



# NIH Public Access

## Author Manuscript

*Conf Proc IEEE Eng Med Biol Soc.* Author manuscript; available in PMC 2013 September 17.

Published in final edited form as:

*Conf Proc IEEE Eng Med Biol Soc.* 2011 ; 2011: 5774–5777. doi:10.1109/IEMBS.2011.6091429.

## Fusion with Language Models Improves Spelling Accuracy for ERP-based Brain Computer Interface Spellers

**Umut Orhan<sup>1</sup>, Deniz Erdogmus<sup>1</sup>, Brian Roark<sup>2</sup>, Shalini Purwar<sup>1</sup>, Kenneth E. Hild II<sup>2</sup>, Barry Oken<sup>2</sup>, Hooman Nezamfar<sup>1</sup>, and Melanie Fried-Oken<sup>2</sup>**

<sup>1</sup>Cognitive Systems Laboratory, Northeastern University, Boston, MA, USA

<sup>2</sup>Oregon Health and Science University, Portland, OR, USA

### Abstract

Event related potentials (ERP) corresponding to a stimulus in electroencephalography (EEG) can be used to detect the intent of a person for brain computer interfaces (BCI). This paradigm is widely utilized to build letter-by-letter text input systems using BCI. Nevertheless using a BCI-typewriter depending only on EEG responses will not be sufficiently accurate for single-trial operation in general, and existing systems utilize many-trial schemes to achieve accuracy at the cost of speed. Hence incorporation of a language model based prior or additional evidence is vital to improve accuracy and speed. In this paper, we study the effects of Bayesian fusion of an n-gram language model with a regularized discriminant analysis ERP detector for EEG-based BCIs. The letter classification accuracies are rigorously evaluated for varying language model orders as well as number of ERP-inducing trials. The results demonstrate that the language models contribute significantly to letter classification accuracy. Specifically, we find that a BCI-speller supported by a 4-gram language model may achieve the same performance using 3-trial ERP classification for the initial letters of the words and using single trial ERP classification for the subsequent ones. Overall, fusion of evidence from EEG and language models yields a significant opportunity to increase the word rate of a BCI based typing system.

### Index Terms

Brain computer interfaces; Bayesian fusion; Language model; Event related potential

### I. Introduction

There exist a considerable number of people with severe motor and speech disabilities. Brain computer interfaces (BCI) are a potential technology to create a novel communication environment for this population, especially persons with completely paralysed voluntary muscles [1], [2]. One possible application of BCI is typing systems; specifically, those BCI systems that use electroencephalography (EEG) have been increasingly studied in the recent decades to enable the selection of letters for expressive language generation [1], [2], [3]. However the use of noninvasive techniques on a letter-by-letter system lacks efficiency due to low signal to noise ratio and variability of background brain activity. Therefore current BCI typing system suffer from low symbol rates and researchers have turned to various hierarchical symbol trees to achieve system speedups [3], [4], [5]. Slow throughput greatly diminishes the practical usability of such systems. Incorporation of a language model, which predicts the next letter using the previous letters, into the decision-making process can greatly affect the performance of these systems by improving the accuracy and speed. If the

symbol decisions done using only the EEG evidence are not accurate enough, the usage of a text prediction without improving the decision-making is not feasible. Therefore we propose the usage of the language model to improve symbol selection accuracy and this paper investigates the effect of its use in letter classification (target vs. non-target).

As opposed to the matrix layout of the popular P300-Speller [1] or the hexagonal two-level hierarchy of the Berlin BCI [3], our approach, which we refer to as RSVP Keyboard, utilizes another well-established paradigm: rapid serial visual presentation (RSVP) [6], [7]. This paradigm relies on presenting one stimulus at a time, each subsequent stimulus replacing the previous one on the screen, while the subject tries to perform mental target matching between the intended symbol and the sequence, which is presented at relatively high speeds. EEG responses corresponding to the visual stimuli are classified using regularized discriminant analysis (RDA) applied to stimulus-locked temporal features from multiple channels.

In this paper, we investigate the fusion of language model scores in an offline/simulated manner with the EEG classification scores, by randomly sampling contexts of each letter in a large text corpus. For each sampled context, we derive the language model probabilities of all 26 letters given the context, and these language model probabilities and the EEG classification scores are fused using a Bayesian approach, assuming that these two pieces of evidence are conditionally independent given class labels – a reasonable assumption. We present a performance analysis that compares different scenarios with varying language model orders and numbers of visual presentation sequences used in EEG classification. The results are very promising.

## II. RSVP Based BCI and ERP Classification

RSVP is an experimental psychophysics technique in which visual stimulus sequences are displayed on a screen over time on a fixed focal area and in rapid succession. The Matrix-P300-Speller [1] used by Wadsworth and Graz groups (especially G.tec) opts for a spatially distributed presentation of possible symbols, highlighting them in different orders and combinations to elicit P300 responses. Berlin BCI's recent variation utilizes a 2-layer tree structure [3] where the subject chooses among six units (symbols or sets of these) where the options are laid out on the screen while the subject focuses on a central focal area that uses an RSVP-like paradigm to elicit P300 responses. In contrast, our approach is to distribute the stimuli temporally and present one symbol at a time using RSVP and seek a binary answer to find the desired letter in a right-branching tree. The latter method has the advantage of not requiring the user to look at different areas of the screen.

In the current study, which is an offline analysis, our RSVP paradigm utilizes stimulus sequences consisting of letters in the English alphabet presented sequentially with random ordering where the user is expected to show positive intent for only one predesignated letter for each epoch (see details below). When the user sees the predesignated infrequent (1 in 26) target, the brain generates an event related potential (ERP) in the EEG; the most prominent component of this ERP is the P300 wave, which is a positive deflection in the scalp voltage primarily in frontal areas

$$P(W_i|W_{i-1}, W_{i-2}, \dots, W_{i-n+1}) = \frac{P(W_i, W_{i-1}, \dots, W_{i-n+1})}{P(W_{i-1}, \dots, W_{i-n+1})}, \quad (1)$$

$$Q = P(c_{w_i} = c | \delta_{\text{RDA}}(\mathbf{X}_{w_i,1}), \delta_{\text{RDA}}(\mathbf{X}_{w_i,2}), \dots, \delta_{\text{RDA}}(\mathbf{X}_{w_i,N_s}), W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1}) \quad (2)$$

$$= \frac{P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,1}), \delta_{\text{RDA}}(\mathbf{X}_{w_i,2}), \dots, \delta_{\text{RDA}}(\mathbf{X}_{w_i,N_s}), W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1} | c_{w_i} = c) P(c_{w_i} = c)}{P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,1}), \delta_{\text{RDA}}(\mathbf{X}_{w_i,2}), \dots, \delta_{\text{RDA}}(\mathbf{X}_{w_i,N_s}), W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1})} \quad (3)$$

$$= \frac{P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,1}), \delta_{\text{RDA}}(\mathbf{X}_{w_i,2}), \dots, \delta_{\text{RDA}}(\mathbf{X}_{w_i,N_s}) | c_{w_i} = c) P(W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1} | c_{w_i} = c) P(c_{w_i} = c)}{P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,1}), \delta_{\text{RDA}}(\mathbf{X}_{w_i,2}), \dots, \delta_{\text{RDA}}(\mathbf{X}_{w_i,N_s}), W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1})} \quad (4)$$

$$= \frac{(\prod_{n_s=1}^{N_s} P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,n_s}) | c_{w_i} = c)) P(c_{w_i} = c | W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1}) P(W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1})}{P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,1}), \delta_{\text{RDA}}(\mathbf{X}_{w_i,2}), \dots, \delta_{\text{RDA}}(\mathbf{X}_{w_i,N_s}), W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1})} \quad (5)$$

$$L = \frac{(\prod_{n_s=1}^{N_s} P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,n_s}) | c_{w_i} = 1)) P(c_{w_i} = 1 | W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1})}{(\prod_{n_s=1}^{N_s} P(\delta_{\text{RDA}}(\mathbf{X}_{w_i,n_s}) | c_{w_i} = 0)) P(c_{w_i} = 0 | W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1})} \quad (6)$$

and that generally occurs with a latency over 300 ms. This natural response of the brain to the event of visual stimulus matching the rare sought target allows us to make binary decisions about user's intent.

The intent detection problem becomes a signal classification problem when the EEG signals are windowed in a stimulus-time-locked manner over a duration with sufficient length – in this case 500ms. As a consequence, the signals acquired from each EEG channel will be incorporated and classified to determine the class label: ERP or non-ERP. The preprocessing steps used before classification are as follows. For each channel, the time windowed EEG signals are filtered by a bandpass filter; temporal feature vectors containing filtered-windowed signals from each channel are subjected to a linear dimension reduction using the vector covariances estimated over training samples and eliminating zero-variance directions (in practice using Principal Component Analysis). Afterwards the data vectors obtained for each channel are concatenated to create a data vector corresponding to the stimuli. This process amounts to a channel-specific energy-preserving orthogonal projection of raw temporal features. Regularized Discriminant Analysis (RDA) [8] is used to determine a classification discriminant score for each stimulus indicating whether it is a response to a target letter or not; this score is used in conjunction with a language model to make the final Bayesian decision on the class label of each letter.

RDA is a modified quadratic discriminant analysis (QDA) model. If each class is assumed to have multivariate normal distribution and classification is made according to the comparison of posterior distributions of the classes, the optimal Bayes classifier resides within the QDA model family. Under this assumption, QDA depends on the inverse of the class covariance matrices, which are to be estimated from training data, hence for small sample sizes in high dimensional problems, singularities of these matrices are problematic. RDA applies regularization and shrinkage procedures to the class covariance matrix estimates to eliminate the singularity problem. The shrinkage procedure makes the class covariances closer to the overall data covariance, and therefore to each other, thus making the quadratic boundary closer to a linear one. Shrinkage is applied as

$$\widehat{\sum}_c(\lambda) = (1 - \lambda) \widehat{\sum}_c + \lambda \widehat{\sum}, \quad (7)$$

where  $\lambda$  is the shrinkage parameter;  $\widehat{\Sigma}_c$  is the class covariance matrix estimated for class  $c \in \{0,1\}$  with  $c=0$  for non-target class and  $c=1$  for target class;  $\widehat{\Sigma}$  is the weighted average of class covariance matrices. Regularization is administered as

$$\widehat{\sum}_c(\lambda, \gamma) = (1 - \gamma) \widehat{\sum}_c(\lambda) + \frac{\gamma}{d} \text{tr}[\widehat{\sum}_c(\lambda)] \mathbf{I}, \quad (8)$$

where  $\gamma$  is the regularization parameter,  $\text{tr}[\cdot]$  is the trace function and  $d$  is the dimension of the data vector.

After carrying out the regularization and shrinkage on the estimation covariance matrices, the Bayesian classification rule [9] is defined as the comparison of the log-of-the-posterior-ratio using the posterior probability distributions with a threshold, which can incorporate the relative risks or costs of making an error for each class. The corresponding log-of-the-posterior-ratio is given by

$$\delta_{\text{RDA}}(\mathbf{X}) = \log \frac{f_{\mathcal{N}}(\mathbf{X}; \widehat{\mu}_1, \widehat{\Sigma}_1(\lambda, \gamma)) \widehat{\pi}_1}{f_{\mathcal{N}}(\mathbf{X}; \widehat{\mu}_0, \widehat{\Sigma}_0(\lambda, \gamma)) \widehat{\pi}_0} \quad (9)$$

Where  $\mu_c$ ,  $\pi_c$  are estimates of class means and priors respectively;  $\mathbf{X}$  is the data vector to be classified and  $f_{\mathcal{N}}(\mathbf{X}; \mu, \Sigma)$  is the pdf of a multivariate normal distribution.

The letter candidates, which contain all possible selectors, can be shown multiple times to achieve a higher classification accuracy in EEG-scores by making use of independent visual stimulus trial responses, as is commonly the case in EEG-based spellers<sup>1</sup>. We define a sequence to be a randomly ordered set of all letters shown as stimuli. Since the randomness of the target stimulus position in any given sequence is key to eliciting an ERP, a random permutation of the letters is used for each sequence in our experiments. Thereafter all or some of the sequences can be used to classify if a letter is target or non-target, depending on the operational mode of the ERP classifier, that is whether it is using a single-trial, 2-trial, or 3-trial approach.

Although the stimuli were presented in random order, we can simulate conditions when a language model would be operative by randomly sampling instances of each target letter in a large text corpus, and combining the language model prediction from the sampled context with the classifier. In the next section, we present our language modeling methods, along with details of our corpus sampling procedure.

### III. 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 [10]. Typically, the prefix string (what has already been typed) is used to predict the next symbol(s) to be typed. The next letters to be typed become highly predictable in certain contexts, particularly word-internally. In applications where text

---

<sup>1</sup>The typical number of repetitions of visual stimuli is usually 8 or 16, although G.tec claims one subject is able to achieve reliable operation with 2-trials (verbal communication)

generation/typing speed is very slow, the impact of language modeling can become much more significant. BCI-spellers, including the RSVP Keyboard paradigm presented here, can be extremely low-speed letter-by-letter writing systems, and thus can greatly benefit from the incorporation of probabilistic letter predictions from an accurate language model during the writing process.

The language model used in this paper is based on the  $n$ -gram sequence modeling paradigm, very widely used in all of the application areas mentioned above. It estimates the conditional probability of any letter in a sequence given  $n - 1$  previous letters using a Markov model of order  $n - 1$ . Let  $W$  be a sequence of letters where  $W_i$  is the  $i$ th letter in the sequence. For an  $n$ -gram model the estimate of the conditional probability of the letter  $W_i$  is obtained from (1), where the joint probabilities are estimated by regularized relative frequency estimation from a large text corpus. If the language model order is 1, then  $P(W_i)$  is equal to the context-free letter occurrence probabilities in the English language, which is not dependent on the previous letters. If the language model order is 0, then language modeling has no effect on the decision process, since in this case  $W_i$  is assumed to be drawn from a uniform distribution over the alphabet.

For the current study, all n-gram language models were estimated from a one million sentence (210M character) sample of the NY Times portion of the English Gigaword corpus. Corpus normalization and smoothing methods were as described in [10]. Most importantly for this work, the corpus was case normalized, and we used Witten-Bell smoothing for regularization. For each letter, 1000 contexts were randomly sampled (without replacement) from a separate 1M sentence subset of the same corpus.

#### IV. Fusion of Language Model Probabilities and ERP Classifier Scores

The prediction of the current letter obtained by the language model using the previously typed letters can be used to improve the performance of the ERP classifier explained in Section II. For each letter to be written, an epoch of selections is going to be shown in BCI. Let  $N_S$  be the number of sequences per epoch (i.e. number of trials ERP classifier scores generated) used to classify stimulus responses corresponding to each letter in a particular epoch and  $\delta_{RDA}(\mathbf{X}_{W_i, n_s})$  be the corresponding posterior ratio scores obtained from RDA for letter  $W_i$ , where  $i - 1$  letters are already written, and sequence  $n_s$ , where  $n_s \in \{1, 2, \dots, N_S\}$ . Then the posterior probability of letter  $W_i$  to be in class  $c$  given the classification scores for letter  $W_i$  trials in each sequence and the previous letters is given in (2), where  $c_{W_i}$  is the candidate class label of letter  $W_i$ ,  $W'_j$  is the  $j$ th letter previously written and  $n_{LM}$  is the order of the language model. Using Bayes' Theorem on (2), we obtain (3). If we assume that the scores obtained from RDA<sup>2</sup> for the stimuli corresponding to the current letter and previously written letters are conditionally independent given class label, i.e.

$\delta_{RDA}(\mathbf{X}_{W_i, 1}), \delta_{RDA}(\mathbf{X}_{W_i, 2}), \dots, \delta_{RDA}(\mathbf{X}_{W_i, N_S}) \perp W'_{i-1}, W'_{i-2}, \dots, W'_{i-n_{LM}+1} | c$ , we obtain (4). Using Bayes' Theorem on (4) and assuming the conditional independence of the scores corresponding to EEG responses for different trials of the same letter in different sequences, we obtain (5). Hence the ratio of the posterior probabilities becomes (6), which can be compared to a risk-based threshold,  $\tau$ , to decide if the letter is a target or not. In our current implementation,  $P(\delta_{RDA}(\mathbf{X}_{W_i, n_s}) | c_{W_i} = c)$  is estimated using kernel density estimation on training data, using a Gaussian kernel whose bandwidth is selected using Silverman's rule of thumb that assumes the underlying density has the same curvature as a matching normal distribution [11]. The classification rule, which makes decisions using the ratio of posteriors

---

<sup>2</sup>The RDA scores are used as one dimensional EEG features for fusion purposes.

that incorporated information from ERP classifiers and language model predictions, finally comes forth as:

$$\delta(L) = \begin{cases} 1 & \text{if } L \geq \tau; \\ 0 & \text{if } L < \tau. \end{cases} \quad (10)$$

## V. Experiments and Results

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

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

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

The language model 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 language model training data) for testing purposes. For each letter sample we simulate the fusion of EEG responses and the language model in the following way: (i) each sample is assumed to be the target letter of a typing process using BCI; (ii) the predecessor letters of the target letter for each epoch are taken from the corpus to calculate the letter probabilities of the n-gram language models for each letter in the alphabet<sup>3</sup>; (iii) under the assumption of independence of EEG responses with the previous letters selected, for each epoch, the EEG responses for every letter is converted to EEG classifier scores; (iv) the matching model probabilities for each letter are obtained from the language model; (v) and the fusion of ERP classifier scores and language models 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.

---

<sup>3</sup>Since subjects only focus to 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.

Fusion results were obtained for n-gram model orders  $n = 0, 1, 4$ , and  $8$ . The EEG scores were assumed to have been evaluated for  $N_S = 1, 2$ , and  $3$  sequences (to evaluate the contribution of multi-trial information) to decide if a letter under evaluation was a desired target letter or not. In the results, only EEG data from the first  $N_S$  sequences of the epoch was used for classification for each selected sequence count. Receiver operating characteristics (ROC) curves are obtained using the decision rule given in 10 for different orders of the language model, for different number of sequences used and for different positions of the sample target letter in the corresponding word from the corpus. In Table I the area under the ROC curves are compared where each entry contains the pair of minimum and maximum areas over the sessions. In Table II, Table III, and Table IV the correct detection rates are given for the false positive rates of  $1\%$ ,  $5\%$ , and  $10\%$ , respectively.

## VI. Discussion

Our analysis supports the hypothesis that using a language model to support ERP classification can improve BCI-speller performance. As the number of stimulus repetitions and as the model order for the language model increase, the performance of the letter classification as target or nontarget improves as expected. A 0-gram language model (EEG-only) performs worst and the language model makes significant contribution in single-trial decision-making. The language model contributes more to letters that appear word-internally than in word-initial position. Large model orders for the language model can help significantly after the first letter of a word and must be investigated further. The language model order is not as influential for the first letters in a word while number of stimulus repetitions is; consequently, the results suggest that for first letters of words the BCI system could switch to multi-trial mode, while for subsequent letters, single-trial EEG evaluation with high-order language model could be beneficial. Reduction in the number of repetitions is a direct multiplier factor for reduction in time to type a given length text.

Additionally, this fusion approach is also implemented in real time and the preliminary results seem very promising. Online comparative analysis and further experiments will be done as a future work.

## Acknowledgments

This work is supported by NSF under grants ECCS0929576, ECCS0934506, IIS0934509, IIS0914808, BCS1027724 and by NIH under grant 1R01DC009834-01. The opinions presented here are those of the authors and do not necessarily reflect the opinions of the funding agencies.

## References

1. Krusienski DJ, Sellers EW, McFarland DJ, Vaughan TM, Wolpaw JR. Toward enhanced P300 speller performance. *Journal of neuroscience methods*. 2008; 167, no. 1:15–21. [PubMed: 17822777]
2. Pfurtscheller G, Neuper C, Guger C, Harkam W, Ramoser H, Schlogl A, Obermaier B, Pregenzer M. Current trends in Graz brain-computer interface (BCI) research. *IEEE Transactions on Rehabilitation Engineering*. 2000; 8, no. 2:216–219. [PubMed: 10896192]
3. Treder MS, Blankertz B. (C) overt attention and visual speller design in an ERP-based brain-computer interface. *Behavioral and Brain Functions*. 2010; 6, no. 1:28. [PubMed: 20509913]
4. Serby H, Yom-Tov E, Inbar GF. An improved P300-based brain-computer interface. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*. 2005; 13, no. 1:89–98.
5. Wolpaw JR, Birbaumer N, McFarland DJ, Pfurtscheller G, Vaughan TM. Brain-computer interfaces for communication and control. *Clinical neurophysiology*. 2002; 113, no. 6:767–791. [PubMed: 12048038]

6. Mathan, S.; Erdogmus, D.; Huang, Y.; Pavel, M.; Ververs, P.; Carciofini, J.; Dorneich, M.; Whitlow, S. CHI'08 extended abstracts on Human factors in computing systems. ACM; 2008. Rapid image analysis using neural signals; p. 3309-3314.
7. Mathan S, Ververs P, Dorneich M, Whitlow S, Carciofini J, Erdogmus D, Pavel M, Huang C, Lan T, Adami A. Neurotechnology for Image Analysis: Searching for Needles in Haystacks Efficiently. Augmented Cognition: Past, Present, and Future. 2006
8. Friedman JH. Regularized discriminant analysis. *Journal of the American statistical association*. 1989; 84, no. 405:165–175.
9. Duda, RO.; Hart, PE.; Stork, DG. Pattern classification. Citeseer; 2001.
10. Roark, Brian; de Villiers, Jacques; Gibbons, Christopher; Fried-Oken, Melanie. Scanning methods and language modeling for binary switch typing; Proceedings of the NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies; 2010; p. 28-36.
11. Silverman, BW. Density estimation for statistics and data analysis. Chapman & Hall/CRC; 1998.

**TABLE I**

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

		<b>1 sequence</b>	<b>2 sequences</b>	<b>3 sequences</b>
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	Word-initial	(0.892, 0.941)	(0.954, 0.983)	(0.977, 0.991)
	Word-internal	(0.975, 0.983)	(0.985, 0.992)	(0.991, 0.997)
8-gram	Word-initial	(0.905, 0.945)	(0.960, 0.984)	(0.979, 0.992)
	Word-internal	(0.991, 0.993)	(0.995, 0.997)	(0.995, 0.998)

**TABLE II**

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

		<b>1 sequence</b>	<b>2 sequences</b>	<b>3 sequences</b>
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	Word-initial	(0.263, 0.416)	(0.434, 0.774)	(0.621, 0.810)
	Word-internal	(0.597, 0.684)	(0.748, 0.849)	(0.848, 0.927)
8-gram	Word-initial	(0.294, 0.448)	(0.465, 0.782)	(0.647, 0.835)
	Word-internal	(0.810, 0.854)	(0.886, 0.932)	(0.936, 0.972)

**TABLE III**

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

		<b>1 sequence</b>	<b>2 sequences</b>	<b>3 sequences</b>
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	Word-initial	(0.606, 0.688)	(0.740, 0.884)	(0.886, 0.971)
	Word-internal	(0.842, 0.899)	(0.912, 0.966)	(0.960, 0.989)
8-gram	Word-initial	(0.614, 0.716)	(0.766, 0.905)	(0.899, 0.971)
	Word-internal	(0.951, 0.971)	(0.972, 0.990)	(0.986, 0.996)

**TABLE IV**

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

		<b>1 sequence</b>	<b>2 sequences</b>	<b>3 sequences</b>
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	Word-initial	(0.692, 0.836)	(0.857, 0.961)	(0.948, 0.990)
	Word-internal	(0.933, 0.961)	(0.966, 0.991)	(0.983, 0.996)
8-gram	Word-initial	(0.729, 0.840)	(0.873, 0.964)	(0.950, 0.990)
	Word-internal	(0.983, 0.990)	(0.992, 0.997)	(0.995, 0.998)