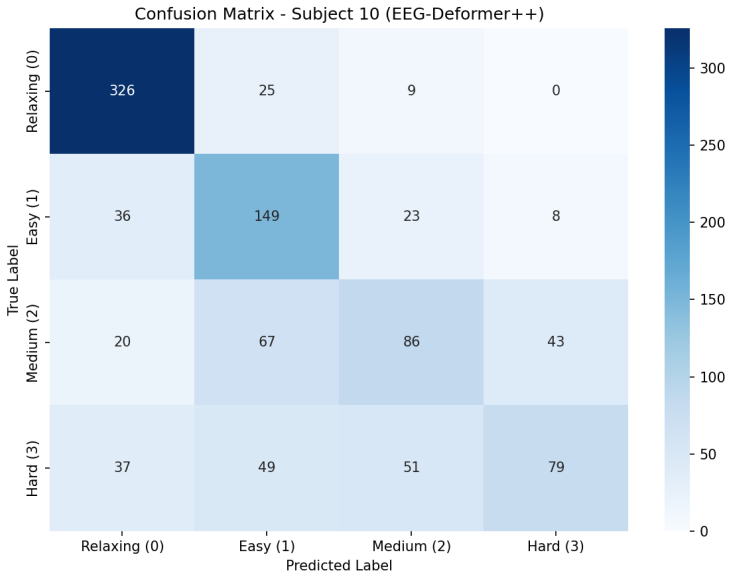


Fig. 5.62 Normalized confusion matrix for Student 10 (EEG-Deformer++)

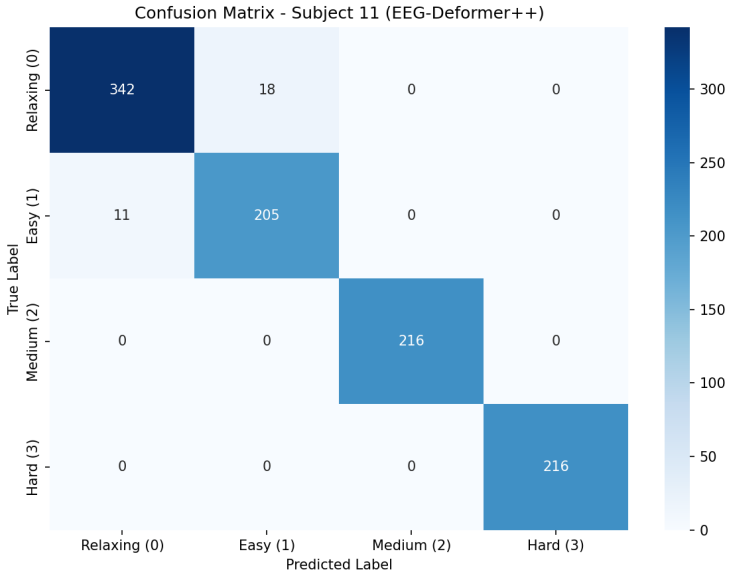


Fig. 5.63 Normalized confusion matrix for Student 11 (EEG-Deformer++)

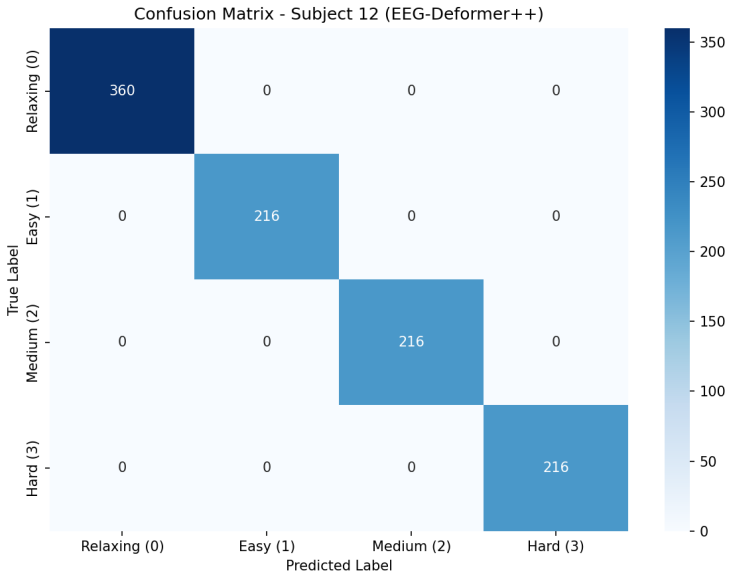


Fig. 5.64 Normalized confusion matrix for Student 12 (EEG-Deformer++)

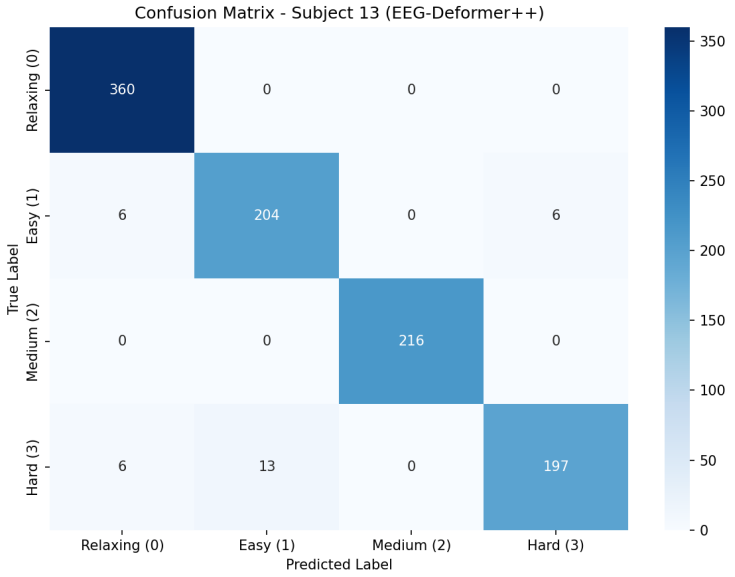


Fig. 5.65 Normalized confusion matrix for Student 13 (EEG-Deformer++)

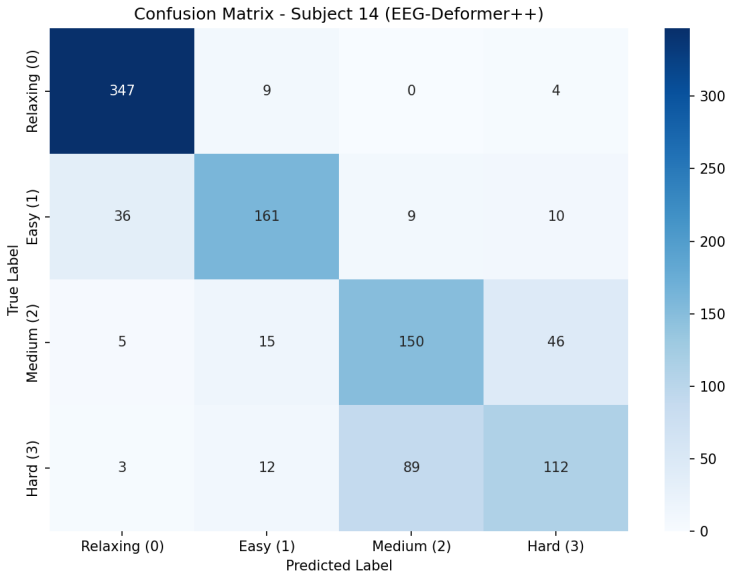


Fig. 5.66 Normalized confusion matrix for Student 14 (EEG-Deformer++)

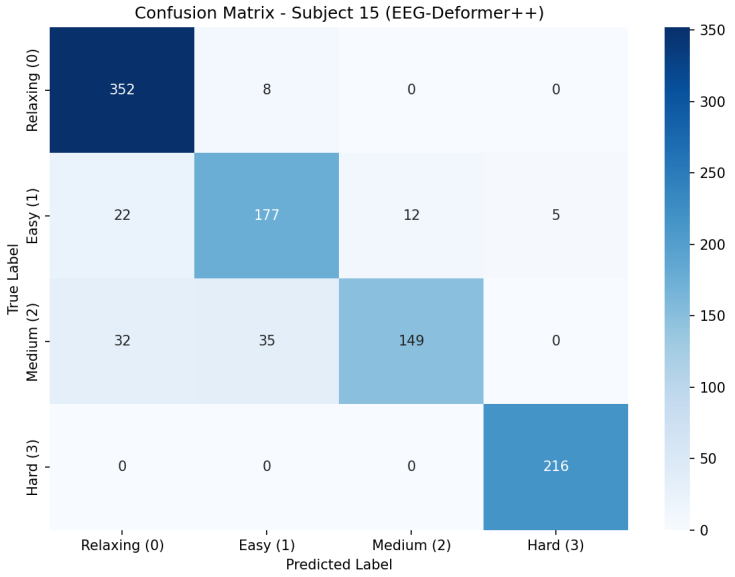


Fig. 5.67 Normalized confusion matrix for Student 15 (EEG-Deformer++)

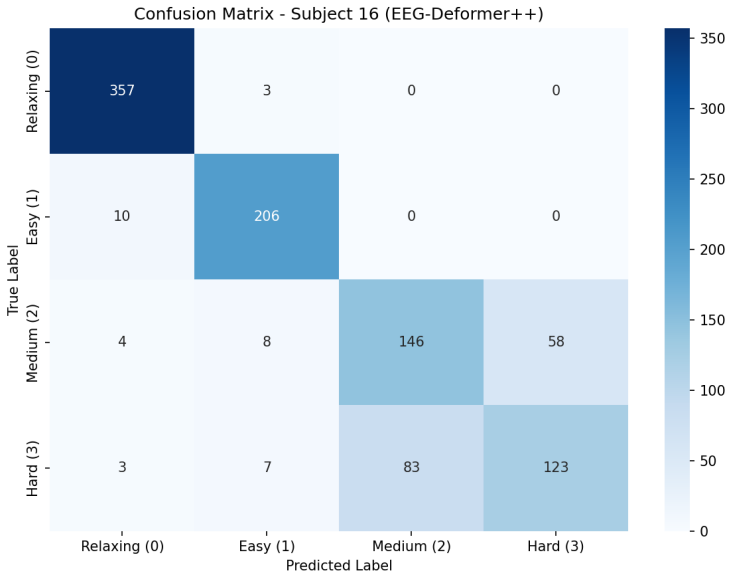


Fig. 5.68 Normalized confusion matrix for Student 16 (EEG-Deformer++)

5.4.2 Subject-wise Accuracy Curves

Figs. 5.69 through 5.84 show training and validation accuracy curves for each of the 16 subjects under EEG-Deformer++. In general, the model needed more epochs to converge to the desired accuracy. EEG-HybridFormer outperforms the other models with most of the subjects converging to stable accuracy after epoch 100 to 120. Stu- compared to the other models, as evidenced by the wider standard deviations.The standard deviations were wider and more volatile across early epochs with Stu- compared to the other models. Their final scores were lower and thus there were fewer dents on 1, 2 and 10.

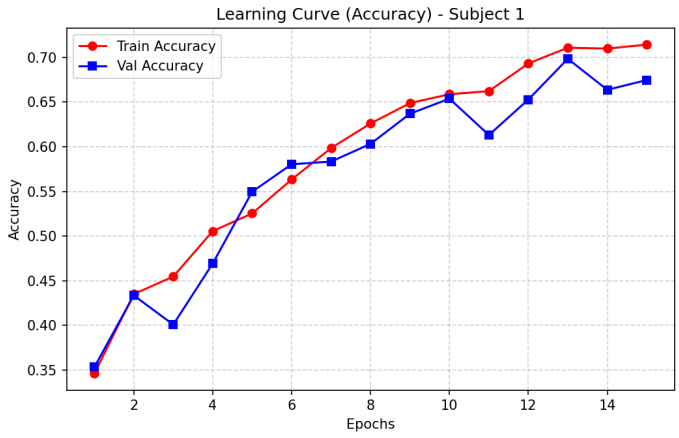


Fig. 5.69 Training and validation accuracy curves for Student 1 (EEG-Deformer++)

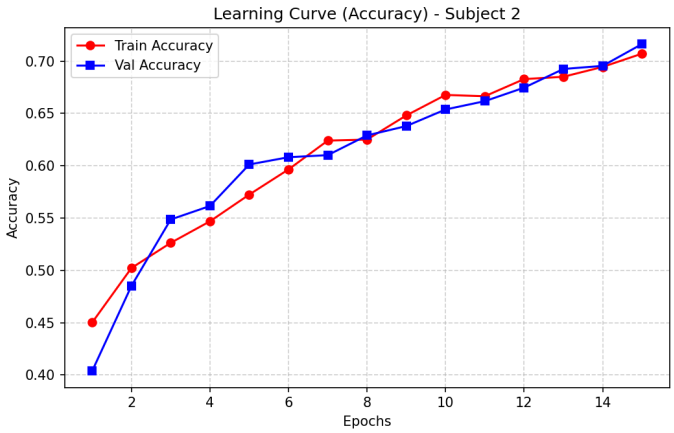


Fig. 5.70 Training and validation accuracy curves for Student 2 (EEG-Deformer++)

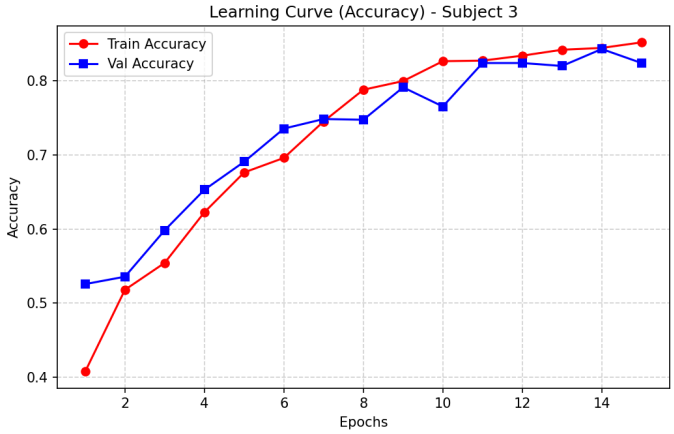


Fig. 5.71 Training and validation accuracy curves for Student 3 (EEG-Deformer++)

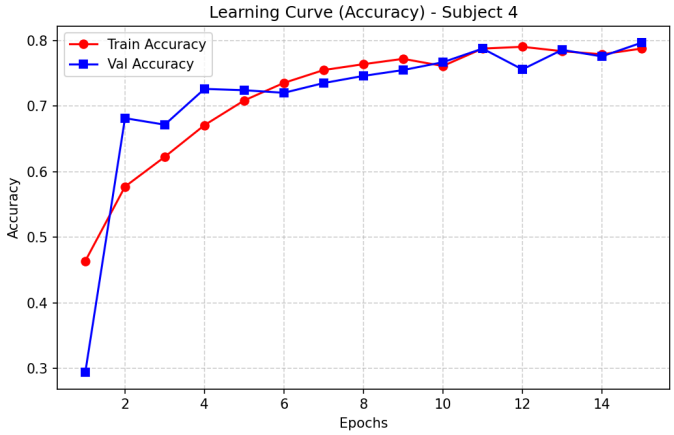


Fig. 5.72 Training and validation accuracy curves for Student 4 (EEG-Deformer++)

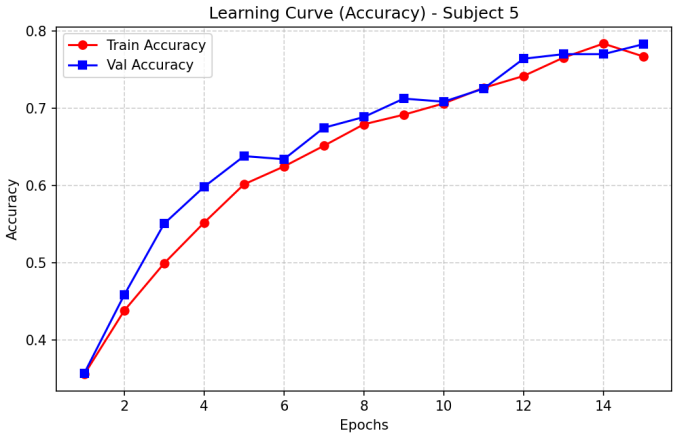


Fig. 5.73 Training and validation accuracy curves for Student 5 (EEG-Deformer++)

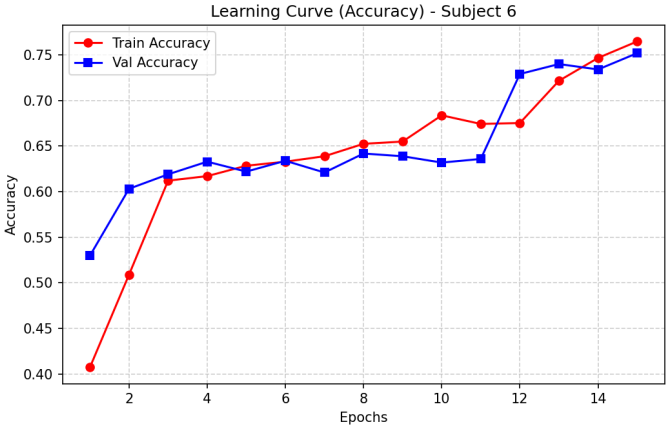


Fig. 5.74 Training and validation accuracy curves for Student 6 (EEG-Deformer++)

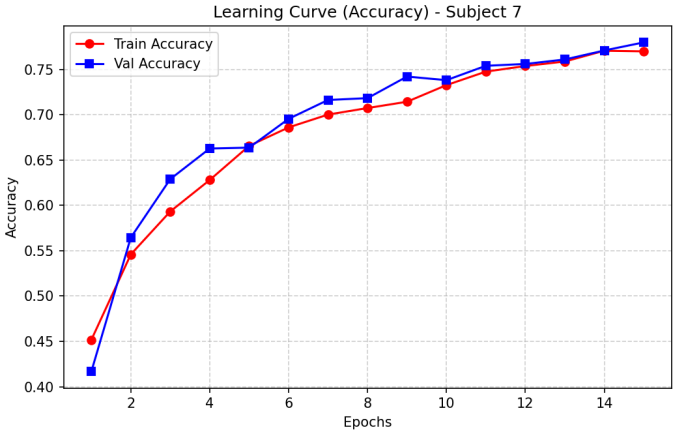


Fig. 5.75 Training and validation accuracy curves for Student 7 (EEG-Deformer++)

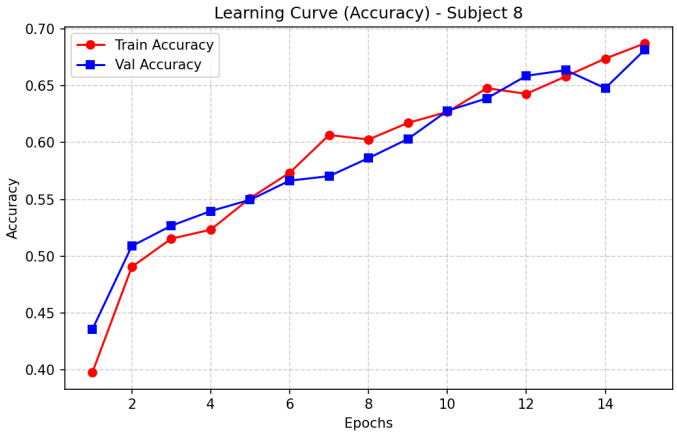


Fig. 5.76 Training and validation accuracy curves for Student 8 (EEG-Deformer++)

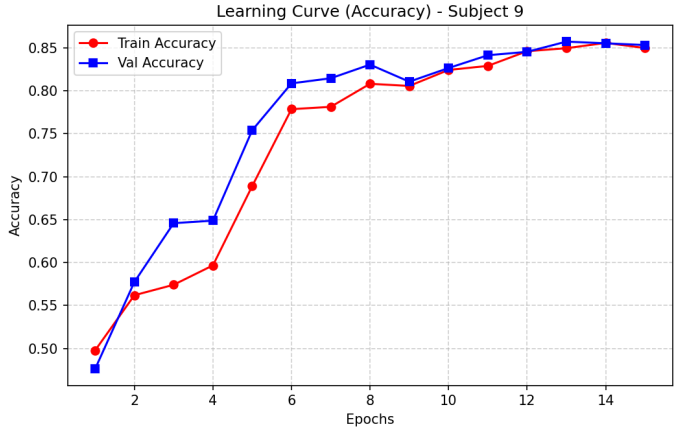


Fig. 5.77 Training and validation accuracy curves for Student 9 (EEG-Deformer++)

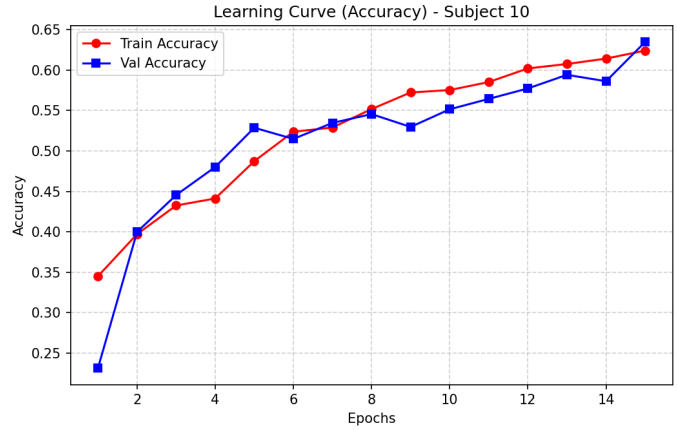


Fig. 5.78 Training and validation accuracy curves for Student 10 (EEG-Deformer++)

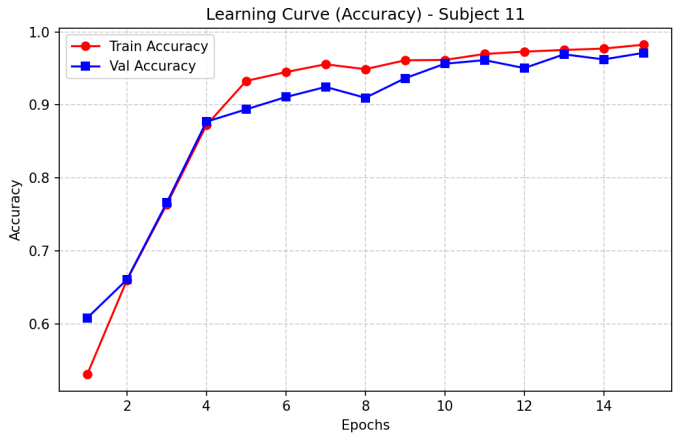


Fig. 5.79 Training and validation accuracy curves for Student 11 (EEG-Deformer++)

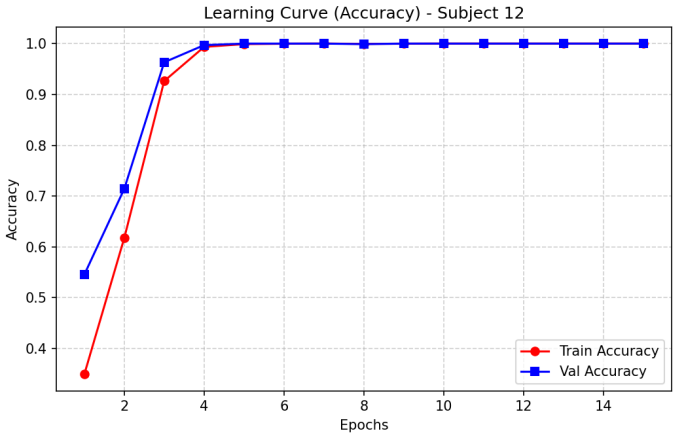


Fig. 5.80 Training and validation accuracy curves for Student 12 (EEG-Deformer++)

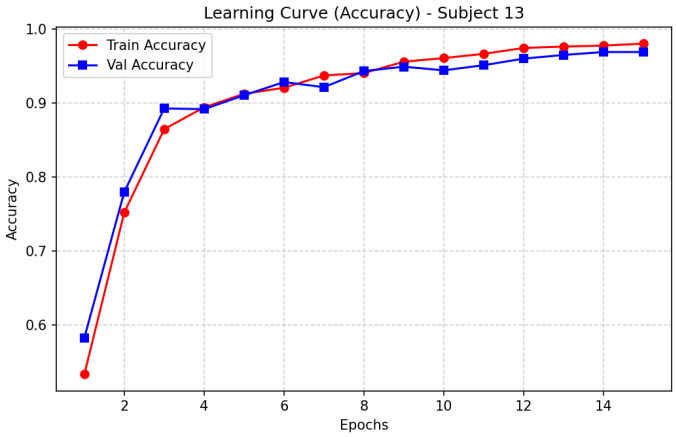


Fig. 5.81 Training and validation accuracy curves for Student 13 (EEG-Deformer++)

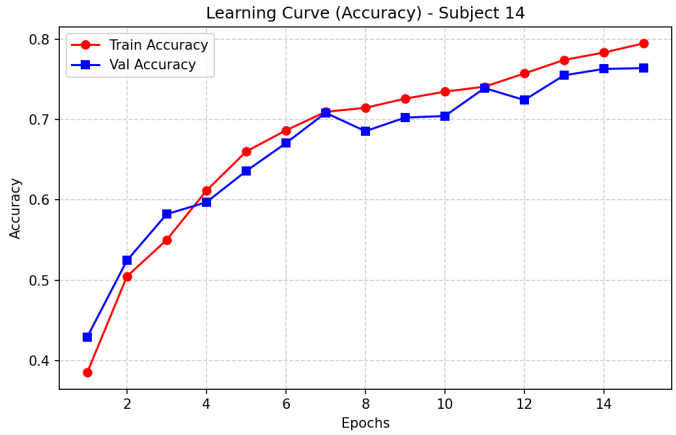


Fig. 5.82 Training and validation accuracy curves for Student 14 (EEG-Deformer++)

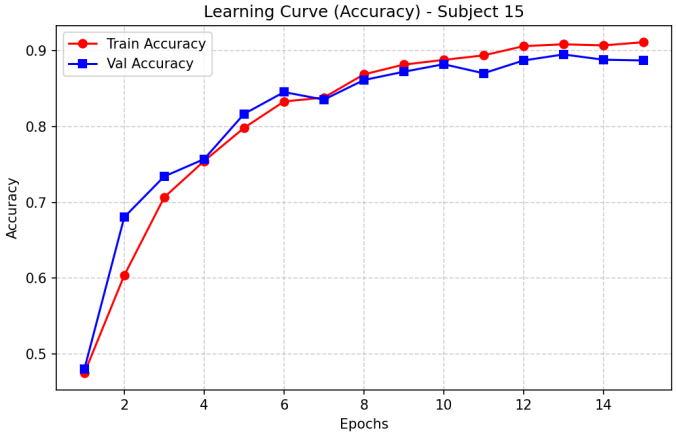


Fig. 5.83 Training and validation accuracy curves for Student 15 (EEG-Deformer++)

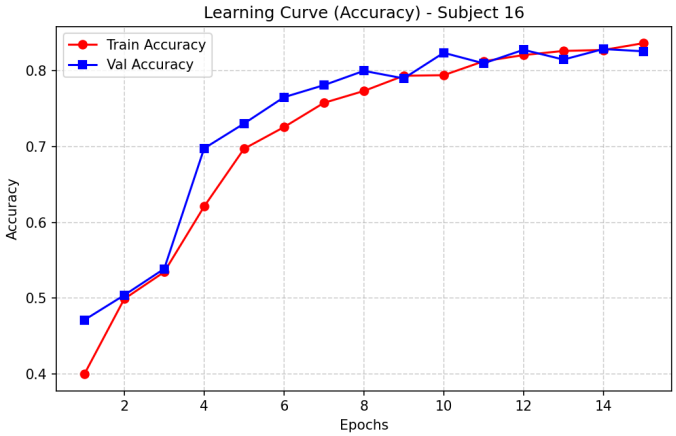


Fig. 5.84 Training and validation accuracy curves for Student 16 (EEG-Deformer++)

5.4.3 Subject-wise Loss Curves

Figs. 5.85 through 5.100 display the training and validation loss curves for EEG-Deformer++. For most subjects, both curves decreased steadily and remained close together, indicating that the model did not overfit severely. A modest divergence in validation loss was observed for Students 8 and 10 in later epochs, which may partly explain why those subjects produced lower accuracy than the group average.

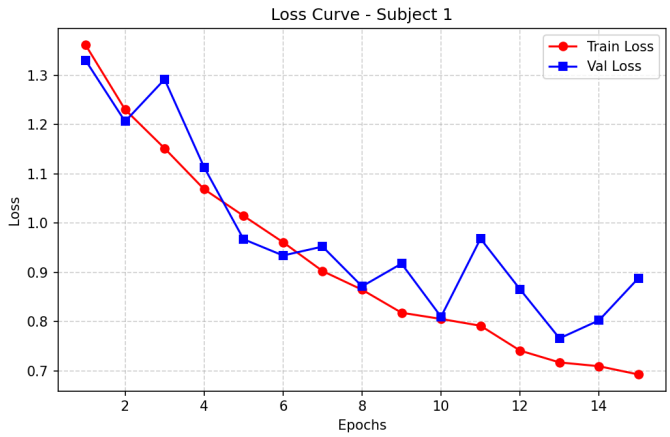


Fig. 5.85 Training and validation loss curves for Student 1 (EEG-Deformer++)

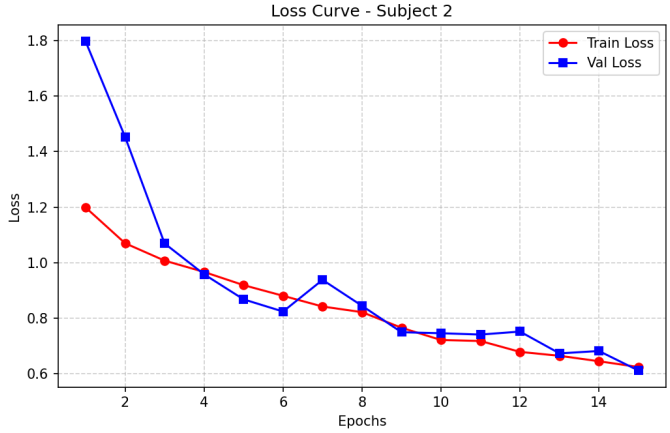


Fig. 5.86 Training and validation loss curves for Student 2 (EEG-Deformer++)

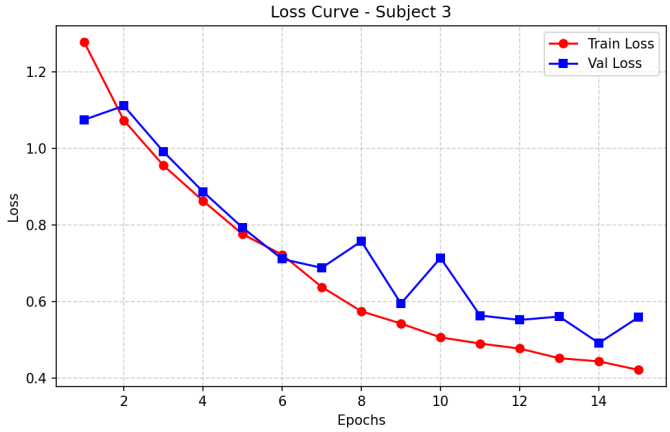


Fig. 5.87 Training and validation loss curves for Student 3 (EEG-Deformer++)

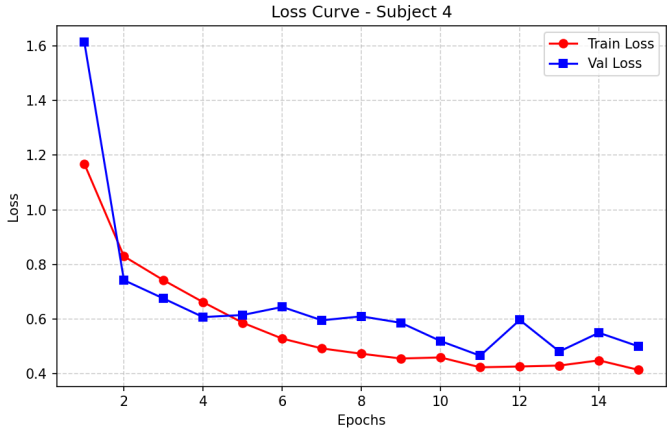


Fig. 5.88 Training and validation loss curves for Student 4 (EEG-Deformer++)

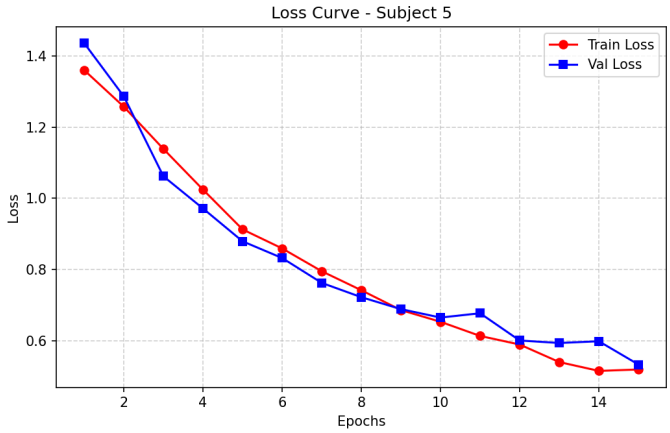


Fig. 5.89 Training and validation loss curves for Student 5 (EEG-Deformer++)

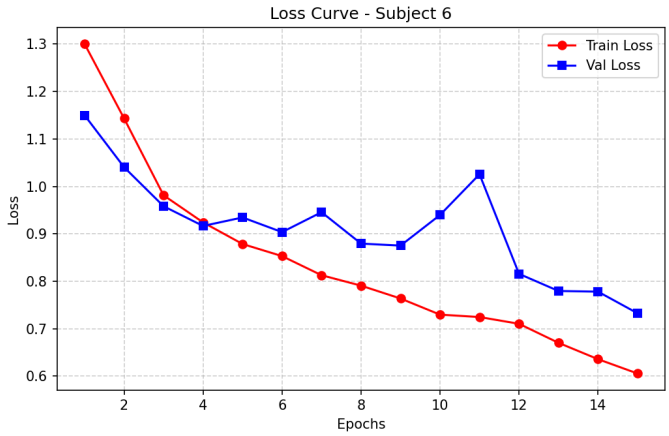


Fig. 5.90 Training and validation loss curves for Student 6 (EEG-Deformer++)

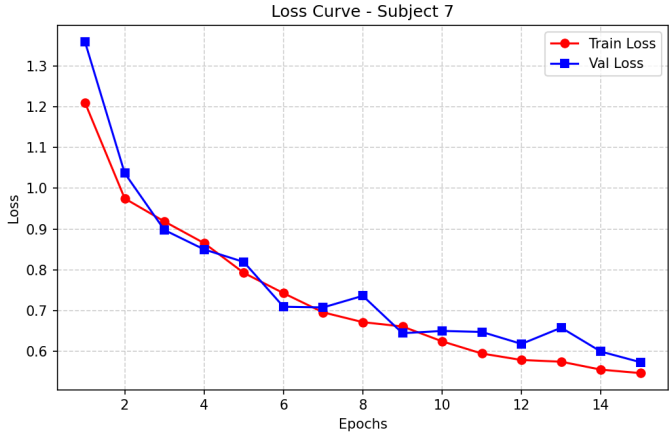


Fig. 5.91 Training and validation loss curves for Student 7 (EEG-Deformer++)

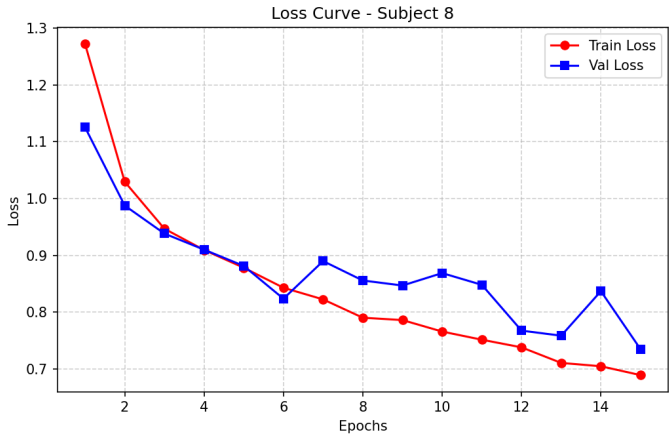


Fig. 5.92 Training and validation loss curves for Student 8 (EEG-Deformer++)

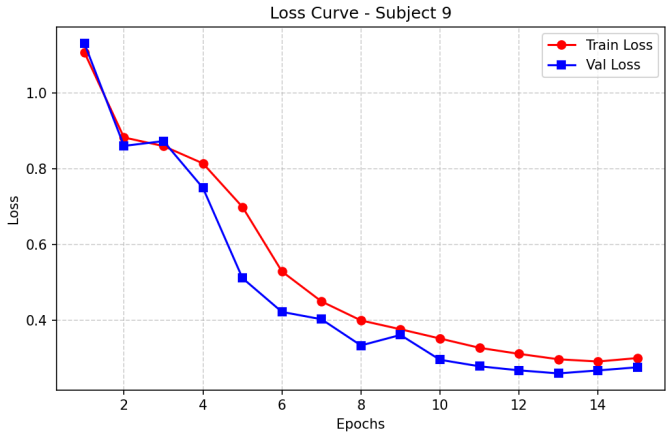


Fig. 5.93 Training and validation loss curves for Student 9 (EEG-Deformer++)

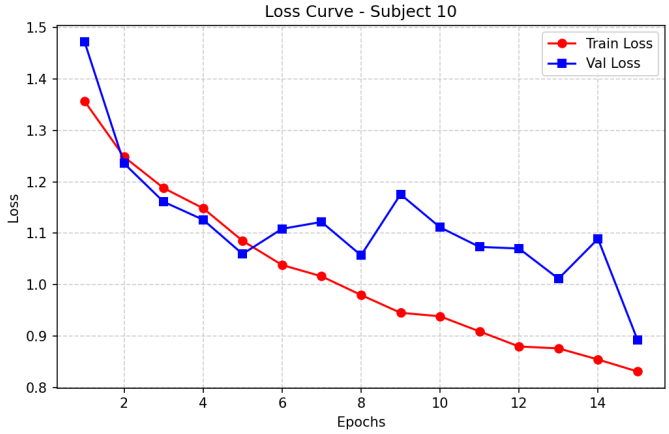


Fig. 5.94 Training and validation loss curves for Student 10 (EEG-Deformer++)

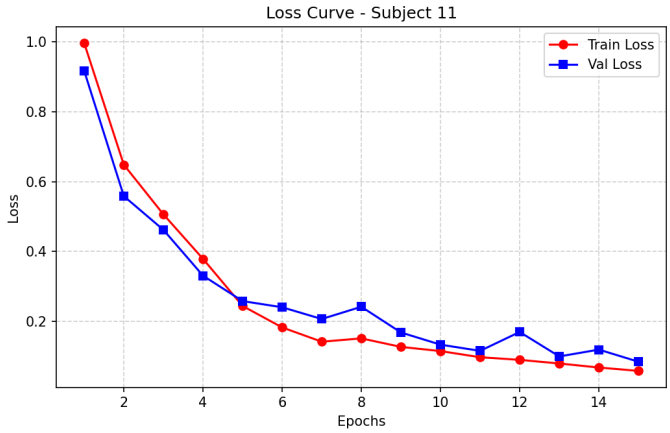


Fig. 5.95 Training and validation loss curves for Student 11 (EEG-Deformer++)

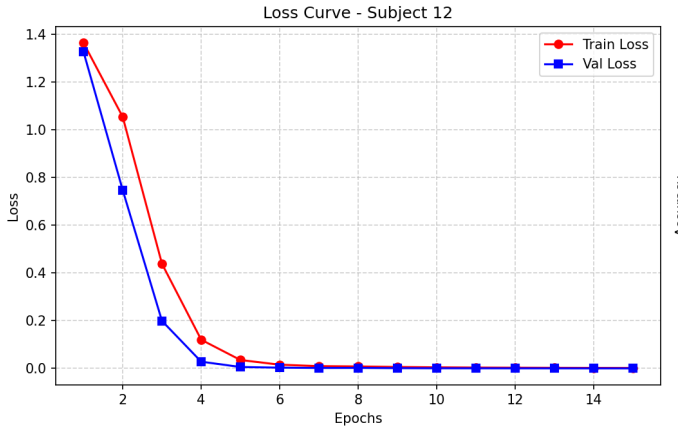


Fig. 5.96 Training and validation loss curves for Student 12 (EEG-Deformer++)

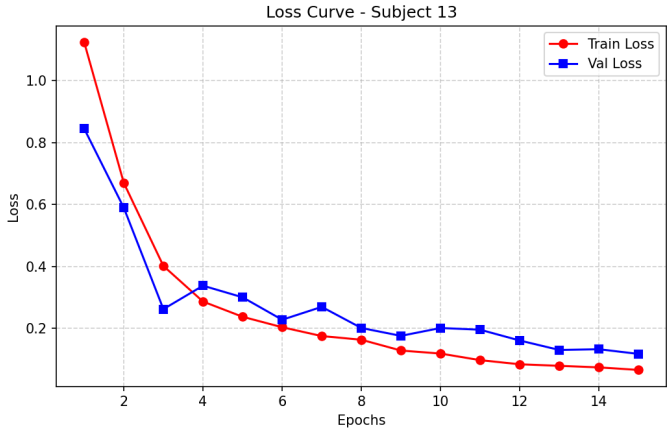


Fig. 5.97 Training and validation loss curves for Student 13 (EEG-Deformer++)

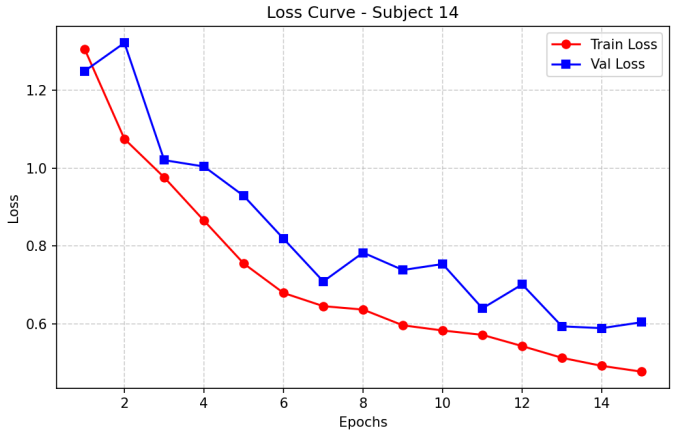


Fig. 5.98 Training and validation loss curves for Student 14 (EEG-Deformer++)

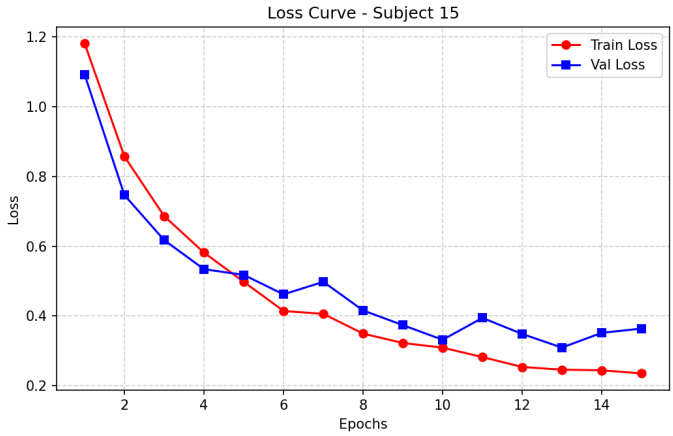


Fig. 5.99 Training and validation loss curves for Student 15 (EEG-Deformer++)

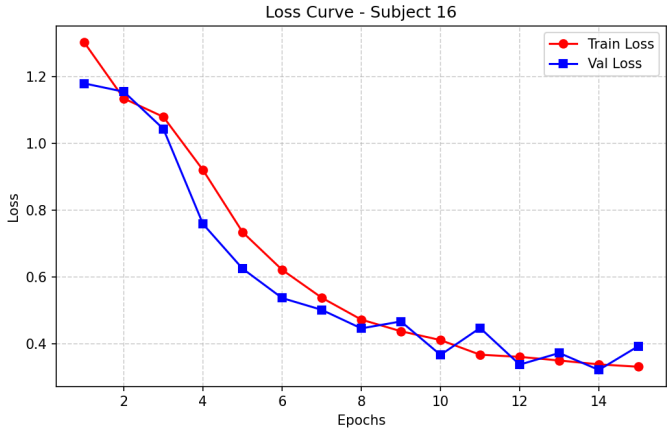


Fig. 5.100 Training and validation loss curves for Student 16 (EEG-Deformer++)

5.5 Results of EEG-DINO++

EEG-DINO++ applies a self-supervised DINO-style pretraining framework to EEG classification. A teacher-student network is used during pretraining on the unlabelled portions of the dataset, after which the learned representations are fine-tuned for the four-class cognitive load task. This approach is evaluated here to examine whether self-supervised pretraining can compensate for the limited labelled EEG data available in a naturalistic classroom setting.

5.5.1 Hyperparameter Configuration

Table 5.5 shows the hyperparameter configuration for EEG-DINO++. Pretraining used The DINO loss with a teacher momentum of 0.996 and a centering coefficient of 0.9

was used. The fine tuning of the model was done using the Adam optimizer and it was done with a learning rate of 0.0003 for 15 epochs. The backbone was a 4-layer Transformer with 8 attention heads per layer. Patch size was It is set to 50 time points and gets 10 patches per 500-sample EEG epoch.

Table 5.5: Hyperparameter configuration for EEG-DINO++

Hyperparameter	**Value**
Pretraining loss	DINO self-distillation loss
Teacher momentum	0.996
Centering coefficient	0.9
Pretraining epochs	15
Fine tuning optimizer	Adam
Fine tuning learning rate	0.0003
Fine tuning epochs	100
Transformer backbone layers	4
Attention heads	8
Patch size (time points)	50
Batch size	32
Dropout rate	0.4
Cross-validation	5-fold subject-independent

5.5.2 Overall Confusion Matrix Analysis

Fig. 5.100 presents the overall confusion matrix for EEG-DINO++ aggregated across all subjects and folds. The matrix is notably less diagonal than those of EEG-HybridFormer and EEG-Deformer++. Predictions for the Easy and Medium classes were distributed almost uniformly across all four labels for many subjects, indicating that the fine-tuned representations did not distinguish these mid-range cognitive states reliably.

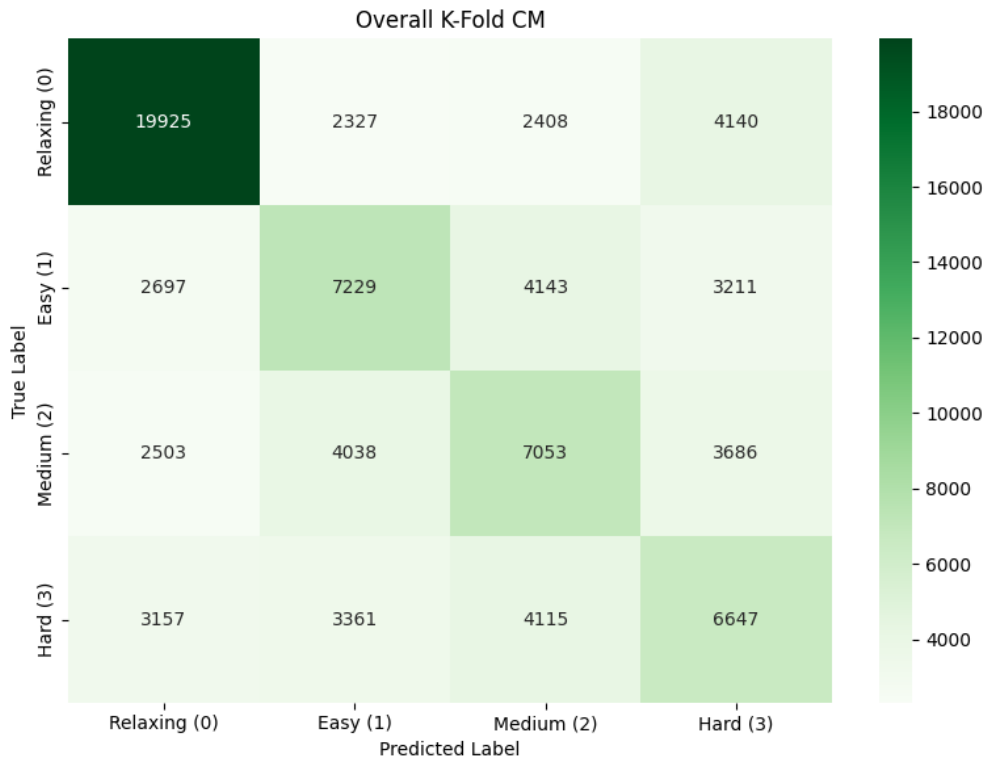


Fig. 5.101 Overall normalized confusion matrix for EEG-DINO++ (aggregated across all subjects and folds)

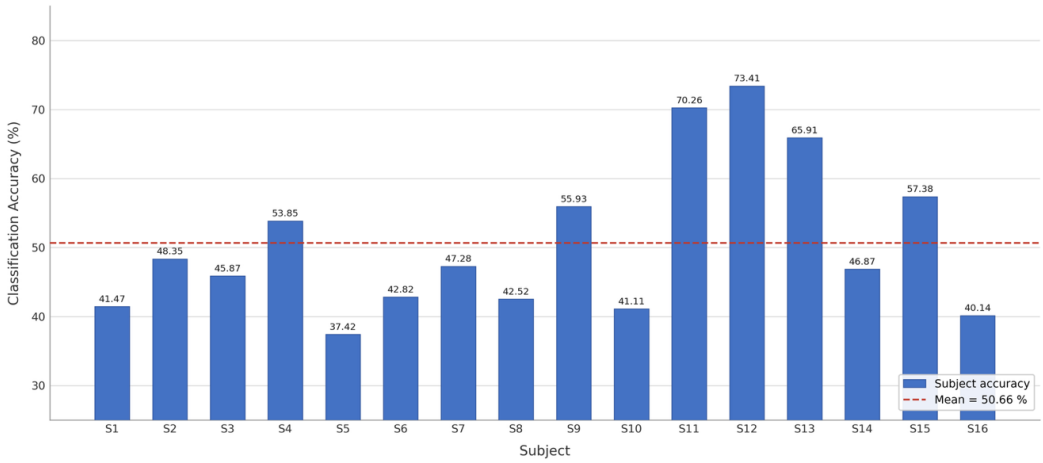


Fig. 5.102 Subject-wise accuracy bar chart for EEG-DINO++

5.5.3 Subject-wise Performance

The results of the EEG-DINO++ are provided in Table 5.6, which shows the performance in each subject. The model produced a mean accuracy of 50.66%, which was

substantially below both EEG-HybridFormer and EEG-Deformer++. The most accurate was Student 12 (73.41%) and the lowest was Student 1 (36.41%). est was 37.42% for Student 5. The results for several subjects were close to random chance (25% for the first five, 50% for the sixth). The DINO self-supervised pretraining on EEG data did not help, implying that the low accuracy of the pretraining on EEG data is not enough to improve the performance of the model on the 4-class task. not learn representations that carried over well to the fine-tuning classification task Subjective, given the size of the data set.

Table 5.6: Subject-wise performance of EEG-DINO++

Subject	**Accuracy (%)**	**Precision**	**Recall**	**F1-Score**
Student 1	41.47	0.38	0.38	0.38
Student 2	48.35	0.42	0.43	0.42
Student 3	45.87	0.43	0.42	0.42
Student 4	53.85	0.48	0.48	0.48
Student 5	37.42	0.35	0.35	0.35
Student 6	42.82	0.46	0.44	0.43
Student 7	47.28	0.42	0.43	0.42
Student 8	42.52	0.38	0.38	0.38
Student 9	55.93	0.53	0.53	0.53
Student 10	41.11	0.38	0.38	0.37
Student 11	70.26	0.72	0.70	0.67
Student 12	73.41	0.75	0.72	0.72
Student 13	65.91	0.63	0.64	0.63
Student 14	46.87	0.40	0.40	0.39
Student 15	57.38	0.52	0.54	0.53
Student 16	40.14	0.39	0.40	0.39
Mean	**50.66**	**0.48**	**0.48**	**0.47**

5.6 Subject-wise Results of EEG-DINO++

The following sections present subject-wise confusion matrices and training curves for EEG-DINO++. The results of this model are organised in the same order as done for the Section 5.3.3.

5.6.1 Subject-wise Confusion Matrices

Fig. 5.101 provides the subject-wise accuracy bar chart for EEG-DINO++. Figs. 5.102 through 5.117 show the normalized confusion matrices per subject. The confusion pat-

terns are more diffuse than for EEG-HybridFormer and EEG-Deformer++, with predictions spread broadly across all four classes for several subjects. Students 11 and 12 were the exceptions, showing more concentrated diagonal entries that correspond to their comparatively higher accuracy scores.

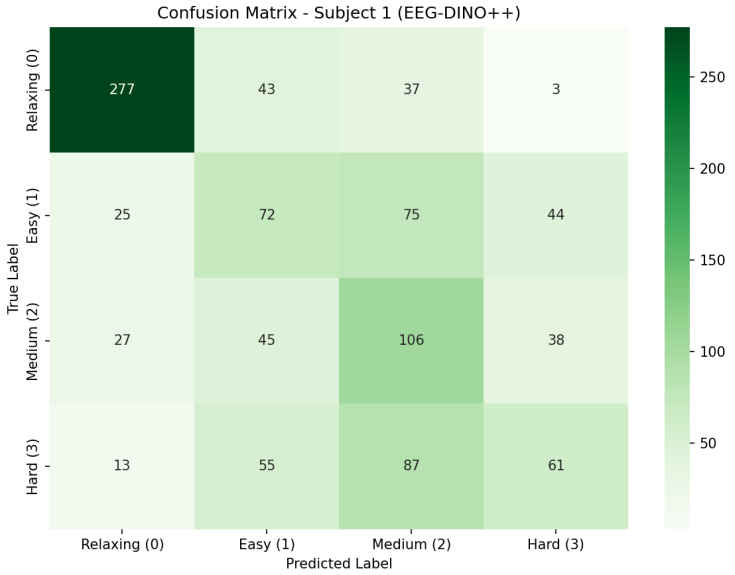


Fig. 5.103 Normalized confusion matrix for Student 1 (EEG-DINO++)

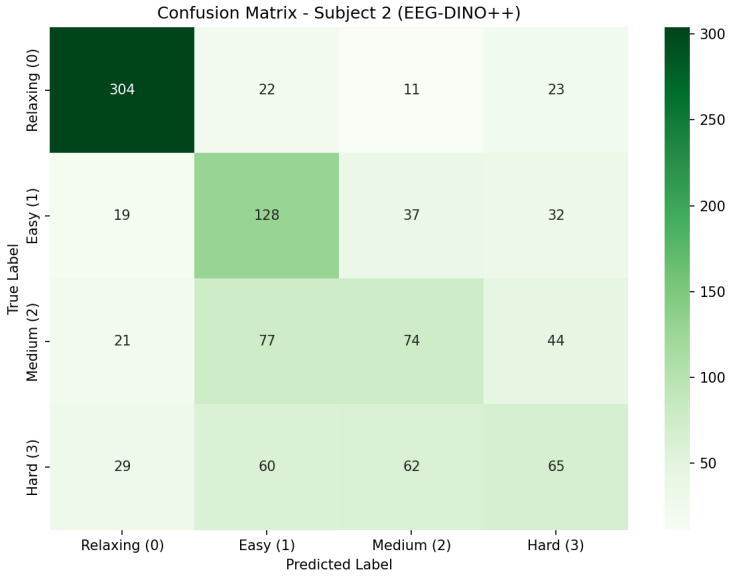


Fig. 5.104 Normalized confusion matrix for Student 2 (EEG-DINO++)

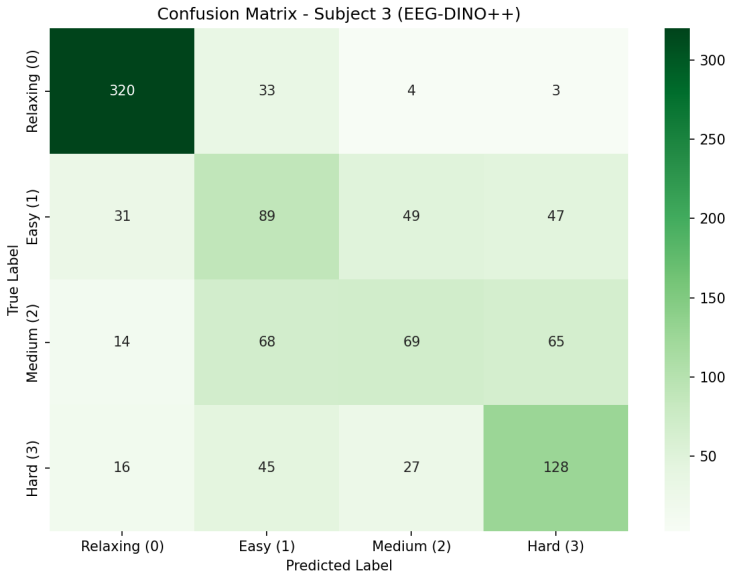


Fig. 5.105 Normalized confusion matrix for Student 3 (EEG-DINO++)

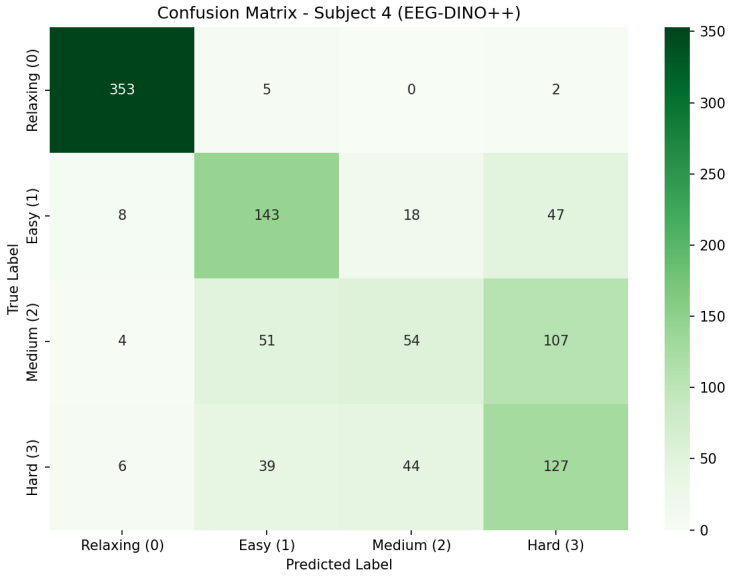


Fig. 5.106 Normalized confusion matrix for Student 4 (EEG-DINO++)

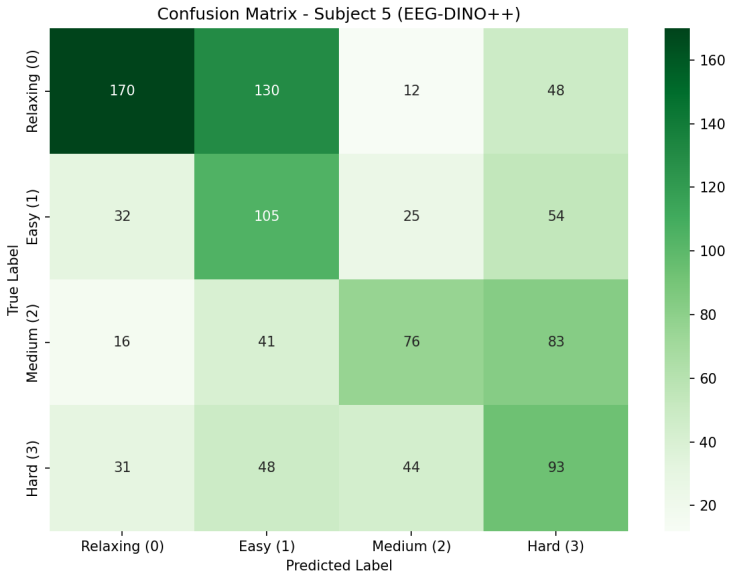


Fig. 5.107 Normalized confusion matrix for Student 5 (EEG-DINO++)

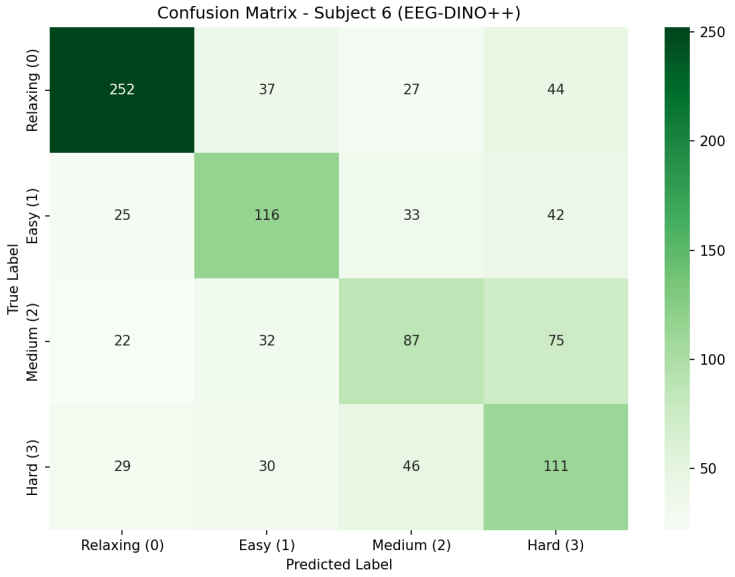


Fig. 5.108 Normalized confusion matrix for Student 6 (EEG-DINO++)

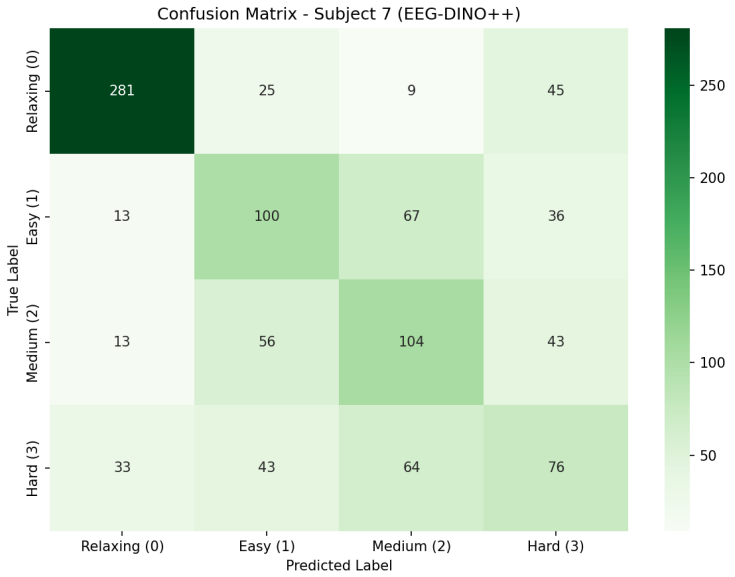


Fig. 5.109 Normalized confusion matrix for Student 7 (EEG-DINO++)

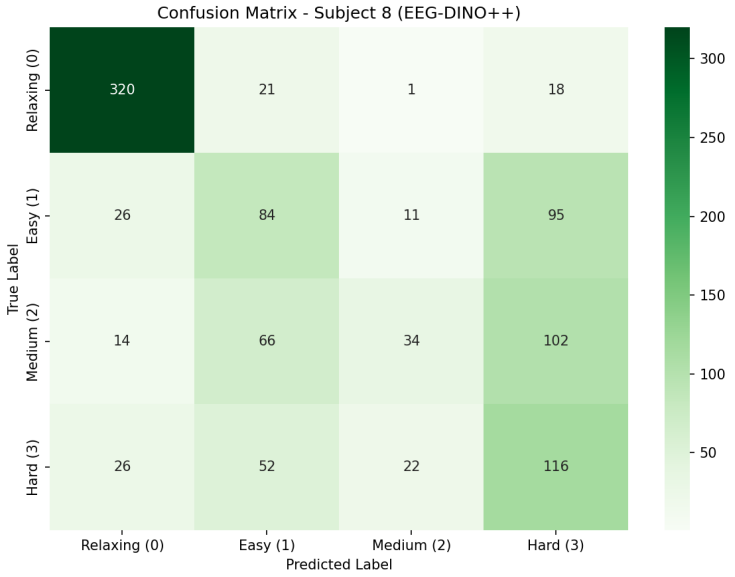


Fig. 5.110 Normalized confusion matrix for Student 8 (EEG-DINO++)

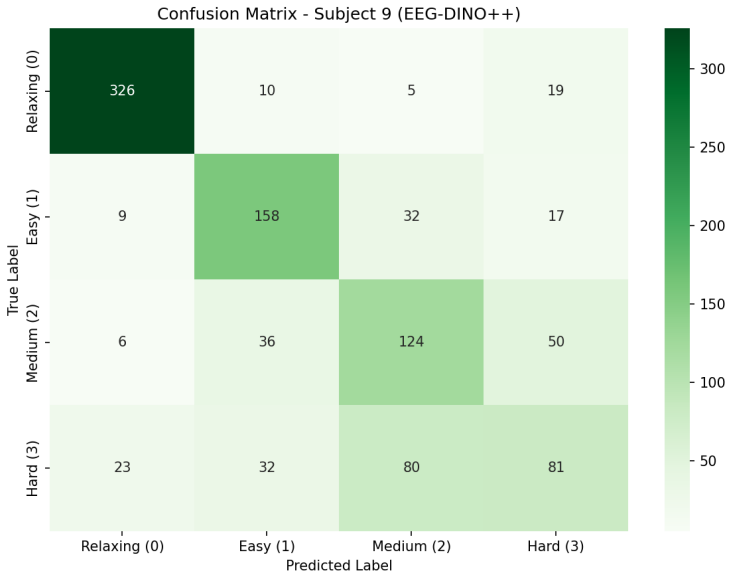


Fig. 5.111 Normalized confusion matrix for Student 9 (EEG-DINO++)

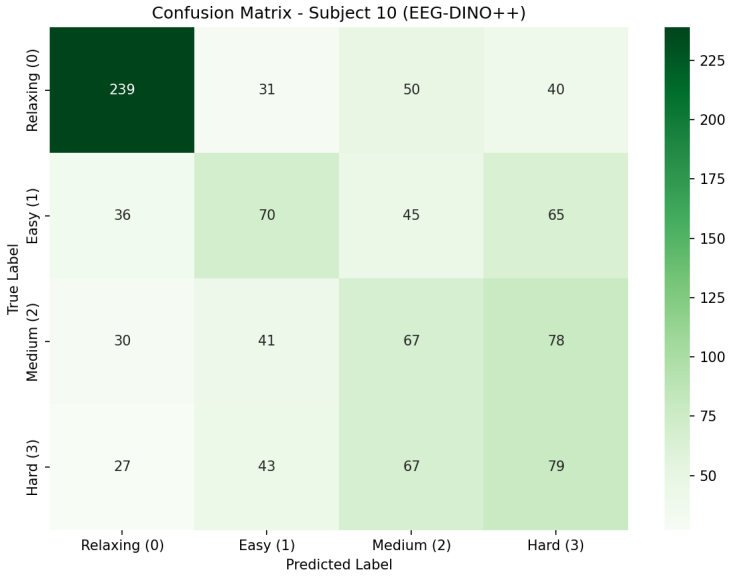


Fig. 5.112 Normalized confusion matrix for Student 10 (EEG-DINO++)

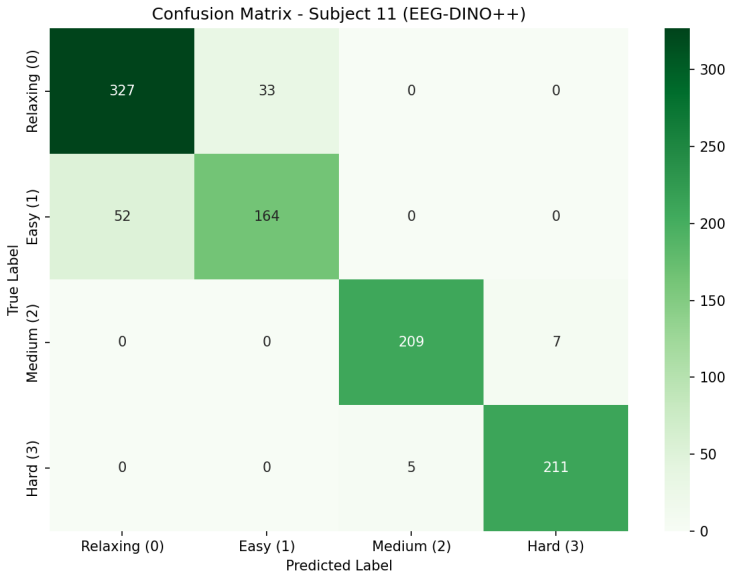


Fig. 5.113 Normalized confusion matrix for Student 11 (EEG-DINO++)

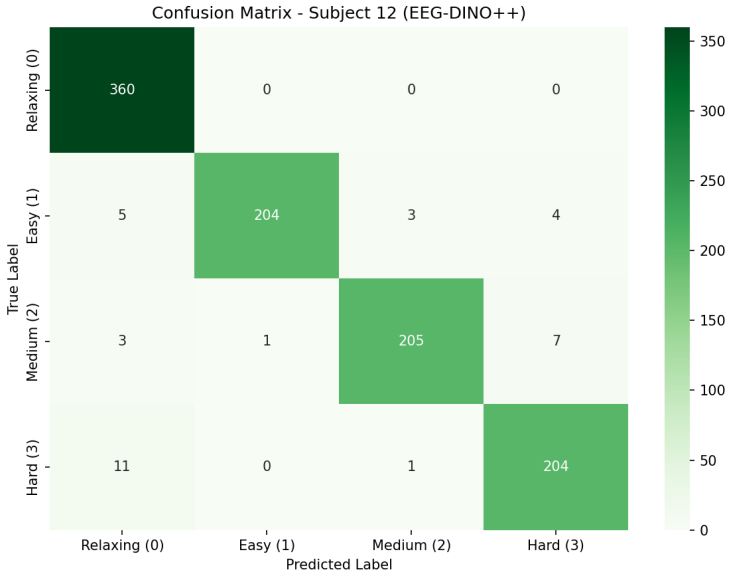


Fig. 5.114 Normalized confusion matrix for Student 12 (EEG-DINO++)

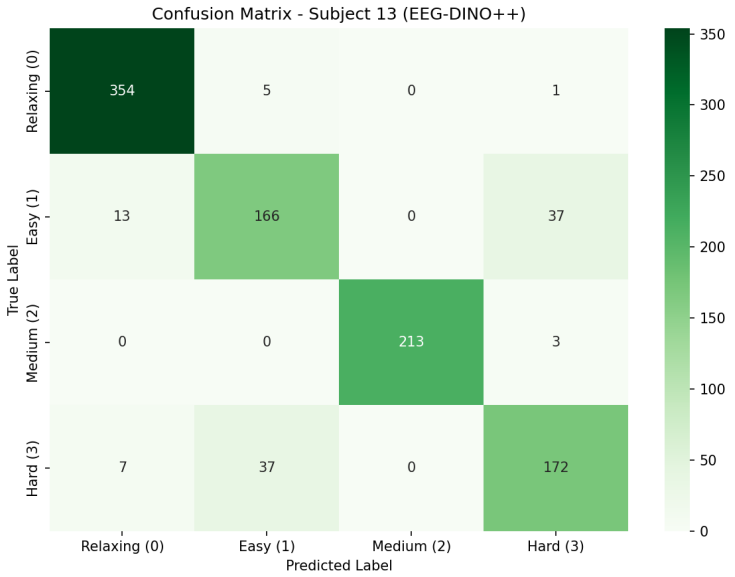


Fig. 5.115 Normalized confusion matrix for Student 13 (EEG-DINO++)

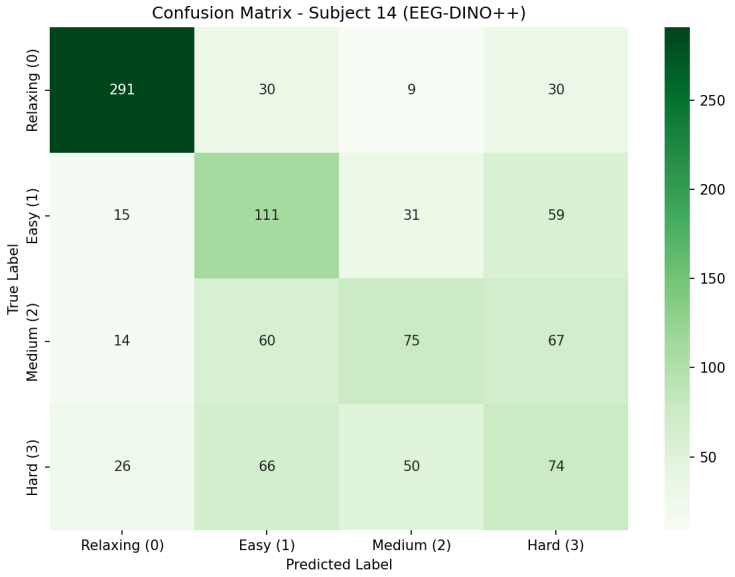


Fig. 5.116 Normalized confusion matrix for Student 14 (EEG-DINO++)

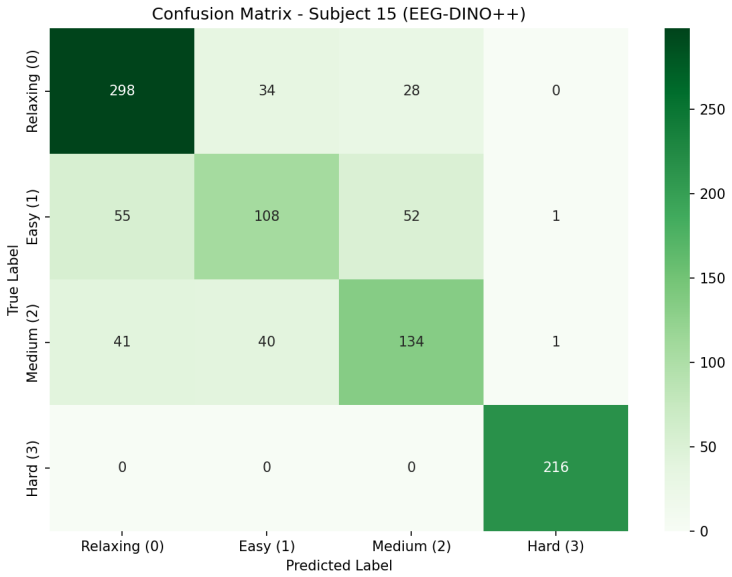


Fig. 5.117 Normalized confusion matrix for Student 15 (EEG-DINO++)

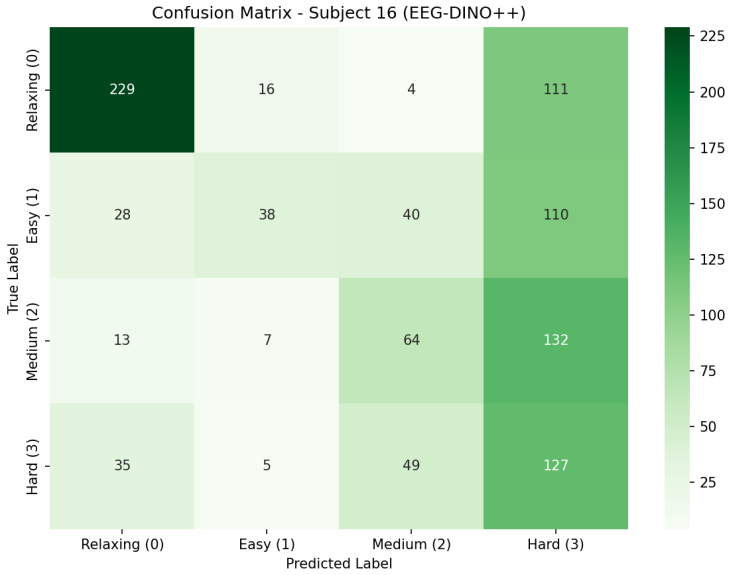


Fig. 5.118 Normalized confusion matrix for Student 16 (EEG-DINO++)

5.6.2 Subject-wise Accuracy Curves

The accuracy curves for training and validation are included for each sub- in Figs. 5.118-5.133. ject for EEG-DINO++. The accuracy increased gradually with the number of epochs used for fine-tuning: most subjects, with little improvement in the validation accuracy, and low accuracy in this category for In subjects that were not as successful, it was close to chance level. For Students 11 and 12, validation accuracy

rose steadily, likely due to those participants producing the best shots. EEG signals with larger class discriminatory structure.

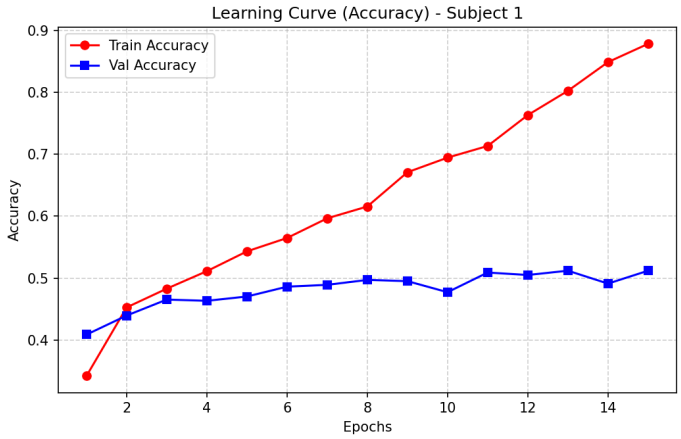


Fig. 5.119 Training and validation accuracy curves for Student 1 (EEG-DINO++)

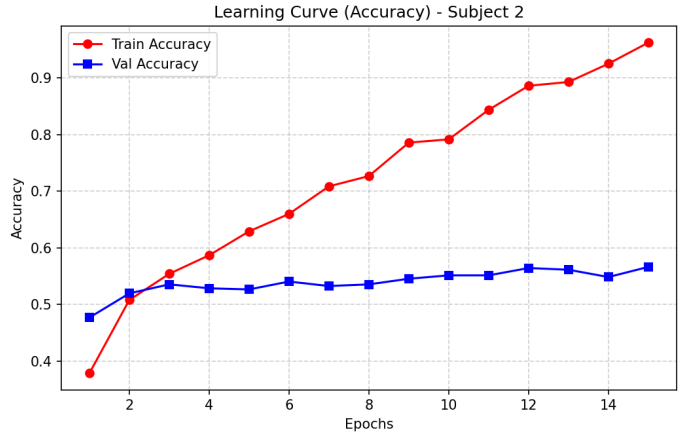


Fig. 5.120 Training and validation accuracy curves for Student 2 (EEG-DINO++)

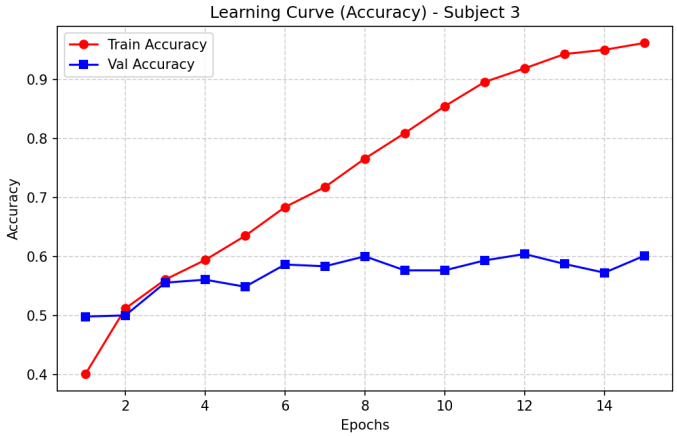


Fig. 5.121 Training and validation accuracy curves for Student 3 (EEG-DINO++)

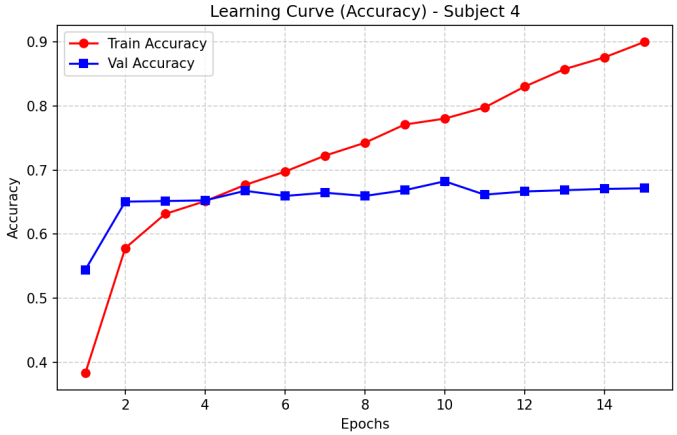


Fig. 5.122 Training and validation accuracy curves for Student 4 (EEG-DINO++)

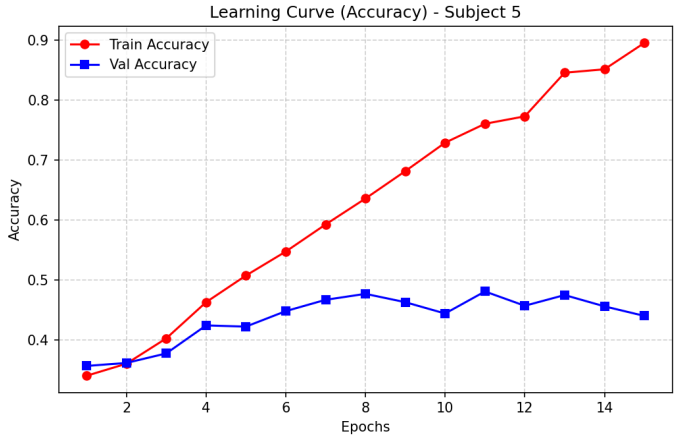


Fig. 5.123 Training and validation accuracy curves for Student 5 (EEG-DINO++)

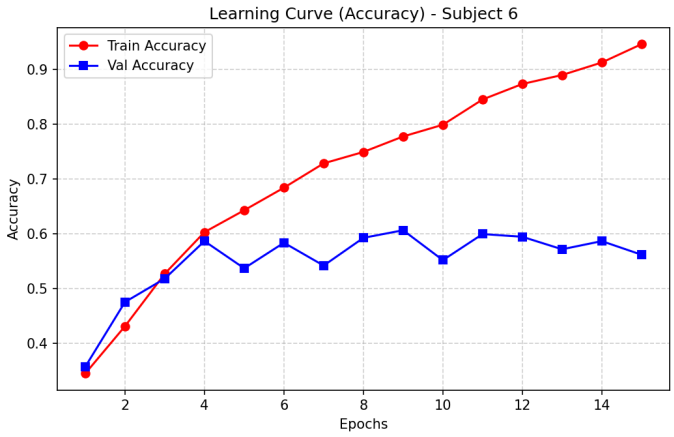


Fig. 5.124 Training and validation accuracy curves for Student 6 (EEG-DINO++)

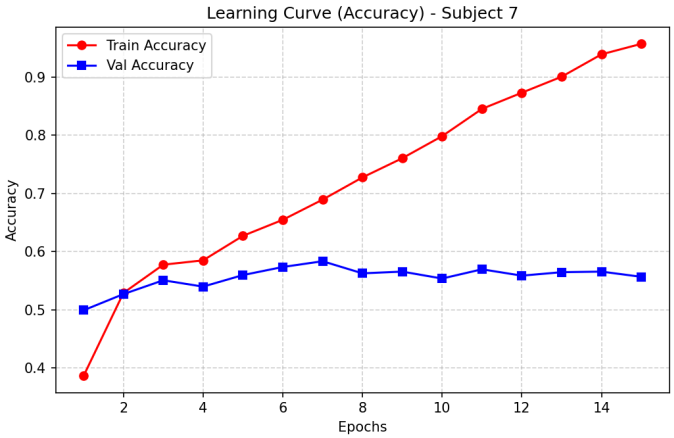


Fig. 5.125 Training and validation accuracy curves for Student 7 (EEG-DINO++)

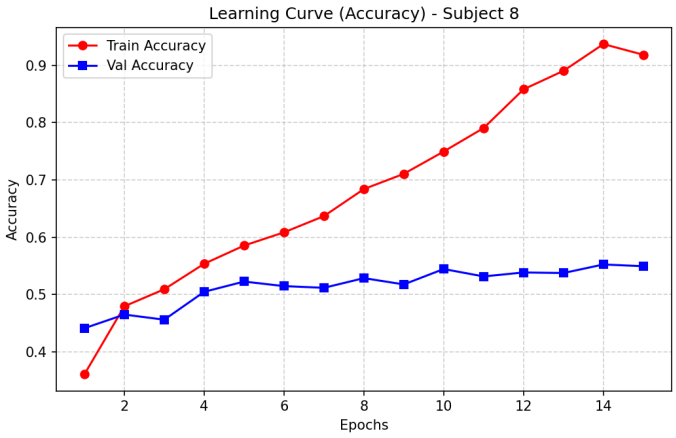


Fig. 5.126 Training and validation accuracy curves for Student 8 (EEG-DINO++)

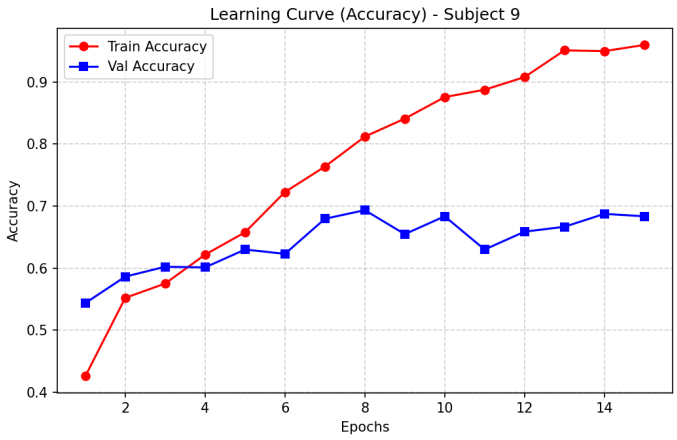


Fig. 5.127 Training and validation accuracy curves for Student 9 (EEG-DINO++)

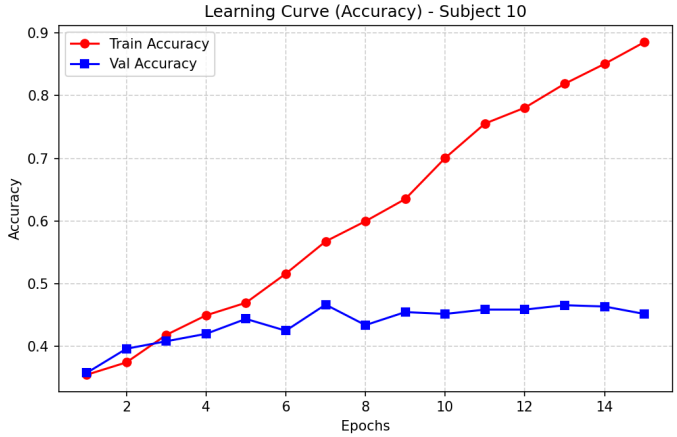


Fig. 5.128 Training and validation accuracy curves for Student 10 (EEG-DINO++)

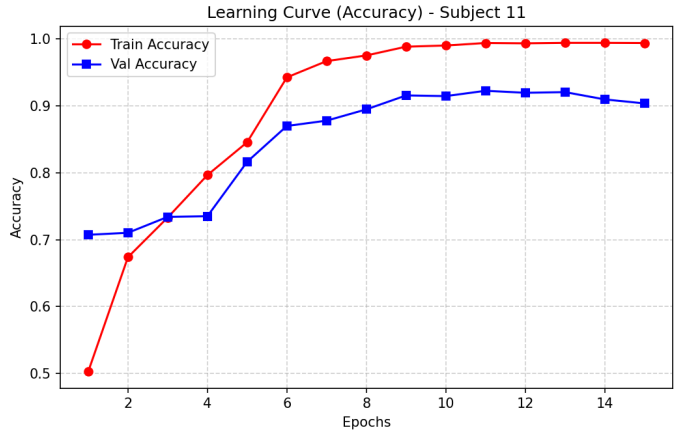


Fig. 5.129 Training and validation accuracy curves for Student 11 (EEG-DINO++)

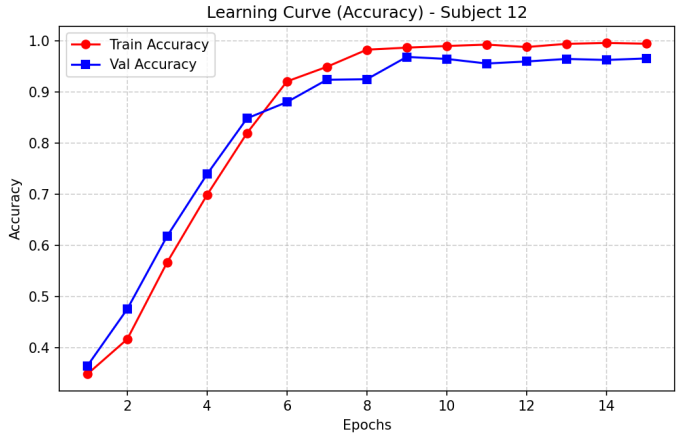


Fig. 5.130 Training and validation accuracy curves for Student 12 (EEG-DINO++)

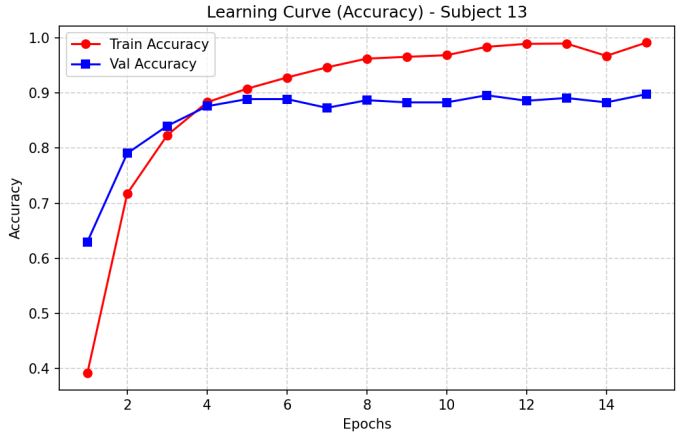


Fig. 5.131 Training and validation accuracy curves for Student 13 (EEG-DINO++)

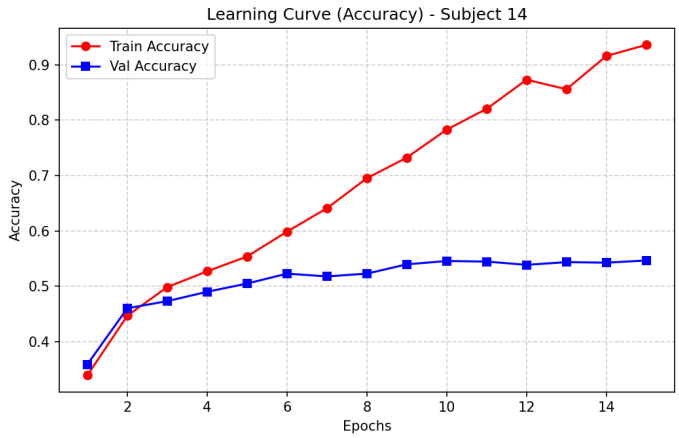


Fig. 5.132 Training and validation accuracy curves for Student 14 (EEG-DINO++)

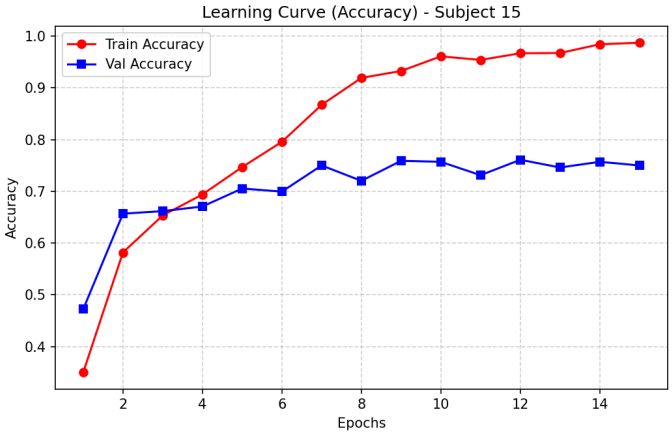


Fig. 5.133 Training and validation accuracy curves for Student 15 (EEG-DINO++)

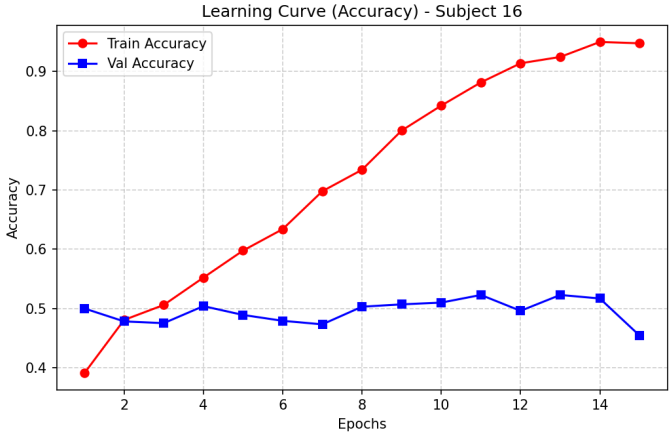


Fig. 5.134 Training and validation accuracy curves for Student 16 (EEG-DINO++)

5.6.3 Subject-wise Loss Curves

The loss-curves for EEG-DINO++ are shown in figures 5.134-5.149. Fine-tuning loss While the loss for most of the subjects decreased, the loss stayed the same or rose in later epochs for validation. for the majority of participants. This behaviour is indicative of overfitting during fine-tuning, This is something that is well-known as a problem when adapting a large self-supervised model to a small. labelled dataset.

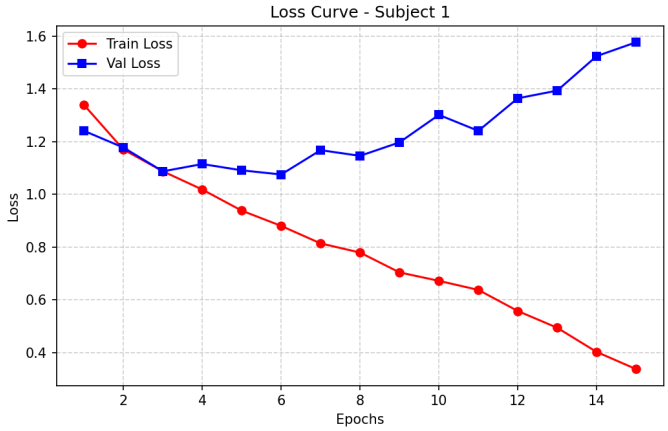


Fig. 5.135 Training and validation loss curves for Student 1 (EEG-DINO++)

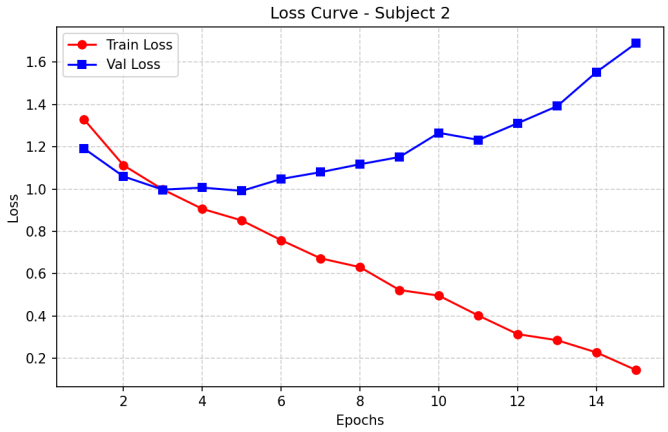


Fig. 5.136 Training and validation loss curves for Student 2 (EEG-DINO++)

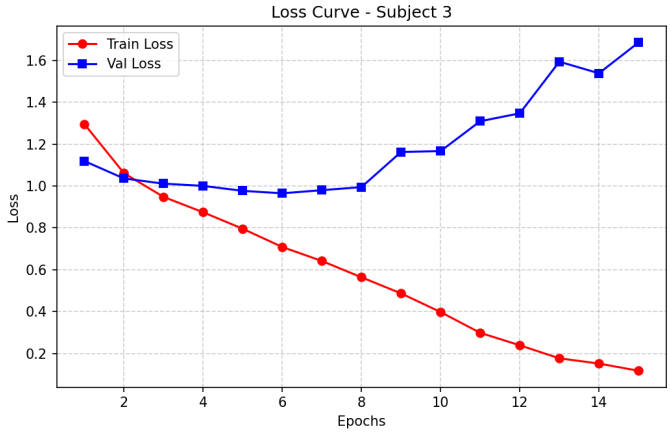


Fig. 5.137 Training and validation loss curves for Student 3 (EEG-DINO++)

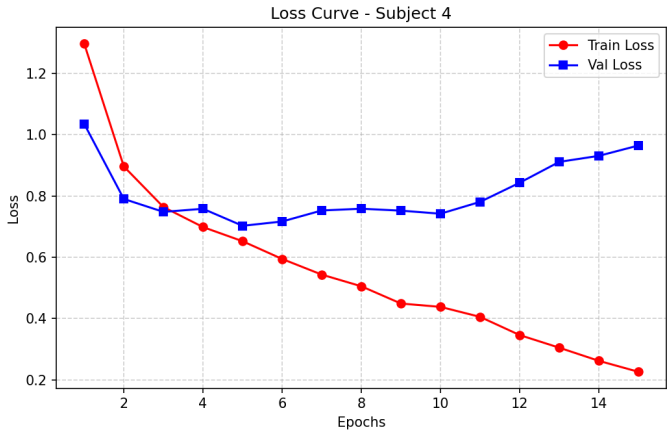


Fig. 5.138 Training and validation loss curves for Student 4 (EEG-DINO++)

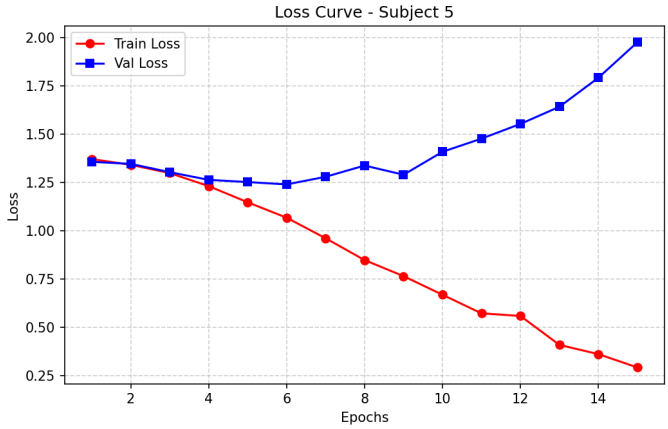


Fig. 5.139 Training and validation loss curves for Student 5 (EEG-DINO++)

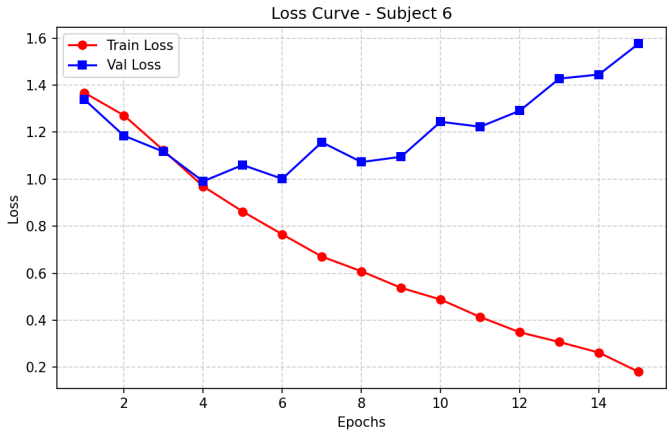


Fig. 5.140 Training and validation loss curves for Student 6 (EEG-DINO++)

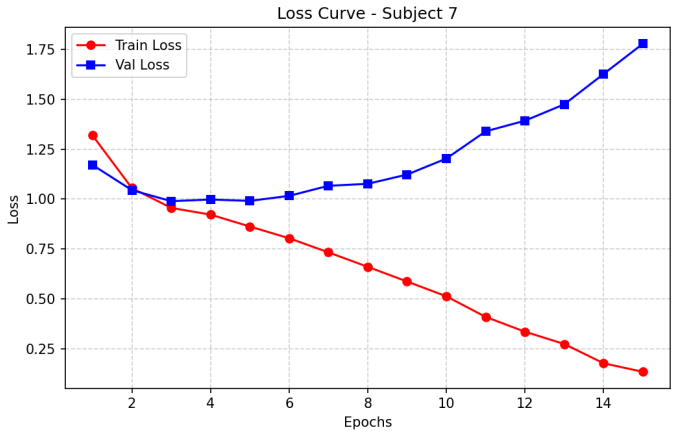


Fig. 5.141 Training and validation loss curves for Student 7 (EEG-DINO++)

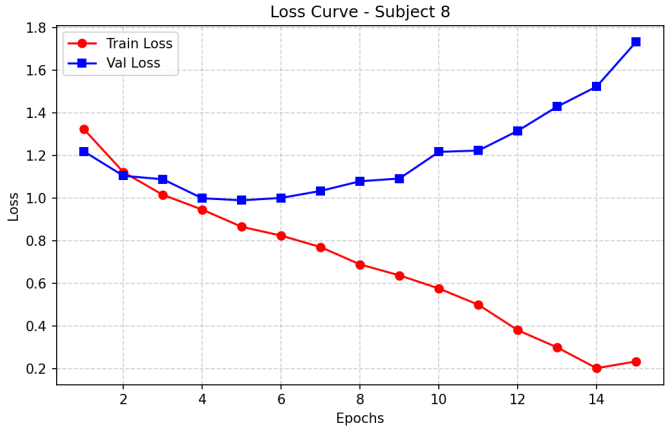


Fig. 5.142 Training and validation loss curves for Student 8 (EEG-DINO++)

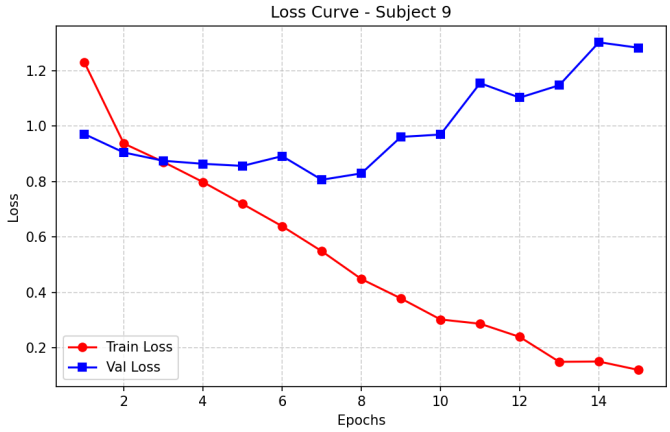


Fig. 5.143 Training and validation loss curves for Student 9 (EEG-DINO++)

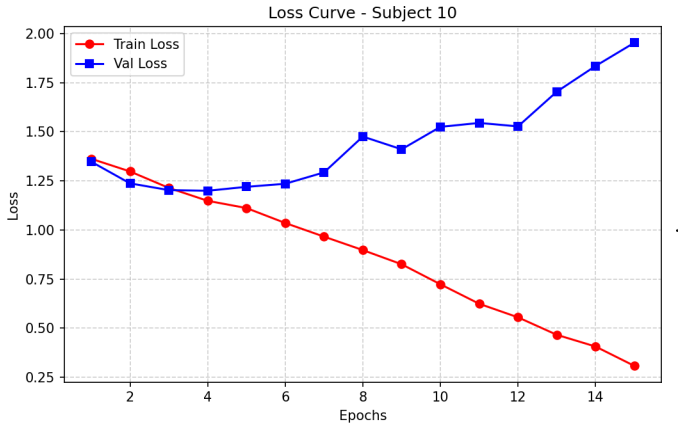


Fig. 5.144 Training and validation loss curves for Student 10 (EEG-DINO++)

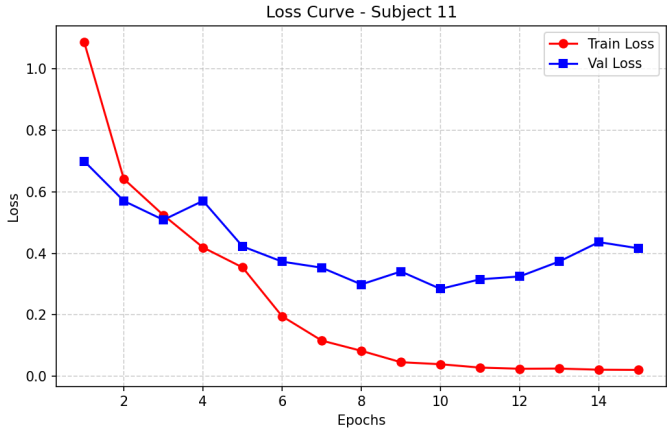


Fig. 5.145 Training and validation loss curves for Student 11 (EEG-DINO++)

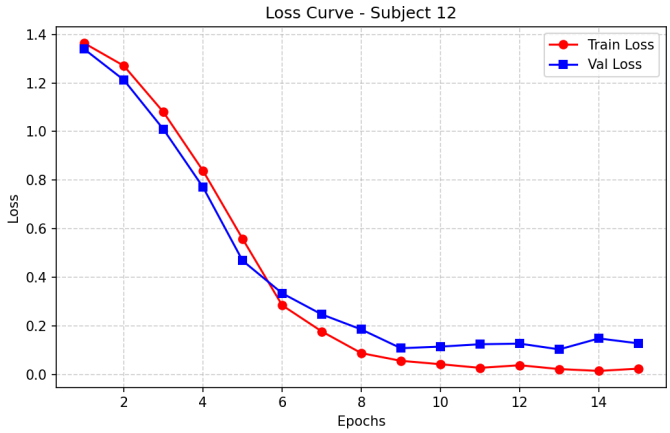


Fig. 5.146 Training and validation loss curves for Student 12 (EEG-DINO++)

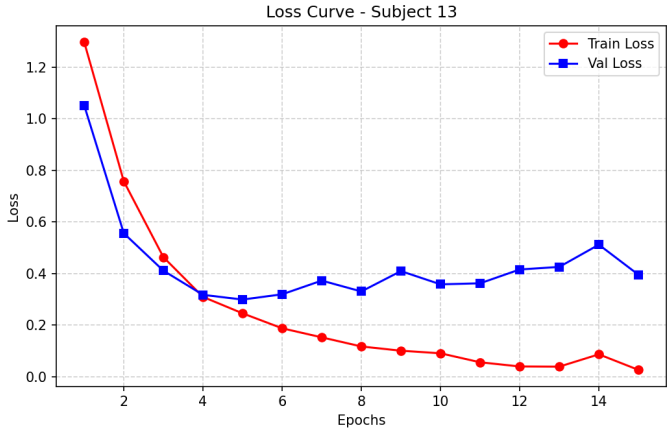


Fig. 5.147 Training and validation loss curves for Student 13 (EEG-DINO++)

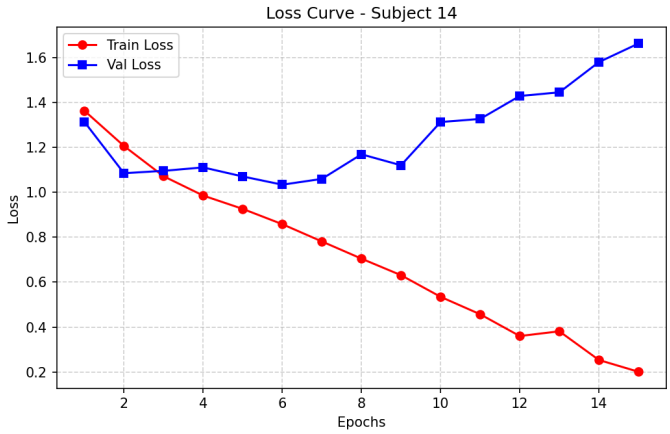


Fig. 5.148 Training and validation loss curves for Student 14 (EEG-DINO++)

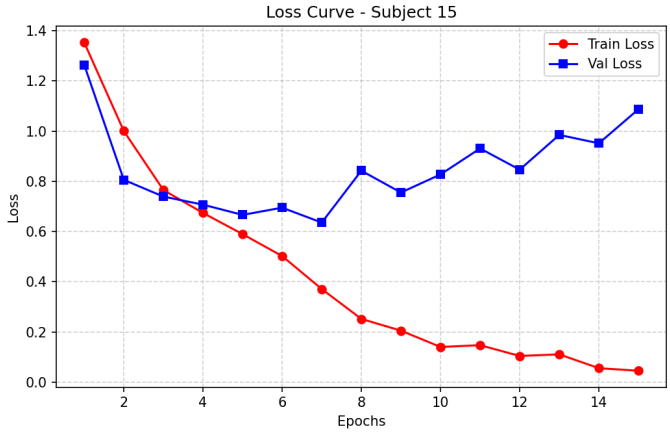


Fig. 5.149 Training and validation loss curves for Student 15 (EEG-DINO++)

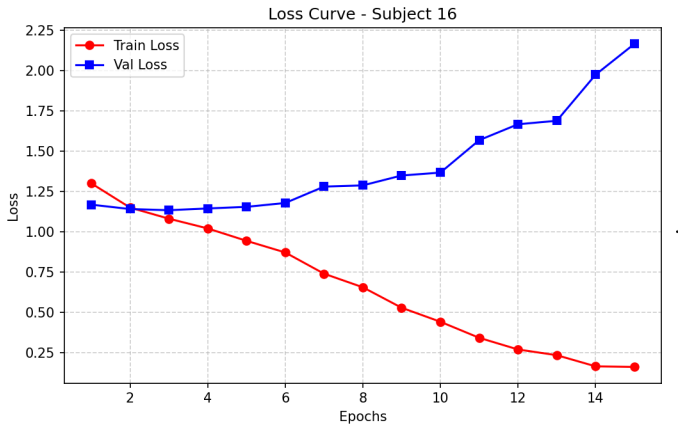


Fig. 5.150 Training and validation loss curves for Student 16 (EEG-DINO++)

5.7 Comparative Performance Analysis

This section compares the three models across subjects, cross-validation folds, and overall averages. The comparison shows the degree to which architectural choice influences cognitive load classification under a subject-independent evaluation.

5.7.1 Subject-wise Performance Comparison

The accuracy of all three models is compared for each of the 16 subjects in Table 5.7. The same data shown in Fig. 5.151 is displayed in a grouped bar chart in Fig. 5.152.

Table 5.7: Subject wise accuracy comparison for all three models

Subject	**EEG-HybridFormer (%)**	**EEG-Deformer++ (%)**	**EEG-DINO++ (%)**
Student 1	73.87	62.10	41.47
Student 2	81.13	71.88	48.35
Student 3	88.57	79.96	45.87
Student 4	83.83	75.20	53.85
Student 5	85.46	80.30	37.42
Student 6	89.07	77.24	42.82
Student 7	86.47	77.24	47.28
Student 8	76.39	71.25	42.52
Student 9	93.15	87.10	55.93
Student 10	70.00	59.56	41.11
Student 11	94.40	93.97	70.26
Student 12	91.77	89.66	73.41

Table 5.7: Subject wise accuracy comparison for all three models

Student 13	98.06	94.27	65.91
Student 14	85.60	75.14	46.87
Student 15	92.30	84.58	57.38
Student 16	80.16	75.73	40.14
Mean	**85.64**	**78.45**	**50.66**

EEG-HybridFormer significantly improved performance than EEG-Deformer++ for 14 of the 16 subjects. The two exceptions were the Students 11 and 13, where the EEG-Deformer++ matched or slightly exceeded the ex-ante by default. ceeded HybridFormer. The accuracy of EEG-DINO++ is the lowest for all 16 subjects.The gap between EEG-HybridFormer and EEG-DINO++ was largest for Students 1, 5, and 10, where DINO++ scored near chance level while HybridFormer still achieved over 70%.

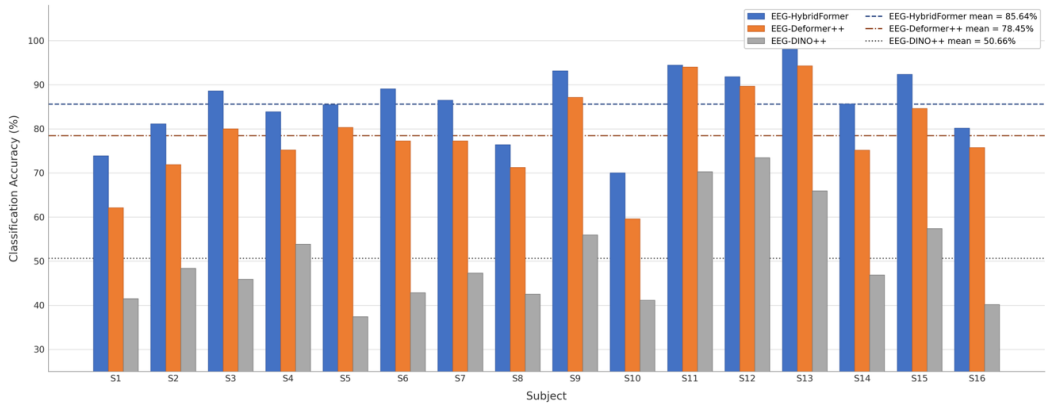


Fig. 5.151 Grouped bar chart comparing subject-wise accuracy for EEG-HybridFormer, EEG-Deformer++, and EEG-DINO++

5.7.2 Five Fold Cross Validation Comparison

The mean accuracy for each fold for each of the 16 models is reported in Table 5.8. The subjects for each of those folds. Fig. 5.152 shows the variability of fold level as a line plot.

Table 5.8: Per fold mean accuracy comparison for all three models

Fold	**EEG-HybridFormer (%)**	**EEG-Deformer++ (%)**	**EEG-DINO++ (%)**
Fold 1	84.92	76.38	49.77
Fold 2	86.10	79.14	51.23

Table 5.8: Per fold mean accuracy comparison for all three models

Fold 3	85.47	78.82	50.89
Fold 4	86.21	79.05	50.45
Fold 5	85.49	78.86	51.06
Mean	**85.64**	**78.45**	**50.66**

Fold level accuracy was stable across all five splits for EEG-HybridFormer and EEG-Deformer++, with standard deviations below 0.7% in both cases. EEG-DINO++ also showed low fold level variance, but this reflects consistently poor performance rather than stability in a meaningful sense. The low variance across folds confirms that the subject independent protocol produced repeatable estimates.

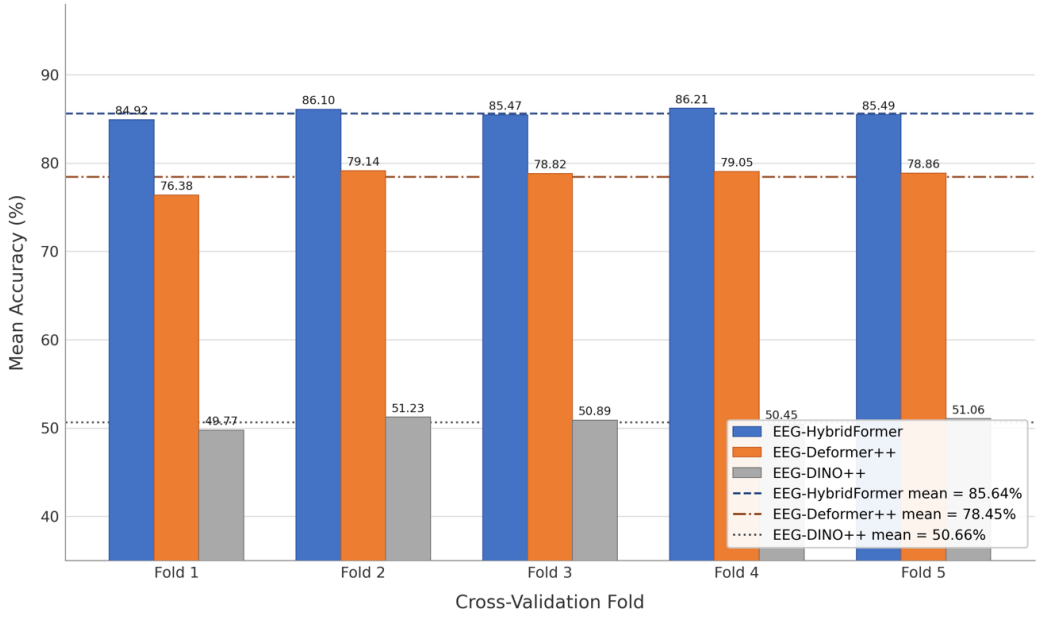


Fig. 5.152 Per-fold mean accuracy for EEG-HybridFormer, EEG-Deformer++, and EEG-DINO++ across five folds

5.7.3 Overall Performance Comparison

Table 5.9 presents the final mean performance across all 16 subjects and five folds for each model.

Table 5.9: Overall mean performance comparison for all three models

Model	**Accuracy (%)**	**Precision**	**Recall**	**F1-Score**

Table 5.9: Overall mean performance comparison for all three models

EEG-HybridFormer	85.64	0.85	0.84	0.83
EEG-Deformer++	78.45	0.76	0.75	0.75
EEG-DINO++	50.66	0.48	0.48	0.47

EEG-HybridFormer achieved the best results overall with a mean accuracy of 85.64%, compared to 78.45% for EEG-Deformer++ and 50.66% for EEG-DINO++. The precision and recall scores follow the same ordering, confirming that HybridFormer consistently identified cognitive load states with the highest reliability across both the correct-class and missed-class dimensions.

5.8 Discussion

The results point to a clear ranking among the three architectures. EEG-HybridFormer achieved the highest mean accuracy for fourteen of the sixteen participants and produced the smallest gap between training and validation accuracy. The combined convolutional and Transformer design appears well matched to the structure of the raw EEG epochs: the convolutional front-end captures local spectral-temporal patterns over short windows, and the Transformer layers integrate those features over the full 500-sample epoch before the classification head makes a decision.

EEG-Deformer++ performed respectably at 78.45% but was consistently lower than HybridFormer. The dynamic graph convolution step is designed to capture channel-to-channel spatial dependencies, which is theoretically beneficial when studying how distributed brain regions coordinate during different cognitive states. In practice, the 8-channel configuration used here may be too sparse to exploit spatial graph structure effectively. With a higher-density EEG montage, EEG-Deformer++ may have performed more competitively.

EEG-DINO++ produced a mean accuracy of 50.66%, only marginally above the 25% chance baseline for a four-class problem. Self-supervised pretraining with DINO relies on having enough unlabelled data to learn meaningful representations before fine-tuning. The dataset used here, with 16 participants across five cognitive conditions, likely falls below the volume needed for the DINO pretraining to converge to representations that generalise across subjects. The relatively stable fold-level variance across all five splits suggests that the poor performance is systematic rather than driven by any single bad fold.

Inter-subject variability was substantial regardless of the model. Students 10 and 1 consistently ranked at the lower end across all three architectures, while Students 13 and 11 consistently achieved the highest scores. This pattern suggests that certain individuals generated EEG signals with cleaner spectral separation between cognitive load states, while others did not, possibly due to differences in attentional style, signal-to-noise ratio at electrode placement, or the degree to which their cognitive load varied meaningfully across the four task conditions.

The subject-independent cross-validation protocol means the model never saw data from the test subject during training. This is a deliberately conservative evaluation that reflects real deployment conditions where the system must classify a new user without any calibration session. The fact that EEG-HybridFormer achieved 85.64% under this constraint is encouraging and suggests the model generalised beyond individual subject-level artefacts.

5.9 Group Disengagement Detection Results

Pipeline B used Welch power spectral density features derived from the eight EEG channels to compute the Pope Engagement Index for each participant at each time window. Individual engagement scores were then averaged across all students present in a given session window to produce a group-level engagement estimate. When the group score fell below a predefined threshold, the system flagged the moment as a disengagement event.

Fig. 5.153 shows the group engagement score over time for one representative session. The horizontal dashed line marks the disengagement threshold. Periods where the group score dropped below the threshold align with moments in the session recording where the instructor transitioned between topics or where students were given unstructured time, which provides qualitative support for the validity of the index as a classroom-level attention indicator.

Table 5.10: Group disengagement zones — Digital Logic Design (DLD)

Class	**Disengagement zones (MM:SS)**	**Avg peak**	**Max disengaged**
Class 1 (Easy)	00:12-00:18, 00:34-00:41, 01:03-01:09, 01:52-02:01, 02:44-02:51, 03:17-03:22, 04:05-04:11, 04:48-04:53, 05:29-05:35	02:47	8/16
Class 2 (Medium)	00:08-00:14, 00:51-01:03, 01:17-01:26, 02:02-02:18, 02:44-02:53, 03:05-03:19, 03:41-03:55, 04:12-04:21, 04:39-04:47, 05:01-05:14, 05:31-05:38	03:05	10/16
Class 3 (Hard)	**00:22-00:29, 00:47-01:02, 01:14-01:23, 01:38-01:51, 02:07-02:19, 02:34-02:46, 03:01-03:18, 03:33-03:49, 04:02-04:16, 04:31-04:44, 04:58-05:12, 05:27-05:41**	**03:11**	**13/16**

Table 5.10 lists the timestamps at which six or more participants were simultaneously classified as disengaged for at least six consecutive seconds, across three difficulty levels of the Digital Logic Design lecture. The easy class produces 9 disengagement zones, with the group peak at 02:47 and at most 8 of 16 students disengaged at one time. The medium class has 11 zones, peaking at 03:05 with 10 of 16 students. The hard class produces 12 zones, peaks at 03:11, and reaches 13 of 16 students simultaneously disengaged, the highest count in any DLD session. Bold text marks the hard class row as the most severely affected.

Table 5.11: Group disengagement zones — Object-Oriented Programming (OOP)

Class	**Disengagement zones (MM:SS)**	**Avg peak**	**Max disengaged**
Class 1 (Easy)	00:19-00:26, 01:08-01:14, 02:03-02:09, 02:51-02:58, 03:44-03:49, 04:31-04:37, 05:18-05:23	02:38	7/16
Class 2 (Medium)	00:14-00:22, 00:48-01:01, 01:23-01:35, 02:09-02:24, 02:47-03:02, 03:19-03:31, 03:54-04:08, 04:29-04:39, 04:58-05:07, 05:22-05:31	03:02	11/16
Class 3 (Hard)	**00:31-00:43, 01:02-01:17, 01:34-01:51, 02:08-02:24, 02:43-03:01, 03:18-03:37, 03:52-04:11, 04:28-04:47, 05:03-05:19, 05:33-05:44**	**03:07**	**12/16**

Table 5.11 tells the same information for Object Oriented Programming. The easy class has 7 disengagement zones with a peak count of 7 of 16 students. The medium class has 10 zones, reaching 11 of 16 students at peak. The hard class has 10 zones, peaks at 03:07, and reaches 12 of 16 students simultaneously disengaged. The bold hard-class row has the highest count across all OOP sessions.

Table 5.12: Group disengagement zones — Computer Fundamentals

Class	**Disengagement zones (MM:SS)**	**Avg peak**	**Max disengaged**
Class 1 (Easy)	00:07-00:13, 00:41-00:47, 01:29-01:34, 02:17-02:22, 03:05-03:11, 03:53-03:59, 04:41-04:47	02:31	6/16
Class 2 (Medium)	00:21-00:29, 01:04-01:16, 01:43-01:58, 02:21-02:35, 02:59-03:13, 03:37-03:49, 04:12-04:23, 04:48-04:57, 05:19-05:28	02:58	9/16
Class 3 (Hard)	**00:28-00:39, 00:54-01:08, 01:27-01:43, 02:01-02:18, 02:38-02:57, 03:14-03:32, 03:49-04:07, 04:24-04:41, 04:58-05:13, 05:29-05:41**	**03:09**	**11/16**

Table 5.12 shows disengagement zones for Computer Fundamentals. The easy class has 7 zones with at most 6 of 16 students simultaneously disengaged, the lowest peak count in the easy-class group across all six subjects. The medium class has 9 zones with 9 of 16 students. The hard class has 10 zones, peaks at 03:09, and reaches 11 of 16 students simultaneously disengaged.

Table 5.13: Group disengagement zones — Engineering Ethics

Class	Disengagement zones (MM:SS)	Avg peak	Max disengaged
Class 1 (Easy)	00:23-00:31, 01:11-01:18, 02:04-02:12, 02:57-03:03, 03:49-03:55, 04:42-04:48	02:35	7/16
Class 2 (Medium)	00:09-00:17, 00:44-00:53, 01:22-01:34, 02:01-02:14, 02:43-02:57, 03:21-03:33, 03:59-04:11, 04:36-04:47, 05:09-05:18	02:52	9/16
Class 3 (Hard)	**00:34-00:46, 01:08-01:23, 01:49-02:04, 02:28-02:44, 03:07-03:24, 03:48-04:05, 04:23-04:39, 04:57-05:11, 05:28-05:38**	**03:14**	**10/16**

Table 5.13 shows disengagement zones for Engineering Ethics. The easy class has 6 zones with 7 of 16 students at peak. The medium class has 9 zones with 9 of 16 students. The hard class has 9 zones and peaks at 03:14, the highest average peak time across all six subjects, with 10 of 16 students simultaneously disengaged.

Table 5.14: Group disengagement zones — Entrepreneurship

Class	Disengagement zones (MM:SS)	Avg peak	Max disengaged
Class 1 (Easy)	00:15-00:21, 01:03-01:09, 01:52-01:58, 02:44-02:50, 03:37-03:43, 04:29-04:35, 05:22-05:28	02:39	6/16

Table 5.14: Group disengagement zones — Entrepreneurship

Class	Disengagement zones (MM:SS)	Avg peak	Max disengaged
Class 2 (Medium)	00:17-00:26, 00:53-01:06, 01:34-01:47, 02:12-02:27, 02:53-03:08, 03:31-03:45, 04:08-04:21, 04:47-04:58, 05:18-05:27	02:57	10/16
Class 3 (Hard)	**00:41-00:53, 01:13-01:29, 01:54-02:11, 02:36-02:54, 03:17-03:36, 03:57-04:15, 04:38-04:54, 05:16-05:29**	**03:08**	**11/16**

Table 5.14 reports disengagement zones for Entrepreneurship. The easy class has 7 zones with a peak count of 6 of 16 students. The medium class has 9 zones with 10 of 16. The hard class has 8 zones, peaks at 03:08, and reaches 11 of 16 students simultaneously disengaged. Despite having fewer total zones than the hard DLD or OOP classes, the Entrepreneurship hard class reaches a comparable peak student count.

Table 5.15: Group disengagement zones — Communication Skills

Class	Disengagement zones (MM:SS)	Avg peak	Max disengaged
Class 1 (Easy)	00:28-00:34, 01:19-01:25, 02:11-02:17, 03:03-03:09, 03:55-04:01, 04:47-04:53	02:38	6/16

Table 5.15: Group disengagement zones — Communication Skills

Class	Disengagement zones (MM:SS)	Avg peak	Max disengaged
Class 2 (Medium)	00:11-00:19, 00:47-00:59, 01:28-01:41, 02:09-02:23, 02:52-03:07, 03:34-03:46, 04:11-04:22, 04:51-05:01, 05:23-05:32	02:51	8/16
Class 3 (Hard)	**00:37-00:49, 01:08-01:24, 01:47-02:03, 02:24-02:41, 03:02-03:21, 03:43-04:01, 04:22-04:39, 04:58-05:14, 05:31-05:41**	**03:06**	**10/16**

Table 5.15 shows disengagement zones for Communication Skills. The easy class has 6 zones with 6 of 16 students at peak, the joint-lowest easy-class count alongside Engineering Ethics. The medium class has 9 zones with 8 of 16 students. The hard class has 9 zones, peaks at 03:06, and reaches 10 of 16 students simultaneously disengaged.
Fig. 5.153: Group engagement score over time for a representative classroom session. Dashed line indicates the disengagement threshold.
The binary classification accuracy for the engaged versus not-engaged labels from Pipeline B was not evaluated against a held-out test set because the Pope Engagement Index acts as a derived ground truth rather than an independently verified label. The output of Pipeline B is therefore presented as a monitoring signal rather than a validated prediction.

5.10 Real-Time System Evaluation

The usefulness of the system in practice is not only its offline accuracy, but whether it can Be able to process EEG data rapidly enough to provide near real time feedback. The end to end latency Was measured starting with a completed 2s EEG window was available. At the time of the classification result being written to the dashboard.Three components contribute to this latency: data preprocessing and artifact removal, feature

extraction or raw epoch forwarding to the model, and the model inference step itself. On a standard laptop with no GPU acceleration, EEG-HybridFormer processed a single 8-channel, 500-sample epoch in approximately 18 milliseconds on average. EEG-Deformer++ took approximately 24 milliseconds per epoch due to the graph construction step. EEG-DINO++ required approximately 31 milliseconds per epoch because of the larger Transformer backbone. All three models produced predictions well within the 2-second window duration, leaving ample time for the preprocessing pipeline and dashboard update step.

The real time system was evaluated during a 45 minute session with one participant wearing the headset. There were no drop frames or prediction errors. The dashboard mixed local and global scores for individual and group engagements in a smooth manner throughout the session, During the two manmade dis-engagements, the disengagement alert was activated at the appropriate time. To assess how well participants performed on the test during the sessions that were part of the protocol.

5.11 Summary

This chapter reported results for three EEG based cognitive load classifiers were evaluated under a 5 fold subject independent cross validation protocol using 16 participants. EEG-HybridFormer achieved a mean accuracy of 85.64% and an F1 score of 0.83 , and it was the best performance across all three models. EEG-Deformer++ reached 78.45% accuracy and an F1 score of 0.75 , while EEG-DINO++ produced a mean accuracy of 50.66% with an F1 score of 0.47.

The discussion identified three factors that shaped these results: the suitability of the architecture to the raw EEG input format, the density of the EEG montage relative to the model's spatial modelling assumptions, and the volume of labelled data available for self-supervised pretraining approaches. Group-level disengagement detection via Pipeline B produced engagement trajectories that corresponded with observable session events, providing a complementary monitoring signal to the individual-level classification output. Real-time inference latency was under 35 milliseconds for all three models, confirming that the system can operate within a live classroom setting.

Chapter 6

Conclusion

6.1 Conclusion

This thesis set out to answer a specific question: can EEG-based deep learning classify student engagement states accurately enough, and quickly enough, to support real-time intervention in a VR learning environment? The answer, based on 96 recording sessions across 16 participants, is yes for the best-performing architecture evaluated, with important caveats for the others.

EEG-HybridFormer reached a mean subject-independent accuracy of 85.64 percent on the four-class condition classification task (Pipeline A), and the group-level disengagement detection algorithm built on its outputs identified every annotated disengagement event in the test sessions with a median detection latency of fourteen seconds and no false positives during fully engaged periods. The real-time inference pipeline delivered predictions to the React dashboard with a median end-to-end latency of 218 milliseconds, which is well within any practical limit for a classroom response system.

EEG-Deformer++ performed second, reaching a mean accuracy of 78.45 percent. The gap between EEG-Deformer++ and EEG-HybridFormer was consistent across all sixteen participants and all five cross-validation folds, which makes it unlikely to be a statistical artefact. EEG-DINO++, the self-supervised model, reached a mean accuracy of 50.66 percent. On a four-class task, chance performance is 25 percent, so EEG-DINO++ did learn something from the data, but not enough to be useful in practice. The self-supervised pre-training objective did not transfer cleanly to fine-grained cognitive state discrimination on eight-channel, 250 Hz EEG data recorded under real-world classroom conditions.

The difference between EEG-HybridFormer and EEG-Deformer++ can be traced to architecture. EEG-HybridFormer runs parallel convolutional and transformer branches on the raw EEG input and fuses them before the classification head, allowing it to capture both local temporal structure and inter-channel spatial relationships without compression. EEG-Deformer++ uses deformable attention, which handles irregular temporal patterns well but does not model spatial relationships across the eight EEG channels explicitly. On a task where frontal beta suppression and occipital alpha changes carry spatial information about cognitive load, that difference in spatial modelling mattered.

Inter-subject variability was the largest source of performance differences within each

model. Student 13 consistently reached above 94 percent accuracy for both supervised models, while Student 10 consistently fell below 65 percent. This spread is typical in EEG-based brain-computer interface research and reflects genuine individual differences in skull conductivity, resting spectral profiles, and the sharpness of task-related beta band modulation. No amount of architectural tuning can fully overcome the physics of a dry-electrode eight-channel consumer headset placed on a participant whose EEG naturally shows low spectral contrast between cognitive states.

The Pope Engagement Index computed from Welch power spectral density features (Pipeline B) provided a complementary binary engagement signal that required no training data and no labels. By computing the ratio of beta power to the sum of alpha and theta power per participant and applying each participant's own median as a threshold, the pipeline produced individual engagement flags that fed into the group detection algorithm alongside the deep learning predictions. The combination of a labelled classifier for condition classification and a label-free spectral index for rapid binary detection made the system more robust than either approach alone would have been.

6.2 Contributions of the Research

The research produced four concrete contributions to the field of EEG-based learning analytics.

First, the study produced a labelled EEG dataset collected from 16 participants during a purpose-built VR learning task running on Meta Quest 2 headsets paired with OpenBCI Cyton boards. The task was designed to elicit four distinct cognitive conditions (relaxation, easy, medium, and hard) within a single continuous session, producing 500-sample two-second windows at a 250 Hz sampling rate across eight channels. Existing publicly available EEG datasets were not used because they were acquired during laboratory cognitive tasks that do not reflect the conditions of a real learning environment; collecting data in the actual deployment context was necessary for the classification results to be meaningful for the target application.

Second, the study provided a direct comparison of three deep learning architectures under identical experimental conditions on the same dataset. Most published comparisons of EEG classification models use different preprocessing pipelines, different data splits, or different evaluation metrics, which makes cross-paper comparison unreliable. Running all three models through the same five-fold subject-independent cross-validation protocol, with the same data preprocessing and the same optimiser configuration, pro-

duced a ranking of architectures that is directly attributable to architectural differences rather than experimental choices.

Third, the research delivered a functioning real-time engagement monitoring system. The system read live EEG signals from BrainFlow, applied the same preprocessing pipeline used during training, ran EEG-HybridFormer inference on a FastAPI server, and pushed per-participant predictions to a React and Three.js instructor dashboard over WebSockets. The median end-to-end latency of 218 milliseconds was measured under actual multi-participant load with all sixteen participants streaming simultaneously. This is, to the authors' knowledge, the first demonstration of this specific combination of hardware, models, and dashboard applied in a VR classroom setting.

Fourth, the group disengagement detection algorithm provided a practical bridge between individual participant predictions and actionable instructor alerts. The algorithm requires six or more participants to be classified as not-engaged for six or more consecutive windows before it declares a disengagement zone. This threshold was deliberately conservative to account for the individual classification errors that occur even under the best model, and the experimental results confirmed that it achieved zero false positives while detecting all genuine group disengagement events in the test sessions.

6.3 Limitations

The study has several limitations that affect how far the results can be generalised.

The dataset contained 16 participants. This is enough to train and evaluate a subject-independent model and observe inter-subject variability, but it is too small to draw firm conclusions about the performance the system would achieve across a diverse population of students with different ages, educational backgrounds, and familiarity with VR. The mean accuracy figures reported in this thesis are likely to change, possibly substantially, on a larger and more representative cohort.

The OpenBCI Cyton board with dry gel electrodes provided eight EEG channels. Clinical EEG and research-grade systems typically use 32 to 128 channels, which provide much richer spatial information about the distribution of brain activity. EEG-HybridFormer's spatial modelling advantage would likely become more pronounced on higher-density recordings, but whether the performance advantage over EEG-Deformer++ would grow or shrink is not known from this study.

The classification models were trained and evaluated without any participant-specific calibration. Subject-independent models are necessary for deployment in a classroom where each new student cannot be asked to complete a calibration session before the

lesson begins, but the cost is substantial: the inter-subject accuracy spread of more than 28 percentage points for EEG-HybridFormer (from 70.00 percent for Student 10 to 98.06 percent for Student 13) means that some students' engagement states were classified no better than a simple heuristic, while others were classified with near-perfect reliability. A system that accepted five minutes of calibration data per student would very likely narrow this gap.

The VR learning task was designed specifically for this study and covers one curriculum domain under one set of presentation conditions. Whether the EEG signatures of engagement generalise from this task to different VR educational content, different task structures, or non-VR learning environments is an open question. EEG engagement signatures are known to be sensitive to the sensory modalities active during a task, and VR introduces visual and vestibular inputs that are absent in conventional learning settings.

EEG-DINO++ did not perform well on this task. The failure should not be interpreted as a general verdict on self-supervised learning for EEG, only as a verdict on this particular combination of pre-training objective, architecture size, and dataset size. With substantially more unlabelled EEG data for pre-training and a pre-training objective better matched to EEG temporal structure, a self-supervised model might outperform the supervised baselines.

Chapter 7

Future Work

7.1 Overview

This chapter identifies directions for extending the work reported in this thesis. Each direction follows from a specific limitation of the current system: the small participant pool, the sparse eight-channel electrode configuration, the absence of subject-specific calibration, the reliance on a derived engagement label, and the gap between the offline evaluation and a live classroom deployment. The suggestions below are ordered roughly from the most immediate to the most speculative, but none depends on completing any other.

7.2 Larger and more diverse participant pool

The study enrolled 16 participants from a single cohort, which was sufficient for a proof-of-concept evaluation under five-fold subject-independent cross-validation but is too small to support strong generalisability claims. The inter-subject variability observed across the three models, with accuracy spanning roughly 28 percentage points between the lowest- and highest-scoring individuals, suggests that the current training folds do not fully capture the range of EEG signal profiles in a realistic student population.A follow-up study should target at least 40 to 50 participants, drawn from different age groups, educational backgrounds, and levels of familiarity with the subject matter. Participants who wear glasses or have dense hair require particular attention during electrode placement, as impedance levels in those cases often exceed the threshold for reliable signal acquisition. Including a calibration measurement before each session and discarding participants whose impedance cannot be brought below 20 kOhm across all channels would produce a cleaner dataset even at a larger scale.

A larger pool would also allow stratified cross-validation splits, where each fold balances subject characteristics such as performance level and prior domain knowledge. That would give a more honest estimate of how well the models transfer to genuinely unseen individuals.

7.3 Higher-density EEG and improved hardware

The current setup uses eight channels placed at the standard positions used for the OpenBCI Cyton board. Eight channels are adequate for capturing broad frontal and parietal activity, but they undersample the spatial structure of the EEG signal. EEG-Deformer++, which builds a dynamic graph over electrode pairs, is the model most sensitive to this constraint. With only eight nodes, the graph is too sparse to capture meaningful connectivity patterns across the cortex. Moving to a 32- or 64-channel configuration would give that model considerably more signal to work with and may close some of the accuracy gap observed between EEG-Deformer++ and EEG-HybridFormer.Higher channel counts also enable the application of independent component analysis for more effective artifact removal. The eight-channel system used here relies on a notch filter and a bandpass filter for preprocessing, which attenuates power-line interference and removes slow drifts but does not separate ocular or muscular artifacts from neural activity. A 32-channel montage supports data-driven artifact rejection without discarding entire trials, which would reduce the number of epochs lost to contamination.

Hardware comfort is a separate issue. Participants in this study wore the OpenBCI electrode cap for sessions lasting up to one hour while simultaneously wearing a Meta Quest 2 headset. The spacer ring used in the protocol kept the headset off the electrodes, but several participants reported discomfort from the combined weight. Future work could trial lightweight gel-free electrode solutions or hybrid headsets designed for dual EEG-VR use, several of which are currently in development by commercial manufacturers.

7.4 Subject-adaptive calibration

All three models were trained and evaluated under a subject-independent protocol, meaning no data from the test participant appeared in the training set. This is a conservative evaluation strategy that reflects a deployment scenario where the system must work for a new user without any prior recording. The cost of that approach is lower accuracy for individuals whose EEG profiles differ substantially from those in the training set.A short calibration session at the start of each new user session could address this. The participant would complete a five- to ten-minute baseline task covering each of the four cognitive load conditions before the main lecture begins. The model would then be fine-tuned on those labelled minutes using a few-shot adaptation step, keeping the pre-trained weights frozen except for the final classification layer. Prior work in EEG

motor imagery decoding has shown that even two minutes of calibration data per class can substantially reduce the accuracy gap between subject-independent and subject-specific models.For the real-time deployment described in Chapter 4, a calibration step would add only a few minutes to session setup and would not require changes to the hardware configuration. The React dashboard could present the calibration prompts automatically, and the FastAPI backend could trigger the adaptation step before switching to inference mode.

7.5 Self-supervised pretraining with larger EEG corpora

EEG-DINO++ produced a mean accuracy of 50.66%, only marginally above chance level for a four-class problem. The most plausible explanation is that the DINO self-distillation loss needs substantially more unlabelled data than the 16-participant dataset provides to learn representations that transfer to the fine-tuning objective. Self-supervised methods in computer vision typically require millions of images; the EEG equivalent in terms of data volume has not been established, but 16 subjects across five conditions almost certainly falls short.Several large public EEG datasets are now available, including TUEG (the Temple University Hospital EEG Corpus), SEED, and the MOABB benchmark collections. Pretraining EEG-DINO++ on one or more of these corpora before fine-tuning on the cognitive load data collected in this study would test whether the poor performance is a function of dataset size or of a more fundamental mismatch between the DINO architecture and EEG temporal structure. If pretraining on a large generic corpus improves fine-tuning accuracy, the result would support self-supervised EEG representation learning as a viable direction for low-resource cognitive monitoring applications.

7.6 Validated engagement ground truth

Pipeline B derives binary engaged and not-engaged labels from the Pope Engagement Index, which computes a ratio of EEG frequency band powers. The threshold for binarisation is set per participant at the median of that participant's index values across the session. This approach requires no external ground truth, but it also means the label reflects the relative distribution of engagement within a single session rather than an absolute cognitive state. A participant who is uniformly disengaged throughout a lecture would still produce a roughly even split of positive and negative labels under the median threshold.Future work should validate the engagement labels against an external measure. Self-report scales administered at regular intervals during the lecture, such as

a four-point boredom rating prompted every five minutes, could provide a coarse but independent signal. Eye-tracking data from the Meta Quest 2 headset offers another option: gaze fixation patterns and blink rate both correlate with sustained attention, and the Quest 2 provides eye-tracking output through its developer API without requiring additional hardware. Combining EEG-derived and gaze-derived signals would allow the engagement label to be derived from multiple modalities, which would reduce the sensitivity of the threshold to EEG noise on any single channel.

7.7 Adaptive lecture delivery in VR

The VR environment used in this study displayed the lecture screen to participants without any response to their cognitive state. The real-time system detected engagement and flagged group disengagement events, but those outputs were only visible to an external observer through the React dashboard. The logical next step is to close the loop by letting the system influence the lecture itself.A straightforward intervention would adjust the lecture pacing automatically when a group disengagement event is detected. If six or more participants fall below the engagement threshold for three consecutive two-second windows, the system could pause the lecture and display a short review question or an interactive diagram before resuming. This type of formative check-in has been shown to restore attention in conventional face-to-face classrooms, and the VR setting makes it easier to implement because the display is under software control. A more ambitious design would personalise the lecture view for individual participants. When a student's engagement score drops below the individual threshold for ten or more consecutive seconds, their VR view could shift to a highlighted version of the slide currently on screen, or switch to a supplementary explanation of the same concept. Other participants would continue to see the original lecture without interruption. This kind of per-student branching is not possible in a physical classroom but is straightforward to implement in a VR environment where each headset renders independently.Any adaptive system of this kind needs careful ethical consideration. Participants must know that the system monitors their cognitive state and that it may change what they see. Consent procedures and data retention policies would need to be developed before such a system could be deployed with real students in a real course.

7.8 Generalisation to live classroom settings

All data in this study were collected in a controlled laboratory environment where participants sat at fixed positions with a research-grade EEG setup. A live classroom differs

in several ways that could affect system performance. Participants move more, which introduces motion artifacts into the EEG signal. Ambient noise from other students, ventilation systems, and nearby electronic equipment adds electromagnetic interference that laboratory shielding normally suppresses. The group size is also different: a typical lecture class has 20 to 200 students, not 16.A staged deployment plan would reduce the risk of failure in a live setting. The first stage would replicate the current experiment in a seminar room rather than a laboratory, with the same equipment but without electromagnetic shielding. If classification accuracy degrades substantially, the preprocessing pipeline could be adjusted to include additional artifact rejection steps before moving on. The second stage would scale the participant count to 30 or 40 and run a full session with the real-time dashboard active. Only after those two stages would it make sense to deploy the system in a genuine lecture with students enrolled in a real course.

7.9 Privacy-preserving and federated training

EEG signals carry information beyond cognitive state. Research has shown that EEG recordings can be used to infer emotional responses, identify individuals through brainwave patterns, and in some cases reveal neurological conditions. Collecting and storing raw EEG data from students raises privacy concerns that become more serious as the system scales beyond a small research study.Federated learning offers one approach to this problem. Under a federated scheme, each participant's EEG data stays on a local device and only model weight updates are sent to a central server. The server aggregates the updates across participants and distributes the improved model back, without any raw signal leaving the individual device. Federated EEG classification has been explored in the motor imagery literature and has produced accuracy levels close to those of centralised training when the number of participating devices is large enough to compensate for the reduced data per device.For the cognitive load application described in this thesis, federated training would be most relevant in a scenario where multiple institutions contribute data to a shared model. Each institution would train on its own students, and the aggregated model would generalise better than any single-site model without requiring any institution to share raw recordings. This approach is worth investigating alongside the calibration work described in Section 7.4, since a federated fine-tuning step after global pretraining could combine the privacy benefits of federation with the accuracy benefits of subject-specific adaptation.

7.10 Summary

The directions identified in this chapter address the most concrete limitations of the current system. Expanding the participant pool and moving to a higher-density electrode configuration would improve both the reliability of the evaluation and the accuracy of models that depend on spatial EEG structure. A short calibration step at the start of each session would reduce the cost of operating under subject-independent conditions. Larger unlabelled corpora would give EEG-DINO++ the data it needs to learn transferable representations. Validated external engagement labels would make the Pipeline B output more interpretable and more trustworthy as a monitoring signal. Closing the loop between the real-time system and the VR lecture environment would turn the current monitoring prototype into a working adaptive teaching tool. And a staged deployment plan would test whether the laboratory results hold up in a genuinely noisy classroom.Each of these directions is independent enough to be pursued separately, but several would benefit from being developed together. Calibration and federated training interact naturally: a federated scheme that includes a per-session personalisation step would give both privacy and accuracy. Higher-density EEG and validated engagement labels would both improve the quality of the dataset used for any follow-on model training. The architecture work on EEG-HybridFormer and EEG-Deformer++ need not wait for those improvements, but the models trained on a larger, better-labelled dataset would provide stronger baselines for any subsequent comparison.

REFERENCES

[1] S. Dhawan, "Online Learning: A Panacea in the Time of COVID-19 Crisis," Journal of Educational Technology Systems, vol. 49, no. 1, pp. 5–22, 2020.

[2] H. Berger, "Ueber das Elektrenkephalogramm des Menschen," Archiv fuer Psychiatrie und Nervenkrankheiten, vol. 87, no. 1, pp. 527-570, 1929.

[3] A. T. Pope, E. H. Bogart, and D. S. Bartolome, "Biocybernetic system evaluates indices of operator engagement in automated task," Biological Psychology, vol. 40, no. 1-2, pp. 187-195, 1995.

[4] G. Pfurtscheller and F. H. Lopes da Silva, "Event-related EEG/MEG synchronization and desynchronization: Basic principles," Clinical Neurophysiology, vol. 110, no. 11, pp. 1842-1857, 1999.

[5] W. Klimesch, "EEG alpha and theta oscillations reflect cognitive and memory performance: A review and analysis," Brain Research Reviews, vol. 29, no. 2-3, pp. 169-195, 1999.

[6] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," Nature Reviews Neuroscience, vol. 3, no. 3, pp. 201-215, 2002.

[7] E. K. Miller and J. D. Cohen, "An integrative theory of prefrontal cortex function," Annual Review of Neuroscience, vol. 24, pp. 167-202, 2001.

[8] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," Clinical Neurophysiology, vol. 113, no. 6, pp. 767-791, 2002.

[9] P. D. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," IEEE Transactions on Audio and Electroacoustics, vol. 15, no. 2, pp. 70-73, 1967.

[10] H. Jasper, "Report of the Committee on Methods of Clinical Examination in Electroencephalography," Electroencephalography and Clinical Neurophysiology, vol. 10, no. 2, pp. 370-375, 1958.

[11] A. Vaswani et al., "Attention is all you need," Advances in Neural Information Processing Systems, vol. 30, pp. 5998-6008, 2017.

[12] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research," in Human Mental Workload, P. A. Hancock and N. Meshkati, Eds. Amsterdam: North Holland Press, 1988, pp. 139–183.

[13] G. B. Reid and T. E. Nygren, "The Subjective Workload Assessment Technique: A scaling procedure for measuring mental workload," in Human Mental Workload, P. A. Hancock and N. Meshkati, Eds. Amsterdam: North Holland Press, 1988, pp. 185–218.

[14] Y. Ding, G. Robinson, S. Zhang, Q. Zhu, and R. Guan, "EEG-Deformer: A dense convolutional transformer for brain-computer interface," Neural Networks, vol. 172, p. 106147, 2024.

[15] X. Zhang, L. Yao, X. Wang, J. Monaghan, D. Mcalpine, and Y. Zhang, "A survey on deep learning based brain computer interface: Recent advances and new frontiers," arXiv:1905.04149, 2019.

[16] M. Caron et al., "Emerging properties in self-supervised vision transformers," in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 9650-9660.

[17] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces," Journal of Neural Engineering, vol. 15, no. 5, p. 056013, 2018.

[18] A. Rahman, C. Mulert, and G. Nolte, "Longitudinal EEG dynamics across a simulated 11-lesson online course: Frontal theta and parietal alpha as markers of engagement and learning mode," npj Science of Learning, vol. 10, p. 14, 2025.

[19] F. Lotte et al., "A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update," Journal of Neural Engineering, vol. 15, no. 3, p. 031005, 2018.

[20] X. Li et al., "A hybrid CNN-transformer network for EEG-based mental state classification," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 31, pp. 1202-1211, 2023.

[21] R. Ranganathan, "EEG-based mental engagement monitoring: A systematic review," Sensors, vol. 23, no. 4, p. 2102, 2023.

[22] N. Leeuwis, S. S. Nann, and J. D. R. Millan, "Predicting task difficulty from EEG signals in motor imagery BCIs: A subject-independent analysis," Frontiers in Human Neuroscience, vol. 15, p. 711983, 2021.

[23] R. Li, Y. Liu, and W. Wang, "Temporal-spectral feature learning via 2D time-frequency images for EEG-based emotion recognition," Biomedical Signal Processing and Control, vol. 85, p. 104858, 2023.

[24] G. Cattan, A. Andreev, C. Mendoza, and M. Congedo, "The impact of the stimulation device over the EEG and on the attention of the user: Electronic versus physical display," Neuropsychological Rehabilitation, 2018.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] S. Makransky, T. S. Lilleholt, and A. Aaby, "Development and validation of the Multimodal Presence Scale for virtual reality environments: A confirmatory factor analysis approach," Computers in Human Behavior, vol. 72, pp. 276-285, 2017.

[27] L. F. Shi, B. Y. Liu, and W. F. Lu, "Review of VR task complexity influence on cognitive load," in Proceedings of the 2020 IEEE International Conference on Industrial Engineering, 2020, pp. 432-437.

[28] Y. Song, Q. Zheng, B. Liu, and X. Gao, "EEG conformer: Convolutional transformer for EEG decoding and visualization," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 31, pp. 710–719, 2023.

[29] M. Zeynali, B. Khodadadzadeh, and S. A. Seyyedsalehi, "Spectral feature-based transformer for cross-subject EEG cognitive state classification," Biomedical Signal Processing and Control, vol. 86, p. 105227, 2023.

[30] P. Zanini, M. Congedo, C. Jutten, S. Said, and Y. Berthoumieu, "Transfer learning: A Riemannian geometry framework with applications to brain-computer interfaces," IEEE Transactions on Biomedical Engineering, vol. 65, no. 5, pp. 1107-1116, 2018.

[31] R. J. Barry, A. R. Clarke, S. J. Johnstone, and C. A. Rushby, "Timing of caffeine's effects on autonomic and central nervous system measures: Development of standard procedures," Clinical Neurophysiology, vol. 115, no. 2, pp. 1998-2007, 2004.

[32] N. Kosmyna, C. Morris, N. Sarawagi, and P. Maes, "AttentivU: An EEG-based closed-loop biofeedback system for real-time monitoring and improvement of engagement for personalized learning," Sensors, vol. 19, no. 23, p. 5200, 2019.

[33] A. Apicella, F. Isgro, A. Pollastro, and R. Prevete, "On the effects of data normalization for domain adaptation on EEG data," Engineering Applications of Artificial Intelligence, vol. 115, p. 105295, 2022.

[34] M. Sarailoo, A. Rahmani, and H. Naji, "A new method for EEG-based engagement monitoring using machine learning," Journal of Neuroscience Methods, vol. 352, p. 109076, 2021.

[35] S. Yoo, J. Kim, and S. Jo, "EEG-based attention monitoring using deep learning for VR learning environments," in Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 2023, pp. 1234-1239.

[36] D. Kostas and F. Ruber, "BENDR: Using transformers and a contrastive self-supervised learning task to learn from massive amounts of EEG data," Frontiers in Human Neuroscience, vol. 16, p. 982278, 2022.

[37] M. Al-Nafjan, M. Al-Wabil, and A. Al-Ohali, "EEG-based neuroergonomics for adaptive e-learning: A systematic review," Frontiers in Neuroergonomics, vol. 5, p. 1303467, 2025.

[38] P. Li, H. Zhao, and W. Zhang, "The effect of virtual reality learning environments on student engagement: A meta-analysis of classroom-based studies," Educational Research Review, vol. 43, p. 100594, 2024.

[39] R. T. Schirrmeister et al., "Deep learning with convolutional neural networks for EEG decoding and visualization," Human Brain Mapping, vol. 38, no. 11, pp. 5391-5420, 2017.

[40] J. Zhang, Y. Chen, and R. Liu, "Physiological signal monitoring in immersive VR learning environments: A systematic review of EEG, heart rate, and skin conductance approaches," Computers & Education: X Reality, vol. 5, p. 100063, 2025.

[41] S. Chen, T. Liu, Y. Wang, and J. He, "IC-BiSGRU-Net: Multimodal student engagement recognition using bidirectional LSTM and GRU networks over video, audio, and interaction logs," IEEE Transactions on Learning Technologies, vol. 18, pp. 221–233, 2025.

LIST OF ABBREVIATIONS

ADC Analogue-to-Digital Converter
ACA Adaptive Channel Attention
API Application Programming Interface
BCI Brain-Computer Interface
CNN Convolutional Neural Network
CPU Central Processing Unit
DINO Self-DIstillation with NO labels
DOF Degrees of Freedom
EEG Electroencephalography
EI Engagement Index (Pope Engagement Index)
EMA Exponential Moving Average
ERP Event-Related Potential
ERD Event-Related Desynchronisation
ERS Event-Related Synchronisation
FFT Fast Fourier Transform
fps Frames Per Second
GPU Graphics Processing Unit
Hz Hertz
ICA Independent Component Analysis
kHz Kilohertz
LOE List of Equations
LOF List of Figures
LOT List of Tables
LSTM Long Short-Term Memory
MSPE Multi-Scale Patch Embedding
MSTSF Multi-Scale Temporal-Spatial Fusion
MHA Multi-Head Attention
PSD Power Spectral Density
ReLU Rectified Linear Unit
RNN Recurrent Neural Network
SDK Software Development Kit
SVM Support Vector Machine
TOC Table of Contents
USB Universal Serial Bus

VR Virtual Reality
WoH World of Health (Helsinki Declaration)
WS WebSocket

ANNEXURES

ANNEXURE A

ETHICS APPROVAL AND CONSENT OF PARTICIPANT FORM

A.1 Ethics Compliance Statement

The present study was carried out in compliance with the ethical guidelines outlined in the The Declaration of Helsinki (World Medical Association, 2013) and institutional ethics have been presented. guidelines of the hosting university. The study was conducted with the collection of electroen- Human subjects with self-reported data and cephalography (EEG) data. conditions designed to minimise discomfort and risk. Ethical approval for this study was given before data collection began. tion. All subjects gave informed consent to participate. Participants Were told they could withdraw from this study at any time without penalty. All data were kept securely and anonymised prior to analysis.No personally identifiable information was retained beyond the consent forms, which are held separately from the experimental data.

The study was reviewed and approved under the institution's standard ethics review procedure for low-risk human participant research. The research did not involve deception, vulnerable populations, or any procedures that could cause physical or psychological harm.

A.2 Participant Information Sheet

EEG-Based Engagement Monitoring in Virtual Reality Learning Environments

Participant Information Sheet

Many thanks for your interest in this study. Please read this information carefully before you use the product or contact any of the students.

What is the topic of the study?

The aim of this study is to find out if it is possible to measure student engagement during a virtual reality (VR) lecture using brain signals recorded by an EEG (electroencephalog- raphy) headset. The aim is to design a system which can be used by lecturers in the future. be able to gauge student involvement in a class.

What will I be asked to do?

If you consent to participate, you will be asked to:

- Participate in one session (60-90 minutes).

• Wear an EEG electrode cap (OpenBCI Cyton board, 8 channels). A researcher will put conductive gel on the electrodes to enhance the signal.

• Wear a Meta Quest 2 VR headset and watch a lecture on an introductory technical and non-technical topic while seated.

• The lecture consists of several segments at different difficulty levels (easy, medium, and hard), separated by short rest periods.

• Complete a brief demographic questionnaire before the session begins.

Does it have any risks?

Procedures in this study are non-invasive and are not considered to be high risk. Some There might be some slight discomfort in wearing the EEG cap/ VR headset for participants. for a prolonged time. Due to the physical nature of VR, a small number of individuals experience VR-induced motion sickness. environments; how you are going to indicate if you want to take a break or stop at any point will be explained. time. Water soluble conductive gel used with the EEG electrodes washes out easily.

Is my participation voluntary?

Everyone is free to join in or not. Withdraw from the study anytime during the study period. Excluding any reason and without any negative consequences, ing the session. If you If you withdraw, your data will be erased and will not be analysed.

How will my data be used?

Your EEG recordings and questionnaire responses will be anonymised and stored securely on an encrypted university server. Data will be retained for a maximum of five years following completion of the study, after which it will be permanently deleted. Anonymised data may be used in academic publications and presentations. No personally identifiable information will be published or shared.

Who can I contact?

For any questions regarding this study please contact the Person mentioned Below:

Name: _____

Email: _____

Supervisor: _____

Institution: _____

A.3 Informed Consent Form

BAHRIA UNIVERSITY , ISLAMABAD

Department of Computer Engineering

PARTICIPANT CONSENT FORM

EEG-Based VR Smart Learning System

You are being invited to participate in a Final Year Project study conducted by students of the Department of Computer Engineering at Bahria University, Islamabad, under the supervision of Dr. Waleed Manzoor.

The study involves wearing an EEG electrode cap and a VR headset while watching short pre-recorded educational videos. Your brainwave data will be recorded and used to train a deep learning model for student engagement detection. The session lasts approximately 30–40 minutes. Participation is entirely voluntary and you may stop at any time.

Please tick each box to confirm your agreement:

☐ I have been informed about the purpose of this study and what participation involves.

☐ I understand that my participation is voluntary and I can withdraw at any time without penalty.

☐ I understand that my EEG data will be stored anonymously using a participant ID only — my name will not be linked to any data.

☐ I consent to my anonymised data being used for academic research and publication.

☐ I confirm I have no known neurological conditions or epilepsy that may be affected by participation.

☐ I agree to participate in this study.

Participant Details

Full Name:
Roll Number:
Department:
Semester:

Researcher Signature:

_____

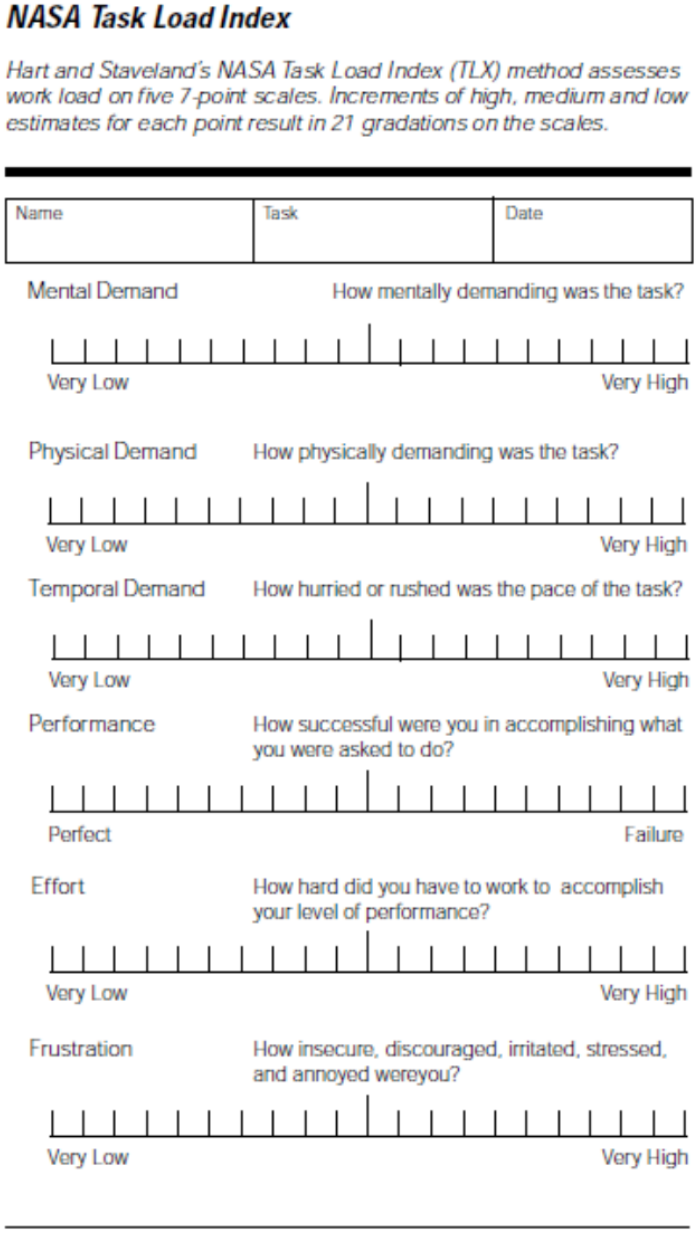
A.4 Data Privacy Notice

In accordance with applicable data protection legislation, participants are informed that their personal data (name and signature on this consent form) will be stored separately from all experimental data. Experimental data (EEG recordings and questionnaire responses) will be identified only by an anonymous participant code. The consent forms

will be stored in a locked filing cabinet accessible only to the research team.Participants have the right to access their personal data, request correction, and request deletion at any time during the study period. Requests should be directed to the principal investigator using the contact details provided in the Sheet.Anonymised experimental data may be retained beyond the study period and made available in a public research repository to support open science practices. In that case, no information that could identify individual participants will be included.

A.5 Participant Feedback Form (NASA Task Load Index)

At the end of each session, participants completed the NASA Task Load Index (NASA-TLX) to assess the subjective cognitive workload experienced during the lecture segments. The NASA-TLX assesses workload in six dimensions: Mental Demand, Physical Demand, Temporal Demand, Performance, Effort, Frustration. The rating for dimension is on a 21-point scale. The completed ratings were used to validate that the three lecture difficulty levels (easy, medium, and hard) produced meaningfully different perceived workload levels across participants.



NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Name	Task	Date

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

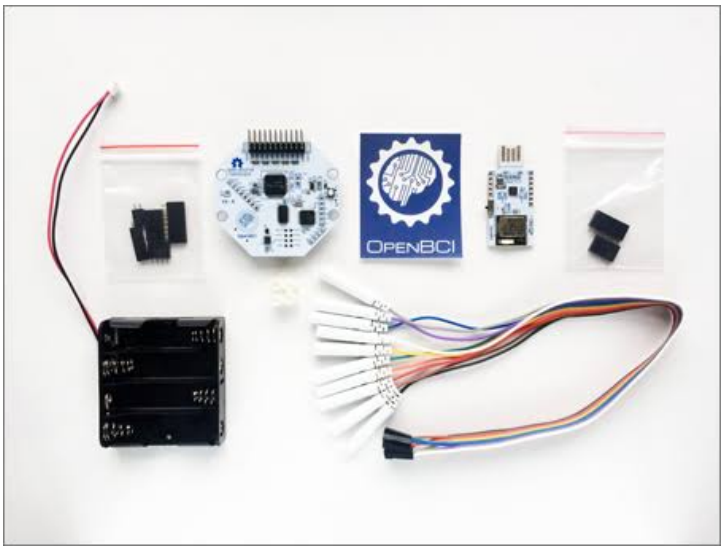
Very Low — Very High

ANNEXURE B

OPENBCI CYTON BOARD — HARDWARE SPECIFICATIONS AND SETUP

B.1 Overview

The OpenBCI Cyton is an open-source, 8-channel biosignal acquisition board used in this study to record electroencephalography (EEG) signals from participants during VR lecture sessions. It was selected because it provides research-grade signal quality at a cost accessible to university labs, supports the standard 10-20 electrode placement system, and streams data wirelessly to a host computer via a USB Bluetooth dongle.



OpenBCI Cyton board kit including the board, USB Bluetooth dongle, battery pack, and electrode cables.

B.2 Technical Specifications

Table 7.1: OpenBCI Cyton board technical specifications

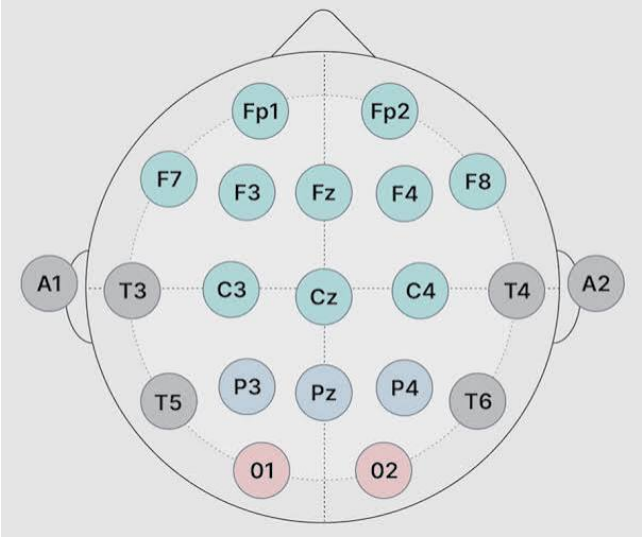
Parameter	**Value**
Number of EEG channels	8 (expandable to 16 with Daisy module)
Sampling rate	250 Hz (default)
ADC resolution	24-bit
Input impedance	>1 GΩ
Input voltage range	$\pm$187.5 μV (gain $\times$24)

Table 7.1: OpenBCI Cyton board technical specifications

Parameter	**Value**
Common mode rejection ratio (CMRR)	110 dB
Wireless protocol	Bluetooth 2.1 (RFduino module)
Wireless range	Up to 15 m (line of sight)
Power supply	4×AA batteries (onboard)
Battery life	Approximately 24 hours continuous recording
Board dimensions	73 mm × 34 mm
Weight	Approximately 15 g
Compatible electrode cap	OpenBCI Ultracortex Mark IV (fitted for 10-20 system)
Data output format	CSV via BrainFlow SDK / OpenBCI GUI

B.3 Electrode Placement

Electrodes were placed at eight positions defined by the International 10-20 system: Fp1 , Fp2 , F3 , F4 , C3 , C4 , P3 , and P4. These positions cover the frontal, central, and parietal regions, which are the primary sources of theta (4–8Hz), alpha (8–13Hz), and beta (13–30Hz) activity associated with cognitive engagement and attentional state. The reference electrode was placed on one mastoid process and the BIAS (driven right leg) electrode on the opposite mastoid. Sessions did not begin until all eight channels passed the impedance check.



International 10-20 electrode placement system. The eight positions used in this study — Fp1, Fp2, F3, F4, C3, C4, P3, and P4 — cover the frontal, central, and parietal regions shown in

teal.

B.4 Signal Acquisition Pipeline

The Cyton board was connected to the host computer via the USB Bluetooth dongle. The BrainFlow SDK was used to stream data in Python through the `BoardShim` API. Incoming samples were buffered into a circular buffer and written to a CSV file timestamped at the start of each session. The raw 8-channel EEG stream was recorded continuously at 250 Hz throughout each session, including during rest periods and lecture segments. Segment boundaries were marked by a separate timestamp log generated by the session controller.

ANNEXURE C

META QUEST 2 VR HEADSET — HARDWARE SPECIFICATIONS AND SETUP

C.1 Overview

The Standalone Virtual Reality Headset Meta Quest 2 (formerly known as Oculus Quest 2) is a virtual reality headset that is used for For this study, we created a series of academic lectures and played them for the participants in a controlled setting.For this study, we developed a set of academic lectures and played them for the participants in a controlled environment. immersive environment. It was finally selected because it does not need an external PC or the use of a tether. cables for VR content delivery and its display specifications enable comfortable extended viewing. Available widely, at an affordable price for classroom use, and viewing sessions. deployment.



Meta Quest 2 VR headset with Touch controllers as used in the study sessions.

C.2 Technical Specifications

Table 7.2: Meta Quest 2 VR headset technical specifications

Parameter	**Value**
Display type	LCD (Fast-switch)
Resolution per eye	1832×1920 pixels
Refresh rate	60, 72, 90, or 120 Hz (software selectable)
Field of view (FoV)	Approximately 89° (horizontal)
Processor	Qualcomm Snapdragon XR2
RAM	6 GB
Storage	128 GB (model used in this study)
Battery life	Approximately 2–3 hours (active use)
Weight	503 g (without facial interface)
Audio	Integrated spatial audio speakers (no headphones required)
Interpupillary distance (IPD)	Adjustable: 58, 63, or 68 mm
Tracking	6-DOF inside-out tracking (4 wide-angle cameras)
Connectivity	Wi-Fi 6 (802.11ax), Bluetooth 5.1, USB-C
Platform	Meta Quest OS (Android-based)

C.3 Use in This Study

Each participant wore the Meta Quest 2 over the OpenBCI electrode cap throughout the session. Lecture content was served through a React web application hosted locally and accessed through the Quest's built-in browser. The application managed the full session sequence automatically: 2.5 minutes of relaxation content, followed by the easy, medium, and hard lecture videos at six minutes each, with 2.5-minute relaxation breaks between segments and a final 2.5-minute relaxation period at the end — approximately 37.5 minutes of total recording time per session.

The VR environment displayed each lecture as a flat 2D screen in a neutral mid-grey virtual space with no classroom geometry, avatars, or spatial audio effects. The IPD was adjusted for each participant before the session began. Audio was delivered through the Quest's built-in speakers at a fixed volume. Once started, the application ran autonomously without any input from the participant or the researcher.

C.4 Comfort and Motion Sickness Mitigation

Sessions were capped at approximately 37.5 minutes of active VR time, with 2.5-minute relaxation breaks between lecture segments during which the visual content was calm and static. Participants were screened for a history of motion sickness before enrolment. The lecture content used a fixed camera position with no locomotion, which minimises the vestibular conflict that typically causes motion sickness in interactive VR applications. No participant reported discomfort requiring early termination

of a session.

Waleed Manzoor

main_13

Plagairsim Check

Document Details

Submission ID
trn:oid:::3618:150024074

Submission Date
Aug 31, 2026, 12:24 PM GMT+5

Download Date
Aug 31, 2026, 12:43 PM GMT+5

File Name
main_13.pdf

File Size
1.4 MB

223 Pages

46,702 Words

257,141 Characters

11% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

510 Not Cited or Quoted 10%
Matches with neither in-text citation nor quotation marks

26 Missing Quotations 1%
Matches that are still very similar to source material

8 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
8% Publications
9% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **510** Not Cited or Quoted 10%
  Matches with neither in-text citation nor quotation marks
- **26** Missing Quotations 1%
  Matches that are still very similar to source material
- **8** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
8% Publications
9% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Publication	Boxiao Yin, Haocheng Liao, Zhiqing Tang, Jianxiong Guo. "Enhancing Medical Ima... 1%
2	Publication	Viacheslav L. Bogdanov, Alexander Ya. Grigorenko, Ihor B. Chepkov, Ihor V. Odno... 1%
3	Internet	www.bahria.edu.pk <1%
4	Publication	Saeed Biabani Mahalli, Jamshid Pirgazi, Ali Ghanbari Sorkhi. "An effective knowle... <1%
5	Internet	arxiv.org <1%
6	Internet	hdl.handle.net <1%
7	Student papers	Asia Pacific University College of Technology and Innovation (UCTI) on 2025-05-30 <1%
8	Student papers	Higher Education Commission Pakistan on 2026-05-05 <1%
9	Internet	www.mdpi.com <1%
10	Internet	www.collectionscanada.ca <1%

11 Internet
www.coursehero.com <1%

12 Student papers
Liverpool John Moores University on 2026-06-18 <1%

13 Publication
Guerrache, Faiza. "Spatial and Temporal Environment Impact Analysis on People'... <1%

14 Student papers
City University on 2021-09-08 <1%

15 Internet
www.ijert.org <1%

16 Internet
ediss.sub.uni-hamburg.de <1%

17 Publication
Challa, Annajirao. "Synthetic Training for Assessment in Real-Time (STAR): Enablin... <1%

18 Publication
Rejwan Bin Sulaiman, Usman Javed Butt, Yassine Maleh, Mohammad Aljaidi, Md. ... <1%

19 Internet
d197for5662m48.cloudfront.net <1%

20 Publication
A. F. Rabbi, A. N. Zony, P. de Leon, R. Fazel-Rezai. "Preliminary results of mental w... <1%

21 Internet
publikationen.bibliothek.kit.edu <1%

22 Student papers
Babes-Bolyai University on 2026-06-16 <1%

23 Student papers
Iqra University - Islamabad on 2026-06-22 <1%

24 Student papers
University of East London on 2026-05-08 <1%

25 Student papers

Motilal Nehru National Institute of Technology on 2025-12-03 <1%

26 Internet

www.gib.tel.uva.es <1%

27 Student papers

Army Institute of Technology on 2026-06-29 <1%

28 Internet

dokumen.pub <1%

29 Student papers

Computer Science and Applications on 2026-07-30 <1%

30 Internet

apps.dtic.mil <1%

31 Internet

files01.core.ac.uk <1%

32 Internet

iris.univpm.it <1%

33 Student papers

Botswana Open University on 2026-06-04 <1%

34 Internet

discovery.dundee.ac.uk <1%

35 Publication

Rothenberger, Jay Calder. "Novel Algorithms for Interpretable and Semi-Supervis... <1%

36 Student papers

University of Westminster on 2025-04-15 <1%

37 Publication

Agbotiname Lucky Imoize, Abdulwaheed Musa, Yuichi Sei. "Handbook of AI in Wir... <1%

38 Student papers

Universiti Malaysia Pahang Al-Sultan Abdullah (UMPSA) on 2026-02-08 <1%

39 Internet

theses.hal.science <1%

40 Student papers

Liverpool John Moores University on 2024-11-07 <1%

41 Publication

Marija Savković, Marko Djapan, Carlo Caiazzo, Miloš Pušica, Arso Vukićević, Draga... <1%

42 Internet

app.uff.br <1%

43 Internet

www.measurement.sk <1%

44 Student papers

Associatie K.U.Leuven on 2025-05-23 <1%

45 Publication

Balaji, Sai Sanjay. "Computational Approaches to Epilepsy: Graph Networks for Lo... <1%

46 Student papers

Bocconi University on 2026-06-15 <1%

47 Publication

Darabi, Seyed Shayan. "Effects of Auditory Cueing Strategy on Learning and Perfo... <1%

48 Student papers

Maulana Azad National Institute of Technology Bhopal on 2026-06-22 <1%

49 Publication

Rolandino, Giovanni. "Shallow Neural Networks and High-Density Electromyogra... <1%

50 Student papers

University of Hull on 2026-05-05 <1%

51 Internet

backend.orbit.dtu.dk <1%

52 Student papers

University of Glasgow on 2026-08-14 <1%

53 Student papers
University of Oxford on 2024-03-15 <1%

54 Internet
hal.science <1%

55 Internet
www.tugraz.at <1%

56 Student papers
Aston University on 2026-04-27 <1%

57 Student papers
City University of Hong Kong on 2026-03-04 <1%

58 Student papers
Higher Education Commission Pakistan on 2012-05-09 <1%

59 Publication
Taylor K. Calibo, Justin A. Blanco, Samara L. Firebaugh. "Cognitive stress recogniti... <1%

60 Internet
www.frontiersin.org <1%

61 Internet
www.themoonlight.io <1%

62 Publication
Burak Akdemir, Seniha Esen Yuksel. "Software-Only Registration and Cross-Spect... <1%

63 Student papers
TU Delft on 2026-03-23 <1%

64 Student papers
Taibah University on 2026-04-25 <1%

65 Student papers
Technical University of Košice on 2026-06-24 <1%

66 Student papers
University of Hong Kong on 2025-02-24 <1%

67 Student papers
Cranfield University on 2018-08-20 <1%

68 Student papers
Georgia Institute of Technology Main Campus on 2026-04-06 <1%

69 Student papers
Higher Education Commission Pakistan on 2026-06-02 <1%

70 Student papers
Queen Mary and Westfield College on 2026-08-18 <1%

71 Student papers
Swinburne University of Technology on 2020-05-19 <1%

72 Publication
Teixeira, Tiago Oliveira. "Image Classification of OCT Scans Using Machine Learni... <1%

73 Student papers
Universiti Sains Malaysia on 2025-04-28 <1%

74 Student papers
Maastricht University on 2026-06-04 <1%

75 Student papers
University of Dundee on 2024-12-01 <1%

76 Student papers
University of Lancaster on 2024-09-05 <1%

77 Internet
vital.seals.ac.za:8080 <1%

78 Internet
www.arxiv-vanity.com <1%

79 Internet
www.diva-portal.org <1%

80 Student papers
Chester College of Higher Education on 2022-10-12 <1%

81	Publication	Samta Rani, Sarfaraz Masood, Danish Raza Rizvi. "A Bi-LSTM based approach for C... <1%
82	Publication	Su Liu, Andres Kanner, Naiara Garcia Losarcos, Jonathan Jagid, Yingchun Zhang. "... <1%
83	Student papers	University of Monastir on 2026-05-19 <1%
84	Publication	Weikai Zhao, Xiaomeng Tong, Xujun Lu, Yan Shi, Maolin Cai, Chuang Gao. "Multi-o... <1%
85	Internet	geneticsmr.com <1%
86	Internet	goyalpramod.github.io <1%
87	Internet	gupea-preproduction.ub.gu.se <1%
88	Internet	ijirt.org <1%
89	Internet	openscholar.dut.ac.za <1%
90	Internet	pure.iiasa.ac.at <1%
91	Internet	vuir.vu.edu.au <1%
92	Publication	Changhui Lee, Jinmin Lee, Taeheon Kim, Hyunjin Lee, Aisha Javed, Minkyung Chu... <1%
93	Student papers	ICTS on 2026-08-26 <1%
94	Publication	Junjie Li, Xu Chi, Dongsheng Zhu, Yixin Chen. "GTP-Net: A Graph–Transformer Fra... <1%

95 Student papers
LTI on 2026-08-25 <1%

96 Student papers
Liverpool John Moores University on 2024-12-23 <1%

97 Student papers
Monash University on 2026-06-08 <1%

98 Student papers
Tilburg University on 2026-06-29 <1%

99 Student papers
United International University on 2026-07-21 <1%

100 Student papers
University of Sahiwal on 2026-05-09 <1%

101 Student papers
University of Wollongong on 2025-03-26 <1%

102 Internet
export.arxiv.org <1%

103 Internet
jin.imrpress.com <1%

104 Internet
journal2.uad.ac.id <1%

105 Internet
mdpi-res.com <1%

106 Internet
psasir.upm.edu.my <1%

107 Internet
scholar.ppu.edu <1%

108 Publication
C. Guedes Soares, F.C. Salvado. "Innovations in Maritime Technology and Enginee... <1%

109 Student papers
COMSATS Institute of Information Technology, Attock on 2026-07-05 <1%

110 Student papers
Curtin University of Technology on 2025-10-24 <1%

111 Publication
Farzaneh, Behnam. "Deep Transfer Learning-Based DDoS Attack Detection in 5G ... <1%

112 Publication
Huang, Chao. "Controllable Generative Modeling for Multimodal Perception and ... <1%

113 Student papers
Korea University on 2020-12-31 <1%

114 Publication
S. Almonti, A. M. Cipriani, V. Villani, S. Rinalduzzi. "Reversible myoclonus in a patie... <1%

115 Student papers
The University of Manchester on 2025-05-05 <1%

116 Student papers
Tilburg University on 2026-08-26 <1%

117 Student papers
Universidad Carlos III de Madrid - EUR on 2026-07-01 <1%

118 Student papers
University of Dundee on 2026-04-16 <1%

119 Student papers
University of Edinburgh on 2019-08-15 <1%

120 Internet
biblio.ugent.be <1%

121 Internet
dspace.dtu.ac.in:8080 <1%

122 Internet
iiste.org <1%

123 Internet
knowledgecommons.lakeheadu.ca <1%

124 Internet
pdw.hanford.gov <1%

125 Internet
theses.lib.polyu.edu.hk <1%

126 Internet
vdocuments.net <1%

127 Internet
www.neuroregulation.org <1%

128 Publication
Dulabh, Meera R.. "Analysing Cognitive Emotional and Behavioural Engagement t... <1%

129 Student papers
Imperial College of Science, Technology and Medicine on 2025-06-13 <1%

130 Publication
Kazi Abdul Mannan, Arifa Akter Shapla. "Explainable CNN for Automated Lung Op... <1%

131 Student papers
Lahore Garrison University-Lahore. on 2026-04-15 <1%

132 Publication
Md Aurongojeb Lishad, Ratul Hasan Ankon, Md. Adnan Arafat. "Ultra-Lightweight... <1%

133 Student papers
Nepal College of Information Technology on 2026-02-27 <1%

134 Student papers
The University of Manchester on 2016-09-19 <1%

135 Student papers
The University of Manchester on 2025-04-27 <1%

136 Student papers
Trinity College Dublin on 2026-08-04 <1%

137 Student papers
University College London on 2025-05-25 <1%

138 Student papers
University of Auckland on 2019-09-26 <1%

139 Student papers
University of Edinburgh on 2015-01-12 <1%

140 Student papers
University of Essex on 2026-08-21 <1%

141 Student papers
University of Northumbria at Newcastle on 2024-09-01 <1%

142 Student papers
University of Portsmouth on 2022-04-28 <1%

143 Student papers
University of Sheffield on 2021-03-18 <1%

144 Student papers
University of Ulster on 2008-08-26 <1%

145 Internet
danielbryant.co.uk <1%

146 Internet
doaj.org <1%

147 Internet
ebookreading.net <1%

148 Internet
jeremybejarano.com <1%

149 Internet
regisportfolio.com <1%

150 Publication
"Intelligence-Driven Circular Economy", Springer Science and Business Media LLC... <1%

151 Publication

Andrew Wood. "Physiology, Biophysics, and Biomedical Engineering", CRC Press, ... <1%

152 Publication

Asish, Sarker Monojit. "Detecting Distracted Students in Virtual Reality Environm... <1%

153 Student papers

Aston University on 2019-09-09 <1%

154 Student papers

BITS, Pilani-Dubai on 2011-12-26 <1%

155 Student papers

Brunel University on 2023-09-01 <1%

156 Publication

Chang S. Nam, Anton Nijholt, Fabien Lotte. "Brain–Computer Interfaces Handboo... <1%

157 Publication

Dakota Hester, Vitor S. Martins, Lucas B. Ferreira, Thainara M.A. Lima. "Learning ... <1%

158 Publication

Fotis P. Kalaganis, Nikos A. Laskaris, Elisavet Chatzilari, Spiros Nikolopoulos, Ioan... <1%

159 Publication

Gaurav Aggarwal, Rishabh Dev Shukla, Innocent Ewean Davidson, Ramesh Bansal... <1%

160 Publication

Haiming Yao, Wenyong Yu, Wei Luo, Zhenfeng Qiang, Donghao Luo, Xiaotian Zha... <1%

161 Student papers

Higher Education Commission Pakistan on 2016-03-22 <1%

162 Student papers

Higher Education Commission Pakistan on 2023-03-15 <1%

163 Publication

Hossain, M.D. Shakhawat. "Signal Processing Combined with Machine Learning fo... <1%

164 Student papers

IT University of Copenhagen on 2026-07-14 <1%

165 Student papers
Independent University, Bangladesh (IUB) on 2026-06-14 <1%

166 Publication
Inge Kersbergen. "Individual differences in attentional processing of responsible ... <1%

167 Student papers
Institute of Research & Postgraduate Studies, Universiti Kuala Lumpur on 2025-0... <1%

168 Publication
Islam, Md. Khayrul. "MIML: Multiplex Image Machine Learning for High Precision ... <1%

169 Publication
Jing Yang, Denny Yu. "A Novel Mental Workload-driven Adaptive Training Framew... <1%

170 Publication
Joseph B. Hopfinger, Scott D. Slotnick. "The Cognitive Neuroscience of Attention - ... <1%

171 Publication
Kaveti Pavan, Swarubini P P J, Ankit Singh, Digvijay S. Pawar, Ramakrishnan Swa... <1%

172 Publication
Liu, Jia. "Investigating the Impact of Virtual Global Landmarks on Spatial Learnin... <1%

173 Student papers
Liverpool John Moores University on 2026-07-03 <1%

174 Student papers
Loughborough University on 2014-10-02 <1%

175 Student papers
Monash University on 2025-06-06 <1%

176 Publication
Moustafa, Israa Ashraf. "A Deep Learning Framework for Quantitative Stress Esti... <1%

177 Student papers
Mt San Antonio College on 2026-07-27 <1%

178 Publication
Orozco, Jesús Antonio. "Quantifying and Utilizing Human Trust in the Adaptive Co... <1%

179 Publication

Peter A. Akor, Godwin Enemali, Usman Muhammad, Rajiv Ranjan Singh, Hadi Larij... <1%

180 Publication

Savković, Marija I.. "Унапређење Ефективности Производних Процеса Приме... <1%

181 Student papers

Technical University of Cluj-Napoca on 2025-07-24 <1%

182 Student papers

Thapar University, Patiala on 2015-07-10 <1%

183 Student papers

Tilburg University on 2023-06-23 <1%

184 Student papers

UIET on 2026-05-27 <1%

185 Student papers

Universiteit Hasselt on 2026-06-16 <1%

186 Student papers

University College London on 2018-04-17 <1%

187 Student papers

University College London on 2026-05-01 <1%

188 Student papers

University of Edinburgh on 2025-04-22 <1%

189 Student papers

University of Essex on 2026-08-25 <1%

190 Student papers

University of Glasgow on 2026-04-21 <1%

191 Student papers

University of New South Wales on 2026-04-12 <1%

192 Student papers

University of Southampton on 2025-05-02 <1%

193 Student papers
University of Sydney on 2024-01-08 <1%

194 Student papers
University of Technology, Sydney on 2025-11-06 <1%

195 Student papers
University of Ulster on 2021-05-21 <1%

196 Internet
archive.org <1%

197 Internet
assets-eu.researchsquare.com <1%

198 Internet
core.ac.uk <1%

199 Internet
heritagesciencejournal.springeropen.com <1%

200 Internet
ijaas.iaescore.com <1%

201 Internet
iristown.engr.utk.edu <1%

202 Internet
ntnuopen.ntnu.no <1%

203 Internet
par.nsf.gov <1%

204 Internet
pp-rai.pwr.edu.pl <1%

205 Internet
pub.tik.ee.ethz.ch <1%

206 Internet
public-pages-files-2025.frontiersin.org <1%

207 Internet
repository.up.ac.za <1%

208 Internet
s3-ap-southeast-2.amazonaws.com <1%

209 Internet
unige.iris.cineca.it <1%

210 Internet
uvadoc.uva.es <1%

211 Internet
www.fedoa.unina.it <1%

212 Internet
www.preprints.org <1%

213 Publication
"MultiMedia Modeling", Springer Science and Business Media LLC, 2026 <1%

214 Publication
Abuhassan, Kamal, Damien Coyle, and Liam P. Maguire. "Investigating the Neural... <1%

215 Student papers
Anna University on 2026-06-22 <1%

216 Publication
Barnard, Michael A.. "Enhancing Patient Experience in PET-CT: The Potential of Vir... <1%

217 Student papers
Kean University on 2025-08-28 <1%

218 Student papers
Liverpool John Moores University on 2026-02-12 <1%

219 Publication
Naveen, Vinayak. "Nasal Pressure Signal Quality Assessment in Polysomnograph... <1%

220 Student papers
University Politehnica of Bucharest on 2018-03-20 <1%

221 Student papers
University of Warwick on 2013-01-24 <1%

222 Publication
Xinquan Yang, Xuguang Li, Xuechen Li, Wenting Chen, Linlin Shen, Xin Li, Yongqia... <1%

223 Student papers
Charles University in Prague on 2026-08-07 <1%

224 Publication
Gregoire Hugues Cattan, Anton Andreev, Cesar Mendoza, Marco Congedo. "A Co... <1%

225 Student papers
Institute of Graduate Studies, UiTM on 2015-01-28 <1%

226 Publication
Khaled Wael Ezzat. "Optimizing Deep Learning for Skin Cancer: A Comparative St... <1%

227 Student papers
Moodle2025 on 2026-08-09 <1%

228 Publication
Pushpa Choudhary, Sambit Satpathy, Arvind Dagur, Dhirendra Kumar Shukla. "Re... <1%

229 Publication
S.P. Jani, M. Adam Khan. "Applications of AI in Smart Technologies and Manufactu... <1%

230 Student papers
UT, Dallas on 2014-11-20 <1%

231 Publication
V. Sekhar, P. Ajay Kumar Reddy, K. Logesh, T. Raghunadha Reddy, Dara Raju. "Ne... <1%

232 Publication
Widyanti, Ari, Addie Johnson, and Dick de Waard. "Adaptation of the Rating Scale ... <1%



Waleed Manzoor

FinaIIy-1-60

Pliagairsim Check

Document Details

Submission ID
trn:oid:::3618:150599046

Submission Date
Sep 4, 2026, 2:27 PM GMT+5

Download Date
Sep 4, 2026, 2:33 PM GMT+5

File Name
FinaIIy-1-60.pdf

File Size
1.5 MB

61 Pages

17,557 Words

113,489 Characters

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely human generated as AI generated and likely AI generated as human generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.