An Active Recursive State Estimation Framework for Brain-Interfaced Typing Systems

*Aziz Koçanaoğulları
Northeastern University, MA, USA
akocanaogullari@ece.neu.edu

* Yeganeh M. Marghi
Northeastern University, MA, USA
ymmarghi@ece.neu.edu

Murat Akçakaya
University of Pittsburgh, PA, USA
akcakaya@pitt.edu

Deniz Erdoğmuş
Northeastern University, MA, USA
erdogmus@ece.neu.edu

Abstract

Typing systems driven by noninvasive electroencephalogram (EEG)-based brain-computer interfaces (BCIs) can help people with severe communication disorders (including locked-in state) communicate. These systems mainly suffer from lack of sufficient accuracy and speed due to inefficient querying to surpass a hard pre-defined threshold. We introduce a novel recursive state estimation framework for BCI-based typing systems using active querying through sequences of stimuli and stopping criterion. Previously, we proposed a history-based objective called *Momentum* which is a function of posterior changes across sequences. In this paper, we first extend the definition of the Momentum objective and we propose a unified framework that employs this extended Momentum objective both for querying and stopping. We show that this framework leads to significant improvement in the accuracy/speed ratio in human intent estimation for BCI-based typing systems. To provide a practical example, we employ a language-model-assisted EEG-based BCI typing system called RSVP Keyboard. Our results show the proposed framework on average improves the information transfer rate (ITR) and accuracy at least 52% and 8.7%, respectively, when compared to alternative approaches, such as random or mutual information methods.

Keywords: Recursive state estimation, Active querying, Stopping criterion, BCI typing interface, RSVP Keyboard.

1 Introduction

Noninvasive electroencephalography (EEG)-based brain-computer interface (BCI) typing systems are designed for people with limited speech abilities to establish an alternative communication tool. These systems require presentation (mostly in visual forms) of sequences of stimuli (e.g. a set of symbols from the English alphabet) to induce changes in the EEG of the user to infer the intended symbols [1, 2]. Due to the low signal-to-noise ratio characteristics of the recorded EEG in response to the presented stimuli, generally in order to improve the accuracy of the user intent inference, these systems require repetition of sequences of symbols until a certain confidence level is reached. Moreover, language models (LM) are used to further improve the typing performance of such systems through the fusion of EEG and LM evidences to make the final decision on the user intent [1, 2].

Because typing process needs repetition of multiple sequences in EEG-based BCI typing systems, the user intent inference can be formulated as a recursive state estimation (RSE) problem assuming

*Joint first authors.

that the unknown state represents the intended symbol by the user [3, 4]. The formulation is recursive because, these systems try to make an inference after each sequence. One approach to collect evidence for each correspondent letter, is presenting the entire alphabet in the stimuli. However, by nature, languages are structured and not every possibility is equally likely and hence repeating the entire alphabet is not efficient. Therefore, a subset query selection is usually preferred to make these systems more practical and real-world-worthy. In this scenario, the system is tasked to decide on a subset from the alphabet to update its belief over state estimates.

Query optimization for BCI typing systems is not a well-studied problem. To the best our knowledge, there is a limited number of studies that addressed the query optimization problem for the BCI typing system designs. Omar [5] proposed a posterior matching scheme for a typing task. Higger [3] used maximum mutual information (MMI) coding for query selection to maximize the information transfer rate (ITR) in the typing task. In the last study done by Moghadamfalahi [4], authors used expected posterior maximization (EPM) for query selection for a BCI typing system. However, all of these query selection methods result in the selection of the *N-best* stimuli based on the posterior distribution [6, 7]. In this scenario, with respect to the current posterior distribution, letters are selected with descending order of associated probability mass function. Choosing the *N-best* queries based on the current belief (prior information), however, does not always provide the best performance in RSE problems. Because, current belief may not be always trusted and may include misleading information. In the case of the BCI typing systems, for instance, the current belief may be negatively influenced by the prior information provided by a language model. The language model provides probability values over the alphabet that is statistically learned from a dataset. As a matter of fact, word choices are topic dependent and it is not possible for the statistical model to capture each possibility. This yields some word choices to be statistically uncommon for the language model. If the user intent (target state) is an uncommon phrase (e.g. an English word starting with letter X), the prior behaves in an adversarial manner, causing a longer estimation session, or may lead to a wrong state estimation due to limitations of EEG evidence such as noise and limited number of typing sequences. Therefore, such BCI systems also require exploration beyond the current belief (posterior probability over letters).

In addition, due to noisy measurements in BCI applications, e.g. EEG-based BCI, making confident decisions about the user intent becomes a challenge. Accordingly, BCI typing systems typically require excessive querying process to reach a pre-defined threshold to provide a minimum confidence for every chosen letter. In RSE problems, to compensate for the detrimental effect of noise in measurements, the confidence threshold is set to a high value to decrease ambiguity in the state estimation. However, incremental gains over the posterior probability of a particular estimate come at the disproportionately high expense of time and budget. For instance, in BCI typing systems, in order to increase accuracy, the system repeats the presentation of stimuli multiple times, which translates to reduced typing speed and user frustration. Moreover, the inclusion of *backspace* command in such interfaces for error correction can exacerbate these issues. Therefore, when facing constraints for the inference problem, a trade-off between accuracy and speed must be considered.

To address both challenges for query optimization and stopping of the decision process, in this paper, we propose a new recursive state estimation framework that utilizes an evidence history-based objective called *Momentum*. Previously, we have shown that including Momentum of states in the stopping criterion of the estimation process provides a better accuracy/speed ratio in RSE problems [8]. Moreover, in [9], we have shown that Momentum of queries can provide a trade-off between exploration and exploitation during query selection by increasing the chance for the unlikely queries to be selected compared to *N-best* approaches. However, since our previous approach only considered non-zero Momentum for already queried stimuli. The selection of non-queried but potentially intended state was significantly delayed. This can introduce a significant problem for a typing interface that uses a language model. In this manuscript, we extend beyond our existing approaches, redefine the Momentum to consider all the possible stimuli that can be used for querying and propose a unified state estimation framework that utilizes the new Momentum objective both for query selection and as stopping criterion simultaneously during recursive user intent inference. To examine the performance of the proposed framework, we use an EEG-based BCI typing system called RSVP Keyboard. We present typing performance results for both simulation and actual human-in-the-loop typing schemes.

2 Method

In BCI typing systems, the user has an intended phrase in mind. Estimating the intended symbol is the smallest decision block in a BCI typing system. For the sake of simplicity in the application, in such typing systems, users are instructed to type one symbol at a time. We refer to the user-intended symbol (target which is the unknown state in the recursive state estimation) as σ which is an element of a finite set of alphabet denoted by $\mathcal{A}$. The BCI system proceeds with the estimation through a sequential decision making process containing *sequences* indexed by s of multiple trials indexed by i. In BCI designs, a trial is a finite-length window of EEG signal time-locked to the onset of a trial, which is a visual stimulus (query). In the typing system, the system decides on a subset of queries (symbols), $\Phi_s \triangleq \{\phi_s^1, \phi_s^2, ..., \phi_s^K\}$ where any $\phi \in \mathcal{A}$ to present to the users. Here, K is the number of trials in a sequence. After Φ_s is selected, the BCI system observes feedback from the user e.g., recorded EEG, $\boldsymbol{\varepsilon}_s \triangleq \{\varepsilon_s^1, \varepsilon_s^2, ..., \varepsilon_s^K\}$ before moving on to the next sequence. Table 1 summarizes the frequently used notation in this study.

Estimation of σ requires repetition of multiple sequences, the estimation process can be well formulated in the recursive state estimation (RSE) framework as follows.

2.1 Mathematical Formulation of RSE Problem

Figure 1 shows the probabilistic graphical model proposed for an RSE problem for a BCI typing system. Here, δ_s denotes the system's state including the probability of all states at s-th sequence. It should be noted that δ_0 includes the prior information about the decision. δ_S is the BCI system decision at the last sequence S. It is assumed that the system state represents the target state which belongs to a finite discrete space. Although, the observation space is bounded and continuous. As shown in Figure 1, the dynamics of the system state δ_s follows a *Markov decision process* (MDP), which is similar to a Markov chain, except that the transition matrix depends on the actions taken by the system at each sequence.

We can decompose the RSE problem into inference I and querying Q objectives. Using Bayesian framework, the inference objective is function of the posterior probability, $p(\sigma|\boldsymbol{\varepsilon}, \Phi, \mathcal{H}_s)$, where $\mathcal{H}_s$ is the history of previous EEG evidence and new (not observed) evidence $\boldsymbol{\varepsilon}$. At query optimization step, the system queries the environment until the inference constraints are satisfied. To extract information from the observation, the system needs to explore various sequences. Accordingly, the aim of the system is to query a subset of queries to estimate the state, which is the target letter. Therefore, based on the collected evidence if a decision is not possible, the system selects a subset of queries for the upcoming sequence to improve its confidence. The querying process is being continued until the speller achieves a minimum confidence level required to type a symbol, i.e. τ. This process is formulated by query optimization, which is defined as expected value of posterior with respect to the prior distribution or *expected posterior maximization* (EPM) as follows.

$$
\begin{aligned}
(I): \quad & \hat{\sigma} = \arg\max_{\sigma \in \mathcal{A}} p(\sigma|\mathcal{H}_s) \\
& \text{s.t. } p(\sigma|\mathcal{H}_s) \geq \tau \\
(Q): \quad & \Phi_{s+1} = \arg\max_{\Phi} \mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon}, \Phi, \mathcal{H}_s) \right] \right]
\end{aligned}
\tag{1}
$$

In this equation the constraint in (I) plays the role of stopping criterion. At a sequence s it is possible there exists no σ that satisfies the constraint and hence the system continues with recursions until such candidate σ emerges. Since $\log(.)$ is a monotonically increasing function, the querying problem in (1), can be written as:

$$
\Phi_{s+1} = \arg\max_{\Phi} \log \left(\mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon}, \Phi, \mathcal{H}_s) \right] \right] \right) \tag{2}
$$

Using Jensen's inequality, the objective in (2) is bounded from below by,

$$
\log \left(\mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon}, \Phi, \mathcal{H}_s) \right] \right] \right) \geq \mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\log \left(\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon}, \Phi, \mathcal{H}_s) \right] \right) \right] \tag{3}
$$

Instead of maximizing the original Q objective in (1), we can optimize the lower bound in (3).

$$\Phi_{s+1} = \arg\max_{\Phi} \mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\log \left(\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon},\Phi,\mathcal{H}_s) \right] \right) \right] \tag{4}$$

To simplify the querying objective in (4), Moghadamfalahi in work [4] approximated $\mathbf{E}_{p(\boldsymbol{\varepsilon}|\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon},\Phi,\mathcal{H}_s) \right]$ with a sub-optimal linear approximation around the mean, by assuming concentrated-unimodal distribution condition over the $p(\boldsymbol{\varepsilon}|\Phi)$, which are not necessarily valid assumptions in many applications, such as EEG-based BCI systems with noisy measurement.

Instead of solving (4), we can consider another lower bound for f_Q in (1), which is:

$$\mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\log \left(\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[p(\sigma|\boldsymbol{\varepsilon},\Phi,\mathcal{H}_s) \right] \right) \right] \geq \mathbf{E}_{p(\sigma|\mathcal{H}_s)} \left[\mathbf{E}_{p(\boldsymbol{\varepsilon}|\sigma,\Phi)} \left[\log \left(p(\sigma|\boldsymbol{\varepsilon},\Phi,\mathcal{H}_s) \right) \right] \right] \tag{5}$$

Minimizing the right term in (5) is identical to minimizing the conditional entropy $H(\sigma|\boldsymbol{\varepsilon},\Phi,\mathcal{H}_s)$ or maximizing mutual information $I(\sigma,\boldsymbol{\varepsilon}|\Phi,\mathcal{H}_s)$. Accordingly, query selection using mutual information can be written as the following objective:

$$\begin{aligned} \Phi_{s+1} &= \arg\max_{\Phi} I(\sigma,\boldsymbol{\varepsilon}|\Phi,\mathcal{H}_s) \\ &= \arg\max_{\Phi} \; -H(\sigma|\boldsymbol{\varepsilon},\Phi,\mathcal{H}_s) \end{aligned} \tag{6}$$

Before introducing the proposed framework, here, we present the following assumptions for the proposed framework.

- Number of trials in a sequence is capped at $K \in \mathbb{N}$, and the number of sequences is defined as $\{j \in \mathbb{N} \mid 0 < j \leq S\}$, which implies that the stop action must be selected (a decision needs to be made) in at most S sequences.
- After each sequence, EEG observations are only function of the user intent and the query subset presented in that sequence. The observations are independent of the task history $\mathcal{H}_{s-1}$, such that,
$$p(\boldsymbol{\varepsilon}_s|\sigma,\Phi_s,\mathcal{H}_{s-1}) = p(\boldsymbol{\varepsilon}_s|\sigma,\Phi_s)$$
$$p(\boldsymbol{\varepsilon}_s|\Phi_s,\mathcal{H}_{s-1}) = p(\boldsymbol{\varepsilon}_s|\Phi_s)$$
- There is a prior information $p(\sigma|\mathcal{H}_0)$ provided by the LM, conditioned on previously typed symbols (if any).
- **Independent evidence trials:** The BCI system proceeds with the estimation through a sequential decision making process containing sequences indexed by s of multiple trials indexed by i. Observation corresponding to different trials in a sequence are independent conditioned on the unknown state σ. Therefore the posterior probability at time s is written as:
$$p(\sigma|\boldsymbol{\varepsilon}_s,\Phi_s,\mathcal{H}_{s-1}) \quad = p(\sigma|\mathcal{H}_0) \prod_{j=1}^{s} \prod_{i=1}^{t_i} \frac{p(\varepsilon_j^i|\sigma,\phi_j^i)}{p(\varepsilon_j^i|\phi_j^i)}$$
- **Unimodality in evidence distributions:** It is assumed that all observations originate from two unknown unimodal probability distributions conditioned on state and query tuples: (i) conditioned on target class (intended state), f_{σ^*,ϕ_s^i}, and (ii) conditioned on non-target class, $f_{\not\sigma^*,\phi_s^i}$.
- Each query subset includes a unique set of stimuli, i.e., $\forall i \neq r \in \{1,2,\cdots,K\}, \;\; \phi_j^i \neq \phi_j^r$.
- **Estimability:** We assume that there exists at least one query for any two states that enables unique identification of the unknown state such that
$$\forall a \neq b \in \mathcal{A} \; \exists \; \phi \text{ s.t. } f_{a,\phi,}(\boldsymbol{\varepsilon}) \neq f_{b,\phi,}(\boldsymbol{\varepsilon}) \; \forall \boldsymbol{\varepsilon}$$
- **Asymptotic convergence:** Given the *estimability* condition, every state can be estimated by sufficient number of queries. Additionally the set of queries $\phi_{1:j}$ is not unique,
$$\forall \sigma_i \; \exists \; \Phi_{1:j} \text{ s.t. } p(\sigma|\boldsymbol{\varepsilon}_{1:j},\Phi_{1:j}) \in \{0,1\}$$

2.2 Active Querying

In state estimation, mutual information is a measure of expected information gain with the current belief (posterior) of an estimate. It can be observed that the posterior for the state is expected to increase with further evidence observation, based on the *estimability* and *asymptotic convergence* assumptions. Since mutual information is a submodular set function and observations within a sequence are assumed to be independent, i.e. *independent evidence trials* assumption, the sequential subset query selection can be achieved via a greedy approach by optimizing (6) for each query with theoretical guarantees [10, 11]. Therefore, query selection objective can be reformulated for each query as:

$$\phi^i_{s+1} = \arg\max_{\phi\in\mathcal{A}} I(\sigma, \varepsilon|\phi, \mathcal{H}_s) \tag{7}$$

It has been discussed in [9] that using greedy selections such as MMI or minimizing entropy may be misleading especially if the intended symbol is highly unlikely for the LM (adversarial case). Authors in [9] introduced a history-based objective called *Momentum*. The objective is a function of the average posterior changes across sequences, i.e. $m(\phi|\mathcal{H}_j)$, which is the summation of probability displacement multiplied by the probability mass. This objective is used as a measure of speed towards a specific estimate. Therefore, in a scenario where the true state has small initial probability (e.g. due to the user aiming for a very unlikely letter/word/phrase), incorporating speed allows up-and-coming-candidates, whose probability is steadily increasing sequence after sequence. This includes the possibility that the correct state (letter) gets queried earlier in the process while its absolute probability is relatively low in the alphabet and additionally its probability increments have been in the positive direction. In the previous work [9], Momentum function was only computed for previously queried states and its value has been set to zero if the state was not queried. In this work this is extended to all possible states. To compute the posterior changes for all states, here the Momentum objective is defined as:

$$M(\phi|\mathcal{H}_s) = \frac{1}{s}\sum_{j=1}^{s} m(\phi|\mathcal{H}_j) \tag{8}$$

where

$$m(\phi|\mathcal{H}_j) = \sum_{\sigma\in\mathcal{A}} p(\sigma|\mathcal{H}_{j-1}) \Big[\log p(\sigma|\varepsilon^i_j, \phi, \mathcal{H}_{j-1}) - \log p(\sigma|\mathcal{H}_{j-1})\Big] \tag{9}$$

It should be noted that the initial Momentum is zero for all symbols, i.e. $m(\phi|\mathcal{H}_0) = 0, \forall\phi$.

Accordingly, the query selection objective can be defined as combination of mutual information and Momentum objectives in (7) and (8), respectively to balance between exploration and exploitation including all states as follows:

$$\phi^i_{s+1} \quad = \arg\max_{\phi\in\mathcal{A}} \; I(\sigma, \varepsilon|\phi, \mathcal{H}_s) + \lambda M(\phi|\mathcal{H}_s)\;, \;\; \lambda \geq 0 \tag{10}$$

where, λ is a tuning parameter that balances between mutual information and Momentum.

2.3 Improving Stopping Criterion

As we discussed in (1), in conventional RSE framework, the BCI system makes a decision when the state posterior achieves a pre-set confidence threshold, τ. However, meeting this fix threshold as a stopping of the estimation process may unnecessarily need more queries. The querying process is costly, particularly in BCI applications that require several sequences in order to precisely estimate the user intent. In [8], we have shown that using similar history-based objective, Momentum, the system can achieve to the target state faster than the threshold-based approach. Accordingly, the inference problem can be expressed as:

$$\begin{aligned} &\hat{\sigma} = \arg\max_{\sigma\in\mathcal{A}} p(\sigma|\mathcal{H}_s) \\ &\text{s.t. } p(\sigma|\mathcal{H}_s) + \lambda M(\sigma|\mathcal{H}_s) \geq \tau, \quad \lambda \geq 0 \end{aligned} \tag{11}$$

where

$$M(\sigma|\mathcal{H}_s) = \frac{1}{s}\sum_{j=1}^{s} p(\sigma|\mathcal{H}_{j-1}) \big[\log p(\sigma|\mathcal{H}_j) - \log p(\sigma|\mathcal{H}_{j-1})\big] \tag{12}$$

and λ is a hyperparameter. Since, Momentum is function of the user intent σ, we do not have any weighted average over $p(\sigma|\mathcal{H}_s)$.

Algorithm 1 Active Recursive State Estimation for Letter Decision in BCI

initialize $\mathcal{A},\ \lambda_1, \lambda_2 \in [0, \infty),\ \tau, K \in \mathbb{N}$
$s \leftarrow 0, \mathcal{H}_s \leftarrow \{\mathcal{H}_0\}$
stop $\leftarrow \{0|\forall \sigma \in \mathcal{A}\}$
while $x < \tau | \forall x \in stop$ **do**
  $\mathcal{A}' \leftarrow \mathcal{A}$
  for $i \in [0, K]$ **do** ▷ Batch query selection
    $\phi^i_{s+1} \leftarrow \arg\max_{\phi \in \mathcal{A}'} I(\sigma, \varepsilon^i_{s+1}|\phi, \mathcal{H}_s) + \lambda_2 M(\phi, \mathcal{H}_s)$ ▷ (10)
    $\mathcal{A}' \leftarrow \mathcal{A}' \setminus \{\phi^i_{s+1}\}$
  $\Phi_{s+1} \leftarrow \{\phi^i_{s+1}|\forall i \in [0, K]\}$
  ε_{s+1} observed evidence from the user for Φ_{s+1} ▷ EEG recording from BCI
  $\mathcal{H}_{s+1} \leftarrow \mathcal{H}_s \cup \{\varepsilon_{s+1}, \Phi_{s+1}\}$ ▷ Update History
  $s \leftarrow s + 1$
  stop $\leftarrow \{p(\sigma|\mathcal{H}_s) + \lambda_1 M(\sigma|\mathcal{H}_s)|\forall \sigma \in \mathcal{A}\}$ ▷ (13)
$\hat{\sigma} = \arg\max_{\sigma \in \mathcal{A}} p(\sigma|\mathcal{H}_s)$
return $\hat{\sigma}$

2.4 Active RSE Framework for BCI-based Typing Systems

Finally, using the Momentum in both inference constraint and query optimization, the final RSE framework can be presented as:

$$\begin{aligned} (I):&\quad \hat{\sigma} = \arg\max_{\sigma \in \mathcal{A}} p(\sigma|\mathcal{H}_s) \\ &\quad \text{s.t. } p(\sigma|\mathcal{H}_s) + \lambda_1 M(\sigma|\mathcal{H}_s) \geq \tau, \quad \lambda_1 \geq 0 \\ (Q):&\quad \phi^i_{s+1} = \arg\max_{\phi \in \mathcal{A}} \ I(\sigma, \varepsilon|\phi, \mathcal{H}_s) + \lambda_2 M(\phi|\mathcal{H}_s)\ , \ \ \lambda_2 \geq 0 \end{aligned} \tag{13}$$

The pseudocode of the proposed framework is demonstrated in Algorithm 1.

In the following sections, we will illustrate how using the introduced RSE framework can enhance accuracy and speed in the BCI-based typing systems.

3 Experiment Design

As a BCI typing system, we use **RSVP Keyboard**, which is a language-model-assisted EEG-based BCI typing interface. This typing interface uses RSVP paradigm that stands for rapid serial visual presentation [1, 2]. It includes a finite set of symbols from the English alphabet together with symbols corresponding to backspace and space, $\mathcal{A} = \{A, B, C, ..., Z, \_, <\}$, where _ and $<$ represent space and backspace, respectively. RSVP Keyboard system consists of the following main components:

Presentation: Controls the paradigm of visual stimuli presentation. The current system features three presentation paradigms: (i) rapid serial visual presentation paradigm, (ii) row column flash matrix presentation paradigm, and (iii) single symbol flash matrix presentation paradigm [12]. Figure 2a illustrates an example of keyboard presentation at each paradigm. In paradigm (i), a set of pseudo-randomly ordered stimuli are presented on a pre-fixed location of the screen in a rapid serial manner. Each stimulus is a trial. A set of trials which has been presented with no time gap in between, is called a sequence. Every sequence contains only a single target stimulus. Time-series analysis is used to identify the stimulus on which the attention (target) is placed on. In paradigm (ii), there exits an $R \times C$ matrix of symbols. Typically, each row and column of the matrix is flashed in a pseudo-random fashion, while the participants count the number of highlighted rows or columns that include the target symbol. Paradigm (iii) has similar matrix presentation to paradigm (ii), with the exception that in paradigm (iii) only single characters at each cell of the matrix are flashed (again in a pseudo-random fashion).

EEG Feature Extraction: Collects EEG evidence and applies some preprocessing steps including filtering, dimensionality reduction, etc. depending on the application. Since EEG signals are noisy with very low signal-to-noise ratio (SNR), it is essential that the EEG signal features are extracted as evidence. In EEG-BCIs the primary interest of filtering is to extract the P300 components [1, 13, 14]. Typically, filtering part includes drift removal (frequencies « 1Hz) and a bandpass filter to remove the artifact-related high frequency components in the measurement. After filtering, EEG is windowed to extract the respective evidence at each channel for stimuli presentations. Time-windowed data from different EEG channels is usually concatenated to obtain the EEG feature vector that has a high dimension because of using a multi-channel measurement. Therefore, dimensionality reduction using ICA or PCA is also needed [1]. Specifically, our system relies on reducing EEG time series into one dimensional feature vector. Simultaneously, the system also tries to achieve maximum separation between two classes as target and non-target respectively. Filtered multi-channel EEG data time windows are passed through channel-wise principal component analysis where the outputs are concatenated to an intermediate feature vector. We assume in each class, feature vectors are drawn from a multivariate Gaussian distribution we use Regularized discriminant analysis (regularized quadratic discriminant analysis [15]) that results in one dimensional representation of the signal. RDA regularizes the covariance matrices of each data class. For a binary classification problem with class notation $k \in \{0, 1\}$, let $x = \{x_1, x_2, \cdots, x_N\}$ denote the N sample data with respective labels $y = \{y_1, y_2, \cdots, y_N\}$. RDA involves two steps shrinkage and regularization, and these two steps require sample mean and covariance estimates for class k.

$$\mathcal{M}_k = \frac{1}{N_k} \sum_{i|y_i=k} x_i, \quad \Sigma_k = \frac{1}{N_k} \sum_{i|y_i=k} (x_i - \mathcal{M}_k)(x_i - \mathcal{M}_k)^T$$

The shrinkage adjusts the covariance matrices as the following;

$$\hat{\Sigma}_k(\gamma_1) = \frac{(1-\gamma_1)N_k\Sigma_k + \gamma_1 \sum_k N_k\Sigma_k}{(1-\gamma_1)N_k + \gamma_1 \sum_k N_k}$$

Observe in shrinkage parameter that $\gamma_1 \in [0, 1]$ adjusts the similarity between two class covariances and $\gamma_1 = 1$ leads to linear discriminant analysis (LDA). The regularization of the adjusted covariance matrices with the identity matrix I_p, where p denotes covariance matrix dimension, is achieved through;

$$\tilde{\Sigma}_k(\gamma_1, \gamma_2) = (1-\gamma_2)\hat{\Sigma}_k(\gamma_1) + \gamma_2 \frac{1}{p} \mathrm{tr}(\hat{\Sigma}_k(\gamma_1)) I_p$$

Here $\gamma_2 \in [0, 1]$ determines how circular the covariance matrix is. In inference, Gaussian distributions with parameters $(\mathcal{M}_k, \tilde{\Sigma}_k)$ for class k are used. In our system, parameters γ_1, γ_2 are collectively optimized to maximize the average cross validation area under receiver operation characteristic curve (AUC) with a equi-partitioned 10 folds. We refer readers to Orhan's work [16] for a detailed explanation and reasoning for the EEG feature extraction pipeline.

Language Model (LM): Provides the prior distribution that is used for computing the posterior distribution of a symbol. The most widespread LM is the n-gram statistical models that estimate the probability of a word given $(n-1)$ preceding words. In the current system, we are using an Online-Context Language Model (OCLM) that provides prior distributions given EEG evidence as part of our BCI system.

User Intent Detection (Inference): Estimate the intended user letter according to the EEG evidence collected under specified presentation paradigm and prior knowledge provided by LM component. EEG evidence typically is assumed to be a Gaussian process with unknown mean and covariance, and the parameter estimation is required for BCI inference. In order to estimate the posterior probability, the context prior $p\left(\sigma|\mathcal{H}_0\right)$ is provided by the LM, which estimates the conditional probability of every letter in the alphabet based on $n-1$ previously typed letters in a Markov model framework. Here, the BCI inference follows the introduced f_I objective in (13) using an active recursive state estimation framework.

Moreover, the described RSVP Keyboard system has three important system operation modes that are utilized in most of BCI applications:

Calibration mode: During calibration, users are asked to pay attention to pre-defined target symbols within randomly-ordered sequences to collect labeled EEG data. The data collected in this mode are used in the estimation of class-conditional EEG evidence distributions and classifier parameters.

Copy phrase task mode: In this mode, users are presented with a set of pre-defined phrases. Each phrase includes a missing word and the users are asked to complete the phrase by typing the missing word. This mode is designed to assess the system and the typing performance in terms of speed and accuracy in the presence of a language model. Figure 2b shows an example of a user attending Copy phrase task.

Simulation: This mode is basically a *copy phrase* task that is simulated without user intervention. We use the proposed probabilistic simulation framework in [17], in which the Monte-Carlo sampling method is used to draw samples from the class conditional distributions learned in the calibration mode. This mode that could replicate the operational performance of the system, could really be beneficial for evaluating new BCI designs and can report the possible system performance in terms of speed and accuracy.

Typically the RSVP presentation paradigm has lower speed than the matrix presentation. Therefore, for the experimental study, we only use RSVP paradigm to show the impact of the proposed framework on the typing speed and accuracy. More detailed information about the RSVP Keyboard system can be found in [1, 13, 14].

4 Results and Discussion

To assess the performance of the proposed query selection method, 10 healthy participants (six females), 20-35 years old were recruited under an approved IRB to assess the performance of the proposed system. EEG signals were acquired from 20 sensors according to international 10-20 system locations: Fp1, Fp2, Fz, F3, F4, F7, F8, Cz, C3, C4, T3, T4, T5, T6, P3, P4, O1, O2, A1 and A2. A DSI-24 Wearable Sensing EEG Headset was used for data acquisition, at a sampling rate of 300 Hz with active dry electrodes. All participants performed the *calibration* session containing 100 sequences; each sequence includes 5 trials; and one trial in each sequence is the target symbol which is displayed on the screen prior to each sequence (RSVP paradigm). The time interval between trials is 200 ms. Optimal parameters for both target and non-target class distributions were learned using the calibration data, which are used in simulation studies and *copy phrase* task.

After performing *calibration* session, all participants attended four *copy phrase* sessions. In *copy phrase* sessions, users were asked to type the following phrases in a pseudo-randomly ordered fashion.

- "It is **too** hot."
- "They **are** happy."
- "I am an **arctic** explorer."
- "It **occurred** randomly."
- "He read the **annals** of US history."
- "She needs one month to **convalesce**."

Here, the target words were written in bold (with green color during the experiment). We have attempted to pick different phrases with different difficulty levels in terms of prior probability provided by the LM. For instance, words such as "**too**" or "**are**" are relatively easy to type. Because, their initial symbol is very likely based on the LM prior. However, words like "**convalesce**" or "**occurred**" are fairly difficult to type. It is important to adjust the phrase difficulty to correctly assess the user performance and querying effects. The difficulty is measured by the confidence of the language model on the target phrase both on the letter and word level. None of the given phrases above is the likely candidate given the respective context with respect to the language model and hence the experimental design avoids auto typing. On letter base evaluation, 10 out of 36 target letters are the most likely candidates with respect to the language model yielding 28% accuracy in typing letters. In our experiments all users surpassed this baseline, by minimum 58% as shown in Section 4.2. Additionally, to quantify the importance of information obtained from the EEG signal, we present the experimental results for one of the subjects for typing “are, annals, arctic, too, convalesce”. Here, the blue bars show the prior probability provided by the language model and the red bars show the poster probability of each letter. Differences between blue and red bars show the probability increment of each letter due to EEG evidence (blue).

To have a comprehensive assessment, we examined the performance of the proposed RSE framework for both simulation and real-time typing experiments.

4.1 Simulation Experiments

To evaluate the empirical performance of the proposed query selection, a *copy phrase* task was simulated using EEG data collected during the calibration sessions. Using class conditional distributions f_{σ^*,ϕ_s^i} (target class) and $f_{\not\sigma^*,\phi_s^i}$ (non-target class), we used Monte-Carlo sampling method to draw samples from class distributions. Figure 4 shows the simulation results for typing symbols '*C*', '*O*', and first '*N*' for phrase '*SHE_NEEDS_ONE_MONTH_TO_CONVALESCE*' using four different methods in typing including random, only Momentum-based objective, MMI, and the introduced framework. For each simulation, the number of Monte-Carlo samples is chosen to be 500.

Figure 4 illustrates the average probability changes for the target symbol with different prior during the RSE process using different estimation approaches. All results presented for two users with different calibration performance; one with a higher calibration AUC, i.e. 0.82 and one with a lower calibration AUC, i.e. 0.67. AUC is the area under the receiver operating characteristics curve. The bar plot next to each plot shows the LM prior at the beginning of a decision cycle. For instance, in Figure 4a it can be seen that the LM probability for '*C*' is very low and it is not quite likely to start a word with this letter. Accordingly, MMI method that is performing as *N-best* selection, is highly influenced by the LM prior, needs more sequences to estimate the target symbol. Looking to the performance of the Momentum approach, we can see that in the early sequences of the decision process, this approach on average is faster than MMI to pick the intended symbol for the query subset. Although, after some sequences, because of EEG measurement noise and miss-classification, Momentum gets close to zero and could not pick the intended symbol. As expected, random method, due to the random query selection and decision does not perform well likewise. Overall, the proposed framework outperforms the other three methods in selecting the target symbols.

In Figure 4, we can also see that when there is a likely symbol like '*O*', MMI method and the proposed framework perform similarly in terms of learning the probability values of the symbols. By comparing the simulation results of user 7 (with higher AUC) with user 1 (lower AUC), it can be seen that all of the methods are faster for the user 7 with higher calibration performance due to larger gap between f_{σ^*,ϕ_s^i} and $f_{\not\sigma^*,\phi_s^i}$, which provides less miss-classification in the estimation process.

Another observation in Figure 4 is that for user 1 with lower performance where there is more overlap between class conditional distributions, the proposed framework can estimate the target symbol quite fast. However, it is more difficult for Momentum method to capture the target symbol. In such cases, MMI method also requires more number of sequences for a precise estimation.

Figure 5 presents the average probability changes and accuracy as a function of number of sequences for different discussed approaches. In this simulation, we used both conventional fix threshold and the proposed Momentum-based criterion for each query optimization method. The pre-defined threshold τ is 85%. The results for 500 Monte-Carlo simulation show that in all cases, the system can achieve similar accuracy level using the Momentum-based stopping criterion. In all sub-figures, again we visualized the results for user 1 and 7 with different typing performances. Figure 5a shows the simulation results for random method. Due to the low AUC value for user 1, there is no stopping point and the system continued querying until it reached the maximum number of sequences, i.e. 40 sequences. In Figure 5b Momentum method again could not precisely estimate the target symbol for both users. Comparing results in Figure 5c and Figure 5d, we can observe that for both MMI method and proposed framework, the Momentum-based stopping criterion terminated the estimation earlier (on average 3 sequences) compared to the fix threshold, with marginal accuracy loss.

4.2 Human-in-the-Loop Experiments: *Copy Phrase* Task

The calibration session proceeds with a short break (10 minutes) followed by four *Copy phrase* tasks including typing six introduced phrases with different typing difficulty levels. In this experiment, $\tau = 85\%$ and maximum number of sequences for typing a letter is 50. Each user participates four copy phrase tasks with different methods including random, Momentum, MMI, and the proposed framework. In the random method, the subset of queries was selected randomly and the stopping criterion was based on the fix threshold. In the Momentum method, the query selection was only using Momentum objective and the stopping criterion was based on the Momentum measure. For MMI, the query selection was based on mutual information and the stopping criterion is the conventional fix threshold. In the proposed one, we used the proposed objectives introduced in (13). The order of the tasks was randomly assigned to the participants to avoid the learning effect on the typing performance.

Table 2 shows the average typing performances of each user using different querying and stopping approaches. In this table, we reported the typing performance in terms of four measures: AUC of the classifier (trained using the *calibration* data); accuracy in typing a letter correctly (ATL) that is the total number of correctly typed letters divided by the total number of typed letters; accuracy of the phrase completion (APC) which is the average ATL for completed phrases; and probability of the phrase completion (PPC) that is total number of correctly typed phrases divided by the total number of phrases.

As another typical evaluation metric for BCI typing systems, we used information transfer rate (ITR) [18], which presents the information common to both the user intent σ and estimated symbol $\hat{\sigma}$. In fact, ITR summarizes the accuracy and speed into a single metric and it is commonly used to measure typing performance. Figure 6 illustrates the ITR values for all users sorted according to their calibration session AUC values.

Table 3 shows the average typing performance of all estimation methods for all users for *copy phrase* sessions. It shows the proposed framework outperforms the other methods in terms of speed and accuracy measures. Our statistical analysis also shows that the proposed framework significantly improved typing performance compared to MMI and random query selection ($p < 0.02$ based on Wilcoxon signed-rank test), which includes at least 52% enhancement for ITR, 30% for speed (1/Numseq), and 8.2% for ATL. Table 4 includes a summary of comparison between results reported in six different studies and results of the proposed framework in this study. The reported average ITR for the proposed method is much higher than reported ITRs for the other studies, which presents the impact of the proposed query selection and the stopping criterion methods on ITR values. We note that not all of these studies used RSVP-based BCI spellers; Only studies [4] and [16] used a similar RSVP presentation paradigm to what we have used in this paper and other studies employed either matrix speller or multi-RSVP paradigm.

5 Conclusion

A new recursive state estimation framework has been proposed for the non-invasive EEG-based BCI typing systems to enhance the typing speed and accuracy. The intended symbol estimation requires a selection of queries to be presented to the user, which can be formulated as a RSE problem. Taking advantages of posterior probability changes due to asking different questions across sequences, we introduce an active RSE framework for inference and sequence optimization by expanding the Momentum function proposed for active query optimization. To assess the proposed framework performance, a language-model-assisted EEG-based BCI typing system called RSVP Keyboard has been used. Using RSVP paradigm of this typing system, we examine the performance of the framework in both *simulation* and *copy phrase* modes. Our results showed that using Momentum jointly as stopping criterion and query optimization provides higher probability in picking the target letter that can lead to higher accuracy and speed in typing, especially in typing unlikely words. Our results for both *simulation* and *copy phrase* experiments showed that the proposed framework significantly outperforms the alternative estimation approaches.

6 Acknowledgment

We thank Bahar Azari for helping us in data collection. This work is supported by NSF (IIS-1149570, CNS-1544895, IIS-1715858, IIS-1717654, IIS-1844885), DHHS (90RE5017-02-01), and NIH (R01DC009834).

References

[1] Murat Akcakaya, Betts Peters, Mohammad Moghadamfalahi, Aimee Mooney, Umut Orhan, Barry Oken, Deniz Erdogmus, and Melanie Fried-Oken. Noninvasive brain computer interfaces for augmentative and alternative communication. *IEEE Reviews in Biomedical Engineering*, 7(1):31–49, 2014.

[2] Stephanie Lees, Natalie Dayan, Hubert Cecotti, Paul Mccullagh, Liam Maguire, Fabien Lotte, and Damien Coyle. A review of rapid serial visual presentation-based brain–computer interfaces. *Journal of neural engineering*, 15(2):021001, 2018.

[3] Matt Higger, Fernando Quivira, Murat Akcakaya, Mohammad Moghadamfalahi, Hooman Nezamfar, Mujdat Cetin, and Deniz Erdogmus. Recursive bayesian coding for bcis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(6):704–714, 2017.

[4] M. Moghadamfalahi, M. Akcakaya, H. Nezamfar, J. Sourati, and D. Erdogmus. An Active RBSE Framework to Generate Optimal Stimulus Sequences in a BCI for Spelling. *ArXiv e-prints (arXiv:1607.03578 [cs.HC])*, July 2016.

[5] Cyrus Omar, Abdullah Akce, Miles Johnson, Timothy Bretl, Rui Ma, Edward Maclin, Martin McCormick, and Todd P Coleman. A feedback information-theoretic approach to the design of brain–computer interfaces. *Intl. Journal of Human–Computer Interaction*, 27(1):5–23, 2010.

[6] Aziz Koçanaogulları, Murat Akçakaya, and Deniz Erdogmus. On analysis of active querying for recursive state estimation. *IEEE SIGNAL PROCESSING LETTERS*, 25(6):743, 2018.

[7] Theodoros Tsiligkaridis, Brian M Sadler, and Alfred O Hero. Collaborative 20 questions for target localization. *IEEE Transactions on Information Theory*, 60(4):2233–2252, 2014.

[8] Yeganeh M Marghi, Aziz Koçanaoğulları, Murat Akçakaya, and Deniz Erdoğmuş. A history-based stopping criterion in recursive bayesian state estimation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3362–3366. IEEE, 2019.

[9] Aziz Koçanaoğulları, Yeganeh M Marghi, Murat Akçakaya, and Deniz Erdoğmuş. Optimal query selection using multi-armed bandits. *IEEE Signal Processing Letters*, 25(12):1870–1874, 2018.

[10] Oliver Johnson and Andrew Barron. Fisher information inequalities and the central limit theorem. *Probability Theory and Related Fields*, 129(3):391–409, 2004.

[11] Vivek F Farias and Ritesh Madan. The irrevocable multiarmed bandit problem. *Operations Research*, 59(2):383–399, 2011.

[12] Mohammad Moghadamfalahi, Paula Gonzalez-Navarro, Murat Akcakaya, Umut Orhan, and Deniz Erdogmus. The effect of limiting trial count in context aware bcis: a case study with language model assisted spelling. In *International Conference on Augmented Cognition*, pages 281–292. Springer, 2015.

[13] Umut Orhan, Kenneth E Hild, Deniz Erdogmus, Brian Roark, Barry Oken, and Melanie Fried-Oken. Rsvp keyboard: An eeg based typing interface. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 645–648. IEEE, 2012.

[14] Mohammad Moghadamfalahi, Umut Orhan, Murat Akcakaya, Hooman Nezamfar, Melanie Fried-Oken, and Deniz Erdogmus. Language-model assisted brain computer interface for typing: A comparison of matrix and rapid serial visual presentation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(5):910–920, 2015.

[15] Jerome H Friedman. Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175, 1989.

[16] Umut Orhan. Rsvp keyboard (tm): an eeg based bci typing system with context information fusion. *Dissertations & Theses-Gradworks*, 2013.

[17] Umut Orhan, Hooman Nezamfar, Murat Akcakaya, Deniz Erdogmus, Matt Higger, Mohammad Moghadamfalahi, Andrew Fowler, Brian Roark, Barry Oken, and Melanie Fried-Oken. Probabilistic simulation framework for eeg-based bci design. *Brain-Computer Interfaces*, 3(4):171–185, 2016.

[18] Bernhard Obermaier, Christa Neuper, Christoph Guger, and Gert Pfurtscheller. Information transfer rate in a five-classes brain-computer interface. *IEEE Transactions on neural systems and rehabilitation engineering*, 9(3):283–288, 2001.

[19] Dennis J McFarland, William A Sarnacki, and Jonathan R Wolpaw. Brain–computer interface (bci) operation: optimizing information transfer rates. *Biological psychology*, 63(3):237–251, 2003.

[20] Emanuel Donchin, Kevin M Spencer, and Ranjith Wijesinghe. The mental prosthesis: assessing the speed of a p300-based brain-computer interface. *IEEE transactions on rehabilitation engineering*, 8(2):174–179, 2000.

[21] Zhimin Lin, Chi Zhang, Ying Zeng, Li Tong, and Bin Yan. A novel p300 bci speller based on the triple rsvp paradigm. *Scientific reports*, 8(1):3350, 2018.

[22] Hilit Serby, Elad Yom-Tov, and Gideon F Inbar. An improved p300-based brain-computer interface. *IEEE Transactions on neural systems and rehabilitation engineering*, 13(1):89–98, 2005.

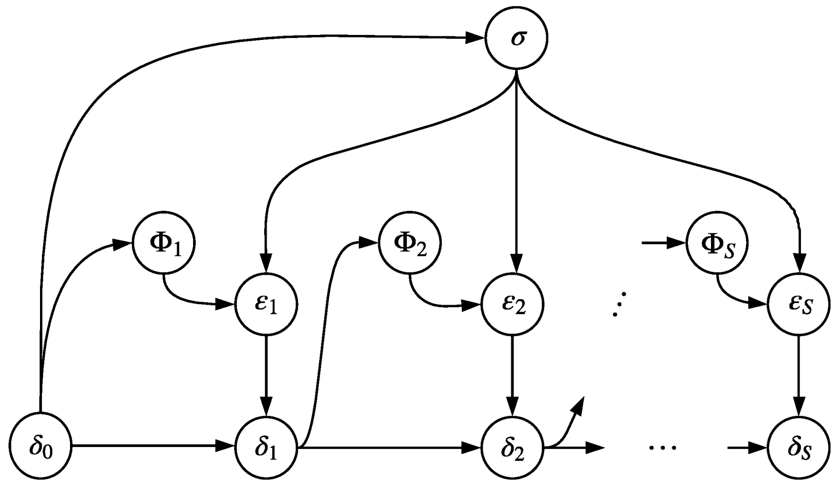
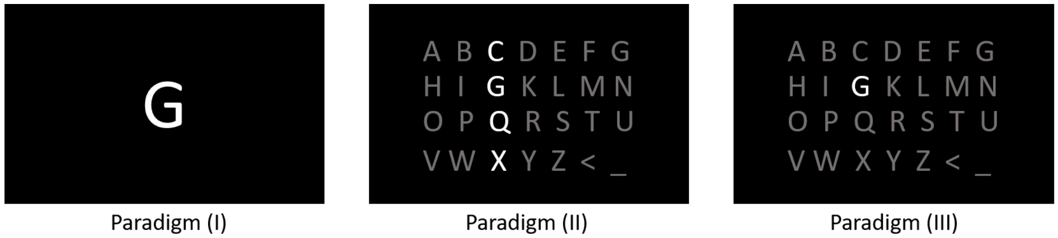
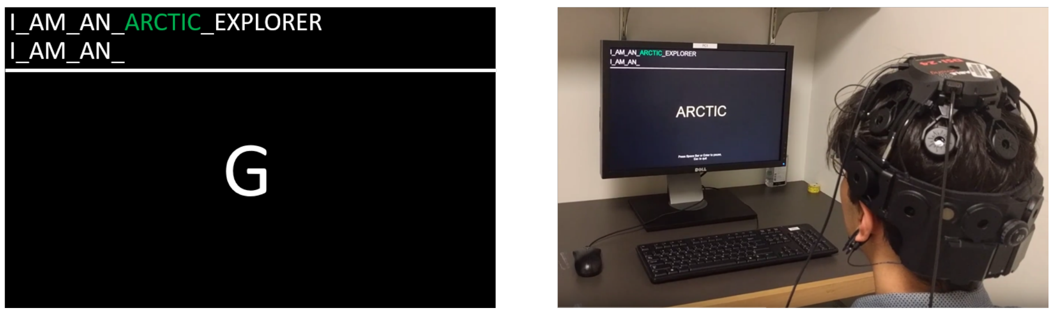

σ
Φ_1
Φ_2
Φ_S
ε_1
ε_2
ε_S
δ_0
δ_1
δ_2
δ_S

Figure 1


G
A B C D E F G
H I G K L M N
O P Q R S T U
V W X Y Z < _
A B C D E F G
H I G K L M N
O P Q R S T U
V W X Y Z < _
Paradigm (I)
Paradigm (II)
Paradigm (III)

(a)


I_AM_AN_ARCTIC_EXPLORER
I_AM_AN_
G
ARCTIC

(b)

Figure 2

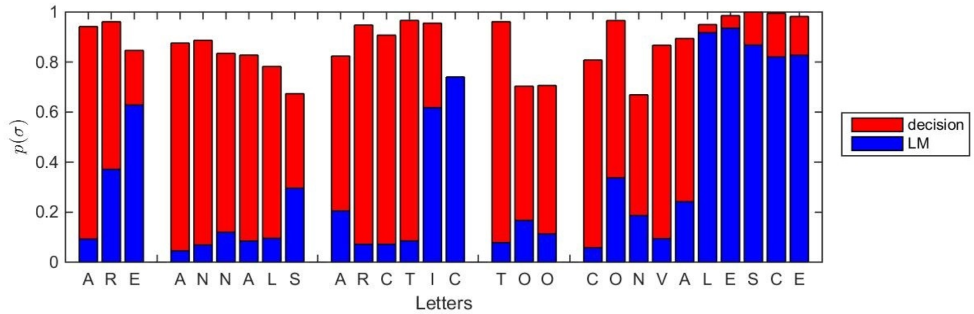
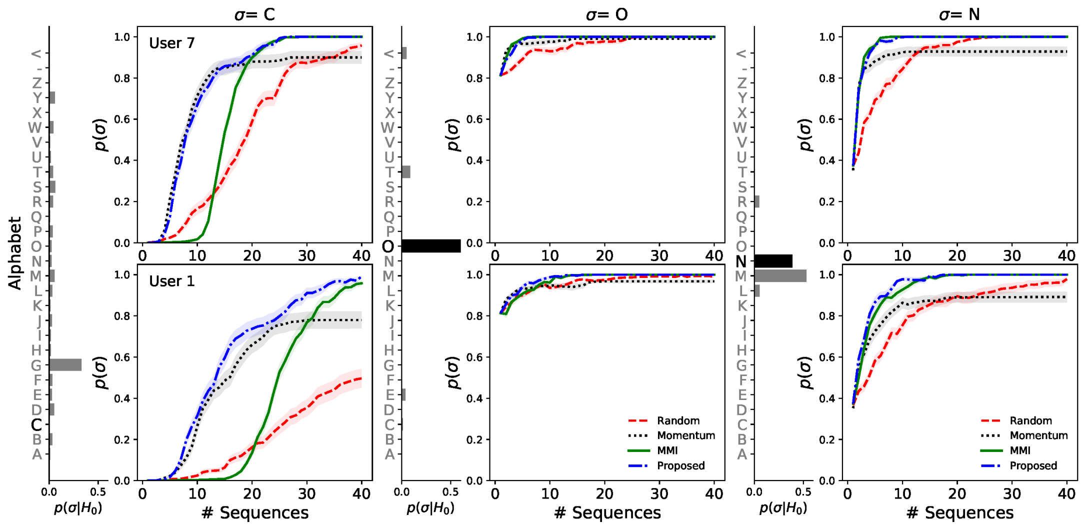

1
0.8
0.6
0.4
0.2
0
$p(\sigma)$
A R E
A N N A L S
A R C T I C
T O O
C O N V A L E S C E
Letters
decision
LM

Figure 3


σ= C
σ= O
σ= N
User 7
User 1
Alphabet
p(σ)
p(σ|H0)
# Sequences
Random
Momentum
MMI
Proposed

(a) (b) (c)

Figure 4

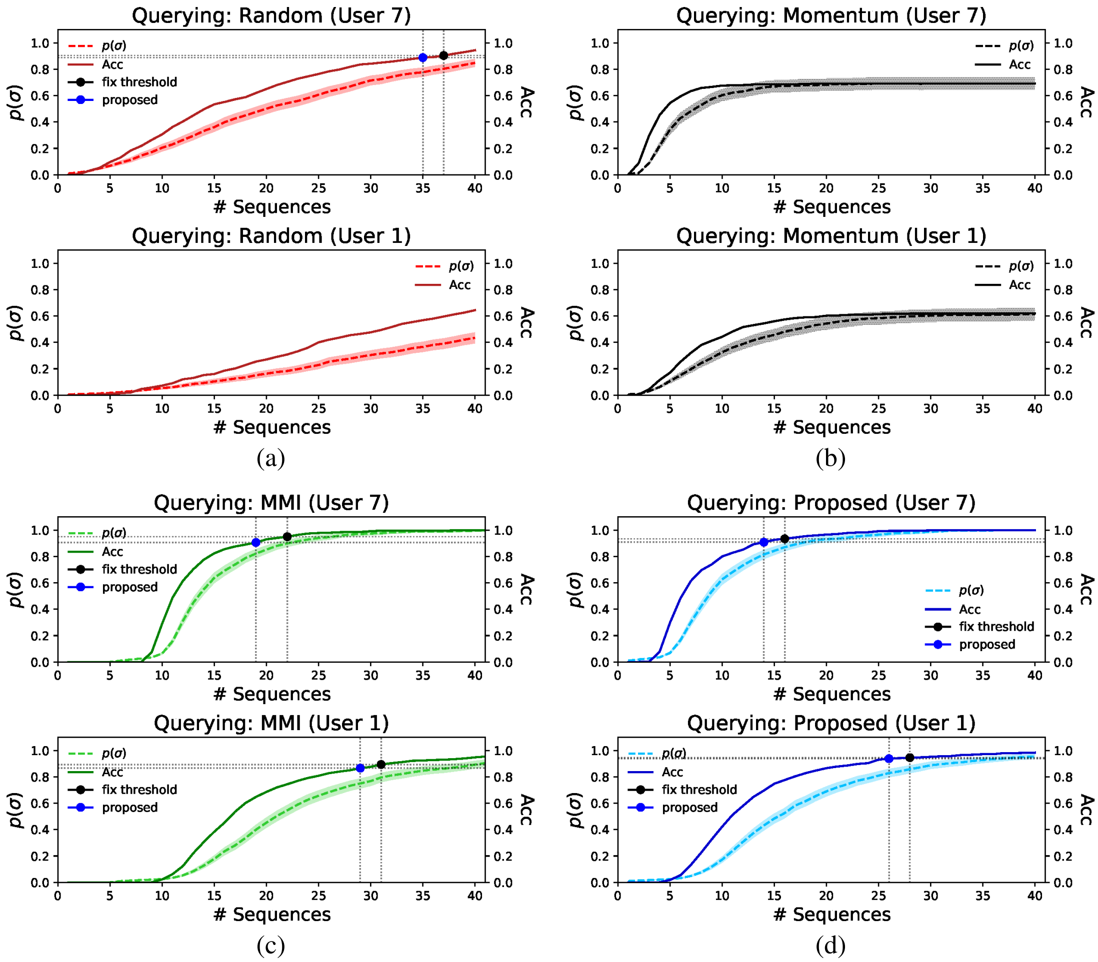

Querying: Random (User 7)
Querying: Momentum (User 7)
Querying: Random (User 1)
Querying: Momentum (User 1)
(a)
(b)
Querying: MMI (User 7)
Querying: Proposed (User 7)
Querying: MMI (User 1)
Querying: Proposed (User 1)
(c)
(d)
$p(\sigma)$
Acc
fix threshold
proposed
Sequences

Figure 5

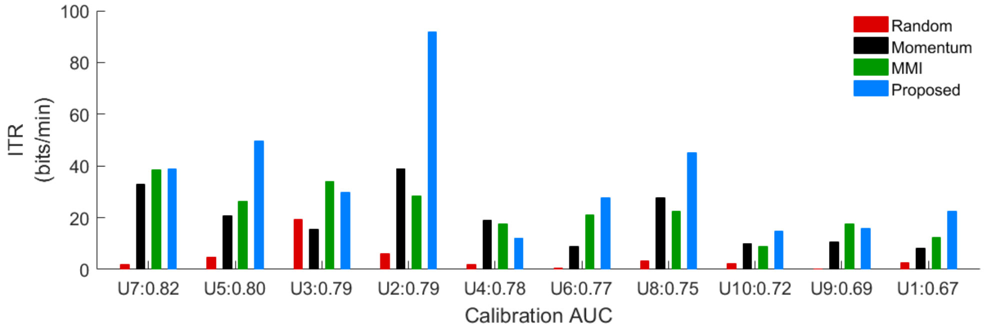

Random
Momentum
MMI
Proposed
ITR
(bits/min)
0
20
40
60
80
100
U7:0.82
U5:0.80
U3:0.79
U2:0.79
U4:0.78
U6:0.77
U8:0.75
U10:0.72
U9:0.69
U1:0.67
Calibration AUC

Figure 6

Table 1: Summary of the frequently used notation.

Notation	Definition
σ	user intended symbol
$\mathcal{A}$	finite set of typing symbols including alphabet, space and backspace
Φ_s	subset of queries (visual stimuli) at sequence s
ϕ_s^i	single query at trail i, sequence s
ε_s	recorded EEG at sequence s
$\mathcal{H}_0$	previously typed symbols (if exists)
$p(\sigma\|\mathcal{H}_0)$	prior information provided by the LM
$\mathcal{H}_s$	task history including $\{\varepsilon_{1:s}, \Phi_{1:s}, \mathcal{H}_0\}$
$I(\sigma, \varepsilon)$	mutual information between the user intent and collected EEG
$M(\sigma\|\mathcal{H}_s)$	Momentum function
τ	minimum confidence level for typing a symbol
λ	hyperparameter of the typing framework
S	total number of sequences

Table 2: Typing performance for 10 participants performing RSVP Keyboard task using four estimation approaches. AUC$_{ca}$ represents the AUC value for the calibration task and AUC$_{cp}$ presents the AUC value of the *copy phrase* task.

		Random				Momentum				MMI				Proposed			
User	**AUC$_{ca}$**	**AUC$_{cp}$**	**ATL**	**APC**	**PPC**	**AUC$_{cp}$**	**ATL**	**APC**	**PPC**	**AUC$_{cp}$**	**ATL**	**APC**	**PPC**	**AUC$_{cp}$**	**ATL**	**APC**	**PPC**
1	0.67	0.90	0.22	0.00	0.00	0.89	0.49	0.75	0.50	0.91	0.63	0.73	0.67	0.89	0.69	1.00	0.67
2	0.79	0.96	0.49	0.43	0.17	0.93	0.84	0.69	1.00	0.95	0.75	0.88	0.83	0.95	0.97	0.95	1.00
3	0.79	0.96	0.77	0.62	0.50	0.95	0.71	0.64	0.83	0.91	0.81	0.61	1.00	0.87	0.79	0.57	1.00
4	0.78	0.95	0.18	0.01	0.00	0.93	0.76	0.49	1.00	0.85	0.68	0.53	0.83	0.88	0.67	0.73	0.67
5	0.80	0.76	0.28	0.60	0.33	0.75	0.62	0.70	0.50	0.83	0.71	0.82	0.83	0.76	0.83	1.00	0.83
6	0.77	0.75	0.12	0.00	0.00	0.83	0.51	0.82	0.33	0.86	0.66	1.00	0.67	0.85	0.70	0.76	0.67
7	0.82	0.94	0.18	0.00	0.00	0.92	0.82	0.64	1.00	0.92	0.76	0.88	0.83	0.90	0.73	0.82	0.83
8	0.75	0.91	0.26	0.00	0.00	0.90	0.83	0.74	0.83	0.89	0.77	0.49	1.00	0.82	0.86	0.71	1.00
9	0.69	0.76	0.04	0.00	0.00	0.69	0.85	0.59	0.50	0.83	0.58	0.60	0.50	0.85	0.64	0.85	0.67
10	0.72	0.90	0.21	1.00	0.17	0.89	0.48	1.00	0.33	0.92	0.55	0.80	0.50	0.92	0.61	1.00	0.67

Table 3: Average typing performance across all participants performing RSVP Keyboard task using four estimation approaches. The reported average values for Random and Momentum belong to cases that the participant completed the task before reaching the maximum number of sequences (mandatory termination).

	Random	**Momentum**	**MMI**	**Proposed**
ITR (bits/min)	03.6	18.6	21.6	34.8
NumSeq	45	38	31	24
ATL	0.28	0.69	0.69	0.75
APC	0.27	0.71	0.73	0.84
PPC	0.12	0.68	0.77	0.80

Table 4: Comparing the performance of the proposed typing framework with other BCI spellers using EEG measurement and different stimuli paradigms.

Method	Paradigm	# Subjects(# Healthy)	ITR [bits/min]
McFarland's work [19]	Matrix	8 (6)	08.4
Orhan's work [16]	RSVP	12 (10)	09.6
Moghadamfalahi's work [4]	RSVP	12 (12)	16.2
Donchin's work [20]	Matrix	10 (10)	19.8
Lin's work [21]	3-RSVP	13 (13)	20.4
Serby's work [22]	Matrix	6 (6)	22.2
Our work	RSVP	10 (10)	**34.8**

Figure Captions

Figure 1: The graphical representation of recursive state estimation for BCI-based typing systems. The dynamics of the system follows a Markov decision process. σ denotes the state, ε_s, Φ_s represent observed evidence and queries for sequence s respectively. δ_0 denotes the prior information before the typing (e.g. language model).

Figure 2: (a) Schematic illustration of three presentation paradigms in RSVP Keyboard system. The system has three present modalities including RSVP speller, single letter matrix speller and row-column flash matrix speller.(b) Copy phrase task for EEG-based BCI using RSVP paradigm. The user is presented by a pre-determined phrase and tasked to complete it. In the example the user is tasked to type *ARCTIC*.

Figure 3: Experimental results obtained from one of our subjects for the typing tasks "are, annals, arctic, too, convalesce". The figure represents the prior probabilities provided by the language model (LM) (blue) and the posterior probabilities when the decision is made (red). The differences between the bars are the results of the user's EEG response. It is apparent from the figure EEG drives the selection.

Figure 4: Probability of the letter completion for 500 Monte-Carlo simulations for typing three target symbols in phrase '*CONVALESCE*'. Intended symbols contain: (a)'C', (b)'O', and (c) 'N' in the target phrase. Simulation results are reported for two users with different calibration performances. User 7 with AUC $= 0.82$ has lower performance than user 1 with AUC $= 0.67$. Bar plots show the LM prior probability over all typing symbols before typing each letter.

Figure 5: Probability of the letter completion for 500 Monte-Carlo simulations for typing a single letter using four query optimization methods and two stopping criteria. (a) Simulation results for random querying using fix threshold and the Momentum-based stopping criteria. (b) Simulation results for Momentum querying using fix threshold and the Momentum-based stopping criteria. (c) Simulation results for MMI querying using fix threshold and the Momentum-based stopping criteria. (d) Simulation results for the proposed querying using fix threshold and the Momentum-based stopping criteria. In all sub-figures, the top plot belongs to User 7 with AUC $= 0.82$ and the bottom plot belongs to User 1 with AUC $= 0.67$. Dots presents the stopping point according to each criterion.

Figure 6: Average of information transfer rate for four query selection methods: Random (red), Momentum (black), MMI (green), and the proposed framework (blue). All of the results belongs to 10 users attending the copy phrase task in RSVP Keyboard experiment.