ADAPTATION IN P300 AND MOTOR IMAGERY-BASED BCI SYSTEMS

by

İsmail Yılmaz

Submitted to the Graduate School of Engineering and Natural Sciences in partial fulfilment of the requirements for the degree of Master of Science

Sabanci University

September 2015

ADAPTATION IN P300 AND MOTOR IMAGERY-BASED BCI SYSTEMS

APPROVED BY

Assoc. Prof. Dr. Müjdat ÇETİN ....
(Thesis Supervisor)

Assoc. Prof. Dr. Kemal KILIÇ ....

Assoc. Prof. Dr. Tolga TAŞDiZEN ....

DATE OF APPROVAL: ....

...to my big supporter, my mom Esma Yılmaz

Acknowledgments

I want to express my gratitude to my supervisor Müjdat Çetin for his guidance, motivation, suggestions and encouragement throughout my graduate studies. It was a great experience to work with him.

I would like to thank Tolga Taşdizen for his guidance, precious suggestions and support on my thesis study and his participation in the Thesis committee.

I would like to thank Kemal Kılıç for his support on my graduate courses, motivation and his participation in the Thesis committee.

I would also like to thank Sümeyra Demir for her guidance, patience, non-stop helps and support for my Master thesis.

I am also thankful to TÜBİTAK for providing the financial support for my graduate education[1].

My special thanks for Ozan Özdenizci and Sezen Yağmur Günay for sharing their experience with me, for their non-stop helps and continous support during my graduate.

I owe special thanks to my family for their unconditional love and support.

[1]This work was partially supported by the Scientific and Technological Research Council of Turkey under Grant 111E056, and by Sabanci University under Grant IACF-11-00889.

ADAPTATION IN P300 AND MOTOR IMAGERY-BASED BCI SYSTEMS

İsmail Yılmaz

EE, M.Sc. Thesis, 2015

Thesis Supervisor: Müjdat Çetin

Keywords: BCI, Brain Computer Interfaces, EEG, adaptivity, HMM, language models, P300, BLDA, semisupervised learning, motor imagery movements, sensory motor rhytms, ErrP

Abstract

Brain Computer Interface (BCI) is an alternative communication tool between human and computer. Motivation of BCI is to create a non-muscular communication environment for the use of external devices. Electroencephalography (EEG) signals are analyzed for understanding the user's intent in BCI systems. The non-stationary behavior of brain electrical activity (such as EEG), caused by changes in subject brain activities, environment conditions and calibration issues, is one of the main challenges of BCI systems. Another set of challenges involves limited amount of training data and subject-dependent characteristics of EEG. In this thesis, we suggest a semi-supervised adaptation approach for P300 based BCI speller systems to address these types of problems. The proposed approach is applied on a P300 speller which also incorporates a language model using Hidden Markov Models (HMM). The estimated labels from the classifier are used to retrain the classifier for adaptation. We have analyzed the effects of this adaptation approach on BCI systems with non-stationary EEG data and small size of training data. We propose to solve both problems by updating the BCI system with labels obtained from the classifier. We have shown that such an adaptation approach would improve BCI performance around 30% for systems with limited amount of training data, and 40% for transferring the system subject-to-subject. Moreover, we have investigated the potential use of error related potential (ErrP) signals in the P300-based BCI systems. The detection and classification of ErrP signals in BCI setting are presented

along with the experimental analysis of ErrP.

BEYİN-BİLGİSAYAR ARAYÜZÜ TABANLI P300 VE HAYALİ MOTOR HAREKETLERİNDE UYARLAMA

İsmail Yılmaz

EE, Yüksek Lisans Tezi, 2015

Tez danışmanı: Müjdat Çetin

Anahtar Kelimeler: BBA, beyin-bilgisayar arayüzleri, elektroensefalografi, uyarlama, SMM, dil modelleme, P300, güdümlü ögrenme, hayali motor hareketleri

Özet

Beyin Bilgisayar Arayüzü (BBA), beyin ile bilgisayar arasında yeni bir iletişim kanalıdır. BBA motivasyonu, felç gibi rahatsızlıkları olan hastaların kas hareketlerini kullanmadan iletişim kurabilme ve harici çevresel cihazları kullanabilme fikrini taşır. Elektroensefalografi (EEG) sinyalleri, kullanıcının isteğini anlama amacıyla analiz edilir. Kullanıcıların beyin aktivitelerindeki değişikliklerden, çevresel koşullardan ve kalibrasyonla ilgili sebepler nedeniyle beyin elektrik aktivitelerinin durağan olmaması, BBA sistemlerinin temel sorunlarından biridir. Eğitim verisinin sınırlı miktarda olması ve farklı kullanıcıların eğitim ile test verilerinin uyumsuzluğu da BBA sistemlerindeki diğer problemler arasında gösterilebilir. Bu tezde, bu tip sorunları çözmek üzere P300 heceleme sistemine yarı-güdümlü uyarlama metodu kullanımını öneriyoruz. Bu uyarlama yaklaşımını, bir dil modeli ile eğitilmiş Saklı Markov Modelini (SMM) içeren BBA tabanlı bir P300 heceleme sistemine uyguladık. Sınıflandırıcıdan alınan tahmini etiketlerle sınıflandırıcı tekrar eğitilerek sistem uyarlanır. Bu uyarlama yöntemini, hayali motor hareketleri tabanlı BBA sistemlerinde karşılaşılan durağan olmayan veri probleminin çözümü için de kullandık. Bu tez çalışmada bahsedilen problemleri, yarı güdümlü öğrenme yöntemi ile BBA sistemini kullandığımız sınıflandırıcının çıktısı olan etiketlerle güncelleyerek çözmeyi öneriyoruz. Buna ek olarak hata ile ilgili potansiyellerin (ErrP) P300 sis-

temlerinde adaptasyon amaçlı kullanımını araştırdık. ErrP sinyallerinin tespiti ve sınıflandırılması ile deneyler ve çalışmalar yaptık. Yaptığımız deneylerde, az eğitim verisi olan BBA performansının yaklaşık 30%, bir kişide eğitilen sınıflandırıcının başkasında kullanıldığı BBA performansının yaklaşık 40% oranında arttığını gösterdik.

Table of Contents

Acknowledgments **v**

Abstract **vi**

Özet **viii**

1 Introduction **1**
1.1 Scope and Motivation . . . 2
1.2 Contributions . . . 6
1.3 Outline . . . 6

2 Background **8**
2.1 Introduction . . . 8
2.2 EEG Signals . . . 9
2.2.1 Electrodes . . . 9
2.3 BCI Systems . . . 13
2.3.1 Sensorimotor Rhyhtms . . . 14
2.3.2 Event Related Potentials and P300 . . . 16
2.3.3 ErrP signals . . . 21
2.3.4 Classification methods . . . 23
2.4 Adaptation on BCI . . . 25
2.5 Language Model . . . 26

3 Adaptation in a Motor Imagery-based BCI System **28**
3.1 Processing BCI data set . . . 28
3.1.1 Feature Extraction . . . 31
3.1.2 Classification . . . 32
3.2 Adaptation . . . 34
3.2.1 Adaptation Algorithm . . . 35
3.2.2 Results . . . 37

4 Adaptation in a Language Model-based P300 Speller **42**
4.1 Terminology . . . 42
4.2 Methods . . . 43
4.2.1 Classification and Language Models . . . 43

4.2.2 Adaptation Algorithm . . . 44
4.3 Experiments . . . 45
4.3.1 Data . . . 45
4.3.2 Results . . . 46
4.4 ErrP based BCI system . . . 55
4.4.1 Processing ErrP dataset . . . 56
4.4.2 ErrP Classification . . . 58
4.4.3 Experiment and results . . . 59
4.4.4 Implementation of an ErrP classification to a P300 speller system . . . 60

5 Conclusions and Future Work 63
5.1 Conclusions . . . 63
5.2 Future work . . . 64

A Language Model 68
A.1 Forward-Backward Algorithm . . . 68
A.2 Viterbi Algorithm . . . 70

Bibliography 70

List of Figures

1.1 An example of BCI system. 2

1.2 BCI based robotic experiments. 2

1.3 A BCI based motor imagery system. 5

1.4 A BCI based P300 speller. 5

2.1 64 channel electrode cap for international 10-20 electrode distribution. 10

2.2 Conductive gel tubes used to reduce skin resistance. 10

2.3 Active electrode set. 11

2.4 Active electrode set. Left image shows Flat-Type electrodes whereas right image shows Pin-Type electrodes. 12

2.5 Electrode placement layout according to 10-20 electrode system. Courtesy of [1]. 12

2.6 Applying electrode gel in holes and clicking active electrodes into holders (Electrodes and holders are color labelled). 13

2.7 A typical BCI system model. 14

2.8 Left figure represents ERD and right figure represents ERS processing. A decrease of band power indicates ERD and an increase of band power ERS. Courtesy of [2]. 16

2.9 P300 signal is collected from Cz channel. Courtesy of [3]. 17

2.10 First P300 speller paradigm used by Donchin. 18

2.11 Hex-o-Spell interface developed by Blankertz. 19

2.12 RSVP interface. 19

2.13 BCI2000 interface. 20

2.14 SU-BCI P300 Speller. 21

2.15 ErrP signal collected from FCz electrode. 22

3.1 Location of the C3, Cz, C4 electrodes. 29

3.2 Description of Data set 1 stimuli. 30
3.3 Description of Data set 2 stimuli. 31
3.4 The accuracies for four different classifiers with Data set 1. 33
3.5 Adaptivity algorithm scheme. 35
3.6 Adaptivity algorithm scheme, when train data is not extended. 37
3.7 LDA classifier results with limited training data. 38
3.8 SVM classifier results with limited training data. 39

4.1 A sequential HMM. 43
4.2 Impact of adaptation as a function of the training data size for the BLDA classifier. 48
4.3 Impact of adaptation as a function of the training data size for the language model based classifier with the forward algorithm. 49
4.4 Impact of adaptation as a function of the training data size for the language model based classifier with the forward backward algorithm. 50
4.5 Impact of adaptation as a function of the training data size for the language model based classifier with the Viterbi algorithm. 51
4.6 Accuracy as a function of the amount of data used for adaptation. . . 54
4.7 Averages of error and correct trial. 55
4.8 The target character “S”. 57
4.9 Matrix flashes. 57
4.10 Feedback character “S”. 57
4.11 6x6 Matrix is showed to user. 61
4.12 Feedback character. 62

List of Tables

2.1 Properties of Brain Activity Measurement (Courtesy of [4]). 9

3.1 Adaptation results of time varying signals (subject s4). 40

3.2 Adaptation results of time varying signals (subject x11). 40

4.1 Subject-to-subject adaptation results. 52

4.2 Individual results for subject-to-subject adaptation where training data is from user U1 and test data is from user U2. 52

4.3 Individual results for subject-to-subject adaptation where training data is from user U3 and test data is from user U4. 52

4.4 Individual results for subject-to-subject adaptation where training data is from user U5 and test data is from user U6. 53

4.5 Results of ErrP experiment when Pe=0.5. 60

4.6 Results of ErrP experiment when Pe=0.2. 60

Chapter 1

Introduction

There are many ways people can communicate, with each other and/or with their surroundings [5], [6]. Talking, gestures, and writing are some of these communication methods. A Brain Computer Interface (BCI) provides a new communication way between a user and a computer. Disorders and conditions, such as stroke and amyotrophic lateral sclerosis (ALS) limit communication with the environment; or completely eliminate it, when motor nerve cells located in the central nervous system and in areas such as spinal cord are impaired. BCI systems apply the idea of creating a non-muscular communication channel between brain and computer to patients with such motor neuron disorders [7].

There are various invasive and non-invasive techniques to monitor brain signals. Non-invasive technologies such as functional magnetic resonance imaging (fMRI), magnetoencephalography (MEG), positron emission tomography (PET), functional near infrared spectroscopy (fNIRS) and electroencephalography (EEG) are commonly used for BCI applications [8], [9], [10].

Among these methods, the most widely applied technique to acquire brain signals is, recording EEG from the scalp via electrodes. Collecting EEG signals is inexpensive, safe, and practical. With this non-invasive approach, EEG signals are recorded using an electrode cap, and classified to interpret the users needs [11]. Current BCIs allow users to control external devices [12], [13] and cursor movements [14], [15], and to select letters or icons from computer screen [16], [17]. An example of BCI system can be seen in Fig. 1.1, and Fig. 1.2 illustrates example of BCI based robotic experiments.

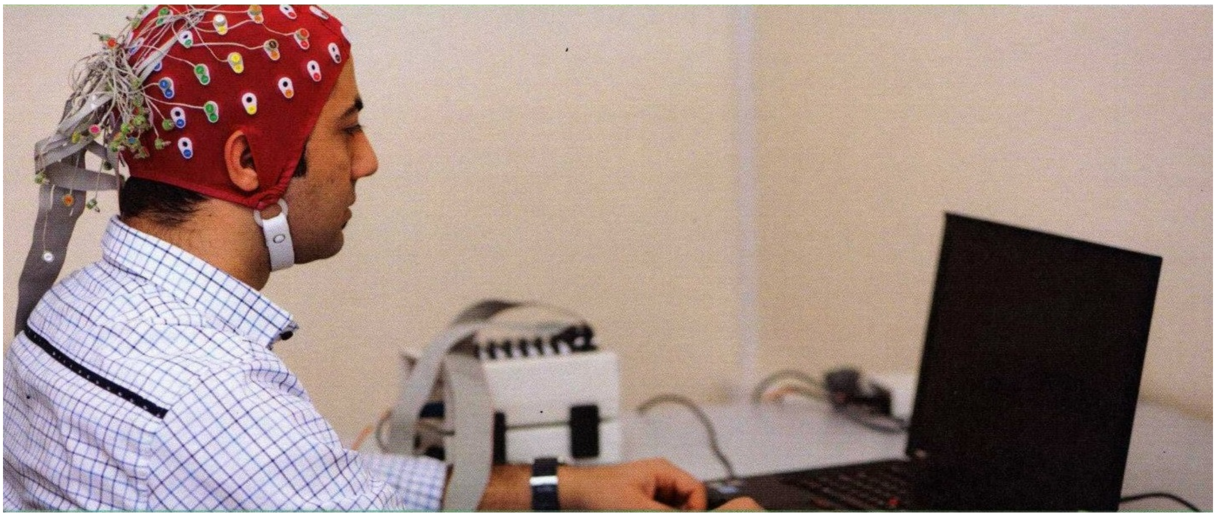

Figure 1.1: An example of BCI system.

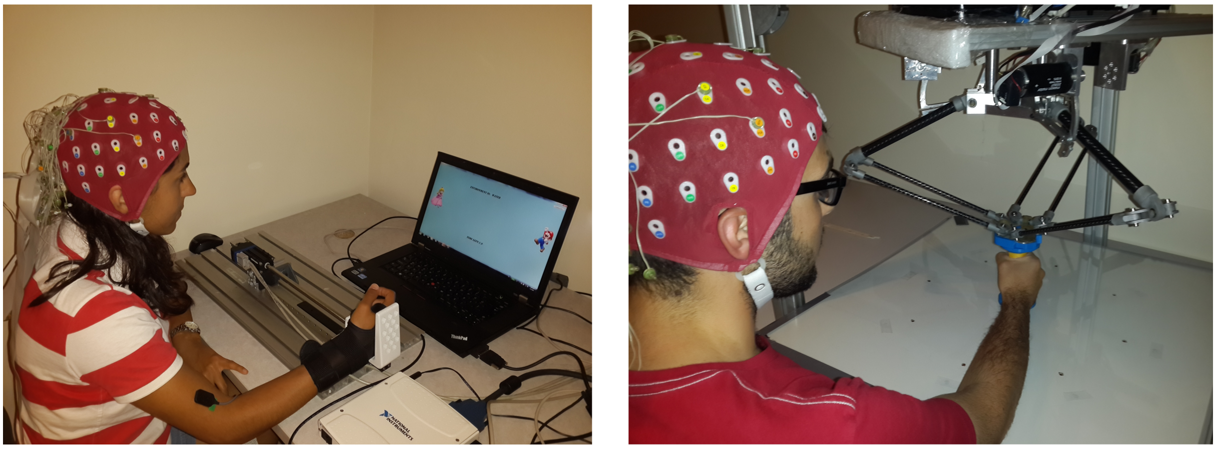

Figure 1.2: BCI based robotic experiments.

1.1 Scope and Motivation

In this thesis, we focus on three particular EEG potentials (P300, sensorimotor rhythms and error related potentials (ErrP)), and semi-supervised adaptation of BCI systems.

The widely used P300 speller systems enable users to spell text on a computer screen by processing and classifying EEG signals. The P300 signal is a specific type of event related potential (ERP). The ERP, an unconscious stereotyped brain potential, contains series of negative and positive wave components. These positive and negative waves are recognized by their polarity and time of occurrence. P300 signal is named after a positive displacement of an event related potential (ERP) occurring after the presentation of the visual or auditory stimulus. Sometimes, P300 component can be called P3 component because of the design of component serial order. P3 means that third positive wave after stimulus [18]. Farwell first introduced the P300 speller paradigm in 1988, where a 6x6 matrix is presented on a computer screen [19]. Rows and columns flash randomly, and P300 responses occur when user focuses on a character of interest that is flashing. Pattern recognition and machine learning methods are used to detect P300 signals [20]. An example of BCI based P300 speller experiment can be seen in Fig. 1.4.

Another important and widely used potential for BCI systems is sensorimotor rhythms. Users can control external devices and cursor movements; and activate electronic or mechanical agents using sensorimotor rhythms that are generated by primary sensory cortex and primary motor cortex (PMC). Sensorimotor rhythms show rhythmic EEG oscillations with spectral power energies around 16-24 Hz (beta), 12-16 Hz (sigma), 8-12 Hz (alpha). These spectral power energies show a sequence of attenuation called event-related desynchronization (ERD) and followed by a rebound called even-related synchronization (ERS) [21]. During motor tasks, ERD can be observed as a decrease of band power and ERS can be observed as an increase of band power. ERD and ERS patterns help the classifier to identify motor imagery movement, which is a mental motor process without any motor output [2]. An example of BCI based motor imagery experiment can be seen in Fig. 1.3.

One of the main problem in BCI systems is the non-stationary nature of recorded signals. The characteristics of brain signals may vary after training due to task difference, movement of the electrodes, gel drying, user's level of attention and fatigue; causing the signals to become non-stationary. Because of this non-stationary behavior, a classifier trained on previous EEG data may not be optimal for following sessions [10].

Adaptation allows classifiers to update their parameters according to the incoming data. It is an effective method to improve the classification performance and learn the changing patterns of the data. Supervised approaches need the true labels of the data which may not be always available. An alternative solution is semi-supervised learning approach which uses a small amount of labeled data to adapt with a large amount of unlabeled data [22].

In this thesis, we propose a semi supervised adaptation method on a P300 speller system; which incorporates a language model using Hidden Markov Model (HMM). The classification scores from Bayesian Linear Discriminant Analysis (BLDA) are combined with a language model using forward-backward, forward, or Viterbi algorithms on a Hidden Markov Model (HMM); to determine the letters typed by the users. The language model is a fully probabilistic approach which exploits information from the previous and future letters for the current letter [23]. We have also used proposed adaptation approach on motor imagery BCI data. Alpha, sigma, beta frequency bands are used for feature extraction. These bands are analyzed by applying Short Time Fourier Transform to each trial. We have also explored the potential use of ErrP signals to improve BCI systems' performance. ErrP signal is a specific type of ERP that contains early negative deflection (error negativity) and positive deflection (error positivity). It is generated by anterior cingulate cortex, where error negativity and error positivity are observed on frontal central regions and paretial regions, respectively [24].

Researchers aim to improve BCI systems to make them faster, robust, and more accurate. For these reasons, researchers try to reduce training time [25], non-stationaries [26]. Some adaptation methods are applied to P300 BCI systems [27]. ErrP signal is used to improve system performance [28]. BCI inclined to make errors in the detection of the user's intent [29], using ErrP can help the system to be more accurate.

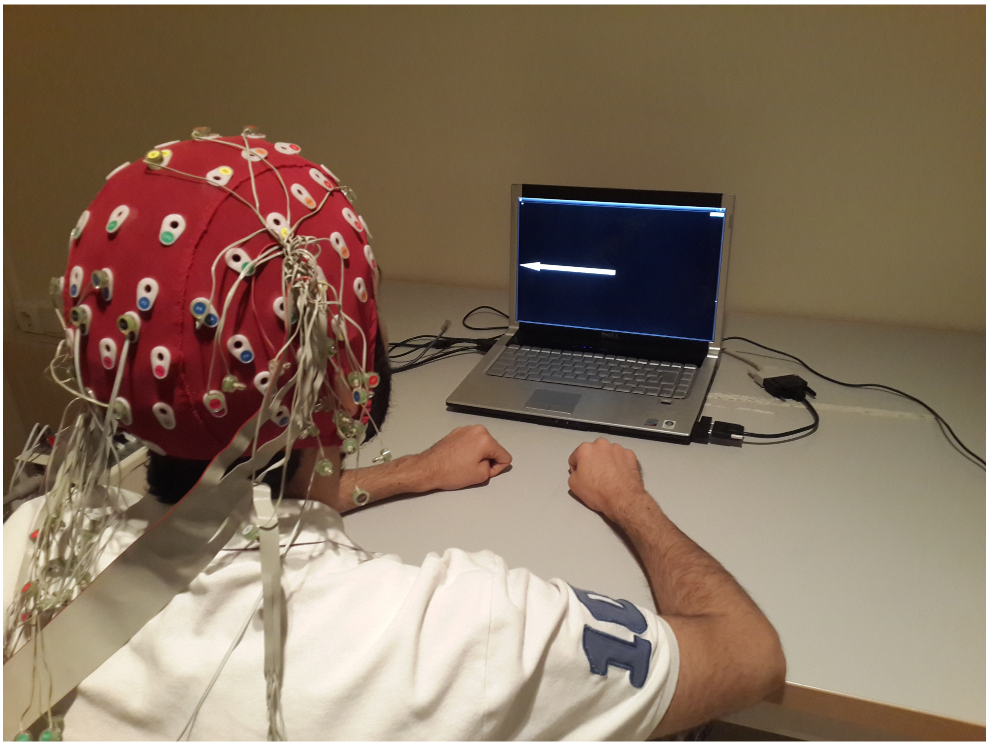

Figure 1.3: A BCI based motor imagery system.

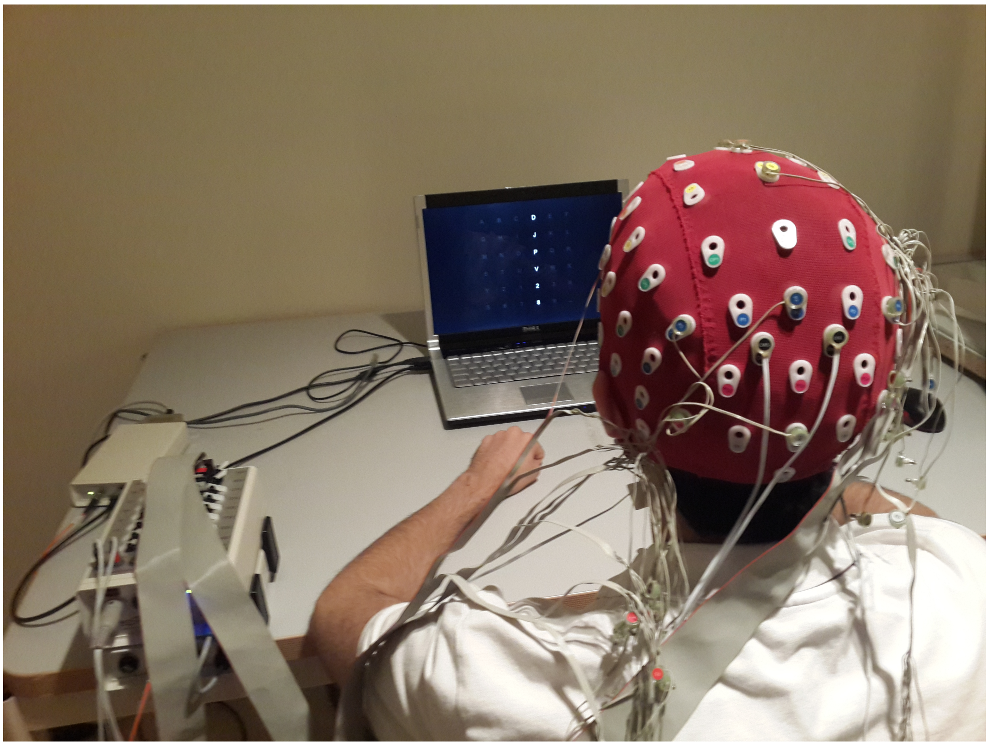

Figure 1.4: A BCI based P300 speller.

1.2 Contributions

We propose and evaluate a semi-supervised learning approach for adaptation (updating the classifier) of a language model-based P300 spelling system. To the best of our knowledge, adaptation of this particular type of classifier for BCI has not been studied before. Our experimental analysis is aimed at determining two extreme points of performance for various training and test data sizes. First we evaluate adaptation using the true labels of the test data, which would of course not be available in practice, to get a sense of the "upper bound" of performance one can hope to get through adaptation. On the other extreme, we perform adaptation using simply the labels generated by the existing classifier. This is a trivial adaptation approach used here to evaluate how much such a simple idea could improve performance. A practical adaptation approach that exploits some further aspects of the test data (than just the labels generated by the existing classifier) would be expected to generate an accuracy that lies in between the two extremes produced by our analysis.

In this sense, an important contribution of our work is this analysis of how much one should expect to improve the performance of a language model-based BCI system through adaptation. We also apply the same type of analysis on a motor imagery system. Our experiments involve adaptation in several different settings: limited training data, subject-to-subject transfer and changing patterns of EEG signals over time.

1.3 Outline

The rest of the thesis is organized as follows:

Chapter 2 presents background information on EEG signals, BCI systems, classification methods and adaptation on brain-computer interfaces by presenting a survey about published works and language models.

Chapter 3 covers the proposed adaptation method for a motor imagery based BCI system. We present our motor imagery experiments, feature extraction and classification methods. We describe the adaptation method and our experimental results.

In Chapter 4, we present adaptation of a language model based P300 speller. Classification of P300 signals, using language models and adaptation algorithm are introduced. Performance metrics and results of our experiments can be found in this chapter. We have also introduced ErrP signals and our ErrP stimulus system, and we have analyzed ErrP signals and results of ErrP experiments.

Chapter 5 provides a summary of the contributions and the results of this thesis, and suggests several potential future research directions.

Chapter 2

Background

This chapter provides the basic concepts of EEG signals processing, BCI, P300 signals and P300 spellers, motor imagery systems, classification methods and adaptation theory. It also includes a survey of published work, methods and results.

2.1 Introduction

In today's world, the number of people with spinal cord impairment, has increased as a result of vehicle accidents, injuries, stroke, diseases such as Amyotrophic Lateral Sclerosis (ALS) or some other trauma. Spinal cord impairment limits people's communication with the environment. Over the last two decades, studies have shown that, brain signals collected from the scalp can provide a new communication pathway without the requirement of muscle controls. In this Brain Computer Interface (BCI) systems, specific signals are collected, processed and classified to predict the user's intent [30].

Invasive and non-invasive methods have been used in BCI systems to record brain signals. Using invasive technologies, such as electro-corticography (ECoG), electrodes are placed under the scalp during an operation. This approach provides high resolution signals but is rather expensive and might involve risks for the patient. On the other hand, non-invasive methods allow the electrodes to be placed directly on the scalp. The signal resolution is lower with such technique with respect to invasive methods, but it does not require an operation and is easier to implement. Even though some non-invasive technologies such as magnetoencephalography (MEG), positron emission tomography (PET) and functional magnetic resonance imaging (fMRI) provide a good spatial resolution; they are expensive, their temporal resolution is low and/or it may not be appropriate to utilize them outside the laboratories, in daily life [11]. In contrast, EEG is inexpensive, more practical and can be transferred into daily life. Considering all these facts, EEG is a common approach to record brain signals for BCI systems. In Table 2.1, properties of brain activity acquisition methods are listed.

	EEG	MEG	fMRI	ECoG
Deployment	Noninvasive	Noninvasive	Noninvasive	Invasive
Measured activity	Electrical	Magnetic	Hemodynamic	Electrical
Temporal resolution	Good	Good	Low	High
Spatial resolution	Low	Low	Good	Good
Portability	High	Low	Low	High
Cost	Low	High	High	High

Table 2.1: Properties of Brain Activity Measurement (Courtesy of [4]).

2.2 EEG Signals

Current field in the brain is caused by flow of cations of calcium (Ca), potassium (K), sodium (Na) and anions of chlorine (Cl) between neural cell membranes. This current causes potential changes between neurons. An EEG signal measures the voltage difference between neurons in the cerebral cortex. This signal can be measured from scalp using specific electrodes designed for EEG. Hans Berger first introduced the presence of human EEG signals in 1929 [31]. The EEG signals are mainly used to gather information about neurological disorders.

To record the signals, the electrodes are placed on the surface of the scalp. After signals are detected (5-20 micro voltage range) by electrodes, amplifiers are used to magnify the signals. Analog to digital converters convert the EEG signal to a digital format and lastly, signals are recorded using record devices.

2.2.1 Electrodes

The electrode cap contains the electrodes of EEG, little small disks of Ag/AgCl, and it is placed to the subject's scalp. An example of an electrode cap is illustrated in Fig. 2.1.

Figure 2.1: 64 channel electrode cap for international 10-20 electrode distribution.

Electrodes don't interact directly with skin. Therefore, a conductive gel (Fig. 2.2) is applied to the skin to reduce skin resistance or voltage offset, and to have a stable, stationary conductive medium for proper measurements [32].

Figure 2.2: Conductive gel tubes used to reduce skin resistance.

The active electrodes mostly used in BCI systems are shown in Fig. 2.3 and Fig. 2.4. Active electrodes propose high resistance to interface, long term DC stability and also prevent high electrode impedances and cable shielding [33]. Electrode cap is placed to the scalp of subject as shown in the Fig. 2.6 according to an international system called 10-20 system, proposed by American EEG society [34]. In 10-20 system, electrodes are located in 10 % - 20% distance from each other. This distance is determined with respect to the total distance between the nasion and inion of the subject. The layout of the 64 channel EEG system that we use in our own recordings is presented in Fig. 2.5.

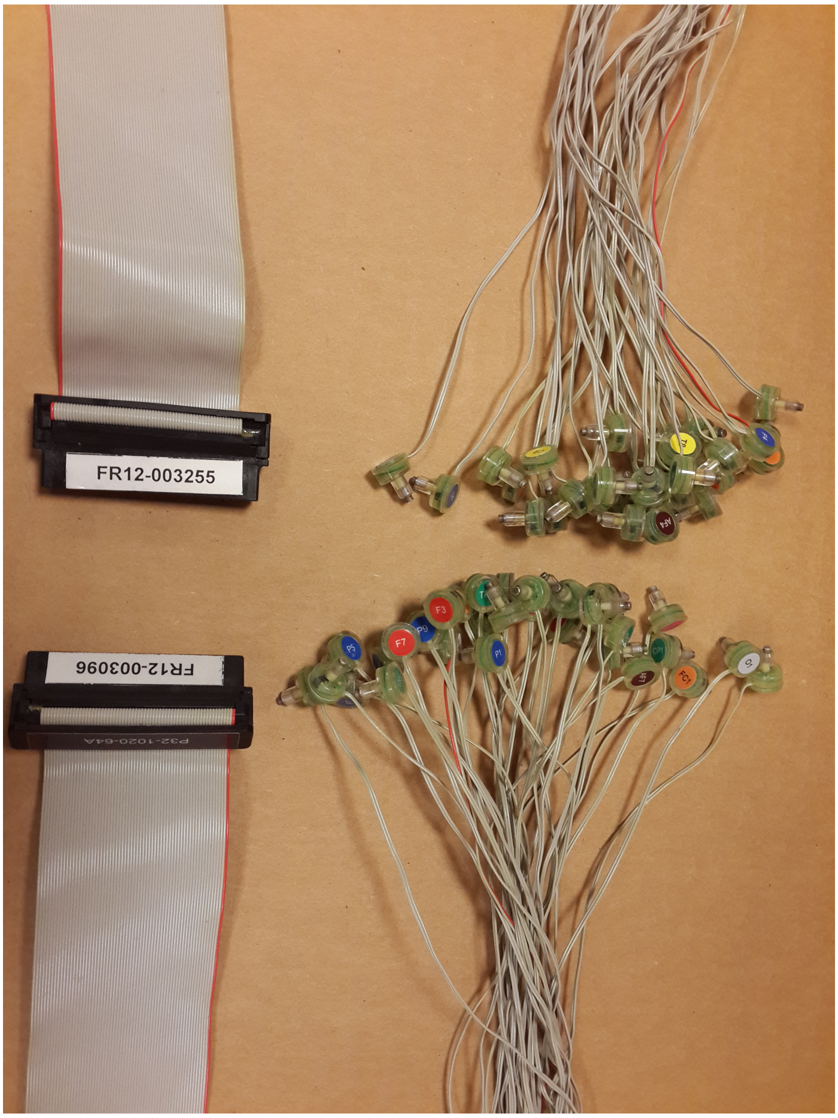

Figure 2.3: Active electrode set.

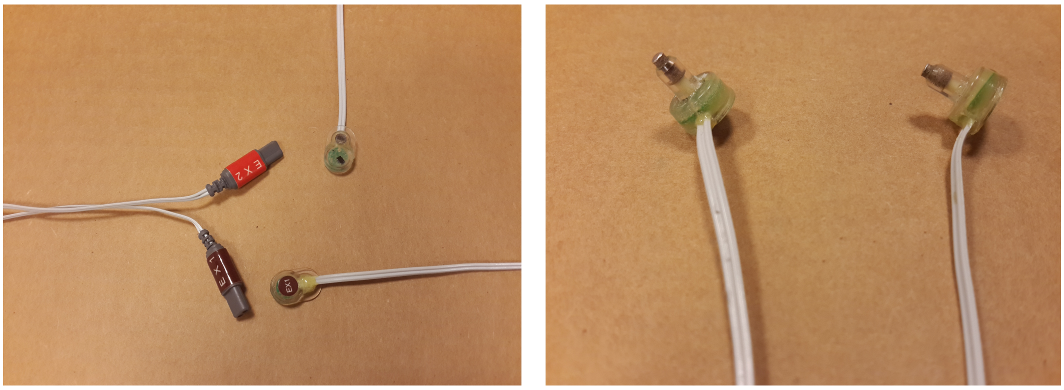

Figure 2.4: Active electrode set. Left image shows Flat-Type electrodes whereas right image shows Pin-Type electrodes.

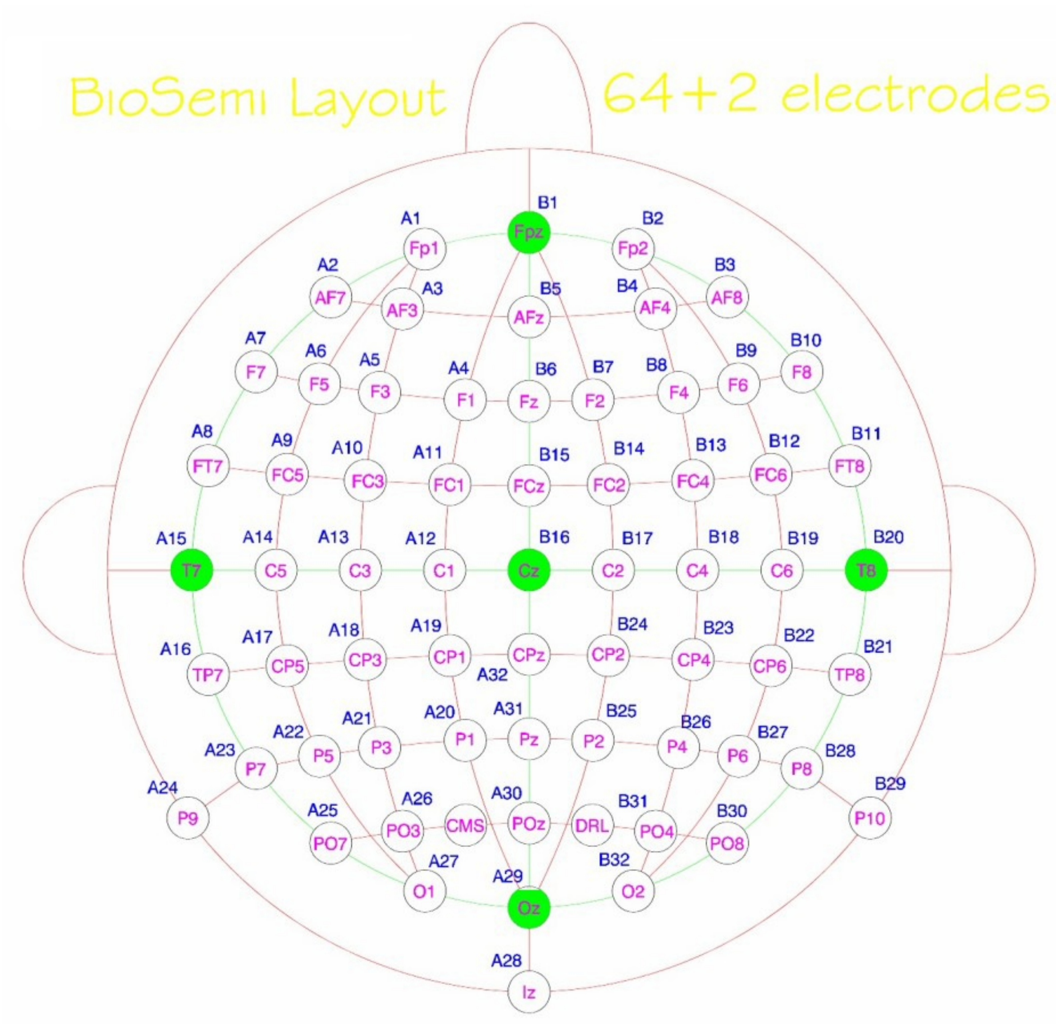

Figure 2.5: Electrode placement layout according to 10-20 electrode system. Courtesy of [1].

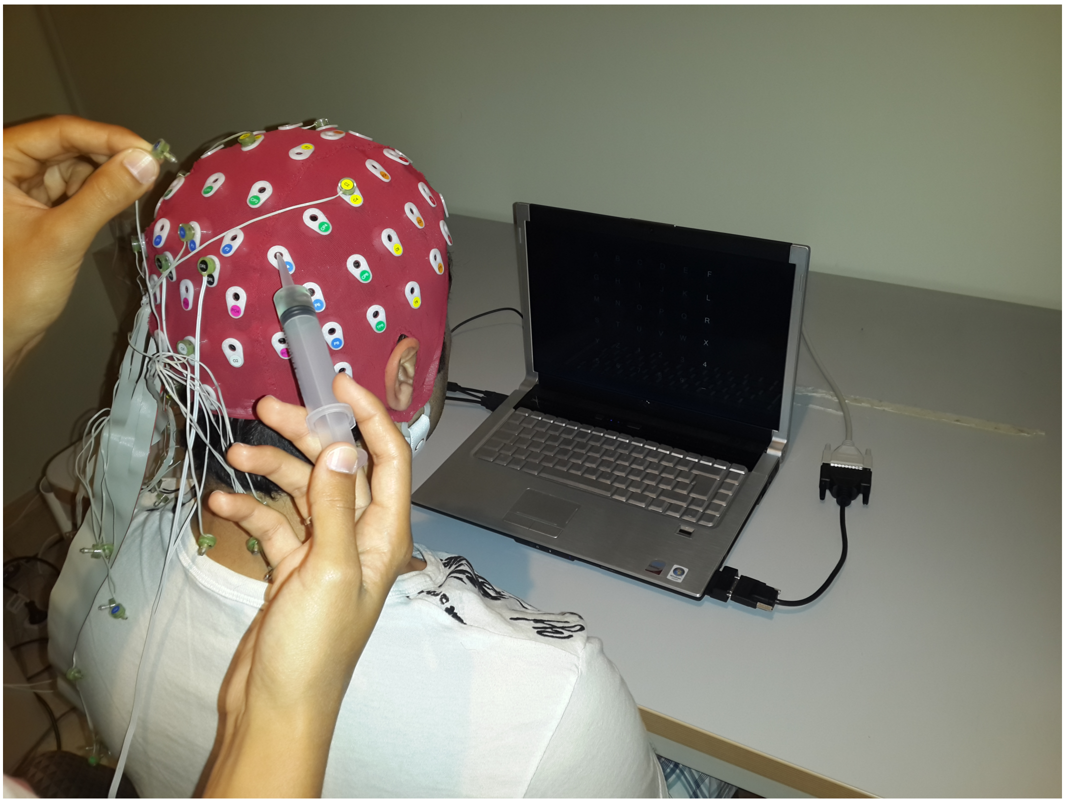

Figure 2.6: Applying electrode gel in holes and clicking active electrodes into holders (Electrodes and holders are color labelled).

2.3 BCI Systems

During daily activities such as thinking, moving, or even feeling, neurons are active as well. The neurons produce electrical signals to carry physiological and pathological information [35]. Sensing neural activity is a method of observing signals. Analysis of the observed signals can provide control over some external devices. This communication line was first introduced by Vidal in 1970's [36]. A typical BCI system model can be seen in Fig. 2.7.

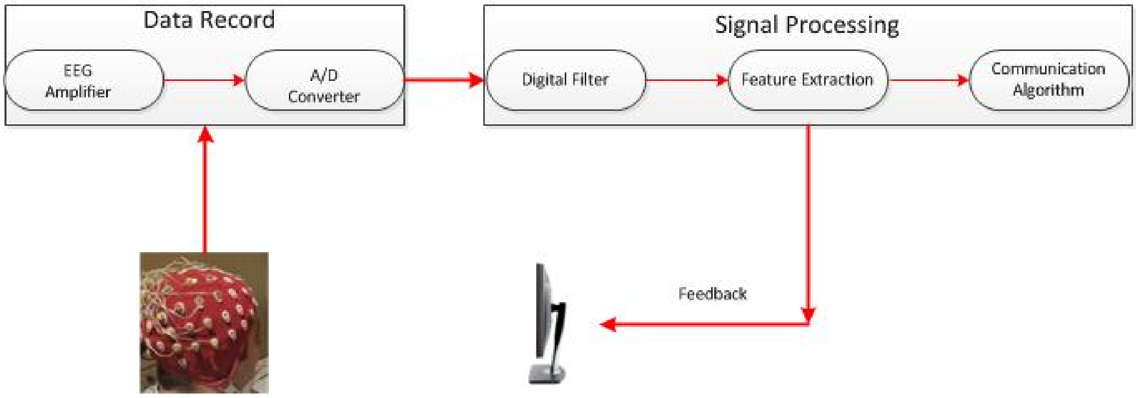

Figure 2.7: A typical BCI system model.

Signals are collected using electrodes and transmitted to the EEG amplifiers. Multiple filters are applied by the amplifier to remove artifacts: Notch filter to remove mains interference, low-pass filter to cancel DC components, high-pass filter to cancel irrelevant frequency components [32]. After that, the analog signals are converted to digital signals using an analog/digital converter. Depending on the application, related electrodes and bandwidths are selected. To determine the intention of the subject, various classification methods have been applied either directly to the signals or to the features extracted from them. As a result, the output can be used to control external devices such as typing on computer screen, controlling robotic devices or a cursor on the screen. Some of the well known BCI applications are: speller systems [37], generic cursor control applications [38], computer games and environmental control [39].

2.3.1 Sensorimotor Rhyhtms

Sensorimotor rhythms, called as mu-rhythms as well, repeat at the frequencies between 8 and 11 Hz, and are often mixed with beta (20 Hz) and gamma (40 Hz) components. These rhythms are recorded from somatosensory cortices over the channels C3 and C4 according to an international system called 10-20 system. The power of sensorimotor rhythms decrease or desynchronize with the movement or preparation of movement, and increase or synchronize in the post movement period. In addition, mu-rhythms can also desynchronize with the imagination of movements without the movement (motor imagery) [10]. In 1990s, motor imagery movements are decoded from the phenomena of event-related synchronization (ERS) and desynchronization (ERD) [2]. Nowadays, motor imagery related tasks, such as cursor control, are one of the most popular applications of BCI systems. In motor imaginary tasks, users imagine one of the actions among two or more (such as imagining left or right hand movement), and BCI system determines which action is intended [40]. The most important work on motor imagery systems is done by Wolpaw's group in Albany and Pfurtscheller group in Austria. In early 1990s, Wolpaw and his group have developed the first motor imagery based BCI system that controls cursor movement by EEG signals. They used the 8-12 Hz mu rhythm collected from the scalp with electrodes for moving a cursor to target in computer screen. Their computer calculate the segment's mu rhythm amplitude by taking square root of power and expressed in volts and compared with 5 voltage ranges preset by the operator. As a result of this compare, mu rhythm translated the amplitude into one of 5 possible cursor movements. The voltage ranges represent the cursor movement [41]. Following that, Wolpaw et al. have improved their system to a 2-D cursor movement control with a short training time. They used mu and beta rhythms and square root of power for controlling cursor movement. Their study extends the possible application of non-invasive BCI technology to include real time multi dimensional movement control [15]. Pfurtscheller et al. have succeeded to discriminate between three imagined movements (feet, left and right hand movements) using spectral features. Later, Pfurtscheller and his colleagues have developed a spelling application "Virtual Keyboard" [42]. This application makes binary selections through the movements of a cursor controlled by motor-imagery. Moreover, Royer and his group have demonstrated that, a model helicopter can be controlled in 3-D space using sensorimotor rhythms [43]. Blankertz et al. made single-trial analysis. Beta frequency band oscillatory features and readiness potential are used in their work. They reduced the training effort of the users on motor imagery systems by employing advanced machine learning techniques and using 128 number of electrodes features [25]. Objective of Galan et al. is evaluating robustness of an non-invasive BCI for continuous mental control of a wheelchair. They used power spectral density in the 8-48 Hz band range as features. Also Gaussian classifier is used in their work [44].

In motor imagery studies, spectral power densities around 16-24 Hz for beta, 12-16 Hz for sigma, 8-12 Hz for alpha bands are used. Increasing and decreasing of band powers can be seen in Fig. 2.8.

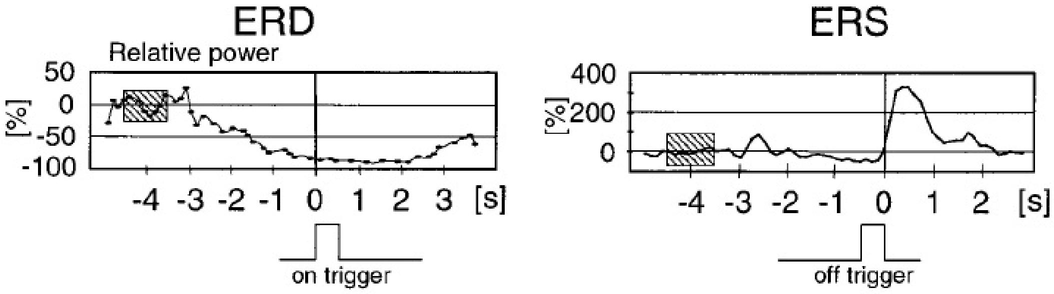

Figure 2.8: Left figure represents ERD and right figure represents ERS processing. A decrease of band power indicates ERD and an increase of band power ERS. Courtesy of [2].

2.3.2 Event Related Potentials and P300

The event-related potentials (ERPs) are, unconscious stereotyped brain waves related to the specific internal or external events such as stimuli, responses or decisions. BCI systems can recognize ERPs; before, after or during sensory, motor or psychological events. These events are called stimuli. ERP potentials contain a series of negative and positive components and carry information about cognitive and affective processes [45].

One main application area of BCI is the P300 speller, which is named after positive displacement of an event related potential (ERP) occurring around 300 ms after the presentation of the visual or auditory stimulus. In 1988, the P300 speller paradigm was introduced by Farwell [19]. The speller has a 6x6 matrix presented on a computer screen. One underscore, 26 letters and 9 numbers are included in this P300 speller matrix. The rows and columns of the P300 speller matrix are flashed in a block-randomized fashion. The subject is told to count the number of occurrences of a target stimulus that contains the target letter. In the matrix, the row and column containing the target letter are the relevant incidents or target stimuli where in a block of 12 flashes, there are two such incidents. Other incidents, rows and columns that do not contain the target letter are the irrelevant incidents or non-target stimuli, and there are ten such events in a block consisting 12 flashes

[32].

Fig. 2.9 shows the typical P300 response averaged over trials. This signal is recorded from Cz electrode. P300 signals are usually recorded from Cz and Fz electrodes [46].

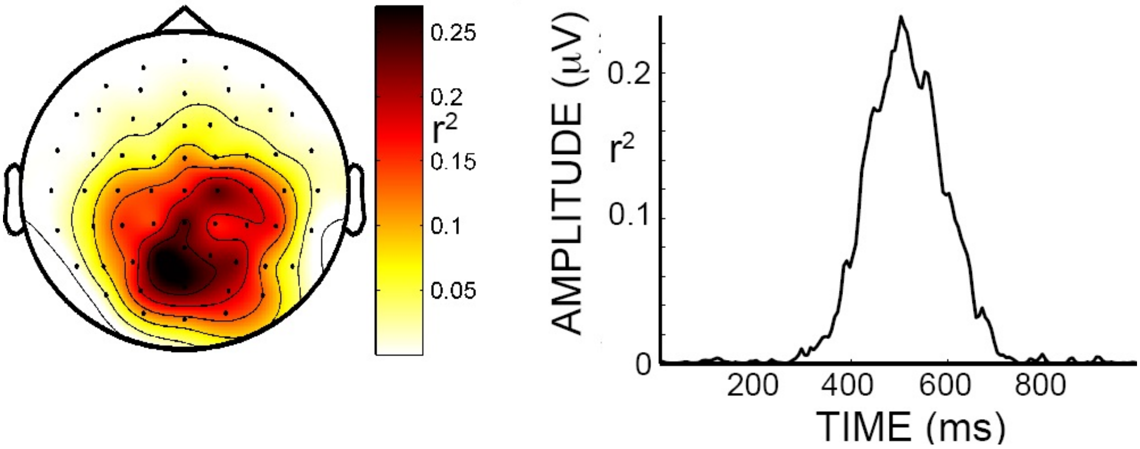

Figure 2.9: P300 signal is collected from Cz channel. Courtesy of [3].

P300 systems have been widely studied since they were first developed and tested by Donchin in 1988. Fig. 2.10 presents the first P300 speller paradigm. In P300 speller paradigm, there is a 6 by 6 matrix made up of 36 cells. Matrix has 26 letters of the English alphabet; and it also has different 1-word commands to control the system. In the figure, the subject is told to spell the word "BRAIN" letter by letter using the paradigm. Subjects are instructed to focus on the character of interest that is flashing. Rows and columns of matrix are flashed randomly. If a row or column with the target character is flashed, P300 signal will occur at the stimulus onset. With P300 feature selection and classification techniques, the attended character of the matrix can be estimated and then displayed to the subject. After the letters of 'BRAIN' are spelled, the subject selects 'TALK' command in matrix, and the word is read by computer [19].

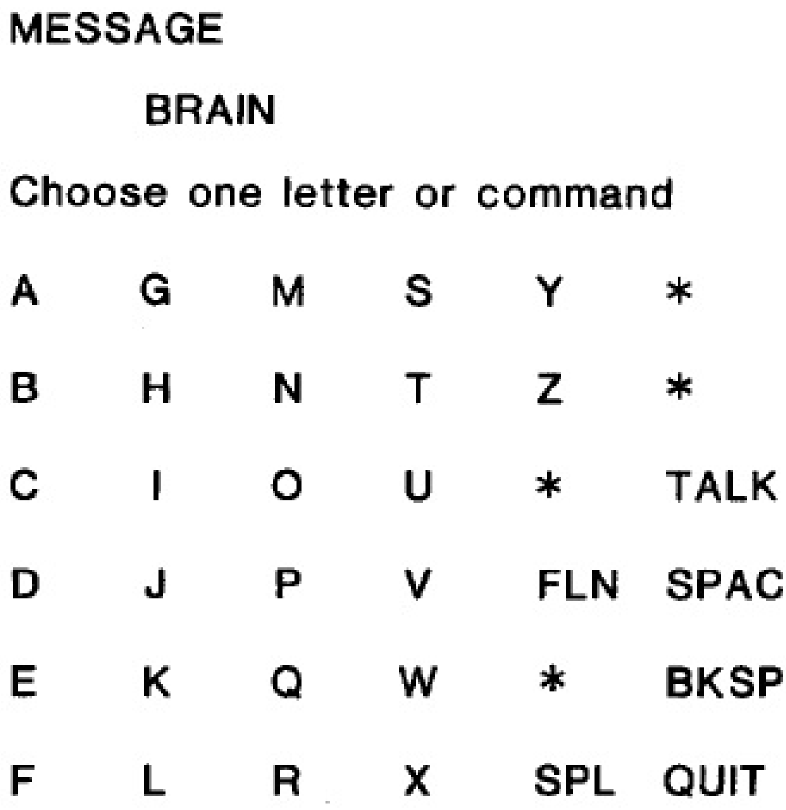

Figure 2.10: First P300 speller paradigm used by Donchin.

Even though the 6x6 matrix is widely used in P300 speller systems, researchers have developed various interfaces and investigated the affect of the interface on the system performance. "Hex-o-Spell" is one of those P300 spellers with different flashing paradigms. It is developed by Blankertz et. al. (Berlin BCI group). "Hex-o-Spell" speller displays multiple characters in an appealing visualization based on hexagons. Blankertz et. al. have achieved a mental text spelling performance of 7.6 character per minute [47]. "Hex-o-Spell" speller can be seen in Fig. 2.11. Moreover, one of the popular paradigms is rapid serial visual presentation (RSVP) keyboard. It shows visual stimulus sequences on the computer screen over time on a fixed focal area in quick successions. RSVP keyboard can be seen in Fig. 2.12 [48].

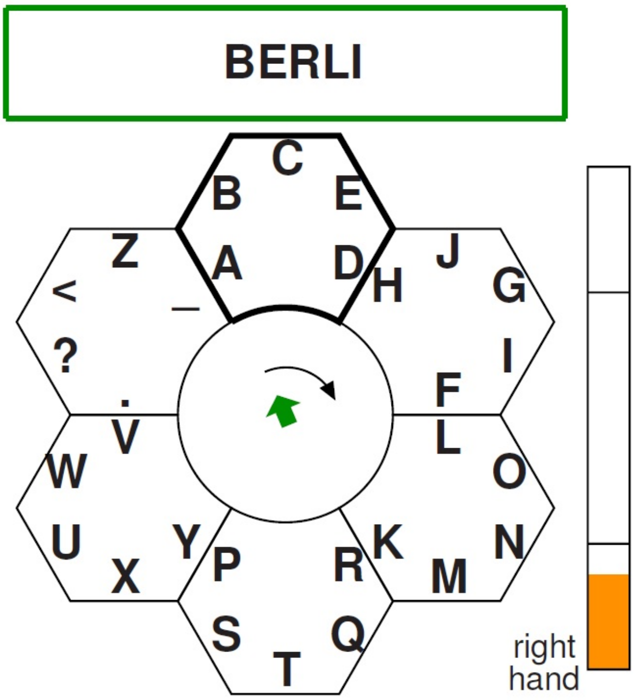

Figure 2.11: Hex-o-Spell interface developed by Blankertz.

Figure 2.12: RSVP interface.

EEG research groups use complete set of tools "BCI 2000" all over the world. This tools are developed by Schalk et. al [30]. Featuring a module-based system, BCI 2000 has the capability of data acquisition from several hardware, two stage (feature extraction and feature translation) signal processing phase. It also has an application interface where the subject decides an action with the help of translated control signals, and an operator interface to set various parameters and monitor other software and/or experiment related information (see Fig. 2.13).

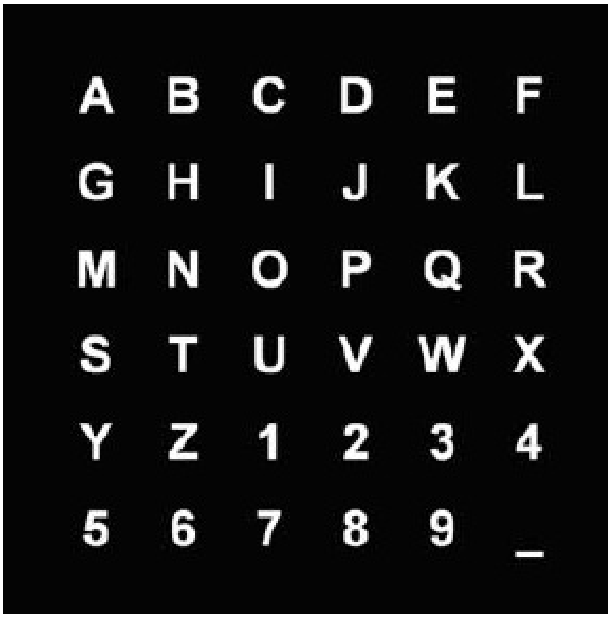

Figure 2.13: BCI2000 interface.

The SU-BCI P300 stimulus software was developed in Signal Processing and Information System Laboratory (SPIS), Sabanci University [32]. This is essentially a matrix based system similar to the one introduced by Donchin. SU-BCI P300 stimulus software is designed to deliver the subject the required visuals or directions to evoke the necessary potentials. The software allows any matrix size, cell content customization (letters or shapes), different coloring and stimulation schemes as displayed in Fig. 2.14.

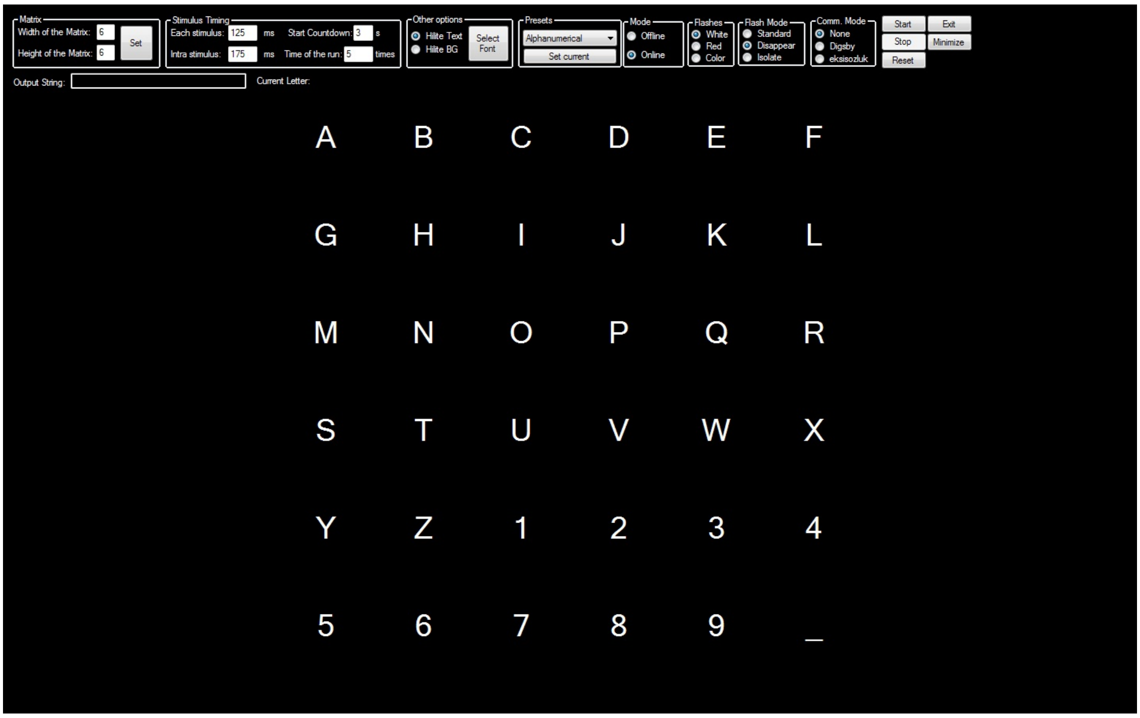

Figure 2.14: SU-BCI P300 Speller.

Sellers et al. research effectiveness of BCI system that operates by detecting a P300 elicited by one of four (YES, NO, PASS, END) stimuli. They tested 2 groups of users (first group has 3 ALS patients, second group has 3 non-ALS users). From their experiment results, it is seen that, two ALS patients classification accuracy were equal to those achieved by the non-ALS users. The result of their work propose that, P300 based BCI system provide as a non-muscular communication device in both ALS, and non-ALS control groups [49].

2.3.3 ErrP signals

The error related potential (ErrP) is a specific type of event related potential (ERP). ErrP signals occur when the user realizes an error in the system either as a result of user's actions or BCI misclassification [40]. Detecting and interpreting the errors from neural activities propose a new approach to improve BCI performance [50]. In early 1990s, presence of ErrP components are shown by two independent research groups [51], [52]. ErrP signals have 2 main components: early negative deflection (error negativity ERN) and a positive deflection (error positivity, Pe). Error negativity component is generated at fronto central regions that can be seen at nearly 100ms following the error stimulus, and error positivity component is generated at paretial regions as seen in between 200 and 500 ms after the error stimulus in Fig. 2.15. The components of ErrP signals are recorded generally from Fz, FCz and Cz channels [53].

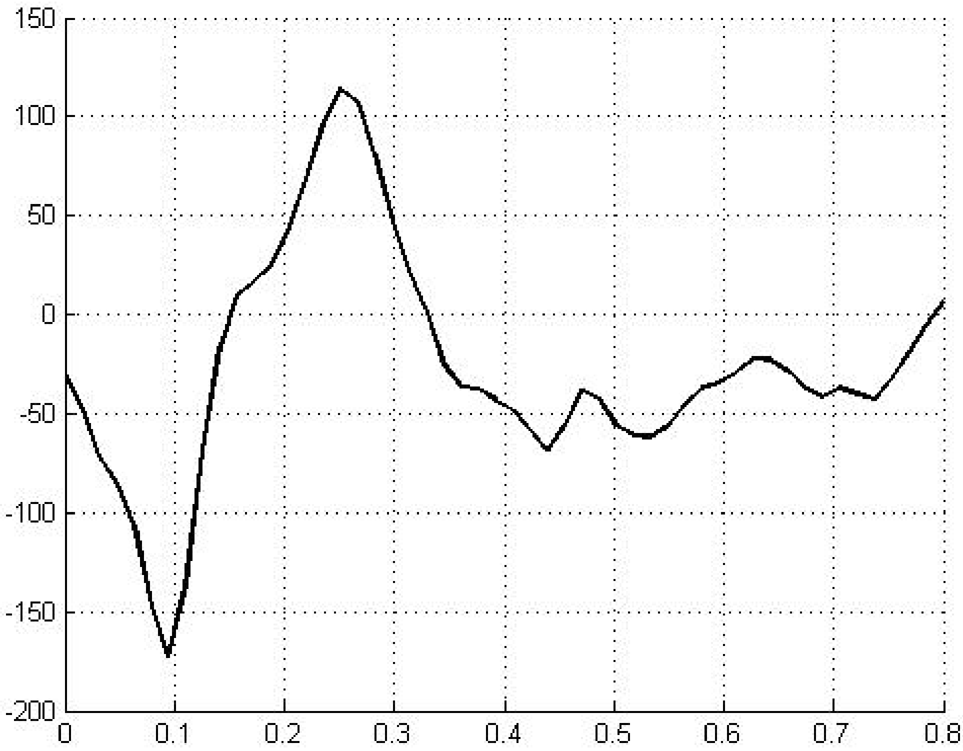

Figure 2.15: ErrP signal collected from FCz electrode.

Since 1990s, there are many studies focusing on ErrP signals. Schalk et al. studied the presence of ErrP components during cursor movements collecting mu and beta rhythms. They have established a positive peak at 40 ms after the end of erroneous trials at Cz electrode. Detecting ErrP signals provides a possible direction to improve BCI performance by automatically detecting errors made by the BCI [54], [54]. Blankertz et al. have designed an interface where user presses right key in response to a visual stimulus [55]. Later, Blankertz has achieved to minimize the need for subject training. For this propose they try to increase the classification accuracy by using ErrP signals. They have studied online use of the error potential as a line of research and have used the regularized Fisher discriminant for classification [56]. Moreover, Millan et. al. have studied the possibility of automatically detecting

ErrP occurring due to BCI mistakes, to improve the system performance and to adapt the classifier online [57]. In their research, a negative peak occurred 270 ms after feedback and a positive peak occurred between 350 and 450 ms after feedback. They have used a customized Gaussian mixture model (GMM) for classification of ErrP signals with 2-class motor imagery task. The experiment setup is designed to give the correct feedback with 80% or 60% probability [58]. Combaz and his colleagues have focused on improving the classification accuracy of a P300 speller by incorporating ErrP information. They have proposed to update the classifier scores when an error is detected. They have used SVM and FLDA for classification [59]. Milekovic et al. have studied 2 different type of errors: the execution error, due to inaccurate decoding of the subjects movement intention; the outcome error, due to not achieving the goal of the movement. They have used regularized linear discriminant analysis (RLDA) for classification [60].

2.3.4 Classification methods

This section gives brief information about the classification methods used in the P300 BCI context and motor imagery systems.

Linear Discriminant Analysis (LDA) is a binary classification method looks for a linear combination of features that characterizes or separates two classes. The classes are assumed to have normal density distribution [61]. Discriminant function identifies a hyperplane to separate classes, this function can be defined as:

$$f(x) = \alpha_x + \alpha_0 \tag{2.1}$$

where, α is a weight vector and α_0 is a bias. During classification, the sample of x that results α>0 or α<0 is assigned to class 1 or class 2 respectively. LDA is used to classify motor imagery movements using ERD components.

Different linear methods can have the same structure with different weight calculation processes. Fisher's Linear Discriminant Analysis (FLDA) is the benchmark for determining the optimal separating hyperplane [62]. FLDA tries to find set of weights α that maximize the ratio:

$$J(\alpha) = \frac{\alpha^T S_B \alpha}{\alpha^T S_w \alpha} \tag{2.2}$$

where S_B is the scatter matrix between classes S_w is the scatter matrix within a class.

Bayesian Linear Discriminant Analysis (BLDA) is an extension of FLDA. BLDA classifier is based on Bayesian theory. It assigns the feature vector to the suitable class with highest probability. BLDA prevents overfitting to high dimensional and possibly noisy datasets [63]. Through a Bayesian analysis, the degree of regularization can be estimated automatically and quickly from training data without the need for time consuming cross-validation. Details are in [32].

Another popular classification method is Support Vector Machine (SVM) in BCI systems. Vilademir Vapnik has developed SVM in the end of 1960s [64]. SVM determines the optimum separate hyper plane like LDA but the difference is SVM maximizes the distance between hyperplane and each data points. It maximizes the margin between classes. SVM is usually applied with linear kernel functions. Non-linear kernel function can also be applied instead of linear SVM. In this thesis, SVM is used in motor imaginary task based BCI.

K nearest neighbors is a simple algorithm that uses all available samples to compute a similarity measure. Later, it classifies new samples based on this similarity measure. Here, a sample is classified by a majority vote of its neighbors; the class it is assigned is the most common class amongst its K nearest neighbors measured by a distance function. Various functions are used to compute the distance, but the most common function is Euclidian distance function:

$$\sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \tag{2.3}$$

If K = 1, then the case is simply assigned to the class of its nearest neighbor [64]. The optimal K for most datasets has been between 3-10 for most of the applications.

The naive Bayes classifier is based on the assumption that the attribute values are conditionally independent given the target value [65]. The naive Bayes classifier can be described as:

$$v_{NB} = \arg\max_{v_j \in v} P(v_j) \prod_i P(a_i|v_j) \tag{2.4}$$

where a_i is attribute values, v is finite set, v_{NB} is the target value output by the classifier.

Generally, k-means is used for solving unsupervised clustering problems. K-means algorithm classify the dataset through a certain number of clusters. The algorithm chooses k randomly defined points inside the hyper-volume containing the pattern set. After that, algorithm assigns each pattern to the closest cluster center. Then, the cluster centers using the current cluster memberships is recalculated. If a convergence criterion is not met, algorithm reassigns each pattern to the closest cluster center. This algorithm aims minimal reassignment of patterns to new cluster centers, or minimal decrease in squared error [66]. The squared error over all K clusters:

$$J(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} ||x_i - \mu_k||^2 \tag{2.5}$$

2.4 Adaptation on BCI

One of the main problems in BCI systems is the non-stationary nature of the recorded signals. The characteristics of brain signals may vary after training due to task difference, movement of the electrodes, gel drying, user's level of attention and fatigue, causing the signals to become non-stationary. Because of this non-stationary behavior, a classifier trained on previous EEG data may not be optimal for following sessions [10]

The patterns of EEG signals may differ after training due to task difference, movement of the electrodes, gel drying, user's level of attention and fatigue, causing the signals to become non-stationary [26]. Since static classifiers have difficulties identifying these non-stationary EEG patterns [67], several approaches are proposed for online adaptation. Blumberg et al. have used expectation maximization (EM) to update the mean and covariance values of a Linear Discriminant Analysis (LDA) classifier. They have applied semi-supervised adaptation approach to motor imagery data [26]. Vidaurre et al. have developed a framework to find the best feature extraction and classification method for adaptation. In the paper, they have proposed an online adaptation approach using adaptive estimation of the information matrix (ADIM), which is derived from Quadratic Discriminant Analysis (QDA). Later, they have extended their work to update LDA with Kalman filtering [68]. Adaptation of P300 spellers is also explored by researchers. Li et al. have used Support Vector Machine (SVM) with a self-training procedure to reduce training efforts and have improved P300 spelling accuracy through adaptation [22]. Panicker et al. developed a co-training based adaptive method using Bayesian Linear Discriminant Analysis (BLDA) and Fisher Linear Discriminant (FLD) for P300 speller systems [69]. Dal Seno et al. have designed the first BCI system, which detects ErrP signals in an on-line P300 system. They have proposed the use of a genetic algorithm to detect P300s and added an automatic error-correction system. They have used a logistic classifier for P300 detection and single-sweep for ErrP detection [70]. Lu et. al. have studied subject-independent classification model (SICM) for P300 data in an unsupervised manner. Firstly they classified EEG data by the SICM online at the initial adaptation stage. An adapted subject-specific classification model (ASSCM) is then built by learning from the just classified user data and the corresponding labels predicted by the SICM. Afterwards, incorporating the ensuing new user's data iteratively update the ASSCM. Corresponding labels predicted by either the SICM or the ASSCM itself, depending on its confidence score [27].

In this thesis, we investigate the effect of semi-supervised adaptation on a P300 speller system integrated with language models and on motor imaginary based BCI systems. In our system, we propose a solution for the systems dealing with small training data. We have also studied the subject dependent nature of BCI systems to offer subject-to-subject transfer solutions. Details will be introduced in chapter 3 and chapter 4.

2.5 Language Model

Language model can be defined as a statistical model of a specific natural language. This model characterizes, captures and exploits the rule of natural languages [ref33]. Language models have various application areas such as; speech recognition, machine translation and text input.

In a language model, a sentence is not an arbitrary sequence of words and letters. It follows some rules inherent to the language and common uses. For example, given the first few letters of a word, it is possible to predict the best match that would complete the word. If preceding and succeeding letters are provided, task will be even easier. The reason behind this process is, some sequences of letters or words will occur more often and others will occur less, so different letter sequences can be formed based on the context. A statistical language model assigns probability distribution over sequences of words or letters to capture these probabilities [71].

P300 speller systems are designed as mental typewriters utilizing brain signals to spell some text or meaningful letter sequences. The letter probabilities obtained from a statistical language model can be used as prior information in the decision algorithm for letter estimation [72]. Based on this approach, we have worked on exploiting a language model in conjunction with the information coming from EEG data to merge them in a single decision-making algorithm.

When characters are typed within words in a particular language, neighboring characters would provide information about the current characters as well. Cagdas et al. suggest a fully probabilistic approach for incorporation of such information into a BCI-based speller through HMM trained by a language model (Forward-Backward and Viterbi algorithms). Their approach takes advantage of both the past and the future and previously declared letters can be updated as new information arrives [23]. In our works, we used language model with HMM in semi supervised adaptation method on P300 speller system.

Chapter 3

Adaptation in a Motor Imagery-based BCI System

Moving a limb or contracting a muscle causes changes in EEG signals. Essentially, preparation or imagination of movement results in increase or decrease in sensorymotor rhythms (SMR) recorded from motor areas. SMR rhythms refer to EEG oscillations with spectral power energies around 16-24 Hz (beta), 12-16 Hz (sigma), 8-12 Hz (alpha). The decrease and increase of oscillatory activity in the frequency bands are called event-related desynchronization (ERD) and even-related synchronization (ERS), respectively. ERD/ERS patterns can be volitionally produced by motor imagery, which is the imagination of movement without actually performing the movement [73].

Motor imagery experiments we have designed and implemented using our BCI system are described in this chapter. This chapter is composed of two main parts: in the first part, we evaluate motor imagery BCI systems on standard BCI competition data; in the second part, we apply proposed adaptation approach as a solution to specific problems faced by researchers.

3.1 Processing BCI data set

Two different data sets are used for motor imagery BCI experiments. First data set, Data set 1, is supplied by Department of Medical Informatics, Institute for Biomedical Engineering, University of Technology, Graz, Austria for BCI Competition 2003 [74]. It was generated to research the fundamental patterns of motor imagery movements. The dataset was collected from a healthy female subject (age 25). The subject sat in a normal chair in armrest position during whole experiment. The subject was instructed to control one dimension feedback bar by imagining left or right hand movements (Fig. 3.2). The experiment had seven runs, each run has 40 trials and between each run there were breaks. The experiment is completed in one day. The training set contains 140 trials, and test set contains 140 trials as well. Length of one trial is 9 seconds. In duration of a trial, first 2 s were quiet, beginning of the trial was showed by acoustic stimulus at t=2 and " + " was exhibited for 1 second. At 3rd second left or right cue (arrow) was appeared for 6 seconds. The order of cues was random for preventing any systematic effect. The subject used her imagination as described above to move the feedback bar into the direction of the cue. EEG signals are recorded from C3, Cz, C4 channels (Fig. 3.1) with Ag-AgCl electrodes. EEG signals were filtered between 0.5-30 Hz, and sample rate of the signal was 128 Hz.

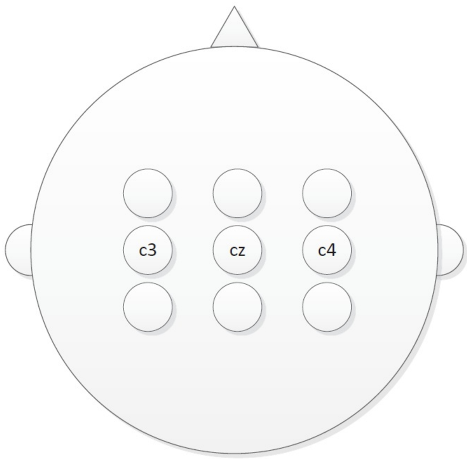

Figure 3.1: Location of the C3, Cz, C4 electrodes.

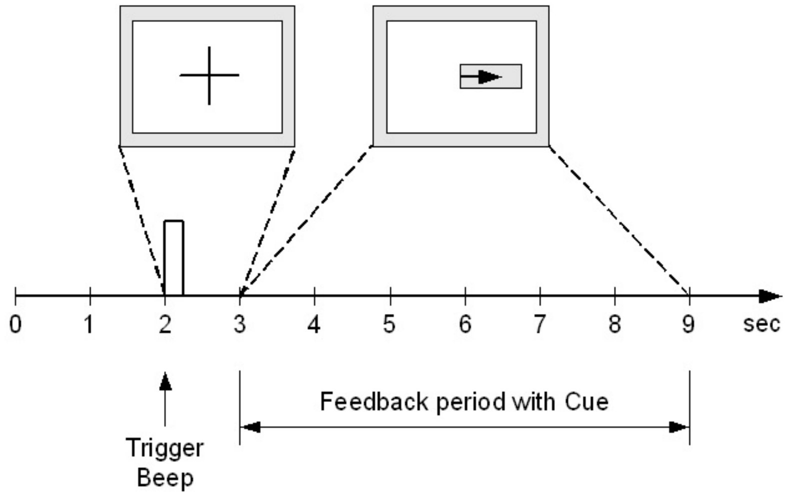

Figure 3.2: Description of Data set 1 stimuli.

Second data set, Data set 2, was supplied by the Institute for HumanComputer Interfaces, University of Technology GrazBCI Lab for BCI Competition 2005 [75]. It was recorded to research the non-stationary behavior of EEG signals. The experiment was similar to a “basket paradigm” [76]. At the beginning of each trial, subject saw a black screen for 3 seconds. After that, green and red baskets appeared at the bottom of the screen and a green ball at the top of the screen (Fig. 3.3). After one second, the ball began to fall downward with constant speed. The task of the experiment was, to imagine moving left or right hand to steer the ball towards the target direction (green or red basket). Length of each trial was 7 seconds, and between each trial there was 1 or 2 s random interval. The order of left and right cues was random. The dataset contained two-class EEG, and it was collected from three subjects (sat on a relaxing chair in armrest position) in three sessions performed at different days. Each data set contained recordings from consecutive sessions during experiment. In our study, we have used non-stationary data set that is collected in the same experimental setup from “x11” and “s4” users. Number of trials for these subjects is 1080 trials (9 runs were recorded during each session, and one run consisted of 40 feedback trials). Between each run there were breaks between 1 and 15 minutes. These non-stationary EEG signals are recorded from C3, Cz, C4 channels with Ag-AgCl electrodes and a g.tec amplifier (Guger Technologies OEG Austria). Signals were filtered using a bandpass filter with cutoff frequencies 0.5 and 30 Hz, and sample rate of the signal was 125 Hz [77].

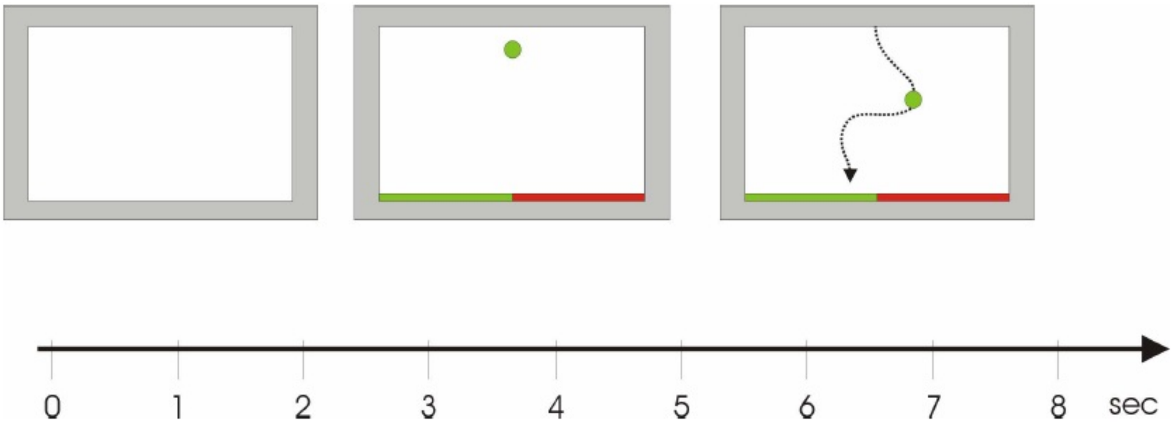

Figure 3.3: Description of Data set 2 stimuli.

3.1.1 Feature Extraction

Extraction of features leads us to the interesting and informative sections of data. It can be viewed as a dimension reduction technique as well. Features can be extracted from time, frequency and spatial domains. Time domain features are a function of time. They are used to represent an ERP signal following the appearance of a stimulus. The change of oscillatory activities makes it easy to extract features in frequency domain. Spatial domain features model the brain activity by combining signals from different EEG channels. Spatial and frequency domain features are commonly used for motor imagery BCI systems [78].

In our work, the spectral power computed in the EEG frequency bands are used as features. Sensorimotor rhythms show rhythmic EEG oscillations with spectral power energies around 16-24 Hz (beta), 12-16 Hz (sigma), 8-12 Hz (alpha), as we mentioned in Section 1. Power spectrums of these frequency bands (which describe the frequency content of the EEG signal) are computed as relevant features for classification. The power spectrum can be calculated using Short Time Fourier Transform (STFT) applied to each trial with a window. An overlapping window is designed to have 512 samples, and it was shifted for 64 samples in each step.

Neurophysiological activities are observed in the brain after the appearance of the cue, and the affect of the stimuli decreases with time. Therefore, we have analyzed the signals between 3s and 7s instead of analyzing entire signal. After that, we have calculated power spectrum densities (PSD) of alpha, beta, sigma bands in that window. As a conclusion of this process, the extracted features of a trial consists of 3 averaged PSDs for 3 channels, and were used as the input for the classifier [61].

3.1.2 Classification

This section provides insight on classification of EEG signals and proposed adaptation approach using classifiers such as linear discriminant analysis (LDA), support vector machines (SVM), k-nearest neighbor (k-NN) and naive bayes (NB). We have used MATLAB's Statistical Toolbox and 'prtools' Toolbox for classification purposes.

The accuracies for four different classifiers with Data set 1 are shown in Fig. 3.4. We have extracted the features for four different time windows; 3.2 to 5 seconds, 3.2 to 7 seconds, 3.5 to 5 seconds and 3.5 to 7 seconds.

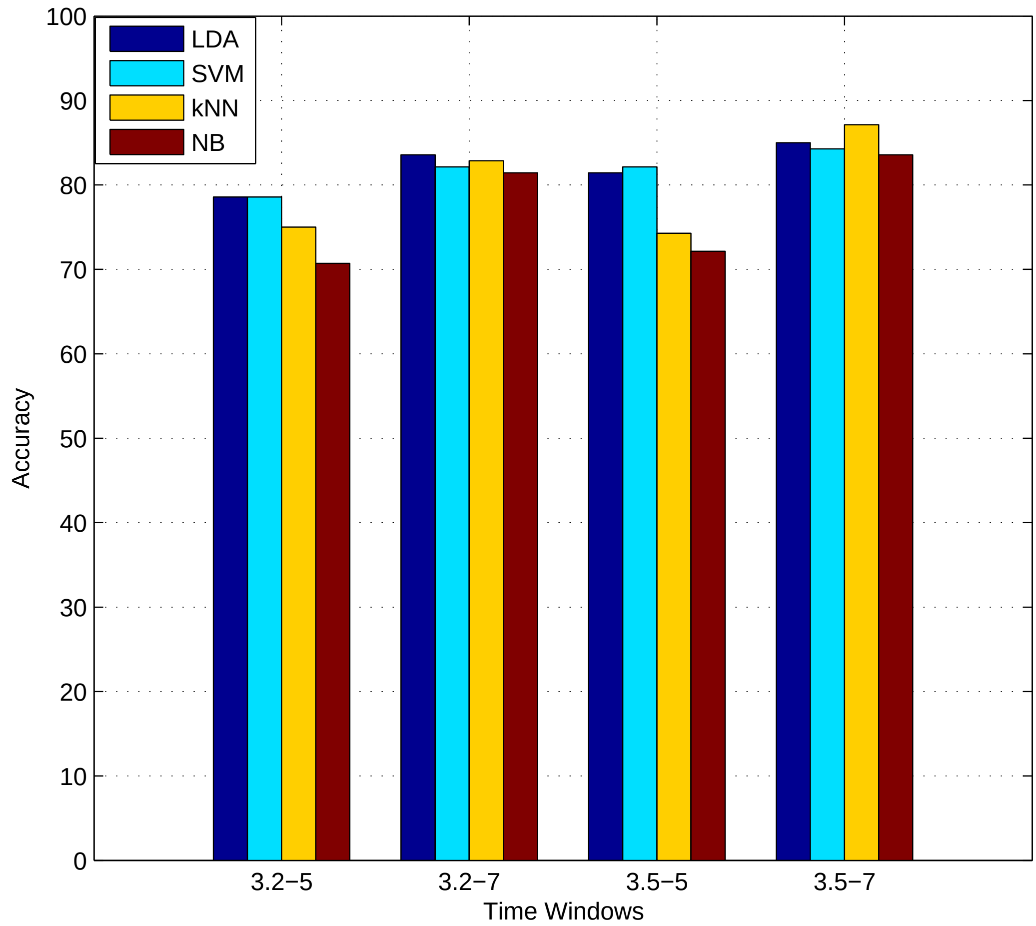

Figure 3.4: The accuracies for four different classifiers with Data set 1.

In this figure, x-axis represents the timing windows which contain the data, and y-axis represents accuracy of classifiers. In particular, blue, light blue, yellow, brown bars indicate the accuracy of LDA, SVM, kNN and Naive Bayes classifiers respectively. As seen in Fig. 3.4, from 16 different classification results for this data set, the lowest accuracy is 72.14% from Naive Bayes, and the best accuracy is 87.14% from kNN. We can analyze the affect of time windows on the accuracies as well. For all four classifiers, the features extracted from 3.5 seconds to 7 seconds appear to be more discriminative.

3.2 Adaptation

The non-stationary behavior of brain signals is a major problem for BCI systems. Static classifiers fail to recognize the pattern in these signals accurately. Some of the reasons behind this non-stationary nature of EEG signals are as follows:

- Electrode displacement,
- Impedance of electrodes,
- Gel drying,
- Internal state of user mind,
- Attention level of user,
- Noise caused by amplifier or environment,
- Variations between the calibration measurement and online performance of BCI.

Non-stationary behavior of EEG signals was discussed in Section 1 in detail. Analysis of recorded EEG signals is used to identify the intentions. One common approach is training a classifier using data with available labels, and testing the classifier on separate data. To be able to obtain satisfactory performance, a large training data is required; which causes a long and tiresome training process. Even after a long training process, the classifier may not perform well over time due to the reasons listed above. Another issue BCI researchers have faced is the subject dependent systems. A classifier trained with one user's EEG signals is not applicable to other users', which makes BCI systems subject dependent [79]. The time and effort required for training affects the attention level of user as well. This study aims to reduce the required training time without compromising the performance for BCI systems. It also addresses the non-stationary nature of EEG signals.

There are some studies in the literature focusing on the adaptive BCI systems. Some of them rely on supervised techniques, whereas, some researchers propose other approaches such as Kalman filtering [80].

In this study, we take a semi-supervised learning perspective, and propose solving two problems mentioned before (non-stationary behavior of EEG signals and decreasing training data) by updating the BCI system with labels obtained from the outputs of the classifier. To test the approach, data from motor imagery BCI system are used. Attributes extracted from EEG signals are classified with Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM).

3.2.1 Adaptation Algorithm

The BCI system is adapted to the data using the estimated labels from the classifier as we collect the EEG signals. In this work, we have followed two different adaptation approaches for two different scenarios: limited amount of training data and non-stationary EEG signals.

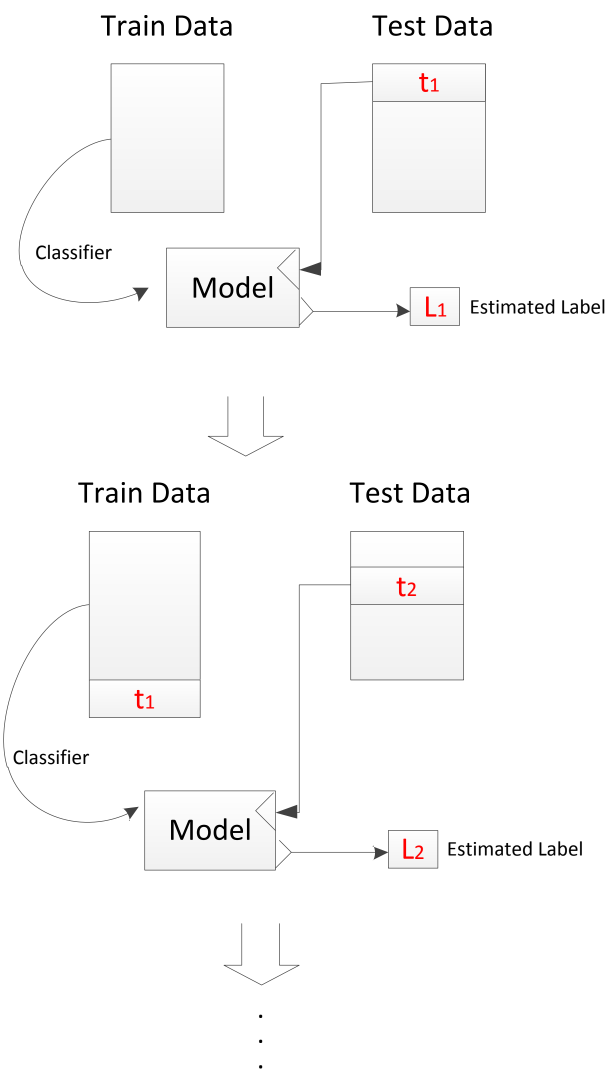

Figure 3.5: Adaptivity algorithm scheme.

Regarding the first problem where the size of the training data is small, we have used the method represented in Fig. 3.5 based on Blumberg's adaptation method [26]. First, the classifier is trained with the limited amount of available data. Resulting classifier is tested on a group of samples from the test data, in order to create the estimated labels. At this point, training data is extended to include the tested samples with estimated labels from the classifier. Classifier is re-trained using the extended training data, and these steps are repeated until all samples in the test data are used. This method can be summarized in a formula as the following:

$$M^{i+1} = f_{classifier}(Tr^{i+1}_{n\times m}, Tr\_L^{i+1}_{n\times 1}) = f_{classifier}(Tr^{i} \frown T(t), Tr\_L^{i} \frown M^{i}(T(t))) \tag{3.1}$$

where "M" and "Tr_L" stand for the labels created by the classifier and the labels of the training data set, respectively. "Tr" represents the training data for the "$f_{classifier}$" function, and "T(t)" represents the test data. In order to represent the column based attachment process, $\frown$ sign is utilized in the following manner:

Defining a and b as $a = [a_1\ a_2\ ...\ a_n]^T$, $b = [b_1\ b_2\ ...\ b_n]^T$ using of the attachment sign will result in $a \frown b = [a_1\ a_2\ ...\ b_1\ b_2\ ...\ ]^T$.

While the second adaptation approach is similar to the first one described above, the training data is not extended as it can be seen in Fig. 3.6. Since the idea is to adapt to the nonstationarities in the EEG data, we remove the earlier sections of training data during the retraining process. The data used for this purpose are gathered from a different set of experiments. In these experiments, when the tested samples are included with estimated labels from the classifier, the same amount of older samples is removed from the training data. Along this process, the classifier is adapted to the changes in the data by updating the training data accordingly.

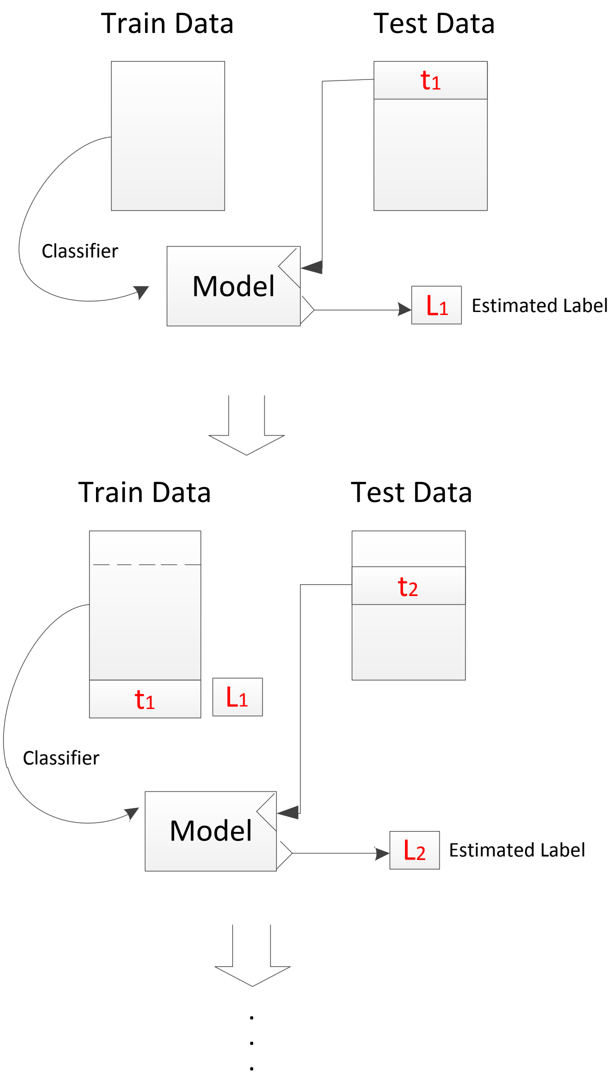

Figure 3.6: Adaptivity algorithm scheme, when train data is not extended.

3.2.2 Results

We have applied the proposed semi-supervised adaptation method to Data set 1, to reduce training data without compromising the performance. We have used randomly selected 15 trials out of 140 trials as initial training data. After that, we have updated the classifier as we have included trials one by one using the labels obtained from classifier. In this operation, training data is extended to include all trials tested by classifier. Later, the initial random selection of 15 trials is repeated 10 times and a final average classification accuracy is computed. We have repeated the same experiment by initially using 20 trials for training. The results of this process are analyzed by increasing the initial size of the training data until it became 70 trials.

In Fig. 3.7 and Fig. 3.8, the accuracy results of LDA and SVM classifiers for the growing training data experiment are presented. In these graphs, x-axis represents the number of trials for the initial training data, and y axis represents average accuracy results for the corresponding classifier. Here, green line indicates the accuracy of classifier without adaptation, i.e. static classifier. Blue and red lines show the accuracy of adaptation with estimated and true labels respectively. We have compared the results of static and dynamic (with adaptation) classifiers. For the dynamic classifiers, we have examined two different cases where classifiers are updated with true labels or with estimated labels. Based on the results, we can say that, improvement of the performance with adaptation is more obvious for smaller amounts of initial training data. Expectedly, results of adaptation with true labels are better than results of adaptation with estimated labels. However the true labels are not available in real-time applications. The accuracy results of adaptation with true labels point the highest performance we can achieve using the proposed adaptation approach.

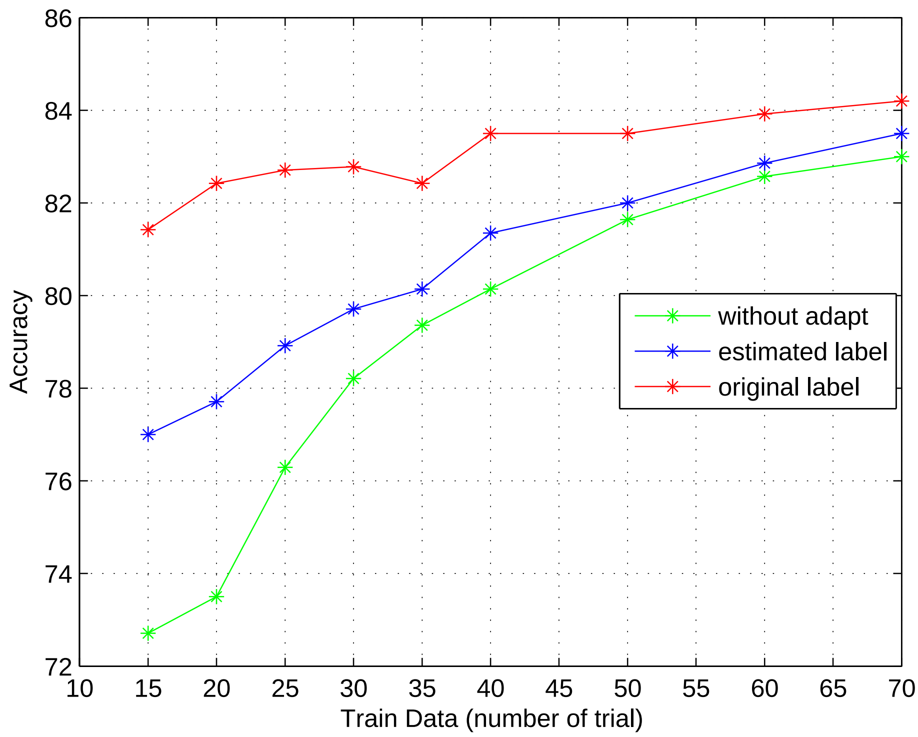

Figure 3.7: LDA classifier results with limited training data.

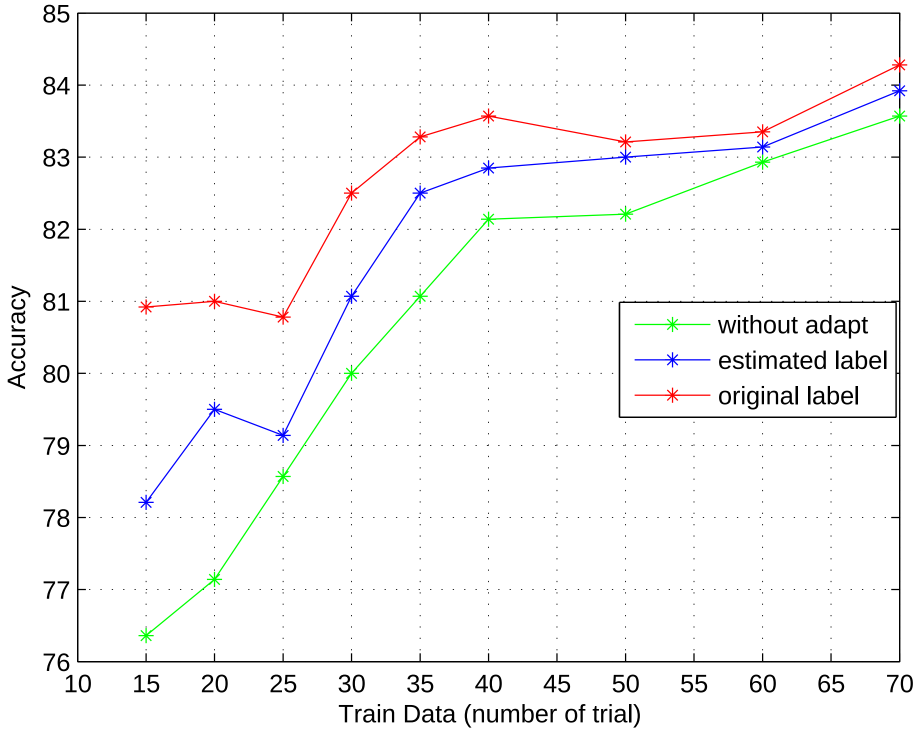

Figure 3.8: SVM classifier results with limited training data.

In the second experiment, we have used non-stationary data (Data set 2) explained in section 3.1 to adapt the system to time varying signals with semi-supervised learning. In this case, the data consist of three sessions for each user since it is recorded at three different times. These sessions are called s1 (session 1), s2 (session 2) and s3 (session 3) for the rest of the study. We have used s1 data set (155 trial group is for s4 user, 180 trial group is for x11 user) as the initial training data. The rest of the data, s2 (355 trial group is for s4 user, 358 trial group is for x11 user) and s3 (305 trial group is for s4 user, 355 trial group is for x11 user) data sets are treated as separate test data. The aim in this experiment is to run the system with the data at hand and let the classifier to adapt to the changing data in time, without the need of training and re-calibration after the data is saved in the first day.

We have observed the effect of adaptation with Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM) classifiers on non-stationary signals, and presented the results in Table 3.1 and Table 3.2 for each user. Again, we have compared the results of classification without adaptation to the results with adaptation. It is observed that, adaptation the classifier using true labels results in higher improvement with respect to the adapting using predicted labels.

Train-Test	Classifier	Without adapt.	Estimated label	Original label
s1-s2	LDA	67.10	69.85	71.86
s1-s3	LDA	66.75	68.78	70.50
s1-s2	SVM	67.49	69.91	71.58
s1-s3	SVM	67.18	69.01	73.91

Table 3.1: Adaptation results of time varying signals (subject s4).

Train-Test	Classifier	Without adapt.	Estimated label	Original label
s1-s2	LDA	70.39	72.09	77.18
s1-s3	LDA	83.35	84.53	85.63
s1-s2	SVM	70.87	73.57	76.84
s1-s3	SVM	83.86	85.24	86.06

Table 3.2: Adaptation results of time varying signals (subject x11).

For the motor imagery based BCI experiments designed during this study, we have analyzed the performance of semi-supervised adaptation method, which uses the classifiers output labels. We have proposed this approach as a solution to two common problems BCI systems face. One of them is to decrease the long and tiresome training and calibration phase. Second problem is the changing patterns of EEG signals in time due to various reasons, which cause poor classification results. When we start with a very small initial training data, such as 1 trial, it is observed that; the dynamic classifier improves the accuracy around 5% compared to the static classifier. The difference in performances is decreased as the initial training data size is increased, but the dynamic classifier continued to show better performance. In the case of true label usage for adaptation, the performance improvement is even higher as expected. The accuracy results of adaptation with true labels indicate the endpoint of the improvement one can achieve, using semi-supervised adaptation approach described in here. Experiments are carried out with LDA classifier, frequently used in BCI experiments, and SVM classifier, a more complex classifier; and it is observed that both classifiers have improved performances with adaptation. Due to the experiments being conducted in different days, second dataset consists of non-stationary EEG signals. In this case, instead of extending training data during the test phase, the initial training data is replaced with the incoming data. This adaptation approach improved the performance of the classifier around 4% with respect to the classifier trained only with the initial training data.

Chapter 4

Adaptation in a Language Model-based P300 Speller

In this section, we describe a semi-supervised adaptation method for language model-based P300 spellers. In the particular speller we consider here, the EEG data pass through a preliminary BLDA classifier to produce scores for each letter. The scores are added up in consecutive repetitions of stimuli (called trial groups) for a character; and then combined with a language model using forward, forward-backward, or Viterbi algorithms on a HMM to estimate the letters typed by the users. The language model is incorporated within a fully probabilistic approach, which exploits information from the previous and future letters for the current letter [23]. We propose an adaptation method for a language model based P300 speller. Even though different adaptation approaches have been proposed for P300 spellers in the literature, to the best of our knowledge, adaptation of a P300 speller combined with a language model has not been studied before. We focus on two main problems of EEG based BCI systems: small size training data and subject dependent characteristics of EEG. Updating the classifier using the labels obtained from the classifier for these two scenarios, we have observed significant improvement compared to the static classifiers.

4.1 Terminology

Some terms which used in this section are should be clarified [35], [32].

- **Target letter:** This is the character the user is showed to focus on at that instant.
- **Trial:** This denotes the intensification of each row or column, the timing of

which is denoted by trigger signals in the collecting procedure.

- **Trial group:** It is the group of trials which include each row and column intensification that is flashed exactly once. For instance, in 6x6 speller matrix, a trial group consists of 12 individual flashes in just one row and column flashed once.

- **Run:** Run is the collection of some trial groups. A run is recorded for each letter defined in one session. There can be a period of a few seconds between each run, but the recording is continued.

- **Session:** Session is the time block in which the recordings for all previously defined number of target letters are done.

4.2 Methods

4.2.1 Classification and Language Models

Bayesian Linear Discriminant Analysis (BLDA) is used to classify P300 signals, and classification scores for each letter are calculated. Classification scores of independent letters and a Turkish Language Model are integrated into the Hidden Markov Model (HMM). Forward, forward-backward, and Viterbi algorithms are utilized for inference to decide the outcome of each position in the letter sequence [23]. Fig. 4.1 represents the sequential chain of HMM.

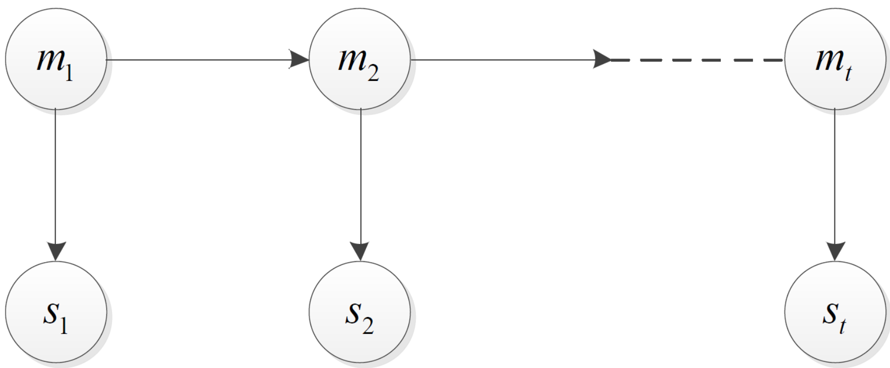

Figure 4.1: A sequential HMM.

In this figure, $M = \{m_1, ...m_t\}$, each $m_i \in$ E where E is the set of all elements in the speller matrix. $S = \{s_1, ...s_t\}$, and for each instant i,s_i illustrates the EEG data collected during the spelling of the i-th character. BLDA enables the extraction of the marginal conditional densities between the letters and the data at each instant. The overall conditional distribution of M given S for an N-th order HMM is;

$$\begin{aligned} P(M|S) \propto & P(m_1) \prod_{i=1}^{N} P(m_i|m_{i-1}) P(s_i|m_i) \\ & \prod_{i=N+1}^{T} P(m_i|m_{i-N}, ..., m_{i-1}) P(s_i|m_i). \end{aligned} \tag{4.1}$$

To estimate the character with the highest probability for each time instance, we can maximize the marginal probability of a letter sequence given the EEG data. For all i in [1, T],

$$m'_i = \arg\max_{m_i} p(m_i|s_{1:T}). \tag{4.2}$$

This problem can be solved using the forward-backward algorithm.

To find the most likely letter sequence, we can maximize the joint probability of a letter sequence given the EEG data:

$$M' = \arg\max_{M} p(M|s_{1:T}). \tag{4.3}$$

Viterbi algorithm can be used to solve this problem. The details on solving these equations using forward-backward or Viterbi algorithms are discussed in [23].

Another algorithm forward algorithm is a specialized case of the Forward-Backward algorithm. In forward-backward algorithm, the marginal posterior probability of character is estimated for each time instance. This is not a real time algorithm. Whereas, forward algorithm is real time and it outputs the most likely letter based on data up to that time point:

$$\arg\max_{m_i} p(m_i|s_{1:i}). \tag{4.4}$$

The details of the algorithms can be found in the Appendix.

4.2.2 Adaptation Algorithm

In this section, we describe how the BCI system is adapted to the data using the estimated labels from the language model based classifier, as we collect the EEG signals. We propose two different adaptation approaches for two different objectives: (1) to compensate for the limited amount of training data, and (2) to test the system trained with one subject on a different subject.

Regarding the problem where the size of the training data is small, we use the adaptation scheme sketched in Fig. 3.5 based on Blumberg's adaptation method [26]. This method is the same with the one explained in section 3. First, the classifier is trained with the limited amount of available labeled data. The resulting classifier is tested on a group of samples from the test data in order to create the estimated labels. At this point, the training data are extended to include the tested samples with the estimated labels from the classifier. The classifier is re-trained using the extended training data, and these steps are repeated until all samples in the test data are used. This method is summarized in formula 3.1.

According to Thulasidas et al. [81], P300 systems trained with one user do not generate reliable results for another user. In the second adaptation approach, we propose inter-subject adaptation as a solution to such subject dependent BCI systems, which allows a classifier trained on one subject to be used on a different subject without a training phase. For this purpose, a P300 speller trained with one user is tested on a different user, while adapting the system to the new user's unlabeled EEG data. The process of updating the training data with generated labels is similar to the previous method. The only difference is, when new data is included in the training data set with estimated labels, the same amount of data are removed from the old training data; as it can be seen in Fig. 3.6. With this strategy, eventually the training data includes samples only from the new user with labels provided by the classifier.

4.3 Experiments

4.3.1 Data

The data set used in this study has been collected at the SPIS Laboratory with the approval of the Sabanci University Research Ethics Council. It is collected from 7 healthy subjects (ages between 18 and 30) during offline spelling experiments, and just 2 subjects had prior BCI experience. The procedure was explained to the subjects before the experiment, and their informed consent was obtained. The subjects sat in a normal chair without movement during the experiments. Data were sampled at 2048 Hz using the widely used P300 spelling interface, a 6×6 matrix of characters. Inter-stimulus interval (ISI) was 125 ms and stimulus duration was 50 ms for all conditions. Rows and columns are flashed in a block-randomized fashion. Each row and column is flashed once in 12 flashes. Each flash lasts for 125 ms, and after each flash there is a period of 175 ms where none of the cells are highlighted. Therefore, each stimulus lasts 300 ms. EEG data were recorded in 2 sessions, one training and one test. The training session contained 14 runs (characters) with 2 Turkish words (KALEM_YOLCULUK), and the test session contained 26 runs with 4 Turkish words (KITAP_MASA_AGLAMAK_SIKINTI). The data were recorded using the Biosemi ActiView software. We used Fp1, Fp2, P3, P4, PO7, PO8, Fz, Cz, Pz and Oz electrodes. For details about data pre-processing methods, hardware setup and the stimulus software setup please check [82].

4.3.2 Results

We have studied two main problems of P300 spellers: (1) Low accuracy results caused by using a small amount of training data, (2) Low performance of the system which is trained with a different subjects.

For the first scenario, the number of trial groups is set to 5 to allow the classifier performance to improve with adaptation. Previously, the trial group is set to 15 for the static classifiers used by [23], [32]. Higher number of trial groups means larger training data for the same number of letters. Even though we decrease the number of training letters, 15 trial groups result in good performance results even without adaptation. To be able to emphasize the performance improvement, we decided to have 5 trial groups. We have started with an initial training set including one letter and updated the classifier as we include letters one by one using the labels obtained from the classifier. In the process, the training data is extended to include all letters tested by the classifier. The results of this process are analyzed by increasing the initial size of the training data one by one until it includes all 14 characters.

Fig. 4.2-4.6 show the impact of adaptation on various classifiers as a function of the size of the initial training data. In particular, Fig. 4.2 presents the accuracy results for the simple BLDA classifier, whereas Fig. 4.4-4.6 present the results with the use of language models with the forward, forward-backward, and the Viterbi algorithms, respectively. In these figures, the x-axis represents the amount of initial training data (number of characters) and the y-axis represents the accuracy of the classifier. The green curve indicates the accuracy of static classifier, the blue and red lines indicate the accuracy of adaptation with the estimated labels and the true labels, respectively. The accuracy is calculated by averaging the accuracy of each letter used as test data. Here, classification results with adaptation and without adaptation were compared. In the adaptation method, we have examined two different cases; where classifiers are updated with true labels of the test data (which of course would not be available in practice – this was done to obtain an "upper bound" on performance that can be achieved by adaptation) or estimated labels. Based on our results, the improvement of the performance with adaptation is higher for smaller amount of training data. As expected, adaptation with true labels results higher accuracy values with respect to adaptation with estimated labels. As seen in Fig. 4.2 when initial training data is only 1 letter, without a language model, the average accuracy of P300 is 20% and is increased to 36% with adaptation using estimated labels. If we combine the system with a language model using the Viterbi algorithm, the accuracy of adaptation approach increases to 50% (Fig. 4.5). Results in Fig. 4.5 suggest that, using language models with adaptation improves the accuracy of the system, and Viterbi algorithm performs best among these three algorithms. Starting with an initial training data consists of 14 letters, the average accuracy with adaptation reaches 75% for forward algorithm, 78% for forward-backward, and 82% for Viterbi.

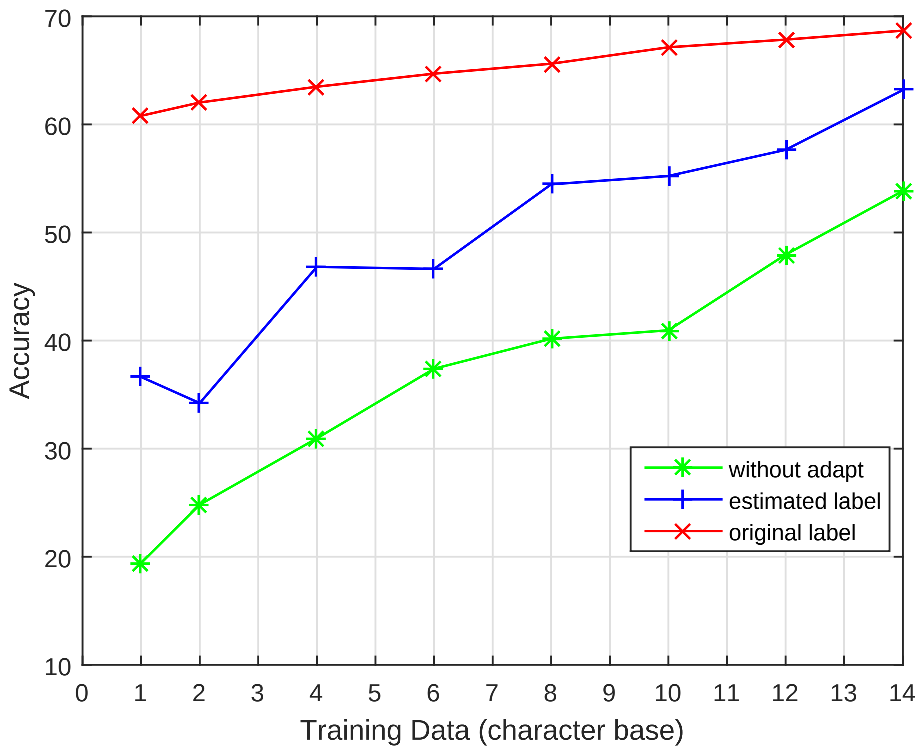

Figure 4.2: Impact of adaptation as a function of the training data size for the BLDA classifier.

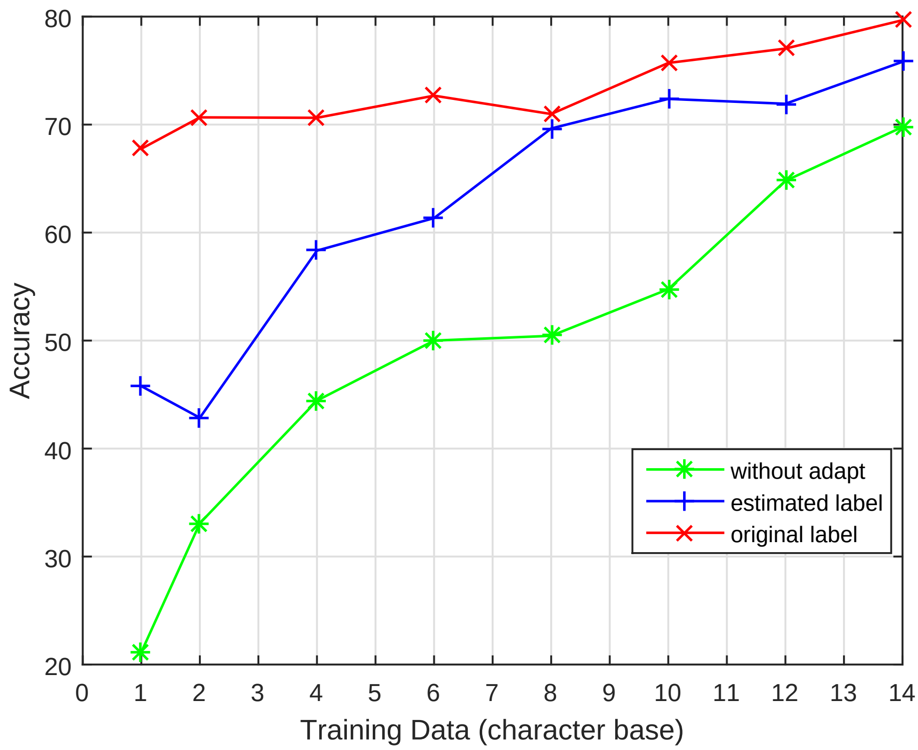

Figure 4.3: Impact of adaptation as a function of the training data size for the language model based classifier with the forward algorithm.

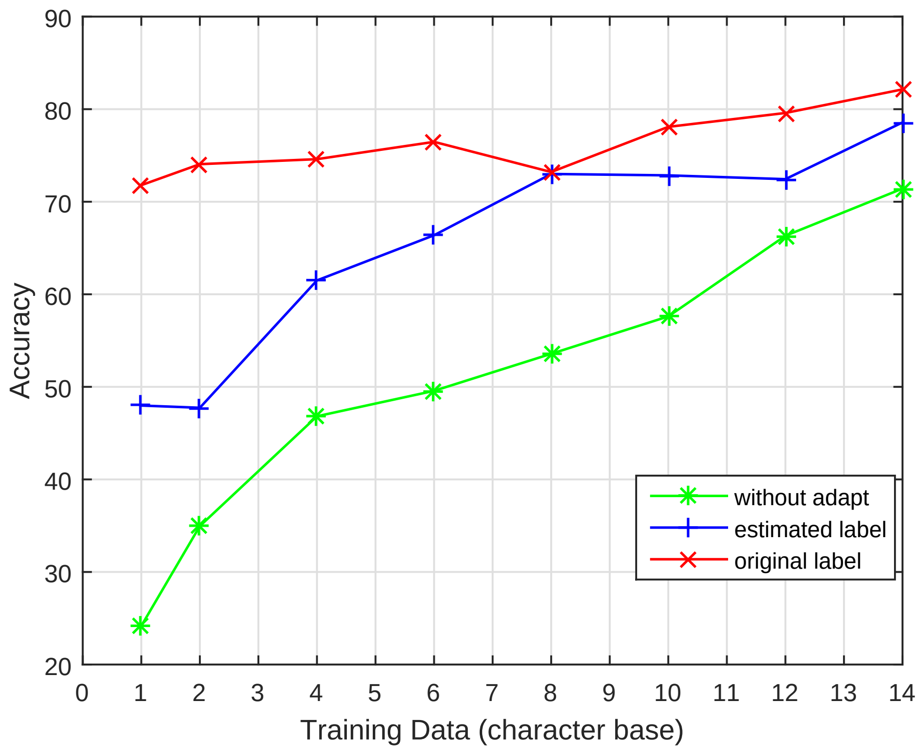

Figure 4.4: Impact of adaptation as a function of the training data size for the language model based classifier with the forward backward algorithm.

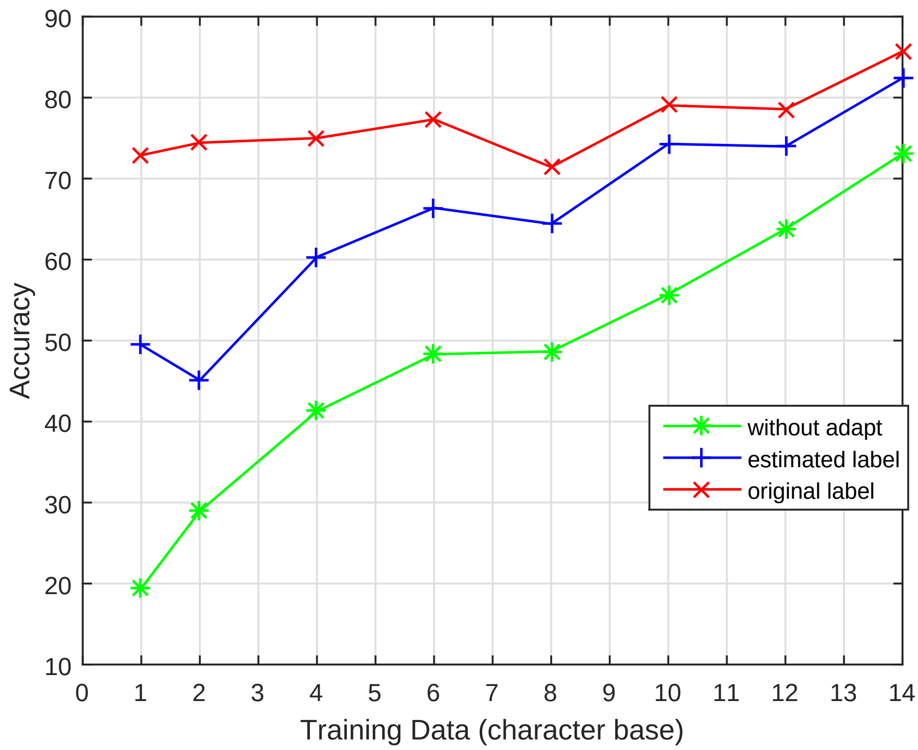

Figure 4.5: Impact of adaptation as a function of the training data size for the language model based classifier with the Viterbi algorithm.

In the second experiment, our subject-to-subject adaptation approach is tested. The accuracy results are very low with this type of experiment using static classifiers. Therefore, the trial group size is set to 15 to obtain reliable results for different users. In this experiment, training and test data are collected from 7 different users. Training the classifier with one user's training data, test data is used from 6 different users resulting in 42 matches. The average classification accuracy of 42 experiments can be seen in 4.1 for all classification algorithms considered. Using the Viterbi algorithm, the average accuracy of subject-to-subject classifiers is 39% without adaptation, and it increases to 73% using estimated labels for adaptation. We have also analyzed individual results for each subject pairs. While adaptation improves the performance for all subject pairs, we observe that different subject pairs exhibit different levels of performance. Some of the individual results for different subjects can be seen in Table 4.2 - 4.4. It seems that, some users' data are more compatible, for example in Table 4.4 classifier trained with data from U5 user performs well on the test data from user U6 even without adaptation. Like our first experiment, adaptation with true labels yields best accuracy results, and using language models with adaptation improves the performance as expected. For most of the experiments, the best accuracy is achieved with Viterbi algorithm.

Method	Without Adaptation	Estimated Label	Original Label
BLDA	28.64	62.90	83.30
Forward	36.02	70.69	89.84
F-B	38.67	71.26	91.45
Viterbi	39.23	73.04	92.59

Table 4.1: Subject-to-subject adaptation results.

Method	Without Adaptation	Estimated Label	Original Label
BLDA	7.69	19.23	84.61
Forward	7.69	23.07	84.61
F-B	11.53	23.07	88.46
Viterbi	7.69	30.76	84.61

Table 4.2: Individual results for subject-to-subject adaptation where training data is from user U1 and test data is from user U2.

Method	Without Adaptation	Estimated Label	Original Label
BLDA	8	38.46	58.84
Forward	16	57.69	73.07
F-B	20	65.38	76.92
Viterbi	20	69.23	84.61

Table 4.3: Individual results for subject-to-subject adaptation where training data is from user U3 and test data is from user U4.

Method	Without Adaptation	Estimated Label	Original Label
BLDA	42.30	84.61	88.46
Forward	61.53	92.31	96.15
F-B	65.38	96.15	100
Viterbi	65.38	96.15	100

Table 4.4: Individual results for subject-to-subject adaptation where training data is from user U5 and test data is from user U6.

We have designed another experiment to analyze the effect of adaptation at each step. For this experiment, the trial-group is set to 15 and only 1 character is used for initial training; where test data consists of 8 fixed characters. The rest of the data (31 characters) are used for adaptation process. After that, we updated the classifier as we include letters one by one using the labels obtained from the classifier. At each step, we have observed accuracy as a function of the amount of data used for adaptation. In Fig. 4.6, the x axis represents the number of letters whose labels are estimated and used for adaptation, and the y-axis is the accuracy of the classifier. Here, blue and red lines indicate the accuracy of adaptation (forward language model) with estimated labels and true labels respectively. Based on our results it is seen that, performance increases with the increase of number of letters used for adaptation to a certain point.

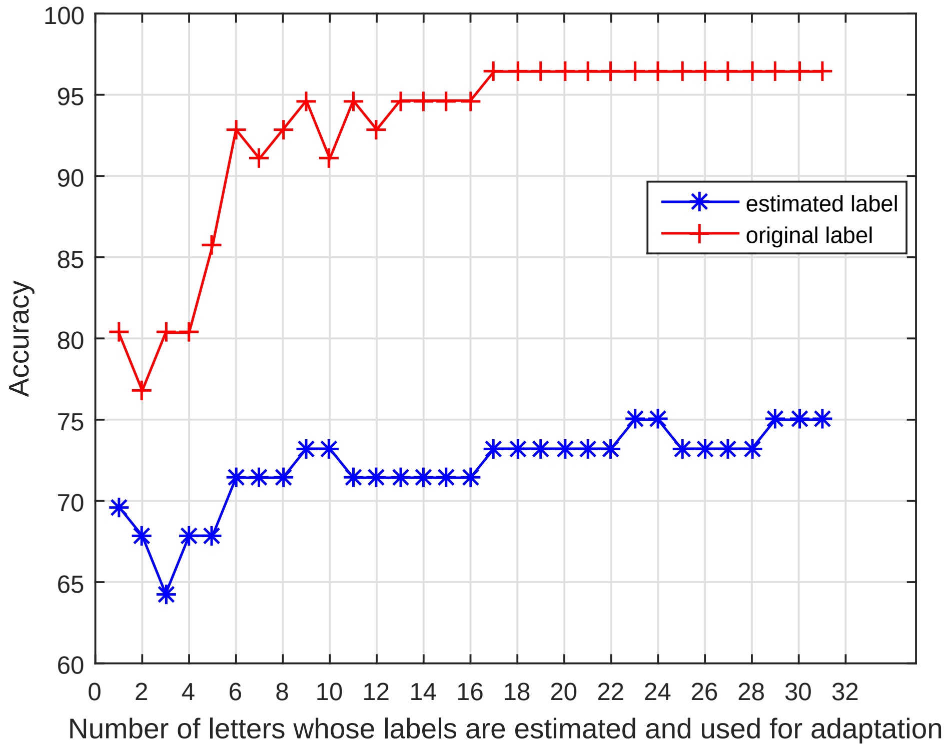

Figure 4.6: Accuracy as a function of the amount of data used for adaptation.

The goal of this section was primarily to assess the potential impact of adaptation on language-model based P300 spellers. In the case of limited amount of training data, it is shown that, the performance of the system increased approximately 17% without language models; and almost 30% with the inclusion of language models with respect to the static classifier. The difference between static and adapted classifiers decreased with the growing size of initial training data, but the adapted classifier continued to show better performance. In subject-to-subject BCI experiments, the adapted classifier obtained 34% more accurate results than the static classifier. Using language models improved the accuracy to 73.04% with estimated labels, to 92.59% with true labels. We have explored the performance of adaptation using true labels to get an idea of the experimental upper bound one can hope to achieve by adaptation. Estimated labels from the classifiers are not always the true labels, which means we sometimes feed wrong information to the system. It would be interesting to somehow predict when classifier output is not correct. To that extent, one idea could be the use of error-related potentials, which is a topic for our future work. Error related potentials occur when the user detects an error in the process. We expect such an approach would improve the performance obtained by adaptation through estimated labels we presented in this work.

4.4 ErrP based BCI system

As we have discussed in Chapter 2, sometimes BCI systems misidentify the users intent. Researchers have tried different approaches to improve system performance. One of the popular methods is to recognize these errors and correct them [40]. Another approach is detecting errors made by the system with the help of ErrP signals and updating the classifier accordingly [57], [26]. In this section, we investigate the potential use of ErrP signals in the P300-based BCI systems. The detection and classification of ErrP signals in BCI setting is presented along with the experimental analysis of ErrP. An example ErrP signal recorded during an experiment at the SPIS Laboratory can be seen in Fig. 4.7. In this figure, the x-axis represents the elapsed time (in milliseconds) after delivery of feedback, y-axis shows the range of signal in [-6 6]μV. Red curve represents the error signal and blue curve represents the correct signal.

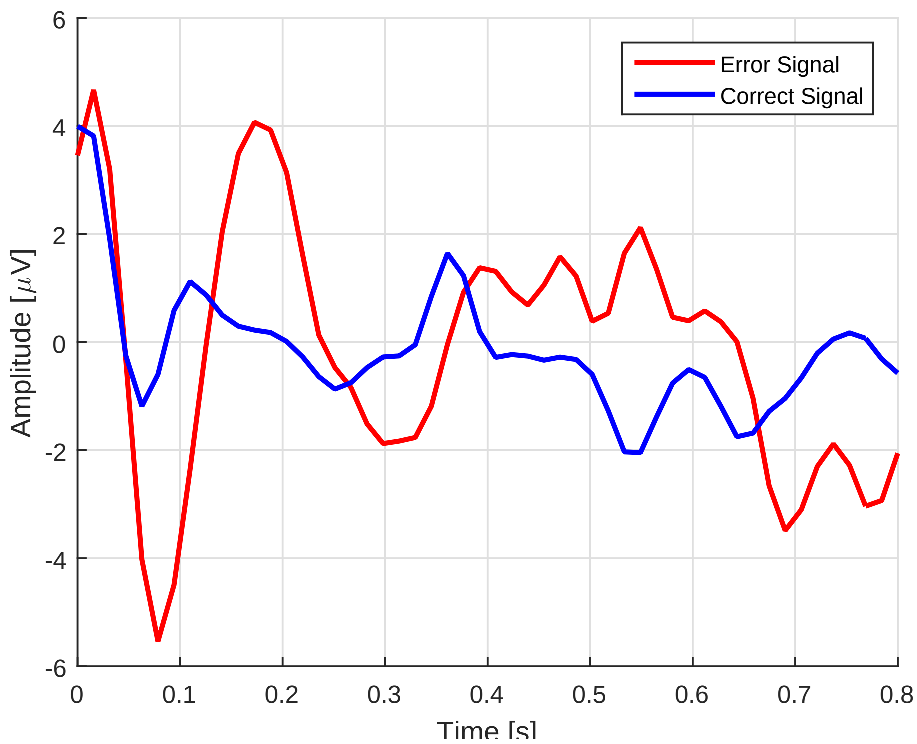

Figure 4.7: Averages of error and correct trial.

4.4.1 Processing ErrP dataset

The data set used in this study has been collected at the SPIS Laboratory with the approval of the Sabanci University Research Ethics Council. It is collected from 8 healthy subjects (ages between 18 and 25) during offline spelling experiments. The procedure was explained to the subjects before the experiment, and their informed consent was obtained. Data were sampled at 2048 Hz. A P300-ErrP interface is developed to record ErrP signals during spelling experiments. Inter-stimulus interval (ISI) was 125 ms and stimulus duration was 50 ms for all conditions. EEG data were recorded in 2 sessions: training and test session. The training session contained 200 runs (characters), and the test session contained 200 runs. The data were recorded using the Biosemi ActiView software. Data pre-processing methods and hardware setup are described in previous work by Amcalar and Cetin [32]. Common average reference is used and we applied 1-10 Hz band-pass filter.

In the experiments, we have used the P300-speller protocol described in Section 4.2. We have told subjects what to type before the experiment, to be able to recognize errors and have a ground truth. Subjects were led to believe that BCI system selects characters based on subjects' P300 signals. But in reality, BCI selected the correct letter with a probability of 80% (Pe=0.2), and it did not consider the P300 signals at all. We chose an 80% probability because it is reasonable for such speller systems [40].

We have updated the SU-BCI P300 stimulus interface [32] to detect and process ErrP signals. The experiments are conducted with the SU-BCI P300-ErrP stimulus in 3 main steps:

- Target step: Before each run, the target character is presented in grey color in the matrix (for 3 seconds duration) to show the location of the character in the matrix (Fig. 4.8).

- After that, rows and columns of the matrix start flashing in a block-randomized fashion. The subject is instructed to count the number of flashes of the target character. Interface shows black screen (for 1 second duration) between matrix and feedback steps (Fig. 4.9).

- Feedback step: After the flashes, interface presents the target letter (for 3 seconds duration) with an accuracy of 80%, in green color (Fig. 4.10).

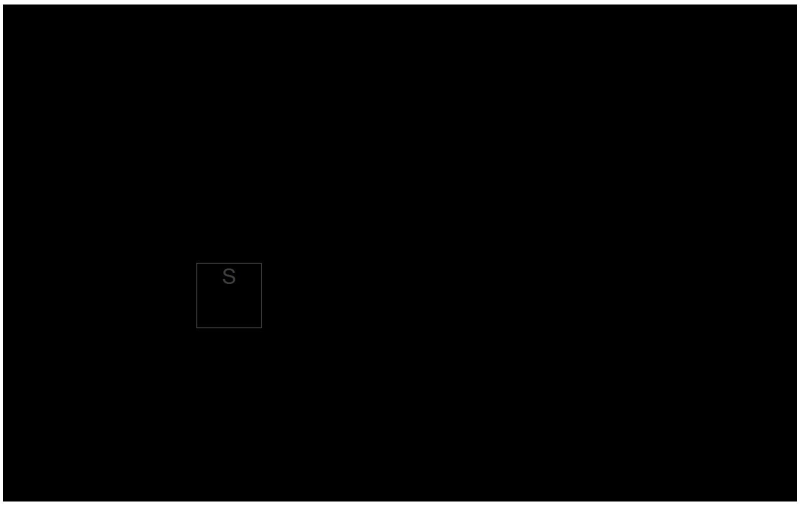

Figure 4.8: The target character “S”.

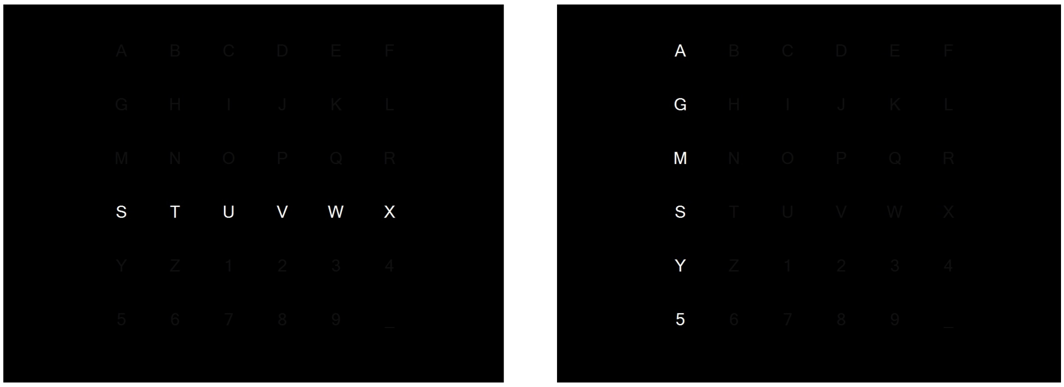

Figure 4.9: Matrix flashes.

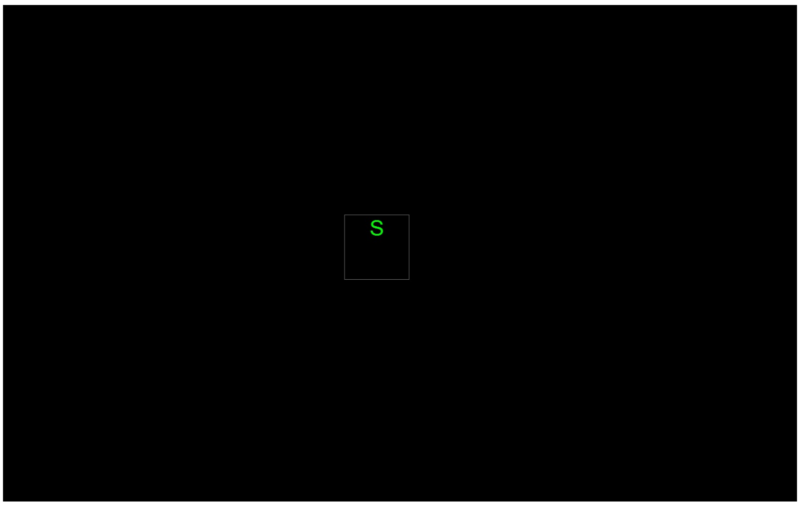

Figure 4.10: Feedback character “S”.

4.4.2 ErrP Classification

We have analyzed EEG signals following erroneous and correct trials to detect ErrP. We have used a customized Gaussian classifier to decide whether an ErrP has occurred or not based on Millans work (see [28], [83] for details). Gaussian classifier separates "correct" and "error" trials, by estimating the posterior probability of ErrP occurrence.

In this Gaussian Mixture Model, a total number of 6 prototypes (Gaussians) were used to represent each class. Conditional probability density function of a class, C_k is a superposition of the prototypes. The activity a_k^i of the i^{th} prototype of the C_k for the input vector, x derived from a trial is:

$$a_k^i(x) = |\Sigma_k|^{-1/2} \exp(-1/2(x - \mu_k^i)^T {\Sigma_k}^{-1}(x - \mu_k^i)) \tag{4.5}$$

Here, center of the i^{th} prototype of the C_k is μ_k^i. Each prototype has its own covariance matrix Σ_k. The posterior probability y_k is:

$$y_k(x) = p(x|C_k) = \frac{a_k(x)}{A(x)} = \frac{\sum\limits_{i=1}^{N_p} a_k^i(x)}{\sum\limits_{k=1}^{2}\sum\limits_{i=1}^{N_p} a_k^i(x)} \tag{4.6}$$

In this formula, A is the total activity. We used kmeans to initialize the center of the prototypes, μ_k^i. In order to reduce the number of parameters, we restrict our model to a diagonal covariance matrix:

$$\Sigma_k = \frac{1}{|S_k|} \sum_{x \in S_k} (x - \mu_k^{i*})(x - \mu_k^{i*})^T \tag{4.7}$$

Here, the set of the training samples belonging to the class C_k is S_k. The cardinality of this set is $|S_k|$, and $i*$ is the nearest prototype of this class to the sample x.

We have tried to improve the initial estimations iteratively to minimize the mean square error during learning. Gradient of the error functions:

$$\Delta\mu_k^i(x) = \alpha \frac{dE}{d\mu_k^i}(x) = \alpha \frac{a_k^i(x)}{A(x)} \frac{(x - \mu_k^i)}{\Sigma_k} e_k(x) \tag{4.8}$$

$$\Delta\Sigma_k^i(x) = \alpha \frac{dE}{d\Sigma_k^i}(x) = \beta \frac{a_k^i(x)}{A(x)} \frac{\left(x - \mu_k^i\right)^2}{\left(\Sigma_k^i\right)^3} e_k(x) \tag{4.9}$$

α and β are the learning rates and e_k is error function. For each training trial, after updating μ_k^i and ${\Sigma_k}^i$, the covariance matrices of all the prototypes of the same class are averaged to obtain the common class covariance matrix Σ_k.

4.4.3 Experiment and results

In this study, we have proposed a statistical classifier to detect ErrP signal. Fz, FCz and Cz electrodes are used according to the standard 10/20 international system for this purpose. Window range following the feedback was selected independently for each subject based on the classification performance. ErrP, generally observed within 150-600 ms window range after visual stimuli is displayed and this window range is user-dependent [28]. To separate correct and error trials successfully, the window range with highest performance is selected using cross-validation.

In our experiments, firstly we have generated the same number of correct and error trials (Pe=0.5) to have enough samples representing error trials. Table 4.5 shows the accuracy of detecting ErrP signals for the first experiment. We have collected data from 4 users, and we have used Cz and FCz electrodes. Following the feedback, data between 150-450 ms are used. Here, the classifier was not able to detect ErrP signals efficiently. Since there was a system error of 50%, we believe users questioned the competency of the system and they lost their motivation during the experiment. Therefore they did not generate ErrP for the error trials. Highest accuracy was achieved with subject 4 (from FCz electrode and 200-450 ms time window). One of the main challenges designing these experiments was deciding the duration of the experiment. Firstly, we set our experiments to have more than 200 runs to collect as much data as possible. Long experiments caused users to lose their motivation, and even sleep. Therefore, we have fixed the number of runs to 200 for our training and test sessions.

	subject 1	subject 2	subject 3	subject 4
Electrodes	FCz	Cz	Cz	FCz
Time window (ms)	200-300	200-310	150-300	200-450
Correct (%)	55	65	56	61
Error (%)	54	50	61	60

Table 4.5: Results of ErrP experiment when Pe=0.5.

In our second experiment, we fixed the error rate to 20%, Pe=0.2. When we compare accuracy of the classifier in Table 4.6 with Table 4.5, it is seen that, the detection performance is improved. Here, data are collected from 4 users and we have used Cz and FCz electrodes. We choose time windows between 250-560 ms. For subject 5 (from FCz electrode and 250-400 ms time window), the classifier was able to detect with accuracy of error 68% and for subject 6 (from FCz electrode and 230-380 ms time window), the classifier was able to detect with accuracy of correct 78%.

	subject 5	subject 6	subject 7	subject 8
Electrodes	FCz	FCz	Cz	Cz
Time window (ms)	250-400	250-380	250-350	450-560
Correct (%)	60	78	71	72
Error (%)	68	60	61	65

Table 4.6: Results of ErrP experiment when Pe=0.2.

4.4.4 Implementation of an ErrP classification to a P300 speller system

This chapter focuses on online analysis of P300 and ErrP signals. In Chapter 4.1-4.5, we researched the capabilities of offline analysis, where the brain signals are recorded first and the analysis is performed separately afterwards, allowing one to do various kinds of analyses on the data. In online analysis, brainwaves are pre-processed and analyzed concurrently, as the recording continues [32].

Firstly, in offline work, P300 and ErrP classifiers are trained by using the ErrP stimuli. After that online test is started. During the online spelling analysis, data are first divided into epochs and then filtered and processed locally in epochs, instead of filtering and processing as a whole and then dividing into epochs, which is what we did during the offline analysis [35]. In ErrP online analysis, we set the time windows of online classifier same as time windows of offline classifier. Data were sampled at 2048 Hz and ISI set 125 ms and stimulus duration set 50 ms. Common average reference is used and 1-10 Hz band-pass filter is applied.

Subject's aim is to select the target letter and focus on it when matrix flashes. In the beginning of the experiment, a 6x6 matrix (Fig. 4.11) is shown to user for choosing character to focus on. After that matrix is flashed. At the end of each trial group (EEG score for each of the 12 row/column flashes are obtained), EEG scores and n-gram probabilities calculated from the text corpus and they are fed into HMM. Posterior probability scores of each letter were obtained by either Forward, Forward-backward, or Viterbi decoding. The most probable letter is displayed (Fig. 4.12) as the answer of the classification [35]. At this point, if feedback character is the same character that user focused, user would not produce error signal. If feedback character is wrong, we expect to see an error signal.

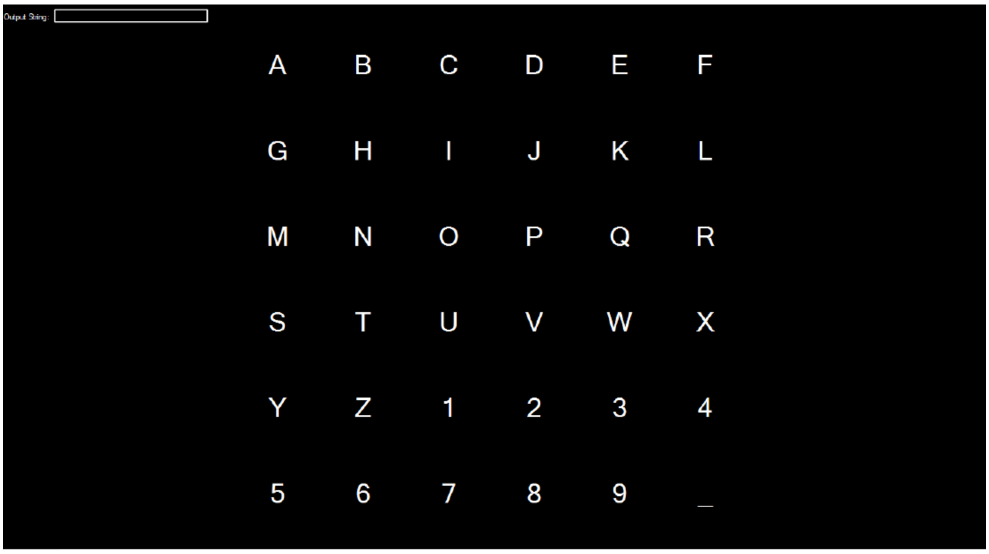

Figure 4.11: 6x6 Matrix is showed to user.

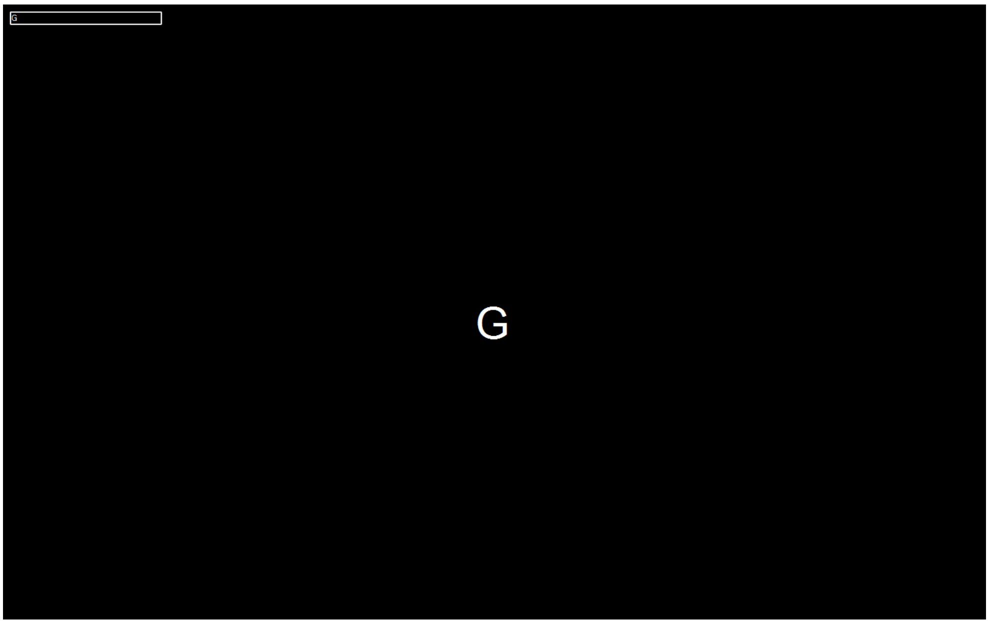

Figure 4.12: Feedback character.

Since the optimum time range to detect ErrP signals is different for each subject, one would need to conduct a number of experiments to obtain reliable results. We did not have time to conduct many online ErrP experiments, hence so we are not presenting our online-ErrP results. We made controlled experiments in offline mode that is presented in the previous section. With sufficient number of samples from online experiments, it is expected that we would obtain similar results. Our aim for online-ErrP work is building infrastructure for ErrP based adaptation. Our approach for adaptation ErrP signals is described in Chapter 5.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we have presented an adaptation approach for motor imagery BCI systems and P300 spellers, and we have proposed, designed, implemented and evaluated the use of ErrP for the P300 speller BCI systems. We have developed experimental scenarios to adapt motor imagery and P300 systems, and to integrate ErrP classifier in the P300 systems.

We have proposed a semi-supervised adaptation method to address non-stationary nature of EEG signals. Limited amount of training data and subject dependent classifiers are two other problems of BCI systems we have investigated. We have used data collected in our laboratory as well as BCI competition datasets. We use the estimated labels from the classifier to retrain the classifier for adaptation. We have compared the results of static and adapted classifiers. Using EEG data from motor imagery and P300 experiments, we have shown that the proposed approach improved the accuracy of the classifier for all three scenarios mentioned in Section 3 and Section 4. We have also analyzed the results of supervised adaptation using true labels instead of estimated labels. These results point us the highest accuracy one can achieve using proposed adaptation approach. For the limited amount of training data, we see that improvement with adaptation is more obvious for smaller sizes of initial training data. Classifier performances are improved 4% with semi-supervised adaptation methods in motor imagery experiments for limited amount of training data case.

The proposed approach is applied on a P300 speller which incorporates a language model using HMM. To the best of our knowledge, adaptation of a language model based P300 speller as we propose in this work has not been studied before. In this work, we collected our data sets in SPIS laboratory (7 healthy subjects) using SU-BCI P300 Speller. In limited data experiments, our results show that, the performance of the system increased approximately 17% without language models, and almost 30% with the inclusion of language models with respect to the static classifier (BLDA). In the case of subject-to-subject BCI experiments, system performance has improved 34% with adapted classifier. We have observed that some of the subject pairs showed higher performances with respect to other pairs, which would mean the EEG data of those users are more compatible with each other.

We have suggested that, using ErrP signals in P300 system can improve the performance of the system. We have developed an ErrP classifier and analyzed the performance. We were able to detect ErrP signals with an accuracy of approximately 65%.

5.2 Future work

As a future direction, we are planning to adapt BCI systems with the help of ErrP signals.When the classifier detects the error signal, we understand that the system or user made a mistake. For instance, in P300 speller system, we can add backspace in the speller matrix so when there is a mistake, user can use backspace and correct his/her character. Another approach is, using semi-supervised adaptation that we have suggested in Chapter 3 and Chapter 4. In semi-supervised adaptation method, we re-train our classifier with estimated or true labels. Here, we are planning to update the classifier using the information provided by the ErrP signals. When the ErrP classifier detects an error, it would mean the output of the P300 classifier is wrong. In that case, updating the system with the estimated label would mean feeding the wrong information to the system. We believe that by restricting the use of estimated labels to the time when there is no error detected would further improve proposed semi-supervised adaptation approach and bring the performance closer to the supervised adaptation using true labels.

In literature, Blankertz [55], [56] worked on error detection and suggested that it could serve as an efficient online confirmation/correction tool for improvement of bit rates in a future BCI setting. Researchers have explored the use of ErrP signals in motor imagery applications. Zeyl et al. [84], [85] worked on how the ErrP can be incorporated into an adaptive classification method. According to this work, event-related potentials incorporated with BCI errors can have the potential to be used as online labels for adaptation of the classifier. In their motor imagery experiments after ErrP signal is observed they adapt the mean and covariance parameters of the LDA classifier.

In our studies, we used adaptation methods on motor imagery experiments but we did not use ErrP for adaptation. According to the literature, using error feedback can be suitable in our motor imagery tasks. In the proposed adaptation method which is explained in Chapter 3, the BCI system is updated with labels (estimated or true labels) obtained from the outputs of the classifier. Here, when ErrP signal is detected, assuming ErrP classifier is accurate, one can conclude that the output of the BCI classifier is wrong. For binary motor imagery tasks, it is easy to estimate the correct label and update the classifier using that information. Moreover, online data label can be changed with observing ErrP signal and class means and variances can be updated. This approach can be suitable in our motor imagery experiments.

Combaz [59], proposed to improve the classification accuracy of a P300 speller by incorporating ErrP information. They introduce a 6×6 speller matrix which has backspace symbol. In their system, if any error is detected by ErrP detector in the feedback period, system overrides the P300 speller and cancels the last selection. If ErrP signal is not detected, system will not override. In that case, user can focus on backspace symbol and he/she can cancel last character and continue. This backspace system is used by Delsano as well [70]. Combaz et al. suggested another approach for improving P300 classifier performance by using ErrP signals. In this approach, the classification system produces a score for each row and column, and it choses the column and row with the highest score. From those scores, a ranking of all the symbols of the matrix can be deduced. After ErrP is detected, the sequence of intensifications is repeated and ranking is updated. This approach can take significant time. Their second suggestion is, when ErrP is detected, to select the second best symbol according to the classifier's ranking. Moreover, Spuler et al. [86]

recognize ErrP signals during operation of a P300 speller, and on detection of ErrP, the interface informs users that the incorrect letter was automatically deleted. After that, the system erases the last character and system starts to flash again. Spuler tried to find the best classifier for ErrP detection by using 10-fold cross-validation. SVM with radial basis function (RBF) kernel is found to be the best classifier among LDA, SWLDA and SVM. Error correction system (ErrP detection and correction system) needs a long training session. Spuler et al tried to decrease this duration by decreasing the inter-trial interval; this approach succeeded in 2 groups out of 3.

Mattaut [87] and Margaux [88] implemented a P300-based BCI speller system, including online error detection and automatic correction. Here, when ErrP signal is detected, a new decision is made based on the second best guess of a probabilistic classifier. According to them, automatic correction has a higher bit rate than a respelling strategy. Multidimensional Gaussian Mixture Model is used for classifying ErrP signals. Furthermore, Zeyl [89] presents a combination of P300-speller and ErrP for improving online error detection and correction accuracies. They developed a system to combine ErrP scores with P300 confidence levels derived from a ranking system based on a Gaussian naive Bayes estimate of the probabilities of rows and columns. A random forest is built for combining P300 and ErrP information. In their stimulus system, firstly, each row is flashed in a pseudorandom order. User's task is to focus on the character he/she wants to spell and mentally count the number of flashes of the target character. Following that, feedback is given to the user by replacing the characters in all rows with those from the selected row. If an error is observed, the user has to continue to focus on the target character. At this point, if the output is the same, it means that the correct row was selected, but ErrP is misclassified. If the letter is changed, this signifies that the incorrect row was selected. These two steps are applied for columnwise. In the end, the target character is selected. Even tough this is a well designed approach to use ErrP signals for P300 adaptation, it takes a long time to spell a word.

Some of the approaches mentioned above can be applied to our system. Combination of the language model with the use of ErrP signals for adaptation would furthermore improve P300 speller performance.

Additionally, the P300 system in this project is using BLDA classifier which produces scores for all characters in 6×6 matrix in each trial. When another row or column is flashed, BLDA accumulates new scores on previous ones. Output of the classifier is determined based on these scores. An alternative approach is when ErrP symbol is detected, instead of stopping the process and correcting the character, to repeat the sequence of intensifications, and update scores with this new ranking. In general the ErrP classifier needs a long training session, sometimes ErrP classifiers can not work accurately. We can calculate a "certainty measure" for the output of P300 classifier. After training the P300 classifier, we can compare the score of the first ranked letter with other letters' scores. Using these scores, a "certainty measure" can be calculated. This "certainty measure" can be used as a prior probability of error to weight the results of the ErrP classifier. These ideas (inspired by the work in [59]) can be suitable for semi-supervised adaptation methods described in chapter 3 and chapter 4.

Another way of using information provided by ErrP signals is post analysis and post correction. In online experiments, after P300-ErrP interface shows the feedback to the user, the character is written at the top of the screen (in an output string box) and user can follow his/her spelling from there. Also, "_" character is used as space. In another potential line of work, until user spells the "_" character, the feedback characters are not written in the output string box. If ErrP is detected at some point during the spelling of the whole word, the system decides there is a mistake. At this point, by updating the probability weights of the character where ErrP is detected and by using our probabilistic language models, system can estimate the most likely word and it can be shown in output string box to the user.

In the proposed semi-supervised adaptation method, classifier is re-trained by using estimated or true labels. In this process, when ErrP signal is detected after feedback, we would not update the classifier with the estimated label and do not write that character to the output string box. Here our suggestion is, classifier can be updated with second most likely character as the estimated label. Following the ErrP signal, going back to the speller matrix and letting the system continue flashing rows and columns as before, the algorithm can compute and add the new scores to update the last character's scores. After updating the scores, new character can be written to the output string box.

Appendix A

Language Model

In this appendix, we describe the basics of the language model-based inference algorithms used in this study. The description here is based on Ulas's thesis [35].

Viterbi and Forward-Backward algorithms use all available data for estimating the character at a specific time instant. Therefore, these algorithms are called "smoothing" recursive estimation algorithms.

A.1 Forward-Backward Algorithm

Let m_t indicate the state at time t, and $t \in \{1, 2, ..., T\}$. Here $S = s_1 s_2 .... s_T$ is the observation sequence where each s_k indicates scores of all possible symbols for kth letter of the word. First part of the forward-backward algorithm is, computing a set of forward probabilities for all $t \in \{1, 2, ..., T\}$. The joint probability of the partial observation sequence until time t, (i.e., $s_{1:t}$) and the state at time t (i.e., m_t) is defined by this part. After that, the algorithm computes backward probabilities which provides the probability of the partial observation sequence from $t + 1$ to T, given the state i at time t. Afterwards, two cases of probabilities are combined to estimate the probability distribution over states at any particular time as follows:

$$P(m_t = i | s_{1:T}) \propto P(s_{1:t}, m_t = i) P(s_{t+1:T} | m_t = i). \tag{A.1}$$

Here, $P(s_{t+1:T} | m_t = i)$ indicates forward probability at time t and $P(s_{1:t}, m_t = i)$ term indicates backward probability at time t. Let us denote first term as $\varphi_t(i)$ and second term as $\phi_t(i)$. These terms can be recursively computed using following equations for an N-th order HMM:

$$\varphi_1(i_1) = P(m_1 = i_1)P(s_1|m_1 = i_1) \tag{A.2}$$

$$\varphi_t(i_t) = [\sum_{i_{t-N}} \dots \sum_{i_{t-1}} \varphi_{t-1}(i_{t-N}, ..., i_{t-2}, i_{t-1}) a_{i_{t-N},...,i_{t-1},i_t}] P(s_t|m_t = i_t) \tag{A.3}$$

where $a_{i_{t-N},...,i_{t-1},i_t} = P(m_t = i_t|m_{t-N} = i_{t-N}, ..., m_{t-1} = i_{t-1})$, $1 < t \leq T$ and each $i_{t-N}, ..., i_{t-1}, i_t \in S$.

For determining the backward probabilities, $\phi_T(i_T) = 1$ and $\phi_t(i_t)$ is described for $T - 1 \geq t \geq 1$ as follow:

$$\phi_t(i_t) = [\sum_{i_{t+1}} \dots \sum_{i_{t+N}} \phi_{t+1}(i_{t+1}, i_{t+2}, ..., i_{t+N}) a_{i_t,i_{t+1},...,i_{t+N}} P(s_{t+1}|m_{t+1} = i_{t+N})]. \tag{A.4}$$

$P(s_t|m_t = i)$ can be determined for each $t \in \{1, 2, ..., T\}$ and for any number of available trial groups N_t, if we assume all scores of one run are conditionally independent given the class labels:

$$P(s_t|m_t = i) = \prod_{n=1}^{N_t} p(s_t(i, n)|cl_i)(\prod_{n=1}^{N_t} \prod_{i' \in S \setminus \{i\}} p(s_t(i', n)|cl_{i'})). \tag{A.5}$$

Let $s_t(i, n)$ and $s_t(i', n)$ denote the scores including the letter i at the nth trial group and those not including the letter i respectively. Here, cl_i is the class label, the study in [35] has confirmed $p(s_t(i, n)|cl_i = 1)$ and $p(s_t(i', n)|cl_{i'} = -1)$ are normally distributed by analyzing the distribution of the training data scores. We have estimated the parameters of Gaussian densities to the test scores for attended and not-attended classes from training data scores, where $cl_i = 1$ for attended epochs and $cl'_i = 1$ for non-attended epochs in the training set.

An n-gram language model is used for estimating the initial probability, $\pi_i = P(m_1 = i)$ and the transition probabilities $a_{i_{t-n_{lm}+1},...,i_{t-1},i_t}$. Here, n_{lm} is the order of the language model.

Having calculated the forward and backward probabilities based on above equations, the probability of being in state i at time t given the all observation sequence S can be expressed as follows:

$$P(m_t = i|s_{1:T}) = \gamma_t(i) = \frac{\varphi_t(i)\phi_t(i)}{\sum\limits_i \varphi_t(i)\phi_t(i)} \tag{A.6}$$

The individually most likely state or character at any time t where $1 \leq t \leq T$ can be estimated as:

$$\hat{m_t} = \arg\max_i[\gamma_t(i)]. \tag{A.7}$$

A.2 Viterbi Algorithm

Viterbi Algorithm finds the most likely letter sequence by maximizing the joint probability of a letter sequence given the EEG data. The required state transition probabilities and observation symbol probabilities for this algorithm were already produced in the Appendix A.1 section. Viterbi algorithm creates the most probable letter sequence of a corresponding word for multiple-trial EEG data for each letter in the sequence.

Bibliography

[1] Biosemi Headcaps, http://www.biosemi.com/headcap.htm.

[2] G. Pfurtscheller and F. L. Da Silva, "Event-related EEG/MEG synchronization and desynchronization: basic principles," *Clinical neurophysiology*, vol. 110, no. 11, pp. 1842–1857, 1999.

[3] User Tutorial:Introduction to the P300 Response, http://www.bci2000.org/wiki/index.php/User_Tutorial:Introduction_to_the_P300_Response.

[4] H. Gürkök and A. Nijholt, "Brain–computer interfaces for multimodal interaction: A survey and principles," *International Journal of Human-Computer Interaction*, vol. 28, no. 5, pp. 292–307, 2012.

[5] J. A. Jacko, *Human-Computer Interaction: Interaction Techniques and Environments: 14th International Conference, HCI International 2011, Orlando, FL, USA, July 9-14, 2011, Proceedings.* Springer, 2011, vol. 6762.

[6] J. DiGiovanna, B. Mahmoudi, J. Fortes, J. C. Principe, and J. C. Sanchez, "Coadaptive brain–machine interface via reinforcement learning," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 1, pp. 54–64, 2009.

[7] I. Yilmaz, S. Demir, T. Tasdizen, and M. Cetin, "Semi-supervised adaptation of motor imagery based BCI systems," in *Signal Processing and Communications Applications Conference (SIU), 2015 23th.* IEEE, 2015, pp. 1841–1844.

[8] A. M. Dale and E. Halgren, "Spatiotemporal mapping of brain activity by integration of multiple imaging modalities," *Current opinion in neurobiology*, vol. 11, no. 2, pp. 202–208, 2001.

[9] A. M. Dale, A. K. Liu, B. R. Fischl, R. L. Buckner, J. W. Belliveau, J. D. Lewine, and E. Halgren, "Dynamic statistical parametric mapping: combining fMRI and MEG for high-resolution imaging of cortical activity," *Neuron*, vol. 26, no. 1, pp. 55–67, 2000.

[10] G. Dornhege, *Toward brain-computer interfacing.* MIT press, 2007.

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

[12] G. R. Müller-Putz, R. Scherer, G. Pfurtscheller, and R. Rupp, "EEG-based neuroprosthesis control: a step towards clinical practice," *Neuroscience letters*, vol. 382, no. 1, pp. 169–174, 2005.

[13] F. Galán, M. Nuttin, E. Lew, P. W. Ferrez, G. Vanacker, J. Philips, and J. d. R. Millán, "A brain-actuated wheelchair: asynchronous and non-invasive brain–computer interfaces for continuous control of robots," *Clinical Neurophysiology*, vol. 119, no. 9, pp. 2159–2169, 2008.

[14] G. E. Fabiani, D. J. McFarland, J. R. Wolpaw, and G. Pfurtscheller, "Conversion of eeg activity into cursor movement by a brain-computer interface (BCI)," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 12, no. 3, pp. 331–338, 2004.

[15] J. R. Wolpaw and D. J. McFarland, "Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 51, pp. 17 849–17 854, 2004.

[16] U. Hoffmann, J.-M. Vesin, T. Ebrahimi, and K. Diserens, "An efficient P300-based brain–computer interface for disabled subjects," *Journal of Neuroscience methods*, vol. 167, no. 1, pp. 115–125, 2008.

[17] E. W. Sellers, D. J. Krusienski, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "A P300 event-related potential brain–computer interface (BCI): the

effects of matrix size and inter stimulus interval on performance," *Biological psychology*, vol. 73, no. 3, pp. 242–252, 2006.

[18] S. L. Bressler and M. Ding, "Event-related potentials," *Wiley encyclopedia of biomedical engineering*, 2006.

[19] 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*, vol. 70, no. 6, pp. 510–523, 1988.

[20] R. C. Panicker, S. Puthusserypady, and Y. Sun, "Adaptation in P300 brain–computer interfaces: A two-classifier cotraining approach," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 12, pp. 2927–2935, 2010.

[21] S. Lemm, C. Schäfer, and G. Curio, "BCI competition 2003-data set iii: probabilistic modeling of sensorimotor μ rhythms for classification of imaginary hand movements," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1077–1080, 2004.

[22] Y. Li, H. Li, C. Guan, and Z. Chin, "A self-training semi-supervised support vector machine algorithm and its applications in brain computer interface," in *Acoustics, IEEE International Conference on Speech and Signal Processing, 2007. ICASSP 2007.*, vol. 1. IEEE, 2007, pp. I–385.

[23] C. Ulas and M. Cetin, "Incorporation of a language model into a brain computer interface based speller through hmms," in *Acoustics, IEEE International Conference on Speech and Signal Processing (ICASSP), 2013.* IEEE, 2013, pp. 1138–1142.

[24] N. M. Schmidt, B. Blankertz, and M. S. Treder, "Online detection of error-related potentials boosts the performance of mental typewriters," *BMC neuroscience*, vol. 13, no. 1, p. 19, 2012.

[25] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Müller, V. Kunzmann, F. Losch, and G. Curio, "The berlin brain-computer interface: EEG-based communication without subject training," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 147–152, 2006.

[26] J. Blumberg, J. Rickert, S. Waldert, A. Schulze-Bonhage, A. Aertsen, and C. Mehring, "Adaptive classification for brain computer interfaces," in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE.* IEEE, 2007, pp. 2536–2539.

[27] S. Lu, C. Guan, and H. Zhang, "Unsupervised brain computer interface based on intersubject information and online adaptation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 2, pp. 135–145, 2009.

[28] K Nearest Neighbors - Classification, http://www.saedsayad.com/k_nearest_neighbors.htm.

[29] P. W. Ferrez *et al.*, "Error-related EEG potentials generated during simulated brain–computer interaction," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 923–929, 2008.

[30] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "BCI2000: a general-purpose brain-computer interface (BCI) system," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.

[31] S. Sanei and J. A. Chambers, *EEG signal processing.* John Wiley & Sons, 2013.

[32] A. Amcalar, "Design, implementation and evaluation of a real-time P300-based brain-computer interface system," Master thesis, Sabanci University, 2010.

[33] Biosemi Active electrodes, http://www.biosemi.com/active_electrode.htm.

[34] E. Niedermeyer and F. H. L. da Silva, *EEG Recording and Operation of the Apparatus*, 5th ed. Lippincott Williams and Wilkins, 2005.

[35] C. Ulas, "Incorporation of a language model into a brain computer interface based speller," Master thesis, Sabanci University, 2013.

[36] J. J. Vidal, "Toward direct brain-computer communication," *Annual review of Biophysics and Bioengineering*, vol. 2, no. 1, pp. 157–180, 1973.

[37] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "Toward enhanced P300 speller performance," *Journal of neuroscience methods*, vol. 167, no. 1, pp. 15–21, 2008.

[38] L. J. Trejo, R. Rosipal, and B. Matthews, "Brain-computer interfaces for 1-D and 2-D cursor control: designs using volitional control of the EEG spectrum or steady-state visual evoked potentials," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 225–229, 2006.

[39] A. Nijholt, D. P.-O. Bos, and B. Reuderink, "Turning shortcomings into challenges: Brain–computer interfaces for games," *Entertainment Computing*, vol. 1, no. 2, pp. 85–94, 2009.

[40] B. D. Seno, "Toward an integrated P300 and ErrP based brain computer interface," Phd thesis, Politecnico di Milano, 2009.

[41] J. R. Wolpaw, D. J. McFarland, G. W. Neat, and C. A. Forneris, "An EEG-based brain-computer interface for cursor control," *Electroencephalography and clinical neurophysiology*, vol. 78, no. 3, pp. 252–259, 1991.

[42] B. Obermaier, G. Muller, and G. Pfurtscheller, "Virtual keyboard" controlled by spontaneous EEG activity," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, no. 4, pp. 422–426, 2003.

[43] A. S. Royer, A. J. Doud, M. L. Rose, and B. He, "EEG control of a virtual helicopter in 3-dimensional space using intelligent control strategies," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 6, pp. 581–589, 2010.

[44] F. Galán, M. Nuttin, E. Lew, P. W. Ferrez, G. Vanacker, J. Philips, and J. d. R. Millán, "A brain-actuated wheelchair: asynchronous and non-invasive brain–computer interfaces for continuous control of robots," *Clinical Neurophysiology*, vol. 119, no. 9, pp. 2159–2169, 2008.

[45] S. J. Luck, "Event-related potentials," *APA handbook of research methods in psychology*, vol. 1, pp. 523–546, 2012.

[46] M. Wang, "Design of a modified P300 speller system based on prediction by partial matching language model," Ph.D. dissertation, University of Cincinnati, 2012.

[47] B. Blankertz, M. Krauledat, G. Dornhege, J. Williamson, R. Murray-Smith, and K.-R. Müller, "A note on brain actuated spelling with the berlin brain-computer interface," in *Universal Access in Human-Computer Interaction. Ambient Interaction.* Springer, 2007, pp. 759–768.

[48] K. Hild, U. Orhan, D. Erdogmus, B. Roark, B. Oken, S. Purwar, H. Nezamfar, and M. Fried-Oken, "An ERP-based brain-computer interface for text entry using rapid serial visual presentation and language modeling," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations.* Association for Computational Linguistics, 2011, pp. 38–43.

[49] E. W. Sellers and E. Donchin, "A P300-based brain–computer interface: initial tests by als patients," *Clinical neurophysiology*, vol. 117, no. 3, pp. 538–548, 2006.

[50] T. Milekovic, T. Ball, A. Schulze-Bonhage, A. Aertsen, and C. Mehring, "Error-related electrocorticographic activity in humans during continuous movements," *Journal of neural engineering*, vol. 9, no. 2, p. 026007, 2012.

[51] M. Falkenstein, J. Hohnsbein, J. Hoormann, and L. Blanke, "Effects of crossmodal divided attention on late ERP components. ii. error processing in choice reaction tasks," *Electroencephalography and clinical neurophysiology*, vol. 78, no. 6, pp. 447–455, 1991.

[52] W. J. Gehring, B. Goss, M. G. Coles, D. E. Meyer, and E. Donchin, "A neural system for error detection and compensation," *Psychological science*, vol. 4, no. 6, pp. 385–390, 1993.

[53] M. Falkenstein, J. Hoormann, S. Christ, and J. Hohnsbein, "ERP components on reaction errors and their functional significance: a tutorial," *Biological psychology*, vol. 51, no. 2, pp. 87–107, 2000.

[54] G. Schalk, J. R. Wolpaw, D. J. McFarland, and G. Pfurtscheller, "EEG-based communication: presence of an error potential," *Clinical Neurophysiology*, vol. 111, no. 12, pp. 2138–2144, 2000.

[55] B. Blankertz, C. Schäfer, G. Dornhege, and G. Curio, "Single trial detection of eeg error potentials: A tool for increasing BCI transmission rates," in *Artificial Neural NetworksICANN 2002*. Springer, 2002, pp. 1137–1143.

[56] B. Blankertz, G. Dornhege, C. Schäfer, R. Krepki, J. Kohlmorgen, K.-R. Müller, V. Kunzmann, F. Losch, and G. Curio, "Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial EEG analysis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, no. 2, pp. 127–131, 2003.

[57] A. Buttfield, P. W. Ferrez, and J. R. Millan, "Towards a robust BCI: error potentials and online learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 164–168, 2006.

[58] R. Chavarriaga and J. d. R. Millán, "Learning from EEG error-related potentials in noninvasive brain-computer interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 4, pp. 381–388, 2010.

[59] A. Combaz, N. Chumerin, N. V. Manyakov, A. Robben, J. A. Suykens, and M. M. Van Hulle, "Towards the detection of error-related potentials and its integration in the context of a P300 speller brain–computer interface," *Neurocomputing*, vol. 80, pp. 73–82, 2012.

[60] T. Milekovic, T. Ball, A. Schulze-Bonhage, A. Aertsen, and C. Mehring, "Error-related electrocorticographic activity in humans during continuous movements," *Journal of neural engineering*, vol. 9, no. 2, p. 026007, 2012.

[61] E. Koyas, "Design and analysis of a brain-computer interface-based robotic rehabilitation system," Master thesis, Sabanci University, 2013.

[62] J. F. D. Saa, "Probabilistic graphical models for brain computer interfaces," Phd thesis, Sabanci University, 2014.

[63] W. Zhou, Y. Liu, Q. Yuan, and X. Li, "Epileptic seizure detection using lacunarity and bayesian linear discriminant analysis in intracranial EEG," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 12, pp. 3375–3381, 2013.

[64] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[65] T. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

[66] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[67] Y. Li, H. Kambara, Y. Koike, and M. Sugiyama, "Application of covariate shift adaptation techniques in brain–computer interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 6, pp. 1318–1324, 2010.

[68] C. Vidaurre, R. Cabeza, R. Scherer, G. Pfurtscheller *et al.*, "Study of on-line adaptive discriminant analysis for EEG-based brain computer interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 3, pp. 550–556, 2007.

[69] R. C. Panicker, S. Puthusserypady, and Y. Sun, "Adaptation in P300 brain–computer interfaces: A two-classifier cotraining approach," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 12, pp. 2927–2935, 2010.

[70] B. Dal Seno, M. Matteucci, and L. Mainardi, "Online detection of P300 and error potentials in a BCI speller," *Computational intelligence and neuroscience*, vol. 2010, p. 11, 2010.

[71] H. Verschore, "A brain-computer interface combined with a language model: the requirements and benefits of a P300 speller," Master thesis, Gent University, 2012.

[72] W. Speier, C. Arnold, J. Lu, R. K. Taira, and N. Pouratian, "Natural language processing with dynamic classification improves P300 speller accuracy and bit rate," *Journal of neural engineering*, vol. 9, no. 1, p. 016004, 2012.

[73] B. Graimann, B. Z. Allison, and G. Pfurtscheller, *Brain-computer interfaces: Revolutionizing human-computer interaction.* Springer Science & Business Media, 2010.

[74] B. Blankertz, K.-R. Müller, G. Curio, T. M. Vaughan, G. Schalk, J. R. Wolpaw, A. Schlögl, C. Neuper, G. Pfurtscheller, T. Hinterberger *et al.*, "The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1044–1051, 2004.

[75] B. Blankertz, K.-R. Müller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlögl, G. Pfurtscheller, J. R. Millan, M. Schröder, and N. Birbaumer, "The BCI competition iii: Validating alternative approaches to actual BCI problems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 153–159, 2006.

[76] G. Krausz, R. Scherer, G. Korisek, and G. Pfurtscheller, "Critical decision-speed and information transfer in the graz brain–computer interface," *Applied psychophysiology and biofeedback*, vol. 28, no. 3, pp. 233–240, 2003.

[77] C. Vidaurre, A. Schlogl, R. Cabeza, R. Scherer, and G. Pfurtscheller, "A fully on-line adaptive BCI," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 6, pp. 1214–1219, 2006.

[78] L. Sbattella and R. Tedesco, "A predictive speller for a brain-computer interface based on motor imagery."

[79] R. C. Panicker, "Adaptation and control state detection techniques for brain-computer interfaces," Ph.D. dissertation, 2011.

[80] P. Sykacek, S. J. Roberts, and M. Stokes, "Adaptive BCI based on variational bayesian kalman filtering: an empirical evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 5, pp. 719–727, 2004.

[81] M. Thulasidas, C. Guan, and J. Wu, "Robust classification of EEG signal for brain-computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 1, pp. 24–29, 2006.

[82] A. Amcalar and M. Cetin, "Design, implementation and evaluation of a real-time P300-based brain-computer interface system," in *International Conference on Pattern Recognition (ICPR), 2010 20th.* IEEE, 2010, pp. 117–120.

[83] R. Chavarriaga and J. d. R. Millán, "Learning from EEG error-related potentials in noninvasive brain-computer interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 4, pp. 381–388, 2010.

[84] T. Zeyl and T. Chau, "Strategies for adaptive motor imagery classification using error-related potential derived labels have unique risk profiles."

[85] T. J. Zeyl and T. Chau, "A case study of linear classifiers adapted using imperfect labels derived from human event-related potentials," *Pattern Recognition Letters*, vol. 37, pp. 54–62, 2014.

[86] M. Spüler, M. Bensch, S. Kleih, W. Rosenstiel, M. Bogdan, and A. Kübler, "Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-bci," *Clinical Neurophysiology*, vol. 123, no. 7, pp. 1328–1337, 2012.

[87] J. Mattout, M. Perrin, O. Bertrand, and E. Maby, "Improving BCI performance through co-adaptation: Applications to the P300-speller," *Annals of physical and rehabilitation medicine*, vol. 58, no. 1, pp. 23–28, 2015.

[88] P. Margaux, M. Emmanuel, D. Sébastien, B. Olivier, and M. Jérémie, "Objective and subjective evaluation of online error correction during P300-based spelling," *Advances in Human-Computer Interaction*, vol. 2012, p. 4, 2012.

[89] T. Zeyl, E. Yin, M. Keightley, and T. Chau, "Adding real-time bayesian ranks to error-relatedpotential scores improves error detection and auto-correction in a P300 speller," 2015.