RSVP Keyboard™ : An EEG Based BCI Typing System with Context Information Fusion

A Thesis Presented

by

Umut Orhan

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in the field of

Electrical and Computer Engineering

Northeastern University
Boston, Massachusetts

December 2013

Northeastern University

Abstract

Department of Electrical and Computer Engineering

Doctor of Philosophy in Electrical and Computer Engineering

by Umut Orhan

Worldwide, there are millions of people with severe motor and speech impairment, which prohibits them from participating in daily functional activities such as personal care. While many individuals may understand language fully and retain cognitive skills, they have no way to produce speech. Communication with other people, especially with their family members and care providers, becomes a significant challenge. Brain computer interface (BCI) is a technology that uses neural signals as the physiological input for various assistive technology applications. In BCIs, electroencephalography (EEG) has drawn increasing attention of the researchers due to noninvasiveness, portability, feasibility, and a relatively low cost.

In this work, we design an EEG based letter-by-letter BCI typing system, RSVP Keyboard$^{\text{TM}}$, which utilizes rapid serial visual presentation (RSVP). Differently from the commonly used visual BCI presentation schemes, RSVP aims to be accessible to the population with limited eye gaze control by presenting sequences of symbols on a screen over time at a fixed focal area and in rapid succession. As a response to the infrequent novel target stimulus (oddball paradigm), brain generates P300, an event related potential which is a positive deflection in the scalp voltage mainly in the centro-parietal areas with an average latency just over 300 ms. This natural novelty response allows us to design interfaces by detecting the intent using EEG.

The intent detection of the BCI systems suffers from low symbol rates and accuracy due to existence of low signal-to-noise ratios and the variability of background activity. Currently, most designers utilize high number of repetitions to increase the accuracy by sacrificing the speed. Alternatively, we propose a probabilistic incorporation of a language model in decision making and we show that our method significantly improves typing speed and accuracy. Additionally, we propose modeling of trial dependencies to improve the typing speed.

Acknowledgments

First and foremost, I would like to thank Dr. Deniz Erdogmus for an invaluable research experience. His firm commitment to scientific approach has been an inspiring guidance for me. He is the only person I have felt to be always on the same wavelength. I also would like to express my gratitude to my committee members, Dr. Jennifer Dy, Dr. Frank Guenther and Dr. Melanie Fried-Oken for their valuable comments and precious time. I would like to specially thank Dr. Murat Akcakaya, who not only served in my dissertation committee, but also acted as my secondary supervisor.

During my Ph.D. research, I collaborated with many people with various backgrounds. I am very grateful for their contributions. Dr. Brian Roark and Andrew Fowler shared their insight on natural language processing in addition to building the language model being used. The neurophysiological expertise of Dr. Barry Oken greatly improved our understanding of the brain. Most importantly, I greatly benefited from very different points of view of our clinical team in Oregon Health and Science University, Dr. Melanie Fried-Oken, Betts Peters, Aimee Mooney, Meghan Miller and Elena Goodrich. Thanks to them and Greg Bieker, my attitude towards people with disabilities completely changed. I greatly enjoyed our progress achieved by constant struggle to understand each other. I would also like to thank Dr. Kenneth Hild and Dr. Dana Brooks for their stimulating collaboration and discussion. Additionally, I am grateful to all members of Cognitive Systems Laboratory for creating a collaborative, joyful work environment and for their contributions. Specially, Dr. Erhan Bas, Seyhmus Guler, Shalini Purwar, Hooman Nezamfar, Sheng You, Asieh Ahani, Esra Ataer-Cansizoglu, Jamshid Sourati, Marzieh Haghighi, Sina (Mohammad Moghadamfalahi), Nastaran Ghadar, Abhimanyu Sheshashayee, Delia Fernandez, Matt Higger, Fernando Quivira, Amirali Haghayeghi, Dr. Olexiy Kyrgyzov, Dr. Jernej Zupanc, Dr. Burak Erem, Jaume Coll-Font.

My loneliness during this adventure diminished thanks to Senem Bas, Mert Korkali, Cem Bila, Ebru Korbek-Erdogmus, Cihan Tunc, Kivanc Kerse, Adnan Korkmaz, Bulent Bilir and One Piece. Glory Noethe and Faith Crisley, who significantly eased my journey, are also unforgotten.

I am indebted to thank my parents, Emel and Celal Orhan, who unconditionally and adamantly support me. They have never ceased their love and care despite being distant and in darkness.

Additionally, I would like to thank Mustafa Kemal Ataturk, the founder of the Republic of Turkey. His modernization efforts constitute the foundation for the scientific education and research originating from Turkey.

The research in this dissertation was supported by National Institutes of Health by grant NIH-5R01DC009834 and National Science Foundation by grants NSF-IIS0914808, NSF-CNS1136027, NSF-IIS1149570, and NSF-SMA0835976 (CELEST).

Contents

List of Figures

List of Tables

To the followers of scientific skepticism,

and the opposers of dogmatism...

Chapter 1

Introduction

1.1 Historical Overview

In 1970s, Vidal's first attempts to demonstrate the feasibility of utilizing brain signals for "man-computer-dialogue" started the era of brain computer interfaces (BCI) [1, 2]. It had taken more than 40 years for BCI to be born since the first recording of human electroencephalography (EEG), measurement of scalp voltage potentials, by Berger [3]. We, including the people with severe motor and speech impairments (SSPI), had to wait 20 more years for computational power, accessibility, cost and stability of computers and EEG amplifiers to attain a certain level for BCI field to draw exploding interest from researchers. Growth of interest increased further owing to the courtesy of researchers who built and shared packages of readily available EEG processing or BCI tools, e.g. BCI2000, EEGLAB, OpenViBE [4–7]. Probably due to heavy reliance on these available tools by new researchers entering the field, the improvement of BCI system performance (e.g. speed of typing) slowed down, contrary to the growing interest. For example, context awareness is ostensibly a required component of BCI systems. However very limited amount of effort has been put to utilization of context information.

1.2 Contributions of this Dissertation to the BCI Field

The first contribution of this dissertation is the design of a rapid serial visual presentation (RSVP)-based BCI typing system. Since making the presentation at a single focal location, RSVP does not require a strong gaze control, it is potentially a good candidate for people with the most severe motor impairments. Correspondingly, RSVP started to draw more attention from researchers especially following a publication demonstrating the gaze dependency of by far the most popular BCI presentation technique, the matrix speller [8]. The matrix speller utilizes a matrix layout of symbols with various intensifications, whereas RSVP presents the symbols at a central location with temporal separation.

Second, we constructed a theoretical framework for tight probabilistic context awareness in P300-based BCIs. In our case the context is language and the letters needed to spell out words. We incorporated a probabilistic fusion utilizing n-gram language models and estimation of probability densities of the features. Such a fusion would allow achieving high accuracies under low repetition conditions, correspondingly speeding up overall typing performance. Offline and online analyses, demonstrated significant improvements on typing performance.

Third, we designed a fusion and presentation methodology that does not commit to a decision, but instead keeps track of all the history and correspondingly updates a history of candidate symbols. In offline analysis, this recursive Bayesian fusion approach demonstrated significant improvement compared to the original fusion idea. However, it is not implemented for operation in real time due to its current limitations for cognitive load and memory requirement. After resolving such shortcomings, this approach has a high potential on increasing the typing speeds.

Finally, almost none of the P300 based BCIs consider the temporal dependency between trials. We created a model considering the temporal dependency between trials to potentially improve the typing speed.

Chapter 2

Brain Computer Interfaces

2.1 Augmentative and Alternative Communication

BCI is now considered as a possible access method for communication by individuals with severe speech and physical impairments (SSPI) who cannot meet their expressive language needs through natural speech, handwriting, or typing. BCIs interpret brain activity directly, bypassing physical movement and relying on neurophysiologic signals as an access method [9]. BCI for communication falls into a class of assistive technology (AT) and is placed with other augmentative and alternative communication (AAC) devices as an access means for language expression [10, 11]. Historically, AAC devices with different interfaces (i.e., mouse, joystick, binary switches, head control or eye gaze) have offered individuals means to generate and speak messages, when speech and writing are no longer functional [12]. A number of recent developments in AAC access strategies for people with minimal movement have been proposed that involve tracking of head and eye movement, recognition of residual speech and of gestures. BCI is one recent development that relies on monitoring the electrical activity of the brain [13]. Together, these strategies should provide even

greater access to face-to-face and electronic communication options to support engagement for health management and social interactions [14] for people with SSPI. As with any AT for communication, BCI translational research and development can be discussed in terms of five components [15]: (1) the input modalities for the device (for this work, we limit our discussion to electroencephalography (EEG)); (2) the processing demands of the device (here we refer to the signal detection and classification options); (3) language representation (for BCI, this refers to the graphical user interface (GUI) for language presentation and the manipulation of language units by the device); (4) the output modalities (for BCI, this is usually text output, though speech output is a possibility); and (5) the functional gains of the device (here we refer to the target populations and the clinical demands they bring to the task of BCI use). The long term objective of BCI translational research is to find a reliable means to enhance communication and control so that individuals with the most severe impairments have a means to participate in daily life for health, employment, social interaction, and community involvement. Critical to any discussion of BCI for communication is the concept of user-centered design. Based on the needs and preferences of the target population who will use this technology for verbal engagement, we must evaluate functionality, satisfaction, and expected outcomes of the users. We must consider the homes and environments where BCI will be implemented and the involvement required of the care providers and family members who will be operating the systems. The time for set up, the demands for technical assistance, and the ease of problem solving for this new technology must be considered with the users. These factors will ultimately be the true measures of success [16]. We must caution, however, that BCI is not a practical, dependable application for AT at this time. The sophisticated operations of the technology and the challenges of the target population are huge; obstacles to functional use have not yet been solved for independent implementation in users' homes. Expert end-users have told us that our challenge is to design a BCI that is safe, reliable, and that restores function at near normal levels [17]. Despite the benefits that AAC technologies offer

people with impairments, the potential of independent communication has not been fully realized for a group of individuals who present with such severe physical impairments that they cannot reliably or consistently control devices through available access methods. BCI is the hopeful, though not yet practical, solution for them.

2.2 Electroencephalography

EEG measures the potential differences between various points on the scalp. It is utilized to measure the neural activity corresponding to the electrical activity of neurons with similar spatial orientation. Since the electrical field generated by these neurons is extremely small, the activity below the cerebral cortex (outermost level) reflects very poorly to EEG. Moreover, scalp acts as a low pass filter, correspondingly blocking most high frequency content.

EEG components corresponding to a neural activity generally have an amplitude of a couple of μV. However, general EEG activity may have a significantly higher magnitude due to artifacts, overall background noise and environmental interference. Some of the disadvantages of EEG can be listed as low spatial resolution, low signal-to-noise ratio (SNR), setup time consumption. On the other hand, the advantages make it very attractive for BCI research, e.g. high temporal resolution, safety, cost-effectiveness and portability.

2.3 Input Modalities

EEG-based BCIs have become increasingly popular due to their portability, cost-effectiveness, high temporal resolution, and demonstrated reliability. In the following sections, we will categorize noninvasive BCI systems for expressive communication based on the stimulus presentation paradigms. A number of physiological signals

have been used in noninvasive BCI to detect user intent. Most popularly, BCI systems have exploited:

Auditory and visual event related potentials (A-ERP/V-ERP): As a response to infrequent novel/target stimuli, the brain generates a P300 response, a positive deflection in centro-parietal scalp voltage with a typical latency just over 300ms [18] and other accompanying waves. This natural novelty detection or target matching response of the brain allows designers to detect user intent from EEG signals, using either auditory or visual stimuli to elicit this response.

Volitional cortical potentials (VCP): Volitional synchronization and desynchronization of cortical electrical activity have been utilized in numerous BCI systems that control external devices, including, cursors, avatars, and robotic agents to perform simple activities of daily living, as well as to control typing interfaces for communication.

Steady-state evoked potentials (SSEP): Fluctuating auditory or flickering visual stimuli (following periodic or other structured patterns) will elicit steady state auditory/visual evoked potentials (SSAEP/SSVEP) in the auditory and visual cortex areas, respectively. Focusing auditory or visual attention on one of several such stimuli causes temporally matching electrical oscillations in the cortex. Time-frequency features can be analyzed to identify with high accuracy which stimulus the attention is placed on.

2.3.1 Event Related Potentials

In their pioneering work, Farwell and Donchin illustrate the feasibility of P300 as a control signal for BCI-based communication [19]. In this study, the subjects view a 6x6 matrix (matrix speller) consisting of letters in the English alphabet, numbers

from 1 to 9 and a space symbol. Since the publication of this work, extensive research has focused on various configurations or algorithms designed to improve the speed and the accuracy of communication with the matrix speller, as well as other audio, visual, and tactile stimulus presentation techniques for eliciting P300 responses. In the following subsections, we will first review these stimulus presentation techniques and then the signal processing and inference techniques used.

2.3.1.1 Visuospatial Presentation Techniques

Existing visuospatial presentation techniques can be categorized under the following heading.

Matrix Presentation: The Matrix Speller generally uses an RxC matrix of symbols with R rows and C columns (Fig. 2.1) depicts a 6x6 symbol matrix with the second column highlighted with the intention of inducing an ERP if the target letter is in this column). To generate an oddball paradigm, traditionally each row and column (and in modern versions each one of alternatively designed subsets of symbols) is intensified in a pseudorandom fashion, while the participants count the number of highlighted rows or columns (or, in general, subsets) that include the desired symbol. Usually a sequence is defined as the intensification of all the rows and columns in the matrix. The highlighting of the row and column containing the target symbol are rare events, and will induce a P300 response. The objective of the BCI system is to detect these deviations to identify the target letter to enable typing.

EEG signals suffer from low signal-to-noise ratio; therefore, to achieve a desired accuracy level, matrix speller systems require multiple presentation sequences before a decision can be made. For example, using bootstrapping and averaging the trials in different sequences, it was demonstrated that the matrix speller can achieve 7.8 characters/minute with 80% communication accuracy [20]. This speed and accuracy

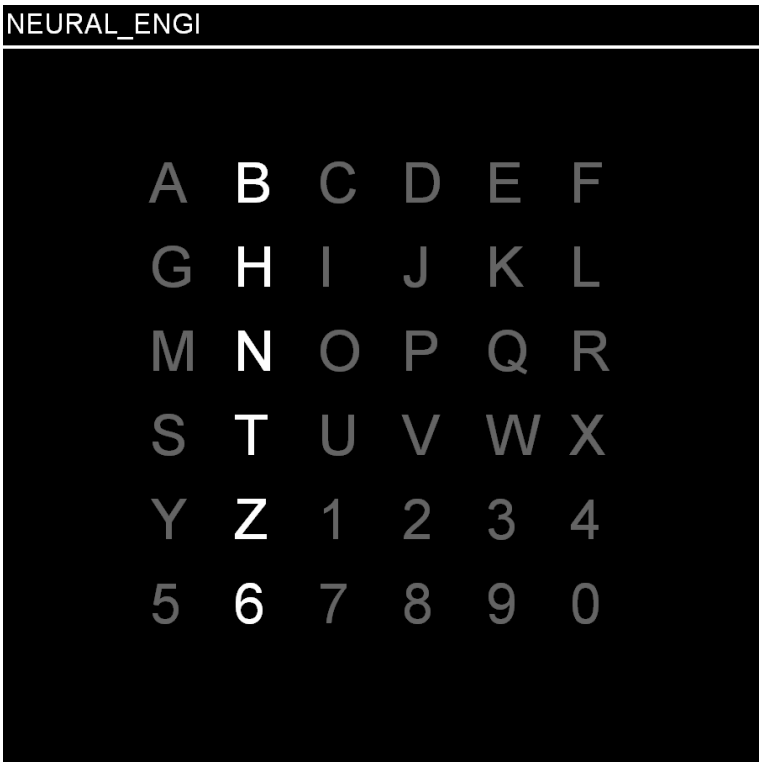


FIGURE 2.1: Sample presentation screen for matrix speller.

may not satisfactorily meet the needs of the target population. Therefore, various signal processing and machine learning techniques have been proposed to develop ERP-based matrix speller systems with higher speed and accuracy [21–43]. The matrix speller was shown to be highly accurate in overt attention mode, but in covert attention mode its performance degrades significantly [8, 44]. To overcome such performance drops, BCI researchers have proposed gaze-independent stimulus presentation techniques such as rapid serial visual presentation and balanced-tree visual presentation.

Rapid Serial Visual Presentation (RSVP): RSVP is a technique in which stimuli are presented one at a time at a fixed location on the screen (as depicted in 2.2), at a rapid rate and in pseudorandom order. When the target is presented (a rare event since there is one target symbol in the entire alphabet) and observed by the user, ERP containing the P300 wave is generated in EEG as a consequence

of the target matching process that takes place in the brain. Consequently, BCI systems can be designed to detect these responses for typing. By utilizing temporal separation of symbols in the alphabet instead of spatial separation as in the matrix speller, RSVP aims to be less dependent on gaze control [45–48]. RSVP-based BCIs

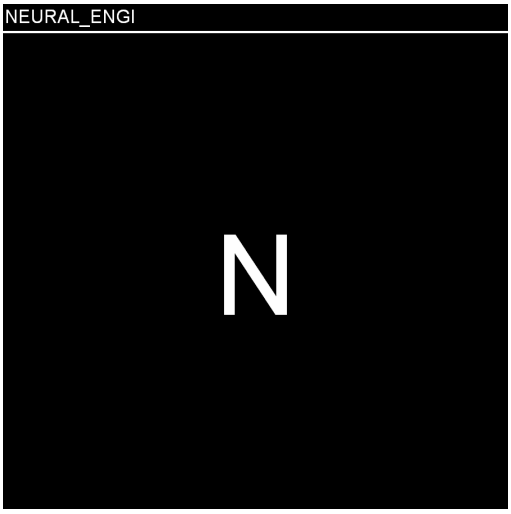


FIGURE 2.2: Sample presentation screen for RSVP Keyboard™ .

that use only EEG evidence may be slower than matrix spellers, as the binary tree that leads to symbol selections in a matrix speller could exploit the opportunity to highlight multiple symbols at a time to reduce expected bits to select a symbol (determined by entropy), while RSVP must follow a right-sided binary tree, which is highly structured and could lead to larger expected bits per symbol. RSVP-based typing has been demonstrated to achieve up to 5 characters/minute by Berlin BCI and RSVP Keyboard™ groups [45–48]. Color cues and language models have been used in an attempt to improve typing speeds with RSVP [46, 48]. On the positive side, RSVP is potentially feasible even for completely locked-in users, who may have difficulty with gaze control. RSVP BCIs, such as the RSVP Keyboard™ [48] and Center Speller [49] have similar signal processing and machine learning demands as matrix presentation based BCIs.

Balanced-Tree Visual Presentation Paradigms: Balanced-tree visual presentation refers to a technique in which visual stimuli are distributed into multiple presentation groups with equal numbers of elements. A variation would have been distributing elements into groups balanced in probability according to a Huffman tree based on a language model [50], but we have not encountered this approach in the BCI literature. In Berlin BCI's Hex-o-Spell, a set of symbols is distributed among multiple presentation groups; for example, 30 symbols may be distributed among 6 circles each containing 5 symbols, as shown in Fig. 2.3. Every presentation group is highlighted in a random fashion to induce an ERP for the selection of the group that contains the desired symbol. After the initial selection, the symbols in the selected presentation group are distributed individually to different presentation groups, typically with one empty group which represents a command to move back to the first presentation stage. At this point, the individual symbols are highlighted to elicit an ERP for selection of the desired symbol within the selected group [49, 51]. In Geospell, 12 groups of 6 symbols are arranged in a circular fashion similar to Hex-o-Spell presentation [52, 53] ; and in another study these 12 groups are presented to a user in RSVP manner in a random order to be employed in an ERP-based BCI speller [54]. In these systems, the 12 groups represent all the possible rows and columns of the 6x6 matrix speller such that the intersection of the selected row and column gives the desired symbol.

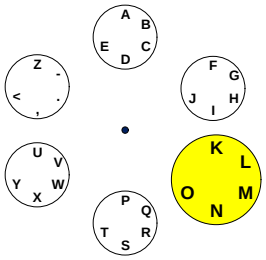


FIGURE 2.3: Sample presentation screen for Hex-o-Spell.

Other Visual Presentation Paradigms: The visual presentation paradigms explained above do not exhaustively cover all the possible presentation techniques

that could be (and have been) used in an ERP-based BCI system for communication. Various alternatives have been proposed and tested for limited communication. Here, we categorize systems that vary in their vocabulary extent from a few icons all the way down to binary (yes/no) communication as limited communication systems. Examples include:

- Icon-based limited communication - for example (i) systems for appliance or gadget control in which icons are flashing in sequences of random order one at a time [55, 56], and (ii) a system for expressing basic needs and emotions by answering yes/no questions [57]. RSVP iconMessenger is a variation of RSVP Keyboard^{TM} that uses limited-vocabulary icon representations (based on Rupal Patel's iconCHAT system).

- Cursor control - for example, a system in which four flashing stimuli map to movements of the cursor to one of four directions (up, down, left, right) [58–61]. Exogenous-icon (four arrows or four icons flashing on the sides of the screen) and endogenous-letter (letters representing directions) paradigms were tested on users with ALS, revealing that the endogenous paradigm provides better performance for a gaze-independent BCI [60]. Qualitatively, results were similar when the signal processing approach was improved [59].

- Web browser - for example, (i) the Virtual Keyboard (RoBIK) project, which employs a matrix-speller paradigm to provide the user with different tags which are mapped to elements of the web browser [62]; and (ii) a system that employs a matrix speller paradigm to allow complete keyboard and mouse control to navigate through web browser options [57].

2.3.1.2 Auditory Presentation Techniques

A-ERP signals have recently drawn attention for BCI design as an alternative or supplement to visual presentation methods due to their applicability in the population of users with impaired vision. Most A-ERP based BCIs employ a sequential stimulus arrangement. In these arrangements, there exists a single stream of stimuli, and users are expected to attend to the targets in the stream. Examples of stimulation methods include various combinations of tones for target and nontarget stimuli [63–65], utilization of cues with different pitch [66, 67], utilization of different sounds (bell, bass, ring, thud, chord, buzz) [68] and pronunciation of the stimuli [69]. These techniques induce ERPs when the target stimulus is perceived. Some groups also add directionality to the cues to improve discriminability or to utilize it as an additional stimulation method [66, 67]. In most A-ERP based BCIs, auditory presentation is utilized as a potential supplement for visual presentation and audio-visual presentations are done jointly. Accompanying the visual cue with an auditory one resulted in increased P300 amplitude and detection accuracy compared to only visual correspondence. Systems relying only on auditory stimulation performed significantly worse than visual BCIs. Although they are currently less accurate than visual BCIs, auditory BCIs are an important alternative for people who are unable to use visual BCIs.

2.3.1.3 Tactile Presentation Techniques

For users who cannot control their eye gaze or who have visual and/or hearing impairments, a tactile presentation technique could be used as an alternative to visuospatial and auditory presentation methods in BCI speller design [70]. One tactile speller interface assigns a set of symbols to each of six fingers, with six symbols in each set [71]. Symbols are selected in a two-stage process, as in the balanced tree presentation techniques described above. The user first selects a symbol set by focusing on a specific finger. The six letters in the selected set are then assigned to

the six fingers, and the user again focuses on a specific finger to select the desired symbol. A BCI system that employs this tactile presentation technique was shown to demonstrate a typing accuracy performance similar to matrix and Hex-o-Spell presentation techniques.

2.3.2 Volitional Cortical Potentials

Starting with motor imagery induced synchronization and desynchronization of cortical potentials, BCI designs quickly started exploiting the ability of the brain to learn new skills, including the volitional control of time-frequency characteristics of cortical potentials. Consequently, among all designs, BCIs based on these synchronization and desynchronization effects of volitional user brain activity can benefit most from user training. In fact, it has been observed that subjects may achieve some level of proficiency in highly variable durations, from a few hours of practice to tens of hours or more [72]. It has also been noted that individual characteristics may be influential factors in the ability to generate mu rhythms (see below) [73]. By training and reinforcement, users can improve their skills and accordingly system performance. The following VCP have been exploited to design BCI systems for communication:

- Slow cortical potentials (SCP) are gradual changes in EEG voltage. These fluctuations can last from hundreds of milliseconds to several seconds. Contingent negative variation (CNV), readiness potential, and movement-related potentials (MRP), are instances of SCRs, and some include P300 and N400 in this category as well.

- Mu rhythms (also known as comb, wicket, or sensorimotor rhythms), are 8-13 Hz synchronized patterns found primarily over the motor cortex in brain regions that control voluntary movements. The mu pattern is suppressed when a motor action is performed or even thought about. This phenomenon is an

example of event related desynchronization (ERD). Alpha rhythm, a signal with similar frequency range, but observed primarily over the visual areas of the brain while eyes are closed and the brain is at rest, is not to be confused with mu rhythm in BCI design.

- Beta rhythms, occurring in the frequency range 12-30 Hz, are typically considered in three subbands: low beta (12-16 Hz), beta (16-20 Hz), and high beta (20-30 Hz). These waves are suppressed over the motor cortex when there is a muscle contraction prior to and during movement. Beta energy is increased when movement has to be resisted or voluntarily suppressed.

VCP based BCIs typically require long user training sessions [74–78]. The Thought Translation Device (TTD) [86] is an example of this type of system. TTD utilizes SCP, which are known to be producible in every subject, unlike EEG rhythmic components. Although improvements in classification algorithms [74] and determination of mental strategies for more effective control of VCP [77, 78] have enhanced performance, long training sessions are still necessary. Some researchers, including the Berlin BCI group, have shifted the burden of adaptation more towards the machine learning algorithm to compensate for extensive user training requirements [79–82].

2.3.3 Steady State Evoked Potentials

SSEP-based interfaces include those that use auditory and visual stimulation intended to evoke responses by flickering lights or fluctuating auditory stimuli (such as click trains, tone pulses, or amplitude-modulated sounds). Several SSVEP-based typing interfaces have been developed, beginning with Sutter [83, 84], who uses phase shifted m-sequences to flicker each symbol on a matrix keyboard layout. Spuler et al [42] investigate a similar design using phase shifted 63-length m-sequences as stimuli to enable typing on a 32-symbol matrix keyboard. Hwang et al [85] have a 30-symbol matrix keyboard layout where each symbol has a dedicated flickering LED

with a unique frequency (between 5-10Hz, separated by frequency gaps on the order of 0.1Hz). Cheng et al [86] utilize a phone key layout for digits and introduce a few additional buttons, all flickering at different frequencies. Yin et al [87] use simultaneously flashing (to elicit ERPs) and different flickering frequencies for a matrix layout keyboard with 36 symbols. Cecotti [88] uses a hierarchical balanced tree approach and breaks the alphabet of 29 symbols into a 3-level tree with three branches at each (non-leaf) node. With this, they have 3 boxes that contain symbols and two additional stimuli that represent delete and repeat commands, leading to five flickering frequencies. On the other hand, Bremen BCI uses a 1-gram letter probability based keyboard layout. The user navigates a cursor on it by attending visually to one of four flickering arrows and selects the intended letter when ready using a fifth flickering stimulus in the corner [89–94]. In systems using SSAEP, which have been investigated only in recent years, dichotic fluctuating auditory stimuli are presented using speakers or earphones. Specifically, in the streaming stimulus arrangement, the stimuli are presented at the same time as multiple streams and distinguished by detecting the stream the user is attending to [95]. To improve the effectiveness of dichotic presentation, an amplitude modulation on the stream can be induced [74]. Hohne et al [67] combine streaming and sequential stimulus arrangements by considering sequential pitch-based cues applied to left, right or both ears and utilizing a combination of SSAEP and A-ERP evidence to determine user intent.

Chapter 3

EEG Signal Processing

Signal processing is a required component of any BCI system, including RSVP Keyboard$^{\text{TM}}$. Therefore, we will give a brief literature review of the existing EEG signal processing methodology and a detailed explanation of the procedures applied in RSVP Keyboard$^{\text{TM}}$ for the purpose of completeness. However we do not make any claim on the comparison of these techniques with other ones as the main focus of this dissertation is not EEG feature extraction. The main purpose of this dissertation is not to find the best feature extraction methodology, but to demonstrate the effectiveness of the overall design choices including tight probabilistic fusion with context information. The entire raw-EEG-to-feature pipeline could be improved, and this is the subject of future research and development. Most of the proposed context information fusion and temporal dependency modeling ideas can be utilized with other preprocessing and classification algorithms.

In the remainder of the dissertation, we use the following definitions.

- *Trial:* A single stimulus.
- *Sequence:* A list of trials that are shown consecutively. At the end of each sequence, a decision might be attempted to be made.

- *Epoch:* A list of sequences, which the subject only have a single intended symbol common to all of the sequences in the epoch. An epoch ends when a decision is made.

3.1 Literature Survey

The signal processing and inference techniques used for BCI-based communication systems can be used with little or no modification for other applications of BCI. However, the particular application also presents some customization opportunities to be exploited by designers of BCI-based communication systems.

3.1.1 Preprocessing and Dimensionality Reduction

EEG signals acquired as a response to presented stimuli are not only noisy, with very low signal-to-noise ratio, but also have nonstationarities due to various factors such as physiological or environmental artifacts, sensor failure, and subject fatigue. To design an effective inference method for BCI, it is essential that the most salient EEG signal features are extracted as evidence. Preprocessing and dimension reduction are steps aimed at such feature extraction. In ERP-based BCIs the P300, in VCP mu rhythms, and in SSVEP occipital rhythms are of primary interest and statistical preprocessing spatiotemporal filters with priors that favor these components can be designed. In all designs, the removal of DC drift (the baseline fluctuations due to frequencies $\ll$ 1Hz) and possibly artifact-related high frequency components in EEG are partially achieved with a properly designed bandpass filter. This initial bandpass filtering is a common step in all BCI systems. It is recommended that linear-phase FIR (finite impulse response) filters be used to prevent phase-response-induced distortions to waves and rhythms, as well as to make accounting for group delay easy for downstream operations in the signal processing and inference pipeline.

In particular, for visually evoked potentials the group delay of the bandpass filter must be considered when aligning (unfiltered) event markers to filtered EEG. This also means that for real-time operation the bandpass filter group delay should be kept as small as possible (considering the tradeoff between having a high quality magnitude response for desired and undesired frequencies and the delay introduced to the inference process and the close-loop control dynamics; the latter consideration is relevant in robotic agent control applications). After the initial bandpass filtering, time-windowed data from different EEG channels are usually concatenated to obtain the EEG feature vector. Based on the sampling frequency and the number of channels used, this vector could have a high dimensionality. Several methods are employed, before or after concatenation as suitable, for feature dimension reduction and further noise and artifact reduction: grand average over all trials [20, 25, 57, 96], downsampling [20, 21, 35, 36, 49, 51, 55, 71, 97], discrete/continuous wavelet transform [20, 22, 24], feature selection by stepwise linear discriminant analysis [30], decimation by moving average filtering [25, 26, 30, 31, 33, 38, 43], channel selection [23, 26, 30, 35, 36], artifact removal through independent component analysis (ICA) [39, 58–61, 69, 98], enhancing P300 response by adaptive spatial filtering including common spatial pattern (CSP) and xDAWN algorithm [23, 24, 37, 40, 85, 99], and dimensionality reduction through principal component analysis (PCA) [43, 47]. For SSVEP-based designs two main inference techniques emerge: if flickering stimuli are discriminated by frequency, then the sum of powers at the first two or three harmonics of candidate frequencies are obtained from a power spectrum estimate [85–88]; if the flickering stimuli are discriminated by pseudorandom code phase shifts (or with different codes), canonical correlation analysis (acting like a matched filter) is employed [42, 91].

3.1.2 Classification

The purpose of the classifier in ERP-based systems is to detect the existence of ERP (especially P300) in the EEG response following each stimulus (e.g., intensification of rows/columns/subsets in the matrix speller, presentation of letters/symbols in the RSVP paradigm, or finger tapping events in a tactile stimulation paradigm). In SSVEP/SSAEP-based systems the classifier uses temporal or frequency domain features to detect which stimulus the user is attending to (e.g., flickering arrows or textures on the screen for SSVEP/codeVEP or tones/clips in SSAEP paradigms). In VCP, the classifier attempts to identify which imagery-induced brain rhythm is prominent in EEG, especially over motor cortical areas for motor imagery paradigms, using spatiotemporal filtering and feature extraction. The most commonly used classification approaches include (1) linear discriminant analysis (LDA) based classifiers (e.g. Fisher LDA (FLDA), Stepwise LDA (SWLDA), and Bayesian LDA) [20–23, 30, 31, 33, 37, 38, 41, 42, 49, 51, 52, 54, 55, 57, 64–66, 68, 78, 87, 97, 100], and (2) support vector machine (SVM) [24–27, 30, 32, 35, 36, 43, 58–60, 69]. Other , classifiers for BCI system include genetic algorithms [59], logistic linear regression [71, 74], neural networks [61, 101, 102], matched filters [39], Pearson's correlation method [30], and regularized discriminant analysis (RDA) and its special cases [45–48, 67, 103]. In addition, unsupervised and semisupervised methods including those that assume hierarchical Gaussian distribution models for EEG [28, 29], that are based on co-training of FLDA and BLDA [104], and that are based on offline learning of the ERP classifier from EEG using data from a pool of subjects followed by online adaptation for different individuals [34] have also been employed. Semisupervised classifier adaptation promises to reduce calibration data collection duration and possibly adaptability against nonstationarities in EEG during test phase. A BCI system's performance depends not only on the choice of classifier, but also on preprocessing methods, selected features, the users who participate in the study, and a multitude of other factors [105]. Therefore, a comparison among different studies to choose the "best" classifier for a BCI speller system is not feasible. However,

within individual studies, comparisons among classifiers have been attempted. For example, using offline EEG data, it was demonstrated that SWLDA and FLDA provided better overall classification performance compared to Pearson's correlation method, linear SVM, and Gaussian Kernel SVM [30], a matched filter based classifier outperformed a maximum likelihood based classifier [39], and BLDA outperformed LDA, SWLDA and neural networks [106].

3.2 Processing in RSVP Keyboard™

3.2.1 Preprocessing

EEG evidence needs to be extracted from the aforementioned P300-based novelty detection or target recognition ERP in order to allow the detection of user intent from EEG signals. This process starts with extracting stimulus-time-locked bandpass (BP) filtered EEG signals for each stimulus in the sequence. Since, physiologically, the most relevant non-motor signal components are expected to occur within the first 500ms following the stimuli, the [0,500)ms portion of the EEG following each stimulus is extracted. If the subject engages in intentional or unintentional motor activity following target stimuli, this also allows omitting most of the related motor cortex activity from the target-response feature extraction. For locked-in users with motor capabilities, this restriction could be relaxed to allow the system to benefit from any such motor responses for successful operation (at the cost of not truly being a brain interface anymore).

To extract well separated features, we use the following methodology: (1) Time-windowed EEG signals in each channel are filtered by [1.5,42]Hz bandpass filter (FIR, linear phase, length 153, 0 DC-gain) to remove low frequency signal drifts and noise at higher frequencies for each channel. Lower high-cutoff frequencies may be used. (2) For each channel individually, signals corresponding to the [0,500)ms

post-stimulus interval (after taking into account the group delay of the BP filter) are vectorized and projected to a slightly lower dimensional space by linear dimension reduction. In the existing system, this is achieved by principal component analysis (PCA) to remove directions with negligible variance (as determined by calibration EEG data). The principal directions with variance lower than 10^{-5} times that of the first principal direction are removed. (3) Projected vectors corresponding to each channel are concatenated to create a single aggregated feature vector. This process amounts to a channel-specific energy preserving orthogonal projection of raw temporal features through eigenfiltering and downsampling. (4) Using RDA [107], a quadratic projection of this aggregate feature vector to $\mathbb{R}$ is achieved. This reduces the dimensionality of EEG evidence for each trial to one in the current implementation. Without a doubt, one can employ one of the numerous other feature extraction methodologies, which might perform better or worse.

3.2.2 Regularized Discriminant Analysis

RDA is a modified quadratic discriminant analysis (QDA) model. If the feature vector corresponding to each class is assumed to have multivariate normal distribution, the optimal Bayes classifier resides in the QDA family. This is due to the fact that the logarithm of the ratio of two multivariate normal density functions is a quadratic function. Consequently, under the Gaussianity assumption, QDA-projection in step (4) above yields the best one dimensional feature in the minimum expected risk sense. It only depends on the class means, covariance matrices, class priors, and a relative-risk based threshold/weighting (threshold if only EEG evidence is used to classify). All of these are to be estimated from the training data, in this case, from a calibration session, except for the risk-dependent threshold. The calibration session is supervised data collection to learn the signal statistics. When there exist a small number of samples for high dimensional problems, singularities in the estimation of

these covariance matrices become problematic. This is generally the case for ERP-classification, since the duration of the calibration session is limited. Henceforth, the maximum likelihood estimate of the covariance matrices will not be invertible, which is needed for the corresponding QDA solution.

RDA is proposed as a remedy to this obstacle. It modifies the covariance estimates to eliminate the singularities by applying shrinkage and regularization on them. The shrinkage procedure makes the class covariance matrices closer to the overall data covariance, and therefore to each other. Thus, it makes the discriminant closer to being a linear function of the feature vectors instead of a quadratic one. Let $\mathbf{y}_v \in \mathbb{R}^p$ be the set of feature vectors used to learn the discrimination function of RDA from, where p is the number of features to administer RDA on and $v \in \{1, 2, \cdots, N\}$ is the index of the samples. Correspondingly, the maximum likelihood estimates of the means and covariances are given by

$$\boldsymbol{\mu}_k = \frac{1}{N} \sum_{c(v)=k} \mathbf{y}_v,$$

and

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\mathbf{S}_k}{N_k} = \frac{1}{N_k} \sum_{c(v)=k} (\mathbf{y}_v - \boldsymbol{\mu}_k)(\mathbf{y}_v - \boldsymbol{\mu}_k)^T,$$

where $c(v) \in \{0, 1\}$ is the class label of sample v, $k \in \{0, 1\}$ represents the class label and N_k is the number of samples belonging to class k, i.e. $|\{v : c(v) = k\}|$. The shrinkage procedure is administered as, $\forall k \in \{0, 1\}$

$$\hat{\boldsymbol{\Sigma}}_k(\lambda) = \frac{\mathbf{S}_k(\lambda)}{N_k(\lambda)}, \tag{3.1}$$

where

$$\mathbf{S}_k(\lambda) = (1 - \lambda)\mathbf{S}_k + \lambda\mathbf{S},$$

and

$$N_k(\lambda) = (1 - \lambda)N_k + \lambda N,$$

with

$$N = \sum_{k\in\{0,1\}} N_k, \quad \mathbf{S} = \sum_{k\in\{0,1\}} \mathbf{S}_k.$$

The shrinkage parameter, $\lambda \in [0, 1]$, determines how much the individual covariance matrices are to be shrinked towards the pooled estimate. Further regularization is applied as,

$$\hat{\mathbf{\Sigma}}_k(\lambda, \gamma) = (1 - \gamma)\hat{\mathbf{\Sigma}}_k(\lambda) + \frac{\gamma}{p}\text{tr}[\hat{\mathbf{\Sigma}}_k(\lambda)]\mathbf{I},$$

where $\hat{\mathbf{\Sigma}}_k(\lambda)$ is given by (3.1), $\text{tr}[\cdot]$ is the trace operator and $\mathbf{I}$ is the $p \times p$ identity matrix. For a given λ, the regularization parameter, $\gamma \in [0, 1]$ controls the shrinkage towards the circular covariance.

RDA provides a broad family of regularization options. The four special cases of λ and γ represent various well-known classifiers:

- $\lambda = 0,\ \gamma = 0$: quadratic discriminant analysis
- $\lambda = 1,\ \gamma = 0$: linear discriminant analysis
- $\lambda = 0,\ \gamma = 1$: weighted nearest-means classifier
- $\lambda = 1,\ \gamma = 1$: nearest-means classifier

For $\gamma = 0$, varying λ corresponds to the models between QDA and LDA.

To illustrate how much these operations are effective on decreasing the singularities, we can investigate the ranks of the covariance matrices before and after. Before shrinkage $\text{rank}[\hat{\mathbf{\Sigma}}_k] \leq N_k$, after shrinkage,

$$\text{rank}[\hat{\mathbf{\Sigma}}_k(\lambda)] \leq \begin{cases} N_k, & \text{if } \lambda = 0 \\ N, & \text{otherwise} \end{cases}.$$

With the further application of regularization ranks of the covariance estimates become,

$$\text{rank}[\hat{\mathbf{\Sigma}}_k(\lambda,\gamma)] \begin{cases} \leq N_k, & \text{if } \lambda = 0, \gamma = 0 \\ \leq N, & \text{if } \lambda > 0, \gamma = 0 \\ = p, & \text{otherwise} \end{cases} \cdot$$

Since $N_k \ll p$ and $N < p$ most of the cases, shrinkage and regularization steps are both expected to be helpful to reduce the singularities.

After carrying out classifier shrinkage and regularization on the estimated covariance matrices, the Bayes classifier [108] is defined by the comparison of log-posterior-ratio with a risk-dependent threshold. The corresponding discriminant score function is given by,

$$\delta_{\text{RDA}}(\mathbf{y}) = \log \frac{f_{\mathcal{N}}(\mathbf{y}; \hat{\boldsymbol{\mu}}_1, \hat{\mathbf{\Sigma}}_1(\lambda,\gamma))\hat{\pi}_1}{f_{\mathcal{N}}(\mathbf{y}; \hat{\boldsymbol{\mu}}_0, \hat{\mathbf{\Sigma}}_0(\lambda,\gamma))\hat{\pi}_0},$$

where $\boldsymbol{\mu}_k$, $\hat{\pi}_k$ are estimates of class means and priors respectively; $f_{\mathcal{N}}(\mathbf{y}; \boldsymbol{\mu}, \mathbf{\Sigma})$ is the pdf of a multivariate normal distribution and $\mathbf{y}$ is the feature vector to apply RDA on. In this work, we will consider the discriminant score function as a nonlinear projection from $\mathbf{y}$ to x, i.e. $\mathsf{x} = \delta_{\text{RDA}}(\mathbf{y})$. Subsequently, $x = \delta_{\text{RDA}}(\mathbf{y})$ will be the one dimensional EEG feature for the fusion with language models as explained in Chapter 6.

3.2.3 Kernel Density Estimation

KDE is a nonparametric method to estimate the probability density function (pdf) of a random variable. It aims to obtain a smooth pdf using an i.i.d. sample set. KDE is applied as following,

$$\hat{f}(\mathsf{x} = x) = \frac{1}{Nh} \sum_{n=1}^{N} K_h(x - x_i), \tag{3.2}$$

where $\{x_1, x_2, \cdots, x_N\}$ are the samples to be used in the estimation, h is the bandwidth parameter and $K_h(.)$ is the kernel function with bandwidth h. Selection of a very small h would result with an undersmooth and jittery estimate of the pdf, on the other hand a large h would result with an oversmooth one. For Gaussian kernel, the bandwidth h is estimated using Silverman's rule of thumb [109] for each class. This assumes the underlying density has the same curvature as its variance-matching normal distribution. For estimating the class conditional probabilities, trivially it can be used as

$$\hat{f}(\mathsf{x} = x|\mathsf{c} = k) = \frac{1}{N_k} \sum_{c(v)=k} K_{h_k}(x - x(v)), \tag{3.3}$$

where $x(v)$ is the discriminant score corresponding to a sample v in the training data, that is to be calculated during cross validation, and $K_{h_k}(.)$ is the kernel function with bandwidth h_k.

Chapter 4

Performance Evaluation Methods

4.1 Information Transfer Rate

Information transfer rate (ITR) is currently the most popular metric in the BCI literature for assessing the performance [110]. The formula that has been heavily utilized in BCI field has the following form,

$$B = \log_2 |\mathcal{S}| + p \log_2 p + (1-p) \log_2[(1-p)/(|\mathcal{S}|-1)], \tag{4.1}$$

where B represents ITR (bits/symbol), p is the probability of accurately deciding the intended symbol, and $|\mathcal{S}|$ is the size of the symbol alphabet [9, 111, 112]. In bits/min, it is usually employed as

$$B_t = \frac{60B}{T_e}, \tag{4.2}$$

where T_e is the duration of one epoch, i.e. duration which user tries to select a symbol. Equivalently $T_e = N_s T_s$, where N_s is the number of sequences and T_s is the duration of a sequence. For the calculation of ITR under adaptive number of sequences, we will consider N_s as the average number of sequences per epoch.

ITR has been inspired by Shannon's channel coding theorem [113, 114].

Channel Coding Theorem: For a discrete memoryless channel, all rates below capacity C are achievable.

ITR measure considers the system between intent generation and decision making as the channel of concern and tries to estimate the capacity, i.e. the upper bound on the communication rate. Correspondingly, it does not represent the capacity that is achievable using BCIs, instead it estimates an upper bound of the rate for a certain decoder. Consecutively, by designing a different decoder, capacity might be changed. For estimation purposes, the following assumptions are made,

- Channel is discrete and memoryless.
- Each symbol of the alphabet are equilikely to be the intended symbol.
- The channel is m-ary symmetric and the transition matrix of the channel has the following form

$$p(y|x) = \begin{cases} p, & x = y \\ (1-p)/(|\mathcal{S}|-1), & x \neq y. \end{cases}$$

Unfortunately, these assumptions become significantly questionable for some BCIs. If the context information has been utilized in a BCI, the assumption of the channel being memoryless fails. Additionally, in a typing process in a context, the intended symbols are not equilikely. Moreover, the transition probabilities of an unintended symbols usually are not equal; due to not only potential similarities of the letters, but also the existence of a bias based on the context. Since these concerns are applicable to RSVP Keyboard$^{\text{TM}}$, ITR with such assumptions is not a fit measure for measuring performance. It might potentially overestimate the performance of RSVP Keyboard$^{\text{TM}}$. However we still calculate ITR for the sake of compliance to the field.

Increasing popularity and usage of information theory also cause many people to misinterpret the concept of channel capacity. Even though channel coding theorem states that any rate below the channel capacity is achievable, it merely proves the existence of a channel code that can decrease the error as much as we want. Without applying proper channel coding, not all rates below capacity might be achievable. For some BCIs it might not even be practically possible to apply channel coding without relaxing the assumptions, e.g. channel being memoryless. For example, in a BCI based on volitional cortical potentials expecting a user to imagine a code of movements (e.g. left hand-right foot-right foot-right hand) just to send a single message might not be feasible for the user.

4.2 Expected Symbol Duration

The typing duration can be directly related to the correct decision accuracy. If the correct decision making probability, p, is constant and greater than 0.5, the expected symbol selection duration becomes

$$\frac{T_s N_s}{2p-1},$$

where T_s is the duration of a sequence, and N_s is the number of sequences. If $p < 0.5$, with a nonzero probability the system will not be able to type the intended symbol, since it will not be able to correct its mistakes. Therefore, $p < 0.5$ is considered to be failure, as users might get stuck at a symbol. A similar relationship between expected time and probability of detection is also independently obtained in [115]. An alternative derivation for the formula is given in the following.

Catalan numbers: Catalan numbers arise in various counting problems [116]. One of the problems that results with Catalan numbers is the nested parentheses problem. The number of orderings of k left parentheses and k right parentheses

so that it becomes a properly nested ordering, correspond to k^{th} Catalan number. Explicitly it is, $C_k = \frac{1}{k+1}\binom{2k}{k}$. If we replace left parentheses with 0 and right parentheses with 1, the problem can equivalently be restated as the number of ways to order k 0s and k 1s so that no initial segment contains more 1s than 0s.

Lemma 1: Let $\mathsf{B}_n \in \{0, 1\}$ be i.i.d. Bernoulli random variables with success probability of p, i.e. $P(\mathsf{B}_n = 1) = p$. Let $b_1 b_2 \cdots b_l$ be a successful serie if following two conditions are satisfied,

(i) $\forall r \in \{1, 2, \cdots, l-1\}$ $b_1 b_2 \cdots b_r$, contains at least as many 0s as 1s,

(ii) $b_1 b_2 \cdots b_l$ contains more 1s than 0s.

The probability of achieving a successful serie is

$$P(\text{success}) = \begin{cases} 1, & \text{if } p \geq 0.5 \\ \frac{1-2p}{1-p}, & \text{if } p < 0.5 \end{cases}.$$

Proof. Using binomial series expansion

$$(1+y)^v = \sum_{k=0}^{\infty} \binom{v}{k} y^k, \tag{4.3}$$

where $\binom{v}{k}$ is the Binomial coefficient at $y = -4z$ and $v = -\frac{1}{2}$,

$$\begin{aligned}
(1-4z)^{1/2} &= \sum_{k=0}^{\infty} \binom{1/2}{k} (-4z)^k = \sum_{k=0}^{\infty} \frac{\prod_{r=0}^{k-1}(\frac{1}{2}-r)}{k!}(-4)^k z^k \\
&= 1 + \sum_{k=1}^{\infty} -\frac{\prod_{r=1}^{k-1}(2r-1)}{k!} 2^k z^k \\
&= 1 + \sum_{k=1}^{\infty} -\frac{(2k-2)!}{2^{k-1}(k-1)!k!} 2^k z^k \\
&= 1 - 2\sum_{k=1}^{\infty} \frac{1}{k} \binom{2(k-1)}{k-1} z^k \\
&= 1 - 2z \sum_{k=0}^{\infty} \frac{1}{k+1} \binom{2k}{k} z^k.
\end{aligned}$$

This converges if $|4z| \leq 1$. Rearranging the equation,

$$\sum_{k=0}^{\infty} \frac{1}{k+1} \binom{2k}{k} z^k = \frac{1-(1-4z)^{1/2}}{2z}. \tag{4.4}$$

Let s_m be the number of 1s in $b_1 b_2 \cdots b_m$. $\forall m \in \mathbb{N}$, $s_{m+1} \leq s_m + 1$, with the equality condition is satisfied only if $b_{m+1} = 1$. If $b_1 b_2 \cdots b_l$ is a successful serie, $s_l \geq \lceil (l+1)/2 \rceil$ from condition (ii), and $s_{l-1} \leq \lfloor (l-1)/2 \rfloor$ from condition (i), where $\lfloor . \rfloor$ and $\lceil . \rceil$ represent the floor and ceil operators, respectively. Combining these three inequalities, we obtain

$$\lceil (l+1)/2 \rceil \leq s_l \leq s_{l-1} + 1 \leq \lfloor (l-1)/2 \rfloor + 1 = \lfloor (l+1)/2 \rfloor \leq \lceil (l+1)/2 \rceil.$$

Consequently, $\leq s_l = s_{l-1} + 1$, and $\lfloor (l+1)/2 \rfloor = \lceil (l+1)/2 \rceil$. Hence $b_l = 1$ and l is odd, i.e. $l = 2k+1$. Therefore $P(\text{Success using } 2k \text{ trials}) = 0$. Furthermore $s_l = \lceil (2k+1+1)/2 \rceil = k+1$. As a conclusion, a successful serie of length $l = 2k+1$ contains k zeros and $k+1$ ones. Since each element of the serie is obtained via independent Bernoulli trials, achieving a given successful serie of length $2k+1$ has the probability of $p^{k+1}(1-p)^k$. The probability of achieving success with $2k+1$

trials is

$$P(\text{Success using } 2k+1 \text{ trials}) = p^{k+1}(1-p)^k C_k, \tag{4.5}$$

where C_k is the number of different successful series with length $2k+1$. In all successful series, $b_{2k+1} = 1$, therefore C_k is the number of different ways k 0s and k 1s where $\forall r \in \{1, 2, \cdots, 2k\}$, $b_1 b_2 \cdots b_r$ has no more 1s than 0s. Hence C_k is k^{th} Catalan number, $C_k = \frac{1}{k+1}\binom{2k}{k}$. Correspondingly the probability of success becomes,

$$\begin{aligned} P(\text{Success}) &= \sum_{l=1}^{\infty} P(\text{Success using } l \text{ trials}) \\ &= \sum_{k=0}^{\infty} P(\text{Success using } 2k+1 \text{ trials}) \\ &= \sum_{k=0}^{\infty} p^{k+1}(1-p)^k \frac{1}{k+1}\binom{2k}{k} \end{aligned}$$

Using (4.4) for $z = p(1-p)$,

$$\begin{aligned} P(\text{Success}) &= p\frac{1-(1-4p(1-p))^{1/2}}{2p(1-p)} \\ &= \frac{1-(1-4p+4p^2))^{1/2}}{2(1-p)} \\ &= \frac{1-|2p-1|}{2(1-p)} \\ &= \begin{cases} 1, & \text{if } p \geq 0.5 \\ \frac{p}{1-p}, & \text{otherwise} \end{cases}. \end{aligned}$$

□

Corollary. If $p < 0.5$, with $\frac{1-2p}{1-p}$ probability there will never be a success.

Lemma 2: If $p \geq 0.5$, the expected length of the series with success is $1/(2p-1)$, i.e.,

$$\mathbb{E}[\mathsf{L}] = \frac{1}{2p-1},$$

where L is a rv representing the length of the successful series.

Proof. Similar to Lemma 1, using Binomial series for (4.3) for $y = -4z$ and $v = -1/2$,

$$\begin{aligned}(1-4z)^{-1/2} &= \sum_{k=0}^{\infty} \binom{-1/2}{k} (-4z)^k = \sum_{k=0}^{\infty} \frac{\prod_{r=0}^{k-1}(-\frac{1}{2}-r)}{k!}(-4)^k z^k \\ &= \sum_{k=0}^{\infty} \frac{\prod_{r=0}^{k-1}(2r+1)}{k!} 2^k z^k \\ &= \sum_{k=0}^{\infty} \binom{2k}{k} z^k\end{aligned}$$

For $z = p(1-p)$,

$$\sum_{k=0}^{\infty} \binom{2k}{k} p^k (1-p)^k = (1-4p(1-p))^{-1/2} = \frac{1}{2p-1} \tag{4.6}$$

Since from Lemma 1, all of the probability mass is contained by successful series for $p > 0.5$, using (4.5) we obtain,

$$\begin{aligned}
\mathbb{E}[\mathsf{L}] &= \sum_{l=0}^{\infty} lP(\text{Success with } l \text{ trials}) \\
&= \sum_{k=0}^{\infty} (2k+1)P(\text{Success with } 2k+1 \text{ trials}) \\
&= \sum_{k=0}^{\infty} (2k+1)p^{k+1}(1-p)^k C_k \\
&= \sum_{k=0}^{\infty} (2k+1)p^{k+1}(1-p)^k \frac{1}{k+1}\binom{2k}{k} \\
&= \sum_{k=0}^{\infty} p^{k+1}(1-p)^k \frac{1}{2}\binom{2(k+1)}{k+1} \\
&= \frac{1}{2(1-p)}\left(-1 + \sum_{k=0}^{\infty} p^k(1-p)^k \binom{2k}{k}\right)
\end{aligned}$$

Using 4.6,

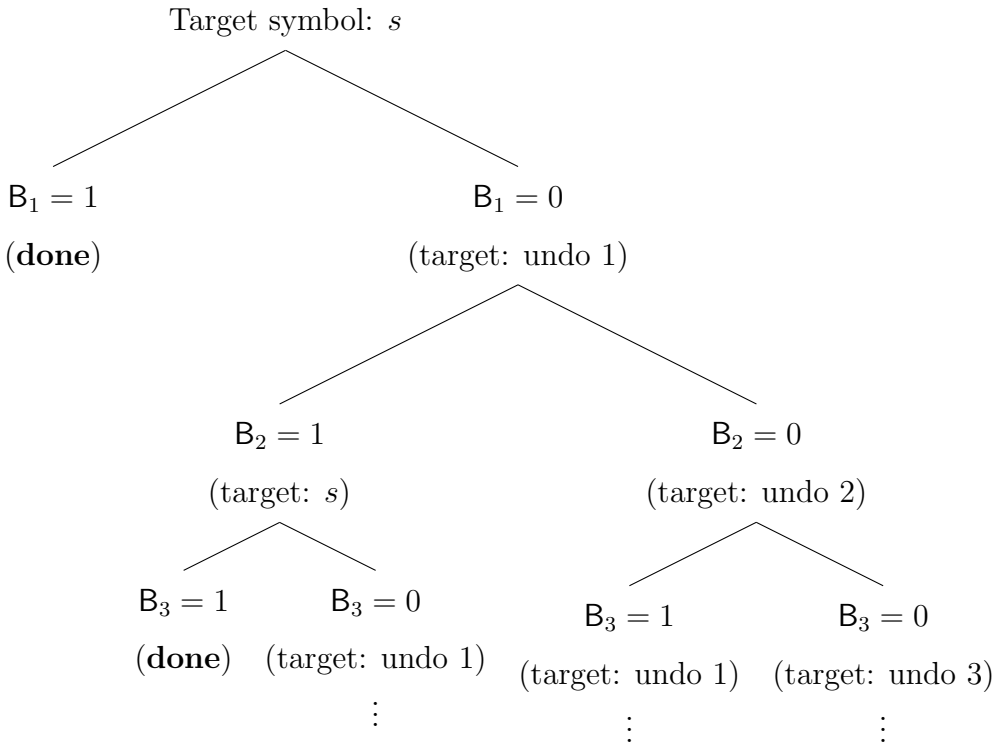
$$\mathbb{E}[\mathsf{L}] = \frac{1}{2(1-p)}\left(-1 + \frac{1}{2p-1}\right) = \frac{1}{2(1-p)} \cdot \frac{2(1-p)}{2p-1} = \frac{1}{2p-1}$$

□

4.2.1 BCI perspective

Assume that the correctness of a decision at the end of each epoch be i.i.d. Bernoulli random variables with success probability of p and the duration of the epoch, the duration a decision is made, to be T.

To be able to type a symbol accurately, we have the following tree.



Whenever the number of correct selections is more than the number of incorrect selections, the target symbol is typed correctly. The correct typing of a symbol is equivalent to having a successful serie as defined in Lemma 1. As an example, the successful conditions using 7 epochs are,

0101011

0100111

0011011

0010111

0001111.

Let the duration of an epoch is T. If we assume that the success of the decisions are independent and identically distributed, than the problem becomes equivalent to Lemma 2. Correspondingly we can calculate the expected symbol typing duration. If $p > 0.5$, expected duration of typing a symbol becomes $\frac{T}{2p-1}$ by Lemma 2. If

$p < 0.5$, the system would not be able to operate, since the there is a nonnegative probability of failure.

Chapter 5

Language Modeling

Language modeling is very important for many text processing applications, such as speech recognition, machine translation, as well as for the kind of typing application being investigated here [117]. In the BCI field, there exist recent attempts to use word prediction to speed up the typing process [38]. Typically, these approaches do not directly influence the decision making for an individual epoch, but instead give options for word completion. On the other hand, if the language models are incorporated into the decision making process, it might be possible to increase the speed and accuracy of the selection and our previous off-line analyses support this expectation. To utilize this idea, the prefix string (what has already been typed) is used to predict the next symbol(s) to be typed, using an appropriate language model. Consequently, the next letters to be typed become highly predictable in certain contexts, particularly letters internal to a word (after the first letter). In applications where text generation/typing speed is very slow, the impact of language modeling can become much more significant. BCI-spellers, including the RSVP Keyboard paradigm presented here, can be extremely low-speed letter-by-letter writing systems, and thus can greatly benefit from the incorporation of probabilistic letter predictions from an accurate language model during the writing process.

Recently other researchers in BCI literature have also utilized language models, by incorporating word prediction [38, 118, 119] or Bayesian fusion methods with n-gram models [41, 120, 121], similarly to this dissertation.

5.1 n-gram Models

The language model used in this dissertation is based on the n-gram sequence modeling paradigm, widely used in all of the application areas mentioned above. An L-gram model estimates the conditional probability of every letter in the alphabet given $L-1$ preceding letters using a Markov model of order $L-1$. In this context, let $\mathsf{s}_t : \Omega \to \mathcal{S}$ be the random variable corresponding to the correct symbol for epoch t, i.e. during t^{th} symbol selection, where $\mathcal{S}$ is the set of possible symbols. Since there might be an operation of deletion, the total number epochs might be larger than the number of characters written. Hence, we can define the number of characters written as a function of epoch index; in other words let i_t be the number of characters already typed until epoch t. With this notation, we have $i_t < t$. Additionally, the corresponding random sequence of the last $L-1$ characters written prior to epoch t are represented as $\mathsf{w}_j : \Omega \to \mathcal{A}$ where $j \in \{i_t - L + 2, \cdots, i_t - 1, i_t\}$ and $\mathcal{A} \subset \mathcal{S}$ is the alphabet (the set containing all symbols that can be selected, including backspace, letters, punctuation, digits, etc., in a general typing system). For representation simplicity, let $\mathbf{w}_t = [\mathsf{w}_{i_t}, \mathsf{w}_{i_t-1}, \cdots, \mathsf{w}_{i_t-L+2}]$ correspond to the random string of last $L-1$ characters during the selection of the target of t^{th} epoch and $\mathbf{w} = [w_1, w_2, \cdots, w_{L-1}]$ corresponds to a character string of length $L-1$. In L-gram models, the symbol prediction is made using the latest string of length L as

$$P(\mathsf{s}_t = s|\text{context}) \approx P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w}) = \frac{P(\mathsf{s}_t = s, \mathbf{w}_t = \mathbf{w})}{P(\mathbf{w}_t = \mathbf{w})}, \tag{5.1}$$

from Bayes' Theorem. In this equation, the joint probability mass functions are estimated using a large text corpus.

If the language model order is 1, the prediction probability is equal to the context-free letter occurrence probabilities in the English language, which is not dependent on the previous letters, i.e. $P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w}) = P(\mathsf{s}_t = s)$. Zero-gram model is defined as, having no active language model or equivalently $P(\mathsf{s}_t = s)$ is assumed to be a uniform distribution over the alphabet, i.e. $P(\mathsf{s}_t = s) = 1/|\mathcal{S}|$.

For the online system, a 6-gram language model was estimated from a one million sentence (210M character) sample of the NY Times portion of the English Gigaword corpus. Corpus normalization and smoothing methods were as described in [117]. Most importantly for this work, the corpus was case normalized, and we used Witten-Bell smoothing for regularization [122].

5.2 Backspace Probability

The ability to erase a typed symbol is an integral component of any typing system. However since the probability of the user intending to erase a symbol significantly depends on the error rate of the system, it is not accessible from the corpus language model is trained on. As the system makes more errors, user intends to select backspace considerably. To select the backspace probability we utilize the following ideas in RSVP Keyboard$^{\text{TM}}$.

- *Constant backspace probability:* The probability of the backspace according to the context is assumed to be constant.

$$P(\mathsf{s}_t = \leftarrow |\text{context}) = \text{constant},$$

  where $\leftarrow$ corresponds to the backspace symbol. Currently, the backspace probability has been set to 0.05. The constant backspace probability may be further optimized to increase the overall typing accuracy and decrease the typing speed using the simulation method explained in Section 7.3.

- *Confidence based backspace probability:* The probability of the backspace for the context is determined according to the confidence of the previous decision. If the estimated probability of correctness for the selected symbol in the previous epoch is p, correspondingly the probability for the decision being erroneous becomes $1-p$. Correspondingly, before we collect any additional EEG evidence for the current decision, the context probability of backspace being the intended symbol may be set as the estimated probability of the previous decision being erroneous. In other words,

$$P(\mathsf{s}_t = \leftarrow |\text{context}) = P(\mathsf{s}_{\tau_{i_t}} \neq w_{i_t} | \text{context}, \text{EEG}),$$

where w_{i_t} corresponds to the last typed symbol at the t^{th} epoch and τ_{i_t} represents the epoch w_{i_t} is selected. The method for making a joint decision and estimating $P(\mathsf{s}_{\tau_{i_t}} \neq w_{i_t} | \text{context}, \text{EEG})$ with the utilization of both of the evidences from the context and EEG is explained in detail in Chapter 6.

Chapter 6

Joint Decision Making with the Context Information

6.1 Single Trial Decision Making

The evidence obtained from EEG and the language model can be used collaboratively to make a more informed decision about the class that each symbol belongs to. An epoch, i.e. multiple repetitions of each sequence, is going to be shown for each symbol to be selected. Each symbol is assumed to belong to the class of either positive or negative attentional focus or intent. Let $\mathsf{c} : \Omega \rightarrow \{0, 1\}$ be the random variable representing the class of intent, where 0 and 1 corresponds to negative and positive intents, respectively and $\mathbf{x} : \Omega \rightarrow \mathbb{R}^d$ be a random vector of EEG features corresponding to a trial. For example, an ERP discriminant function that projects the EEG data corresponding to a trial into a single dimension may be used as a feature extraction method. In this case, since $d = 1$, which represents there is only one EEG feature per trial. The fusion methodology explained here does not depend on the feature extraction method, and practically can be used with any feature vector

in $\mathbb{R}^d$. The only requirement is an estimate of the conditional probability density function of EEG features given the class label, i.e. $f(\mathbf{x} = \mathbf{x}|\mathsf{c} = c)\ \forall c \in \{0, 1\}$.

Specifically, let $\mathbf{x}_{t,s,r}$ be the random EEG feature vector corresponding to a trial for epoch $t \in \mathbb{N}$, symbol $s \in \mathcal{S}$ and repetition $r \in \{1, 2, \cdots, R\}$, R is the total number of repetitions or sequences of the symbols per epoch. Furthermore, let $\mathsf{c}_{t,s}$ be the random variable representing the class of epoch t and symbol s. Consequently, for a symbol s, the posterior probability of the class being c using the $L-1$ previous symbols and EEG features for all of the repetitions of symbol s in epoch t can be written as,

$$\begin{aligned} Q = P(\mathsf{c}_{t,s} = c|\mathbf{x}_{t,s,1} = \mathbf{x}_1, \mathbf{x}_{t,s,2} = \mathbf{x}_2, \cdots, \mathbf{x}_{t,s,R} = \mathbf{x}_R, \\ \mathsf{w}_{i_t} = w_1, \mathsf{w}_{i_t-1} = w_2, \cdots, \mathsf{w}_{i_t-L+2} = w_{L-1}), \end{aligned} \tag{6.1}$$

where $\mathbf{x}_r \in \mathbb{R}^d$ for $r \in \{1, 2, \cdots, R\}$. Using Bayes' Theorem on (6.1), we obtain

$$Q = \frac{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \cdots, \mathbf{x}_{t,s,R} = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w}|\mathsf{c}_{t,s} = c)P(\mathsf{c}_{t,s} = c)}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \cdots, \mathbf{x}_R = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}. \tag{6.2}$$

We can assume that the EEG features corresponding to the symbol in question and the text that has been written already are conditionally independent given the class label of the symbol. This assumption is reasonable, because after the subject decides on a target symbol by considering the previously typed text, he/she is expected to show positive intent for the target and negative intent for the others, and after the class of a symbol is decided the EEG response is expected not to get affected by the text already written. This assumption can formally be written as $\mathbf{x}_{t,s,1}, \cdots, \mathbf{x}_{t,s,R} \perp\!\!\!\perp \mathbf{w}_t|\mathsf{c}_{t,s}$. Accordingly, (6.2) transforms to,

$$Q = \frac{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \cdots, \mathbf{x}_{t,s,R} = \mathbf{x}_R|\mathsf{c}_{t,s} = c)P(\mathbf{w}_t = \mathbf{w}|\mathsf{c}_{t,s} = c)P(\mathsf{c}_{t,s} = c)}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \cdots, \mathbf{x}_R = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}. \tag{6.3}$$

It can be further assumed that the EEG responses for each repetition of the symbol

are conditionally independent given the class of the symbol. This assumption expects intents to be independent and identically distributed for a symbol in an epoch. As an example, if the subject shows a stronger intent for the second repetition, then the assumption fails. Since estimating such a joint conditional probability density function would be difficult as the number of repetitions gets higher, this assumptions constitutes a useful simplifying approximation. More formally, this can be written as $\mathbf{x}_{t,s,1} \perp\!\!\!\perp \mathbf{x}_{t,s,2} \perp\!\!\!\perp \cdots \perp\!\!\!\perp \mathbf{x}_{t,s,R} | \mathsf{c}_{t,s}$, reducing (6.3) to,

$$Q = \frac{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r | \mathsf{c}_{t,s} = c)\right) P(\mathbf{w}_t = \mathbf{w} | \mathsf{c}_{t,s} = c) P(\mathsf{c}_{t,s} = c)}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \cdots, \mathbf{x}_{t,s,R} = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}.$$

Using Bayes' Theorem once again on $P(\mathbf{w}_t = \mathbf{w}_t | \mathsf{c}_{t,s} = c)$, we obtain,

$$Q = \frac{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r | \mathsf{c}_{t,s} = c)\right) P(\mathsf{c}_{t,s} = c | \mathbf{w}_t = \mathbf{w}) P(\mathbf{w}_t = \mathbf{w})}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \cdots, \mathbf{x}_{t,s,R} = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}. \tag{6.4}$$

We can apply the likelihood ratio test for $\mathsf{c}_{t,s}$ to make a decision between two classes. The likelihood ratio of $\mathsf{c}_{t,s}$, can be written from (6.1) as,

$$\Lambda(\mathsf{c}_{t,s} | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) = \frac{P(\mathsf{c}_{t,s} = 1 | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})}{P(\mathsf{c}_{t,s} = 0 | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})},$$

where $\mathbf{X}_{t,s} = \{\mathbf{x}_{t,s,1}, \mathbf{x}_{t,s,2}, \cdots, \mathbf{x}_{t,s,R}\}$ and $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_R\}$. Using the form we obtained after simplifications and approximations from (6.1) to (6.4), likelihood ratio can be rewritten as

$$\Lambda(\mathsf{c}_{t,s} | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) = \frac{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r | \mathsf{c}_{t,s} = 1)\right) P(\mathsf{c}_{t,s} = 1 | \mathbf{w}_t = \mathbf{w})}{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r | \mathsf{c}_{t,s} = 0)\right) P(\mathsf{c}_{t,s} = 0 | \mathbf{w}_t = \mathbf{w})}.$$

In terms of the probabilities obtained from the language model, we can directly plug in these equivalent probabilities, $P(\mathsf{c}_{t,s} = 1 | \mathbf{w}_t = \mathbf{w}) = P(\mathsf{s}_t = s | \mathbf{w}_t = \mathbf{w})$ and

$P(\mathsf{c}_{t,s} = 0|\mathbf{w}_t = \mathbf{w})) = 1 - P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w})$. Finally, ratio of class posterior probabilities can be estimated as,

$$\Lambda(\mathsf{c}_{t,s}|\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) = \frac{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r|\mathsf{c}_{t,s} = 1)\right) P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w})}{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r|\mathsf{c}_{t,s} = 0)\right) (1 - P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w}))}. \tag{6.5}$$

In this equation, $f(\mathbf{x}_{t,s,r} = \mathbf{x}_r|\mathsf{c}_{t,s} = c)$ is to be estimated using the feature extraction algorithm and $P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w})$ is to be estimated using the language model. Therefore the decision on the class label of symbol s_t for epoch t, $\hat{c}_{t,s}$ may be done comparing the likelihood ratio with a risk dependent threshold, τ, i.e.,

$$\Lambda(\mathsf{c}_{t,s}|\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) \gtrless_{\hat{c}_{t,s}=0}^{\hat{c}_{t,s}=1} \tau, \tag{6.6}$$

or in other words,

$$\hat{c}_{t,s} = \begin{cases} 1, & \text{if } \Lambda(\mathsf{c}_{t,s}|\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) > \tau \\ 0, & \text{otherwise} \end{cases}.$$

6.1.1 Experimental Offline Analysis

Two healthy subjects, one man and one woman, were recruited for this study. Each subject participated in the experiments for two sessions. In each session 200 letters are selected (with replacement, out of 26) according to their frequencies in the English language and randomly ordered to be used as target letters in each epoch. In each epoch, the designated target letter and a fixation sign are each shown for 1s, followed by 3 sequences of randomly ordered 26 letters in the English alphabet with 150 ms inter-stimuli interval. Correspondingly each sequence takes 4.9 seconds including the 1 second fixation duration. Subjects are asked to look for the target letter shown at the beginning of the epoch.

The signals recording is done using a g.USBamp biosignal amplifier using active g.Butterfly electrodes from G.tec (Graz, Austria) at 256Hz sampling rate. The EEG electrodes are applied with g.GAMMAcap (electrode cap) and the positioned according to the International 10/20 System were O1, O2, F3, F4, FZ, FC1, FC2, CZ, P1, P2, C1 C2, CP3, CP4. Signals were filtered by a nonlinear-phase 0.5-60 Hz bandpass filter and a 60 Hz notch filter (G.tec's built-in design). Afterwards signals filtered further by the previously mentioned 1.5-42 Hz linear-phase bandpass filter (our design). The filtered signals were downsampled to 128Hz. For each channel, stimulus-onset-locked time windows of [0,500)ms following each image onset was taken as the stimulus response.

Let us denote by e_j the jth epoch in a given session and let $\mathbb{E}$ be the ordered set containing all epochs in the session. $\mathbb{E}$ is partitioned into 10 equal-sized nonintersecting blocks, $\mathbb{E}_k$; for every e_j there is exactly one k_j such that $e_j \in \mathbb{E}_{k_j}$. For every e_j acting as a test sample, the ERP classifier is trained on the set $\mathbb{E}\backslash\mathbb{E}_{k_j}$. During training, the classifier parameters λ and γ are determined using this 10-fold cross-validation approach and grid search within the set $\mathbb{E}\backslash\mathbb{E}_{k_j}$. The kernel density estimates of the conditional probabilities of classification scores for EEG classifiers are obtained using scores obtained from $\mathbb{E}\backslash\mathbb{E}_{k_j}$. The trained classifiers are applied to their respective test epochs to get the 10-fold cross-validation test results presented in the tables.

In signal detection theory, receiver operating characteristic (ROC) curve has usually been used to assess the performance [123]. ROC is a parametric plot of false alarm probability and detection probability for changing the test threshold. Since it is preferable to select an operating point with high detection rate and low false alarm rate, the shape of the ROC is an important representative of the performance of the detector. Consequently, the area under the curve (AUC) of an ROC is usually utilized as a summary statistic representing the ROC. AUC ($\in [0, 1]$) becomes 1 for the perfect detector (no false alarm, no miss) and becomes 0.5 for no discrimination. Even though it has been frequently utilized as a tool, it is important to note that

AUC does not capture all of the information contained in the ROC curve and does not represent the detectors actual operation.

An example of the change in AUC during the grid search is given in Fig. 6.1 for single sequence with no language model (0-gram). This figure demonstrates that the regularization and shrinkage are both necessary and significantly effective. However if the regularization is applied too much it might degrade the performance.

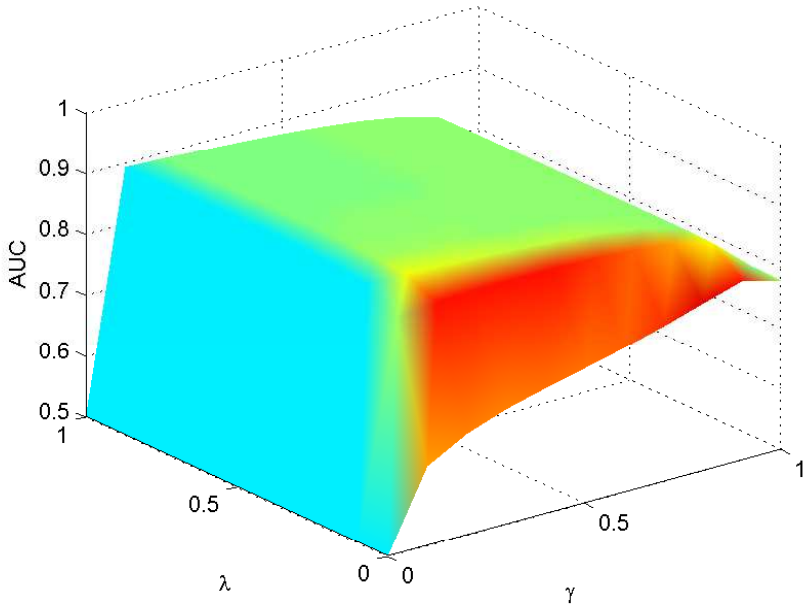


FIGURE 6.1: An example of the change in AUC while searching shrinkage (λ) and regularization (γ). Highest AUC is obtained for $\lambda = 0.6$ and $\gamma = 0.1$.

The language model was trained as described in Chapter 5. For each letter in the alphabet, 1000 random samples were drawn from the same corpus (separate from the language model training data) for testing purposes. For each letter sample we simulate the fusion of EEG responses and the language model in the following way: (i) each sample is assumed to be the target letter of a typing process using BCI; (ii) the predecessor letters of the target letter for each epoch are taken from the corpus to calculate the letter probabilities of the n-gram language models for each letter in the alphabet (Since subjects only focus on a single target letter without knowing the predecessor letters of the typing process in this experiment, it is assumed that the

EEG responses created during an epoch are independent from the predecessors.); (iii) under the assumption of independence of EEG responses with the previous letters selected, for each epoch, the EEG responses for every letter is converted to EEG classifier scores; (iv) matching model probabilities for each letter are obtained from the language model; (v) and the fusion of ERP classifier scores and language model predictions was achieved as described above, resulting in a joint discriminant score that needs to be compared with a threshold depending on risk ratios for missing a target letter and a false selection.

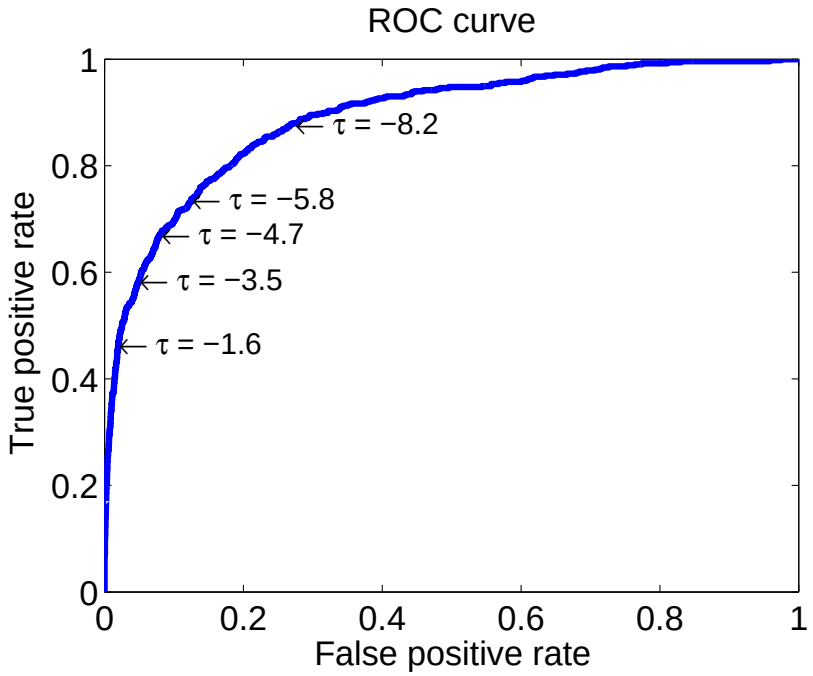


FIGURE 6.2: An example of an ROC curve corresponding to single sequence and 0-gram language model.

Fusion results were obtained for n-gram model orders $L = 0$, 1, 4, and 8. The EEG scores were assumed to have been evaluated for $R = 1$, 2, and 3 sequences (to evaluate the contribution of multi-trial information) to decide if a letter under evaluation was a desired target letter or not. In the results, only EEG data from the first R sequences of each epoch were used to classify each selected sequence count. Receiver operating characteristics (ROC) curves were obtained using the decision rule given in (6.5) by changing the risk based threshold, τ. An example

ROC curve is given in Fig. 6.2. Area under the ROC curves were calculated for different orders of the language model, for different number of sequences used and for different positions of the sample target letter in the corresponding word from the corpus. In Table 6.1, the area under the ROC curves (AUC) are compared. Each entry contains the pair of minimum and maximum AUC over the sessions, i.e. each pair represent the performance of the worst and best session. In Table 6.2, Table 6.3, and Table 6.4 the correct detection rates are given for false positive rates of 1%, 5%, and 10%, respectively. These correspond to different values of the τ, since selecting a point on the ROC curve correspond to a false alarm rate, true positive rate and τ triplet.

		1 sequence	2 sequences	3 sequences
0-gram		(0.812, 0.884)	(0.907, 0.956)	(0.957, 0.985)
1-gram		(0.892, 0.922)	(0.944, 0.973)	(0.972, 0.986)
4-gram	If 1st letter	(0.892, 0.941)	(0.954, 0.983)	(0.977, 0.991)
	If not 1st letter	(0.975, 0.983)	(0.985, 0.992)	(0.991, 0.997)
8-gram	If 1st letter	(0.905, 0.945)	(0.960, 0.984)	(0.979, 0.992)
	If not 1st letter	(0.991, 0.993)	(0.995, 0.997)	(0.995, 0.998)

TABLE 6.1: The minimum and the maximum values of the area under the ROC curves obtained using fusion classifier under different scenarios. The comparison is made using different number of sequences for classification, different letter positions in the word and different language model orders.

		1 sequence	2 sequences	3 sequences
0-gram		(0.101, 0.348)	(0.500, 0.532)	(0.625, 0.698)
1-gram		(0.255, 0.371)	(0.468, 0.583)	(0.591, 0.698)
4-gram	If 1st letter	(0.263, 0.416)	(0.434, 0.774)	(0.621, 0.810)
	If not 1st letter	(0.597, 0.684)	(0.748, 0.849)	(0.848, 0.927)
8-gram	If 1st letter	(0.294, 0.448)	(0.465, 0.782)	(0.647, 0.835)
	If not 1st letter	(0.810, 0.854)	(0.886, 0.932)	(0.936, 0.972)

TABLE 6.2: The minimum and the maximum values of the detection rates for 1% false detection rate using fusion classifier under different scenarios.

The letter decisions after an epoch may be made by selecting the symbol with maximum likelihood ratio. This corresponds to selecting the symbol with maximum $\Lambda(\mathsf{c}_{t,s}|\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})$ from (6.5). If applied with our offline analysis structure,

		1 sequence	2 sequences	3 sequences
0-gram		(0.453, 0.548)	(0.700, 0.810)	(0.828, 0.889)
1-gram		(0.556, 0.660)	(0.767, 0.841)	(0.900, 0.953)
4-gram	If 1st letter	(0.606, 0.688)	(0.740, 0.884)	(0.886, 0.971)
	If not 1st letter	(0.842, 0.899)	(0.912, 0.966)	(0.960, 0.989)
8-gram	If 1st letter	(0.614, 0.716)	(0.766, 0.905)	(0.899, 0.971)
	If not 1st letter	(0.951, 0.971)	(0.972, 0.990)	(0.986, 0.996)

TABLE 6.3: The minimum and the maximum values of the detection rates for 5% false detection rate using fusion classifier under different scenarios.

		1 sequence	2 sequences	3 sequences
0-gram		(0.550, 0.661)	(0.800, 0.906)	(0.900, 0.969)
1-gram		(0.633, 0.797)	(0.817, 0.905)	(0.917, 0.984)
4-gram	If 1st letter	(0.692, 0.836)	(0.857, 0.961)	(0.948, 0.990)
	If not 1st letter	(0.933, 0.961)	(0.966, 0.991)	(0.983, 0.996)
8-gram	If 1st letter	(0.729, 0.840)	(0.873, 0.964)	(0.950, 0.990)
	If not 1st letter	(0.983, 0.990)	(0.992, 0.997)	(0.995, 0.998)

TABLE 6.4: The minimum and the maximum values of the detection rates for 10% false detection rate using fusion classifier under different scenarios.

the correct letter selection probabilities averaged over subjects are given by Fig. 6.3. The plots show correct letter selection probabilities vs inverse of number of repetitions. Since the number of repetitions is a direct measure of epoch duration, we used its inverse as a speed indicator. These curves indicate that usual speed/accuracy trade-offs apply to the proposed typing system and better language models result in better performance.

The typing duration can be directly related to the correct decision accuracy. If the correct decision making probability, p, is constant and greater than 0.5, the expected symbol selection duration becomes,

$$\frac{T_s N_s}{2p-1},$$

where T_s is the duration of the of a sequence, and N_s is the number of sequences. If $p < 0.5$, with a nonzero probability the system will not be able to type the

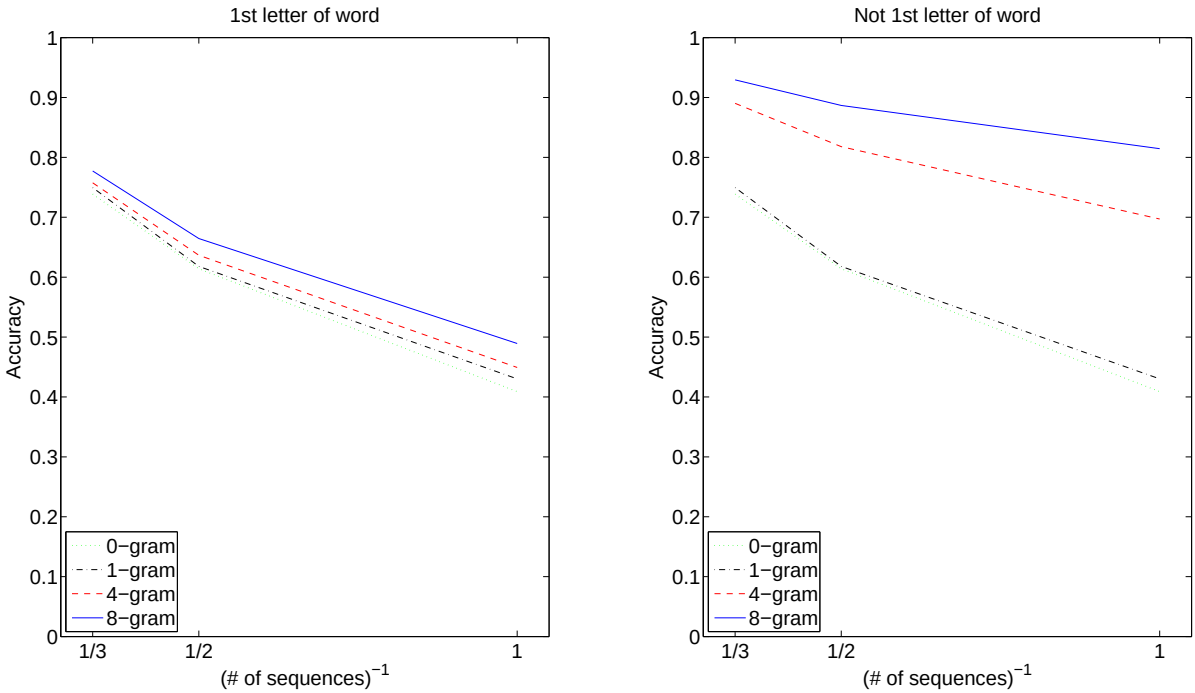


FIGURE 6.3: The average correct letter selection probability vs inverse of the number of repetitions for various language model orders and letter locations in the word.

intended symbol, since it will not be able to correct its mistakes. Therefore $p < 0.5$ is considered to be failure as subject might get stuck at a symbol. A similar relationship between expected time and probability of detection is also independently obtained independently in [115]. An alternative derivation for the formula is given in the appendix.

Typing duration per symbol, estimated using this relation, for various number of sequences, language model orders are given in Fig. 6.4 after averaging over sessions and subjects.

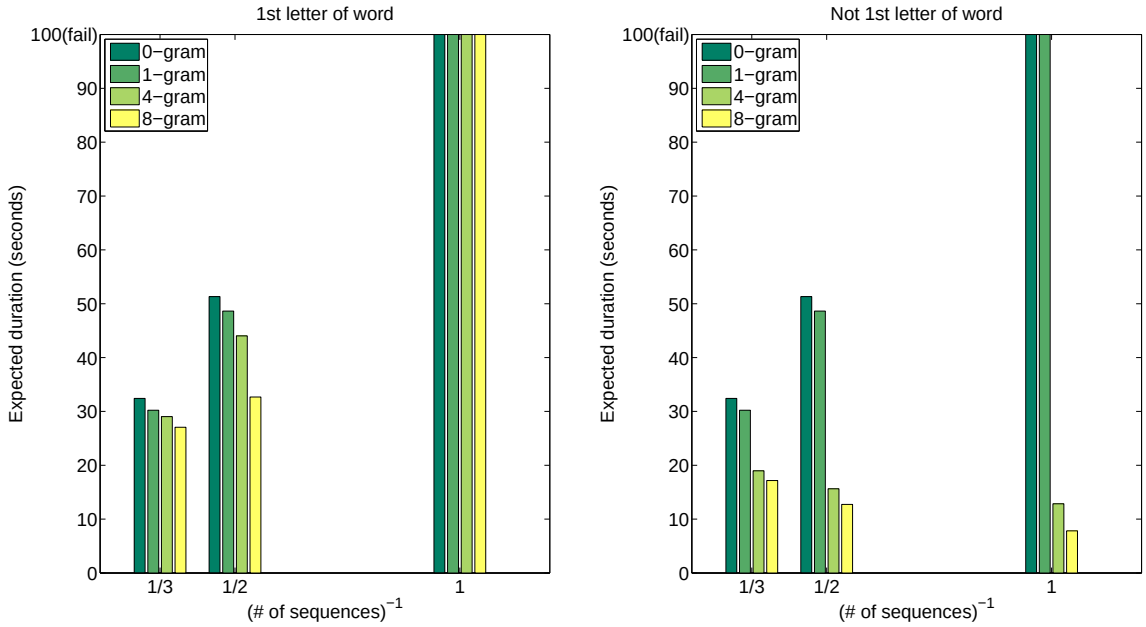


FIGURE 6.4: The average expected value of typing duration vs inverse of the number of repetitions for various language model orders and letter locations in the word. In the graphs 100 seconds mark is used jointly with the failure case. If the subject has a probability of getting stuck it is considered as a failure.

6.2 Naïve Bayesian Fusion

The evidence obtained from EEG and the language model is used collaboratively to make a more informative symbol decision. An epoch, combination of sequences that consist of trials corresponding to different symbols, is shown for each symbol to be selected. There exists two different context incorporation approaches currently available in RSVP Keyboard$^{\text{TM}}$. In both approaches, each symbol is assumed to belong to the class of either positive or negative intent and features for each trial are assumed to be i.i.d. for each class. The main difference between these rules is the assumptions they make.

6.2.1 Fusion Rule 1

Let $\mathsf{c} : \Omega \rightarrow \{0, 1\}$ be the random variable representing the class of intent, where 0 and 1 corresponds to negative and positive intents, respectively and $\mathbf{x} : \Omega \rightarrow \mathbb{R}^d$ be a random vector of EEG features corresponding to a trial. For example, an ERP discriminant function that projects the EEG data corresponding to a trial into a single dimension may be used as a feature extraction method. In this case, since $d = 1$, which represents there is only one EEG feature per trial. The fusion methodology explained here does not depend on the feature extraction method, and practically can be used with any feature vector in $\mathbb{R}^d$. The only requirement is an estimate of the conditional probability density function of EEG features given the class label, i.e. $f(\mathbf{x} = \mathbf{x}|\mathsf{c} = c)\ \forall c \in \{0, 1\}$.

Specifically, let $\mathbf{x}_{t,s,r}$ be the random EEG feature vector corresponding to a trial for epoch $t \in \mathbb{N}$, symbol $s \in \mathcal{S}$ and repetition $r \in \{1, 2, \cdots, R_s\}$, R_s is the total number of repetitions of symbol s in epoch t. Furthermore, let $\mathsf{c}_{t,s}$ be the random variable representing the class of epoch t and symbol s. Consequently, for a symbol s, using Bayes' theorem the posterior probability of the class being c using the $L-1$ previous symbols and EEG features for all of the repetitions of symbol s in epoch t can be written as,

$$P(\mathsf{c}_{t,s} = c|\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) = \frac{f(\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}|\mathsf{c}_{t,s} = c)P(\mathsf{c}_{t,s} = c)}{f(\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})}, \tag{6.7}$$

where $\mathbf{X}_{t,s} = \{\mathbf{x}_{t,s,1}, \mathbf{x}_{t,s,2}, \cdots, \mathbf{x}_{t,s,R_s}\}$ and $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{R_s}\}$ with $\mathbf{x}_r \in \mathbb{R}^d$ for $r \in \{1, 2, \cdots, R_s\}$.

We can assume that the EEG features corresponding to the symbol in question and the text that has been written already are conditionally independent given the class label of the symbol. This assumption is reasonable, because after the subject decides on a target symbol by considering the previously typed text, he/she is expected to show positive intent for the target and negative intent for the others, and after the

class of a symbol is decided the EEG response is expected not to get affected by the text already written. This assumption can formally be written as $\mathbf{X}_{t,s} \perp\!\!\!\perp \mathbf{w}_t | \mathsf{c}_{t,s}$. Accordingly, (6.7) transforms to,

$$P(\mathsf{c}_{t,s} = c | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) = \frac{f(\mathbf{X}_{t,s} = \mathbf{X} | \mathsf{c}_{t,s} = c) P(\mathbf{w}_t = \mathbf{w} | \mathsf{c}_{t,s} = c) P(\mathsf{c}_{t,s} = c)}{f(\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})}. \tag{6.8}$$

It can be further assumed that the EEG responses for each repetition of the symbol are conditionally independent given the class of the symbol. This assumption expects intents to be independent and identically distributed for a symbol in an epoch. As an example, if the subject shows a stronger intent for the second repetition, then the assumption fails. Since estimating such a joint conditional probability density function would be difficult as the number of repetitions gets higher, this assumptions constitutes a useful simplifying approximation. More formally, this can be written as $\mathbf{x}_{t,s,1} \perp\!\!\!\perp \mathbf{x}_{t,s,2} \perp\!\!\!\perp \cdots \perp\!\!\!\perp \mathbf{x}_{t,s,R_s} | \mathsf{c}_{t,s}$, reducing (6.8) to,

$$P(\mathsf{c}_{t,s} = c | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) = \frac{P(\mathbf{w}_t = \mathbf{w} | \mathsf{c}_{t,s} = c) P(\mathsf{c}_{t,s} = c) \prod_{r=1}^{R_s} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r | \mathsf{c}_{t,s} = c)}{f(\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})}. \tag{6.9}$$

Assuming there exists exactly one target symbol among the symbol set in an epoch, which is reasonable since the user is expected to look for only one target symbol, the symbol posterior probability using all of the evidence in the epoch becomes,

$$P(\mathsf{s} = s | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) = P\left(\mathsf{c}_{t,s} = 1 | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}, \sum_{s' \in \mathcal{S}} c_{s'} = 1\right), \tag{6.10}$$

where $\boldsymbol{\mathcal{X}}_t = \{\mathbf{X}_{t,s} | s \in \mathcal{S}\}$ and $\mathcal{X} = \{\mathbf{X}_s | s \in \mathcal{S}\}$. Using the definition of conditional probability on (6.10), we obtain

$$P(\mathsf{s} = s | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) = \frac{P\left(\mathsf{c}_{t,s} = 1, \mathsf{c}_{t,s'} = 0 \ \ \forall s' \neq s \in \mathcal{S} | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}\right)}{\sum_{s'' \in \mathcal{S}} P\left(\mathsf{c}_{t,s''} = 1, \mathsf{c}_{t,s'} = 0 \ \ \forall s' \neq s'' \in \mathcal{S} | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}\right)}.$$

This can be further approximated by the assumption that class labels for each symbol are conditionally independent given all the evidence. Correspondingly we obtain,

$$P(\mathsf{s} = s | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) \approx \beta \frac{P(\mathsf{c}_{t,s} = 1 | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})}{P(\mathsf{c}_{t,s} = 0 | \mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})},$$

where

$$\beta = \frac{1}{\sum_{s' \in \mathcal{S}} P(\mathsf{c}_{t,s'} = 1 | \mathbf{X}_{t,s'} = \mathbf{X}, \mathbf{w}_t = \mathbf{w}) / P(\mathsf{c}_{t,s'} = 0 | \mathbf{X}_{t,s'} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})}.$$

6.2.2 Fusion Rule 2

In general form the posterior probability of the intended symbol given all the evidence can be written as,

$$P(\mathsf{s} = s | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) \propto P(\boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w} | \mathsf{s} = s) P(\mathsf{s} = s), \tag{6.11}$$

after applying Bayes' Theorem.

We can assume that given the intended symbol EEG evidence and previously typed text are conditionally independent and we obtain (6.12). This is reasonable as once the intended symbol is decided by the subject, the response of the subject is expected not to depend on the previously typed text.

$$P(\mathsf{s} = s | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) \propto f(\boldsymbol{\mathcal{X}}_t = \mathcal{X} | \mathsf{s} = s) P(\mathbf{w}_t = \mathbf{w} | \mathsf{s} = s) P(\mathsf{s} = s), \tag{6.12}$$

After applying Bayes' Theorem and assuming EEG evidences for each trial are conditionally independent given the intended symbol, we obtain,

$$P(\mathsf{s}=s|\boldsymbol{\mathcal{X}}_t=\mathcal{X},\mathbf{w}_t=\mathbf{w}) \propto \left(\prod_{s'\in\mathcal{S}}\prod_{r_{s'}=1}^{R_{s'}} f(\mathbf{x}_{t,s',r}=\mathbf{x}_{t,s',r}|\mathsf{s}=s)\right) P(\mathsf{s}=s|\mathbf{w}_t=\mathbf{w}). \tag{6.13}$$

This assumption corresponds to EEG responses to individual trials are independent to each other, if the intended symbol is known. In other words, the response distributions do not depend on the symbol itself, and subject does not respond differently over time.

After simplification, (6.13) reduces to the following final fusion rule

$$P(\mathsf{s}=s|\boldsymbol{\mathcal{X}}_t=\mathcal{X},\mathbf{w}_t=\mathbf{w}) \propto \left(\prod_{r_s=1}^{R_s} \frac{f(\mathbf{x}_{t,s,r}=\mathbf{x}_{t,s,r}|\mathsf{c}_{t,s}=1)}{f(\mathbf{x}_{t,s,r}=\mathbf{x}_{t,s,r}|\mathsf{c}_{t,s}=0)}\right) P(\mathsf{s}=s|\mathbf{w}_t=\mathbf{w}). \tag{6.14}$$

In the current version of RSVP Keyboard$^{\text{TM}}$, the second fusion rule is being used since it has cleaner assumptions.

6.2.3 Final Decision

After applying one of the fusion rules and the intended symbol is decided by selecting the symbol with maximum probability given the EEG and context evidences,

$$\hat{s} = \arg\max_s P(\mathsf{s}=s|\boldsymbol{\mathcal{X}}_t=\mathcal{X},\mathbf{w}_t=\mathbf{w}).$$

At the end of each sequence $P(\mathsf{s}=\hat{s}|\boldsymbol{\mathcal{X}}_t=\mathcal{X},\mathbf{w}_t=\mathbf{w})$ is used as a confidence measure. Using the maximum value in the probability mass function as the measure of certainty corresponds to utilizing Renyi's order-infinity entropy definition as the measure of uncertainty in the decision we make. One could use other definitions of

entropy to measure uncertainty/certainty in the current decision. Decision in the epoch is finalized if one of the following conditions satisfied,

- $\max_s P(\mathsf{s} = \hat{s} | \boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) > \gamma$, where γ is the confidence threshold that is currently set to 0.9.
- Number of sequences reaches the maximum number of sequences allowed per epoch, e.g. 4, 8, 16

6.2.4 Deciding on the Next Sequence

After each sequence the posterior probabilities for the symbols are calculated. If the number of trials per sequence is smaller than $|\mathcal{S}|$, the symbols to be shown in the following sequence is decided either randomly or selecting the most likely symbols according to the current posterior probabilities. In the mode that selects most likely symbols, it might potentially prevent context unlikely symbols not to be shown. However they are expected to show up as subject demonstrates negative intent for the rest. Indeed, the subjects were even able to select very context-unlikely symbols like Q.

6.3 Recursive Bayesian Estimation Based Fusion

The naïve approach has the disadvantage of committing to one selection of a string of symbols, which might potentially cause the language model to be more harmful than beneficial after an erroneous selection. Therefore, as a remedy to this problem, we propose a typing strategy that does not immediately commit to a single selection. Correspondingly we construct a hidden Markov model (HMM) where the substring of last n_{LM} letters of the current epoch constitute the latent variable and EEG scores of all symbols corresponding to an epoch constitute the observed variable. An

illustration of the corresponding HMM is given in Fig. 6.5, where $\boldsymbol{\delta}_{\text{RDA},i}$ represents $\boldsymbol{\delta}_{\text{RDA}}$ for epoch i.

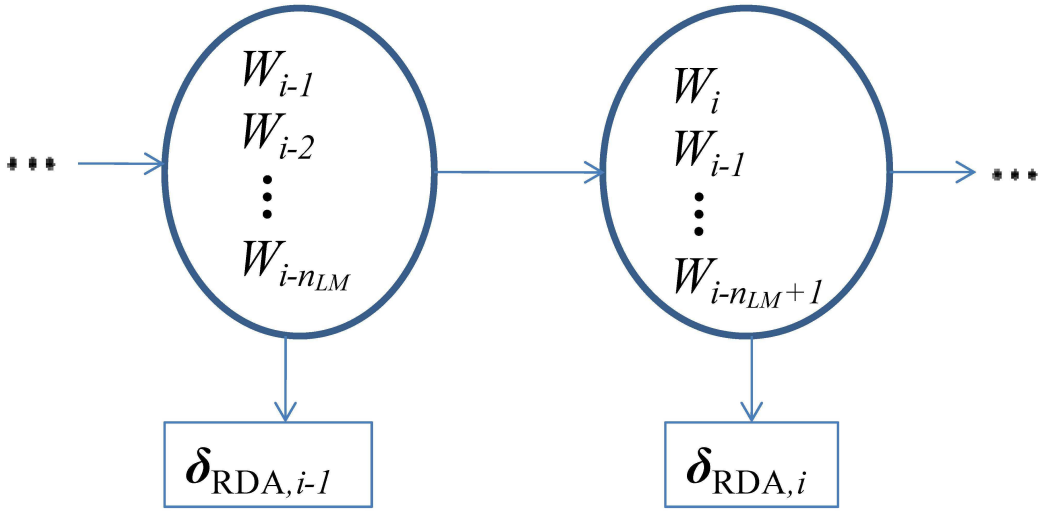


FIGURE 6.5: Hidden Markov Model used in recursive Bayesian approach for the fusion of language model with EEG scores.

Using recursive Bayesian estimation [124], marginalizing over older state variable the current state probability mass given all observed EEG scores is obtained in (6.15) where $\mathbf{s} = \{s_1, s_2, \cdots, s_{last}\}$ is a specific state.

$$P(\mathbf{W}_i = \mathbf{s}|\boldsymbol{\delta}_{\text{RDA},1}, \cdots, \boldsymbol{\delta}_{\text{RDA},i}) \propto p(\boldsymbol{\delta}_{\text{RDA},i}|W_i = s_{last}) \cdot \sum_{\mathbf{s}' \in \mathcal{S}^{n_{LM}}} P(\mathbf{W}_i = \mathbf{s}|\mathbf{W}_{i-1} = \mathbf{s}') P(\mathbf{W}_{i-1} = \mathbf{s}'|\boldsymbol{\delta}_{\text{RDA},1}, \cdots, \boldsymbol{\delta}_{\text{RDA},i-1}) \quad (6.15)$$

There are common variables between consecutive latent states, therefore $P(\mathbf{W}_i = \mathbf{s}|\mathbf{W}_{i-1} = \mathbf{s}')$ becomes zero for many $\mathbf{s}'$ which decreases the number of summations

required. If $i \leq n_{LM}$ it becomes (6.16),

$$P(\mathbf{W}_i = \mathbf{s}|\mathbf{W}_{i-1} = \mathbf{s}') = \begin{cases} P(W_i = s_{last}|\mathbf{W}_{i-1} = s_{1:last-1}) & , \ s_{1:last-1} = s'_{1:last} \\ 0 & , \text{ otherwise} \end{cases} \tag{6.16}$$

consequently the posterior probability of state variable becomes (6.17)

$$P(\mathbf{W}_i = \mathbf{s}|\boldsymbol{\delta}_{\text{RDA},1:i}) \propto p(\boldsymbol{\delta}_{\text{RDA},i}|W_i = s_{last})P(W_i = s_{last}|\mathbf{W}_{i-1} = s_{1:last-1})\cdot P(\mathbf{W}_{i-1} = s_{1:last-1}|\boldsymbol{\delta}_{\text{RDA},1:i-1}). \tag{6.17}$$

If $i > n_{LM}$, similarly we obtain (6.18) and (6.19).

$$P(\mathbf{W}_i = \mathbf{s}|\mathbf{W}_{i-1} = \mathbf{s}') = \begin{cases} P(W_i = s_{last}|\mathbf{W}_{i-1} = s_{1:last-1}) & , \ s_{1:last-1} = s'_{2:last} \\ 0 & , \text{ otherwise} \end{cases} \tag{6.18}$$

$$\begin{aligned} P(\mathbf{W}_i = \mathbf{s}|\boldsymbol{\delta}_{\text{RDA},1:i}) \propto \ & p(\boldsymbol{\delta}_{\text{RDA},i}|W_i = s_{last})\cdot \\ & P(W_i = s_{last}|W_{i-1} = s_{last-1}, \cdots, W_{i-n_{LM}+1} = s_1)\cdot \\ & \sum_{s'_0 \in \mathcal{S}} P(W_{i-1} = s_{last-1}, \cdots, W_{i-n_{LM}+1} = s_1, W_{i-n_{LM}} = s'_0|\boldsymbol{\delta}_{\text{RDA},1:i-1}) \end{aligned} \tag{6.19}$$

Assuming all EEG scores for an epoch are conditionally independent given the class labels, we can calculate $p(\boldsymbol{\delta}_{\text{RDA},i}|W_i = s)$ as (6.20),

$$\begin{aligned} p(\boldsymbol{\delta}_{\text{RDA},i}|W_i = s) = & \left(\prod_{n_s=1}^{N_S} p(\delta_{\text{RDA},i}(\mathbf{x}_{s,n_s})|c_s = 1) \right) \cdot \\ & \prod_{s' \in S \setminus \{s\}} \prod_{n_{s'}=1}^{N_S} p(\delta_{\text{RDA},i}(\mathbf{x}_{s',n_{s'}})|c_{s'} = 0) \end{aligned}$$

where $p(\delta_{\text{RDA},i}(\mathbf{x}_{s,n_s}|c_s = c)$ is estimated using KDE as in 3.2.3, $P(W_i = s_{last}|W_{i-1} =$

$s_{last-1}, \cdots, W_{i-n_{LM}+1} = s_1)$ or $P(W_i = s_{last}|\mathbf{W}_{i-1} = s_{1:last-1})$ is estimated using n-gram language model and $P(\mathbf{W}_{i-1} = s_{1:last-1}|\boldsymbol{\delta}_{\text{RDA},1:i-1})$ is calculated recursively.

Finally, at any epoch, we can estimate the latest substring using maximum likelihood,

$$\hat{\mathbf{W}}_i = \arg \max_{\mathbf{s} \in \mathcal{S}^{n_{LM}}} P(\mathbf{W}_i = \mathbf{s}|\boldsymbol{\delta}_{\text{RDA},1:i}).$$

Similarly, at i^{th} epoch, we can estimate W_j for $1 \leq j \leq i - n_{LM}$ using maximum likelihood as (6.20).

$$\begin{aligned}\hat{W}_j &= \arg \max_{s_0 \in \mathcal{S}} P(W_j = s_0|\boldsymbol{\delta}_{\text{RDA},1:i}) = \arg \max_{s_0 \in \mathcal{S}} P(W_j = s_0|\boldsymbol{\delta}_{\text{RDA},1:j+n_{LM}}) \\ &= \arg \max_{s_0 \in \mathcal{S}} \\ &\sum_{s'_1, \cdots, s'_{n_{LM}-1} \in \mathcal{S}} P(W_{j+n_{LM}-1} = s'_{n_{LM}-1}, \cdots, W_{j+1} = s'_1, W_j = s_0|\boldsymbol{\delta}_{\text{RDA},1:j+n_{LM}}) \quad (6.20)\end{aligned}$$

6.3.1 Experimental Offline Analysis

For this study, the same data as in 6.1.1 are used and the same preprocessing is applied. The language model was trained as described in 5. 36000 words are randomly drawn from a word lexicon weighted according to word occurrences and letter count. Fusion results were obtained for an n-gram model order of 4. The EEG scores

	1 sequence	2 sequences	3 sequences
Naïve Bayesian fusion	(0.26, 0.36)	(0.46, 0.64)	(0.63, 0.82)
Recursive Bayesian fusion	(0.35, 0.49)	(0.64, 0.73)	(0.76, 0.87)

TABLE 6.5: The minimum and the maximum values of the correct symbol selection ratios over different sessions for varying the number of sequences used in EEG classification for naïve Bayesian fusion and recursive Bayesian fusion approaches.

were assumed to have been evaluated for $N_S = 1$, 2, and 3 sequences (to evaluate the contribution of multi-trial information) to decide if a letter under evaluation was a desired target letter or not. In the results, only EEG data from the first N_S

sequences of the epoch were used for classification for each selected sequence count. Ratio of total number of letters selected correctly and the total number of letters is presented in Table 6.5 for all sessions.

In all of the sessions, the approach with recursive Bayesian estimation outperformed considerably the naïve fusion rule for all sequences per epoch, where the performances increase monotonically with the number of sequences as expected. The results are very promising, however it is important to note that there is no erasure symbol in these simulations. Since the naïve approach is less resistant to errors, existence of an erasure symbol might improve its performance more considerably than the other. As future work, we are planning to collect data for a similar analysis including the erasure symbol and using different model orders, which would result in a more accurate comparison.

The incorporation of the recursive Bayesian approach to the real time system would also be useful to assess its performance. However this approach has some disadvantages which might become important during the real time implementation. Firstly, it requires a large amount of memory since it keeps all the states from the previous selection. Secondly, the interface needs to be more confusing for the subject since previously selected symbols are ambiguous. These considerations become significantly important when the system is being implemented in real time.

Chapter 7

Real Time System

7.1 System Design

RSVP Keyboard^{TM} is a complete BCI typing system. It consists of the following main components

- *Presentation:* Presentation paradigms utilized in RSVP Keyboard^{TM} are implemented in Matlab the Psychophysics Toolbox extensions [125–127]. Triggers corresponding to the visual stimuli are sent to the EEG amplifiers via parallel port. For laptops containing express card slot, we have used an ExpressCard to parallel port converter.
- *Data Acquisition:* EEG signal acquisition is done using a g.USBamp biosignal amplifier via Matlab.
- *Language Model:* Pre-trained language models are run on a local server on the experiment computer and accessed via TCP-IP.
- *Signal Monitoring:* A signal monitoring interface is designed to visually observe the signals in real time.

7.2 Operation Modes

RSVP Keyboard^{TM} consists of four operation modes. One of the operation modes is for calibration and the remainder are for typing.

7.2.1 Calibration

Calibration occurs during the session where the signal statistics for the subject are learned. A predefined target symbol is selected and shown to the user before an RSVP sequence. Subjects are instructed to look for the target symbol in the sequence and show intent for the predefined symbol. Using the collected supervised data, the EEG feature extraction parameters are optimized. Typing performance might be estimated using the approach proposed in Section 7.3.

7.2.2 Spelling

Free spelling is the main typing mode. The subjects can type whatever they want and use the system to communicate.

7.2.3 Copy Phrase

Copy phrase mode is a guided typing mode. Subjects are asked to copy a phrase in a sentence, letter by letter. Subjects are also instructed to delete any of their mistakes. Copying continues to the next phrase if one of the following conditions is satisfied.

- Exactly copying the phrase (success).
- Timeout (failure): Reaching to a predefined time allowed for the phrase.

- Sequence Limit (failure): The number of sequences spent is limited by

  $$\kappa \times (\text{Number of letters in phrase}) \times (\text{Max. number of sequences per epoch}),$$

  where κ is a scale parameter, which is currently set to 2.

- Consecutive erroneously typed symbols (failure): If there exists 3 consecutive unintended letters in the typed text, it is considered as failure.

7.2.4 Mastery Task

Mastery task is a variation of the copy phrase mode, that is designed to train the subject on using the system. The phrases are divided into levels with increasing difficulty. The difficulty of the phrases are assessed by the ratio of the context probability of the intended symbol to unintended most likely symbol for each letter in the phrase. Each level consists of multiple sets of phrases. Subjects pass through levels by successfully completing 2/3 of the phrases in a set.

7.3 Performance Estimation Using Simulation

Not all parameters can be easily theoretically optimized in a complex system. Experimentation on human subjects is a very time consuming process, therefore trial and error with the user-in-the-loop is not feasible. In order to be able to get a reliable performance estimate for a setting, one might need to do multiple sessions that might take more than an hour each. If the researcher/clinician would like to do comparative experiments on different operation scenarios, e.g., number of maximum sequences per epoch, backspace probability, bandpass cut-off frequencies, etc., then the experiments might take exponentially increasing amount of time (by parameter space dimensionality, for grid search). The number of sessions needed might exponentially grow with the number of parameters being searched.

As a supplement to experimentation, we utilize a simulation methodology to estimate the performance of the system. In simulation, the copy phrase task is employed, since the user would need to attempt perfect phrase copying and the user strategy can be simulated exactly. In other words, the simulated user in the simulation always attempts to correct any errors. Consequently, at any instant the intended action is known. Afterwards, EEG features are sampled from (3.3), the conditional pdf estimates learned after cross validation. The features corresponding to the trials of the intended symbol are sampled from the target pdf and rest are sampled from the non-target pdf.

The simulations are based on the assumption of i.i.d. target and non-target trials EEG evidences. It ignores potentially different responses to different non-target symbols (nonstationarity). It ignores fatigue effects or change in signal statistics or temporal dynamics of the EEG due to other physiological reasons, as well. However it is still a very beneficial tool for estimating the performance. In our experience, actual copy phrase results with subject-in-the-loop are usually consistent with Monte Carlo simulation estimates.

7.4 Experimental Analysis

EEG is recorded using a g.USBamp biosignal amplifier using active g.Butterfly electrodes from G.tec (Graz, Austria) at 256Hz sampling rate. The EEG electrodes are applied using the g.GAMMAcap (electrode cap) and positioned according to the International 10/20 System with locations of Fp1, Fp2, F3, F4, FZ, Fc1, Fc2, Cz, P1, P2, C1 C2, Cp3, Cp4, P5, P6. Internal to the amplifier, signals were filtered by a nonlinear-phase [2,60]Hz Butterworth bandpass filter and a 60Hz notch filter. Afterwards, signals were filtered further by the previously mentioned [1.5,42]Hz linear-phase bandpass filter (our design). The filtered signals were downsampled to 128Hz. For each channel, stimulus-onset-locked time windows of [0,500)ms following

- THE DOG **WILL** BITE YOU
- LEARN TO WALK BEFORE YOU **RUN**
- GOOD AT ADDITION **AND** SUBTRACTION
- INTERACTIONS **BETWEEN** MEN AND WOMEN
- ACCOMPANIED BY AN **ADULT**
- **PLEASE** KEEP THIS CONFIDENTIAL
- FISH **ARE** JUMPING
- SHE WEARS TOO MUCH **MAKEUP**

TABLE 7.1: Phrase and sentence pairs to be copied in copy phrase task. The phrase to be copied is marked as **bold** in the sentence.

each image onset was taken as the stimulus response and features were extracted as described earlier.

Ten healthy subjects (4 female, 6 male) were recruited for this study. Each subject participated in the experiments for 5 or 6 sessions. The first session was a calibration session with randomly selected 100 sequences of 10 trials with 200 ms inter stimulus interval. If the single trial area under the ROC curve is lower than 0.7, the calibration session is repeated once again. Consequently, 4 copy task sessions with following parameter selections are conducted.

- Scenario 1: No language model, 28 trials/sequence, at most 4 sequences/epoch
- Scenario 2: 6-gram model, 28 trials/sequence, at most 4 sequences/epoch
- Scenario 3: 6-gram model, 14 trials/sequence, at most 8 sequences/epoch
- Scenario 4: 6-gram model, 7 trials/sequence, at most 16 sequences/epoch

In the copy task, 8 phrase and sentence pairs, given in Table 7.1 are presented to be copied. These sentences are selected from MacKenzie & Soukoreff's balanced phrase list [128]. Results of the experiments (one user-in-the-loop session per scenario) and simulations (average of Monte Carlo runs) are given in Fig. 7.1 and Fig. 7.2. As seen from these plots, average simulation estimates of probability of completion and duration are qualitatively similar to actual experimental session outcomes. To

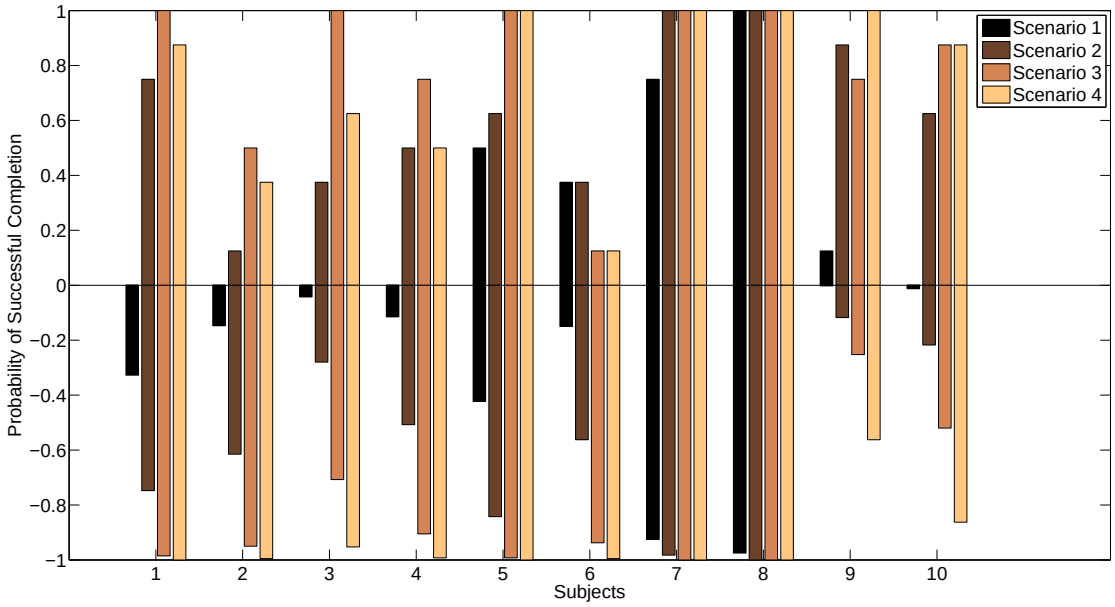


FIGURE 7.1: Successful completion probability for each subject for each scenario for simulation and experiment. Top bars represent the experimental results and bottom bars represent the simulation results with taller the bar is better. Probability of 1 corresponds to successful completion of all phrases and probability of 0 corresponds to failure to complete any of the phrases. Consider only the absolute value of the vertical axis values.

estimate the performance in a similar fashion to the BCI literature, information transfer rates (ITR) calculated according to Section 4.1 are given in Table 7.2 and Table 7.3. The typing speeds according to online performance are given in Table 7.4. In calculation of the typing speed, the amount of text that is written accurately per minute is considered.

We can compare these results with the other leading efforts on online typing BCIs in the literature (Table 7.5). It is important to note that these results are not directly comparable due to differences on the stimulation methodology and performance estimation. These systems might experimentally differ due to gaze dependency, subject inclusion criteria, or text used for estimating the performance. Moreover some of the systems do not require error correcting during the typing process, but try to estimate the additional load corresponding to erasure. According to these

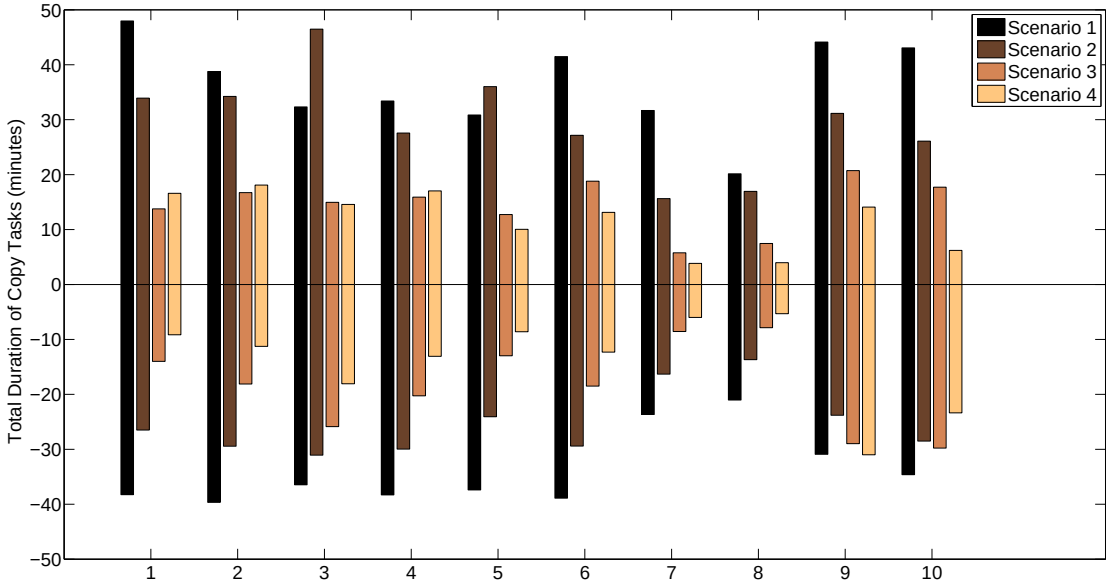


FIGURE 7.2: Duration to copy 8 phrases for each subject for each scenario for simulation and experiment. This corresponds to the total duration spent on attempting to type all of the phrases. Top bars represent the experimental results and bottom bars represent the simulation results with shorter the bar is better. Consider only the absolute value of the vertical axis values.

	Scenario 1 No lang. model Max. 4 seq 28 trials/seq	**Scenario 2** 6-gram model Max. 4 seq 28 trials/seq	**Scenario 3** 6-gram model Max. 8 seq 14 trials/seq	**Scenario 4** 6-gram model Max. 14 seq 7 trials/seq
User 1	1.10	2.18	3.81	2.56
User 2	0.24	0.60	1.68	1.17
User 3	0.10	1.70	3.81	2.19
User 4	0.84	1.87	2.68	1.43
User 5	1.74	2.00	3.81	3.69
User 6	1.36	1.09	0.40	0.37
User 7	3.13	4.51	4.29	4.81
User 8	4.51	4.51	4.29	4.51
User 9	1.06	2.81	2.76	3.16
User 10	0.78	2.23	3.00	3.59
Average	1.49	2.35	3.05	2.75
Maximum	4.51	4.51	4.29	4.81

TABLE 7.2: Information Transfer Rate (bits/symbol) for different scenarios and different users.

	Scenario 1 No lang. model Max. 4 seq 28 trials/seq	**Scenario 2** 6-gram model Max. 4 seq 28 trials/seq	**Scenario 3** 6-gram model Max. 8 seq 14 trials/seq	**Scenario 4** 6-gram model Max. 14 seq 7 trials/seq
User 1	2.17	4.92	9.63	5.56
User 2	0.46	1.27	3.61	2.14
User 3	0.19	3.43	8.71	3.92
User 4	1.66	4.59	5.84	2.30
User 5	3.41	4.47	10.59	9.61
User 6	2.64	2.45	0.82	0.76
User 7	7.08	16.71	29.17	29.14
User 8	12.61	16.11	22.27	30.69
User 9	2.02	6.63	5.95	7.18
User 10	1.50	5.56	7.36	11.85
Average	3.37	6.62	10.40	10.32
Maximum	12.61	16.71	29.17	30.69

TABLE 7.3: Information Transfer Rate (bits/minute) for different scenarios and different users.

	Scenario 1 No lang. model Max. 4 seq 28 trials/seq	**Scenario 2** 6-gram model Max. 4 seq 28 trials/seq	**Scenario 3** 6-gram model Max. 8 seq 14 trials/seq	**Scenario 4** 6-gram model Max. 14 seq 7 trials/seq
User 1	0.16	0.92	1.99	1.07
User 2	0.14	0.36	0.88	0.41
User 3	0.12	0.46	1.80	0.88
User 4	0.15	1.15	1.45	0.60
User 5	0.82	0.83	2.19	1.97
User 6	0.49	0.62	0.27	0.28
User 7	1.35	3.51	6.13	6.06
User 8	2.65	3.39	4.68	6.45
User 9	0.20	1.39	1.04	1.38
User 10	0.32	1.38	1.37	2.85
Average	0.64	1.40	2.18	2.19
Maximum	2.65	3.51	6.13	6.45

TABLE 7.4: Typing speed in characters per minute for different scenarios and different users. Only correctly typed text at the end is considered effective towards the number of characters.

		Best (bpm)	**Best** (sym/min)	**Mean** (bpm)	**Mean** (sym/min)
Townsend et al [129]	Matrix	39	6.3	31	5.1
Serby et al [39]	Matrix	-	6.9	-	4.5
Trader et al [49]	Matrix	-	-	9.8	2
Blankertz et al [80]	Hex-o-spell	-	4.6-7.6	-	-
Acqualagna et al [46]	RSVP	7	1.4	-	1.4
RSVP Keyboard^{TM}	RSVP	31	6.4	10	2.2
Chapter 8	RSVP	36	7.5	13	2.6

TABLE 7.5: Comparison with other BCI typing systems in the literature. It is important to note that these results are not directly comparable due to difference in the operation.

results, the maximum typing speed of RSVP Keyboard^{TM} was at least on par with other BCI typing systems including the ones which are gaze dependent. However average performance was worse than the matrix based spellers.

Chapter 8

Consideration of Trial Dependency

In P300-based BCIs, the problem of detecting a positive intent is often considered as a binary classification problem. This assumption groups all non-target trials in one class and target trials in another. Correspondingly, SVM, LDA, step-wise LDAs or RDA have usually been employed in a two class manner to separate between classes. It is usually assumed that EEG data corresponding to the trials immediately after the target one also belong to the same class or distribution as the rest of the non-target trials. However this model/assumption might not be a good fit in certain circumstances. If the inter-symbol-interval is shorter than the duration of the ERPs, there is an overlap of the consecutive ERPs. The ERP response corresponding to a target stimuli might last longer than 500 ms. Consequently, if the trial duration is 200 ms, there is overlap between ERPs of consecutive trials.

As in most of the BCI literature, we also first assumed all trials to be independent given the *targetness*, i.e. condition of being target or not, of each trial. Consequently, the effect of the overlap of the ERPs were, so far, ignored even for the consecutive trials. To be able to consider this effect, one can relax this assumption. We consider a dependency model that considers the dependency to the targetness of previous trials. The targetness or non-targetness of recent trials are considered to affect the EEG

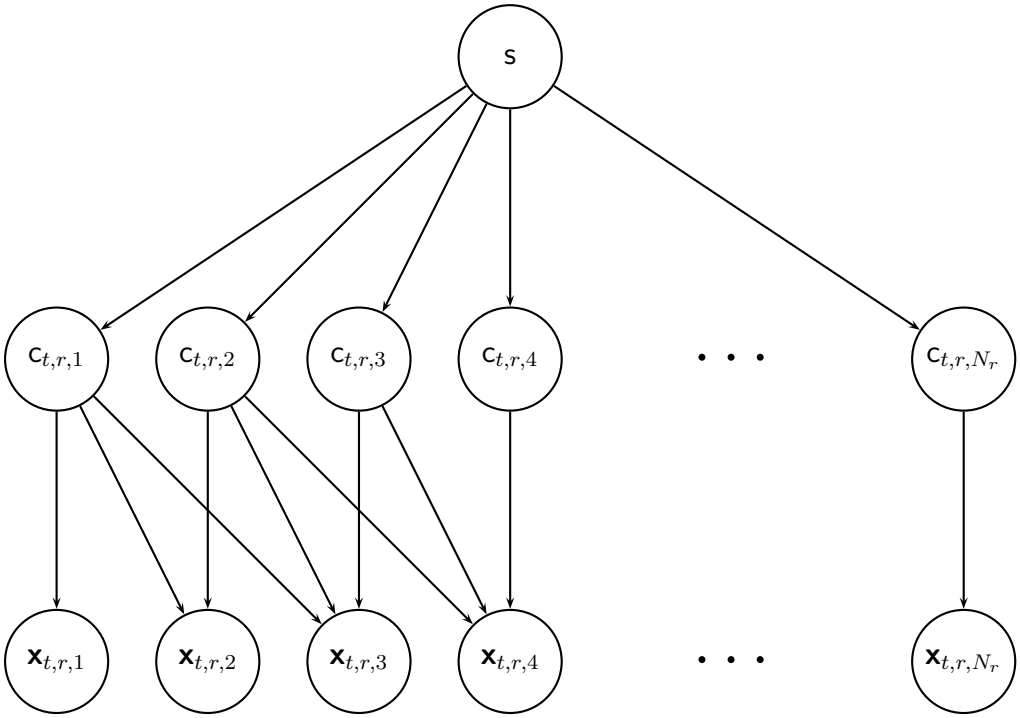


FIGURE 8.1: Graphical model considering the dependency between trials. EEG features corresponding to a trial are affected by last M trials including the current one. In this depiction, M is taken as 3. s represents the intended symbol, $\mathbf{x_{t,r,n}}$ and $\mathsf{c_{t,r,n}}$ represents the EEG features and class labels corresponding to trial n in sequence r and epoch t, respectively.

features in a manner consistent with the duration between trials. The corresponding graphical model is depicted in Fig. 8.1.

According to this revised model, it is assumed that at any time instant at most M last trials affect the current EEG feature. In this approach, given the targetness labels of last M trials, the response comes from the same distribution. For an inter-stimulus-interval of 200 ms, selecting $M > 2$ would be a reasonable choice, considering that the ERP response for the target trials might take longer than 400ms. We will select $M = 3$ in the experimental studies conducted, however in the actual system M is selected according to the inter stimulus interval of the RSVP presentation and the window duration, which specifies the temporal limit on the expected ERP responses.

To be able to construct the fusion rule for this new model we can start from (6.12), which only assumes the conditional independence of the EEG evidence and the

context evidence. Obviously as before (6.12) reduces to (8.1) after utilizing Bayes' Theorem.

$$P(\mathsf{s} = s|\boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) \propto f(\boldsymbol{\mathcal{X}}_t = \mathcal{X}|\mathsf{s} = s)P(\mathsf{s} = s|\mathbf{w}_t = \mathbf{w}). \tag{8.1}$$

We can assume EEG evidences from different sequences are conditionally independent given the intended symbol. This assumption essentially corresponds to the expectation of sequence order invariance of the response of the subject. For example, this assumption ignores that the user might be responding differently to the later sequences compared to earlier ones due to fatigue or excitement. More formally we can write this assumption as $\mathbf{X}_{t,1} \perp\!\!\!\perp \mathbf{X}_{t,2} \perp\!\!\!\perp \cdots \perp\!\!\!\perp \mathbf{X}_{t,R}|\mathsf{s}$, where $\mathbf{X}_{t,r}$ is the r.v. corresponding to the EEG evidence corresponding to r^{th} sequence, R is the number of sequences, and $\boldsymbol{\mathcal{X}}_t = \{\mathbf{X}_{t,1}, \mathbf{X}_{t,2}, \cdots, \mathbf{X}_{t,R}\}$. Utilizing this conditional independence assumption on (8.1), we obtain (8.2).

$$P(\mathsf{s} = s|\boldsymbol{\mathcal{X}}_t = \mathcal{X}, \mathbf{w}_t = \mathbf{w}) \propto \left(\prod_{r=1}^{R} f(\mathbf{X}_{t,r} = \mathbf{X}_r|\mathsf{s} = s)\right) P(\mathsf{s} = s|\mathbf{w}_t = \mathbf{w}), \tag{8.2}$$

where $\mathcal{X} = \{\mathbf{X}_{t,1}, \mathbf{X}_{t,2}, \cdots, \mathbf{X}_{t,R}\}$ corresponding observations split based on sequences.

Now to estimate $f(\mathbf{X}_{t,r} = \mathbf{X}_r|\mathsf{s} = s)$, we will use the new graphical model. Let $\mathbf{x}_{t,r,n}$ be the EEG features corresponding to the temporal window corresponding to times between n^{th} trial and $(n+1)^{\text{st}}$ trials, with $\mathbf{X}_{t,r} = \{\mathbf{x}_{t,r,1}, \mathbf{x}_{t,r,2}, \cdots, \mathbf{x}_{t,r,N_r}\}$, where N_r is the number of trials in sequence r. From the graphical model we obtain the

factorization in (8.3).

$$
\begin{aligned}
f(\mathsf{s}=s,\ \ \mathsf{c}_{t,r,1}=c_{r,1},\cdots,\mathsf{c}_{t,r,N_r}=c_{t,r,N_r},\mathbf{x}_{t,r,1}=\mathbf{x}_{r,1},\cdots,\mathbf{x}_{t,r,N_r}=\mathbf{x}_{r,N_r}) = \\
P(\mathsf{s}=s)P(\mathsf{c}_{t,r,1}=c_{r,1},\cdots,\mathsf{c}_{t,r,N_r}=c_{t,r,N_r}|\mathsf{s}=s)\cdot \\
\left[\prod_{n=1}^{T-1} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,1}=c_{r,1},\cdots,\mathsf{c}_{t,r,n}=c_{r,n})\right]\cdot \\
\left[\prod_{n=T}^{N_r} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,n-T+1}=c_{r,n-T+1},\cdots,\mathsf{c}_{t,r,n}=c_{r,n})\right]
\end{aligned}
\tag{8.3}
$$

If the intended symbol is known, the class label of each trial is uniquely determined. Since the symbols corresponding to each individual trial is known, the class label of a trial becomes 1 (target), if its symbol matches the intended one. Otherwise, the class label becomes 0 (non-target). For each intended symbol, we can construct a class label sequence corresponding to the trials. Let $\zeta_r : \mathcal{S} \rightarrow \{0,1\}^{N_r}$, be the function that evaluates the class label sequence for a symbol, and $\zeta_r(s)_n$ be n^{th} output of $\zeta_r(s)$. For example, if the trial symbols in the sequence are $[K, W, D, S, E, H, Q, G, U, I]$, corresponding ζ functions become,

$$
\begin{aligned}
\zeta_1(A) &= [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] \\
\zeta_1(D) &= [0,0,1,0,0,0,0,0,0,0,0,0,0,0,0] \\
\zeta_1(U) &= [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0].
\end{aligned}
$$

Corresponding to determinism of the class labels given the intended symbol s, we can write $P(\mathsf{c}_{t,r,1}=c_{r,1},\cdots,\mathsf{c}_{t,r,N_r}=c_{t,r,N_r}|\mathsf{s}=s)$ as

$$
P(\mathsf{c}_{t,r,1}=c_{r,1},\cdots,\mathsf{c}_{t,r,N_r}=c_{t,r,N_r}|\mathsf{s}=s) = \prod_{n=1}^{N_r} \delta[c_{r,n},\zeta_r(s)_n], \tag{8.4}
$$

where $\delta[i,j]$ represents the Kronecker delta function, i.e. $\delta[i,j] = \begin{cases} 1, i=j \\ 0, i\neq j. \end{cases}$

Marginalizing (8.3) with respect to class labels and using 8.4, we obtain (8.5).

$$f(\mathsf{s}=s,\mathbf{x}_{t,r,1}=\mathbf{x}_{r,1},\cdots,\mathbf{x}_{t,r,N_r}=\mathbf{x}_{r,N_r})= \sum_{[c_{r,1},\cdots,c_{r,N_r}]\in\{0,1\}^{N_r}} \left\{ P(\mathsf{s}=s)\prod_{n=1}^{N_r}\delta[c_{r,n},\zeta_r(s)_n]\cdot \left[\prod_{n=1}^{T-1} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,1}=c_{r,1},\cdots,\mathsf{c}_{t,r,n}=c_{r,n})\right]\cdot \left[\prod_{n=T}^{N_r} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,n-T+1}=c_{r,n-T+1},\cdots,\mathsf{c}_{t,r,n}=c_{r,n})\right]\right\} \quad (8.5)$$

Using the sifting property of Kronecker delta function, we obtain (8.6).

$$f(\mathsf{s}=s,\mathbf{x}_{t,r,1}=\mathbf{x}_{r,1},\cdots,\mathbf{x}_{t,r,N_r}=\mathbf{x}_{r,N_r})=P(\mathsf{s}=s)\cdot \left[\prod_{n=1}^{T-1} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,1}=\zeta_r(s)_1,\cdots,\mathsf{c}_{t,r,n}=\zeta_r(s)_n)\right]\cdot \left[\prod_{n=T}^{N_r} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,n-T+1}=\zeta_r(s)_{n-T+1},\cdots,\mathsf{c}_{t,r,n}=\zeta_r(s)_n)\right] \quad (8.6)$$

Finally, using the definition of conditional probability, we obtain 8.7.

$$f(\mathbf{X}_{t,r}=\mathbf{X}_r|\mathsf{s}=s)=\left[\prod_{n=1}^{T-1} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,1}=\zeta_r(s)_1,\cdots,\mathsf{c}_{t,r,n}=\zeta_r(s)_n)\right]\cdot \left[\prod_{n=T}^{N_r} f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,n-T+1}=\zeta_r(s)_{n-T+1},\cdots,\mathsf{c}_{t,r,n}=\zeta_r(s)_n)\right] \quad (8.7)$$

8.1 Estimating Trial Conditionals

To be able to calculate $f(\mathbf{X}_{t,r}=\mathbf{X}_r|\mathsf{s}=s)$, the only remaining part is to estimate $f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,1}=\zeta_r(s)_1,\cdots,\mathsf{c}_{t,r,n}=\zeta_r(s)_n)$ and $f(\mathbf{x}_{t,r,n}=\mathbf{x}_{r,n}|\mathsf{c}_{t,r,n-T+1}=\zeta_r(s)_{n-T+1},\cdots,\mathsf{c}_{t,r,n}=\zeta_r(s)_n)$. Due to the limitation on the duration of the calibration session, we would significantly suffer from low number of samples in the

estimation of these conditionals. Therefore, we will make further assumptions or approximations to be able to operate in low sample conditions.

Firstly, we will make the assumption of existence of non-target trials before the first trial of the sequence. This assumption allows all of the trial data, $\mathbf{x}$, to depend on same number of classes including the trials with trial index $n < T$. Equivalently, $\forall n < T$,

$$f(\mathbf{x}_{t,r,n} = \mathbf{x}_{r,n} | \mathsf{c}_{t,r,1} = \zeta_r(s)_1, \cdots, \mathsf{c}_{t,r,n} = \zeta_r(s)_n) =$$
$$f(\mathbf{x}_{t,r,n} = \mathbf{x}_{r,n} | \mathsf{c}_{t,r,n-T+1} = 0, \cdots, \mathsf{c}_{t,r,0} = 0, \mathsf{c}_{t,r,1} = \zeta_r(s)_1, \cdots, \mathsf{c}_{t,r,n} = \zeta_r(s)_n).$$

Conformably, the presentation paradigm can be changed to fit to this assumption. We can insert distractor trials, which are never targets, prior to first trial in the sequence. Similarly, distractors can be added to the end of the sequence to better capture the responses to the last trials. In the experiments, we added $T-1$ randomly selected digits pre-sequence and post-sequence as distractors. Ultimately, to prevent unnecessary time loss in the typing speed, we will let to have undivided continuous presentation of trials, instead of artificially divided sequences. By detecting the attention level of the user, blinks or eye movements, we should aim to let the subject rest whenever they desire, instead of forcing them to obey the compulsory sequencing imposed currently.

Furthermore, if we assume there does not exist more than one repetition of any trial symbol in a sequence, or there exist at least $T-1$ other trials between the repetitions of any trial symbol, then the set of class conditionals requiring estimation becomes significantly restricted. For $T = 3$, we would only need to estimate the following

class conditionals:

$$f(\mathbf{x}_{t,r,n} = \mathbf{x}_{r,n} | \mathsf{c}_{t,r,n-2} = 0, \mathsf{c}_{t,r,n-1} = 0, \mathsf{c}_{t,r,n} = 0)$$
$$f(\mathbf{x}_{t,r,n} = \mathbf{x}_{r,n} | \mathsf{c}_{t,r,n-2} = 1, \mathsf{c}_{t,r,n-1} = 0, \mathsf{c}_{t,r,n} = 0)$$
$$f(\mathbf{x}_{t,r,n} = \mathbf{x}_{r,n} | \mathsf{c}_{t,r,n-2} = 0, \mathsf{c}_{t,r,n-1} = 1, \mathsf{c}_{t,r,n} = 0)$$
$$f(\mathbf{x}_{t,r,n} = \mathbf{x}_{r,n} | \mathsf{c}_{t,r,n-2} = 0, \mathsf{c}_{t,r,n-1} = 0, \mathsf{c}_{t,r,n} = 1).$$

Estimation of class conditionals may be done parametrically, using a distribution model, or non-parametrically, e.g. using kernel density estimation. It is imperative to note that applying a binary classifier for dimensionality reduction might cause this proposed model to lose its meaning. Likewise, in the processing of the previous chapters, we were applying dimensionality reduction via RDA with two class assumption. Consequently, to have consistent assumptions, we utilize multi-class RDA directly as the pdf estimator for the temporal dependency model. This corresponds to assuming each class distribution to belong to a different multivariate normal distribution, where the distribution parameters are estimated via shrinkage and regularization. It is worthwhile to remark that one can apply other dimensionality reduction and pdf estimation methodology.

8.2 Experimental Analysis

For experimental analysis of the proposed temporal dependency model, 8 healthy participants, 5 males and 3 females, were recruited. Users attended three sessions, 1 calibration and 2 copy-phrase tasks. EEG recording and the preprocessing procedure were almost the same as Section 7.4. The differences are listed in the following.

- For both models, the linear-phase bandpass filter was replaced with another design with the cutoff frequencies of [2.5, 17] Hz. Filtering is followed by

an increased downsampling to $256/6 \approx 43$ Hz. Changing the filter allowed us to downsample further without causing aliasing. With the construction of temporal dependency model, the conditional pdf estimates became more reliant on the conditional probability estimates from the RDA model. Downsampling is expected to decrease the number of features, correspondingly decrease the reliance on the shrinkage and regularization procedures. This would potentially give us more reliable pdf estimates.

- We added 2 distractors which were randomly selected digits to the beginning and end of the sequences.
- Maximum number of sequences per epoch was set to 16.
- Number of trials per sequence was 14.

The calibration session consisted of 100 sequences of 10 trials, which contained the target symbol in a randomized location in the sequence. We used the same calibration file to assess the single sequence target selection performance and learn the models to use in the copy-phrase tasks for two different modeling approaches, with and without consideration of temporal dependency between trials. In Fig. 8.1, the offline comparison of the models under cross validation is given. For each sequence, maximum likelihood decision is made on selecting the target. It is important to note that these hit rates are expected to be higher than real time single sequence maximum likelihood selection performance, since there exists only 10 trials per sequence compared to 28 candidate symbols in typing operation. This corresponds to having a chance level of 10% compared to 3.6%. For almost all of the subjects, single sequence target detection rate has increased using the temporal dependency model.

Following and using the calibration, subjects attended to two copy-phrase tasks, one for the original model and one for the model that considers the temporal dependency. In these copy tasks, subjects tried to copy the sentences in Table 7.1. Comparison

	Without the Consideration of Temporal Dependency	**With the Consideration of Temporal Dependency**
User 1	0.37	0.40
User 2	0.44	0.49
User 3	0.70	0.77
User 4	0.52	0.57
User 5	0.31	0.29
User 6	0.76	0.77
User 7	0.26	0.33
User 8	0.45	0.49
Average	0.47	0.51
Maximum	0.76	0.77

TABLE 8.1: Hit rates for correctly selecting the target trial in single sequence under cross validation. Chance level is 0.10.

of ITR and typing speeds are given in Tables 8.2, 8.3 and 8.4. Among these limited number of subjects, we noticed that the subjects with higher typing speeds (above 1 char/min) experienced significant typing speed gains from the temporal dependency model, whereas others experienced performance drops. When we investigated the experimental results in greater detail, we noticed that the temporal dependency model tended to make earlier decisions even if the decision and the intended symbol did not match. This might be due to the high dimensionality and unreliability of the pdf estimation in the temporal dependency model. Correspondingly it might possible to experience non-robustness to outliers. However making a claim on the actual reason requires further investigation and experimentation.

	Without the Consideration of Temporal Dependency	**With the Consideration of Temporal Dependency**
User 1	3.32	1.04
User 2	0.25	4.11
User 3	4.51	3.81
User 4	2.96	2.48
User 5	1.04	0.17
User 6	4.81	4.29
User 7	4.29	0.92
User 8	4.51	2.98
Average	3.21	2.48
Maximum	4.81	4.29

TABLE 8.2: Information Transfer Rate (bits/symbol) with or without temporal dependency modeling for different users.

	Without the Consideration of Temporal Dependency	**With the Consideration of Temporal Dependency**
User 1	3.67	2.74
User 2	1.72	8.06
User 3	18.65	28.04
User 4	8.22	10.70
User 5	0.75	0.86
User 6	28.26	35.81
User 7	4.39	2.53
User 8	11.93	15.02
Average	9.70	12.97
Maximum	28.26	35.81

TABLE 8.3: Information Transfer Rate (bits/minute) with or without temporal dependency modeling for different users.

	Without the Consideration of Temporal Dependency	**With the Consideration of Temporal Dependency**
User 1	0.72	0.32
User 2	0.47	1.69
User 3	3.92	5.79
User 4	1.78	1.97
User 5	0.15	0.49
User 6	5.88	7.53
User 7	0.92	0.36
User 8	2.51	2.79
Average	2.04	2.62
Maximum	5.88	7.53

TABLE 8.4: Typing speed in characters per minute for different scenarios and different users. Only correctly typed text at the end is considered effective towards the number of characters.

Chapter 9

Discussion and Future Work

In this dissertation, we have designed a context aware and gaze independent typing BCI, RSVP Keyboard^{TM} . Gaze independence [1] is accomplished through the use of RSVP. Only very few research groups are utilizing gaze independent presentation paradigms for typing. Most of them suffer from very low symbol rates compared to matrix-based versions. As a remedy, we probabilistically incorporated the context information to improve symbol rates.

In Chapter 6, we proposed a fusion methodology to jointly utilize ERP and context information. Firstly, we proposed a method to improve the labeling decision for a trial; target or nontarget. Via experimental offline analyses, we observed a significant accuracy increase when context information is utilized. Secondly, we proposed two Naïve Bayesian fusion approaches for online decision making purposes based on two different assumption sets. Additionally, we proposed a recursive Bayesian fusion methodology that does not commit to a single decision, but keeps track of all possible histories. This approach improved the performance significantly compared to the Naïve method in offline analysis. However, we did not implement the fusion with a recursive Bayesian method due to practical considerations of additional cognitive load and memory.

[1] of course, the user still needs to be able to see the screen and letters in the RSVP sequences

In Chapter 7, we gave the details for operational modes of the real time system. This includes a simulation methodology to estimate the typing performance for various settings and conditions. A comparative experiment-based approach would be very time consuming. We compared the real time typing performance with the experimental results, and observed a qualitative similarity.

In Chapter 8, we propose a model considering the temporal dependency between trials. Temporal dependency modeling slightly increased target selection performance in offline analysis, this reflected to real time system as a more significant increase in typing speed for users with typing performance above a certain level. However, it did not improve performance for users with lower performance. Temporal dependency model tended to make earlier decisions (sometimes incorrect), which might be due to unreliability of high dimensional pdf estimation with low number of samples. In future work, we will investigate potential ways to increase the conditioning of the probability density function. This might be achievable by artifact filtering and outlier rejection.

Even though RSVP Keyboard$^{\text{TM}}$ has a sufficiently high symbol rate to be able to compete with other BCI typing systems, a significant effort has not been made for the improvement of the EEG signal processing and classification procedure. Moreover, most of the BCI field heavily relies on utilization of readily available, prepackaged algorithms to solve the machine learning problems. Therefore there is potentially significant room for improvements in signal processing and future work. In future work, we would like to investigate the following.

- *Time-frequency analysis of EEG:* Most of the time-frequency analysis in BCI field has been done using discrete Wavelet transforms. However, dWT is potentially weak in denoising of signals with frequency content changing over time. A potentially promising approach would be, time-frequency analysis of target and nontarget ERP responses using Wigner-Ville distributions and

correspondingly designing filters in time or frequency based on the informativeness of time-frequency features [130, 131]. Designing a filter to extract a signal with frequency content changing over time, fractional Fourier transform or its cascaded extensions can also be used [130].

- *Continuous presentation:* Splitting the presentation into sequences significantly reduces the typing speed, since time is lost while nothing is shown on the screen. Modifying the presentation into a continuous sequence and detecting the attention level and blinks, we can activate the typing mode of operation whenever the user pays attention.

Bibliography

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

[2] ——, "Real-time detection of brain events in eeg," *Proceedings of the IEEE*, vol. 65, no. 5, pp. 633–641, 1977.

[3] H. Berger, "Über das elektrenkephalogramm des menschen," *European Archives of Psychiatry and Clinical Neuroscience*, vol. 87, no. 1, pp. 527–570, 1929.

[4] G. Schalk and J. Mellinger, *Human-Computer Interaction: Practical Guide to Brain-Computer Interfacing with Bci2000: General-Purpose Software for Brain-Computer Interface Research, Data Acquisition, Stimulus Presentation, and Brain Monitoring*. Springer, 2010.

[5] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "Bci2000: a general-purpose brain-computer interface (bci) system," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1034–1043, 2004.

[6] A. Delorme and S. Makeig, "Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis," *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.

[7] Y. Renard, F. Lotte, G. Gibert, M. Congedo, E. Maby, V. Delannoy, O. Bertrand, and A. Lécuyer, "Openvibe: an open-source software platform

to design, test, and use brain-computer interfaces in real and virtual environments," *Presence: teleoperators and virtual environments*, vol. 19, no. 1, pp. 35–53, 2010.

[8] P. Brunner, S. Joshi, S. Briskin, J. Wolpaw, H. Bischof, and G. Schalk, "Does the'P300'speller depend on eye gaze?" *Journal of Neural Engineering*, vol. 7, p. 056013, 2010.

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

[10] J. E. Huggins and D. Zeitlin, "The ultimate practical goal of most brain-computer inter-faces (bcis) is to operate devices that provide communi," *Brain-Computer Interfaces: Principles and Practice*, p. 197, 2012.

[11] C. Zickler, A. Riccio, F. Leotta, S. Hillian-Tress, S. Halder, E. Holz, P. Staiger-Sälzer, E.-J. Hoogerwerf, L. Desideri, D. Mattia *et al.*, "A brain-computer interface as input channel for a standard assistive technology software," *Clinical EEG and Neuroscience*, vol. 42, no. 4, pp. 236–244, 2011.

[12] D. Beukelman and P. Mirenda, "Augmentative and alternative communication," 2005.

[13] S. Fager, D. R. Beukelman, M. Fried-Oken, T. Jakobs, and J. Baker, "Access interface strategies," *Assistive Technology*, vol. 24, no. 1, pp. 25–33, 2012.

[14] H. C. Shane, S. Blackstone, G. Vanderheiden, M. Williams, and F. DeRuyter, "Using aac technology to access the world," *Assistive Technology*, vol. 24, no. 1, pp. 3–13, 2012.

[15] I. Fishman, *Electronic communication aids: Selection and use*. College-Hill, 1987.

[16] A. Kübler, E. Holz, T. Kaufmann, and C. Zickler, "A user centred approach for bringing bci controlled applications to end-users," 2013.

[17] J. Wolpaw and E. Wolpaw, *Brain-Computer Interfaces: Principles and Practice*. OUP USA, 2012.

[18] S. Sutton, M. Braren, J. Zubin, and E. John, "Evoked-potential correlates of stimulus uncertainty," *Science*, vol. 150, no. 3700, pp. 1187–1188, 1965.

[19] L. 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] E. Donchin, K. M. Spencer, and R. Wijesinghe, "The mental prosthesis: assessing the speed of a p300-based brain-computer interface," *Rehabilitation Engineering, IEEE Transactions on*, vol. 8, no. 2, pp. 174–179, 2000.

[21] L. Bianchi, S. Sami, A. Hillebrand, I. P. Fawcett, L. R. Quitadamo, and S. Seri, "Which physiological components are more suitable for visual erp based brain–computer interface? a preliminary meg/eeg study," *Brain topography*, vol. 23, no. 2, pp. 180–185, 2010.

[22] V. Bostanov, "Bci competition 2003-data sets ib and iib: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1057–1061, 2004.

[23] H. Cecotti, B. Rivet, M. Congedo, C. Jutten, O. Bertrand, E. Maby, and J. Mattout, "A robust sensor-selection method for p300 brain–computer interfaces," *Journal of neural engineering*, vol. 8, no. 1, p. 016001, 2011.

[24] A. Combaz, N. V. Manyakov, N. Chumerin, J. A. Suykens, and M. Hulle, "Feature extraction and classification of eeg signals for rapid p300 mind spelling,"

in *Machine Learning and Applications, 2009. ICMLA'09. International Conference on*. IEEE, 2009, pp. 386–391.

[25] A. Combaz, N. Chumerin, N. V. Manyakov, A. Robben, J. A. Suykens, and M. M. Van Hulle, "Error-related potential recorded by eeg in the context of a p300 mind speller brain-computer interface," in *Machine Learning for Signal Processing (MLSP), 2010 IEEE International Workshop on*. IEEE, 2010, pp. 65–70.

[26] ——, "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.

[27] M. Kaper, P. Meinicke, U. Grossekathoefer, T. Lingner, and H. Ritter, "Bci competition 2003-data set iib: Support vector machines for the p300 speller paradigm," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1073–1076, 2004.

[28] P.-J. Kindermans, D. Verstraeten, and B. Schrauwen, "A bayesian model for exploiting application constraints to enable unsupervised training of a p300-based bci," *PloS one*, vol. 7, no. 4, p. e33758, 2012.

[29] P.-J. Kindermans, H. Verschore, D. Verstraeten, and B. Schrauwen, "A p300 bci for the masses: Prior information enables instant unsupervised spelling," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 719–727.

[30] D. J. Krusienski, E. W. Sellers, F. Cabestaing, S. Bayoudh, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "A comparison of classification techniques for the p300 speller," *Journal of neural engineering*, vol. 3, no. 4, p. 299, 2006.

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

[32] Y. Li, C. Guan, H. Li, and Z. Chin, "A self-training semi-supervised svm algorithm and its application in an eeg-based brain computer interface speller system," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1285–1294, 2008.

[33] D. J. McFarland, W. A. Sarnacki, and J. R. Wolpaw, "Should the parameters of a bci translation algorithm be continually adapted?" *Journal of neuroscience methods*, vol. 199, no. 1, pp. 103–107, 2011.

[34] R. C. Panicker, S. Puthusserypady, and Y. Sun, "Adaptation in p300 brain–computer interfaces: A two-classifier cotraining approach," *Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 12, pp. 2927–2935, 2010.

[35] A. Rakotomamonjy, V. Guigue, G. Mallet, and V. Alvarado, "Ensemble of svms for improving brain computer interface p300 speller performances," in *Artificial Neural Networks: Biological Inspirations–ICANN 2005*. Springer, 2005, pp. 45–50.

[36] A. Rakotomamonjy and V. Guigue, "Bci competition iii: dataset ii-ensemble of svms for bci p300 speller," *Biomedical Engineering, IEEE Transactions on*, vol. 55, no. 3, pp. 1147–1154, 2008.

[37] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert, "xdawn algorithm to enhance evoked potentials: application to brain–computer interface," *Biomedical Engineering, IEEE Transactions on*, vol. 56, no. 8, pp. 2035–2043, 2009.

[38] D. Ryan, G. Frye, G. Townsend, D. Berry, S. Mesa-G, N. Gates, and E. Sellers, "Predictive spelling with a p300-based brain–computer interface: increasing the rate of communication," *Intl. Journal of Human–Computer Interaction*, vol. 27, no. 1, pp. 69–84, 2010.

[39] H. Serby, E. Yom-Tov, and G. Inbar, "An improved P300-based brain-computer interface," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 13, no. 1, pp. 89–98, 2005.

[40] Y. Shahriari and A. Erfanian, "Improving the performance of p300-based brain-computer interface through subspace-based filtering," *Neurocomputing*, 2013.

[41] 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.

[42] M. Spüler, W. Rosenstiel, and M. Bogdan, "Online adaptation of a c-vep brain-computer interface (bci) based on error-related potentials and unsupervised learning," *PloS one*, vol. 7, no. 12, p. e51077, 2012.

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

[44] M. S. Treder and B. Blankertz, "Research (c) overt attention and visual speller design in an erp-based brain-computer interface," 2010.

[45] L. Acqualagnav, M. S. Treder, M. Schreuder, and B. Blankertz, "A novel brain-computer interface based on the rapid serial visual presentation paradigm," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*. IEEE, 2010, pp. 2686–2689.

[46] L. Acqualagna and B. Blankertz, "Gaze-independent bci-spelling using rapid serial visual presentation (rsvp)," *Clinical Neurophysiology*, 2013.

[47] U. Orhan, D. Erdogmus, B. Roark, B. Oken, and M. Fried-Oken, "Offline analysis of context contribution to erp-based typing bci performance," *Journal of neural engineering*, vol. 10, no. 6, p. 066003, 2013.

[48] U. Orhan, K. Hild, D. Erdogmus, B. Roark, B. Oken, and M. Fried-Oken, "Rsvp keyboard: An eeg based typing interface," in *Acoustics, Speech and*

Signal Processing (ICASSP), 2012 IEEE International Conference on. IEEE, 2012, pp. 645–648.

[49] M. Treder, N. Schmidt, and B. Blankertz, "Gaze-independent brain–computer interfaces based on covert attention and feature attention," *Journal of neural engineering*, vol. 8, no. 6, p. 066003, 2011.

[50] B. Roark, R. Beckley, C. Gibbons, and M. Fried-Oken, "Huffman scanning: using language models within fixed-grid keyboard emulation," *Computer Speech & Language*, 2012.

[51] M. S. Tredera, N. Schmidta, and B. Blankertza, "Gaze-independent visual brain-computer interfaces," in *Proc. of the TOBI (tools for brain–computer interaction) Workshop II*, 2010, pp. 33–34.

[52] F. Schettini, F. Aloise, P. Arico, S. Salinari, D. Mattia, and F. Cincotti, "Control or no-control? reducing the gap between brain-computer interface and classical input devices," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE.* IEEE, 2012, pp. 1815–1818.

[53] P. Aricò, F. Aloise, F. Schettini, A. Riccio, S. Salinari, F. Babiloni, D. Mattia, and F. Cincotti, "Geospell: an alternative p300-based speller interface towards no eye gaze required," in *Proc. of the TOBI (tools for brain–computer interaction) Workshop II*, 2010, pp. 158–159.

[54] Y. Liu, Z. Zhou, and D. Hu, "Gaze independent brain–computer speller with covert visual search tasks," *Clinical Neurophysiology*, vol. 122, no. 6, pp. 1127–1136, 2011.

[55] 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.

[56] J. D. Bayliss, S. A. Inverso, and A. Tentler, "Changing the p300 brain computer interface," *CyberPsychology & Behavior*, vol. 7, no. 6, pp. 694–704, 2004.

[57] S. Blain-Moraes, R. Schaff, K. L. Gruis, J. E. Huggins, and P. A. Wren, "Barriers to and mediators of brain–computer interface user acceptance: focus group findings," *Ergonomics*, vol. 55, no. 5, pp. 516–525, 2012.

[58] M. Marchetti, F. Piccione, S. Silvoni, and K. Priftis, "Exogenous and endogenous orienting of visuospatial attention in p300-guided brain computer interfaces: A pilot study on healthy participants," *Clinical Neurophysiology*, vol. 123, no. 4, pp. 774–779, 2012.

[59] M. Marchetti, F. Onorati, M. Matteucci, L. Mainardi, F. Piccione, S. Silvoni, and K. Priftis, "Improving the efficacy of erp-based bcis using different modalities of covert visuospatial attention and a genetic algorithm-based classifier," *PloS one*, vol. 8, no. 1, p. e53946, 2013.

[60] M. Marchetti, F. Piccione, S. Silvoni, L. Gamberini, and K. Priftis, "Covert visuospatial attention orienting in a brain-computer interface for amyotrophic lateral sclerosis patients," *Neurorehabilitation and neural repair*, vol. 27, no. 5, pp. 430–438, 2013.

[61] F. Piccione, F. Giorgi, P. Tonin, K. Priftis, S. Giove, S. Silvoni, G. Palmas, and F. Beverina, "P300-based brain computer interface: reliability and performance in healthy and paralysed participants," *Clinical neurophysiology*, vol. 117, no. 3, pp. 531–537, 2006.

[62] L. Mayaud, S. Filipe, L. Pétégnief, O. Rochecouste, and M. Congedo, "Robust brain-computer interface for virtual keyboard (robik): project results," *IRBM*, 2013.

[63] S. Halder, A. Furdea, B. Varkuti, R. Sitaram, W. Rosenstiel, N. Birbaumer, and A. Kübler, "Auditory standard oddball and visual p300 brain-computer interface performance," *Int J Bioelectromag*, vol. 13, pp. 5–6, 2011.

[64] S. Halder, E. M. Hammer, S. C. Kleih, M. Bogdan, W. Rosenstiel, N. Birbaumer, and A. Kübler, "Prediction of auditory and visual p300 brain-computer interface aptitude," *PloS one*, vol. 8, no. 2, p. e53513, 2013.

[65] A. Kübler, A. Furdea, S. Halder, E. M. Hammer, F. Nijboer, and B. Kotchoubey, "A brain–computer interface controlled auditory event-related potential (p300) spelling system for locked-in patients," *Annals of the New York Academy of Sciences*, vol. 1157, no. 1, pp. 90–100, 2009.

[66] I. Käthner, C. A. Ruf, E. Pasqualotto, C. Braun, N. Birbaumer, and S. Halder, "A portable auditory p300 brain–computer interface with directional cues," *Clinical neurophysiology*, 2012.

[67] J. Höhne, M. Schreuder, B. Blankertz, and M. Tangermann, "A novel 9-class auditory erp paradigm driving a predictive text entry system," *Frontiers in neuroscience*, vol. 5, 2011.

[68] D. Klobassa, T. Vaughan, P. Brunner, N. Schwartz, J. Wolpaw, C. Neuper, and E. Sellers, "Toward a high-throughput auditory p300-based brain–computer interface," *Clinical Neurophysiology*, vol. 120, no. 7, pp. 1252–1261, 2009.

[69] X. An, B. Wan, H. Qi, and D. Ming, "Digital spelling bci based on visual-auditory associate stimulation," in *Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS), 2012 IEEE International Conference on*. IEEE, 2012, pp. 82–85.

[70] A.-M. Brouwer and J. B. Van Erp, "A tactile p300 brain-computer interface," *Frontiers in neuroscience*, vol. 4, 2010.

[71] M. van der Waal, M. Severens, J. Geuze, and P. Desain, "Introducing the tactile speller: an erp-based brain–computer interface for communication," *Journal of Neural Engineering*, vol. 9, no. 4, p. 045002, 2012.

[72] D. J. McFarland, W. A. Sarnacki, and J. R. Wolpaw, "Electroencephalographic (eeg) control of three-dimensional movement," *Journal of Neural Engineering*, vol. 7, no. 3, p. 036007, 2010.

[73] A. B. Randolph, M. M. Jackson, and S. Karmakar, "Individual characteristics and their effect on predicting mu rhythm modulation," *Intl. Journal of Human–Computer Interaction*, vol. 27, no. 1, pp. 24–37, 2010.

[74] N. Hill and B. Schölkopf, "An online brain–computer interface based on shifting attention to concurrent streams of auditory stimuli," *Journal of neural engineering*, vol. 9, no. 2, p. 026011, 2012.

[75] A. Kübler, B. Kotchoubey, T. Hinterberger, N. Ghanayim, J. Perelmouter, M. Schauer, C. Fritsch, E. Taub, and N. Birbaumer, "The thought translation device: a neurophysiological approach to communication in total motor paralysis," *Experimental Brain Research*, vol. 124, no. 2, pp. 223–232, 1999.

[76] N. Neumann and N. Birbaumer, "Predictors of successful self control during brain-computer communication," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 74, no. 8, pp. 1117–1121, 2003.

[77] N. Neumann, A. Kübler, J. Kaiser, T. Hinterberger, and N. Birbaumer, "Conscious perception of brain states: mental strategies for brain–computer communication," *Neuropsychologia*, vol. 41, no. 8, pp. 1028–1036, 2003.

[78] C. Neuper, G. R. Müller-Putz, R. Scherer, and G. Pfurtscheller, "Motor imagery and eeg-based control of spelling devices and neuroprostheses," *Progress in brain research*, vol. 159, pp. 393–409, 2006.

[79] B. Blankertz, G. Dornhege, M. Krauledat, M. Schröder, J. Williamson, R. Murray-Smith, and K.-R. Müller, "The berlin brain-computer interface presents the novel mental typewriter hex-o-spell." 2006.

[80] 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.

[81] K.-R. Müller and B. Blankertz, "Toward non-invasive brain-computer interfaces," 2006.

[82] K.-R. Müller, M. Tangermann, G. Dornhege, M. Krauledat, G. Curio, and B. Blankertz, "Machine learning for real-time single-trial eeg-analysis: from brain–computer interfacing to mental state monitoring," *Journal of neuroscience methods*, vol. 167, no. 1, pp. 82–90, 2008.

[83] E. E. Sutter, "The visual evoked response as a communication channel," in *Proceedings of the IEEE Symposium on Biosensors*, 1984, pp. 95–100.

[84] ——, "The brain response interface: communication through visually-induced electrical brain responses," *Journal of Microcomputer Applications*, vol. 15, no. 1, pp. 31–45, 1992.

[85] H.-J. Hwang, J.-H. Lim, Y.-J. Jung, H. Choi, S. W. Lee, and C.-H. Im, "Development of an ssvep-based bci spelling system adopting a qwerty-style led keyboard," *Journal of neuroscience methods*, vol. 208, no. 1, pp. 59–65, 2012.

[86] M. Cheng, X. Gao, S. Gao, and D. Xu, "Design and implementation of a brain-computer interface with high transfer rates," *Biomedical Engineering, IEEE Transactions on*, vol. 49, no. 10, pp. 1181–1186, 2002.

[87] E. Yin, Z. Zhou, J. Jiang, F. Chen, Y. Liu, and D. Hu, "A novel hybrid bci speller based on the incorporation of ssvep into the p300 paradigm," *Journal of neural engineering*, vol. 10, no. 2, p. 026012, 2013.

[88] H. Cecotti, "A self-paced and calibration-less ssvep-based brain–computer interface speller," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 18, no. 2, pp. 127–133, 2010.

[89] B. Allison, T. Luth, D. Valbuena, A. Teymourian, I. Volosyak, and A. Graser, "Bci demographics: How many (and what kinds of) people can use an ssvep bci?" *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 18, no. 2, pp. 107–116, 2010.

[90] O. Friman, T. Luth, I. Volosyak, and A. Graser, "Spelling with steady-state visual evoked potentials," in *Neural Engineering, 2007. CNE'07. 3rd International IEEE/EMBS Conference on*. IEEE, 2007, pp. 354–357.

[91] I. Sugiarto, B. Allison, and A. Graser, "Optimization strategy for ssvep-based bci in spelling program application," in *Computer Engineering and Technology, 2009. ICCET'09. International Conference on*, vol. 1. IEEE, 2009, pp. 223–226.

[92] I. Volosyak, H. Cecotti, D. Valbuena, and A. Graser, "Evaluation of the bremen ssvep based bci in real world conditions," in *Rehabilitation Robotics, 2009. ICORR 2009. IEEE International Conference on*. IEEE, 2009, pp. 322–331.

[93] I. Volosyak, H. Cecotti, and A. Gräser, "Impact of frequency selection on lcd screens for ssvep based brain-computer interfaces," in *Bio-Inspired Systems: Computational and Ambient Intelligence*. Springer, 2009, pp. 706–713.

[94] I. Volosyak, "Ssvep-based bremen-bci–boosting information transfer rates," *J. Neural Eng.(in press, 2011)*.

[95] D.-W. Kim, H.-J. Hwang, J.-H. Lim, Y.-H. Lee, K.-Y. Jung, and C.-H. Im, "Classification of selective attention to auditory stimuli: toward vision-free brain–computer interfacing," *Journal of neuroscience methods*, vol. 197, no. 1, pp. 180–185, 2011.

[96] T. Kaufmann, C. Vögele, S. Sütterlin, S. Lukito, and A. Kübler, "Effects of resting heart rate variability on performance in the p300 brain-computer interface," *International Journal of Psychophysiology*, vol. 83, no. 3, pp. 336–341, 2012.

[97] J. Jin, E. W. Sellers, and X. Wang, "Targeting an efficient target-to-target interval for p300 speller brain–computer interfaces," *Medical & biological engineering & computing*, vol. 50, no. 3, pp. 289–296, 2012.

[98] N. Xu, X. Gao, B. Hong, X. Miao, S. Gao, and F. Yang, "Bci competition 2003-data set iib: enhancing p300 wave detection using ica-based subspace projections for bci applications," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1067–1072, 2004.

[99] B. Rivet and A. Souloumiac, "Optimal linear spatial filters for event-related potentials based on a spatio-temporal model: Asymptotical performance analysis," *Signal Processing*, 2012.

[100] A. Furdea, S. Halder, D. Krusienski, D. Bross, F. Nijboer, N. Birbaumer, and A. Kübler, "An auditory oddball (p300) spelling system for brain-computer interfaces," *Psychophysiology*, vol. 46, no. 3, pp. 617–625, 2009.

[101] J. del R Millan, J. Mouriño, M. Franzé, F. Cincotti, M. Varsta, J. Heikkonen, and F. Babiloni, "A local neural classifier for the recognition of eeg patterns associated to mental tasks," *Neural Networks, IEEE Transactions on*, vol. 13, no. 3, pp. 678–686, 2002.

[102] J. Millan, J. Mouriño *et al.*, "Asynchronous bci and local neural classifiers: an overview of the adaptive brain interface project," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 11, no. 2, pp. 159–161, 2003.

[103] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller, "Single-trial analysis and classification of erp components—a tutorial," *NeuroImage*, vol. 56, no. 2, pp. 814–825, 2011.

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

[105] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, B. Arnaldi *et al.*, "A review of classification algorithms for eeg-based brain–computer interfaces," *Journal of neural engineering*, vol. 4, 2007.

[106] N. V. Manyakov, N. Chumerin, A. Combaz, and M. M. Van Hulle, "Comparison of classification methods for p300 brain-computer interface on disabled subjects," *Computational intelligence and neuroscience*, vol. 2011, p. 2, 2011.

[107] J. Friedman, "Regularized discriminant analysis," *Journal of the American statistical association*, vol. 84, no. 405, pp. 165–175, 1989.

[108] R. Duda, P. Hart, and D. Stork, *Pattern classification*. Citeseer, 2001.

[109] B. Silverman, *Density estimation for statistics and data analysis*. Chapman & Hall/CRC, 1998.

[110] P. Yuan, X. Gao, B. Allison, Y. Wang, G. Bin, and S. Gao, "A study of the existing problems of estimating the information transfer rate in online brain–computer interfaces," *Journal of neural engineering*, vol. 10, no. 2, p. 026014, 2013.

[111] J. R. Wolpaw, H. Ramoser, D. J. McFarland, and G. Pfurtscheller, "Eeg-based communication: improved accuracy by response verification," *Rehabilitation Engineering, IEEE Transactions on*, vol. 6, no. 3, pp. 326–333, 1998.

[112] B. Z. Allison and C. Neuper, "Could anyone use a bci?" in *Brain-computer interfaces*. Springer, 2010, pp. 35–54.

[113] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[114] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[115] B. Dal Seno, M. Matteucci, and L. T. Mainardi, "The utility metric: a novel method to assess the overall performance of discrete brain–computer interfaces," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 18, no. 1, pp. 20–28, 2010.

[116] R. P. Stanley, *Enumerative combinatorics*. Cambridge university press, 2011, vol. 49.

[117] B. Roark, J. de Villiers, C. Gibbons, and M. Fried-Oken, "Scanning methods and language modeling for binary switch typing," in *Proceedings of the NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies*, 2010, pp. 28–36.

[118] K. Tobias, S. Völker, L. Gunesch, A. Kübler *et al.*, "Spelling is just a click away–a user-centered brain-computer interface including auto-calibration and predictive text entry," 2013.

[119] S. Lee and H.-S. Lim, "Predicting text entry for brain-computer interface," in *Future Information Technology*. Springer, 2011, pp. 309–312.

[120] E. Samizo, T. Yoshikawa, and T. Furuhashi, "A study on application of rbarq considering probability of occurrence and transition probability for p300 speller," in *Foundations of Augmented Cognition*. Springer, 2013, pp. 727–733.

[121] Ç. Ulaş and M. Çetin, "Incorporation of a language model into a brain-computer interface-based speller through hmms."

[122] I. Witten and T. Bell, "The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression," *Information Theory, IEEE Transactions on*, vol. 37, no. 4, pp. 1085–1094, 1991.

[123] H. V. Poor, "An introduction to signal detection and estimation," *New York, Springer-Verlag, 1988, 559 p.*, vol. 1, 1988.

[124] A. Jazwinski, *Stochastic processes and filtering theory*. Academic Pr, 1970, vol. 63.

[125] D. G. Pelli, "The videotoolbox software for visual psychophysics: Transforming numbers into movies," *Spatial vision*, vol. 10, no. 4, pp. 437–442, 1997.

[126] D. H. Brainard, "The psychophysics toolbox," *Spatial vision*, vol. 10, no. 4, pp. 433–436, 1997.

[127] M. Kleiner, D. Brainard, D. Pelli, A. Ingling, R. Murray, and C. Broussard, "What's new in psychtoolbox-3," *Perception*, vol. 36, no. 14, pp. 1–1, 2007.

[128] I. S. MacKenzie and R. W. Soukoreff, "Phrase sets for evaluating text entry techniques," in *CHI'03 extended abstracts on Human factors in computing systems*. ACM, 2003, pp. 754–755.

[129] G. Townsend, J. Shanahan, D. B. Ryan, and E. W. Sellers, "A general p300 brain-computer interface presentation paradigm based on performance guided constraints," *Neuroscience letters*, 2012.

[130] H. M. Ozaktas, M. A. Kutay, and Z. Zalevsky, *The fractional Fourier transform with applications in optics and signal processing*. Wiley New York, 2001.

[131] E. Wigner, "On the quantum correction for thermodynamic equilibrium," *Physical Review*, vol. 40, no. 5, p. 749, 1932.