The Effect of Limiting Trial Count in Context Aware BCIs: A Case Study with Language Model Assisted Spelling

Mohammad Moghadamfalahi[1](✉), Paula Gonzalez-Navarro[1],
Murat Akcakaya[2], Umut Orhan[3], and Deniz Erdogmus[1]

[1] Northeastern University, Boston, USA
moghadam@ece.neu.edu
[2] University of Pittsburgh, Pittsburgh, USA
[3] Honeywell International, Morristown, USA

Abstract. Deflections in recorded electroencephalography (EEG) in response to visual, auditory or tactile stimuli have been popularly employed in non-invasive EEG based brain computer intefaces (BCIs) for intent detection. For example, in an externally stimulated typing BCI, an accurate estimate of the user intent might require long EEG data collection before the system can make a decision with a desired confidence. Long decision period can lead to slow typing and hence the user frustration. Therefore, there is a trade-off between the accuracy of inference and the typing speed. In this manuscript, using Monte-Carlo simulations, we assess the speed and accuracy of a Language Model (LM) assisted non-invasive EEG based typing BCI, RSVPKeyboard$^{\mathrm{TM}}$, as a function of the maximum number of repetitions of visual stimuli sequences and the inter-trial interval (ITI) within the sequences. We show that the best typing performance with RSVPKeyboard$^{\mathrm{TM}}$can be obtained when ITI=150 ms and maximum number of allowed sequences is 8. Even though the probabilistic fusion of the language model with the EEG evidence for joint inference allows the RSVPKeyboard$^{\mathrm{TM}}$ to perform auto-typing when the system is confident enough t o make decisions before collecting EEG evidence, our experimental results show that RSVPKeyboard$^{\mathrm{TM}}$does not benefit from auto-typing.

1 Introduction

Non-invasive electroencephalography (EEG)-based brain computer interfaces (BCIs) provide means for people with severe motor and speech impairments to communicate their needs and desires to their family members, friends and care takers. A typing interface is a very common application domain for these EEG-based BCIs [1]. A well established signal paradigm used for intent detection in typing BCIs is based on event related potentials (ERPs) which are responses to presented visual, auditory or tactile stimuli [1–4].

Rapid serial visual presentation (RSVP) is a presentation scheme that distributes the symbols chosen for intent communication in time to induce ERPs.

D.D. Schmorrow and C.M. Fidopiastis (Eds.): AC 2015, LNAI 9183, pp. 281–292, 2015.
DOI: 10.1007/978-3-319-20816-9_27

This paradigm is popularly used in visual stimuli based typing BCIs, in which the presented sequence of symbols on the screen usually consists of the 26 letters of the English Alphabet, 10 numerical digits, and space and backspace symbols [5–9]. A sequence is usually repeated multiple times to improve the accuracy in intent detection, due to the low signal-to-noise ratio in the recorded EEG data in response to the presented visual stimuli. This repetition in turn decreases the speed of the typing interface. A faster typing on the other hand usually decreases the typing accuracy [10]. However, any typing interface can benefit from a proper language model to enhance both speed and inference accuracy. In this manuscript, we analyze the effect of the repetition counts of the presented symbols and the time delay between consecutive symbols on the typing performance in the presence of a language model.

We have designed a language model assisted non-invasive EEG based letter by letter typing interface, RSVPKeyboard$^{\text{TM}}$. This system presents a subset of 28 symbols (26 letters and space and backspace symbols) as a sequence after which the BCI attempts to make a decision. If a preset confidence level is reached, a decision is made. Otherwise the sequences are repeated multiple times until the confidence level or a bound on the maximum number of repetitions of the sequences is reached. Such a decision period is denoted as an epoch. To analyze the effect of (1) bound on the maximum number of repetitions in an epoch and (2) the minimum number of sequences before a system can make a decision in an epoch on the typing performance of RSVP Keyboard$^{\text{TM}}$, we use Monte Carlo simulations. Note here that setting the minimum number of sequences to "0" allows the system to auto-type if the maximum of the symbol probabilities provided by the language model exceeds the desired confidence threshold. Therefore, our analysis in this manuscript will also investigate the effects of auto-typing on typing performance.

2 RSVPKeyboard$^{\text{TM}}$

As illustrated in Fig. 1, RSVPKeyboad$^{\text{TM}}$ consists of three major components (A) presentation component, (B) feature extraction component, (C) decision making component; in the following we will describe each component in detail.

2.1 Presentation Component

Presentation component is utilized to present the stimuli for ERP induction and also for providing feedback to the user. RSVP paradigm in which the trials are rapidly presented one at a time at a predefined fixed location on the screen in a pseudo-random order is utilized as the presentation scheme [1,5–9]. As also described in Sect. 1, a presented sequence consists of a subset from the dictionary $\mathcal{A}$ = {26 alphabetical symbols, "<" as the backspace, and "_" for the space character}. After a sequence, the typing interface makes a decision on the user intended symbol if a desired confidence level is reached; otherwise, sequences are repeated until the desired confidence or a preset maximum bound on the number

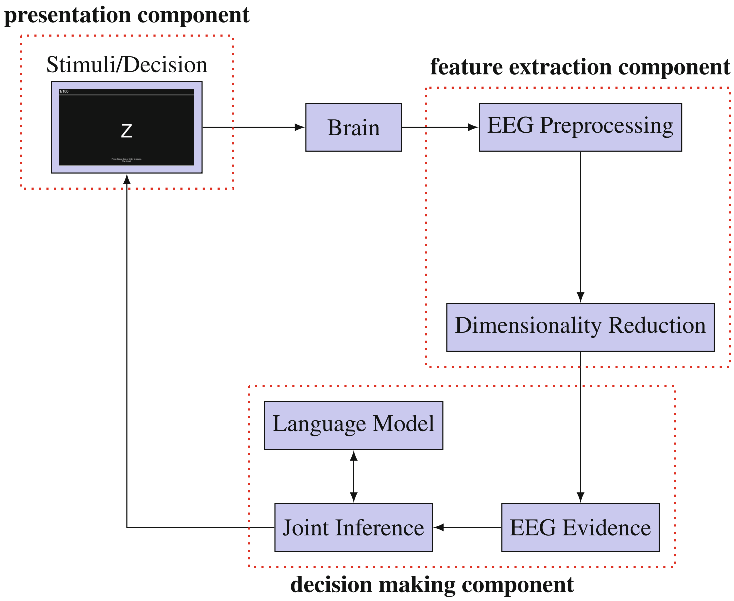


Fig. 1. The RSVPKeyboard$^{\mathrm{TM}}$ flowchart.

of sequence repetitions is reached and a decision is made accordingly. This time period in which a decision is made is denoted as an epoch.

2.2 Feature Extraction Component

EEG Preprocessing: EEG signals are acquired using a g.USBamp bio-signal amplifier with active g.Butterfly electrodes at a sampling rate of 256 Hz, from 16 EEG locations (according to the International 10/20 configuration): Fp1, Fp2, F3, F4, Fz, Fc1, Fc2, Cz, P1, P2, C1, C2, Cp3, Cp4, P5 and P6 . Recorderd signals are filtered by a finite impulse response (FIR) linear-phase bandpass filter cut-off frequencies 1.5 and 42 Hz with zero DC gain, and a notch filter at 60 Hz to improve the signal to noise ratio (SNR), and to eliminate DC drifts.

EEG data from a time window of $[0, 500)$ ms time locked to each trial is processed as the raw data for that trial in order to capture the P300 signal and avoid motor related potentials [11].

Dimensionality Reduction: EEG dimensionality reduction is performed in two steps. First, considering the upper limit of the FIR filter, EEG data is downsampled by the factor of 2. Then, to remove directions with negligible variance at each channel, Principle Component Analysis (PCA) is utilized and the EEG data are projected onto a lower dimensional feature space. Finally, this component concatenates the data from all channels corresponding to the same trial to form a feature vector for each trial. In decision making component, the concatenated feature vectors are non-linearly projected on a one dimensional space (for detail please see Sect. 2.3).

2.3 Decision Making Component

Decision making component fuses the EEG and language model evidence following the assumptions illustrated in the graphical model of Fig. 2 for joint intent inference. This graphical model demonstrates the EEG data collected at the s^{th} sequence of the e^{th} epoch. Throughout this paper we refer to the vocabulary set as $\mathcal{A} = \{\mathscr{a}_1, \mathscr{a}_2, ..., \mathscr{a}_{|\mathcal{A}|}\}$.

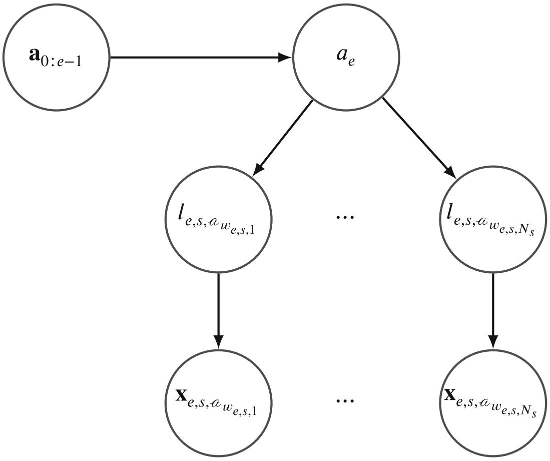

$\mathbf{a}_{0:e-1}$
a_e
$l_{e,s,\mathscr{a}_{w_{e,s,1}}}$
...
$l_{e,s,\mathscr{a}_{w_{e,s,N_s}}}$
$\mathbf{x}_{e,s,\mathscr{a}_{w_{e,s,1}}}$
...
$\mathbf{x}_{e,s,\mathscr{a}_{w_{e,s,N_s}}}$

Fig. 2.

Here“a_e” is the desired character for epoch “e” and “$\mathbf{a}_{0:e-1} = \{a_0, a_1, a_2, ..., a_{e-1}\}$” is the set of all previous characters which define the context and accordingly a prior probability over the epoch's possible outcomes. In RSVP paradigm, each trial contains only one symbol and $\mathbf{w}_{e,s} = \{w_{e,s,1}, w_{e,s,2}, ..., w_{e,s,N_s}\}$ is the index set of symbols, presented in epoch “e” and at sequence “s”. Also N_s is the number of trials in each sequence. The label set for epoch “e” and sequence “s” is defined as $\mathbf{l}_{e,s} = \{l_{e,s,\mathscr{a}_{w_{e,s,1}}}, l_{e,s,\mathscr{a}_{w_{e,s,2}}}, ..., l_{e,s,\mathscr{a}_{w_{e,s,N_s}}}\}$

where:

$$l_{e,s,\mathscr{a}_{w_{e,s,i}}} = \begin{cases} 1, & a_e = \mathscr{a}_{w_{e,s,i}} \\ 0, & a_e \neq \mathscr{a}_{w_{e,s,i}} \end{cases}$$

Finally, $\mathbf{X}_{e,s} = \{\mathbf{x}_{e,s,\mathscr{a}_{w_{e,s,1}}}, \mathbf{x}_{e,s,\mathscr{a}_{w_{e,s,2}}}, ..., \mathbf{x}_{e,s,\mathscr{a}_{w_{e,s,N_s}}}\}$ is the set of EEG evidence for trials in epoch “e” and sequence “s”.

EEG Evidence: Regularized discriminant analysis (RDA) is applied to the features extracted as explained in Sect. 2.2. RDA is a generalization of the quadratic discriminant analysis and it allows full-rank estimation of high dimensional covariance matrices from proportionally few number of supervised realizations by applying shrinkage and regularization on the maximum likelihood covariance estimates [12]. Assume $\mathbf{x}_i \in \mathbb{R}^p$ is p dimensional feature vector and $l_i \in \{0,\ 1\}$ is the class label for $\mathbf{x}_i$ which can represent the target class ("$l_i = 1$") or the non-target class ("$l_i = 0$"). Then the maximum likelihood estimator of the mean and covariance for each class are as in (1).

$$\begin{aligned}\boldsymbol{\mu}_k &= \frac{1}{N_k}\sum_{i=1}^{N}\mathbf{x}_i\delta_{l_i,k}\\ \boldsymbol{\Sigma}_k &= \frac{1}{N_k}\sum_{i=1}^{N}(\mathbf{x}_i-\boldsymbol{\mu}_k)(\mathbf{x}_i-\boldsymbol{\mu}_k)^T\delta_{l_i,k}\end{aligned} \tag{1}$$

where $\delta_{i,j}$ is the Kronecker-δ such that it is equal to 1 when $i = j$ and zero otherwise, $k \in \{0, 1\}$ is the class indicator, and N_k is the number of training data in class k. Define $N = N_0 + N_1$, as the total number of feature vectors from both classes, then through the shrinkage the class conditional covariance estimators are defined as in (2).

$$\hat{\boldsymbol{\Sigma}}_k(\lambda) = \frac{(1-\lambda)N_k\boldsymbol{\Sigma}_k + (\lambda)\sum_{k=0}^{1}N_k\boldsymbol{\Sigma}_k}{(1-\lambda)N_k + (\lambda)\sum_{k=0}^{1}N_k} \tag{2}$$

The shrinkage parameter $\lambda \in [0, 1]$, defines the similarity between the class conditional covariances such that when $\lambda = 1$ both covariances become equal and the RDA reduces to linear discriminant analysis (LDA). Regardless of the shrinkage step, if the dimensionality of the feature space is higher than the total number realizations in every class, (2) cannot produce invertible covariance matrices. Then, the next step in RDA, known as regularization, is applied and it changes the estimator as in (3).

$$\hat{\boldsymbol{\Sigma}}_k(\lambda,\gamma) = (1-\gamma)\hat{\boldsymbol{\Sigma}}_k(\lambda) + (\gamma)\frac{1}{p}tr[\hat{\boldsymbol{\Sigma}}_k(\lambda)]\mathbf{I}_p \tag{3}$$

$tr[.]$ is the trace operator, $\mathbf{I}_p$ is a $p \times p$ identity matrix and $\gamma \in [0, 1]$ is the regularization parameter which determines the circularity of the covariance matrix. Finally, the discriminant score in RDA is defined as in (4).

$$d_{RDA}(\mathbf{x}) = log\frac{f_{\mathcal{N}}(\mathbf{x};\boldsymbol{\mu}_1,\hat{\boldsymbol{\Sigma}}_1(\lambda,\gamma))\hat{\pi}_1}{f_{\mathcal{N}}(\mathbf{x};\boldsymbol{\mu}_0,\hat{\boldsymbol{\Sigma}}_0(\lambda,\gamma))\hat{\pi}_0} \tag{4}$$

In (4), $f_{\mathcal{N}}(\mathbf{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$ is the Gaussian probability density function when $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ and $\hat{\pi}_k$ is the prior probability of class k. After this projection, we utilize kernel density estimation (KDE) to compute the class conditional distributions over the RDA scores using (4).

$$\begin{aligned} &f(\mathbf{x}=\mathbf{y}|l=k) = \\ &f_{KDE}(d_{RDA}(\mathbf{x}) = d_{RDA}(\mathbf{y})|l=k) = \\ &\frac{1}{N_k}\sum_{i=1}^{N}\mathcal{K}_{h_k}(d_{RDA}(\mathbf{x}_i), d_{RDA}(\mathbf{y}))\delta_{l_i,k} \end{aligned} \tag{5}$$

In (5), $\mathcal{K}_{h_k}(\cdot,\cdot)$ is a suitable kernel function with bandwidth h_k. In our work, we employ Gaussian kernels and compute h_k using the Silverman rule of thumb [13].

Language Model: The language model in our system is utilized both for prioritizing the symbols for presentation in an iterative Bayesian framework, and for making inference after each sequence by imposing a prior distribution on symbols. We use an n-gram language model that provides a probability distribution over the vocabulary based on $n-1$ previously type characters as shown in (6) [14].

$$p(a_e = \mathscr{a}|\mathbf{a}_{(0:e-1)} = \mathbf{a}) = \frac{p(a_e = \mathscr{a}, \mathbf{a}_{(0:e-1)} = \mathbf{a})}{p(\mathbf{a}_{(0:e-1)} = \mathbf{a})} \tag{6}$$

Here a_e is the symbol to be typed at the e^{th} epoch and $\mathbf{a}_{(0:e-1)}$ are the typed symbols prior to epoch e. RSVPKeyboard$^{\text{TM}}$, utilizes a 6-gram letter model, which is trained on the NY Times portion of the English Gigaword corpus [14].

Joint Inference: Maximum a posterior (MAP) inference mechanism is used in our system for joint inference after each sequence. According to the graphical model presented in Fig. 2, the posterior probability distribution over the vocabulary is calculated using

$$\begin{aligned} Q = p(a_e = \mathscr{a}|\mathcal{X}_e = \mathbf{X}_e, \mathbf{a}_{(0:e-1)} = \mathbf{a}) \propto \\ p(\mathcal{X}_e = \mathbf{X}_e, \mathbf{a}_{(0:e-1)} = \mathbf{a}|a_e = \mathscr{a})P(a_e = \mathscr{a}) \end{aligned} \tag{7}$$

Then, using the conditional independence assumption on the trials within a sequence, the fusion rule is simplified to

$$Q \propto \left(\prod_{s=1}^{N_e}\prod_{t=1}^{N_s}\frac{f(\mathbf{x}=\mathbf{x}_{e,s,w_{e,s,t}}|l=1)}{f(\mathbf{x}=\mathbf{x}_{e,s,w_{e,s,t}}|l=0)}\right)P(a_e = \mathscr{a}|\mathbf{a}_{(0,e-1)} = \mathbf{a}) \tag{8}$$

In (8), N_e represents the number of sequences presented in the e^{th} epoch in N_s number of sequences. The inference mechanism in our system requires that the posterior probability of the estimated target reaches a predefined threshold. Otherwise, more sequences will be presented until either the mentioned confidence level is reached, or the number of sequences in the epoch e, N_e, attains its upper limit.

2.4 System Operation Modes

RSVPKeyboard$^{\text{TM}}$ can be used in five different operation modes.

(i). *Calibration mode:* The inference mechanism of the system requires supervised EEG data, which are collected during a calibration task. Typically, each calibration session consists of 100 sequences of symbols. Before each sequence the user is asked to attend to a particular symbol. Then a sequence consisting of the target symbol and 9 other non-target symbols is presented to the user in a random order.

(ii). *Copy phrase task mode:* In this mode, a user is asked to type missing words in a set of phrases. Utilizing the language model, 5 difficulty levels for phrases based on the predictability of the missing phrases have been defined: 1 as the most and 5 being the least predictable. The sentences used in a standard copy phrase task are selected such that their difficulty level span a uniform distribution.

(iii). *Mastery task mode:* A complete mastery task consists of 5 levels. To complete each level, the user should successfully copy the missing words in two different phrases with the same difficulty level. The level index is assigned by the difficulty level of the phrases. This mode is specifically designed for user training.

(iv). *Free spelling mode:* In this mode, the system can be employed by the user to type the desired text.

(v). *Simulation mode:* In a simulation session, with Monte Carlo simulations, many copy phrase tasks are completed using the samples drawn from class conditional KDEs (see (5)). These samples are specifically used to construct the EEG scores and are fused with the language model for joint inference as in (8) [7]. Simulation mode reports the system performance in terms of the probability of phrase completion (PPC) and total task duration (TTD).

3 Experiment and Results

In this manuscript, to analyze the typing performance of the RSVPKeyboard$^{\text{TM}}$, we employ the simulation mode. We set the sequence size as 15 symbols/trials (always including a backspace symbol in a sequence); choose the time difference between two consecutive trials, known as inter trial interval (ITI), from the set $\{0.085,\ 0.1,\ 0.15,\ 0.2\}ms$; and assign the minimum ($SN_{min} \in \{0,\ 1\}$) and maximum ($SN_{max} \in \{2,\ 4,\ 8,\ 16\}$) number of sequences in an epoch. Note here that setting $SN_{min} = 0$ represents the auto-typing scenario.

3.1 Participants

In the simulation mode, we employ the calibration data obtained from eleven healthy volunteers (two females, nine males), aged between 24 and 38 years old. The data is collected according to the IRB-approved protocol (IRB130107). Specifically, calibration data with different ITI values were collected in one

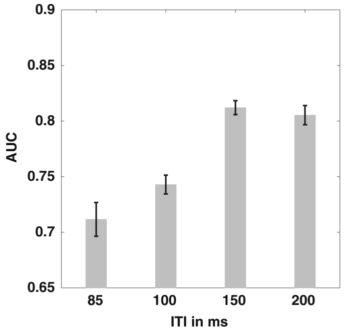


Fig. 3. The bar chart with error bars for average AUC values of classifier when trained one the data collected at each ITI.eps

Table 1. Hypothesis testing results between AUC values for different ITIs. The null hypothesis is that the expected AUC difference of the two considered ITIs is zero. Here we used $\alpha = 0.05$.

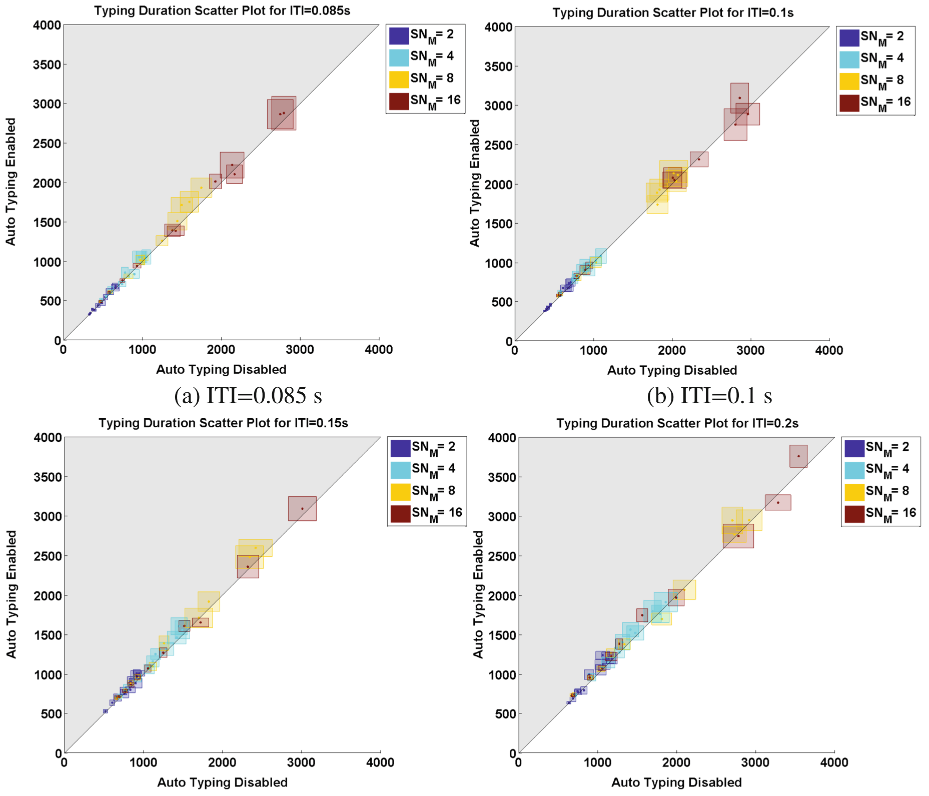
Paired t-test results between different ITI values	
ITI_1 v.s. ITI_2	P-values
85 v.s. 100 ms	0.157
85 v.s. 150 ms	0.001
85 v.s. 200 ms	0.009
100 v.s. 150 ms	0.0008
100 v.s. 200 ms	0.027
150 v.s. 200 ms	0.693

session from each user. To reduce the effect of fatigue and tiredness in group based analysis, we quasi-randomized the ITI values over the four calibration tasks among all users such that, the number of tasks with a specific ITI value for a particular order index are kept equal.

3.2 Data Analysis and Results

Area under the receiver operating characteristics curve (AUC) values, an indication of the classification accuracy, are calculated for every calibration data set using 10-fold cross validation. A paired t-test is run over the AUC values for different ITI conditions. The test results are illustrated in Table 1. This table shows that the classification accuracy significantly depends on the ITI. Moreover, as shown in Fig. 3, the average AUC is higher when the ITI=150ms suggesting that in average we expect to observe a better typing performance from the system when ITI=150ms.

The total typing durations (TTDs), performance metric computed by the simulation mode, for different users are summarized in Fig. 4. More specifically, sub-figure 4a,b,c and d represent the scatter plot of TTDs with (y-axis) and without (x-axis) auto-typing capability of the system at different ITI=85, 100, 150 and 200 ms. Different colors correspond to different SN_{max}. Every data point represents the mean and the variance of TTD, as a dot and a rectangle around that dot, respectively. The TTD variance for $SN_{min} = 0$ (auto-typing enabled) defines the vertical length of the rectangle while the horizontal length represents the variance when $SN_{min} = 1$ (auto-typing disabled). According to these plots, regardless of SN_{max}, auto-typing is not helping the users to type faster especially when the AUC is low (see the most top right data points of every color at each sub-figure). Moreover, in general among all SN_{max}, the TTD values are lower for ITI= 100 ms and ITI= 150 ms while most of the data points demonstrate values less than 2000 s at ITI= 150 ms.



(a) ITI=0.085 s

(b) ITI=0.1 s

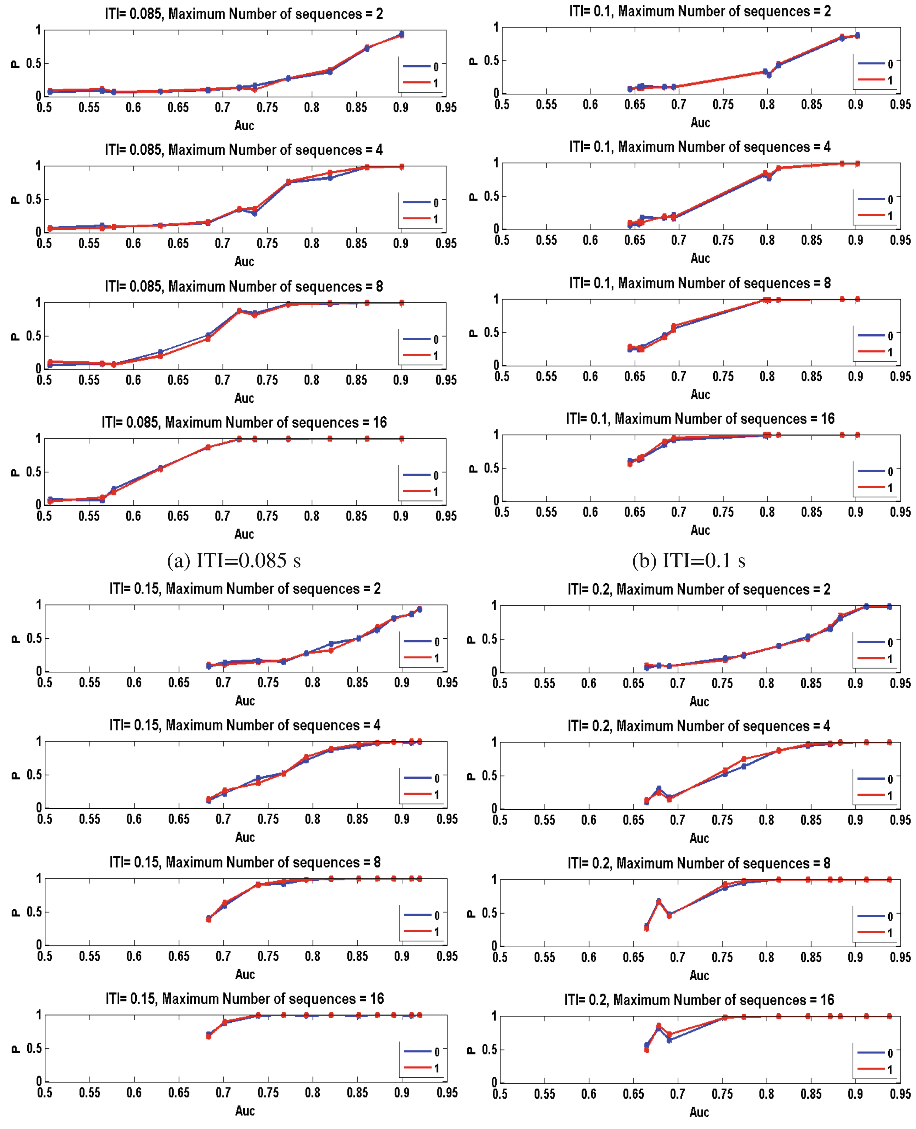
(c) ITI=0.15 s

(d) ITI=0.2 s

Fig. 4. Total typing duration (TTD) scatter plot. X-axis represents the total typing duration when the auto-typing is allowed and Y-axis corresponds to the TTD when auto-typing is disabled. Each point in these plots shows the mean value, and the horizontal and vertical skewness of the box around each point visualizes the variance of the results from 20 Monte-Carlo simulations.

Moreover, the simulation mode of the RSVPKeyboard$^{\text{TM}}$ measures the probability of phrase completion (PPC) as an accuracy measure of the simulated typing tasks. The effects of SN_{max} and ITI on PPC are demonstrated in Fig. 5. Different sub-figure 5a,b,c and d, illustrate the PPC as a function of AUC for different ITI values. In every figure, blue and red curves are used when auto-typing is enabled ($SN_{min} = 0$) or disabled ($SN_{min} = 1$), respectively. From these figures, we observe that increasing the SN_{max} improves the PPC at every ITI especially for lower AUC values. However, the comparison of the blue and red curves does not show any significant gain in PPC when the auto-typing is enabled. Also, as expected from Table 1, in average $ITI = 150$ms has in average higher PPC as shown in sub-figure 5c.

In conclusion, as all results suggest, for RSVPKeyboard$^{\text{TM}}$under the current design assumptions, not only auto-typing does not offer any improvement in typing



(a) ITI=0.085 s

(b) ITI=0.1 s

(c) ITI=0.15 s

(d) ITI=0.2 s

Fig. 5. Probability of phrase completion v.s. AUC. The blue curves correspond to the simulation results when auto-typing is enabled and the red curves represent the performance when auto-typing is disabled.

performance, but it also may decrease the typing speed. Furthermore, the combination of ITI= 150 ms, $SN_{max} = 8$ and $SN_{min} = 1$ provides the best performance of the system in terms of AUC, PPC and TTD for most of the users who participated in these experiments.

4 Discussion and Future Work

In this manuscript, we analyzed the effect of variations in inter-trial interval, maximum number of sequences and minimum number of sequences in an epoch on the typing performance of the RSVPKeyboard^{TM}, a language model assisted non-invasive EEG based letter by letter typing interface. Probability of phrase completion and total typing durations were obtained from 20-Monte-Carlo copy phrase tasks using the simulation mode of the typing interface over the calibration date collected from 11 healthy users. We observed that under the current design assumptions, the system does not benefit from auto-typing. In our future work, we will consider different graphical models to design joint inference engines that can enable the typing interface to benefit not only from auto-typing, but also from error related negativity for auto-correction.

References

1. Akcakaya, M., Peters, B., Moghadamfalahi, M., Mooney, A., Orhan, U., Oken, B., Erdogmus, D., Fried-Oken, M.: Noninvasive brain computer interfaces for augmentative and alternative communication. IEEE Rev. Biomed. Eng. **7**(1), 31–49 (2014)
2. Halder, S., Furdea, A., Varkuti, B., Sitaram, R., Rosenstiel, W., Birbaumer, N., Kübler, A.: Auditory standard oddball and visual p300 brain-computer interface performance. Int. J. Bioelectromag. **13**(1), 5–6 (2011)
3. van der Waal, M., Severens, M., Geuze, J., Desain, P.: Introducing the tactile speller: an erp-based brain-computer interface for communication. J. Neural Eng. **9**(4), 045002 (2012)
4. Farwell, L.A., Donchin, E.: Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. Electroencephalogr. Clin. Neurophysiol. **70**(6), 510–523 (1988)
5. Acqualagna, L., Blankertz, B.: Gaze-independent BCI-spelling using rapid serial visual presentation (RSVP). Clin. Neurophysiol. **124**(5), 901–908 (2013)
6. Acqualagna, L., Treder, M.S., Schreuder, M., Blankertz, B.: A novel brain-computer interface based on the rapid serial visual presentation paradigm. In: Proceedings of Conference on IEEE Engineering in Medicine and Biological Society, vol. 1, pp. 2686–2689 (2010)
7. Orhan, U., Erdogmus, D., Roark, B., Oken, B., Purwar, S., Hild, K., Fowler, A., Fried-Oken, M.: Improved accuracy using recursive bayesian estimation based language model fusion in erp-based bci typing systems. In: Annual International Conference of the IEEE on Engineering in Medicine and Biology Society (EMBC), vol. 2012, pp. 2497–2500. IEEE (2012)
8. Orhan, U., Erdogmus, D., Roark, B., Oken, B., Fried-Oken, M.: Offline analysis of context contribution to ERP-based typing BCI performance. J. Neural Eng. **10**(6), 066003 (2013)
9. Orhan, U., Hild, K. E., Erdogmus, D., Roark, B., Oken, B., Fried-Oken, M.: Rsvp keyboard: An eeg based typing interface. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 645–648. IEEE (2012)
10. Orhan, U., Akcakaya, M., Erdogmus, D., Roark, B., Moghadamfalahi, M., Fried-Oken, M.: Comparison of adaptive symbol presentation methods for rsvp keyboard

11. Sutton, S., Braren, M., Zubin, J., John, E.: Evoked-potential correlates of stimulus uncertainty. Science **150**(3700), 1187–1188 (1965)
12. Friedman, J.H.: Regularized discriminant analysis. J. Am. Stat. Assoc. **84**(405), 165–175 (1989)
13. Silverman, B.W.: Density Estimation for Statistics and Data Analysis, vol. 26. CRC Press, Boca Raton (1986)
14. Roark, B., Villiers, J. D., Gibbons, C., Fried-Oken, M.: Scanning methods and language modeling for binary switch typing. In: Proceedings of the NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies. Association for Computational Linguistics, pp. 28–36 (2010)