J. Neural Eng. **10** (2013) 066003 (14pp) doi:10.1088/1741-2560/10/6/066003

Offline analysis of context contribution to ERP-based typing BCI performance

Umut Orhan[1], **Deniz Erdogmus**[1], **Brian Roark**[2,4], **Barry Oken**[2] **and Melanie Fried-Oken**[3]

[1] Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA
[2] Department of Neurology and Behavioural Neuroscience, Oregon Health and Science University, Portland, OR, USA
[3] Child Development & Rehabilitation Center, Oregon Health and Science University, Portland, OR, USA

E-mail: orhan@ece.neu.edu

Received 21 January 2013
Accepted for publication 2 September 2013
Published 8 October 2013
Online at stacks.iop.org/JNE/10/066003

Abstract

Objective. We aim to increase the symbol rate of electroencephalography (EEG) based brain–computer interface (BCI) typing systems by utilizing context information. *Approach.* Event related potentials (ERP) corresponding to a stimulus in EEG can be used to detect the intended target of a person for BCI. This paradigm is widely utilized to build letter-by-letter BCI typing systems. Nevertheless currently available BCI typing systems still require improvement due to low typing speeds. This is mainly due to the reliance on multiple repetitions before making a decision to achieve higher typing accuracy. Another possible approach to increase the speed of typing while not significantly reducing the accuracy of typing is to use additional context information. In this paper, we study the effect of using a language model (LM) as additional evidence for intent detection. Bayesian fusion of an *n*-gram symbol model with EEG features is proposed, and a specifically regularized discriminant analysis ERP discriminant is used to obtain EEG-based features. The target detection accuracies are rigorously evaluated for varying LM orders, as well as the number of ERP-inducing repetitions. *Main results.* The results demonstrate that the LMs contribute significantly to letter classification accuracy. For instance, we find that a single-trial ERP detection supported by a 4-gram LM may achieve the same performance as using 3-trial ERP classification for the non-initial letters of words. *Significance.* Overall, the fusion of evidence from EEG and LMs yields a significant opportunity to increase the symbol rate of a BCI typing system.

(Some figures may appear in colour only in the online journal)

1. Introduction

There are millions of people worldwide with severe motor and speech disabilities which prohibit them from participating in daily functional activities, such as personal care (Smith and Delargy 2005). 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. Various assistive technologies have been developed to increase the quality of life and functions for these individuals (Fager *et al* 2012). These technologies depend on the extraction and interpretation of various physiological signals at any anatomical site, such as eye movements blinks, or movements of the hand, foot or head. However there is a group of individuals with locked-in syndrome (LIS) who may not have sufficient neuromuscular control to reliably and consistently use switches or intentionally direct their eye gaze, putting them into a state referred to as total LIS. Bypassing all neuromuscular activity by relying on the use of brain activity as a switch activator has been developed as an interface for assistive technologies that allow communication and environmental control (Wolpaw and Wolpaw 2012, Sellers *et al* 2010).

A brain–computer interface (BCI) is a technology that uses neural signals as the physiological input for

[4] Present address: Google, Inc., Portland, OR, USA.

various assistive technology applications (Brunner *et al* 2011, Pfurtscheller *et al* 2000, 2010, Farwell and Donchin 1988, Renard *et al* 2010). BCI systems are based on invasive or noninvasive recording techniques. The noninvasive use of scalp electroencephalography (EEG) has drawn increasing attention due to its portability, feasibility and relative low cost. EEG has been used in BCIs for various communication and control purposes, such as typing systems or controlling robot arms. One of the biggest challenges encountered by most of these systems is to achieve sufficient accuracy or speed despite the existence of a low signal-to-noise ratio and the variability of background activity (Schalk 2008, Cincotti *et al* 2006). To ameliorate this problem in letter-by-letter BCI typing systems, researchers have turned to various hierarchical symbol trees (Wolpaw *et al* 2002, Serby *et al* 2005, Treder and Blankertz 2010). Additionally, there exist attempts to make stimuli more interesting to increase the attention level of the subjects (Kaufmann *et al* 2011). Even though using various approaches on presentation of the options may improve performance, most BCI researchers agree that BCI is a maturing field and there is still a need for improvement on typing speeds (Brunner *et al* 2011, Mak *et al* 2011, Wolpaw and Wolpaw 2012, Millán *et al* 2010). Low accuracy rates for symbol selection considerably reduce the practical usability of such systems. One way to overcome this condition is to increase the number of the repetitions of the stimuli to achieve a sufficient level of typing accuracy, by sacrificing the typing speed (Aloise *et al* 2012, Kaufmann *et al* 2011). Another approach is to incorporate the context information directly into the decision making process to obtain speed-ups and to improve the efficiency of such systems. In the case of letter-by-letter typing BCIs, placing a computational language model (LM), which can predict the next letter from the previous letters, within the decision making process can greatly affect performance by improving the accuracy and speed. If the symbol decisions are made using only the EEG evidence, that is, without using the context information in the language, decisions might not be sufficiently accurate with a low number of repetitions. Under such conditions further word prediction without improving decision accuracy might not be feasible (Ryan *et al* 2010). Thus, we propose to incorporate context information directly into the decision making process and to base symbol classification on the tight fusion of a LM and EEG evidence.

As an application of this proposed LM and EEG feature fusion methodology, we investigate the performance of context based decision making in RSVP Keyboard™, a BCI typing system that uses rapid serial visual presentation (RSVP) (Mathan *et al* 2008). RSVP is a paradigm that presents each stimulus, in this case letters and symbols, one at a time at a visually central location on the screen, instead of using the matrix layout of the popular P300-Speller (Krusienski *et al* 2008, Farwell and Donchin 1988) or the hexagonal two-level hierarchy of Berlin BCI (Treder and Blankertz 2010). In RSVP based BCI, the sequence of stimuli are presented at relatively high speeds, each subsequent stimulus replacing the previous one, while the subject tries to perform mental target matching between intended and presented stimuli. Accordingly, RSVP does not require an individual to perform extensive visual scanning or precise direction of gaze control. The RSVP Keyboard™ is particularly suited for the most severely restricted users including those with LIS.

LIS might be the result of traumatic brain injury, brain-stem stroke, spinal cord injury or a neurodegenerative condition such as amyotrophic lateral sclerosis. It is characterized by near total paralysis, despite the individual being cognitively intact. Motor control deterioration might extend to eye movements even though vision is intact. Even achieving a reliable twitch of a muscle or eye blink might become challenging. Therefore, using complex interactions requiring precise control of muscles or eyes might not be feasible for users with the most severe conditions. Consequently, simpler interactions via BCI hold great promise for effective text communication by such individuals. Yet, these simple interfaces have not been taking full advantage of LMs to ease or speed-up typing.

In this paper, we demonstrate a methodology to expand the EEG–LM fusion concept introduced earlier (Orhan *et al* 2011). We use a graphical model between EEG features and previously written text. An estimation of the conditional probability density functions (pdfs) of the EEG features, assuming they are continuous-valued, and a predictive n-gram LM are employed. This approach can be used with most letter-by-letter typing systems including the audio BCI systems and most EEG feature extraction methods, including linear or nonlinear projections, and classification algorithms with a probabilistic output. To test the proposed method, an offline analysis using EEG data collected with RSVP Keyboard™ can be obtained using this fusion framework. Based on experimental work, we designed the operational real time typing system, RSVP Keyboard™ (Hild *et al* 2011, Orhan *et al* 2012).

2. RSVP based BCI

RSVP is an experimental psycho-physics technique in which visual stimuli are displayed on a screen over time at a fixed focal area and in rapid succession (Mathan *et al* 2006, Huang *et al* 2009). In contrast, the Matrix-P300-Speller (Krusienski *et al* 2008, Farwell and Donchin 1988) used by the Wadsworth and Graz groups relies on a spatially distributed presentation of the symbols, highlighting them using various subset organizations (e.g. a column, a row, checkerboard or an individual symbol) and orders to elicit responses to detect the intent of the user. Berlin BCI's recent variation of their Hexo-Spell utilizes a two-layer tree structure (Treder and Blankertz 2010) where the subject chooses among six units (symbols or sets of these) while the subject focuses on a central focal area that uses an RSVP-like paradigm to generate responses induced by the intent. Recently, the researchers started to investigate and compare these alternative presentation methodologies (Treder *et al* 2011). In both of these BCIs an awareness of the full screen is required. On the other hand, the approach in RSVP Keyboard™ is to distribute the stimuli temporally and present one symbol at a time using RSVP. Correspondingly, a binary answer is sought for each

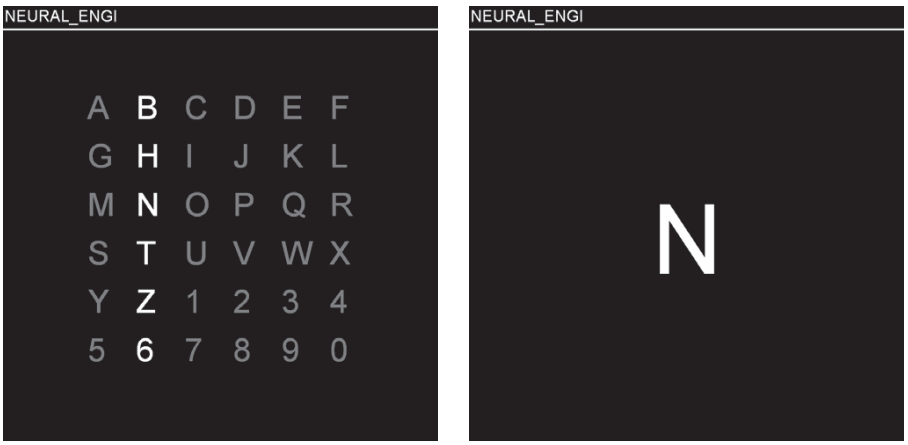


Figure 1. Sample visual presentation screens; matrix presentation (left) and RSVP (right).

presented symbol. The latter method has the advantage of not requiring the user to fixate at different regions of the screen. This can be particularly useful for users with weak or no gaze control, and for those whose cognitive skills do not allow processing of a matrix presentation (Brunner *et al* 2010). Two example screens from the matrix presentation and RSVP are given in figure 1.

The presentation paradigm of RSVP Keyboard™ includes stimulus sequences consisting of the 26 letters in the English alphabet, plus symbols for space and backspace. The set of symbols can easily be extended to a set containing various symbols for word completion or additional commands and punctuation. During the presentation of the sequence, the symbols are ordered randomly for the purpose of strengthening the response for the intended symbol. During the selection of a single symbol in the typing process, which we call an *epoch*, all symbols might be shown multiple times or in *sequences* to improve accuracy. We call the single presentation of a symbol a trial, which represents a single stimulus. The user is expected to react positively only for the target symbol. The problem reduces into decision making between a positive and negative intended target for each symbol.

The human brain generates a natural response, an event related potential (ERP), to the infrequent target stimulus shown in RSVP sequences. The most prominent component of the target ERPs is the P300 wave. It is a positive deflection in the scalp voltage mainly in the centro-parietal areas with a latency of roughly 300 ms. However, the regions, latency and amplitude of this signal may significantly vary between subjects. This target matching response allows us to build BCI systems by detecting ERPs.

The detection of single-trial ERPs solely from EEG is not accurate enough using current approaches. Therefore, it would be extremely beneficial to use additional context information during the sensing of ERPs. For a given probabilistic EEG feature extraction methodology for detecting ERPs, we propose the incorporation of a probabilistic LM, which is now explained.

3. 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 (Roark *et al* 2010). In the BCI field, there exist recent attempts to use word prediction to speed up the typing process (Ryan *et al* 2010). 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 LMs are incorporated into the decision making process, it might be possible to increase the speed and accuracy of the selection. To utilize this idea, the prefix string (what has already been typed) is used to predict the next symbol(s) to be typed, as the LM. Consequently, the next letters to be typed become highly predictable in certain contexts, particularly word-internally. 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 LM during the writing process.

The LM used in this paper is based on the n-gram sequence modeling paradigm, widely used in all of the application areas mentioned above. An L-gram model, it estimates the conditional probability of every letter in the alphabet given $L-1$ previous letters using a Markov model of order $L-1$. In this context, let $\mathsf{s}_t : \Omega \rightarrow \mathcal{S}$ be the random variable corresponding to the correct symbol for epoch t, i.e. during tth 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 $\mathsf{w}_j : \Omega \rightarrow \mathcal{A}$

where $j \in \{i_t - L + 2, \ldots, i_t - 1, i_t\}$ and $\mathcal{A} \subset \mathcal{S}$ is the set containing the symbols which are characters, e.g. letters, punctuation and space. For representational simplicity, let $\mathbf{w}_t = [\mathsf{w}_{i_t}, \mathsf{w}_{i_t-1}, \ldots, \mathsf{w}_{i_t-L+2}]$ correspond to the random string of last $L-1$ characters during the selection of the target of tth epoch and $\mathbf{w} = [w_1, w_2, \ldots, w_{L-1}]$ corresponds to a character string of length $L-1$. In n-gram models, the symbol prediction is made using the latest string of length L as

$$P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w}) = \frac{P(\mathsf{s}_t = s, \mathbf{w}_t = \mathbf{w})}{P(\mathbf{w}_t = \mathbf{w})}, \quad (1)$$

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

If the LM 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. $P(\mathsf{s}_t = s|\mathbf{w}_t = \mathbf{w}) = P(\mathsf{s}_t = s)$. The zero-gram model is defined as having no active LM or, equivalently, $P(\mathsf{s}_t = s)$ is assumed to be a uniform distribution over the alphabet, i.e. $P(\mathsf{s}_t = s) = 1/|\mathcal{S}|$. If the number of characters that are already typed is less than the LM order, a truncated model using all the characters that had been typed is used. As an example, if the second letter is being written, a 6-gram model is truncated to a 2-gram model. However this only happens at the very beginning of the typing process, and it does not restart at the beginning of words.

For the current study, all n-gram LMs were estimated from a one million sentence (210 million character) sample of the NY Times portion of the English Gigaword corpus. Corpus normalization and smoothing methods were as described in Roark *et al* (2010). Most importantly for this work, the corpus was case normalized, and we used Witten–Bell smoothing for regularization (Witten and Bell 1991). For the offline analysis conducted in this paper, we sampled contexts from a separate 1 million sentence subset of the same corpus. More specifically, for each letter, 1000 contexts were randomly sampled (without replacement).

4. Fusion of EEG features and the LM

The evidence obtained from EEG and the LM can be used collaboratively to make a more informative 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 $\mathsf{c} : \Omega \to \{0, 1\}$ be the random variable representing the class of intent, where 0 and 1 corresponds to negative and positive intents, respectively and $\mathbf{x} : \Omega \to \mathbb{R}^d$ 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 $d = 1$, there is only one EEG feature per trial. The fusion methodology explained here does not depend on the feature extraction method, and can practically be used with any feature vector in $\mathbb{R}^d$. The only requirement is an estimate of the conditional pdf 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 $r \in \{1, 2, \ldots, R\}$, 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,

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

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

$$Q = \frac{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \ldots, \mathbf{x}_{t,s,R} = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w}|\mathsf{c}_{t,s} = c)P(\mathsf{c}_{t,s} = c)}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \ldots, \mathbf{x}_R = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}. \quad (3)$$

We can assume that the EEG features corresponding to the symbol in question and the text that has been written are already 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 be affected by the text already written. This assumption can formally be written as $\mathbf{x}_{t,s,1}, \ldots, \mathbf{x}_{t,s,R} \perp\!\!\!\perp \mathbf{w}_t|\mathsf{c}_{t,s}$. Accordingly, (3) transforms to,

$$Q = \frac{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \ldots, \mathbf{x}_{t,s,R} = \mathbf{x}_R|\mathsf{c}_{t,s} = c)P(\mathbf{w}_t = \mathbf{w}|\mathsf{c}_{t,s} = c)P(\mathsf{c}_{t,s} = c)}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \ldots, \mathbf{x}_R = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}. \quad (4)$$

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 (i.i.d.) 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 pdf 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 (4) to,

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

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

$$Q = \frac{\left(\prod_{r=1}^{R} f(\mathbf{x}_{t,s,r} = \mathbf{x}_r|\mathsf{c}_{t,s} = c)\right) P(\mathsf{c}_{t,s} = c|\mathbf{w}_t = \mathbf{w})P(\mathbf{w}_t = \mathbf{w})}{f(\mathbf{x}_{t,s,1} = \mathbf{x}_1, \ldots, \mathbf{x}_{t,s,R} = \mathbf{x}_R, \mathbf{w}_t = \mathbf{w})}. \quad (5)$$

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 (2) 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 $\mathbf{X}_{t,s}=\{\mathbf{x}_{t,s,1},\mathbf{x}_{t,s,2},\ldots,\mathbf{x}_{t,s,R}\}$ and $\mathbf{X}=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_R\}$. Using the form we obtained after simplification and approximation from (2) to (5), the 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 LM, some of the probabilities can be rewritten as $P(\mathsf{c}_{t,s}=1|\mathbf{w}_t=\mathbf{w})=P(\mathsf{s}_t=s|\mathbf{w}_t=\mathbf{w})$ and $P(\mathsf{c}_{t,s}=0|\mathbf{w}_t=\mathbf{w}))=1-P(\mathsf{s}_t=s|\mathbf{w}_t=\mathbf{w})$. Finally, the ratio of the class posterior probabilities can be estimated as,

$$\Lambda(\mathsf{c}_{t,s}|\mathbf{X}_{t,s}=\mathbf{X},\mathbf{w}_t=\mathbf{w})=\frac{\left(\prod_{r=1}^{R}f(\mathbf{x}_{t,s,r}=\mathbf{x}_r|\mathsf{c}_{t,s}=1)\right)P(\mathsf{s}_t=s|\mathbf{w}_t=\mathbf{w})}{\left(\prod_{r=1}^{R}f(\mathbf{x}_{t,s,r}=\mathbf{x}_r|\mathsf{c}_{t,s}=0)\right)(1-P(\mathsf{s}_t=s|\mathbf{w}_t=\mathbf{w}))}. \quad (6)$$

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 LM. Therefore the decision on the class label of symbol s_t for epoch t, $\hat{c}_{t,s}$ may be done by comparing the likelihood ratio with a risk-dependent threshold, τ, i.e.

$$\Lambda(\mathsf{c}_{t,s}|\mathbf{X}_{t,s}=\mathbf{X},\mathbf{w}_t=\mathbf{w})\gtrless_{\hat{c}_{t,s}=0}^{\hat{c}_{t,s}=1}\tau, \quad (7)$$

or in other words,

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

5. EEG feature extraction case study using the RSVP Keyboard

There exist numerous ways to extract features from EEG. They might consist of bandpass filtering, artifact reduction, channel selection, linear or nonlinear projections, extraction of power features from various frequency bands, spatial pattern analysis, extraction of independent components, wavelet filtering etc. The method to extract features highly depends on the application.

In the RSVP Keyboard™, the goal for each stimulus is to decide if it corresponds to an intended symbol or not. Consequently, intent detection for a stimulus is equivalent to deciding if its ERP is induced from the target or non-target category. If we assume that ERPs last only for a limited duration, the problem reduces into a binary classification problem. This can easily be achieved by windowing the signals after the stimuli using a sufficiently long window. For the analysis in this paper, we apply a 500 ms window.

To extract well separated features, we use the following methodology. Firstly, the time windowed EEG signals are filtered by a 1.5–42 Hz bandpass filter (FIR, linear phase, length 153) to remove low frequency signal drifts and noise at higher frequencies for each channel. Secondly, temporal feature vectors containing the filtered and windowed signals from each channel are further projected to a slightly lower dimensional space by linear dimension reduction. For this paper, this is applied using principal component analysis by removing zero variance directions. Afterwards, the feature vectors corresponding to each feature channel are concatenated to create a single aggregated vector. This process amounts to a channel-specific energy preserving orthogonal projection of raw temporal features. Finally, using regularized discriminant analysis (RDA) (Friedman 1989), a nonlinear projection to $\mathbb{R}$ on the aggregate feature vector is applied. Doubtlessly, one can employ one of the numerous other feature extraction methodologies, which might perform better or worse. The main purpose of this paper is not to find the best feature extraction methodology, but to demonstrate the effectiveness of the use of context information.

5.1. Regularized discriminant analysis (RDA)

RDA is a modified quadratic discriminant analysis (QDA) model. If the feature vector corresponding to each class is assumed to have a 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. Under the Gaussianity assumption, QDA depends on class means, class covariance matrices, class priors and a risk based threshold. 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 performed in order to determine the signal statistics. When there exists 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, the discriminant is 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 determine the discrimination function of RDA from, where p is the number of features to administer RDA on and $v\in\{1,2,\ldots,N\}$ 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

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

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, $\forall k \in \{0, 1\}$

$$\hat{\mathbf{\Sigma}}_k(\lambda) = \frac{\mathbf{S}_k(\lambda)}{N_k(\lambda)} \tag{8}$$

where

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

and

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

with

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

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

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

where $\hat{\mathbf{\Sigma}}_k(\lambda)$ is given by (8), tr[·] 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.

The illustrate how effective these operations are at decreasing the singularities, we can investigate the ranks of the covariance matrices before and after. Before shrinkage $\mathrm{rank}[\hat{\mathbf{\Sigma}}_k] \leqslant N_k$, after shrinkage,

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

With the further application of the regularization ranks of the covariance estimates we have,

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

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

After carrying out classifier shrinkage and regularization on the estimated covariance matrices, the Bayes classifier (Duda *et al* 2001) is defined by comparison of the log-posterior-ratio with a risk-dependent threshold. The corresponding discriminant score function is given by,

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

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 paper, we will consider the discriminant score function as a nonlinear projection from $\mathbf{y}$ to x, i.e. $\mathsf{x} = \delta_{\mathrm{RDA}}(\mathbf{y})$. Subsequently, $x = \delta_{\mathrm{RDA}}(\mathbf{y})$ will be the one dimensional EEG feature for the fusion with LMs as explained in section 4.

As a final step, the conditional pdf of x given the class label, i.e. $f(\mathsf{x} = x|\mathsf{c} = k)$ needs to be estimated. It is done non-parametrically using kernel density estimation on the training data using a Gaussian kernel as

$$\hat{f}(\mathsf{x} = x|\mathsf{c} = k) = \frac{1}{N_k}\sum_{c(v)=k} K_{h_k}(x - x(v)), \tag{9}$$

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. For the Gaussian kernel, the bandwidth h_k is estimated using Silverman's rule of thumb (Silverman 1998) for each class k. This assumes the underlying density has the same average curvature as its variance-matching normal distribution. Using the conditional pdf estimates from (9) and LM probabilities from (1) in (6), we can finalize the fusion process.

6. Experimental analysis

6.1. Offline study

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 were 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 were each shown for 1 s, followed by three sequences of the randomly ordered 26 letters in the English alphabet with an 150 ms inter-stimuli interval. Correspondingly each sequence took 4.9 s including the 1 s fixation duration. Subjects were asked to look for the target letter shown at the beginning of the epoch.

The signals were recorded using a g.USBamp biosignal amplifier using active g.Butterfly electrodes from G.tec (Graz, Austria) at a 256 Hz sampling rate. The EEG electrodes were applied with a g.GAMMAcap (electrode cap) and the positions 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 were further filtered by the previously mentioned 1.5–42 Hz linear-phase bandpass filter (our design). The filtered signals were downsampled to 128 Hz. For each channel, stimulus-onset-locked time windows of [0,500) ms following each image onset were taken as the stimulus response.

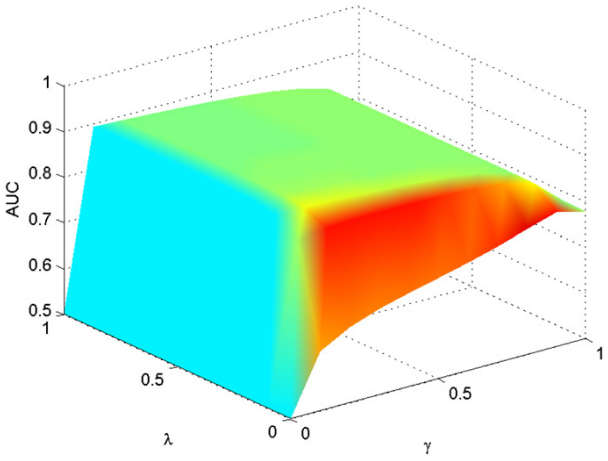


Figure 2. An example of the change in AUC while searching shrinkage (λ) and regularization (γ). The highest AUC is obtained for $\lambda = 0.6$ and $\gamma = 0.1$.

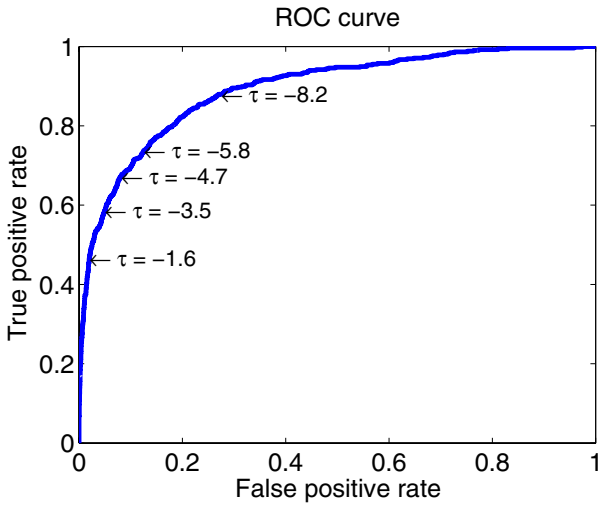


Figure 3. An example of an ROC curve corresponding to a single sequence and 0-gram LM.

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 ten 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 was trained on the set $\mathbb{E}\backslash\mathbb{E}_{k_j}$. During training, the classifier parameters λ and γ were determined using this ten-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 the classification scores for EEG classifiers were obtained using scores obtained from $\mathbb{E}\backslash\mathbb{E}_{k_j}$. The trained classifiers were applied to their respective test epochs to get the ten-fold cross-validation test results presented in the tables.

An example of the change in the area under the receiver operating characteristics (ROC) curves (AUC) during the grid search is given in figure 2 for a single sequence with no LM (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.

The LM was trained as described in section 3. For each letter in the alphabet, 1000 random samples were drawn from the same corpus (separate from the LM training data) for testing purposes. For each letter sample we simulate the fusion of EEG responses and the LM 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 LMs 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 are converted to EEG classifier scores; (iv) matching model probabilities for each letter are obtained from the LM; (v) and the fusion of ERP classifier scores and LM 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.

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. ROC curves were obtained using the decision rule given in (6) by changing the risk based threshold, τ. An example ROC curve is given in figure 3. The AUC were calculated for different orders of the LM, for different numbers of sequences used and for different positions of the sample target letter in the corresponding word from the corpus. In table 1, the AUC are compared. Each entry contains the pair of minimum and maximum AUC over the sessions, i.e. each pair represents the performance of the worst and best session. In tables 2–4 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 corresponds to a false alarm rate, a true positive rate and a τ triplet.

The letter decisions after an epoch may be made by selecting the symbol with the maximum likelihood ratio. This corresponds to selecting the symbol with the maximum $\Lambda(c_{t,s}|\mathbf{X}_{t,s} = \mathbf{X}, \mathbf{w}_t = \mathbf{w})$ from (6). If applied with our offline analysis structure, the correct letter selection probabilities averaged over subjects are given by figure 4. The plots show correct letter selection probabilities versus the inverse of the 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 the usual speed/accuracy trade-offs apply to the proposed typing system and better LMs result in better performance.

Table 1. The minimum and maximum values of the AUC obtained using the fusion classifier under different scenarios. The comparison is made using a different number of sequences for classification, different letter positions in the word and different LM orders.

		One sequence	Two sequences	Three 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 first letter	(0.892, 0.941)	(0.954, 0.983)	(0.977, 0.991)
	If not first letter	(0.975, 0.983)	(0.985, 0.992)	(0.991, 0.997)
8-gram	If first letter	(0.905, 0.945)	(0.960, 0.984)	(0.979, 0.992)
	If not first letter	(0.991, 0.993)	(0.995, 0.997)	(0.995, 0.998)

Table 2. The minimum and maximum values of the detection rates for the 1% false detection rate using the fusion classifier under different scenarios.

		One sequence	Two sequences	Three 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 first letter	(0.263, 0.416)	(0.434, 0.774)	(0.621, 0.810)
	If not first letter	(0.597, 0.684)	(0.748, 0.849)	(0.848, 0.927)
8-gram	If first letter	(0.294, 0.448)	(0.465, 0.782)	(0.647, 0.835)
	If not first letter	(0.810, 0.854)	(0.886, 0.932)	(0.936, 0.972)

Table 3. The minimum and maximum values of the detection rates for the 5% false detection rate using the fusion classifier under different scenarios.

		One sequence	Two sequences	Three 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 first letter	(0.606, 0.688)	(0.740, 0.884)	(0.886, 0.971)
	If not first letter	(0.842, 0.899)	(0.912, 0.966)	(0.960, 0.989)
8-gram	If first letter	(0.614, 0.716)	(0.766, 0.905)	(0.899, 0.971)
	If not first letter	(0.951, 0.971)	(0.972, 0.990)	(0.986, 0.996)

Table 4. The minimum and maximum values of the detection rates for the 10% false detection rate using the fusion classifier under different scenarios.

		One sequence	Two sequences	Three 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 first letter	(0.692, 0.836)	(0.857, 0.961)	(0.948, 0.990)
	If not first letter	(0.933, 0.961)	(0.966, 0.991)	(0.983, 0.996)
8-gram	If first letter	(0.729, 0.840)	(0.873, 0.964)	(0.950, 0.990)
	If not first letter	(0.983, 0.990)	(0.992, 0.997)	(0.995, 0.998)

Obtaining expected typing duration from the probability of correct decisions. 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 intended symbol, since it will not be able to correct its mistakes. Therefore $p < 0.5$ is considered to be failure as the subject might get stuck at a symbol. A similar relationship between expected time and probability of detection has also been obtained independently in Dal Seno *et al* (2010). An alternative derivation for the formula is given in the appendix.

Typing duration per symbol, estimated using this relation, for various numbers of sequences. LM orders are given in figure 5 after averaging over sessions and subjects.

6.2. The online system

The proposed fusion approach is implemented in the RSVP Keyboard™. Consequently, the real time system is trialled in typing by healthy subjects as well as people with LIS. Both groups were able to successfully operate the system for typing. To be able to demonstrate the applicability of this approach, we present results from Orhan *et al* (2012). In the online system, we employ a confidence based dynamic stopping criteria in

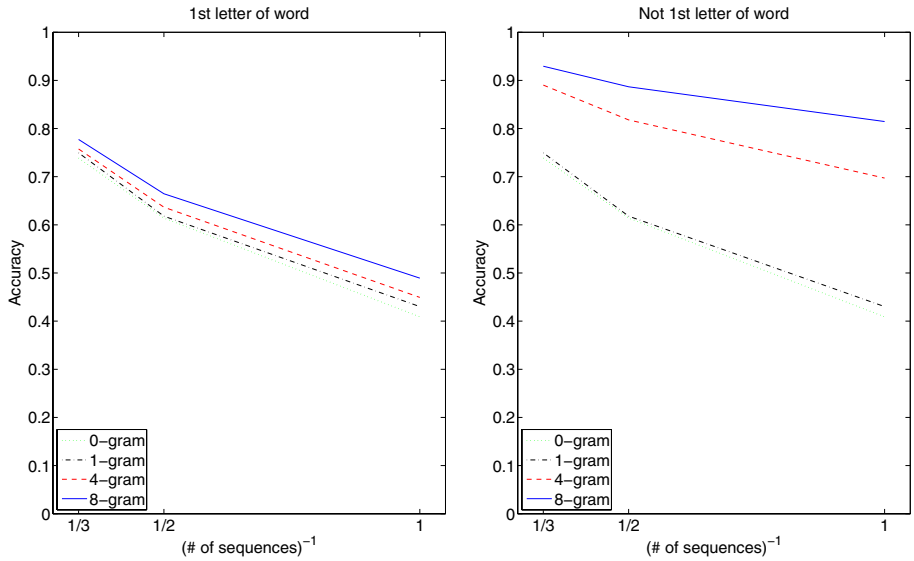


Figure 4. The average correct letter selection probability versus inverse of the number of repetitions for various LM orders and letter locations in the word.

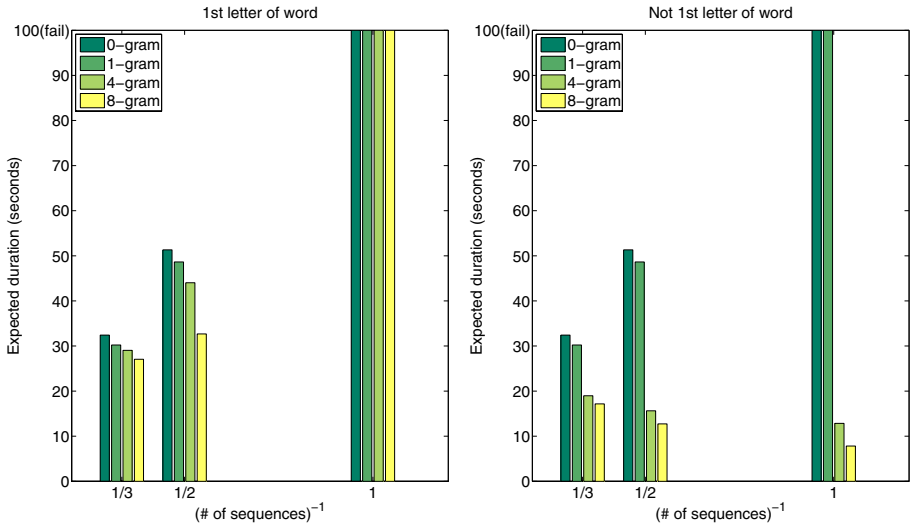


Figure 5. The average expected value of typing duration versus the inverse of the number of repetitions for various LM orders and letter locations in the word. In the graphs the 100 s mark is used jointly with the failure case. If the subject has a probability of getting stuck it is considered as a failure.

addition to the online setting proposed here, with a maximum of 6. The LM used in the online system was limited to a 6-gram model, due to the memory limitations of the 32-bit Windows operating system.

In the online setting, the symbol with the highest posterior probability is selected as the intended symbol. Consequently, no risk based threshold, τ, is selected. This means we employ a maximum *a posteriori* decision rule, which inherently assigns equal risk to every discrete decision candidate. These candidates are the elements of the alphabet. The ROC for the EEG discriminant is shown as an indicator for the quality of the EEG evidence, however, that ROC is not directly used in the decision process. Therefore the risk assignments associated with the τ that traces the ROC are not directly relevant to final typing performance. Hence, a value is not selected.

The experiments were performed on two healthy subjects and one subject with LIS. The typing performances corresponding to the healthy subjects are given in figures 6, 7. The locked-in subject successfully wrote HELLO_THERE_ PEOP and THIS_IS_FAMILY__. Unfortunately the sequence information for each decision was not recorded for the locked-in subject. As extracted from the triggers, overall symbol typing duration was 74.7 and 45 s/symbol, respectively.

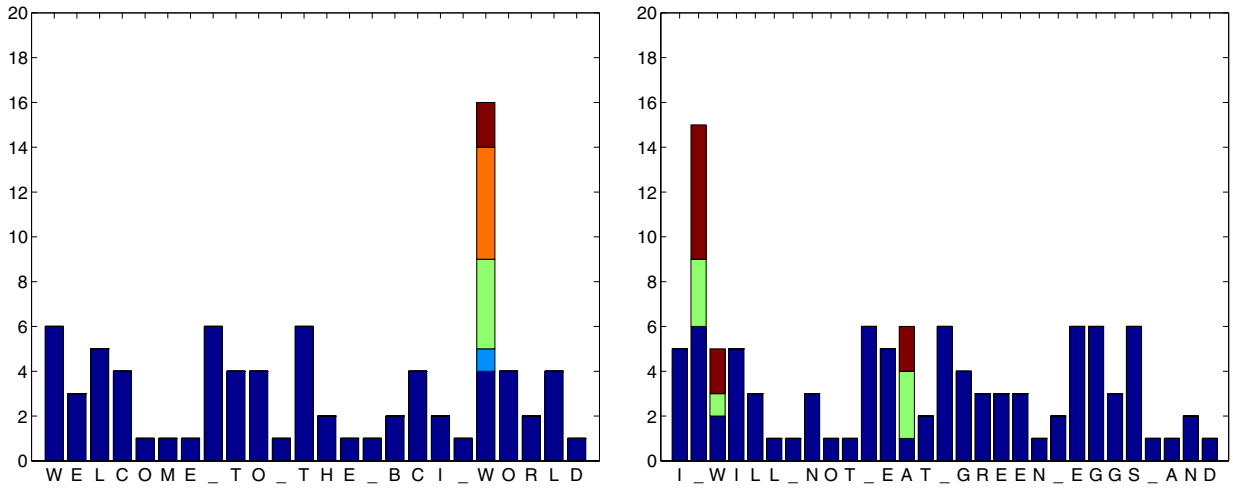


Figure 6. HS1 number of sequences used to type each symbol. If there are multiple epochs needed to correctly type a symbol, unintended decisions and backspaces to correct it are represented with different colors. Typing durations were 17.8 and 23.2 s/symbol, respectively.

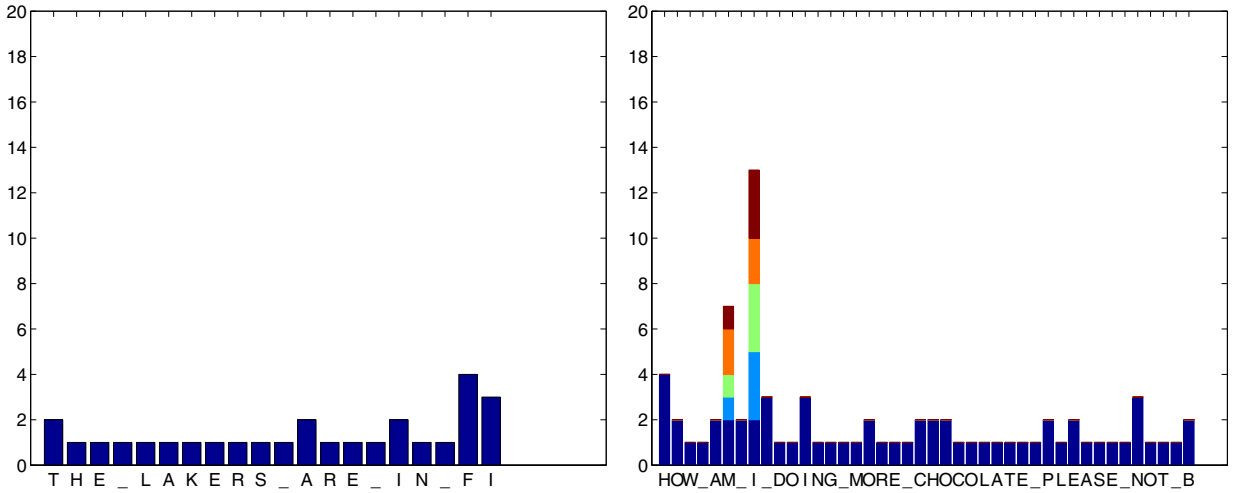


Figure 7. HS2 number of sequences used to type each symbol. If there are multiple epochs needed to correctly type a symbol, unintended decisions and backspaces to correct it are represented with different colors. Typing durations were 17.1 and 23.8 s/symbol, respectively.

7. Discussion

The analysis done in this paper supports the hypothesis that using context information to support ERP classification can improve the performance of BCI typing systems. As the number of stimulus repetitions and as the order for LMs increase, the accuracy of letter classification increases. A 0-gram LM (EEG only) performs the worst and context priors from the LM makes significant contribution in single-trial decision making. The LM contributes more to letters that are internal in words than those appear at the beginning. The LM is not as influential for the first letters whereas the number of repetitions is. When the context is from the beginning of a word, the LM has a high entropy, i.e. uncertainty on the symbols. Therefore it does not substantially improve the decision making process. However when the context is more internal to the word, the LM becomes more peaked for some symbols. Consequently, this phenomenon makes the LM more influential. Thus, the results suggest that the BCI system increases the number of trials for the beginning of each word and decreases for the subsequent letters. Reduction in the number of stimulus repetitions is a direct multiplier factor for reduction in time to type a given length text, hence significant speed-up in typing may be possible.

The context information fusion approach proposed in this paper can be integrated with almost any ERP-based BCI typing system in the field, e.g. P300-speller, by satisfying two main requirements. Firstly, there needs to exist a LM that makes a probabilistic prediction on the following selection based on the context. Secondly, a probabilistic conditional likelihood estimator for the trials is required. This can easily be achieved while using a probabilistic classifier. Alternatively, a parametric or non-parametric estimator, e.g. KDE, may be used to estimate conditional likelihoods. Meeting these conditions

allows a BCI typing system to use the proposed approach to make a more informed decision, consequently an increase in typing speed and accuracy is highly likely to be accomplished.

Acknowledgments

This work is supported by the NIH under grant 5R01DC009834. DE and UO were also partially supported by the NSF under grants IIS-0914808 and CNS-1136027. The opinions presented here are solely those of the authors and do not necessarily reflect the opinions of the funding agency.

Appendix. Accuracy and speed analysis

Catalan numbers: Catalan numbers arise in various counting problems (Stanley 2011). One of the problems that results in Catalan numbers is the nested parenthesis problem. The number of ways k left parenthesis and k right parenthesis can be ordered so that it becomes a properly nested ordering, correspond to the kth Catalan number. Explicitly, it is $C_k = \frac{1}{k+1}\binom{2k}{k}$. If we replace the left parenthesis with 0 and the right parenthesis 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 series if the following two conditions are satisfied,*

(i) $\forall r \in \{1, 2, \ldots, l-1\}$ $b_1 b_2 \cdots b_r$, *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 series is

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

Proof. Using binomial series expansion

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

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 $|4z| \leqslant 1$. Rearranging the equation,

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

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

$$\begin{aligned}\lceil (l+1)/2 \rceil &\leqslant s_l \leqslant s_{l-1} + 1 \leqslant \lfloor (l-1)/2 \rfloor + 1 \\ &= \lfloor (l+1)/2 \rfloor \leqslant \lceil (l+1)/2 \rceil.\end{aligned}$$

Consequently, $\leqslant s_l = s_{l-1} + 1$, 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 series of length $l = 2k+1$ contains k zeros and $k+1$ ones. Since each element of the series is obtained via independent Bernoulli trials, achieving a given successful series of length $2k+1$ has the probability of $p^{k+1}(1-p)^k$. The probability of achieving success with $2k+1$ trials is

$$P(\text{Success using } 2k+1 \text{ trials}) = p^{k+1}(1-p)^k C_k, \tag{A.3}$$

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 $\forall r \in \{1, 2, \ldots, 2k\}$, $b_1 b_2 \cdots b_r$ has no more 1s than 0s. Hence C_k is kth 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 (A.2) for $z = p(1-p)$,

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

□

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

Lemma 2. *If* $p \geqslant 0.5$, *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. Similarly to lemma 1, using the binomial series for (A.1) 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 $z = p(1-p)$,

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

Since from lemma 1, all of the probability mass is contained by successful series for $p > 0.5$, using (A.3) 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 (A.4),

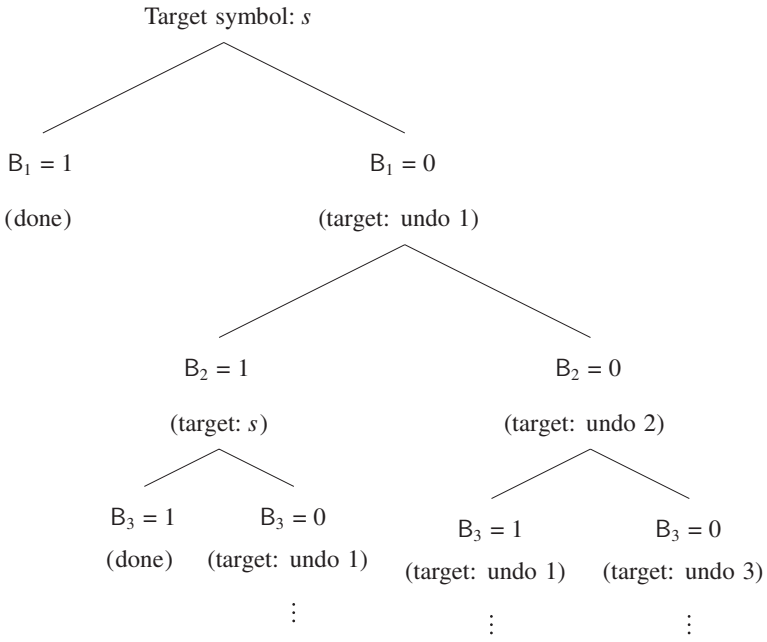
$$\begin{aligned}\mathbb{E}[\mathsf{L}] &= \frac{1}{2(1-p)}\left(-1+\frac{1}{2p-1}\right) \\ &= \frac{1}{2(1-p)}\cdot\frac{2(1-p)}{2p-1} = \frac{1}{2p-1}.\end{aligned}$$

□

A.1. BCI perspective

Assume that the correctness of a decision at the end of each epoch be i.i.d. Bernoulli random variables with a 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 series as defined in lemma 1. As an example, the successful conditions using seven epochs are,

0101011
0100111
0011011
0010111
0001111.

Let the duration of an epoch be T. If we assume that the success of the decisions are i.i.d., then the problem becomes equivalent to lemma 2. Correspondingly, we can calculate the expected symbol typing duration. If $p > 0.5$, the 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.

References

Aloise F, Schettini F, Aricò P, Salinari S, Babiloni F and Cincotti F 2012 A comparison of classification techniques for a gaze-independent p300-based brain–computer interface *J. Neural Eng.* **9** 045012

Brunner P, Bianchi L, Guger C, Cincotti F and Schalk G 2011 Current trends in hardware and software for brain–computer interfaces (BCIs) *J. Neural Eng.* **8** 025001

Brunner P, Joshi S, Briskin S, Wolpaw J, Bischof H and Schalk G 2010 Does the 'P300' speller depend on eye gaze? *J. Neural Eng.* **7** 056013

Cincotti F, Bianchi L, Birch G, Guger C, Mellinger J, Scherer R, Schmidt R, Suarez O and Schalk G 2006 BCI meeting 2005-workshop on technology: hardware and software *IEEE Trans. Neural Syst. Rehabil. Eng.* **14** 128–31

Dal Seno B, Matteucci M and Mainardi L T 2010 The utility metric: a novel method to assess the overall performance of discrete brain–computer interfaces *IEEE Trans. Neural Syst. Rehabil. Eng.* **18** 20–8

Duda R, Hart P and Stork D 2001 *Pattern Classification* (New York: Wiley)

Fager S, Beukelman D R, Fried-Oken M, Jakobs T and Baker J 2012 Access interface strategies *Assist. Technol.* **24** 25–33

Farwell L and Donchin E 1988 Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials *Electroencephalogr. Clin. Neurophysiol.* **70** 510–23

Friedman J 1989 Regularized discriminant analysis *J. Am. Stat. Assoc.* **84** 165–75

Hild K, Orhan U, Erdogmus D, Roark B, Oken B, Purwar S, Nezamfar H and Fried-Oken M 2011 An ERP-based brain-computer interface for text entry using rapid serial visual presentation and language modeling *Proc. 49th Annu. Meeting Association for Computational Linguistics: Human Language Technologies* (Stroudsburg, PA: Association for Computational Linguistics) pp 38–43

Huang Y, Pavel M, Hild K, Erdogmus D and Mathan S 2009 A hybrid generative/discriminative method for EEG evoked potential detection *NER'09: 4th Int. IEEE/EMBS Conf. on Neural Engineering* pp 283–6

Kaufmann T, Schulz S, Grünzinger C and Kübler A 2011 Flashing characters with famous faces improves ERP-based brain–computer interface performance *J. Neural Eng.* **8** 056016

Krusienski D, Sellers E, McFarland D, Vaughan T and Wolpaw J 2008 Toward enhanced P300 speller performance *J. Neurosci. Methods* **167** 15–21

Mak J, Arbel Y, Minett J, McCane L, Yuksel B, Ryan D, Thompson D, Bianchi L and Erdogmus D 2011 Optimizing the p300-based brain–computer interface: current status, limitations and future directions *J. Neural Eng.* **8** 025003

Mathan S, Erdogmus D, Huang Y, Pavel M, Ververs P, Carciofini J, Dorneich M and Whitlow S 2008 Rapid image analysis using neural signals *CHI'08 Extended Abstracts on Human Factors in Computing Systems* (New York: ACM) pp 3309–14

Mathan S, Erdogmus D, Huang Y, Pavel M, Ververs P, Carciofini J, Dorneich M and Whitlow S 2006 Neurotechnology for image analysis: searching for needles in haystacks efficiently *Foundations of Augmented Cognition* ed D Schmorrow *et al* (Arlington, VA: Strategic Analysis) pp 3–11

Millán J *et al* 2010 Combining brain–computer interfaces and assistive technologies: state-of-the-art and challenges *Front. Neurosci.* **4** 161

Orhan U, Erdogmus D, Roark B, Purwar S, Hild K, Oken B, Nezamfar H and Fried-Oken M 2011 Fusion with language models improves spelling accuracy for ERP-based brain computer interface spellers *IEEE Conf. Proc. Engineering in Medicine and Biology Society* pp 5774–7

Orhan U, Hild K, Erdogmus D, Roark B, Oken B and Fried-Oken M 2012 RSVP keyboard: an EEG based typing interface *ICASSP: IEEE Int. Conf. on Acoustics, Speech and Signal Processing* pp 645–8

Pfurtscheller G, Allison B, Brunner C, Bauernfeind G, Solis-Escalante T, Scherer R, Zander T, Mueller-Putz G, Neuper C and Birbaumer N 2010 The hybrid BCI *Front. Neurosci.* **4** 30

Pfurtscheller G, Neuper C, Guger C, Harkam W, Ramoser H, Schlogl A, Obermaier B and Pregenzer M 2000 Current trends in Graz brain-computer interface (BCI) research *IEEE Trans. Rehabil. Eng.* **8** 216–9

Renard Y, Lotte F, Gibert G, Congedo M, Maby E, Delannoy V, Bertrand O and Lécuyer A 2010 OpenViBE: an open-source software platform to design, test and use brain-computer interfaces in real and virtual environments *Presence* **19** 35–53

Roark B, de Villiers J, Gibbons C and Fried-Oken M 2010 Scanning methods and language modeling for binary switch typing *Proc. NAACL HLT '10 Workshop on Speech and Language Processing for Assistive Technologies* (Stroudsburg, PA: Association for Computational Linguistics) pp 28–36

Ryan D, Frye G, Townsend G, Berry D, Mesa-G S, Gates N and Sellers E 2010 Predictive spelling with a p300-based brain–computer interface: increasing the rate of communication *Int. J. Hum.-Comput. Interact.* **27** 69–84

Schalk G 2008 Brain–computer symbiosis *J. Neural Eng.* **5** P1

Sellers E, Vaughan T and Wolpaw J 2010 A brain–computer interface for long-term independent home use *Amyotrophic Lateral Scler.* **11** 449–55

Serby H, Yom-Tov E and Inbar G 2005 An improved P300-based brain-computer interface *IEEE Trans. Neural Syst. Rehabil. Eng.* **13** 89–98

Silverman B 1998 *Density Estimation for Statistics and Data Analysis* (London: Chapman and Hall)

Smith E and Delargy M 2005 Locked-in syndrome *BMJ* **330** 406

Stanley R P 2011 *Enumerative Combinatorics* vol 49 (Cambridge: Cambridge University Press)

Treder M and Blankertz B 2010 (C) overt attention and visual speller design in an ERP-based brain–computer interface *Behav. Brain Funct.* **6** 28

Treder M, Schmidt N and Blankertz B 2011 Gaze-independent brain–computer interfaces based on covert attention and feature attention *J. Neural Eng.* **8** 066003

Witten I and Bell T 1991 The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression *IEEE Trans. Inform. Theory* **37** 1085–94

Wolpaw J, Birbaumer N, McFarland D, Pfurtscheller G and Vaughan T 2002 Brain–computer interfaces for communication and control *Clin. Neurophysiol.* **113** 767–91

Wolpaw J and Wolpaw E 2012 *Brain–Computer Interfaces: Principles and Practice* (Oxford: Oxford University Press)