UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

A Comparative Analysis of the Effectiveness of Contrastive Learning for P300 Classification

David Miguel Baptista Conceição

Mestrado em Informática

Dissertação orientada por:
Prof. Doutor Manuel João Caneira Monteiro da Fonseca

2024

To my Family and Friends

Acknowledgments

I would like to start by acknowledging my supervisor, Prof. Manuel João Caneira Monteiro da Fonseca, who has persistently accompanied me through the process of creating this document, providing the most valuable expertise and support. Our regular weekly meetings, where Prof. Manuel Fonseca would give me his advice, feedback, and guidance were crucial at every step of this journey, a journey that would have been much more challenging if it was not for him. For the perseverance, patience, and dedication I express my heartfelt appreciation. I would also like to thank FCUL, and in particular LASIGE, for providing me with the essential resources that made this dissertation possible. Mainly for providing a dedicated space where I could work on this dissertation, and for giving me access to the remote server crucial for running this entire project.

I also want to thank my family, especially my parents, my aunt, and my grandfather, whose unconditional love and support given to me throughout the years, has been a constant source of motivation, and strength through the best moments and especially during challenging times. To my friends, I want to express my deep appreciation for their friendship, the moments we share, and the support they have given me, especially over the past few months. Their words of encouragement kept me motivated to see this through.

Resumo

As Interfaces Cérebro-Computador (*Brain-Computer Interfaces*, ou *BCIs*, em inglês) interpretam sinais de eletroencefalografia (EEG) e traduzem-nos em comandos utilizados para controlar dispositivos. Estes sistemas podem basear-se em diferentes tipos de sinais EEG, incluindo os Potenciais Relacionados a Eventos (*Event-Related Potentials*, *ERPs*). Um dos componentes mais notáveis dos ERPs é a onda P300, gerada de forma inconsciente quando reconhecemos algo esperado.

Os sinais P300 são amplamente utilizados em sistemas conhecidos como *spellers*, que permitem aos utilizadores escrever texto num computador através da deteção desses sinais. Estes sistemas, denominados P300 *spellers*, empregam o *oddball paradigm* para elicitar os sinais. Nesse processo, o utilizador é apresentado com uma matriz contendo diversos caracteres do alfabeto e comandos adicionais. A tarefa do utilizador é focar-se no carácter que deseja soletrar enquanto as diferentes linhas e colunas da matriz se iluminam aleatoriamente. Quando a linha e a coluna que incluem o caractere pretendido se iluminam, o cérebro emite um sinal P300 que é captado pelo sistema.

Estes sistemas integram, entre outros elementos, modelos de aprendizagem automática capazes de distinguir sinais P300 de sinais não-P300, permitindo assim que utilizadores, particularmente aqueles com deficiências motoras, possam comunicar de forma eficaz.

No entanto, estes sistemas enfrentam alguns desafios. Um dos principais desafios é a natureza ruidosa dos sinais EEG. Outro fator crítico é a elevada variabilidade desses sinais, tanto entre diferentes indivíduos como no mesmo indivíduo ao longo do tempo. A consequência mais evidente desses desafios é a necessidade de realizar uma sessão de calibração dos modelos de aprendizagem automática antes de utilizar o sistema. Estas sessões de calibração são, geralmente, demoradas, o que pode provocar fadiga e frustração nos utilizadores. Esses fatores, por sua vez, comprometem a qualidade dos sinais P300, resultando numa redução do desempenho dos modelos.

Dessa forma, há uma necessidade crescente de desenvolver modelos e sistemas mais independentes dos utilizadores. Ou seja, é essencial criar modelos que consigam classificar sinais de vários utilizadores sem exigir longas sessões de calibração. As Redes Neuronais e as técnicas de Aprendizagem Profunda têm ganho popularidade neste paradigma devido à sua capacidade de extrair características abstratas dos dados e à redução da dependência de etapas de pré-processamento. Contudo, devido à dependência da fase de calibração, muitas dessas arquiteturas são extremamente pequenas para evitar problemas de *overfitting*. Outra abordagem seria utilizar grandes modelos capazes de explorar grandes quantidades de dados disponíveis, aprendendo a generalizar bem para

diferentes indivíduos. Para isso, seria necessária uma etapa de pré-treino robusta que permitisse aos modelos aprender características significativas dos sinais. Esses modelos pré-treinados poderiam, então, adaptar-se a novos utilizadores com menos dados. Esta abordagem é semelhante à utilizada em modelos conhecidos, como o ChatGPT e o BERT.

Neste contexto, a estratégia de pré-treino torna-se essencial. A maioria dos modelos no domínio do P300 utiliza aprendizagem supervisionada. No entanto, técnicas de aprendizagem contrastiva, como o SimCLR e o SupCon, têm mostrado sucesso em áreas como classificação de imagens e processamento de linguagem natural. Estas técnicas baseiam-se na ideia de que dados semelhantes devem ser representados de forma similar pelos modelos. Assim, os modelos treinados com essas estratégias aprendem características significativas ao comparar pares de dados semelhantes (*positive pairs*) com dados distintos (*negative pairs*). Embora tais estratégias estejam a ser exploradas no âmbito dos BCIs, ainda são raros os trabalhos que estudam estes métodos especificamente no paradigma do P300.

O nosso estudo insere-se neste contexto. Após revisão da literatura, identificámos uma lacuna no estudo do impacto das técnicas de aprendizagem contrastiva no paradigma do P300. Assim, realizámos uma análise comparativa entre técnicas de aprendizagem contrastiva e a aprendizagem supervisionada. Utilizámos três abordagens de pré-treino: *SimCLR*, *SupCon* (aprendizagem contrastiva) e *Supervised* (supervisionada). Além disso, introduzimos uma nova estratégia para criar *positive* e *negative pairs*, denominada *Progressive Evokeds*, que consiste em criar um sinal *evoked* a partir da média de vários sinais individuais. Estas abordagens foram usadas para pré-treinar três redes neuronais distintas: *EEGNet*, *EEG-Inception* e *Conformer*.

Para avaliar o desempenho dos modelos pré-treinados, desenhámos dois estudos de avaliação: (1) Avaliação *Intra-Dataset*, e (2) Avaliação *Cross-Dataset*. Os dados de pré-treino foram obtidos do *GIB-UVa ERP-BCI Dataset*. Na Avaliação *Intra-Dataset*, os modelos foram testados com dados do mesmo *dataset*, enquanto na Avaliação *Cross-Dataset* foram testados com dados dos *datasets BCI Competition III Dataset II* e *ALS Dataset*. Para o *BCI Competition III Dataset II* assim como na Avaliação *Intra-Dataset*, a calibração foi realizada com dados do próprio utilizador (*subject-dependent*), enquanto no *ALS Dataset* os modelos foram calibrados com dados de alguns utilizadores e testados em novos utilizadores (*subject-independent).*

Os resultados da Avaliação *Intra-Dataset* indicaram que as técnicas de aprendizagem contrastiva podem beneficiar o desempenho dos modelos, mas apenas sob condições específicas. Esse benefício foi observado para o *SupCon*, especialmente quando houve um aumento na quantidade de dados disponíveis para calibração dos modelos pré-treinados. Por outro lado, o *SimCLR* mostrou-se prejudicial em todas as arquiteturas, alcançando resultados inferiores às outras estratégias. De maneira geral, *SupCon* e *Supervised* exibiram desempenhos semelhantes, embora *SupCon* tenha superado ligeiramente a abordagem supervisionada à medida que a quantidade de dados aumentava.

Na Avaliação *Cross-Dataset*, os resultados apresentaram um cenário distinto e mais promissor. Nesse contexto, o *SimCLR* destacou-se como a estratégia de pré-treino mais eficaz, atingindo os

melhores resultados no *BCI Competition III Dataset II* e no *ALS Dataset*. No entanto, o impacto das técnicas de aprendizagem contrastiva e a eficácia na classificação *cross-dataset* demonstraram ser altamente dependentes da arquitetura do modelo, uma observação também feita na Avaliação *Intra-Dataset*.

Por exemplo, o *EEG-Inception* foi a arquitetura que menos beneficiou dessas técnicas, particularmente do *SimCLR*. Além disso, apresentou desempenho inferior nos dois *datasets* da Avaliação *Cross-Dataset* para todas abordagens de pré-treino, sugerindo que as limitações estão mais relacionadas à própria arquitetura do que à escolha da estratégia de pré-treino. Apesar disso, na Avaliação *Intra-Dataset*, *EEG-Inception* obteve maior sucesso com o *SupCon*, que emergiu como a melhor escolha para o seu pré-treino. Em contraste, e voltando à Avaliação *Cross-Dataset*, o *EEGNet* alcançou os melhores resultados em ambos os *datasets* com o *SimCLR*, evidenciando um maior benefício proveniente da aprendizagem contrastiva.

Estas conclusões representam resultados preliminares sobre o impacto e a eficácia das estratégias de aprendizagem contrastiva em abordar os desafios enfrentados pelos sistemas BCI, especialmente pelos P300 *spellers*. É importante destacar que o desempenho destas estratégias está fortemente dependente da nova metodologia de criação de *augmentations* apresentada neste trabalho, denominada *Progressive Evokeds*. Para compreender plenamente o impacto dessa abordagem, seria necessário um estudo comparativo mais abrangente envolvendo diferentes técnicas de *augmentation*.

Além disso, investigações mais detalhadas são imprescindíveis para explorar outros aspectos destas metodologias, como o uso de modelos maiores, ajustes de hiperparâmetros, análise de diferentes técnicas de aprendizagem contrastiva e avaliação em novos *datasets*. Esses estudos adicionais não apenas aprofundariam o entendimento sobre o impacto das estratégias propostas na classificação de sinais P300, mas também seriam essenciais para validar os resultados apresentados neste trabalho.

De forma geral, os resultados obtidos neste estudo são promissores, sugerindo que as estratégias de aprendizagem contrastiva podem melhorar o desempenho em cenários *cross-dataset*, onde a generalização entre diferentes conjuntos de dados é crucial. Estas descobertas são encorajadoras e constituem o primeiro passo para o estudo da possibilidade de construir modelos maiores, treinados com grandes quantidades de dados provenientes de múltiplos *datasets*, utilizando aprendizagem contrastiva para capacitar os modelos a generalizar eficazmente para dados de diversos utilizadores. Isso poderia reduzir significativamente o tempo necessário para calibração, ou até eliminá-lo, tornando os sistemas BCI mais eficientes, práticos e acessíveis.legere me lius quod ii legunt saepius.

Palavras-chave: Interfaces Cérebro-Computador, P300, Classificação de Sinais, Aprendizagem Contrastiva, Redes Neuronais

Abstract

Brain-Computer Interfaces (BCIs) translate brain activity into precise commands for controlling external devices, providing crucial assistance to individuals with motor disabilities. Within this domain, P300 Speller systems serve as a vital communication tool. However, these systems face significant challenges, including the inherently noisy nature of EEG data and the high variability of signals both across and within individuals. This variability often leads these systems to require an extensive calibration phase, reducing the practicality of P300 spellers in real-world applications. Developing robust models that generalize effectively across users is therefore a key objective in this field.

Contrastive learning approaches have recently gained attention for their ability to produce models capable of extracting meaningful features from data. While these techniques have shown great success in domains like computer vision and natural language processing, their application to the P300 paradigm remains underexplored. This study addresses this gap by conducting a comparative analysis of three pre-training strategies: SimCLR, SupCon, and Supervised learning. These approaches were evaluated using three state-of-the-art neural network architectures, EEGNet, EEG-Inception, and Conformer.

The pre-trained models were evaluated in both intra-dataset and cross-dataset scenarios. The results indicate that the impact of contrastive learning is highly dependent on the model architecture and on the evaluation setup. In an intra-dataset scenario, SimCLR exhibited the worse performances across all models, while SupCon and Supervised displayed comparable results with SupCon achieving slightly higher performances for higher values of retraining data. However, in cross-dataset scenarios, contrastive learning approaches, particularly SimCLR, demonstrated superior performance compared to supervised learning, showcasing their ability to generalize effectively across diverse data distributions. These findings underscore the potential of contrastive learning and their robustness to address the challenges of variability and reduce the reliance on extensive calibration in P300 speller systems, by leveraging data coming from multiple sources.

Keywords: Brain-Computer Interfaces (BCI), P300, Signal Classification, Contrastive Learning, Neural Networks

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The human brain is a marvel of complexity, housing approximately 86 billion neurons, each with thousands of synapses that form an intricate network responsible for our thoughts, emotions, and actions. As our understanding of neurobiology and neuroengineering continues to evolve, so does our ability to tap into the world of the human mind. Brain-Computer Interfaces (BCIs), an interdisciplinary field at the intersection of neuroscience, engineering, and computer science, offer the possibility of bridging the gap between the human brain and external devices. BCIs hold the potential to revolutionize the way we interact with technology, assist individuals with severe neurological conditions, and provide novel insights into the functioning of the brain.

BCI systems measure specific features of brain activity and subsequently translate these signals into precise commands for controlling external devices. Various methods are leveraged by different BCI systems, including slow cortical potentials, sensory-motor signals, and event-related potentials (ERPs). Our work focuses on BCIs that rely on ERPs, with specific attention directed towards a notable component within these ERP signals known as the P300 component.

1.1 Motivation

P300 BCIs play a crucial role in systems like spellers. These spellers offer a lifeline to individuals with physical impairments like Amyotrophic lateral sclerosis (ALS), enabling them to communicate.

However, researchers encounter several challenges in developing practical P300 BCI spellers. The first challenge is the inherently noisy nature of signals acquired from EEG devices. Secondly, P300 signals exhibit significant variability not only among subjects but also within the same subject across different sessions. Consequently, most P300 BCIs need an initial calibration phase. During this phase, the BCI system gathers the brain signals of the users to train a machine-learning model tailored to their specific data. This is known as a subject-dependent BCI. This calibration phase, however, is often an extended and laborious process, which can potentially result in participant boredom or fatigue. These factors, in turn, may weaken the clarity and robustness of the P300 signal, thereby complicating the classification task even more. Therefore, it is essential to develop systems that are robust enough to operate in a subject-independent manner or close to it, meaning they can achieve good performance without relying as much on subject-specific data. This would eliminate, or at least reduce, the need for a calibration phase.

BCI systems face an additional obstacle: due to the noise in EEG data, multiple trials are required to accurately recognize a P300 signal, resulting in a low word-spelling rate. This leads to increased user fatigue, further complicating the classification problem. Therefore, efforts must also focus on reducing the dependency of these systems on repeated trials.

Neural networks have become increasingly popular in this area of research because of their ability to extract high-level features directly from EEG signals. They offer the advantage of eliminating the need for a dedicated feature extraction phase and simplifying the preprocessing stage [26, 34]. However, this advantage often comes with the drawback of requiring larger amounts of training data to ensure proper generalization.

To minimize the data required for calibrating these models, most neural network architectures in this paradigm are designed with a small number of training parameters to reduce the risk of overfitting, given the limited data available. Larger models have the potential to learn more discriminative features but would require significantly more data. For larger networks to be practical, they would have to generalize effectively across users. In this approach, the objective would be to develop a foundational model capable of capturing broad, cross-subject invariant features, enabling adaptation to downstream tasks with minimal calibration data, in this case, adapting to new users. These foundational models typically undergo a pre-training phase that leverages large amounts of data from multiple sources. Such models must effectively address challenges stemming from variations in dataset distributions, learning to extract meaningful features leading to robust generalization. Among the various pre-training approaches, contrastive learning has emerged as one of the most popular.

The majority of studies applying neural networks to the P300 classification problem rely on a supervised learning approach to train these models. However, contrastive learning techniques, which aim to learn meaningful feature representations by contrasting similar and dissimilar sample pairs, have demonstrated success in other domains, such as image classification, and natural language processing. These methods hold the potential to address the challenges specific to P300 classification, including improving generalization across subjects and reducing calibration data requirements.

1.2 Goals and Research Questions

Within the broader BCI paradigm, contrastive learning techniques have been employed as a pre-training strategy in some studies [19, 25]. However, within the specific context of P300 classification, there is a noticeable gap in the literature regarding the application of contrastive learning.

Given the effectiveness of neural networks in P300 classification and the success of contrastive learning in other domains, this work aimed to conduct a comparative analysis of different contrastive learning techniques against the standard supervised learning approach. The focus was to evaluate how these strategies perform under varying scenarios. To achieve this, we set out to explore and answer the following research questions:

- **RQ1**: Can contrastive learning techniques enhance the performance of neural networks in P300 speller systems?
- **RQ2**: Can contrastive learning techniques improve subject-independent classification or reduce the calibration data needed for effective performance?
- **RQ3**: How do different contrastive learning frameworks (e.g. SimCLR, SupCon) compare in addressing the challenges of P300 classification?
- **RQ4**: How do networks pre-trained with contrastive learning compare with state-of-the-art approaches in P300 classification?

1.3 Developed Solution

To answer these questions we explored three distinct pre-training strategies: (1) Supervised, (2) SimCLR, and (3) SupCon. These approaches were applied to three different model architectures, (1) EEGNet, (2) EEG-Inception, and (3) Conformer using an available dataset, GIB-UVa ERP-BCI, with data collected from 73 subjects while performing a Row Column Paradigm (RCP) task [12].

The performance of each pre-trained model was assessed through two distinct evaluation studies: (1) Intra-Dataset Evaluation, which measures classifier performance on data within the same dataset, and (2) Cross-Dataset Evaluation, which evaluates model performance on data from a different dataset. For the latter, data from two separate datasets (BCI Competition III Dataset II, and ALS Dataset) was used. These evaluations were conducted under both subject-dependent and subject-independent frameworks to assess the generalizability and adaptability of the models.

1.4 Results and Contributions

Results from the Intra-Dataset Evaluation reveal that Supervised outperformed the other strategies when only limited fine-tuning data was available, while SupCon achieved better results with larger amounts of fine-tuning data. SimCLR, however, showed the lowest accuracy among the three strategies in this evaluation. These insights were observed across all networks. Among the models, EEG-Inception emerged as the best-performing network, demonstrating strong benefits from both Supervised and SupCon pre-training approaches.

In contrast, the Cross-Dataset Evaluation presented a different outcome. Here, EEG-Inception was the worst-performing architecture, followed by Conformer, while EEGNet demonstrated superior performance. Regarding pretraining strategies and focusing on EEGNet and Conformer (given EEG-Inception's poor results), SimCLR achieved the best performance, followed by SupCon, with Supervised performing the worst.

A comparison of the results between the two evaluation studies suggests that contrastive learning techniques are particularly effective for cross-dataset scenarios, where generalization across diverse dataset distributions and the ability to address the inherent challenges of cross-dataset evaluation are crucial.

This study offered early insights into the potential benefits of contrastive learning pre-training methods for the P300 classification paradigm. Additionally, we introduced a novel augmentation method for contrastive learning called Progressive Evokeds, which creates different views of the same instance by pairing signal epochs with their corresponding evoked signals to form positive pairs. Moreover, several modifications to the model architectures were introduced and detailed in this study, offering possible guidelines for designing new architectures tailored to the specific requirements of contrastive learning methods.

1.5 Structure of the document

This Dissertation is organized as follows. Chapter II provides essential background on the P300 component, explaining its nature and how it is utilized in BCI systems. It further discusses the significance of these systems, their challenges, and concludes with an introduction to contrastive learning. In Chapter III we review the relevant literature, exploring key studies in the field and highlighting some of the most effective solutions to the challenges identified in Chapter II. Chapter IV outlines the experimental protocol and methodology used in this research, detailing the pre-training strategies and the model architectures employed in the study. In Chapter V results of the study are presented and interpreted, discussing their significance and broader implications. Finally, Chapter VI summarizes the key findings and conclusions, and suggests directions for future research.

Chapter 2

Background

In this chapter, we provide context for our problem and introduce key concepts within the EEG paradigm and, more specifically, the P300 paradigm, which are crucial for understanding the problem. We also introduce the contrastive learning paradigm, placing it within the wider context of machine learning, and exploring some technical details.

2.1 EEG Data

Electroencephalography (EEG) is a technique used to record an electrogram of the brain's spontaneous electrical activity. The signals captured by EEG reflect the postsynaptic potentials of pyramidal neurons located in the neocortex and allocortex. This method is generally non-invasive, with electrodes positioned along the scalp.

As with most things, there are advantages and disadvantages to using EEG data when analyzing brain activity. EEG has excellent temporal resolution, meaning it can measure changes in brain activity with high precision in terms of time. This allows researchers to observe and analyze the brain's response to stimuli in real time. It is a non-invasive technique where electrodes are placed on the scalp's surface, eliminating the need for surgery or penetration of the skull. EEG is also sensitive to a wide range of cognitive and sensory processes. It can capture subtle changes in brain activity associated with different mental states. Lastly, compared to some other neuroimaging techniques, EEG equipment is relatively affordable and portable. This makes it accessible for researchers in diverse settings and facilitates applications in clinical environments. On the other hand, it has limitations in spatial resolution. The signals measured on the scalp represent the summed activity of large populations of neurons, making it challenging to pinpoint the precise location of neural activity within the brain [3]. Another drawback is its artifact susceptibility. Various artifacts can affect EEG recordings, such as muscle activity, eye movements, and external electrical interference. Advanced signal processing techniques are often required to distinguish genuine brain signals from artifacts. Lastly, another major challenge when using EEG data is the fact that there is considerable variability in EEG patterns across individuals. This variability can make it difficult to establish universal norms and may require individualized interpretation in certain cases.

As said before, EEG data is recorded by a set of electrodes placed on the human scalp. This placement should not be arbitrary and should follow the International 10–20 system. This system was developed to ensure consistency in recording and reporting EEG findings, it provides a framework for locating electrodes on the scalp relative to underlying brain regions. The name ”10-20” derives from the fact that the distances between adjacent electrodes are either 10% or 20% of the total front-back or right-left distance of the skull. Each electrode is labeled according to the area, where is located. These areas are the pre-frontal (Fp), frontal (F), temporal (T), parietal (P), occipital (O), and central (C) areas. Some electrodes are also labeled with letters such as ”Z” (midline), and ”A” (auricular points). The system incorporates an odd-even rule for hemisphere and region identification. Odd numbers generally refer to electrodes on the left side, while even numbers correspond to the right side. Figure 2.1 depicts an EEG electrode setting following the 10-20 system.

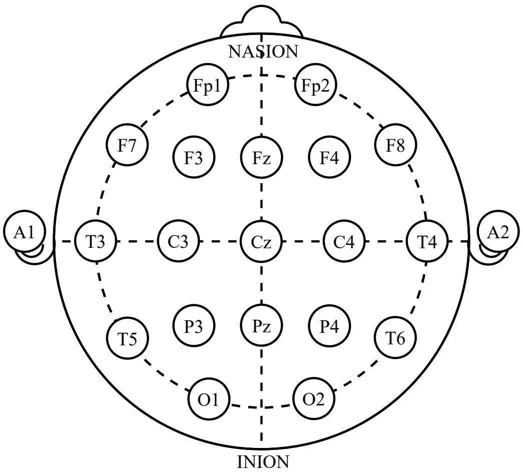


Figure 2.1: EEG electrode positions in the International 10–20 system

2.2 P300 Signal

P300 signals are a component of Event-Related Potentials (ERPs). ERPs are distinct patterns of brain activity that are time-locked to specific events or stimuli. They are thought to reflect the summed activity of postsynaptic potentials produced when a large number of similarly oriented cortical pyramidal neurons (in the order of thousands or millions) fire in synchrony [38]. ERPs are obtained by averaging EEG activity across multiple trials, aligning the data to the onset of a particular stimulus or event. This averaging process helps to eliminate some noise and therefore, increases the signal-to-noise ratio (SNR) which is characteristically low in EEG data, revealing the consistent and event-related components.

The ERP waveform consists of a series of negative and positive peaks that are identified according to their time of occurrence and polarity. These signals can be characterized by: (1) amplitude (μV), which is defined as the difference between the mean pre-stimulus baseline voltage and the largest peak of the ERP waveform within a time window, (2) latency (ms), the time from stimulus onset to the point of maximum amplitude within a time window, and (3) scalp distribution, which is defined as the amplitude change over the midline electrodes (Fz, Cz, Pz) [30]. Figure 2.2 illustrates some components found in an ERP. In this situation, we are interested in the P300 component which is a positive wave that peaks around 300ms after a stimulus event. We can also refer to these components according to their order of occurrence. The P300 is the third positive wave observed after a stimulus, therefore is also often referred to as P3.

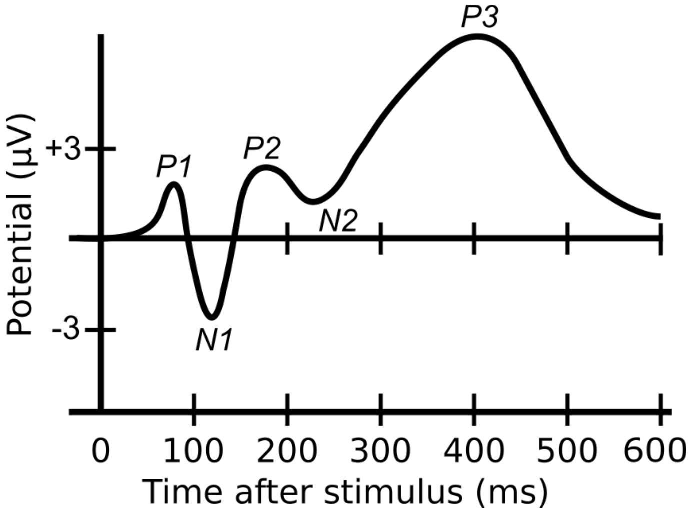


Figure 2.2: Components of an ERP wave.

The P300 signal was first reported in [39]. This ERP component is believed to represent cognitive functions related to the orientation of attention and context-updating [30]. The P300 component is composed of two subcomponents, the P3a and P3b.

P3a reflects an automatic orientation of attention to novel or salient stimuli independent of task relevance and is generated mainly in the prefrontal, frontal, and anterior temporal regions.

P3b is task-related or involves a decision to evoke this component. It has a greater centroparietal distribution due to its reliance on the posterior temporal, parietal, and posterior cingulate cortex mechanisms. Being task-related, this component can be elicited with the so-called oddball paradigm.

P300 BCIs use this oddball paradigm to trigger a P300 response in the brain. This paradigm involves presenting a set of non-target and target stimuli to the user. Non-target stimuli are more frequent than target ones, maximizing the amplitude of the P300 component. The amplitude directly correlates with the event's relevance and inversely with the probability of target stimuli [12].

In a P300 BCI, stimuli are presented to users through various methods. The main approaches involve presenting stimuli either as a sequence of images or by flashing/intensifying rows and columns within a matrix, a technique known as the Row Column Paradigm (RCP) task. However, inconsistent terminology exists within this paradigm regarding the flashing of rows and columns. For example, Figure 2.5 depicts a speller matrix where each row and column is flashed for a specific duration. Some studies refer to the flashing of all rows and columns as a *trial* [23], while others describe it as a *sequence* [31]. Similarly, the total number of matrix flashes required to select a character is called a *trial* in some literature [34], while others refer to it as a *character* [7, 31], indicating the data required to spell a character. In this work, we define the flashing of the entire matrix as a *trial* and the number of trials required to spell a character as either *trials needed to spell a character* or simply *character*.

Another key concept within this paradigm is the inter-stimulus interval (ISI), which refers to the time gap between stimuli. The ISI separates each stimulus and provides a pause before introducing the next one. The specific duration of the ISI can vary across experiments, and the chosen value significantly influences the characteristics of the resulting waves. Shorter ISIs can disrupt the elicited P300 and negatively impact classification performance. On the other hand, longer ISIs slow down the system, potentially leading to fatigue, reduced focus, and frustration with the system, all of which also pose challenges for the classification task [12].

In addition to the ISI, there is also the Stimulus Duration (SD), which specifies the time period during which the stimulus is presented to the user. By combining the SD with the ISI, we obtain the Stimulus Onset Asynchrony (SOA), which indicates the interval between the onset of one stimulus and the onset of the subsequent stimulus. Figure 2.3 illustrates these concepts.

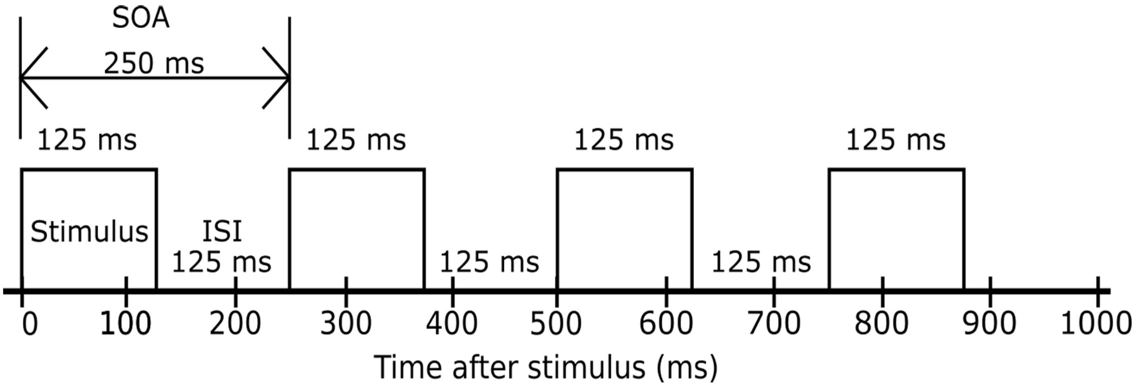


Figure 2.3: Depiction of the Stimulus Onset Asynchrony (SOA), Inter Stimulus Interval (ISI), and Stimulus Duration (SD)

P300 Detection Challenges

In addition to the inherently noisy nature of EEG data, which complicates the extraction of meaningful features, another significant challenge in developing effective BCI systems lies in the variability of ERP signals across subjects. This variability makes most BCI systems subject-dependent, meaning they are tailored to the specific signals of each individual. Figure 2.4, adapted from [40], illustrates the differences in latency and amplitude of P300 waves across nine subjects. Each curve in the figure represents the average of 820 epoch signals, where an epoch signal corresponds to a segment of the signal within a specified time window. Averaging a large number of epoch signals yields clean waveforms with a high SNR. This technique of averaging multiple epoch signals is referred to as generating an evoked signal.

In contrast, a typical process for spelling a character in a classical 6×6 matrix BCI speller requires 15 trials, yielding just 15 signals for each row and column. Averaging these 15 signals would result in a much noisier signal compared to those shown in Figure 2.4. The considerable variability still evident in the figure, despite the high SNR, highlights the immense challenge of accurately classifying these EEG signals.

Adding to the complexity, the characteristics of the wave within the same subject can also fluctuate from one session to another, making this intra-subject variability another challenge that must be addressed. Consequently, most P300 BCI systems depend on a calibration phase at the onset of each session. During this phase, ERP data is utilized to train a specific classifier, ensuring optimal performance tailored to the individual's current session.

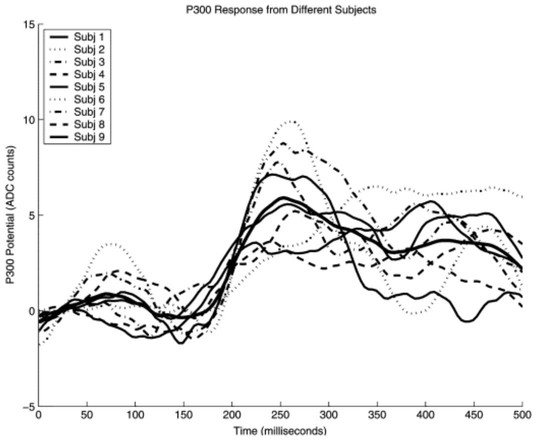


Figure 2.4: P300 potential from nine subjects measured at Cz with the reference set to the average of P7 and P8 potentials. Figure taken from [40] (Fig.3).

These challenges hinder the design of a practical real-world system. Notably, the calibration time holds significant importance; for instance, a prolonged 20-minute calibration phase is not desirable if we have to do it for every session. Furthermore, due to the noisy nature of EEG data, most spellers require multiple trials to accurately recognize a character. This dependence on multiple trials slows down the system, further compromising its practicality.

2.3 P300 BCI Pipeline

To interpret EEG signals, systems typically progress through several stages. Most BCIs follow these key steps to decode the signals.

2.3.1 Preprocessing

This initial stage of the BCI pipeline focuses on cleaning the signal to remove artifacts, enhance the SNR, and prepare the data either for feature extraction or as direct inputs for the classification stage. The following techniques are commonly applied across most BCI paradigms.

Filtering

Filtering EEG data is a critical step in the preprocessing stage because it helps remove unwanted noise and artifacts, improving the quality of the signal for analysis. EEG recordings often capture not only brain activity but also interference from external sources, such as electrical noise (e.g., 50/60 Hz powerline noise), muscle movements, and eye blinks. By applying appropriate filters, such as high-pass, low-pass, or notch filters, specific frequency ranges associated with these artifacts can be attenuated, preserving the relevant brainwave information. Without proper filtering, noise may obscure important signal patterns, leading to inaccurate interpretations or degraded performance of machine learning models. Thus, filtering ensures that the EEG data is clean and focused on the neural activity of interest, enhancing the overall quality of downstream analysis. Most studies in the P300 Speller paradigm employ a band-pass filter with a low cut-off frequency of 0.1 Hz and a high cut-off frequency that varies, typically between 10 and 40Hz.

Artifact Removal

Artifact removal is a crucial step in the preprocessing of EEG data, especially in P300 ERP analysis, as it ensures the extraction of clean and reliable signals related to brain activity. P300 signals are often contaminated by various artifacts such as muscle movements (EMG), eye blinks, and eye movements (EOG), which can significantly distort the signal. Techniques such as Independent Component Analysis (ICA) are commonly used to isolate and remove these artifacts by separating the EEG signal into independent components and identifying non-neural sources. Another popular method is regression-based correction, which models the influence of artifacts like EOG and subtracts them from the EEG data. Additionally, spatial filtering techniques like Common Average Referencing (CAR) can help reduce noise and enhance signal clarity by focusing on brain activity rather than external interference.

Downsampling

Downsampling involves reducing the sampling rate of the recorded signal to make data analysis more efficient without losing critical information. This process is often necessary because EEG systems typically record data at high sampling rates, which can lead to large datasets that are computationally demanding to process. However, downsampling must be done carefully to avoid aliasing, a phenomenon where higher frequencies are misrepresented as lower ones due to insufficient sampling. According to the Nyquist-Shannon theorem [36], to accurately capture the frequency content of a signal, the sampling rate must be at least twice the bandwidth of the signal present in the data. For EEG data, where brain activity of interest often falls below 50 Hz, a typical downsampling rate might reduce the sampling frequency to around 128 Hz, ensuring it is still well above the Nyquist frequency for the EEG's relevant frequency range. Before downsampling, a low-pass filter is typically applied to remove any frequencies above the Nyquist frequency to prevent aliasing, thereby preserving the integrity of the EEG signal.

Normalization

Normalization of EEG signals is a critical preprocessing step in P300 ERP detection, especially when using traditional machine learning models to classify the signals. EEG data can vary widely between subjects, sessions, hardware, and even different electrodes due to individual physiological differences, varying scalp-electrode impedance, and environmental noise. Without normalization, these variations can obscure the true patterns associated with the P300 component, making it harder for machine learning models to learn meaningful representations. Normalization techniques, such as z-score normalization or min-max scaling, standardize the data so that it falls within a consistent range or distribution. This ensures that the amplitude differences between signals do not bias the model, allowing it to focus on temporal or spectral features that are more relevant for detecting the P300 response. Properly normalized data improves the generalization ability of machine learning models by reducing variability unrelated to the underlying neural processes, which is particularly important in applications such as BCIs where the models must perform well across different users and conditions.

2.3.2 Feature Extraction

Feature extraction follows signal preprocessing and involves deriving various features based on the specific paradigm and characteristics of the signal. The primary objective of feature extraction is to reduce dimensionality, thereby enabling efficient use of machine learning models for classification. The extracted features should accurately represent the underlying signal [44].

When working with EEG signals, it is crucial to consider their inherent properties that complicate feature extraction and analysis. These signals are non-stationary, non-linear, and non-Gaussian, and these characteristics should be accounted for at most stages of a BCI pipeline [44].

EEG signal features can be extracted across various domains:

Time-Domain Features

Across the time domain various methods can be applied to extract meaningful features, one of those methods is Autoregressive Modeling (AR) which are fitted to the EEG signal and the AR coefficients are extracted and used to construct a feature vector of that signal [42].

Statistical methods can also be used in feature extraction. Metrics such as mean, standard deviation, kurtosis, and skewness among others, can provide insights into the distribution and behaviour of EEG signals. These statistical features can reveal significant patterns helping in the classification task.

Spatial-Domain Features

One of the most popular techniques within spatial domain analysis is Common Spatial Patterns (CSP) [28]. CSP is a spatial filtering method that transforms EEG signals into a space where the variance of one group is maximized while the variance of the remaining group is minimized.

This technique, however, is subject to an optimal frequency band that varies between individuals. Several adaptations of CSP have been developed to address the limitations of focusing only on spatial features. Among these is the Common Spatio-Spectral Patterns (CSSP) [24] technique, which applies a single band-pass filter to the signal before applying CSP, capturing both spatial and spectral features within a specific frequency band. Another variation, Filter Bank Common Spatial Patterns (FBCSP) [1], enhances this process by using multiple band-pass filters, dividing the signal into various sub-bands, and applying CSP independently to each. These techniques enable more comprehensive feature extraction by combining both spatial and frequency information for better classification.

Although CSP and its variants are most used in motor imagery tasks, they have also been successfully adapted for the P300 paradigm. Notably, several of the most effective neural network architectures used in P300 classification have drawn inspiration from these spatial filtering techniques [21, 34].

Frequency-Domain Features

On the frequency domain, the techniques focus on features that can be extracted from the underlying sinusoids that make up the signals. Some of the most used methods in this domain include the Fourier Transform used to decompose a time-domain signal into its constituent frequencies, and the Power Spectral Density (PSD) which estimates the power (energy) of each frequency component in that signal.

Time-Frequency-Domain Features

Features can also be extracted across multiple domains at once, as an example we have the Short-time Fourier Transform (STFT) that extracts features across both the time and frequency domain.

Many studies within the P300 speller paradigm use preprocessed EEG signals as direct inputs to their classifiers. As previously mentioned, EEG signals are recorded from various brain regions using multiple electrodes with high temporal resolution. Some studies organize the data into matrices with dimensions $N_E \times N_T$, where E represents the number of electrodes, and T denotes the number of time samples per signal. Other studies concatenate these electrode vectors into a single large vector of size $R^{E \times T}$. From this point forward, both formats will be referred to as *epoch features*. By averaging multiple epoch features, we obtain an *evoked feature*.

2.3.3 Classification

After feature extraction, the resulting features are used as inputs for machine learning models. Choosing the appropriate machine learning models requires considering several factors, including the nature of the problem, the characteristics of the signals, the availability of labeled data, and the computational requirements, among others [44].

2.3.4 Feedback

Once the model is trained, it is ready for classifying incoming data. This classification needs to be communicated to the user, making mandatory the incorporation of an effective feedback mechanism in most BCIs. This mechanism ensures effective communication between the user and the system, completing the BCI pipeline.

2.4 Evaluation of P300 Speller Performance

The evaluation of a P300 BCI Speller System is typically conducted in two distinct ways. First, we can use available datasets to assess system performance in what is known as an offline study. Alternatively, we can design real-world experiments to evaluate the system's performance with users in real-time, referred to as an online study. Ideally, both types of studies should be employed to thoroughly evaluate a BCI system. However, online studies are time and resource-intensive, leading most researchers to only evaluate their proposed system with offline studies.

Within these two types of studies, the performance of a P300 speller system can be evaluated in different ways:

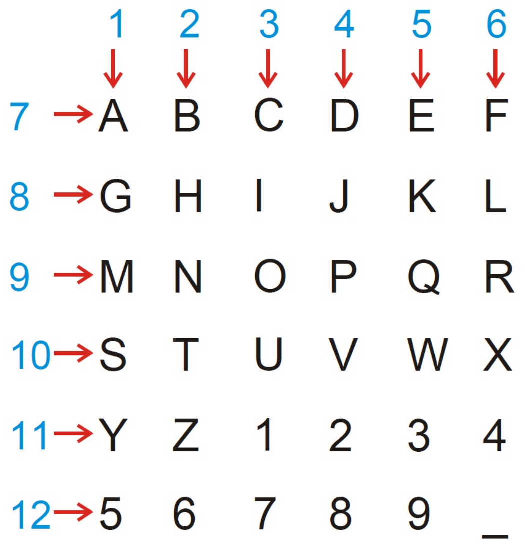
Binary Classification

Binary classification in the context of detecting the P300 signal involves distinguishing between two classes: target and non-target signals. Typically, this form of classification measures a model's ability to correctly identify single trial signals (signal epoch). However, as previously discussed, this task is challenging. Besides the noisy nature of the data and the high variability between and within subjects, there is also no guarantee that users will produce a P300 signal at the correct moment due to potential distractions, making ground truths questionable [7]. Despite these challenges, this type of performance evaluation is still valuable. It can provide insights into the system's performance, with various studies finding correlations between some metrics in this evaluation and character recognition accuracy [26, 7, 16].

When performing binary classification evaluation, it is important to consider the natural data imbalance originating from the oddball paradigm. For instance, in a 6x6 character matrix, each trial will have one target signal for every five non-target signals (1:5 class imbalance). This means that a classifier that always outputs non-target would achieve an accuracy of around 83%. Therefore, it is crucial not only to carefully choose the metrics used to evaluate classifier performance but also to specify the True Negatives (TN), True Positives (TP), False Negatives (FN), and False Positives (FP).

Character Recognition Accuracy

While the previous classification method was a binary problem, this one involves multi-class classification. The goal is to accurately predict the desired character to spell using the outputs from the binary classification. As we can observe in Figure 2.5 each character in the matrix can be identified by a pair (row, column), so the task is to correctly identify the target signal among the rows and the target signal among the columns by accumulating predictions or probabilities over sequential trials for each row/column.



	1	2	3	4	5	6
7	A	B	C	D	E	F
8	G	H	I	J	K	L
9	M	N	O	P	Q	R
10	S	T	U	V	W	X
11	Y	Z	1	2	3	4
12	5	6	7	8	9	_

Figure 2.5: Example of Rows and Columns in a 6x6 Speller Matrix

In this context, accuracy is an appropriate metric since we are comparing the target string to the predicted string. A random classifier would achieve only about 2.8% accuracy in this type of evaluation for a 6x6 speller matrix (36 characters total).

Information Transfer Rate

The Information Transfer Rate (ITR) is a crucial metric for evaluating the performance of a BCI system, quantifying how efficiently the system communicates information. It is measured in bits per minute and is calculated using the following equation:

$$ITR = \left(log_2(N) + P log_2(P) + (1-P) log_2 \frac{(1-P)}{(N-1)} \right) \times \frac{60}{T} \tag{2.1}$$

where N is the number of possible targets (characters in the speller), P is the probability of recognizing a character (Character Recognition Accuracy), and T is the time in seconds required to make a selection.

Both N and T are determined by the experimental protocol, resulting in varying values for these parameters across each dataset discussed later in this work.

2.5 Contrastive Learning

Until recently, supervised learning was the dominant and most successful paradigm for training machine learning models. The popularization of deep neural networks and their capacity for scaling introduced the need for increasingly large amounts of annotated data. However, this reliance on labeled data comes with significant drawbacks, including the time-consuming and costly nature of data annotation, as well as privacy and security concerns[22]. Consequently, there has been a growing demand for unsupervised learning techniques that can leverage the abundant and readily available unlabeled data.

The process of learning features from data is called Representation Learning and involves learning a parametric mapping from input data to a feature vector that captures essential concepts within the data. Approaches to learning valuable representations are typically categorized into generative and discriminative modeling. Generative methods, such as those used in Generative Adversarial Networks (GANs) and Autoencoders, operate under the premise that a model capable of generating realistic data samples must also capture the underlying structure of the data. In contrast, discriminative approaches focus on the decision boundaries that separate different classes without modeling the data generation process. Examples of discriminative models include Support Vector Machines (SVMs), Decision Trees, and Neural Networks.

Historically, most unsupervised representation learning methods fell under the category of generative models [22]. However, recent advancements have introduced successful Self-Supervised Learning (SSL) techniques that adopt discriminative approaches. SSL enables models to learn from unlabeled data. In this framework, the models learn meaningful features by solving pretext tasks, i.e. pre-designed tasks that the network needs to solve during the training process. These pretext tasks generate pseudolabels that replace the supervised learning labels. Examples of pretext tasks include (1) colorization, where images are converted to grayscale, and the model learns to restore original colors; (2) jigsaw puzzles, where images are divided into patches that are scrambled, with the goal of rearranging them correctly; and (3) contrastive predictive coding, which involves predicting future frames or feature sequences based on past observations [15].

Instance discrimination is another of these pretext tasks, and where contrastive learning falls. The objective is to create augmented versions, called views, of the same data sample. These multiple views of the same data point are termed the positive pairs, while other views of different data points are referred to as negative pairs. The goal of this pretext-task is then to maximize the similarity within the positive pairs while minimizing the similarity to the negative pairs.

In this paradigm, the choice of augmentations is critical. The aim is to generate sufficiently diverse views of the same sample to enable the model to learn robust representations, while still preserving essential information for downstream tasks [41].

Contrastive learning can thus be seen as learning by comparing different samples. This is now a commonly used approach to pre-train Deep Neural Networks in a self-supervised manner. This pre-training allows the model to learn valuable features and be fine-tuned later for a specific downstream task using less labeled data. This method of pre-training the model on unlabeled data and then fine-tuning it on a smaller set of labeled data is called Semi-Supervised Learning. In this framework, the idea is to leverage a large unlabeled dataset to improve performance and decrease the dependency on labeled data. This is crucial in cases where labeling data is very costly.

Contrastive learning methods used in this study are structured similarly to the process illus-

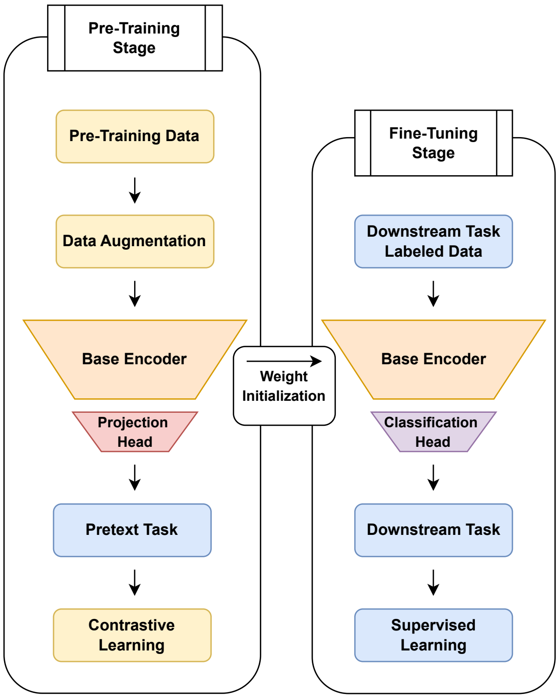


Figure 2.6: Contrastive learning general pipeline

trated in Figure 2.6. During pre-training, the model takes multiple views of the same sample as input, extracts the features and a projection head projects these feature vectors into a N-dimensional feature space. A loss function, such as NTXent Loss [8] or SupCon Loss [17], is then applied to these embeddings to measure the similarity between positive and negative pairs. A lower loss value means that positive pairs are closer together in the feature space and far apart from negative pairs. Minimizing this loss encourages the model to learn representations that are invariant to the variations introduced by the augmentations.

After pre-training, the projection head is removed, and the weights of the base encoder network are used to initialize a new model with a classification head attached to it. This model can now be fine-tuned for a specific downstream task.

Although contrastive learning methods primarily utilize unlabeled data, they face a significant drawback. As highlighted in [17], samples belonging to the same class as the positive pairs can sometimes be treated as negative pairs, causing them to be embedded farther away from the positive pairs. Supervised Contrastive Learning (SupCon) addresses this issue by leveraging the information in labeled data. Here, the positive pairs are all views from the samples in a mini-batch that belong to the same class, while negative pairs are the remaining views of the mini-batch, i.e. all samples belonging to different classes.

In the present study, we examine the Supervised Contrastive Learning strategy (SupCon) alongside the SimCLR self-supervised technique. A more detailed explanation of both techniques is provided in Section 4.3.

2.6 Summary

In this chapter, we established the foundational knowledge needed to understand both the P300 BCI speller paradigm and the contrastive learning training method. We began by discussing the key advantages and challenges of EEG data, notably its high temporal resolution (advantage) and low signal-to-noise ratio (disadvantage). We then explained that the P300 is an event-related potential (ERP) component that occurs around 300 ms after stimulus onset. Following that, we outlined the common BCI pipeline, which consists of four stages: preprocessing, feature extraction, classification, and feedback.

Next, we reviewed various strategies for evaluating these systems. One approach is binary classification, where each signal is classified as either P300 or Non-P300, requiring careful attention to the imbalanced nature of the oddball paradigm when selecting evaluation metrics. Another method is Character Recognition Accuracy, which assesses how accurately the model predicts the intended character. Lastly, we discussed the Information Transfer Rate, which measures how efficiently the system communicates information.

Finally, we introduced the contrastive learning technique, which uses a pretext task to encourage the model to learn meaningful features from augmented data.

Chapter 3

Related Work

In this chapter, we review and discuss key studies related to the P300 Speller paradigm. As previously mentioned, these spellers can be classified into two main categories: Subject-Dependent Systems, which require user-specific data and involve a lengthy calibration phase, and systems that are less reliant on such data or do not require it, resulting in a shorter or nonexistent calibration phase, termed Subject-Independent.

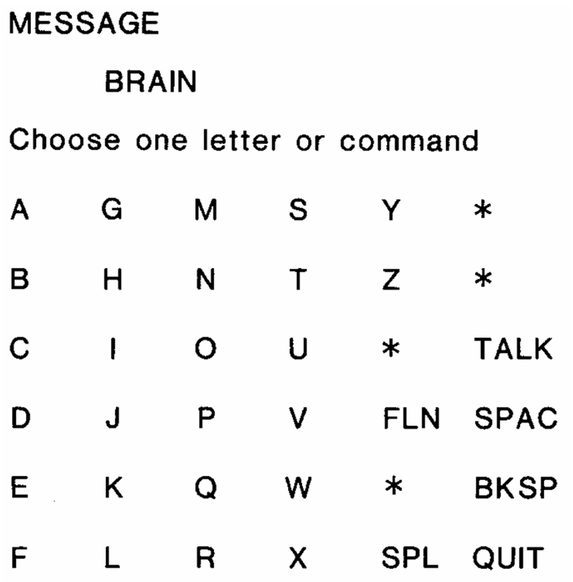
In this chapter, we refer to Character Recognition Accuracy simply as ”accuracy”, distinguishing it from binary classification accuracy.

3.1 Subject Dependent P300 Recognition

The pioneering work in P300 BCI systems was introduced in 1988 by L.A. Farwell and E. Donchin [12]. Their system featured a 6x6 matrix, displayed in Figure 3.1, with letters of the alphabet and some 1-word commands. Participants focused on the letter they wanted to spell while the matrix columns and rows were briefly flashed, and the brain signals were recorded for each intensification. Four classification methods were tested: (a) Stepwise Discriminant Analysis (SWDA), (b) Peak Picking, (c) Area, and (d) Covariance. SWDA and Peak Picking demonstrated the best performance for most individuals and inter-stimulus intervals (ISIs).

For an ISI of 500ms, the algorithms required an average of 26 seconds to reach 95% accuracy. This resulted in a speed of 0.20 bits/sec or 12 bits/min. Assuming 1 character can be typed every 26 seconds with reliability, the spelling rate was 2.3 characters/min. While considerably slower than the average typing speed of 150 characters/min, this method provided a means of communication for individuals with limited motor function.

In the work by Rivet et al. [33], an unsupervised algorithm named xDAWN was introduced for enhancing P300 signals through the application of spatial filters. Essentially, xDAWN projects raw EEG data into an estimated signal subspace. The paper employs a Bayesian Linear Discriminant Analysis (BLDA) classifier with the xDAWN algorithm. The system achieved a 90% accuracy with 10 trials per character and 80% with 5 trials. While this work uses a BLDA, other studies, that we analyzed, use xDAWN in conjunction with various classifiers [14, 43, 21], either propos-



MESSAGE

BRAIN

Choose one letter or command

A	G	M	S	Y	*
B	H	N	T	Z	*
C	I	O	U	*	TALK
D	J	P	V	FLN	SPAC
E	K	Q	W	*	BKSP
F	L	R	X	SPL	QUIT

Figure 3.1: The 6x6 matrix utilized by Farwell and Donchin [12]

ing it as a solution or comparing it to other proposed approaches.

Other popular machine learning models used to classify P300 signals are Support Vector Machines (SVMs). They stand out as one of the most frequently employed algorithms for classifying EEG data, including datasets derived from oddball paradigm tasks. SVMs are particularly effective in pattern recognition, especially when dealing with high-dimensional problems.

Rakotomamonjy et al. [31] described the successful strategy that achieved first place in the 2005 BCI Competition III. They employed an ensemble system composed of 17 SVMs to classify data from two subjects. The features that served as inputs to these SVMs were epoch feature vectors. Each subject's training dataset was divided into 17 partitions, each of which was used to train an individual SVM. The classification of incoming data involved obtaining the outputs from all SVMs and summing these outputs. With 15 trials per character, this approach achieved 96.5% accuracy.

Cecotti et al. presented the first work to employ Convolutional Neural Networks to the task of classifying P300 signals [7]. The authors argued that CNN models are a good approach due to the high variability among P300 signals. Their ability to capture high-level features can also provide a deeper analysis of brain activity by analyzing the receptive fields of the network. Another advantage, already mentioned earlier and highlighted by the authors, is that they do not need advanced or very robust preprocessing techniques to perform well. In their work, the authors compared single CNN classifiers and multi-classifier CNNs obtaining 70%, and 94.5% accuracy for the best single CNN classifier (CNN-1) for 5 and 10 trials, respectively, while their best CNN multi-classifier (MCNN-1) achieved 69% and 95.5% accuracy with 5 and 10 trials, respectively. The authors also conducted a binary classification and observed a strong correlation between Recall (obtained in the binary classification study) and character recognition accuracy. The study reports Recall values of 0.6760 for CNN-1 and 0.7122 for MCNN-1.

Building on the previous work ideas, Liu et al. [26] proposed enhancing a CNN's generalization capability by incorporating Batch Normalization and Dropout, resulting in a model called Batch Normalized Neural Network (BN^3). This model achieved 74.5% and 95.5% accuracy using 5 and 10 trials, respectively, in the BCI Competition III Dataset II, yielding results comparable to state-of-the-art models. Notably, under conditions with fewer trials, the proposed method outperformed other models, achieving 34.5% accuracy with just one trial. This suggests that Batch Normalization plays a crucial role in feature extraction, increasing the SNR in single trials.

The authors also conducted a binary classification study. Contrary to the previous work [7], they found that the F1-score was the metric best correlated with character recognition accuracy, achieving a 0.4919 F1-score.

Attention mechanisms have also started to emerge in the BCI domain. In [37] the EEG Conformer was presented, which integrates convolutional and transformer modules to decode EEG signals effectively, capturing both local and global temporal dependencies. The model consists of three main components: a convolution module for extracting local temporal and spatial features, a self-attention module to capture global dependencies in the EEG data, and a classifier module for predicting output categories. The convolution module first applies temporal and spatial convolutions separately, using average pooling to reduce noise and improve feature robustness, while transforming data points into tokens. These tokens enter the self-attention module, where multi-head attention layers learn long-term dependencies across the temporal sequence. This integration of CNNs and Transformers enables the model to perform end-to-end decoding without extensive preprocessing, achieving state-of-the-art accuracy in Motor Imagery and Emotion Recognition in a subject-dependent framework. Although this network was not originally evaluated for P300 classification, its design and structure can be adapted with minimal modifications for this task.

3.2 Subject Independent P300 Recognition

As already noted, LDAs, SVMs, and Neural Networks are some of the most commonly used and best-performing machine learning algorithms for the P300 classification task. However, Riemannian Classifiers [4] also rank among the top-performing algorithms, with numerous studies focused on exploring and developing new systems based on this approach. A Riemannian Classifier operates on covariance matrices. The Minimum Distance to the Mean (MDM) algorithm classifies a new EEG signal by computing its distance to the Riemannian mean of each class. This is achieved by first calculating covariance matrices for a set of labeled EEG signals and determining the Riemannian mean for each class. The new signal is then assigned to the class whose mean is closest.

One system based on this technique was developed by Barachant and Congedo [5], who introduced a plug-and-play device, i.e a device with no need for a calibration phase. This work demonstrated a system with improving performance, a reduction in the number of labeled data needed for calibration, and good generalization across subjects and sessions. An adaptive implementation was also developed, combining specific matrices with generic ones for a real-life game. This approach enabled participants to play without a calibration phase. At the start of the session, players needed an average of 2.5 trials to destroy a target, which decreased to 1.5 trials by the final session due to the system's adaptation to each user.

Another study exploring this technique was presented in [6]. The authors used the Riemannian Gaussian Distribution to model each class. A Bayesian Accumulation method is used in conjunction with this approach. This method prevents the loss of information because all characters update their confidence or probability even when they are not being intensified. Plus, this system can be stopped dynamically, i.e, when a certain confidence threshold is reached there is no need to continue to further trials, speeding up the BCI system performance.

To evaluate their solution, the authors used the ALS Dataset and the Overt or Covert Dataset. They named their proposed system ASAP (Bayesian accumulation of Riemannian probabilities) and compared it against an MDRM without Bayesian Accumulation, a regularized LDA with xDAWN, and a regularized LDA with simple spatial-temporal features.

The results demonstrated the superior performance of the proposed method, achieving an accuracy of approximately 60% with 5 trials and around 80% with 10 trials in the ALS Dataset. In the Overt (or Covert) Dataset, the method achieved 91% accuracy with 4 trials and 95% accuracy with 8 trials.

Shifting away from Riemannian classifiers but keeping the focus on improving cross-subject P300 classification performance, He et al. [14] proposed a transfer learning alignment technique on the Euclidean Space. This method offers three key advantages: (1) it aligns signals rather than covariance matrices, enabling the application of various signal processing, feature extraction, and machine learning algorithms to these aligned signals; (2) it has a low computational cost; and (3) it operates in an unsupervised manner, requiring no label information. The essence of Euclidean Alignment (EA) lies in rendering the data distributions of different subjects more similar, allowing a classifier trained on this auxiliary data to potentially exhibit improved performance on new subjects. The study compared SVMs and xDAWN-SVMs with and without EA in an offline setting, showing improved performance with EA implementation. The SVM model achieved an accuracy of 67.85% with EA and 64.64% without it, while xDAWN+SVM reached 68.80% with EA versus 64.60% without it.

We now return to Neural Networks to analyze the EEGNet model [21]. This architecture was introduced as a lightweight compact CNN. The primary goal of this model is to enable training the model with limited data while generating interpretable neurophysiological features. A comparative analysis was conducted with other deep learning models, including DeepConvNet and ShallowConvNet both described in [35], as well as to a Riemannian Classifier + xDAWN [5, 33]. In both within-subject and cross-subject evaluations, the Deep Learning methods exhibited superior performance, with no statistically significant differences between them. Notably, they achieved Area Under the Curve (AUC) values exceeding 0.9 for within-subject classification and approaching 0.9 for cross-subject classification. Among these, EEGNet stood out for its significantly reduced number of trainable parameters (an order of magnitude fewer than other CNN models) exhibiting enhanced effectiveness with limited training data.

Based on the EEGNet architecture, [23] introduced a live-training system featuring a Step-Wise Linear Discriminant Analysis (SWLDA) and a zero-train CNN. The zero-training model underwent calibration using an extensive dataset (99 000 samples) and exhibited comparable performance in offline scenarios compared to a CNN trained with the subject's data and a SWLDA also subjected to training. Specifically, the zero-training CNN achieved an accuracy of 89.22%, in contrast to the 94.64% achieved by the trained CNN and the 94.29% by SWLDA. In the online study involving 12 subjects, the zero-training CNN achieved 85% accuracy. Both offline and online study results used 15 trials to spell each character.

The live-training system integrates an adaptive SWLDA and the zero-training CNN. Initially, the system's output corresponds to the CNN output. These outputs also serve as labels for training the SWLDA. Once the system accumulates sufficient data for the SWLDA model to perform even better than the generic CNN, the system switches classifiers and the SWLDA starts classifying the incoming data.

By integrating these two models, the necessity for a calibration phase is eliminated. This is achieved by having a generic classifier that demonstrates robust performance in classifying initial data. Simultaneously, the inclusion of an adaptive classifier allows for an enhancement in performance as it is trained on the specific data of each user. This system needed 15 trials per character, i.e. 33 seconds to spell a character.

Santamaría-Vázquez et al. [34] introduced another state-of-the-art CNN model, EEG-Inception. The model's Inception modules enable multi-temporal scale analysis of the input data by using parallel convolution layers with different kernel sizes. Besides this new architecture, another key contribution of this study is the large dataset used for training and evaluation, which was made publicly available to other researchers. This dataset which includes 701,615 signal epochs collected from 73 subjects (42 healthy, and 31 motor disabled) was partitioned into training, validation, and test sets, with the training and validation sets comprising data from healthy subjects, while the test set contained data from motor-disabled subjects. After training, the model was evaluated with varying levels of fine-tuning data and compared to (1) regularized LDA, (2) the Riemannian Classifier + xDAWN [5], (3) a CNN with a Bidirectional LSTM (BLSTM), (4) DeepConvNet [35], and (5) EEGNet [21]. Results in Character Recognition Accuracy demonstrated significant improvements from EEG-Inception compared to all other models, achieving an accuracy of 84.6%

± 13.2% with the maximum amount of fine-tuning data.

The widely acclaimed Residual Network, which revolutionized image classification, has also been introduced into the P300 classification paradigm. Pereira et al. [29] employed Wide Residual Networks for cross-subject and cross-session P300 classification. Their system utilizes minimal preprocessing to create 2D matrices from incoming signals and incorporates regularization through a cutout technique, which randomly sets a square portion of the 2D matrix to zero. This proposed solution was evaluated on a dataset collected from 66 subjects performing a Virtual Reality task using the oddball paradigm. The method achieved state-of-the-art performance, yielding an F1-score of 0.4585. Additionally, the authors empirically demonstrated that a Residual Network outperforms a CNN with the same architecture.

3.3 Contrastive learning applied to EEG Data

Contrastive learning has been receiving attention due to its success in the computer vision and natural language processing domains. However, its research has been expanded to other domains such as time series analysis, in which EEG data can be included. As already mentioned, contrastive learning is leveraged in the pre-training stage usually tackling a general problem where an abundance of data (normally unlabeled data) can be found. After that, models are fine-tuned to a particular task related to the pre-trained general task. Within the EEG analysis or BCI paradigm, the data has quite a lot of variability. EEG signals vary from user to user, and from session to session. There is also the variability introduced by the different hardware used to collect the signals, the multiple tasks and paradigms within the BCI research domain, among others. Therefore, there are multiple ways of defining the general task from which to collect the data for the pre-training stage. Some authors might collect data from multiple users, across multiple sessions and from varying paradigms such as Motor Imagery, Emotion Recognition, Sleep Stage Classification, ERP Classification, combining all of these data into a large dataset.

This approach was taken in [19] where the authors collected large amounts of unlabeled EEG signals from multiple subjects and sessions and across different paradigms. This massive dataset was used to train the BENDR model (BErt-inspired Neural Data Representations) which consisted of a convolutional encoder to process raw EEG data into compact representations called BENDR features, and a transformer encoder that models these embeddings capturing contextual relationships. Drawing inspiration from the *wav2vec2.0* approach presented in [2] for speech recognition, they pre-trained this model with contrastive self-supervision by masking 10-vector segments in EEG embeddings and learning to correctly match these to the original representations in the sequence. This approach enables the model to develop powerful general features from unlabeled data, which can then be fine-tuned for specific tasks with smaller labeled datasets. Interestingly, the worst-performing downstream task in this work was P300 classification. The authors attributed this result to the shorter sequence length that composes the P300 feature vector, finding that performance increases for larger EEG signal epochs.

An alternative approach for defining a general pre-training task within the BCI paradigm is to treat EEG signals from different users within the same BCI task (e.g. Motor Imagery, or P300 Classification) as distinct but related domains. Given the high variability between subjects, this framework views the classification of EEG data from a specific subject as the downstream task, while using data from multiple other subjects as a broader, generalized task suitable for pre-training. Knowledge acquired in this general pre-training stage can then be transferred to enhance performance on the downstream task.

This approach was implemented in [25], where the authors aimed at enhancing cross-subject classification in the Motor Imagery (MI) paradigm. The proposed model architecture includes a CNN encoder paired with an attention mechanism to extract robust features while focusing on relevant EEG channels. The network is pre-trained with a self-supervised contrastive learning technique similar to SimCLR [8], where τ is set to 0.5. The pre-trained model is then fine-tuned using supervised learning. Here, data from a specific user is treated as the target domain (downstream task), whereas data from other subjects serves as the source domain for pre-training. This study introduced various data augmentation techniques for the unsupervised contrastive pre-training stage, including amplitude addition, amplitude scaling, time warping, cutout and resizing, cutout and zeroing, crop and resizing, horizontal flipping, and permutation. Experimental results on public MI datasets demonstrate that this approach outperforms traditional and other deep learning models in cross-subject MI classification, with improved robustness attributed to contrastive learning and these data augmentation strategies.

Another approach to deal with the cross-subject variance, this time within the P300 classification paradigm, was explored by Cui et al. [11]. They proposed training a model composed of three layers of inception modules using contrastive learning. Their contrastive learning strategy is based on the data sampler that generates mini-batches with data from different users. Instead of using the signal epochs as inputs, they average three of these signal epochs obtaining an evoked signal with slightly improved SNR as the training samples. To learn subject invariant features the positive pairs are the target signals (P300 signals) of different subjects. This way the base encoder is forced to align the target signals of different subjects closer together. The following steps are similar to SimCLR [8]. This method was tested using the dataset made available by [34]. Data from 7 randomly selected subjects was used as the test set while the rest was used as the training set. Their model was trained with this method for 100 epochs. Results showed a significant outperformance of this method over EEG-Inception [34], EEGNet [21], LDA, and xDAWN + Riemannian Classifier [5, 33], achieving 0.7233 AUC in the Binary Classification Evaluation, an improvement of 3.04% and 5.88% over EEG-Inception and EEGNet, respectively. The authors do not specify the value of τ used in the pre-training stage.

As we have seen, several studies have explored the potential of contrastive learning for EEG classification tasks. However, to our knowledge, [11] is the only work that specifically applies a contrastive learning approach to address directly the cross-subject variance problem within the

P300 classification paradigm.

3.4 BCI Competition III Dataset II Studies

During the literature review process, we encountered several studies that used the BCI Competition III Dataset II for training and evaluating their proposed solutions. Displaying results obtained from evaluation on the same dataset allows a fairer comparison between studies. Below, we provide a brief overview of studies that used this dataset. All studies described below fall into the category of subject-dependent systems.

Joshi et al. [16] employed a CNN and a Convolutional Long Short Term Memory (ConvLSTM) model, both using 3D epoch features. The evaluation was conducted on a subject-dependent, single-trial basis, and included individual model assessments as well as combinations into ensembles. The results showed that the Ensemble BN3 + ConvLSTM + CNN3D achieved the best performance, with F1-scores of 0.41 and 0.56 for Subject A and Subject B, respectively.

In [18] the authors of the study used two transfer learning techniques aimed at reducing the required training samples while maintaining high accuracy. The techniques used were fine-tuning and data alignment. Fine-tuning involves taking a pre-trained model and training it with target data (in this case, subject-specific data). The data alignment technique used was the Euclidean Alignment (EA) described in [14].

The training process used all the training data from one subject (source data) and only 30 characters of the training data from the target subject. Evaluation was conducted using all testing data from the target subject. The proposed method outperformed the approaches that did not use either transfer learning and data alignment, achieving 97.5% accuracy with 15 trials, demonstrating its effectiveness in reducing the amount of data needed for training. However, since 30 characters from the subject were still used for training, the system would require a calibration phase of over 15 minutes.

The authors of [20] utilized Sparse Autoencoders (SAE) and Stacked Sparse Autoencoders (SSAE) for feature extraction. These unsupervised feature learning methods are used to extract what they termed ”deep features” from epoch features. These deep features are then concatenated with the original epoch features to serve as input for an SVM Ensemble (ESVM).

The proposed method was evaluated using both the BCI Competition III Dataset II and the ALS Dataset and was compared to several state-of-the-art systems. Both SAE-ESVM and SSAE-ESVM demonstrated superior performance across 5, 10, and 15 trials. For the ALS Dataset, Binary Accuracy (BA) was assessed using five-fold cross-validation. Results indicated that these two models (each achieving 95.5% BA) outperformed a temporal linear classifier and an SWLDA model, which achieved 89.5% and 87.5% BA, respectively.

Cherloo et al. [10] noted that the overlapping of epochs deteriorates the discrimination performance between P300 and non-P300 signals. To solve this problem the authors propose transforming this binary-classification (P300 vs Non-P300) to a multi-class classification task where the different non-P300 signals are grouped together based on their relative position to the previous P300 signal. Employing this strategy the authors report a 74%, and 98% accuracy on the BCI Competition III Dataset II with 5 and 15 trials, respectively. Using the data from the ALS Dataset the authors reported a 78%, and 98% accuracy, with 5 and 10 trials. A potential drawback of this methodology is that the number of groups into which to divide the non-P300 signals must be chosen carefully not only for each subject but possibly for each session, adding even more subject dependence to this already subject dependent study.

3.5 Summary

This literature review highlighted the diverse methods employed in the P300 paradigm, ranging from traditional classifiers like SVM and LDA to more recent advancements such as Deep Learning and Riemannian classifiers. Table 3.1 provides a summary of the key details regarding the methods adopted by the studies discussed in this chapter, offering a comparative overview of the techniques employed across various works within the P300 paradigm.

As we can observe, many works rely on straightforward epoch features when it comes to feature extraction, often formatted as matrices (typically used with Neural Network classifiers) or as vectors (more common in traditional classification methods like SVMs). The ”Results” column presents the best results achieved by the top-performing classifier under optimal experimental conditions. However, direct comparisons between these results should be approached with caution, as the studies use different datasets, leading to significant variability in the data.

Although not mentioned in Table 3.1, most studies apply minimal preprocessing techniques. Common methods across all studies in this stage include band-pass filters, downsampling, and segmentation.

Table 3.1
Overview of Datasets, Features, Classifiers, and Results in several P300 Paradigm Studies

Papers	Dataset	Features	Classifier	Results
[12] (1988)	Online study 4 subject	Epochs Features and Covariance Matrices	(1) SWDA (2) Peak Picking (3) Area (4) Covariance	Subject Dependent 26s to reach 95% acc
[33] (2009)	Online study 3 subject	xDAWN Spatial Features	xDAWN + BLDA	Subject Dependent 80% with 5 trials 90% with 10 trials
[31] (2008)	BCI Competition III Dataset II	Epoch Features	ESVM	Subject Dependent 96.5% with 15 trials
[7] (2011)	BCI Competition III Dataset II	Epoch Features	MCNN-1	Subject Dependent 95.5% with 15 trials
[26] (2018)	BCI Competition III Dataset II	Epoch Features	BN3	Subject Dependent 96.5% with 15 trials
[5] (2014)	Three Brain Invaders P300 Datasets	Epoch Features	MDM	Subject Independent 0.82 AUC
[6] (2022)	ALS Dataset Covert Overt Datasets (10 subjects)	Covariance Matrices	ASAP	Subject Dependent ALS: 80% with 10 trials Overt: 95% with 8 trials
[14] (2020)	RSVP dataset from PhysioNet (11 subjects)	EA to align trials xDAWN Spatial Features PCA for feature extraction	EA+ xDAWN+ SVM	Subject Dependent 68.80% Balanced Accuracy
[21] (2018)	Dataset from RSVP BCI study (15 subjects)	Epoch Features	EEGNet-8,2	Subject Dependent and Independent ∼0.9 AUC in botth
[23] (2020)	Offline study Dataset (55 subjects) Online study (12 subjects)	Epoch Features	CNN-ZT (EEGNet-64,4)	Subject Independent Offline study: 89.22% with 15 trials Online study: 85% with 15 trials
[34] (2020)	GIB-UVa ERP-BCI Dataset	Epoch Features	EEG-Inception	Subject Independent (No Fine-Tuning): 61.4% with 15 trials Subject Dependent (Max Fine-Tuning): 84.6 with 15 trials
[29] (2018)	VR oddball task Dataset (66 subjects)	Padded Square Epoch Features	WideResNet-16-1	All data is mixed 0.4750 F1-Score
[11] (2024)	GIB-UVa ERP-BCI Dataset	Evoked Features for training Epoch features for testing	Inception Model	Subject Independent 0.7233 AUC
[16] (2018)	BCI Competition III Dataset II	3D Epoch Features	BN3+CNN3D+ ConvLSTM	Subject Dependent 0.5112 F1-Score
[18] (2023)	BCI Competition III Dataset II	Deep Features from CNN1 and CNN2 + EA	CNN1+CNN2+ DRBM	Subject Dependent 97.5% with 15 trials
[20] (2019)	BCI Competition III Dataset II ALS Dataset (8 Motor Disabled subjects)	Deep Features + Epoch Features	SAE + ESVM	Subject Dependent BCI Comp: 99% with 15 trials ALS: 95.5% Binary Accuracy
[10] (2023)	BCI Competition III Dataset II ALS Dataset	Epoch Features	ESVM	Subject Dependent BCI Comp: 98% with 15 trials ALS: ∼97% with 10 trials

Chapter 4

Methodology

To investigate whether contrastive learning pre-training techniques can improve network performance mainly in subject-independent scenarios or as close to it as possible, we took an approach where the general pre-training task (source domain) was data from a set of users, while the downstream task was data from different individuals (target domain). With this approach, a robust pre-training stage is essential, as it allows classifiers to learn meaningful feature representations across different subjects. In this context, we explored which pre-training techniques were most effective.

We examined three approaches to pre-train the networks: **Supervised**, **SimCLR**, and **SupCon**. These techniques were applied to three model architectures: **EEGNet**, **EEG-Inception**, and **Conformer**. The model architectures and the training approaches are described in this chapter, along with the datasets and evaluation studies.

4.1 Dataset Description

In this study, we used three different datasets collected from subjects performing a row-column paradigm (RCP) task: GIB-UVa ERP-BCI, ALS, and BCI Competition III Dataset II (referred to as BCI Comp). GIB-UVa ERP-BCI, being the largest dataset, is used for both pre-training and evaluation, while the other two are reserved solely for evaluation. A detailed description of each dataset follows.

4.1.1 GIB-UVa ERP-BCI Dataset

The GIB-UVa ERP-BCI Dataset[1], consists of a compilation of data collected by Santamaria Vazquez et al. across three different studies. Each study employed a different speller matrix configuration. This dataset, referenced in [34], served as a foundation for training and evaluating their proposed EEG-Inception Network. It includes data from 73 subjects, 31 of whom have some form of motor disability. The recordings were taken from eight electrodes (FZ, CZ, PZ, P3, P4, PO7, PO8, and OZ) at a sampling rate of 256 Hz, which was subsequently downsampled to 128

[1] `https://www.kaggle.com/datasets/esantamaria/gibuva-erpbci-dataset`

Hz. A band-pass filter with cutoff frequencies of 0.5 Hz and 45 Hz was applied, along with a common average reference (CAR) and baseline normalization from -200 to 0 ms.

4.1.2 ALS Dataset

This dataset described in [32] comprises the signals of 8 volunteers (3 female and 5 male, with a mean age of 58 ± 12) with definite, probable, or probable with laboratory support ALS diagnosis[2].

The data was recorded using 8 electrodes (FZ, CZ, PZ, P3, P4, PO7, PO8, and OZ) placed according to the 10–10 standard. All channels were referenced to the right earlobe and grounded to the left mastoid. The EEG signal was digitized at 256 Hz and band-pass filtered between 0.1 and 30 Hz.

Participants were required to spell seven predefined words of five characters each. Each character required 10 trials to be recognized. Rows and columns on the interface were randomly intensified for 125ms, with an inter-stimulus interval (ISI) of 125ms, yielding a 250 ms lag between the appearance of two stimuli (SOA).

4.1.3 BCI Competition III Dataset II

The BCI Comp Dataset [3] represents a complete record of P300 signals, gathered using the paradigm first described by Farwell and Donchin, where the user is presented with a 6x6 matrix of characters.

The signals were collected from two subjects across 64 electrodes, digitized at 240 Hz, and bandpass filtered from 0.1-60Hz. Both subjects spelled a total of 185 characters across 5 sessions. In each session, all rows and columns of this matrix were successively and randomly intensified for 100 ms with an ISI of 75 ms, resulting in a 175 ms SOA. For each character classification, there were 15 trials. For the competition dataset, the data has been divided into training data (85 characters) and testing data (100 characters).

Table 4.1 summarizes key aspects of the datasets used in this study. The ”Preprocessing” column indicates the methods that the original authors applied to each dataset beforehand.

4.1.4 Preprocessing Stage

The preprocessing applied to these datasets was minimal. A band-pass filter with cutoff frequencies of 0.1–30 Hz was applied to the BCI Comp data, removing the 50 Hz powerline frequency that could introduce signal artifacts, while also aligning with the cutoff frequencies of the ALS Dataset. Given the different amounts of electrodes in the BCI Comp Dataset, we applied electrode selection, retaining only the electrodes common to all datasets: FZ, CZ, PZ, P3, P4, PO7, PO8, and OZ. To achieve similar sampling rates, the BCI Comp and ALS Datasets were downsampled to 120 Hz and 128 Hz, respectively (GIB-UVa ERP-BCI data is already at 128Hz). All data from all datasets was then segmented into signal epochs with shape $(1, 8, 100)$ corresponding to (1,

[2] https://bnci-horizon-2020.eu/database/data-sets (Dataset 8)

[3] https://www.bbci.de/competition/iii/

Table 4.1
Dataset Details

Dataset	Subjects	SD	ISI	Epochs	Preprocessing
GIB-UVa ERP-BCI	73	62.5 or 75*	100-250*	701,615	0.5-45 Hz Band-pass 128 Hz sampling rate Common Average Reference Baseline Correction
ALS	8	125	125	33,600	0.1-30 Hz Band-pass 256 Hz sampling rate
BCI Comp	2	100	75	66,600	0.1-60 Hz Band-pass 240 Hz sampling rate

The ”Preprocesing” column refers to the techniques applied to the datasets by the authors, not the techniques we apply in our preprocessing stage. SD: Stimulus Duration in ms; ISI: Inter-Stimulus-Interval in ms. *The GIB-UVa ERP-BCI Dataset studied various SD and ISI values so some stimulus have either one or the other value specified for SD while ISI values are random values in the specified range.

$N_{electrodes}$, $N_{timepoints}$). In the datasets with a sampling rate of 128Hz, 100 time points correspond to approximately 780ms. In the case of the BCI Comp Dataset, which has a sampling rate of 120Hz, 100 time points correspond to approximately 830ms.

Lastly, Z-score normalization was applied to each signal epoch individually and for each electrode. Typically, studies that use neural networks with batch normalization skip normalization during preprocessing. Batch normalization computes statistics (mean and standard deviation) within each batch, normalizing the signals accordingly. These statistics, calculated during training, are reused to normalize signals in the testing phase. However, a challenge arises in cross-dataset evaluation, where a network is trained on signals from one dataset and tested on signals from a different one. As shown in Table 4.2, different datasets often have very different statistical values. Using the mean and standard deviation from one dataset to normalize signals from another could lead to inaccuracies. To address this, we apply normalization to each epoch individually before forwarding these signals to the networks, avoiding issues with batch normalization and ensuring reliable cross-dataset analysis.

Table 4.2
Dataset Statistics

Dataset	Data Shape	Min	Max	Mean	Std Deviation
ALS	(33600, 1, 8, 100)	-168.58	449.29	1.2×10^{-4}	9.53
BCI	(66600, 1, 64, 100)	-247.75	286.76	1.36	20.35
GIB-UVa ERP-BCI	(701615, 1, 8, 100)	-132.25	139.71	6.1×10^{-4}	1.50

”Data Shape” refers to ($N_{signals}$, 1, $N_{electrodes}$, $N_{timepoints}$). Min, Max, Mean, and Std Deviation values are expressed in μV.

4.2 Network Architectures

A common characteristic of the networks developed within the BCI paradigm is the use of convolutional layers that separately analyze signals in the time-frequency domain and in the spatial domain.

Convolutional layers dedicated to extracting temporal filters use kernel sizes of $(1, \frac{sf}{n})$, where *sf* is the sampling frequency and n is a positive integer defining the temporal span of the kernel in terms of sampling rate. For example, if *sf* $=$ 128Hz and $n = 2$ the kernel size is (1,64), spanning 64 time samples, equivalent to 500ms. This setup allows the kernel to capture oscillations down to 2 Hz, ($\frac{1}{0.5} =$ 2Hz). If $n = 4$, the kernel size becomes (1,32), capturing frequencies down to 4 Hz. By adjusting n, one can tune the kernel to capture specific frequency ranges based on the sampling rate and the frequency bands of interest.

Convolutional layers responsible for extracting spatial filters have kernel sizes of (C,1), where C is the number of electrodes. These layers usually operate on the outputs of the temporal convolutional layers, extracting spatial information from each temporal filter.

Combining these temporal and spatial filters enables the extraction of meaningful features across time, frequency, and spatial domains.

The following sections outline the architectures used in our study, along with the theoretical reasoning behind their design.

4.2.1 EEGNet

EEGNet, designed by Lawhern et al. [21], was developed with two primary goals: (1) to create a model that could be applied across various BCI paradigms, including motor imagery and ERPs, and (2) to ensure the model could be trained effectively with limited data.

The network's first block consists of a *2D Convolutional* layer, with a kernel size of (1, 64), producing $F1$ feature maps that capture EEG signals across different band-pass frequencies, referred to as temporal filters. Next, a *Depthwise Convolution* with filter size (C, 1), where C is the number of electrodes, is applied to learn spatial filters for each temporal filter. A depth parameter, D, defines the number of spatial filters per temporal filter. This block also incorporates batch normalization, exponential linear units (ELU), and dropout for regularization. Finally, an average pooling layer of size (1, 4) reduces the sampling rate by a factor of 4.

In the second block, a *Depthwise Convolution* of size (1, 16) is followed by $F2$ *Pointwise Convolutions* of size (1, 1), which learn to optimally combine the previous outputs. This combination of Depthwise and Pointwise Convolutions is referred to as a *Separable Convolution*, where $F2$ is typically set to $D * F1$. The block concludes with an average pooling layer of size (1, 8) for further dimensionality reduction. A visual representation of the EEGNet architecture is shown in Figure 4.1.

In this study, we adjusted the kernel sizes of the average pooling layers to better suit the requirements of contrastive learning techniques, which benefit from higher-dimensional representations [13]. Specifically, the kernel size of the first average pooling layer was modified from (1,

4) to (1, 2), and the second average pooling layer was changed from (1, 8) to (1, 4).

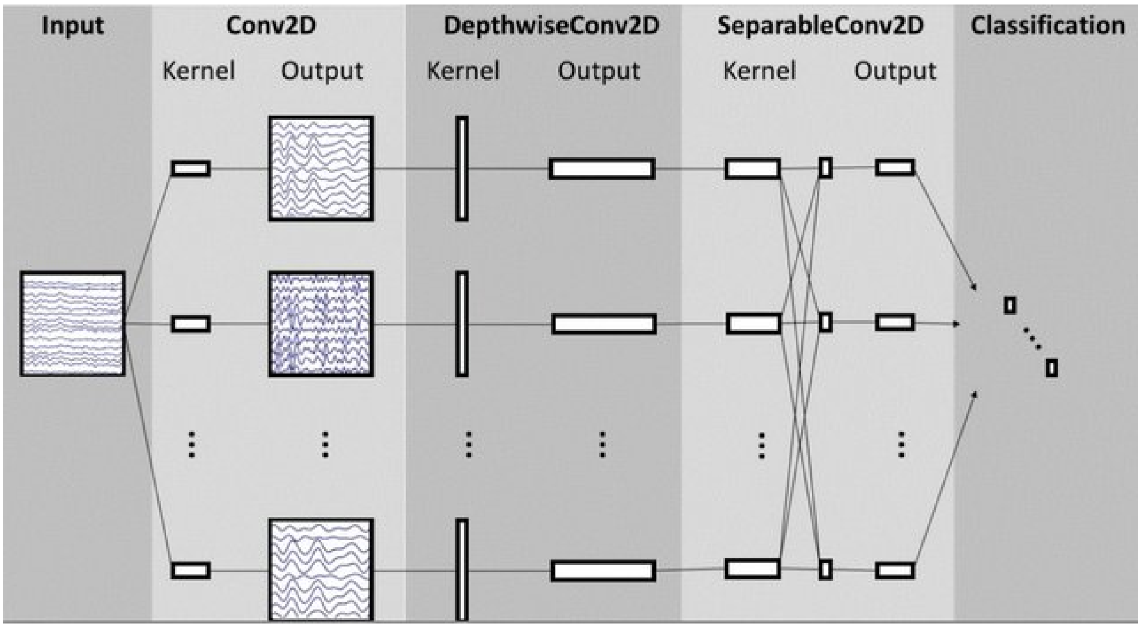


Figure 4.1: EEGNet model architecture. Image taken from [21] (Fig 1)

The notation used to represent the number of temporal and spatial filters in EEGNet is EEGNet-$F1, D$, where $F1$ indicates the number of temporal filters, and D represents the number of spatial filters per temporal filter. In this work we used EEGNet-16,4.

4.2.2 EEG-Inception

EEG-Inception, introduced by Santamaria-Vazquez et al. [34], incorporates inception modules for P300 signal detection. The model architecture is divided into three main blocks.

The first block is an inception module that processes the signal at three different temporal scales. This is achieved by using three separate *2D Convolutional* layers with kernel sizes of (1, 64), (1, 32), and (1, 16). With a sampling rate of 128Hz, the selected kernel sizes correspond to temporal windows of 500ms, 250ms, and 125ms, respectively, and extract features down to 2Hz, 4Hz, and 8Hz, respectively. After these convolutions, *Depthwise Convolutions* are applied to each temporal filter, extracting spatial filters similar to the approach used in EEGNet. The resulting outputs are concatenated, followed by an average pooling layer of size (1, 4) for dimensionality reduction.

The second inception module is structured similarly to the first block of the first module, but with kernel sizes reduced by a factor of four, reflecting the signal reduction performed by the preceding average pooling layer.

The third and final block is the output module, consisting of two convolutional layers designed to compress the extracted features into a lower-dimensional space. A visual representation of the EEG-Inception architecture is shown in Figure 4.2.

Several modifications were made to this network to achieve the same objective of increasing

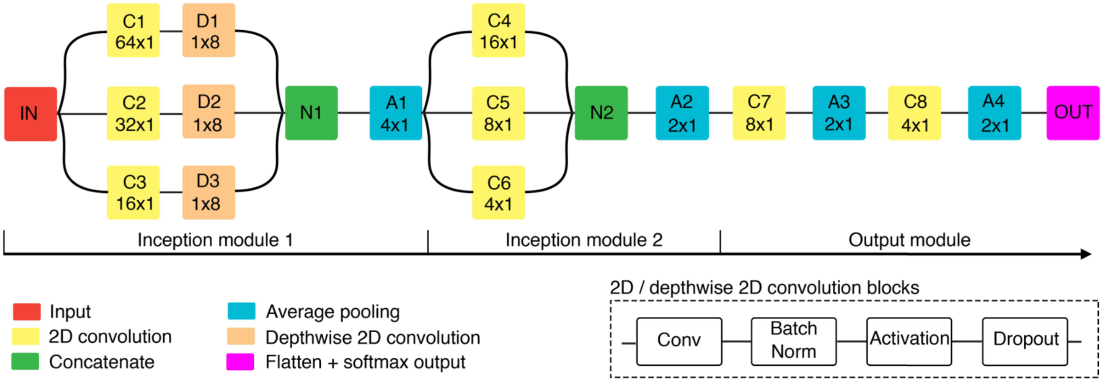


Figure 4.2: EEG-Inception model architecture. 2D convolution blocks and depthwise 2D convolution blocks include batch normalization, activation and dropout regularization. The kernel size is displayed for convolutional and average pooling layers, and are inverted because the authors data was structured differently: (1, $N_{timepoints}$, $N_{electrodes}$) instead of (1, $N_{electrodes}$, $N_{timepoints}$) employed in our study. Image taken from [34] (Fig 1)

the dimensionality of the representations, as previously mentioned. Unlike EEGNet, this network was originally designed to produce very low-dimensional output vectors, needing more extensive adjustments. All average pooling layers were removed to preserve higher-dimensional features. Additionally, the final block of the network, which was tailored for smaller input vectors, featured convolutional layers with smaller kernels. To accommodate the changes, the kernel sizes of the two convolutional layers in the final block were increased from (1, 8) and (1, 4) to (1, 32) and (1, 16), respectively.

4.2.3 Conformer

The Conformer Network, introduced in [37], is made up of three main components: a convolutional module, a self-attention module, and a fully connected classifier as depicted in Figure 4.3.

Convolutional Module: This module follows a similar structure to those in EEGNet and EEG-Inception. It first applies a convolutional layer with a kernel size of (1, 25) to extract time-frequency features from the signal. This is followed by a convolutional layer with a kernel size of (C, 1), where C is the number of electrodes, to capture spatial features from the temporal outputs of the first layer. Batch normalization and an ELU activation function are used to stabilize and enhance feature extraction. An average pooling layer reduces the dimensionality, creating a compact “token” representing the processed features from these layers.

Self-Attention Module: This module focuses on learning global temporal relationships within the signal. It operates by creating tokens from the previous layer’s output and transforming these into three equal-shaped vectors: query (Q), key (K), and value (V). The attention mechanism calculates the correlation between tokens by performing a dot product between Q and K, which reflects how each token relates to others. To maintain stable training, a scaling factor normalizes this result. The output is then passed through a Softmax function to create an “attention score” matrix, which represents the importance of each token in relation to others. This score is used to weight the V vectors via another dot product, capturing the contextually relevant features. To improve feature diversity, a multi-head strategy is applied: the tokens are split into h parts, each part goes through its own self-attention process, and the results are concatenated for the final output.

Classifier Module: This module consists of two fully connected layers that take the output of the Self-Attention Module and map it to a prediction about the class of the EEG signal.

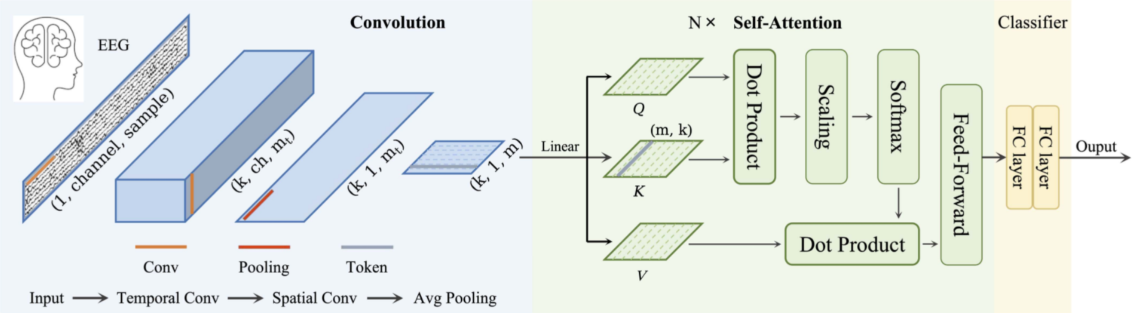


Figure 4.3: Conformer model architecture. Taken from [37] (Fig 1)

This network architecture was originally designed and evaluated for Motor-Imagery and Emotion Recognition datasets. To adapt it for P300 data, we modified the kernel sizes of the first convolutional layer and the average pooling layers in the Convolutional Module. Specifically, the kernel size of the first convolutional layer was adjusted to (1, 64), while the average pooling layer kernel size was set to (1, 6) with a stride of 1.

4.3 Training Methods

To develop systems that are broadly applicable across users, it is essential to leverage data from multiple subjects effectively. Training methods should encourage models to learn cross-subject invariant features so that they can generalize to new users with little to no calibration. Transfer learning is particularly valuable in this context.

In a transfer learning framework, models are first pre-trained on a large dataset with the goal of learning robust representations of the signals. These pre-trained weights are then transferred to a model that is fine-tuned on more specific data, either data from the same user or data from different users within the same dataset.

The main question we now examine is which pre-training strategy is best suited for the P300 BCI classification task.

4.3.1 Supervised

Supervised learning is a type of machine learning where models are trained on labeled data, meaning each training example includes an input and the correct output (or label). This approach enables the model to learn mappings from inputs to outputs, making predictions on new data based on patterns it has observed during training. The process begins with feeding the model a dataset that contains a wide range of examples with known labels, allowing it to minimize the error between its predictions and the real labels by adjusting its internal parameters. As the model trains, it becomes increasingly better at generalizing from training data to unseen data, assuming the data used for training accurately represents the types of examples it will encounter in testing. Typical loss functions employed in this method of training for classification tasks include the Cross Entropy Loss and the **Binary Cross Entropy Loss** used in binary classification, which is the one used in this study since the problem of identifying a P300 signal is a binary task.

4.3.2 SimCLR

The SimCLR framework for contrastive learning, introduced in [8], is a self-supervised learning technique that utilizes an instance discrimination pretext task. In this task, the goal during pre-training is to distinguish between different augmented versions of the same sample. The process of creating two distinct representations from a single sample is illustrated in Figure 4.4.

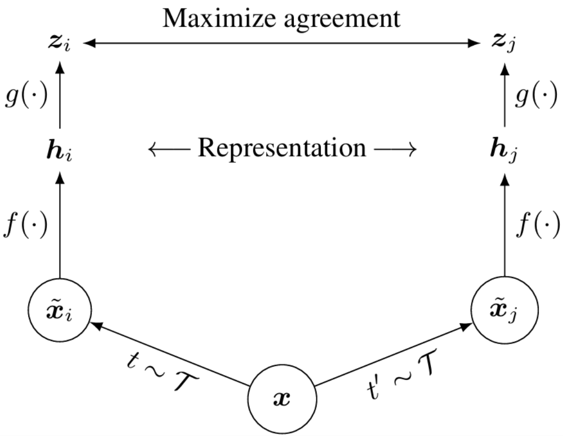


Figure 4.4: SimCLR Framework

First, two different views of the same sample denoted by $\tilde{x}_i$ and $\tilde{x}_j$ are created by applying random augmentations (t and t') to the original sample x. A **base encoder** $f(.)$ which can be a neural network such as ResNet50, VisionTransformer (ViT), or in our case, EEGNet, EEG-Inception, or Conformer, is then used to extract their respective representations h_i and h_j.

Next, a small neural network called the **projection head** $g(.)$ maps h_i and h_j to lower-dimensional projections, resulting in z_i and z_j which are the projections used in the contrastive loss calculation. The projection head applies a nonlinear transformation to the representations, which significantly enhances the performance in downstream tasks. It is important to note that the projection head is only used during the contrastive learning phase (pre-training) and is replaced by a **classification head** in downstream tasks.

After applying augmentations to the samples of a batch with size N, we obtain a batch with $2N$ samples (assuming each sample is augmented into two views). The contrastive loss is computed based on the projections within each batch. The contrastive loss function used in SimCLR is the

NT-Xent loss (Normalized Temperature-Scaled Cross Entropy Loss). Equation 4.1 shows the loss for element i.

$$l_{i,j} = -log \frac{exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} 1_{[k \neq i]} exp(sim(z_i, z_k)/\tau)} \tag{4.1}$$

Here $sim()$ is a similarity metric (in this case, cosine similarity) and τ is the temperature parameter. The indicator function $1_{[k \neq i]}$ evaluates to 1 when $k \neq i$. The numerator reflects the similarity between two views of the same sample (positive pairs), while the denominator is the sum of the similarities between one view and all other samples (both positive and negative). The objective of NT-Xent loss is to maximize the similarity between positive pairs (placing a high similarity score in the numerator) and minimize the similarity to all other (negative) pairs, which results in a low denominator value.

The final loss for a given batch is the mean of the losses for all views of all samples in that batch.

Additionally, the authors of [8] emphasize the importance of choosing the right set of data augmentations for learning effective representations. They also highlight that this method benefits significantly from large batch sizes and extended training sessions.

The NT-Xent loss (4.1) is specifically designed for scenarios where each sample has only two views. However, since our approach incorporates four views per sample, this formulation is not optimal. To address this, we adopt a modified version of the loss function that can handle more than two views per sample, referred to as the Normalized Temperature-scaled Binary Cross-Entropy Loss (**NT-BXent loss**) [27]. The equation below represents the loss for sample i:

$$l_{i,j} = -y_{i,j}.log\sigma(\frac{s_{i,j}}{\tau}) - (1 - y_{i,j}).log\sigma(\frac{1 - s_{i,j}}{\tau}) \tag{4.2}$$

Here, $s_{i,j}$ represents the similarity between i and j, calculated using cosine similarity. The function σ denotes the Sigmoid function, and $y = 1$ if i and j form a positive pair, otherwise $y = 0$.

4.3.3 SupCon

Supervised Contrastive Learning (SupCon) is part of the contrastive learning framework, that uses labeled data. It combines the strengths of self-supervised contrastive learning techniques, such as SimCLR, with supervised learning paradigms, enabling the model to better use label information for representational learning. It was proposed by Khosla et al. in [17], where the authors employed the same general framework used in SimCLR (see Figure 2.6) but introduced a new loss function that significantly improves the performance of supervised models.

While in techniques such as SimCLR the positive pairs are different augmentations of the same data sample, in SupCon the loss function leverages label information to consider all samples in a mini-batch with the same class as positive pairs, and samples with different labels as negatives.

This encourages the model to cluster together the representations of samples belonging to the same class and those of different classes far apart.

The **SupCon loss** function is defined by the following equation

$$\mathcal{L}_{\text{sup}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(sim(z_i, z_p)/\tau)}{\sum_{a \in A(i)} \exp(sim(z_i, z_a)/\tau)} \tag{4.3}$$

Here $P(i)$ denotes the set of all positive pairs for the view i, $A(i)$ is the set of all positives and negatives associated with view i, $sim()$ is again the cosine similarity and τ the temperature parameter.

Now, the loss is computed for a single view i by averaging the loss over all the positive pairs within the set $P(i)$. Specifically, the loss for a view is determined by comparing it to all other views of the same class (positive pairs). The goal is to maximize the similarity between the view i and other positive views (same class), while minimizing the similarity between i and views of other classes (negative pairs).

SupCon helps to create a better feature representation space where the learned features are well-separated based on class labels, leading to improved downstream task performance, such as classification or clustering.

4.4 Data Augmentations

Data augmentations are crucial in contrastive learning frameworks. While certain augmentations have been shown to enhance performance in standard supervised training, contrastive learning techniques typically require more aggressive data augmentations. According to the SimCLR paper [8], these techniques benefit from stronger data augmentation compared to traditional supervised learning.

Pretext tasks, particularly the one used in our study, rely heavily on the data augmentations applied to each sample. These augmentations can introduce biases and help the model become invariant to these transformations [15]. While this invariance can be beneficial in some scenarios, it can be detrimental in others. For example, P300 signals can be defined by their temporal and spatial characteristics. Applying standard augmentations to these domains might cause the model to become invariant to critical features of the P300 signal, impairing its ability to differentiate them effectively.

Therefore, common augmentations used in contrastive learning techniques applied to image classification tasks, such as rotation and cropping, are unsuitable for our problem as they disrupt both the temporal and spatial domains. Instead, there is the need to design custom augmentations that introduce sufficient biases to create challenging positive and negative samples while preserving the temporal and spatial features that are crucial for accurate discrimination.

In order to create multiple views of the same signal epoch capable of introducing sufficient variance for the networks to learn meaningful representations while preserving essential signal features, we developed a novel augmentation strategy, wich we called *Progressive Evokeds*.

4.4.1 Progressive Evokeds

As discussed earlier, an evoked signal is created by averaging multiple signal epochs. This averaging process improves the SNR of the signals. Enhancing the SNR in this way allows the models to perform better in P300 classification.

In addition to handling cross-subject variability, our goal is also to train the model in a way that enables it to represent single-trial signal epochs similarly to the way it represent their less noisy evoked counterparts. A crucial step for single trial classification.

To align our approach with this goal, we compute the evoked signals corresponding to each signal epoch. During the character-spelling process, the speller matrix completes several flashing loops or trials. Throughout these trials, the subject focuses on a single character, so each time a row or column flashes, the elicited P300 signal should theoretically resemble the P300 elicited in other trials. Ideally, all P300 signals should be similar, and we could calculate evoked signals by averaging randomly chosen signal epochs. However, this would overlook the factors contributing to the high variability in P300 signals, such as cross-subject variability, levels of focus, boredom, and fatigue, all of which are known to affect the signal. Differences in experimental setups also add to this variability. Additionally, this averaging approach would require access to labels, which is feasible in SupCon training but not in SimCLR, as it is a self-supervised learning technique.

Our approach, instead, builds the evoked signals by averaging all signal epochs corresponding to each row or column within trials corresponding to the same character. For example, suppose we have a signal epoch corresponding to the flashing of row 1, and it takes 10 trials to spell a character. The signal epochs used to create the evoked signal are all the signal epochs corresponding to row 1, giving us, in this case, 10 signal epochs in total for row 1.

Instead of averaging all 10 signal epochs at once to create a single evoked signal, we do this progressively: we first average the first two epochs, then the first three, and so on, until we reach an evoked signal formed from averaging all 10 epochs. This process yields 9 distinct evoked signals, ranging from the most similar to a single trial signal epoch to one with the highest possible SNR.

The idea is that by forming positive pairs with these progressively averaged evoked signals, the model learns to embed them close to one another. This way, it learns to map low-SNR signal epochs near higher-SNR evoked signals, ultimately improving the classification of signals with less trials.

The GIB-UVa ERP-BCI Dataset (used for pre-training) does not have a fixed number of trials; instead, the number of trials required to spell a character varies significantly, ranging from 4 to 15. Since the minimum number of trials in the pre-training data is four, this technique randomly selects three (since we extract $n - 1$ evoked signals from n trials) corresponding evoked signals to pair with each signal epoch, creating positive pairs for that epoch. This approach adds variability to the pairings each time it is applied by pairing the same signal epoch with different evoked signals when there are more than four trials available.

In the SupCon method, signal epochs are paired with both signal epochs and evoked signals from different subjects, as long as these signals belong to the same class. This approach not only addresses the single-trial classification challenge but also tackles cross-subject variability by encouraging the network to learn subject-invariant features, bringing signals from multiple individuals closer together in the feature space.

4.4.2 Data Augmentation Pipeline

Figure 4.5 provides an overview of the pipeline used during the pre-training stage for creating augmented views. Each signal epoch is paired with its corresponding evoked signals. Our augmentation framework randomly selects three evoked signals to form positive pairs with the signal epoch. Then, each signal is normalized as described in 4.1.4, resulting in four distinct views of the signal, one of which is the original signal itself.

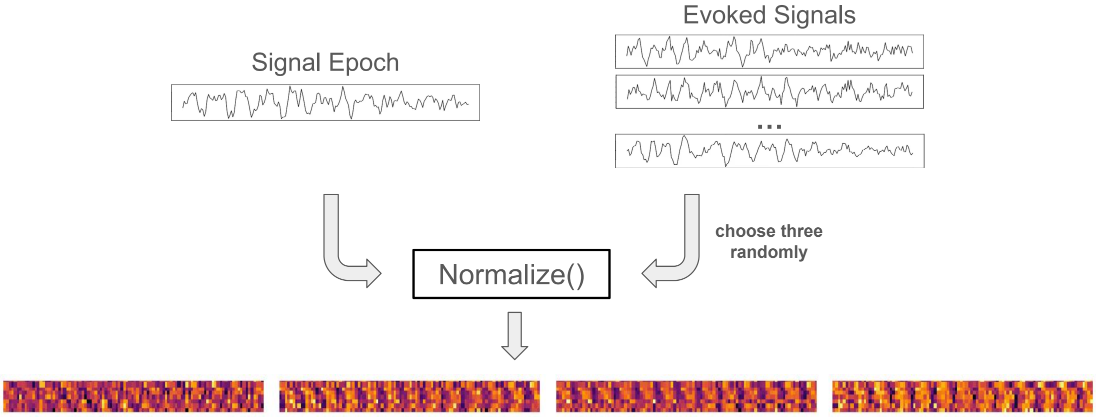


Figure 4.5: Data Augmentations Pipeline

4.5 Pre-Training Setup and Implementation Details

With the model architectures and pre-training methods defined, we now outline the complete pre-training approach used in our work.

Each model was pre-trained with the GIB-UVa ERP-BCI Dataset due to its large size. This dataset was partitioned differently for the two evaluation studies (detailed later in this chapter), but all evaluations divide the data into at least a training and validation set.

In the supervised learning setup, the training data includes both signals and labels, and the entire model (including the classifier) was trained end-to-end. The validation loop evaluates performance on accuracy, AUC, F1 score, recall, precision, and validation loss.

For the contrastive learning methods, we first replace each model's fully connected layer with a projection head before training on the data with the augmentations described.

To address the issue of unbalanced data, we used PyTorch's *WeightedRandomSampler*, which selects samples based on their probabilities, assigning higher probabilities to samples from underrepresented classes. This approach was applied during the pre-training of models using the

Supervised and SupCon methods, as SimCLR, lacking access to labels, could not employ this technique.

Evaluating representations in these frameworks requires a different validation approach, as models lack their fully connected layers during pre-training and cannot produce direct predictions. Instead, we evaluate representation quality using k-nearest neighbors (KNN), a standard offline evaluation protocol for self-supervised learning models [13], and adapt it for online evaluation (for validation), to monitor the training progression. Specifically, we split the validation set into two subsets: 80% for fitting a KNN model and 20% for testing it. The KNN model, set with k=5 and distance weighting, classified the base encoder signal representations based on their closest neighbors, with nearby samples exerting more influence. This approach is also applicable to models pre-trained with SupCon, though it is not strictly self-supervised.

Table 4.3 outlines the hyperparameters selected as well as some implementation details for pre-training the networks using various methods. These hyperparameters are standard choices commonly found in the literature. Regarding batch size, it is well-established that contrastive techniques benefit from larger batches [8]. Therefore, it might seem counterintuitive that the batch size for contrastive pre-training methods was set to 512, compared to 1024 for supervised learning. However, since each signal epoch is paired with three evoked signals, the real batch size for contrastive learning methods is effectively 2048. All models were trained for 200 epochs without early stopping. This decision was based on empirical observations that binary classification metrics (AUC, F1-score, recall, precision, and accuracy) do not strongly correlate with character recognition accuracy (the evaluation metric used during testing). As a result, the best model was not selected based on these metrics. Instead, the final model from the training stage was used. The validation loop was used to display validation metrics, providing a means to monitor the training process and ensure that the models did not overfit. While these metrics are not strongly correlated with character recognition accuracy, significant variations in the validation metrics can offer valuable insights into the performance and stability of the training process.

Table 4.3
Pre-Training Implementation Details

Train	Loss	Batch Size	Learning Rate	Optimizer	Scheduler	Weight Decay	Epochs
Supervised	Binary Cross Entropy	1024	1×10^{-3}	Adam	None	6×10^{-5}	200
SimCLR	NT-BXent	512	1×10^{-3}	Adam	Cosine Annealing	6×10^{-5}	200
Supcon	SupCon Loss	512	1×10^{-3}	Adam	Cosine Annealing	6×10^{-5}	200

This study was fully implemented using PyTorch 2.5.0 and PyTorch Lightning 2.4.0. All stages of the research were conducted with access to four NVIDIA A30 GPUs and CUDA 12.5.

4.5.1 Evaluation Methods for Contrastive Learning Pre-Trained Networks

As mentioned earlier, evaluating the performance of models trained using SimCLR and SupCon is more complex than evaluating models trained in a supervised manner. While the KNN protocol was discussed and implemented in the validation loop of the contrastive learning frameworks, other evaluation protocols are available to assess the performance of pre-trained networks. One such protocol is Linear Evaluation, which is applied after the pre-training phase to evaluate the network's performance and feature quality on a downstream task.

Linear Evaluation

In Linear Evaluation, the projection head is removed, the base encoder's parameters are frozen, and a classification head (either a single linear layer or a multi-layer perceptron (MLP)) is added on top. This classification head is then trained on labeled data to interpret the features extracted by the frozen base encoder.

Freezing the base encoder while training only the classification head allows us to assess how effectively the pre-trained base encoder represents signal epochs in unseen data without modifying its learned parameters. This approach also requires less data and is less susceptible to overfitting, as only the classification head is trained. To ensure consistency and evaluate the quality of representations, all models, including those pre-trained using the Supervised approach, were assessed through Linear Evaluation, using classification heads composed of a single linear layer.

4.6 Evaluations Setups

In this section, we describe the two evaluation setups employed in our work. We explore both Cross-Dataset Evaluation (training on one dataset and evaluating on another) and Intra-Dataset Evaluation (training and testing on data from the same dataset).

The following subsections outline the evaluation setup for each study while Table 4.4 provides a summary of the partitions and data allocation of each dataset.

4.6.1 Intra-Dataset Evaluation Setup

This evaluation setup focuses on assessing model performance within the same dataset, avoiding the challenges associated with cross-dataset studies. Our approach is inspired by the evaluation strategy outlined in [34], which used data exclusively from motor-disabled (MD) subjects. However, we introduced significant modifications to address two primary concerns:

1. **Training Data Quantity and Diversity**: The original evaluation in [34] used less than half of the available dataset for training. This limited the variability in pre-training data, potentially constraining the model's ability to generalize effectively. Pre-training models, especially those using contrastive learning, benefit significantly from diverse data, making it crucial to maximize the amount of data during this stage.

2. **System Users**: While MD subjects gain the most from systems like these, healthy individuals (control subjects, CS) are also potential users. Therefore, it is essential to evaluate system performance across both groups.

Based on these considerations, our evaluation strategy for this study divided the dataset as follows:

- **Test Set**: Includes data from 14 subjects, consisting of 7 CS (subjects = 2, 8, 9, 10, 16, 17, 20) and 7 MD subjects (subjects = 33, 34, 45, 59, 60, 64, 66).
- **Validation Set**: Comprises data from 4 randomly selected subjects (subjects = 0, 4, 18, 63).
- **Training Set**: Contains data from all remaining subjects.

After the pre-training stage, each model undergoes fine-tuning (retraining) using data corresponding to the spelling of N characters. Specifically, N was set to 1, 5, and 10. Here N is the equivalent of a real-world BCI calibration phase. Since we performed a Linear Evaluation of the models, this retraining data only trained the classification head of the network, while the parameters of the pre-trained base-encoder were frozen. The goal of this approach is to assess the quality of the learned representations and their ability to generalize from the training set subjects to the testing set subjects.

This evaluation approach directly addresses **RQ2** by investigating performance at varying calibration lengths, with smaller N representing shorter calibration phases. Additionally, the results provide insights into **RQ1** and **RQ3**.

To ensure robustness, we conducted 50 iterations of the fine-tuning and evaluation processes across all subjects and values of N with varying seeds and retraining characters. Final results were computed by averaging the outcomes across all iterations. The process described here is illustrated in Figure 4.6.

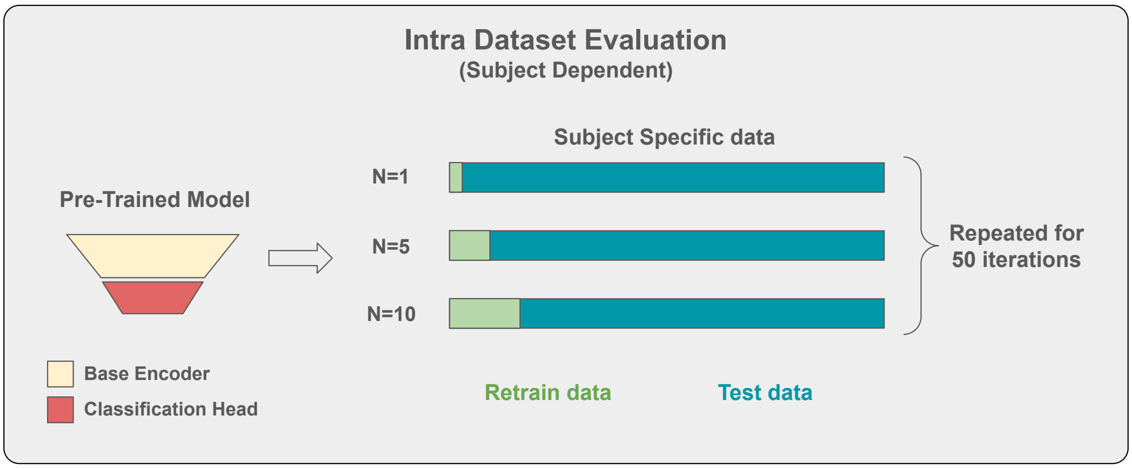


Figure 4.6: Intra Dataset Evaluation Setup

Character Recognition Procedure: The process of predicting characters from each subject's data proved to be more complex than initially anticipated. The diverse experimental protocols used to collect the GIB-UVa ERP-BCI Dataset introduced significant variations in the speller matrix design, the number of characters spelled by each user, and the number of trials conducted to spell each character. To address these challenges, particularly the variability in trial numbers, we ensured that only characters with 15 trials were used for retraining the pre-trained models when N was set to 1 and 5. This approach guaranteed that all users used the same amount of data for retraining.

When $N = 10$, and considering that some users had fewer characters with 15 trials, we reserved at least 10 such characters for the testing phase. Here, the only user that was trained with less data than the other was user 34, who had only 18 characters with 15 trials. For this user, 8 of these characters and 2 additional characters with 7 trials were used for retraining.

During the character recognition stage (testing phase), characters with fewer than 15 trials contributed results only up to the available number of trials. For example, a character with only 7 trials could only provide results for trial counts ranging from 1 to 7.

4.6.2 Cross-Dataset Study

As shown in Table 4.2, there is considerable variation across data from different datasets. This variation is not only present in each dataset global statistics but it encompasses many other aspects:

1. **Variability in Recording Hardware and Task Protocols**: Different datasets employ varied recording devices and electrode setups, leading to discrepancies in signal quality and noise levels. Additionally, variations in task design, stimulus types, timing, and instructions influence P300 response characteristics. For instance, within the datasets used in our study, we observe differences in the size of matrix speller interfaces across datasets. Smaller spellers produce fewer non-target signals, resulting in a higher frequency of P300 signals. This imbalance impacts classification performance and may weaken the P300 signal, as discussed in Chapter 2.

2. **Differences in Preprocessing**: Table 4.1 highlights differences in preprocessing across datasets. For instance, the GIB-UVa ERP-BCI Dataset applies common average referencing (CAR) and baseline correction, preprocessing steps not implemented in the other datasets. These differences present a challenge, particularly when evaluating a network, trained on one dataset, using data from a different dataset.

A robust model capable of handling these challenges would be highly beneficial to BCI paradigms, as it could leverage data from multiple sources, increasing the overall amount of available data.

This evaluation study assessed the ability of pre-trained models to classify data across datasets. First, the GIB-UVa ERP-BCI Dataset was still the dataset used for model pre-training. It was split into training and validation sets, with 4 subjects allocated to the validation set (the same 4 subjects as the Intra-Dataset Evaluation), while the remaining 69 subjects constitute the training set. Once pre-trained, these networks were then evaluated on two different datasets: the BCI Comp and ALS Datasets. Below, we outline the evaluation methodology for each dataset.

BCI Comp Dataset Evaluation: Here, we took a similar approach to the one in the Intra-Dataset Evaluation. As we can see in Figure 4.7, this dataset is split into training and test sets, where data from 85 characters per subject was reserved for training, while data from 100 characters was used for testing. During retraining, N was set to 1, 5, 10, 25, 40, and 85. Retrained models were then evaluated on the 100-character testing set. Both this process and the Intra-Dataset Evaluation Study are subject-dependent, as they require retraining the networks using each subject's data. However, retraining with 1, 5, and 10 characters corresponds to very short calibration phases, lasting approximately 30 seconds, 3 minutes, and 5 minutes, respectively, values significantly below the 20-minute threshold for acceptable calibration time [34]. Again, we ran 50 iterations of the retraining and testing procedures across subjects and values of N with varying seeds and retraining characters.

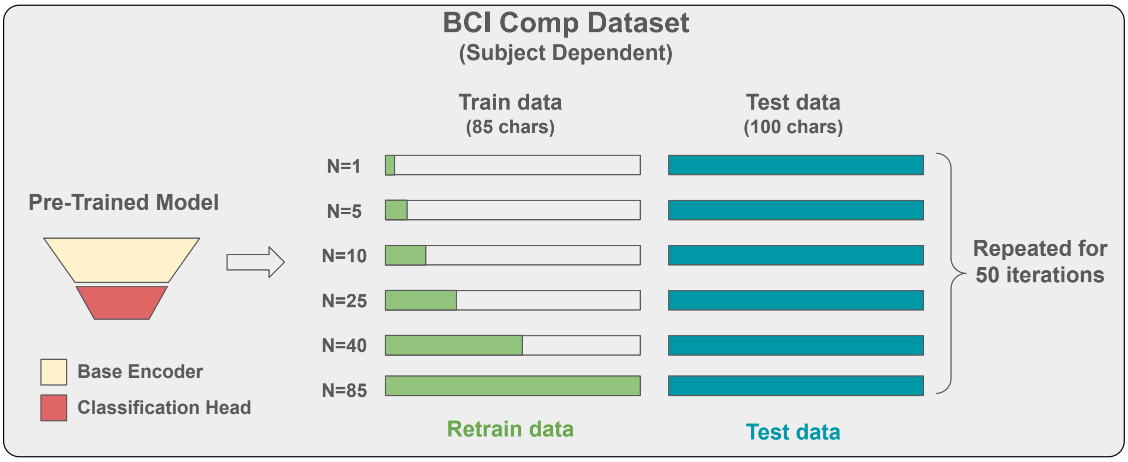


Figure 4.7: BCI Comp Evaluation Setup

Since this dataset is used extensively throughout the literature, besides giving us insights into **RQ1**, **RQ2**, and **RQ3** it also enables a more fair comparison with state-of-the-art networks described in Chapter3, addressing directly **RQ4**.

ALS Dataset Evaluation: This evaluation strategy is different from the ones previously described. Figure 4.8 illustrates the approach taken for this dataset. Here the test set comprised data from a specific subject, while the data used for retraining comprised data from all other subjects. Since no subject-specific data is used for retraining, this setup is considered a subject-independent evaluation, providing an answer to the first part of **RQ2**. Again, we performed 50 iterations of the procedure across all subjects.

4.7 Summary

In this chapter, we presented the complete methodology employed in this study. We began by describing the three datasets used: GIB-UVa ERP-BCI (pre-training and evaluation dataset), BCI Competition (evaluation dataset), and the ALS Dataset (evaluation dataset). Following this, we

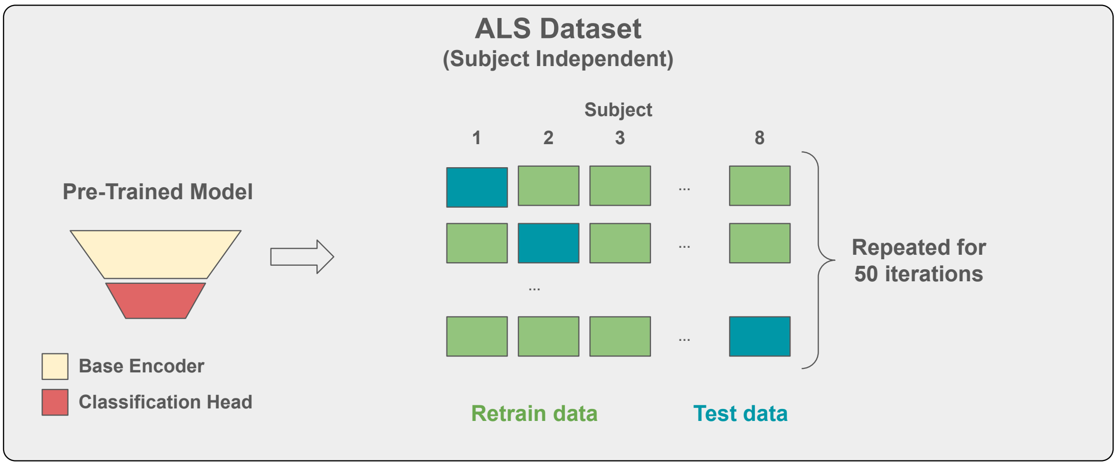


Figure 4.8: ALS Dataset Evaluation Setup

dived into the architectures of the models employed (EEGNet, EEG-Inception, and Conformer) detailing the function of each block in the models and the modifications made to adapt these networks to the specific context of this study.

Next, we explored the various pre-training methodologies (Supervised, SimCLR, and SupCon), outlining the framework of each approach and how can they be applied to the P300 classification paradigm. Within the contrastive learning framework, we identified the need for novel data augmentation techniques, capable of introducing variability while preserving critical signal characteristics. To address this, we proposed and described a new data augmentation technique called Progressive Evokeds.

Finally, we outlined the evaluation strategies employed in this work: Intra and Cross-Dataset Evaluations summarized in Table 4.4. These strategies were designed to analyze different aspects of the pre-trained networks and assess their performance in varied contexts.

Table 4.4
Evaluation Setups

Evaluation	Pre-Train Data	Validation Data	Test Data	Paradigm
Intra-Dataset	23 MD + 32 CS	4 CS	7 CS + 7 MD	Subject-Dependent
Cross-Dataset	30 MD + 39 CS	4 CS	BCI Comp (2 CS) ALS Dataset (10 MD)	Subject-Dependent Subject-Independent

MD: Motor Disabled subjects; CS: Control Subjects (Healthy).

Chapter 5

Results and Discussion

In this chapter, we present the results obtained from the two types of evaluations. For each evaluation, the three models (EEGNet, EEG-Inception, and Conformer) were pre-trained using three different strategies (Supervised, SimCLR, and SupCon) resulting in a total of nine models per evaluation. The primary distinction between the pre-trained models in the two evaluation studies lies in the amount of data used for pre-training, explicit in Table 4.4. Following the presentation of these results, we discuss them and propose possible explanations. Before diving into these analyses, it is essential to establish a baseline by examining the performance of the models under a fully subject-dependent paradigm (the standard approach). These baseline performances will provide a reference point for benchmarking our results.

5.1 Baseline Performances

First, it is important to address the question of the modifications made to the networks. As previously discussed, these networks were initially designed to produce small vector representations. However, contrastive learning techniques benefit from larger representations. To address this, modifications were made to the networks to increase the size of their output representations. Therefore, we start this analysis by confirming that the modifications made to the network architectures do not compromise their performances. Figure 5.1 illustrates a comparison of the original networks' performances with those of the modified networks on the BCI Comp Dataset, across 15 trials, using the entire training set (85 characters) for training. The training process setup was the same for every network, and all networks were trained using supervised learning for 100 training epochs, with early stopping (*patience=20*) monitoring the validation loss. This comparison was conducted only for the EEGNet and EEG-Inception networks, as these were designed and tested for the P300 classification problem. In contrast, the Conformer network was developed for the Sleep Stage Classification and Emotion Recognition paradigms, making it unsuitable to evaluate its original architecture on the P300 classification task. Presented values are the result of averaging 50 iterations of the training and testing process across different seeds for each model. Statistical significance was assessed using the Mann-Whitney U Test and significant trials are displayed as either a red bar ($p < 0.05$) or a gray bar ($p < 0.01$).

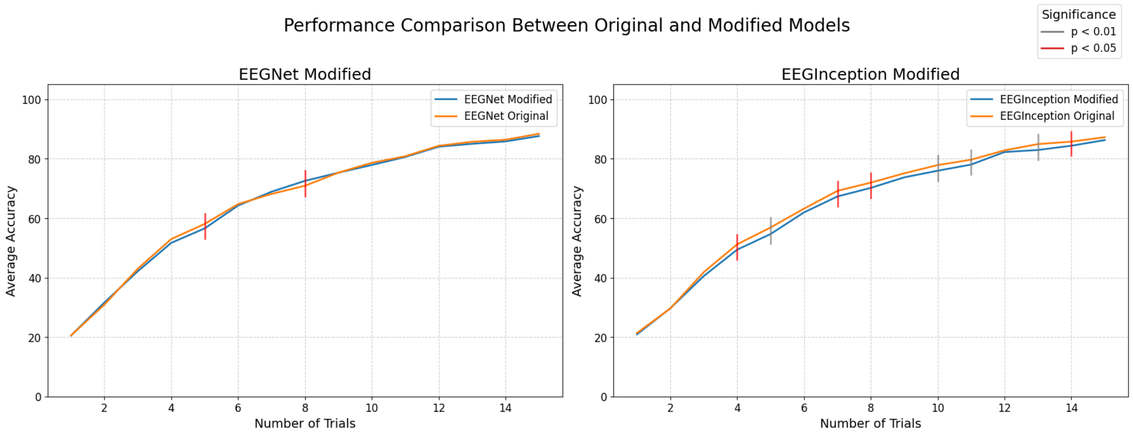


Figure 5.1: Comparison between the modified networks and their original counterparts. Results were obtained by running 50 iterations (with different seeds) of the training (using 85 characters) and evaluation process on the BCI Comp Dataset.

For the EEGNet network, we observe that there are only two trials with significant differences. When using five trials, the original network performs slightly better, but with eight trials, the modified architecture shows higher accuracy. Overall, these results indicate that the modifications made to the EEGNet network do not negatively impact its performance.

In the case of EEG-Inception, a more noticeable performance difference is observed across the number of trials. The original network tends to perform slightly better overall. However, this represents only a small drop in accuracy, which we considered acceptable given the potential advantages of the modifications during the pre-training stage.

Having confirmed that the modifications do not significantly impact performance, we now present the baseline results of these modified architectures, along with the Conformer, under a fully subject-dependent paradigm. All networks were again trained with supervised learning for 100 training epochs, with the same setup as described above. Figure 5.2 displays the performance of the networks on the BCI Comp Dataset alongside Information Transfer Rate (ITR) values (indicated by dotted lines). The networks were trained using the full training set (85 characters), and again, these values represent the average of 50 iterations of the training and testing process with different seeds. This time, statistical significance was evaluated using the Friedman test, followed by Wilcoxon post hoc analysis with Benjamini-Hochberg correction. As anticipated, performance improves with an increasing number of trials. In contrast, ITR peaks for EEGNet and Conformer when using six trials, and reaches its highest point with seven trials for EEG-Inception. ITR reflects the system's efficiency in transmitting information, balancing the trade-off between speed and accuracy.

While the networks exhibit comparable results overall, EEGNet consistently outperforms the others, followed by EEG-Inception, with Conformer demonstrating the lowest performance. The results indicate statistically significant differences between all models across trials ($p < 0.01$), except for EEGNet and EEG-Inception with a single trial, and EEG-Inception and Conformer

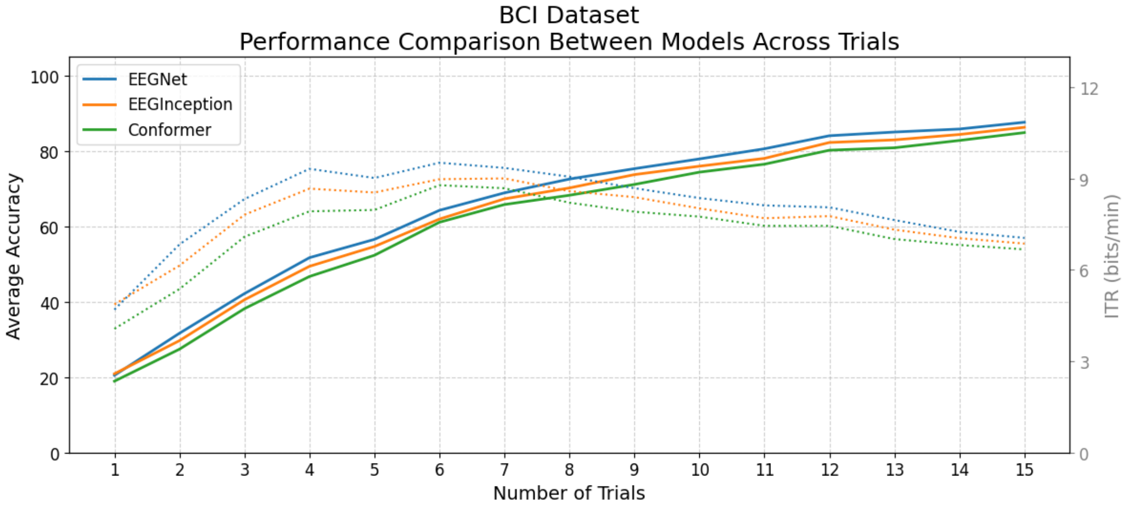


Figure 5.2: Baseline Subject Dependent Models' Performance across trials. Results obtained by averaging results from 50 iterations (with different seeds) of the training (using 85 characters) and evaluation process on the BCI Comp Dataset. Dotted lines indicate ITR values.

with six trials.

From this point onward, all comparisons with baseline models, for the BCI Comp Dataset, will be based on these results, obtained with the modified versions of the networks.

5.2 Intra-Dataset Evaluation

The results of the Intra-Dataset Evaluation are illustrated in Figure 5.3, showcasing performance across trials for the three models, each pre-trained using the three distinct strategies. The results are provided for each value of N and represent the average of 50 iterations, with variations in the random seeds and the selected characters for retraining. Results are also summarized in Table A.1, detailing performance for 1, 5, 10, and 15 trials as well as standard deviations for each. The ”Diff” column highlights the significant differences between the results of the contrastive learning techniques and the Supervised approach. These differences were determined based on trials where results showed statistical significance ($p < 0.05$) compared to the Supervised method. To evaluate statistical differences in performance across trials for the various strategies, the Friedman test was again applied, followed by the Wilcoxon Signed-Rank Test with Benjamini-Hochberg correction. Statistical significance is indicated for each trial by a red line ($p < 0.05$) or a gray line ($p < 0.01$) for comparisons between Supervised and SupCon. Comparisons involving SimCLR consistently showed statistically significant differences ($p < 0.01$).

A notable observation in this evaluation study is the consistently low performance of SimCLR across all models and values of N ($p < 0.01$ for all trials, compared with both Supervised and SupCon). This indicates that SimCLR struggled to generalize from the training subjects to the test data subjects in this evaluation. In contrast, the comparison between Supervised and SupCon yields different results.

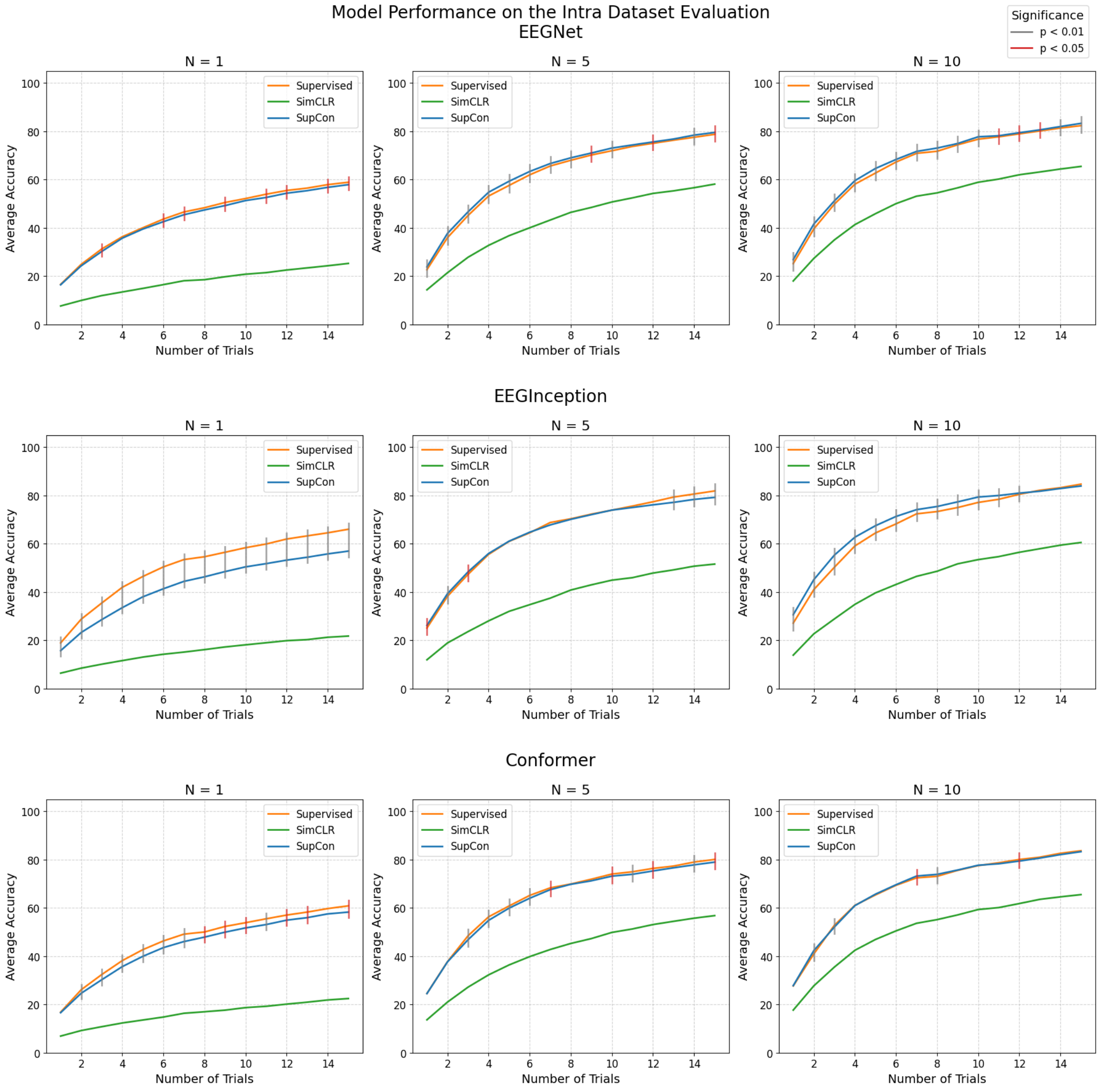


Figure 5.3: Network performance across trials for each strategy and value of N on the Intra-Dataset evaluation. The results represent the average of 50 iterations, calculated across various seeds and choice of training characters.

Examining both Figure 5.3 and Table A.1, the Supervised approach outperforms SupCon when $N = 1$, particularly with the EEG-Inception model, where the largest average difference is observed (-7.89 percentage points compared to -1.15 for EEGNet and -2.27 for Conformer). However, as N increases to 5, SupCon starts closing the gap with Supervised and even surpasses it in the case of EEGNet.

By $N = 10$, SupCon outperforms Supervised across all models. It emerges as the best pre-training strategy for EEGNet across all trials and for EEG-Inception in nearly all trials. However, the performance differences between these strategies are minimal, with the largest difference being 2.74 percentage points for EEG-Inception. Among the models, Conformer shows the smallest gain in performance between SupCon and Supervised, achieving a 0.31 percentage point difference to

Supervised when $N = 10$.

To gain deeper insights into the results, we also analyzed the difference in performance when the models classified CS and MD subjects separately. Table A.2, and Table A.3, present the performance of each model pre-trained with each strategy for the 7 MD subjects, and 7 control subjects, respectively. The results are shown for 1, 5, 10, and 15 trials across the different values of N, and the "Diff" values were calculated the same way as for Table A.1.

MD Subjects: Starting with the results for the MD subjects (Table A.2), EEG-Inception trained with SimCLR was the worst-performing model across all values of N. None of the classifiers demonstrated strong performance with SimCLR, however, EEG-Inception benefited the least from this pre-training strategy compared to the other models, as indicated by the "Diff" values.

The best-performing model was EEG-Inception pre-trained with the Supervised approach, achieving recognition accuracies of 55.83%, 72.55%, and 75.72% with 15 trials, and N set to 1, 5, and 10, respectively. Overall, models pre-trained using supervised learning consistently demonstrated superior performance, especially when $N = 1$, compared to contrastive learning methods. A notable exception is the EEGNet architecture, which showed a minimal improvement in performance with SupCon compared to Supervised for $N = 5$, and 10. Overall, the same trend as before is observed: as the value of N increases, the performance gap between Supervised and contrastive learning strategies narrows.

The relatively low accuracy results for MD subjects were anticipated, as their signals are inherently more challenging to classify. It is also important to note that these results do not reflect an optimization of model performance. Instead, we conducted a simple linear evaluation, where the number of characters used for retraining (N) are used to train a single linear layer while keeping the base encoder parameters frozen. As discussed in Section 4.5, this approach assesses the quality of the representations learned by the base encoders. Therefore, throughout these evaluations, we did not fine-tune the entire network to the specific subject's data. Rather, we focused solely on the representational quality of the networks. Consequently, the base encoders remained untouched by specific user data and relied entirely on the knowledge acquired during pre-training. While full model fine-tuning was not performed in this project, it is expected that such fine-tuning would improve overall model performance, at least for higher values of N, compared to the simple linear evaluation used here.

Control Subjects (CS): Analyzing the CS results presented in Table A.3, we observe a significant improvement in performance compared to MD subjects. Interestingly, for single-trial recognition, the performance is comparable to that of the MD results, and in some cases, models classifying signals from MD subjects even outperform results from control subjects in single-trial scenarios.

Models pre-trained with SimCLR continue to underperform compared to other pre-training strategies. Among these, EEG-Inception remains the least benefited by SimCLR, except when $N = 1$, in this case, the largest negative difference is observed in Conformer. When $N = 10$

EEGNet and Conformer delivered acceptable results with SimCLR, achieving nearly 85% accuracy with 15 trials. However, this still falls short of the performance observed with Supervised and SupCon approaches.

For control subjects, the dynamic between Supervised and SupCon shifts. The performance gap between the two strategies is smaller at $N = 1$ compared to what was observed with MD subjects. Notably, there is no statistically significant difference between the performances of SupCon and Supervised for EEGNet at this value of N. As N increases, SupCon consistently outperforms Supervised, except in the case of Conformer when $N = 5$. The best result in this setting was achieved by EEG-Inception pre-trained with SupCon, presenting 96.09% accuracy with $N = 10$ and 15 trials, closely followed by the Conformer model, which achieved 95.83% under the same conditions.

Key Insights from the Intra-Dataset Evaluation: Overall, the main takeaways from this Intra-Dataset Evaluation are the following:

1. **Performance Trends**: As anticipated, performance improves with an increasing number of trials and higher values of N.

2. **Classification Challenges of MD Subjects**: Model performance on MD subjects is notably lower than on control subjects. This discrepancy has a pronounced impact on contrastive learning techniques, which struggle to generalize effectively for MD subjects. With these subjects, contrastive learning pre-trained models exhibited a larger negative difference in performance to their supervised counterparts.

3. **SimCLR Limitations**: Among all pre-training strategies, SimCLR consistently demonstrated the worst performance across all models.

4. **Supervised vs. SupCon**: Supervised pre-training achieved the best results compared with SupCon for lower values of N. However, as N increases, the performance gap between Supervised and SupCon narrows, with SupCon outperforming Supervised, particularly with EEGNet and EEG-Inception.

5. **Impact of Architecture**: The effectiveness of contrastive learning techniques seems to be dependent on the model architecture. SimCLR paired with EEG-Inception produced the worst results across this evaluation. On the other hand, EEGNet showed the most significant benefit from contrastive learning strategies, with SimCLR achieving relatively better results when paired with EEGNet, followed by the Conformer, and EEGNet SupCon achieving better results than EEGNet Supervised for both $N = 5$, and 10.

5.3 Cross-Dataset Evaluation

As discussed in subsection 4.5.1, networks face significant challenges when attempting to generalize across different datasets. However, the ability to perform well across datasets offers substantial benefits, with the primary advantage being increased data availability for training the networks. Thus, this evaluation aims to assess the robustness of classifiers and pre-training strategies within this cross-dataset paradigm.

In this evaluation, the networks were again pre-trained on data from the GIB-UVa ERP-BCI Dataset. Those pre-trained networks were then, in the case of BCI Dataset Evaluation, retrained using varying numbers of characters (N) from each subject in the BCI Comp Dataset training set, and subsequently evaluated on the 100 characters that comprise the testing data. In the case of the ALS Dataset Evaluation, data from one subject was allocated to the testing set while the remaining subjects' data was used for retraining. This process was repeated for every subject.

In both dataset evaluations, the process of training and testing was repeated for 50 iterations for each subject, model, and value of N (in the case of the BCI Comp Dataset), varying the selected characters and seeds. The statistical tests employed were identical to those described in the previous evaluation study.

5.3.1 BCI Comp Dataset

The results of this evaluation for all models are presented in Table A.4 and will serve as support for the result analysis of this evaluation.

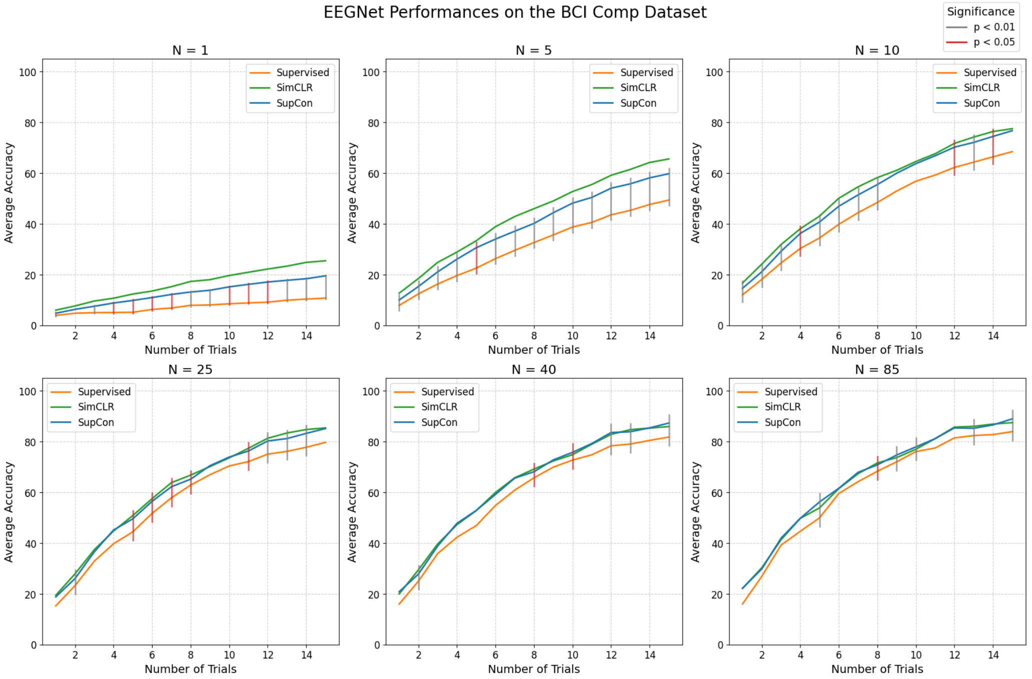


Figure 5.4: BCI Competition Dataset: EEGNet results for each value of N across trials. Performance results reflect the averaging of 50 iterations of the training and testing process. Statistical significance is indicated with red bars ($p < 0.05$) or gray bars ($p < 0.01$).

We begin this analysis with EEGNet, whose results are presented in Figure 5.4. Statistical significance is represented by red bars ($p < 0.05$) or gray bars ($p < 0.01$), highlighting the differences between the Supervised results and the contrastive learning technique with the closest value for each trial.

Unlike the previous evaluation, SimCLR demonstrates superior performance compared to the other strategies, followed by SupCon. This performance gap between supervised and contrastive learning strategies is particularly noticeable for smaller values of N. As N increases, the supervised EEGNet begins to close the gap with the contrastive learning techniques, though it never fully matches their performance. These results suggest that EEGNet trained with supervised learning faces challenges in generalizing from one dataset to another, whereas contrastive learning techniques exhibit greater robustness, achieving satisfactory performance when $N > 10$. Notably, this classifier pre-trained with SupCon achieved the highest accuracy in this evaluation, yielding 87.03% with $N = 85$ and 15 trials as demonstrated in Table A.4.

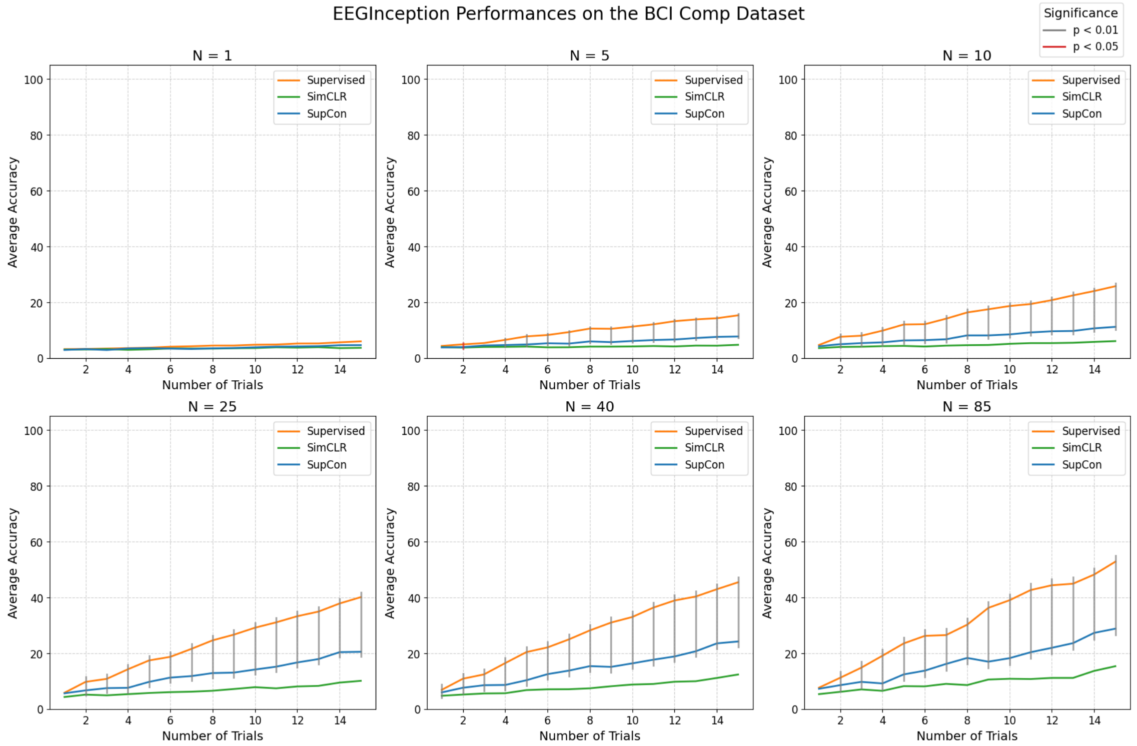


Figure 5.5: BCI Competition Dataset: EEG-Inception results for each value of N across trials. Performance results reflect the averaging of 50 iterations of the training and testing process. Statistical significance is indicated with red bars ($p < 0.05$) or gray bars ($p < 0.01$).

Turning to EEG-Inception, we observe a completely different outcome. This network performed poorly in this Cross-Dataset Evaluation, as shown in Figure 5.5. To put things into perspective, and looking at Table A.4, the best result for EEG-Inception was achieved with Supervised, when $N = 85$ and using 15 trials, yielding an accuracy of 49.75%. With either SimCLR or SupCon and $N = 1$, the network's accuracy was comparable to that of a random classifier (around 2.8%). These findings align with the results observed in the Intra-Dataset Evaluation, where EEG-Inception showed the least benefit from contrastive learning pre-training techniques of all the models, especially SimCLR. However, in contrast to what we observed in the Intra-Dataset Evaluation, the gap between SimCLR and Supervised widens as the value of N increases. This highlights the particularly poor performance of SimCLR for the EEG-Inception network, especially in this cross-dataset setting. The highest accuracies recorded for contrastive learning methods for this network were 37.18% with SupCon and 17.01% with SimCLR, both achieved when $N = 85$ and using 15 trials. The consistently poor performance of this model across all pre-training strategies suggests that the limitations may stem from issues with the network architecture rather than the pre-training strategies themselves.

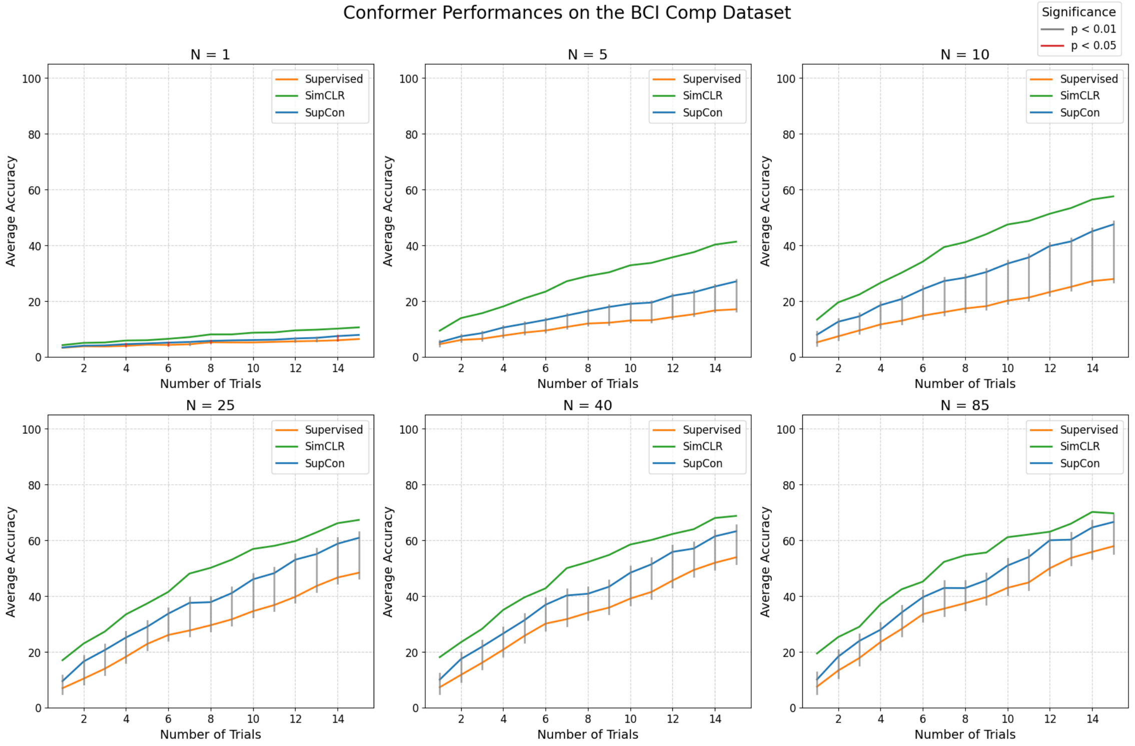


Figure 5.6: BCI Competition Dataset: Conformer results for each value of N across trials. Performance results reflect the averaging of 50 iterations of the training and testing process. Statistical significance is indicated with red bars ($p < 0.05$) or gray bars ($p < 0.01$).

Lastly, we examine the results of the Conformer network, shown in Figure 5.6. Here we observe the same pattern as with EEGNet, where contrastive learning techniques outperform supervised learning across all values of N. However, the overall performance of Conformer is lower than that of EEGNet, highlighting the challenges of cross-dataset evaluation for this network. As seen in Table A.4, the highest accuracy achieved by Conformer is 70.18%, attained with SimCLR, $N = 85$, and 14 trials. While this is approximately a 20 percentual points increase compared to the maximum accuracy of EEG-Inception, it still lags behind EEGNet's best performance. Another notable observation is that, although Conformer pre-trained with supervised learning improves performance as N increases, it does not close the gap as we have seen with EEGNet. This further underscores the superiority of contrastive learning techniques for cross-dataset evaluations with this network.

Comparison with Baseline and Reported Results in the Literature: We now compare the performance of the models evaluated in this study with the baseline networks. We present these comparisons using the best-performing strategy for each model on the BCI Comp Dataset. Figure 5.7 displays the results of the top-performing strategies for each model (with $N = 85$), comparing them against their fully supervised and subject-dependent counterparts, detailed in Section 5.1. The best-performing strategy for EEGNet in this evaluation was SimCLR. As seen in Figure 5.7, EEGNet with SimCLR shows performance comparable to its fully subject-dependent counterpart, also achieving comparable ITR values (indicated by dotted lines). Since only the classification head was trained in the EEGNet SimCLR, these results suggest that the network successfully transferred its learned representations from the pre-training stage to the BCI Comp dataset signals and was able to achieve the same performance as a fully subject-dependent network.

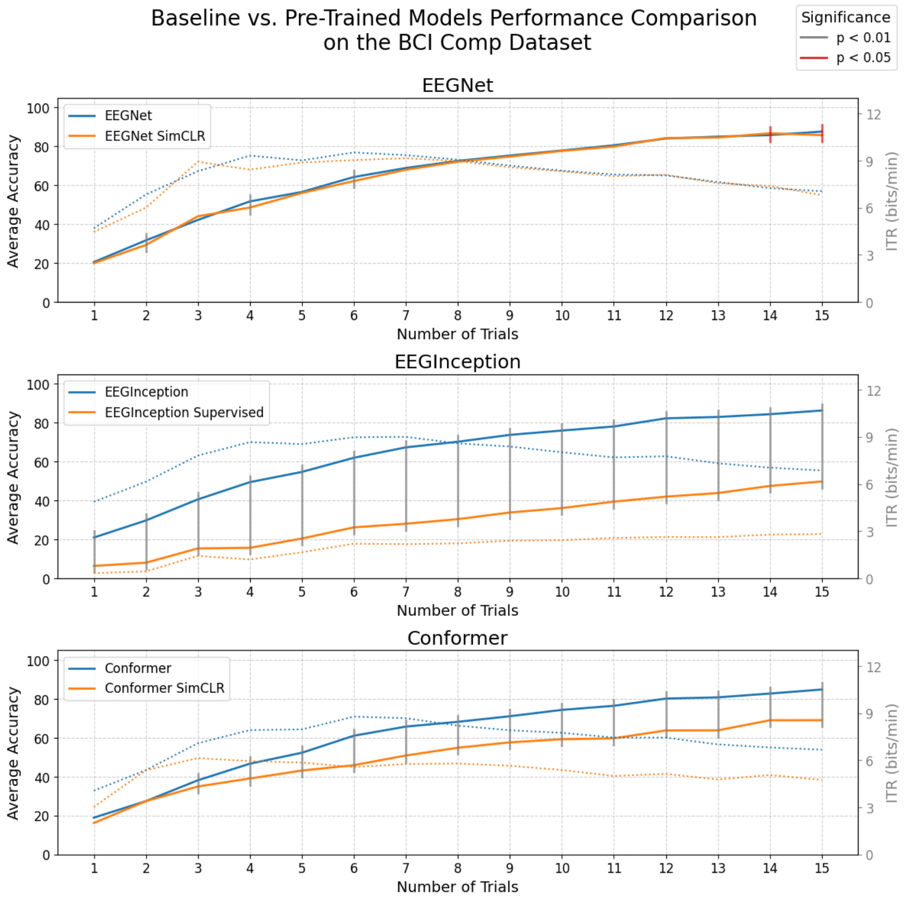


Figure 5.7: Baseline Models vs. Pre-Trained Models Performance Comparison, on the BCI Comp Dataset, using 85 characters for training/retraining. Dotted lines indicate ITR values.

On the other hand, EEG-Inception and Conformer exhibited different performance patterns.

EEG-Inception, when trained in a supervised manner, showed significantly lower performance compared to its fully subject-dependent counterpart. A similar trend was observed with Conformer, where the performance of Conformer SimCLR was lower than that of its subject-dependent version, with the difference here being less pronounced than with EEG-Inception. Conformer SimCLR demonstrated more robustness in cross-dataset scenarios compared to EEG-Inception Supervised. Both networks were trained or retrained with the same amount of data (85 characters), and these results indicate that neither architecture generalizes particularly well in cross-dataset settings, presenting results much below their subject-dependent versions, meaning that the learned representations from their pre-training stages do not transfer as effectively as those of EEGNet SimCLR.

Table 5.1
State-of-the-Art Results on BCI Comp Dataset
Average Character Recognition Accuracy across Trials

Paper	Model	Number of Trials														
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
[26] (2018)	BN3	34.5	49.0	64.0	70.0	74.5	78.5	81.5	86.0	88.0	90.5	92.0	93.5	94.0	95.0	96.5
[20] (2019)	SAE-ESVM	30.5	49.0	63.5	70.0	75.5	81.0	82.5	85.0	90.0	92.0	94.0	95.5	97.5	98.0	99.0
	SSAE-ESVM	31.5	48.5	65.0	69.5	75.5	83.0	81.0	85.0	90.0	91.5	93.0	95.5	96.5	97.5	98.5
[31] (2008)	ESVM	25.5	42.5	57.0	64.0	73.5	_	_	_	_	87.0	_	_	95.0	_	96.5
[10] (2023)	MultiClass-ESVM	27.0	47.0	65.0	67.5	74.0	_	_	_	_	88.5	_	_	_	_	98.0
[18] (2023)	CNN+EA+DRBM	33.5	_	_	_	73.5	_	_	_	_	90.5	_	_	_	_	97.5
[7] (2011)	CNN-1	25.5	42.5	53.0	60.0	70.0	73.0	79.5	83.5	88.5	88.5	90.5	90.5	91.0	92.5	94.5
	MCNN-1	28.5	43.0	56.0	59.0	69.0	73.5	81.0	84.0	85.0	87.0	92.0	93.5	93.0	93.5	95.5
[21] (2018)	EEGNet †	30.33	44.48	59.33	65.21	72.06	77.48	81.78	84.35	85.78	88.63	89.34	91.12	91.78	93.46	94.71
[34] (2020)	EEG-Inception †	30.88	44.91	60.45	66.69	73.88	78.86	83.13	85.0	86.95	90.09	89.9	91.81	91.72	93.26	94.73
[21] (2018)	EEGNet*	20.49	31.64	42.16	51.69	56.55	64.25	68.89	72.55	75.3	77.89	80.58	84.04	85.02	85.82	87.62
[34] (2020)	EEG-Inception*	20.9	29.68	40.53	49.4	54.66	61.94	67.3	70.19	73.73	75.96	78.03	82.25	82.94	84.37	86.27
[37] (2022)	Conformer*	18.93	27.42	38.17	46.7	52.33	61.09	65.79	68.25	71.12	74.38	76.49	80.2	80.84	82.79	84.87
Present study	**EEGNet SimCLR***	19.91	29.27	44.03	48.53	56.01	62.17	67.99	72.1	74.74	77.66	79.92	84.31	84.55	86.75	85.8

Thicker lines are used to distinguish between the results reported in the literature, which were not implemented by us, and the results obtained by the models we implemented, with the best-performing model in this BCI Comp Dataset evaluation (EEGNet SimCLR) highlighted.†These results reflect an evaluation conducted by our implementation of the models with 64 electrodes. *These results reflect an evaluation conducted by our implementation of the models with only 8 electrodes. In contrast, all other studies utilized 64 electrodes. ”_” indicates that no results for these numbers of trials were provided.

Table 5.1 presents a comparison of EEGNet SimCLR’s performance against our baseline models and previous studies utilizing the BCI Comp dataset. It is important to note that the performance values reported in the literature were obtained using 64 electrodes, whereas the models in this study were trained and evaluated with only 8 electrodes. As expected, reducing the number of electrodes leads to a decline in accuracy, making direct comparisons challenging.

This performance decline is evident when comparing EEGNet and EEG-Inception trained with 64 electrodes versus the same architectures trained with only 8 electrodes. For instance, using 15 trials, EEGNet achieves an accuracy of 94.71% with 64 electrodes, but this drops to 87.62% when using just 8 electrodes. These results provide an estimate of the expected accuracy drop when we reduce the number of electrodes.

Despite this, EEGNet SimCLR matches the baseline models (all using 8 electrodes) and even surpasses the fully subject-dependent Conformer baseline. Notably, all networks were trained exclusively on data from the BCI Comp dataset. However, in the case of our best-performing model, EEGNet SimCLR, only the classification head was fine-tuned on this dataset, while the base encoder relied solely on the knowledge gained during the pre-training stage.

5.3.2 ALS Dataset

The ALS Dataset Evaluation procedure assessed model performance in a completely subject-independent paradigm. In this setup, models were evaluated on data from a specific subject, while data from the remaining subjects was used for retraining the classification head. The results of this evaluation are shown in Figure 5.8 and Table A.5. The same procedures as before, concerning statistical significance, number of iterations, and calculation of ”Diff” values, were applied in this evaluation.

In this dataset evaluation, similar patterns to those observed in the BCI Dataset evaluation emerge. EEGNet consistently demonstrates the best overall performance, with contrastive learning techniques clearly outperforming Supervised. The average performance gap is 20.57 percentage points in favor of SimCLR over Supervised and 16.58 percentage points for SupCon over Supervised. The best accuracy achieved by this model was with SimCLR, yielding 66.23% using 10 trials.

EEG-Inception once again underperforms in this evaluation, with the best accuracy reported being 25.71% using 10 trials, achieved by the model trained with Supervised. In this case, contrastive learning techniques continue to show even poorer performance, with an average gap of -6.69 percentage points for SimCLR and -5.40 percentage points for SupCon compared to Supervised.

The Conformer network also shows poor performance in this evaluation. However, unlike EEG-Inception, contrastive learning outperformed Supervised, particularly SimCLR, which achieved a maximum accuracy of 34.54% with 10 trials, an improvement of 11.30 percentage points over Supervised. SupCon, on the other hand, shows a much smaller increase, with only a 1.09 percentage point improvement over Supervised, reaching a maximum accuracy of 17.24% using 10 trials. These results are significantly lower than those observed with EEGNet.

Comparing the performance of our best-performing model (EEGNet SimCLR) with the baseline models classifying signals from the ALS Dataset, as illustrated in Figure 5.9, reveals notable differences. The baseline models were trained exclusively on data from this dataset, using a procedure similar to ours: one subject's data was allocated to the test set while the remaining subjects' data were used for training. This process was repeated for all subjects, with 50 iterations per subject.

In this fully subject-independent paradigm, our best model underperformed compared to the models trained solely on data from this dataset. The lower performance of EEGNet SimCLR on the ALS Dataset, as seen in Figure 5.9, compared to its performance on the BCI Dataset in Figure

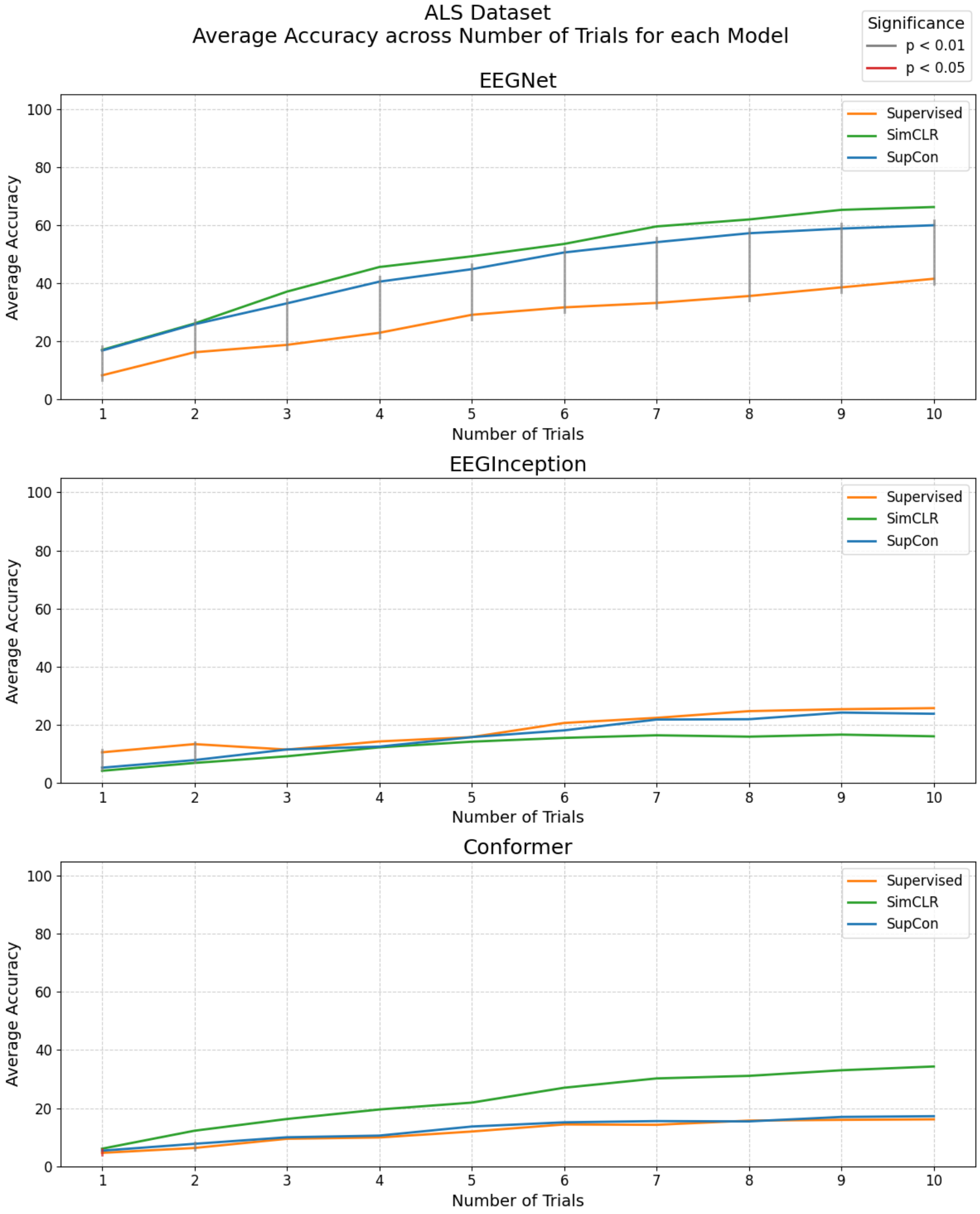


Figure 5.8: ALS Dataset Results for each Model. Performance results reflect the averaging of 50 iterations of the training and testing process. Statistical significance is indicated with red bars ($p < 0.05$) or gray bars ($p < 0.01$).

5.7, highlights the difficulty of generalizing to MD subjects using a pre-training stage that included both CS and MD subjects. Again, it is important to note that the EEGNet SimCLR base encoder was not retrained, only the classification head was updated during retraining. Consequently, the representations produced by the EEGNet SimCLR base encoder rely entirely on the pre-training stage.

Also, it is important to note that the ITR values presented here as dotted lines were calculated based on an assumed interval of 2.5 seconds between the spelling of each character (the same as

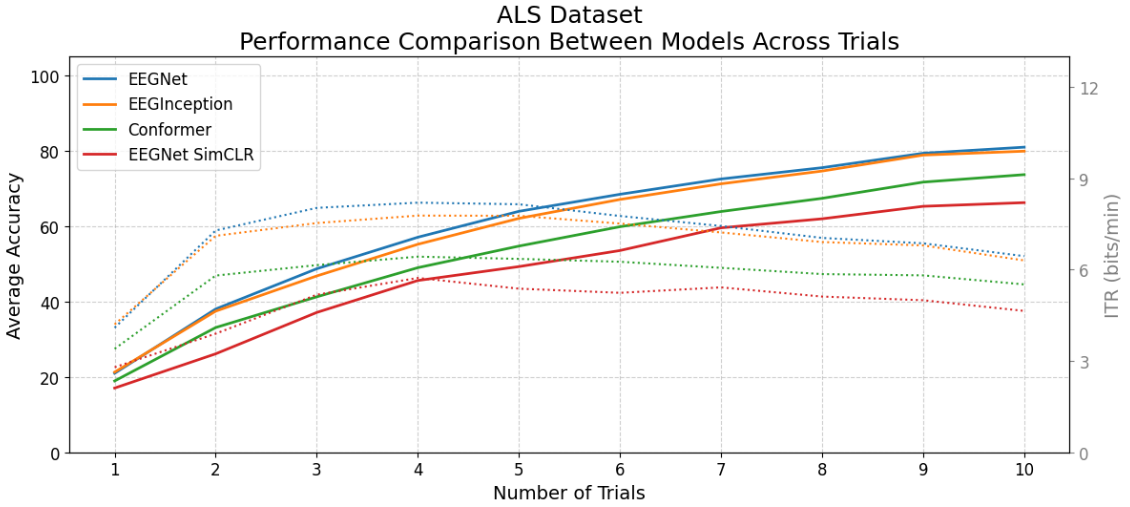


Figure 5.9: ALS Dataset: Performance Comparisons between EEGNet SimCLR and Baseline Models. Dotted lines indicate ITR values.

the one reported for the BCI Comp Dataset), since this information was not provided by the authors of the ALS Dataset. Consequently, these ITR values should be interpreted solely for comparative purposes within this study and should not be extrapolated beyond this context.

Key Insights from the Cross-Dataset Evaluation: Considering the challenges of a cross-dataset evaluation, several key insights can be drawn from the evaluation study presented here:

- **SimCLR Superiority**: With the exception of EEG-Inception, SimCLR consistently outperformed both SupCon and Supervised strategies across the BCI Comp and ALS Datasets.

- **Higher Performance from EEGNet**: EEGNet delivered the best overall results for both datasets. On the BCI Comp Dataset, it achieved approximately 83% accuracy with SimCLR and SupCon using 15 trials and 25 characters for retraining (around 30% of the total training set). On the ALS Dataset, EEGNet pre-trained with SimCLR achieved an accuracy of 66.23% with 10 trials, setting the highest accuracy in this evaluation.

- **EEG-Inception Limitations**: EEG-Inception faced significant challenges in generalizing from pre-training data to evaluation data for both the BCI Comp and ALS Datasets. This evaluation further confirmed the findings from the Intra-Dataset Evaluation, showing that contrastive learning techniques do not benefit this architecture. Among the three models, only EEG-Inception performed worse with contrastive learning strategies compared to Supervised. The consistently low performance across all pre-training strategies suggests that the issue lies with the network architecture rather than the choice of pre-training method.

- **MD Subjects Challenges**: As observed in the Intra-Dataset Evaluation, model performance on MD subjects' signals (ALS Dataset) was lower than with control subjects. However, caution is needed when interpreting these results, as the ALS Dataset Evaluation was conducted in a fully subject-independent paradigm, as some of these lower performance might also be attributed to the shift from a subject-dependent to a subject-independent approach.

5.4 Discussion

Given the results and key insights from both evaluation studies, a crucial question arises: Why did SimCLR perform worse in the Intra-Dataset Evaluation compared to the other strategies but showed the best performance in the Cross-Dataset Evaluation, especially for EEGNet?

A plausible explanation lies in the amount of data used during the pre-training stages. As shown in Table 4.4, the Intra-Dataset Evaluation involved pre-training with a smaller dataset since data from 14 subjects were reserved for evaluation and four for validation, leaving data from 55 subjects for pre-training. In contrast, the Cross-Dataset Evaluation excluded only the data from four validation subjects, resulting in pre-training data coming from 69 subjects. This increase in available pre-training data might be an explanation for the improved performance of SimCLR relative to the other strategies, particularly for EEGNet.

To test this hypothesis, we used the EEGNet SimCLR model from the Intra-Dataset Evaluation (pre-trained with data from 55 subjects), which we refer to as EEGNet SimCLR Intra, in the Cross-Dataset Evaluation with the BCI Comp Dataset, under the same conditions as the EEGNet SimCLR model used for this evaluation (EEGNet SimCLR Cross). The results, presented in Figure 5.10, indicate that for most values of N and number of trials, the differences between the two models, pre-trained with varying amounts of data, are not statistically significant. In the few cases where significant differences were observed, they were only marginal. These findings suggest that the difference in the amount of pre-training data does not significantly impact model performance.

A closer analysis of the results suggests that the observed difference may not be caused from an absolute improvement in SimCLR's performance but rather from a more pronounced decrease in performance by the other strategies.

To illustrate this, let us consider the results for the EEGNet SimCLR model in the Intra-Dataset Evaluation for control subjects (Table A.3) and the BCI Comp Evaluation (Table A.4) when $N = 10$ and using 15 trials. The reported accuracies are 84.72% and 78.22%, respectively, indicating a decrease of 6.5 percentage points from Intra to Cross-Dataset Evaluation. By comparison, EEGNet SupCon exhibits a decrease of 20.44 percentage points, while EEGNet Supervised shows an even steeper decline of 36.04 percentage points.

Extending this analysis to the Conformer model reveals a similar trend: a decrease of 35.64 percentage points for SimCLR, 53.9 percentage points for SupCon, and a substantial drop of 69.01 percentage points for Supervised. This pattern highlights that the relative improvement of SimCLR may result from a greater resilience to performance degradation in cross-dataset evaluations, rather than a performance gain.

Comparing the results of EEGNet from the Intra-Dataset Evaluation for MD subjects (Table A.2) with $N = 10$ and the ALS Dataset Evaluation (Table A.5), both using the maximum number

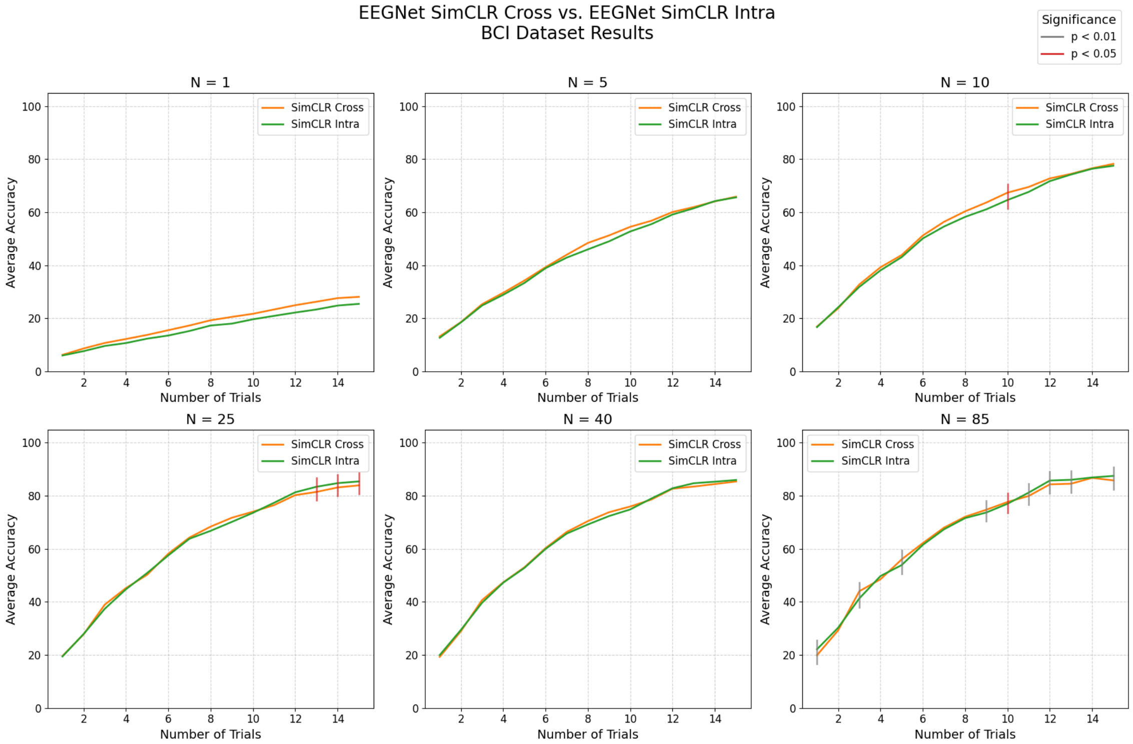


Figure 5.10: EEGNet SimCLR Cross vs. EEGNet SimCLR Intra Results on the BCI Dataset. Performance results reflect the averaging of 50 iterations of the training and testing process. Statistical significance is indicated with red bars ($p < 0.05$) or gray bars ($p < 0.01$).

of trials, we observe that for EEGNet SimCLR, there is an increase of 20.05 percentage points from Intra to Cross-Dataset Evaluations. In contrast, EEGNet SupCon shows a decrease of 11.24 percentage points and EEGNet Supervised drops by 29.14 percentage points. On the Conformer network, SimCLR experiences a decrease of 11.67 percentage points, while SupCon and Supervised show larger declines of 53.53 and 55.72 percentage points, respectively.

However, these findings should be interpreted cautiously, as the Intra-Dataset Evaluation was conducted under a subject-dependent paradigm, whereas the ALS Dataset Evaluation followed a subject-independent paradigm. Additionally, the maximum number of trials differs between the evaluations, with 15 trials in the Intra-Dataset Evaluation and 10 in the ALS Dataset Evaluation.

The follow-up question arising from these results is why SupCon and Supervised strategies experience such significant performance degradation in a cross-dataset evaluation compared to SimCLR. There are two key differences between SimCLR and the other two strategies. The primary distinction is that SimCLR does not use labels during pre-training, whereas both SupCon and Supervised rely on them. The second difference stems from this label availability: with access to labels, the pre-training dataset was balanced for both SupCon and Supervised strategies, whereas SimCLR, lacking label information, cannot leverage such balancing.

In theory, balancing the training dataset should enhance performance on testing data, not decrease it. This approach is widely reported in studies on P300 classification, where training set balancing is a common practice [26, 20].

There are several potential explanations for this, the most plausible one being dataset shifts. Dataset shift refers to a situation where the distribution of data in the training set (source domain) differs from the distribution of data in the test set (target domain). This discrepancy can significantly affect the generalization ability of networks. By incorporating labels during the pre-training stage, Supervised and SupCon strategies might develop higher model confidence on dataset-specific characteristics. This leads to an enhanced ability to represent ”similar” intra-dataset data but limits their ability to generalize to data from a different dataset.

Another possibility is label corruption. As highlighted by Cecotti et al. [7], P300 datasets may be affected by inaccurate labeling. While the experimental paradigm provides an indication of when a P300 response is expected, this response is highly dependent on the user. During the P300 Speller task, participants are asked to focus on a specific character, but various factors, such as distractions or misunderstandings, can interfere with this process. As a result, some signal epochs may be incorrectly labeled as targets, even though no P300 response was actually elicited. However, given that Supervised and SupCon methods yielded better results relying on these labels in the Intra-Dataset Evaluation compared to SimCLR, this explanation seems unlikely.

5.4.1 Research Questions

With the results presented and discussed, we are now able to answer the four Research Questions we set out to answer at the beginning of this project:

RQ1: Can contrastive learning techniques enhance the performance of neural networks in P300 speller systems? Yes, contrastive learning techniques demonstrated their potential to enhance model performance in specific scenarios across both evaluation studies. In the Intra-Dataset Evaluation, SupCon began to demonstrate slightly improved performance compared to Supervised when N was set to 5, and 10. However, the true strength of contrastive learning techniques became evident in the Cross-Dataset Evaluation, where they consistently outperformed the supervised approach, highlighting their ability to generalize more effectively across datasets, especially SimCLR.

RQ2: Can contrastive learning techniques improve subject-independent classification or reduce the calibration data needed for effective performance? In the context of the subject-independent paradigm, our study used the ALS Dataset for evaluation. While our best model (EEGNet SimCLR) did not achieve the same performance levels as the baseline models (Figure 5.9), when using the strategy of leveraging larger datasets, incorporating both CS and MD subjects through a pre-training approach, contrastive learning methods demonstrated promising results, achieving higher performance for both the EEGNet and Conformer architectures as illustrated in Figure 5.8.

To answer the question related to reducing the calibration data needed we are dependent on the type of evaluation conducted. In our evaluations, the value of N represents the calibration phase. During the Intra-Dataset Evaluation, the Supervised approach showed slightly better performance than models pre-trained with contrastive learning when $N = 1$. However when N increased to 5 and 10 SupCon achieved better performances. It is important to remember that all these values of N are well bellow the acceptable time of 20 minutes for a calibration phase [34].

In the Cross-Dataset Evaluation, particularly in the BCI Comp Dataset Evaluation, the results tell a different story. Excluding EEG-Inception due to its poor performance, the other models exhibited improved performance across all values of N. Therefore, this results are optimistic about the benefits of contrastive learning in reducing the calibration data needed for BCI systems.

RQ3: How do different contrastive learning frameworks (SimCLR, and SupCon) compare in addressing the challenges of P300 classification? Our findings indicate that SimCLR performs better in cross-dataset evaluations, while SupCon demonstrates improved performance and greater advantages in intra-dataset evaluations.

RQ4: How do networks pre-trained with contrastive learning compare with state-of-the-art approaches in P300 classification? Figure 5.7 compares the performance of our pre-trained models with that of the baseline models. We found that EEGNet SimCLR performed similarly to its fully subject-dependent baseline counterpart, achieving also comparable ITR values. In contrast, the other architectures exhibited a significant drop in performance compared to their baselines.

Table 5.1 compares these results with those reported in the literature. Direct comparison with EEGNet SimCLR is challenging due to differences in the number of electrodes used. However, by assuming that the baseline model would yield similar results to those reported in the literature using the same number of electrodes, we can indirectly position EEGNet SimCLR alongside state-of-the-art models.

5.4.2 Implications

Robust generalization across different datasets is crucial, not only for ensuring more reliable signal representations across datasets with varying data distributions, but also for enhancing the ability to leverage data from diverse sources. This includes dealing with not only the natural variability across EEG data from different subjects, but also the variance introduced by differences in experimental protocols, signal quality discrepancies due to the use of different recording devices, variations in preprocessing methods, and more. A network that generalizes well under these conditions, along with a pre-training strategy that enables the extraction of more meaningful features from the data, expands the potential dataset size available for pre-training. This increased data availability should result in a further enhancement of the model's generalization capacity for downstream tasks, eliminating the need to rely solely on a single dataset for pre-training.

5.5 Summary

In this chapter, we presented the results of both evaluation studies. In the Intra-Dataset Evaluation, supervised learning emerged as the most effective pre-training strategy when $N = 1$. As N increased, SupCon began to match and eventually surpassed Supervised in performance, although the improvement was marginal. In contrast, networks pre-trained with SimCLR consistently demonstrated the poorest performance in this evaluation.

The Cross-Dataset Evaluation, however, revealed a different outcome. SimCLR emerged as the best-performing pre-training strategy, except for EEG-Inception, which struggled in this evaluation. Of the three models, EEG-Inception was the only one negatively affected by contrastive learning.

Given EEG-Inception's consistently poor performance in Cross-Dataset Evaluations with all pre-training methods, it is reasonable to attribute these results primarily to the model's architecture rather than the pre-training strategy. Focusing on EEGNet and Conformer, the results suggest that the performance differences between the two evaluation studies were not due to a significant improvement in the performance of networks pre-trained with SimCLR. Instead, the primary cause was the substantial performance degradation of models pre-trained with the other strategies.

These findings indicate that SimCLR is a more robust pre-training strategy for cross-dataset evaluations, particularly evident with the EEGNet architecture, which showed the greatest benefit from contrastive learning techniques.

Chapter 6

Conclusion and Future Work

BCI systems provide a vital communication pathway for individuals with motor disabilities, such as ALS patients. These systems utilize EEG signals to generate commands that control external devices. This work focused on P300 speller systems, which employ the row-column paradigm (RCP) to spell characters through the classification of EEG signals. A core requirement for the practicality of these systems is the development of robust models capable of reliably detecting the P300 component.

However, P300 speller systems face two significant challenges. The first is the substantial variability in EEG signals, not only across different users but also within the same user over time. The second challenge is the dependence on multiple trials to achieve satisfactory accuracy in spelling characters. Together, these factors contribute to the subject-dependent nature of most P300 spellers, which rely on user-specific data for optimal performance. This dependency often leads to lengthy calibration sessions, reducing the practicality of these systems. Furthermore, the reliance on multiple trials slows down the overall spelling process. To address these challenges, many approaches employ lightweight machine learning algorithms or neural networks that require minimal data for training.

In contrast, recent advancements in fields like computer vision and natural language processing have demonstrated the potential of contrastive learning techniques to extract meaningful features from large pre-training datasets. These approaches enable networks to develop strong representational capabilities, reducing the need for extensive data during fine-tuning for specific downstream tasks.

Inspired by these advancements, we explored the impact of contrastive learning techniques on the P300 classification task, hypothesizing that these methods might help networks identify cross-subject invariant features. Such capabilities could mitigate calibration time requirements, enhancing the practicality of P300 speller systems.

To investigate this, we selected three state-of-the-art neural network architectures, EEGNet and EEG-Inception, designed specifically for P300 classification, along with Conformer, a model originally developed for other EEG classification tasks but adapted here for P300 classification. These networks were pre-trained using three strategies: SimCLR, SupCon, and Supervised.

To facilitate training with contrastive learning, we introduced a novel augmentation technique called Progressive Evokeds. This method creates positive pairs by pairing signal epochs with their evoked counterparts, enhancing the model's ability to learn meaningful relationships within the data.

The pre-trained networks were evaluated through two studies. In the Intra-Dataset Evaluation, Supervised and SupCon strategies achieved the best performance, with Supervised displaying slightly better results for the smallest value of N (calibration characters) and SupCon showing marginally better performance at higher values of N. In this setup, SimCLR underperformed compared to the other strategies.

The Cross-Dataset Evaluation yielded different results. Here, SimCLR emerged as the most robust pre-training strategy, with the exception of the EEG-Inception network, which performed poorly across this evaluation. These findings indicate that contrastive learning techniques, especially SimCLR, are particularly suited for cross-dataset scenarios, where the ability to generalize across varied data distributions is critical, highlighting its potential to address the challenges of variability and calibration dependency in P300 speller systems.

6.1 Contributions and Limitations

To the best of our knowledge, no studies in the existing literature have conducted a comparative analysis between contrastive learning techniques and supervised approaches for pre-training neural networks within the P300 classification paradigm. While there are other works using contrastive learning for EEG classification, they either fall outside the scope of the P300 paradigm or include it as part of a broader context. For instance, [19] developed a foundational model aimed at generalizing across various EEG classification paradigms, including P300 classification. However, such studies do not focus exclusively on the P300 paradigm nor provide a direct comparison between contrastive learning and supervised approaches.

Recognizing this gap, our work aimed at providing preliminary insights into the impact and potential benefits of contrastive learning within the P300 classification paradigm. Additionally, it introduced a novel data augmentation technique, Progressive Evokeds, which proved effective in generating positive pairs and significantly contributed to the success of contrastive learning techniques observed in this work. We also described network architecture's modifications that could provide guidelines for designing new networks tailored to the specific requirements of contrastive learning approaches.

As expected, there are numerous areas, not addressed in this work, that deserve further exploration, including hyperparameter optimization, analysis of additional network architectures, testing on more datasets, and overall conducting more comprehensive studies to validate the findings presented here.

6.2 Future Work

Understanding the limitations of this study, the following points outline the logical next steps for future research:

- **Evaluating the Effect of Contrastive Learning on Larger Models:** As highlighted in [9], larger networks benefit more from self-supervised learning. The SimCLR [8] and SupCon [17] approaches used ResNet-50 and even larger versions of this architecture. In general, contrastive learning-based models tend to employ much larger architectures compared to those studied in this work. For instance, EEGNet has 7.4 thousand parameters, EEG-Inception has 34 thousand, and Conformer has 458 thousand parameters, while ResNet-50 has approximately 25.6 million parameters. The significantly fewer parameters in models used for P300 classification arise from their subject-dependent nature, which requires fewer parameters to minimize calibration time. However, based on the results presented here, data from multiple sources could be used to pre-train a larger model capable of learning more robust representations, thereby reducing the amount of data needed for fine-tuning when adapting to a new subject.

- **Hyperparameters and Training Process Exploration:** Increasing the training time (epochs) could be beneficial, as contrastive learning techniques are known to improve with longer training durations, as mentioned in [8]. Additionally, this study used the standard temperature value of 0.1, which is commonly used in many works. However, experimenting with different temperature values in the NT-BXent and SupCon losses could help optimize the performance of these processes.

  Another aspect worth investigating is the size of the representations output by the projection head. This study used the commonly reported value of 128, but exploring other sizes could potentially lead to different performance outcomes.

- **Deeper Analysis of Model Feature Extraction:** Understanding what networks focus on when classifying different signals can provide valuable insights into their functioning and help deepen our understanding of the signals themselves. One effective approach is to examine the network's receptive fields, which can offer a more detailed understanding of how the network processes and interprets the data.

- **Develop and Test with Novel Augmentations:** Augmentation techniques are essential in contrastive learning. While this study introduced a new augmentation method, no comparative analysis with other techniques was performed. Exploring new ways to generate variations within data samples could enhance contrastive learning methods and improve model performance.

Bibliography

[1] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, and Cuntai Guan. Filter bank common spatial pattern (fbcsp) in brain-computer interface. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pages 2390–2397. IEEE, 2008.

[2] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460. Curran Associates, Inc., 2020.

[3] Sylvain Baillet, John C Mosher, and Richard M Leahy. Electromagnetic brain mapping. *IEEE Signal processing magazine*, 18(6):14–30, 2001.

[4] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Riemannian geometry applied to bci classification. In Vincent Vigneron, Vicente Zarzoso, Eric Moreau, Rémi Gribonval, and Emmanuel Vincent, editors, *Latent Variable Analysis and Signal Separation*, pages 629–636, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[5] Alexandre Barachant and Marco Congedo. A plugplay p300 bci using information geometry. 8 2014.

[6] Quentin Barthélemy, Sylvain Chevallier, Raphaëlle Bertrand-Lalo, and Pierre Clisson. End-to-end p300 bci using bayesian accumulation of riemannian probabilities. *Brain-Computer Interfaces*, 10:50–61, 2023.

[7] Hubert Cecotti and Axel Gräser. Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:433–445, 2011.

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. 2 2020.

[9] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 22243–22255. Curran Associates, Inc., 2020.

[10] Mohammad Norizadeh Cherloo, Amir Mohammad Mijani, Liang Zhan, and Mohammad Reza Daliri. A novel multiclass-based framework for p300 detection in bci matrix speller: Temporal eeg patterns of non-target trials vary based on their position to previous target stimuli. *Engineering Applications of Artificial Intelligence*, 123, 8 2023.

[11] Yuntian Cui, Xinke Shen, Dan Zhang, and Chen Yang. A contrastive learning based convolutional neural network for erp brain-computer interfaces. *arXiv preprint arXiv:2407.04738*, 2024.

[12] L.A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70:510–523, 12 1988.

[13] Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum. A cookbook of self-supervised learning. *arXiv preprint arXiv:2304.12210*, 2023.

[14] He He and Dongrui Wu. Transfer learning for brain-computer interfaces: A euclidean space data alignment approach. *IEEE Transactions on Biomedical Engineering*, 67:399–410, 2 2020.

[15] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning, 3 2021.

[16] Raviraj Joshi, Purvi Goel, Mriganka Sur, and Hema A. Murthy. Single trial p300 classification using convolutional lstm and deep learning ensembles method. volume 11278 LNCS, pages 3–15. Springer Verlag, 2018.

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Google Research, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning.

[18] Sepideh Kilani, Seyedeh Nadia Aghili, and Mircea Hulea. Enhancing p300-based brain-computer interfaces with hybrid transfer learning: A data alignment and fine-tuning approach. *Applied Sciences (Switzerland)*, 13, 5 2023.

[19] Demetres Kostas, Stephane Aroca-Ouellette, and Frank Rudzicz. Bendr: Using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data. *Frontiers in Human Neuroscience*, 15:653659, 2021.

[20] Sourav Kundu and Samit Ari. P300 based character recognition using sparse autoencoder with ensemble of svms. *Biocybernetics and Biomedical Engineering*, 39:956–966, 10 2019.

[21] Vernon J. Lawhern, Amelia J. Solon, Nicholas R. Waytowich, Stephen M. Gordon, Chou P. Hung, and Brent J. Lance. Eegnet: A compact convolutional neural network for eeg-based brain-computer interfaces. *Journal of Neural Engineering*, 15, 7 2018.

[22] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *Ieee Access*, 8:193907–193934, 2020.

[23] Jongmin Lee, Kyungho Won, Moonyoung Kwon, Sung Chan Jun, and Minkyu Ahn. Cnn with large data achieves true zero-training in online p300 brain-computer interface. *IEEE Access*, 8:74385–74400, 2020.

[24] Steven Lemm, Benjamin Blankertz, Gabriel Curio, and K-R Muller. Spatio-spectral filters for improving the classification of single trial eeg. *IEEE transactions on biomedical engineering*, 52(9):1541–1548, 2005.

[25] Wenjie Li, Haoyu Li, Xinlin Sun, Huicong Kang, Shan An, Guoxin Wang, and Zhongke Gao. Self-supervised contrastive learning for eeg-based cross-subject motor imagery recognition. *Journal of Neural Engineering*, 21(2):026038, 2024.

[26] Mingfei Liu, Wei Wu, Zhenghui Gu, Zhuliang Yu, Fei Fei Qi, and Yuanqing Li. Deep learning based on batch normalization for p300 signal detection. *Neurocomputing*, 275:288–297, 1 2018.

[27] Dhruv Matani. Nt-xent (normalized temperature-scaled cross-entropy) loss explained and implemented in pytorch, 2023. https://towardsdatascience.com/nt-xent-normalized-temperature-scaled-cross-entropy-loss-explained-and-implemented-in-pytorch-cc081f69848f004 - Accessed: 2024-12-12.

[28] Johannes Müller-Gerking, Gert Pfurtscheller, and Henrik Flyvbjerg. Designing optimal spatial filters for single-trial eeg classification in a movement task. *Clinical neurophysiology*, 110(5):787–798, 1999.

[29] Arnaldo E Pereira, Dereck Padden, Jay J Jantz, Kate Lin, and Ramses E Alcaide-Aguirre. Cross-subject eeg event-related potential classification for brain-computer interfaces using residual networks.

[30] John Polich. Updating p300: An integrative theory of p3a and p3b, 10 2007.

[31] Alain Rakotomamonjy and Vincent Guigue. Bci competition iii: Dataset ii- ensemble of svms for bci p300 speller. *IEEE Transactions on Biomedical Engineering*, 55:1147–1154, 3 2008.

[32] Angela Riccio, Luca Simione, Francesca Schettini, Alessia Pizzimenti, Maurizio Inghilleri, Marta Olivetti Belardinelli, Donatella Mattia, and Febo Cincotti. Attention and p300-based bci performance in people with amyotrophic lateral sclerosis. *Frontiers in Human Neuroscience*, 7, 2013.

[33] Bertrand Rivet, Antoine Souloumiac, Virginie Attina, and Guillaume Gibert. xdawn algorithm to enhance evoked potentials: Application to brain-computer interface. *IEEE Transactions on Biomedical Engineering*, 56:2035–2043, 2009.

[34] Eduardo Santamaria-Vazquez, Victor Martinez-Cagigal, Fernando Vaquerizo-Villar, and Roberto Hornero. Eeg-inception: A novel deep convolutional neural network for assistive erp-based brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28:2773–2782, 12 2020.

[35] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[36] C.E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949.

[37] Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. Eeg conformer: Convolutional transformer for eeg decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2022.

[38] Shravani Sur and Vinod Kumar Sinha. Event-related potential: An overview. *Industrial psychiatry journal*, 18(1):70–73, 2009.

[39] Samuel Sutton, Margery Braren, Joseph Zubin, and ER John. Evoked-potential correlates of stimulus uncertainty. *Science*, 150(3700):1187–1188, 1965.

[40] Manoj Thulasidas, Cuntai Guan, and Jiankang Wu. Robust classification of eeg signal for brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14:24–29, 3 2006.

[41] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in neural information processing systems*, 33:6827–6839, 2020.

[42] Arjon Turnip and Keum-Shik Hong. Classifying mental activities from eeg-p300 signals using adaptive neural networks, 2012.

[43] Xiaolin Xiao, Minpeng Xu, Jing Jin, Yijun Wang, Tzyy Ping Jung, and Dong Ming. Discriminative canonical pattern matching for single-trial classification of erp components. *IEEE Transactions on Biomedical Engineering*, 67:2266–2275, 8 2020.

[44] Burak Yildirim, Osman Ulkir, Mahmut Kaya, Anupreet Kaur Singh, and Sridhar Krishnan. Trends in eeg signal feature extraction applications.

Appendix A

Tables of Evaluation Results

A.1 Intra-Dataset Evaluation Results: Average of both CS and MD subjects

Table A.1
Intra-Dataset Evaluation Results (Subject Dependent)
Average Character Recognition Accuracy for the 14 Subjects

N	Network	Pre-Training	Number of Trials				Diff
			1	5	10	15	
1	EEGNet	Supervised	16.59 ± 10.57	40.03 ± 21.63	52.15 ± 25.72	58.89 ± 27.6	-
		SimCLR	7.63 ± 4.15	14.92 ± 11.02	20.81 ± 15.84	25.26 ± 18.88	-26.97
		SupCon	16.4 ± 9.73	39.5 ± 22.25	51.31 ± 26.6	57.89 ± 28.25	-1.15
	EEGInception	Supervised	18.9 ± 10.38	46.44 ± 21.41	58.36 ± 24.02	66.01 ± 25.65	-
		SimCLR	6.38 ± 2.13	13.04 ± 7.66	18.12 ± 12.32	21.74 ± 16.09	-35.19
		SupCon	15.7 ± 7.9	38.01 ± 20.42	50.42 ± 26.02	56.94 ± 28.39	-7.89
	EEG-Conformer	Supervised	16.81 ± 8.9	42.7 ± 21.68	53.87 ± 25.78	60.86 ± 27.7	-
		SimCLR	6.89 ± 4.05	13.55 ± 11.07	18.69 ± 16.19	22.44 ± 19.69	-30.49
		SupCon	16.58 ± 8.66	39.94 ± 21.73	51.71 ± 25.76	58.25 ± 28.03	-2.37
-							
5	EEGNet	Supervised	22.67 ± 11.02	57.59 ± 20.33	71.95 ± 22.79	78.75 ± 21.69	-
		SimCLR	14.33 ± 7.8	36.76 ± 20.37	50.72 ± 25.71	58.12 ± 27.2	-19.66
		SupCon	23.82 ± 10.76	59.32 ± 21.37	73.05 ± 23.28	79.52 ± 21.55	+1.18
	EEGInception	Supervised	25.08 ± 10.5	60.99 ± 18.73	73.92 ± 20.14	81.88 ± 18.55	-
		SimCLR	11.92 ± 5.65	32.0 ± 18.5	44.92 ± 26.54	51.57 ± 29.26	-27.41
		SupCon	26.15 ± 11.15	61.08 ± 23.29	73.94 ± 25.58	79.21 ± 24.4	-0.65
	EEG-Conformer	Supervised	24.55 ± 11.27	60.82 ± 21.67	74.03 ± 23.42	80.08 ± 22.02	-
		SimCLR	13.64 ± 7.38	36.38 ± 21.34	49.84 ± 27.78	56.8 ± 30.36	-22.54
		SupCon	24.5 ± 10.56	59.92 ± 22.06	73.16 ± 24.12	78.97 ± 23.31	-1.12
-							
10	EEGNet	Supervised	25.2 ± 10.67	62.7 ± 19.38	76.73 ± 20.77	82.34 ± 19.11	-
		SimCLR	18.0 ± 8.45	45.86 ± 22.09	58.9 ± 26.49	65.45 ± 26.58	-16.07
		SupCon	26.86 ± 11.01	64.6 ± 19.83	77.72 ± 20.69	83.3 ± 18.63	+1.07
	EEGInception	Supervised	27.1 ± 10.38	64.47 ± 18.33	77.11 ± 19.14	84.69 ± 17.1	-
		SimCLR	13.84 ± 6.26	39.64 ± 19.0	53.42 ± 24.6	60.53 ± 26.42	-22.98
		SupCon	30.59 ± 12.23	67.47 ± 21.85	79.35 ± 21.84	83.92 ± 19.88	+2.74
	EEG-Conformer	Supervised	27.68 ± 11.71	65.37 ± 20.48	77.53 ± 20.9	83.69 ± 18.91	-
		SimCLR	17.66 ± 8.01	46.92 ± 22.79	59.33 ± 27.99	65.51 ± 28.84	-17.36
		SupCon	27.79 ± 11.53	65.69 ± 20.53	77.68 ± 20.97	83.3 ± 19.31	+0.31

The displayed values represent the averages from 50 iterations of the retraining and testing process conducted across all subjects, with varying seeds and retraining characters. The ”Diff” values indicate the average difference, calculated exclusively from the trials that show statistically significant differences between SimCLR and Supervised or SupCon and Supervised.

A.2 Intra-Dataset Evaluation Results for Motor-Disabled Subjects

Table A.2
Intra-Dataset Evaluation Results (Motor-Disabled Subjects)
Average Character Recognition Accuracy for the 7 Subjects

N	**Network**	**Pre-Training**	**Number of Trials**				**Diff**
			1	**5**	**10**	**15**	
1	EEGNet	Supervised	18.67 ± 12.98	37.56 ± 26.32	45.1 ± 29.93	49.25 ± 31.67	-
		SimCLR	8.54 ± 5.45	15.08 ± 14.41	18.44 ± 19.7	20.6 ± 22.13	-23.60
		SupCon	17.54 ± 11.92	34.95 ± 25.1	41.86 ± 28.67	46.77 ± 30.14	-2.89
	EEGInception	Supervised	20.67 ± 12.68	43.13 ± 26.92	49.84 ± 27.93	55.83 ± 30.3	-
		SimCLR	6.5 ± 2.06	10.67 ± 5.66	13.11 ± 8.64	14.49 ± 10.72	-33.08
		SupCon	15.34 ± 9.22	32.04 ± 22.8	39.64 ± 27.77	44.33 ± 30.76	-10.27
	EEG-Conformer	Supervised	17.8 ± 10.73	36.37 ± 25.28	42.2 ± 27.79	47.09 ± 29.74	-
		SimCLR	8.1 ± 5.1	14.22 ± 14.05	17.83 ± 19.84	20.29 ± 23.4	-22.09
		SupCon	16.92 ± 10.26	34.35 ± 25.56	41.06 ± 28.4	45.32 ± 30.88	-2.15
-							
5	EEGNet	Supervised	24.33 ± 13.75	51.48 ± 25.45	61.06 ± 26.81	67.17 ± 25.03	-
		SimCLR	14.4 ± 10.14	29.63 ± 22.68	38.81 ± 26.61	43.97 ± 27.62	-20.69
		SupCon	24.07 ± 13.1	51.49 ± 24.22	61.73 ± 26.72	68.14 ± 24.51	+0.17
	EEGInception	Supervised	26.26 ± 12.3	55.41 ± 23.17	64.54 ± 23.27	72.55 ± 21.36	-
		SimCLR	11.53 ± 6.05	25.05 ± 17.1	34.54 ± 25.22	38.99 ± 28.31	-28.79
		SupCon	23.62 ± 11.99	50.25 ± 25.83	60.11 ± 28.65	65.3 ± 27.32	-5.08
	EEG-Conformer	Supervised	25.06 ± 13.33	52.34 ± 25.66	61.48 ± 26.44	67.49 ± 24.76	-
		SimCLR	14.62 ± 9.46	31.05 ± 24.18	39.84 ± 29.91	44.09 ± 31.34	-20.29
		SupCon	24.42 ± 13.1	50.79 ± 26.19	60.0 ± 27.36	65.15 ± 25.86	-1.80
-							
10	EEGNet	Supervised	26.27 ± 13.17	54.49 ± 23.1	64.91 ± 23.16	70.66 ± 20.87	-
		SimCLR	16.75 ± 10.57	34.93 ± 21.82	41.35 ± 24.53	46.18 ± 23.81	-20.37
		SupCon	26.2 ± 12.83	54.91 ± 21.56	65.27 ± 22.25	71.19 ± 19.52	+0.04
	EEGInception	Supervised	27.36 ± 11.61	57.59 ± 21.48	67.58 ± 21.76	75.72 ± 19.63	-
		SimCLR	13.31 ± 6.6	30.75 ± 16.96	39.85 ± 20.89	45.44 ± 24.06	-26.15
		SupCon	27.21 ± 12.98	56.22 ± 23.53	66.19 ± 23.42	71.76 ± 21.65	-2.72
	EEG-Conformer	Supervised	27.84 ± 13.7	55.81 ± 22.91	64.67 ± 22.11	71.87 ± 20.48	-
		SimCLR	16.9 ± 9.7	36.4 ± 23.67	41.61 ± 25.18	46.02 ± 25.21	-20.92
		SupCon	27.45 ± 14.03	55.55 ± 23.37	64.47 ± 22.12	70.77 ± 20.47	-1.05

The displayed values represent the averages from 50 iterations of the retraining and testing process conducted across all subjects, with varying seeds and retraining characters. The ”Diff” values indicate the average difference, calculated exclusively from the trials that show statistically significant differences between SimCLR and Supervised or SupCon and Supervised.

A.3 Intra Dataset Evaluation Results for Control Subjects

Table A.3
Intra Dataset Evaluation Results (Control Subjects)
Average Character Recognition Accuracy for the 7 Subjects

N	Network	Pre-Training	Number of Trials				Diff
			1	5	10	15	
1	EEGNet	Supervised	14.51 ± 6.81	42.5 ± 15.18	59.2 ± 18.09	68.54 ± 18.3	-
		SimCLR	6.73 ± 1.78	14.75 ± 5.93	23.18 ± 10.11	29.91 ± 13.4	-30.34
		SupCon	15.27 ± 6.67	44.05 ± 17.84	60.75 ± 20.35	69.01 ± 21.0	-
	EEGInception	Supervised	17.13 ± 6.95	49.75 ± 13.05	66.87 ± 15.11	76.18 ± 13.79	-
		SimCLR	6.26 ± 2.18	15.41 ± 8.61	23.13 ± 13.36	28.99 ± 17.25	-37.31
		SupCon	16.06 ± 6.29	43.99 ± 15.58	61.2 ± 18.72	69.56 ± 18.63	-5.51
	EEG-Conformer	Supervised	15.81 ± 6.44	49.03 ± 14.87	65.54 ± 16.87	74.63 ± 16.45	-
		SimCLR	5.68 ± 1.97	12.88 ± 6.84	19.55 ± 11.36	24.6 ± 14.79	-38.89
		SupCon	16.24 ± 6.68	45.54 ± 15.12	62.36 ± 17.13	71.19 ± 16.83	-2.82
-							
5	EEGNet	Supervised	21.02 ± 6.94	63.71 ± 10.18	82.83 ± 9.1	90.33 ± 6.8	-
		SimCLR	14.26 ± 4.32	43.9 ± 14.6	62.63 ± 18.16	72.28 ± 17.78	-18.62
		SupCon	23.57 ± 7.73	67.15 ± 14.27	84.38 ± 10.66	90.9 ± 8.32	+1.96
	EEGInception	Supervised	23.91 ± 8.14	66.58 ± 10.14	83.29 ± 9.7	91.2 ± 7.63	-
		SimCLR	12.3 ± 5.19	38.96 ± 17.18	55.29 ± 23.62	64.15 ± 24.39	-26.03
		SupCon	28.69 ± 9.59	71.92 ± 13.52	87.78 ± 10.27	93.12 ± 7.57	+4.21
	EEG-Conformer	Supervised	24.05 ± 8.72	69.3 ± 11.7	86.58 ± 9.12	92.68 ± 6.28	-
		SimCLR	12.65 ± 4.18	41.71 ± 16.41	59.85 ± 21.19	69.51 ± 23.2	-24.79
		SupCon	24.58 ± 7.18	69.04 ± 10.97	86.32 ± 8.29	92.78 ± 6.0	-0.58
-							
10	EEGNet	Supervised	24.13 ± 7.22	70.91 ± 9.08	88.55 ± 6.86	94.02 ± 4.74	-
		SimCLR	19.25 ± 5.29	56.79 ± 16.15	76.45 ± 13.63	84.72 ± 10.19	-11.77
		SupCon	27.52 ± 8.78	74.28 ± 11.57	90.18 ± 7.14	95.4 ± 4.45	+2.21
	EEGInception	Supervised	26.85 ± 8.97	71.34 ± 10.79	86.65 ± 8.81	93.66 ± 6.21	-
		SimCLR	14.38 ± 5.84	48.53 ± 16.63	67.0 ± 20.12	75.62 ± 19.02	-19.81
		SupCon	33.97 ± 10.4	78.73 ± 12.16	92.51 ± 7.66	96.09 ± 5.09	+6.01
	EEG-Conformer	Supervised	27.51 ± 9.3	74.93 ± 11.46	90.39 ± 7.35	95.5 ± 4.06	-
		SimCLR	18.43 ± 5.76	57.43 ± 16.06	77.05 ± 17.48	84.99 ± 16.41	-13.80
		SupCon	28.14 ± 8.29	75.83 ± 9.54	90.89 ± 6.38	95.83 ± 3.57	+1.39

The displayed values represent the averages from 50 iterations of the retraining and testing process conducted across all subjects, with varying seeds and retraining characters. The ”Diff” values indicate the average difference, calculated exclusively from the trials that show statistically significant differences between SimCLR and Supervised or SupCon and Supervised.

A.4 BCI Comp Dataset Evaluation Results

Table A.4

BCI Comp Dataset Evaluation Results (Subject Dependent)

Average Character Recognition Accuracy for the 2 Subjects

N	Network	Pre-Training	Number of Trials				Diff
			1	5	10	15	
1	EEGNet	Supervised	3.53 ± 0.27	5.31 ± 0.39	6.97 ± 0.19	8.17 ± 0.83	-
		SimCLR	6.22 ± 1.62	13.74 ± 3.72	21.71 ± 5.21	28.09 ± 4.61	+12.29
		SupCon	4.24 ± 0.26	8.81 ± 0.97	12.74 ± 0.7	16.18 ± 0.3	+4.87
	EEGInception	Supervised	3.54 ± 0.54	4.61 ± 0.13	6.15 ± 0.49	8.13 ± 0.55	-
		SimCLR	3.18 ± 0.0	2.94 ± 0.2	3.58 ± 0.56	4.07 ± 0.67	-2.48
		SupCon	2.65 ± 0.01	3.23 ± 0.41	3.83 ± 0.53	4.21 ± 0.65	-2.16
	Conformer	Supervised	2.93 ± 0.05	3.89 ± 0.29	4.81 ± 0.75	5.65 ± 0.69	-
		SimCLR	4.3 ± 0.56	6.08 ± 1.58	8.02 ± 2.34	9.79 ± 2.75	+2.72
		SupCon	3.06 ± 0.18	4.28 ± 0.28	5.75 ± 0.51	7.23 ± 1.39	+1.24
5	EEGNet	Supervised	6.79 ± 1.01	17.2 ± 0.88	25.81 ± 0.99	34.35 ± 0.75	-
		SimCLR	13.18 ± 4.8	34.24 ± 10.24	54.48 ± 9.98	65.85 ± 6.83	+22.49
		SupCon	10.64 ± 3.7	29.56 ± 8.04	47.12 ± 10.08	59.55 ± 7.91	+16.89
	EEGInception	Supervised	4.3 ± 0.62	8.1 ± 0.34	13.42 ± 2.04	18.86 ± 2.12	-
		SimCLR	3.33 ± 0.29	3.7 ± 0.1	4.35 ± 0.51	5.22 ± 1.1	-7.42
		SupCon	3.47 ± 0.41	5.41 ± 0.21	7.12 ± 0.92	8.74 ± 1.44	-5.18
	Conformer	Supervised	4.38 ± 0.56	8.25 ± 0.31	12.36 ± 0.74	16.58 ± 0.2	-
		SimCLR	8.89 ± 4.03	19.47 ± 7.59	28.87 ± 9.31	36.51 ± 9.63	+13.74
		SupCon	5.44 ± 1.14	11.81 ± 0.09	19.16 ± 0.46	26.34 ± 2.42	+5.14
10	EEGNet	Supervised	9.99 ± 2.45	29.25 ± 3.89	46.29 ± 3.91	57.98 ± 0.58	-
		SimCLR	16.89 ± 6.35	43.85 ± 12.83	67.37 ± 12.41	78.22 ± 7.98	+17.31
		SupCon	13.91 ± 4.81	39.65 ± 10.21	61.45 ± 9.61	74.96 ± 5.58	+12.68
	EEGInception	Supervised	4.84 ± 1.06	12.2 ± 0.52	20.84 ± 1.04	30.07 ± 1.51	-
		SimCLR	4.03 ± 0.21	4.53 ± 0.41	5.85 ± 1.57	7.27 ± 2.19	-12.07
		SupCon	4.41 ± 0.21	7.05 ± 0.49	10.33 ± 1.69	13.59 ± 3.01	-8.99
	Conformer	Supervised	5.09 ± 0.93	12.56 ± 2.08	18.43 ± 0.75	26.49 ± 2.77	-
		SimCLR	11.01 ± 4.79	28.54 ± 10.5	39.1 ± 9.8	49.35 ± 8.49	+17.45
		SupCon	7.16 ± 2.08	17.26 ± 1.0	29.48 ± 0.2	41.93 ± 1.39	+8.42
25	EEGNet	Supervised	14.63 ± 4.27	42.11 ± 8.75	63.87 ± 8.71	76.06 ± 2.9	-
		SimCLR	19.59 ± 6.99	50.22 ± 12.9	74.0 ± 11.18	83.93 ± 6.85	+9.22
		SupCon	18.55 ± 5.77	47.5 ± 11.54	71.43 ± 8.15	83.05 ± 5.19	+6.37
	EEGInception	Supervised	5.49 ± 1.59	16.54 ± 0.22	28.99 ± 0.21	41.75 ± 0.37	-
		SimCLR	4.68 ± 0.02	6.16 ± 1.36	8.18 ± 3.02	11.49 ± 4.87	-16.24
		SupCon	5.79 ± 0.87	11.16 ± 0.86	16.42 ± 1.0	23.26 ± 2.34	-10.03
	Conformer	Supervised	6.44 ± 1.36	18.39 ± 3.49	27.63 ± 2.35	40.03 ± 4.41	-
		SimCLR	13.62 ± 4.9	38.1 ± 10.08	53.69 ± 6.43	63.71 ± 2.25	+21.00
		SupCon	10.12 ± 3.68	25.37 ± 2.99	42.19 ± 0.63	56.94 ± 0.16	+11.16
40	EEGNet	Supervised	15.9 ± 4.8	46.15 ± 9.77	69.02 ± 8.46	80.24 ± 3.06	-
		SimCLR	19.28 ± 6.88	53.03 ± 12.67	75.95 ± 10.71	85.39 ± 6.21	+7.23
		SupCon	19.61 ± 6.83	50.07 ± 11.55	73.59 ± 7.51	84.84 ± 4.92	+4.67
	EEGInception	Supervised	5.96 ± 1.74	18.48 ± 1.08	33.0 ± 0.02	45.83 ± 1.55	-
		SimCLR	4.66 ± 0.36	7.5 ± 1.32	9.81 ± 3.29	14.28 ± 5.06	-17.52
		SupCon	6.65 ± 0.81	14.11 ± 0.89	20.92 ± 0.38	30.77 ± 1.03	-8.72
	Conformer	Supervised	6.74 ± 1.94	20.44 ± 4.66	31.31 ± 3.49	45.06 ± 4.66	-
		SimCLR	14.48 ± 4.1	41.09 ± 9.71	57.11 ± 4.77	67.16 ± 0.88	+20.93
		SupCon	11.55 ± 5.19	27.47 ± 3.89	45.43 ± 0.99	59.63 ± 0.73	+10.88
85	EEGNet	Supervised	18.16 ± 4.8	50.47 ± 11.59	72.67 ± 9.69	83.03 ± 3.49	-
		SimCLR	19.91 ± 5.69	56.01 ± 11.93	77.66 ± 11.46	85.8 ± 6.14	+4.81
		SupCon	20.42 ± 7.64	53.0 ± 12.7	75.66 ± 7.36	87.03 ± 5.67	+2.92
	EEGInception	Supervised	6.3 ± 2.72	20.4 ± 2.26	36.06 ± 0.58	49.75 ± 2.73	-
		SimCLR	6.31 ± 0.91	8.91 ± 2.01	11.2 ± 3.98	17.01 ± 5.27	-19.86
		SupCon	7.9 ± 0.78	16.4 ± 1.66	24.93 ± 0.33	37.18 ± 0.26	-7.68
	Conformer	Supervised	6.71 ± 2.11	22.71 ± 5.97	32.97 ± 4.19	48.21 ± 7.03	-
		SimCLR	16.18 ± 3.86	43.17 ± 8.31	59.29 ± 3.75	69.09 ± 0.27	+20.69
		SupCon	12.62 ± 6.04	28.86 ± 5.02	48.02 ± 0.62	61.99 ± 0.51	+10.56

A.5 ALS Dataset Evaluation Results

Table A.5
ALS Dataset Evaluation Results (Subject Independent)
Average Character Recognition Accuracy for the 8 Subjects

Network	**Pre-Training**	**Number of Trials**			**Diff**
		1	**5**	**10**	
EEGNet	Supervised	8.28 ± 5.40	29.12 ± 19.18	41.52 ± 22.35	-
	SimCLR	17.04 ± 9.64	49.26 ± 18.48	66.23 ± 16.40	+20.57
	SupCon	16.80 ± 7.93	44.83 ± 21.49	59.95 ± 20.80	+16.58
EEGInception	Supervised	10.51 ± 6.60	15.75 ± 9.88	25.71 ± 17.23	-
	SimCLR	4.15 ± 4.29	14.17 ± 12.13	16.01 ± 16.01	-6.69
	SupCon	5.22 ± 4.60	15.73 ± 11.56	23.77 ± 15.40	-5.40
Conformer	Supervised	4.61 ± 4.03	11.96 ± 9.37	16.15 ± 12.38	-
	SimCLR	6.04 ± 4.89	21.90 ± 12.21	34.35 ± 18.45	+11.30
	SupCon	5.36 ± 4.60	13.68 ± 9.51	17.24 ± 11.20	+1.09

The displayed values represent the averages from 50 iterations of the retraining and testing process conducted across all subjects, with varying seeds. The ”Diff” values indicate the average difference, calculated exclusively from the trials that show statistically significant differences between SimCLR and Supervised or SupCon and Supervised.