A Probabilistic Graphical Model for Word-Level Language Modeling in P300 Spellers

Jaime F. Delgado Saa[1,3], Adriana de Pesters[2], Dennis McFarland[2] and Müjdat Çetin[1]

[1] Sabanci University, Signal Proc. & Info. Syst. Lab, Istanbul, Turkey
delgado@sabanciuniv.edu, mcetin@sabanciuniv.edu
[2] Wadsworth Center, Lab of Injury and Repair, Albany, NY, USA.
adrianad@wadsworth.org, mcfarlan@wadsworth.org
[3] Universidad del Norte, Robotics and Intelligent Systems Lab, Barranquilla , Colombia.
jadelgado@uninorte.edu.co

Abstract

Motivated by P300 spelling scenarios involving communication based on a limited vocabulary, we propose a probabilistic graphical model-based framework and an associated classification algorithm that uses learned statistical prior models of language at the level of words. Exploiting such high-level contextual information helps reduce the error rate of the speller. The proposed approach models all the variables in the P300 speller in a unified framework and has the capability to correct errors in previous letters in a word given the data for the current one. The structure of our model allows the use of efficient inference algorithms, which makes it possible to use this approach in real-time applications. Our experimental results demonstrate the advantages of the proposed method.

1 Introduction

Recently there has been growing interest in the incorporation of statistical language models into P300 spellers with the intention to reduce the error rate or to increase typing speed [1, 2, 3, 4, 5]. These approaches learn marginal and conditional probabilities of letters in a language based on some corpus and use that information in the form of prior models in a P300-based brain-computer interface (BCI) system. In particular, such priors models are combined with measured electroencephalography (EEG) data in a Bayesian framework to infer the letters typed by the subject. The probabilistic structure in most of this work can be described by hidden Markov models, which are one of the simplest forms of probabilistic graphical models. Work under this theme involves filtering [2, 4], as well as smoothing methods [5]. The latter type of method allows the EEG data for the current letter to affect the decision on a previous letter through the language model, and possibly correct an erroneous decision that would be reached in the absence of such data. In most of this body of work, first a conventional classifier for the P300 speller is utilized and then the scores of that classifier are turned into probabilities to be combined with the language model for Bayesian inference.

In this work, we propose taking one further step incorporating higher-level, in particular word-level, language models into P300 spellers. Our motivation comes from BCI applications that involve typing based on a limited vocabulary. In a particular context, if it is known that a user is likely to type words from a dictionary of, say, a few thousand words, that provides very valuable prior information that can potentially be exploited to improve the performance of the BCI system. Based on this perspective, we propose a discriminative graphical model-based framework for the P300 speller with a language model at the level of words. The proposed model integrates all the elements of the BCI system from the input brain signals to the spelled

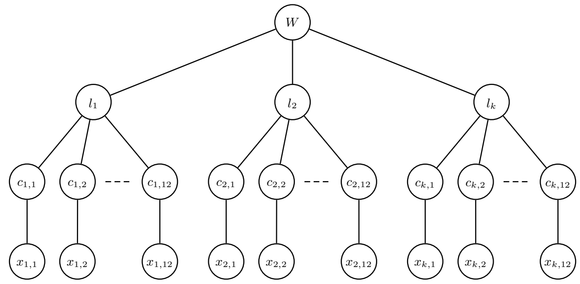

Figure 1: Proposed graphical model framework for the P300 speller.

word in a probabilistic framework in which classification and language modeling are integrated in a single model and the structure of the graph allows efficient inference methods making the system suitable for online applications. Results show that the proposed method provides significant improvement in the performance of the P300 speller by increasing the classification accuracy while reducing the number of flash repetitions.

2 Methods

Proposed Graphical Model: The proposed model is shown in Figure 1. In the bottom (first) layer, the variables $x_{i,j}$ represent the EEG signal recorded during the intensification of each row and column of the spelling matrix. The index i denotes the ordinality of the letter being spelled and the index j represents a row or column ($j = \{1, ..., 6\}$ for rows and $j = \{7, ..., 12\}$ for columns). The second layer contains a set of twelve variables $c_{i,j}$ indicating the presence or absence of the P300 potential for a particular flash. The third layer contains variables l_i representing the letter being spelled. The variables l_i are related to the variables $c_{i,j}$ in the same fashion as in traditional P300 speller systems: the presence of a P300 potential in a particular row-column pair encodes one letter. The fourth layer contains the variable w which can be any member of a particular subset of valid words in the English language. A learned probability mass function for this variable constitutes the language model in this work.

The distributions of all the variables in the model (w, $\mathbf{l} = \{l_1, ..., l_k\}$, $\mathbf{c} = \{c_{1,1:12}, ..., c_{k,1:12}\}$) given the observations ($\mathbf{x} = \{x_{1,1:12}, ..., x_{k,1:12}\}$) can be written as a product of factors over all the nodes and edges in the graph:

$$P(w, \boldsymbol{l}, \boldsymbol{c}|\boldsymbol{x}) = \frac{1}{Z}\Psi_4(w)\prod_i \left\{ \Psi_3(i, w, l_i) \prod_{j=1}^{12} \{\Psi_2(j, l_i, c_{i,j})\Psi_1(j, c_{i,j}, x_{i,j})\} \right\} \tag{1}$$

where Z is a normalization factor and Ψ_4, Ψ_3, Ψ_2 and Ψ_1 are potential functions related to nodes and edges. The potential functions are defined as follows:

$$\Psi_4(w) = e^{\theta_4 f_4(w)} \qquad \Psi_3(i, w, l_i) = e^{\theta_{3\,i} f_3(i,w,l_i)}$$
$$\Psi_2(j, l_i, c_{i,j}) = e^{\theta_{2\,j} f_2(j,l_i,c_{i,j})} \qquad \Psi_1(j, c_{i,j}, x_{i,j}) = e^{\sum_{m=1}^{d} \theta_{1\,j,m} f_{1\,m}(j,c_{i,j},x_{i,j\,m})} \tag{2}$$

where d is the dimensionality of the data. The parameter θ_4 is a vector of weights of length equal to the number of states of the node w (i.e., the number of words in the dictionary). The product $\theta_4 f_4(w)$ models a prior for the probability of a word in the language with the feature function $f_4(w) = \mathbf{1}_{\{w\}}$, where $\mathbf{1}_{\{w\}}$ is a vector of length equal to the number of words in the dictionary, with a single nonzero entry of value 1 at the location corresponding to the argument of f_4. The product $\theta_{3i} f_3(i, w, l_i)$ models a prior for the probability of a letter l_i appearing in the position i of a word with the feature function $f_3(i, w, l_i) = \mathbf{1}_{\{w(i), l_i\}}$. The product $\theta_{2j} f_2(j, l_i, c_{i,j})$ measures the compatibility between the binary random variable $c_{i,j}$ and the variable l_i with the feature function $f_2(j, l_i, c_{i,j}) = \mathbf{1}_{\{C(l_i, j) = c_{i,j}\}}$ where C is a code-book that maps the intersections of rows and columns in the spelling matrix to letters. The product $\theta_{1j,m} f_{1m}(j, c_{i,j}, x_{i,j_m})$ is a measure of the compatibility of the m-th element of the EEG signal $x_{i,j} \in R^d$ with the variable $c_{i,j}$. Here, we use the feature function $f_{1m}(j, c_{i,j}, x_{i,j_m}) = x_{i,j_m} \mathbf{1}_{\{c_{i,j}\}}$. Learning in the model corresponds to finding the set of parameters $\Theta = \{\theta_4, \theta_3, \theta_2, \theta_1\}$ that maximizes the log-likelihood of the conditional probability density function given in Equation 1. Given that the structure of the model does not involve loops, inference in the model can be performed using the belief propagation algorithm which can efficiently provide the posterior probabilities for the words: $P(w|\boldsymbol{l}, \boldsymbol{c}, \boldsymbol{x}) = \sum_{\boldsymbol{l}} \sum_{\boldsymbol{c}} P(w, \boldsymbol{l}, \boldsymbol{c}|\boldsymbol{x})$. We declare the word by maximizing the posterior density: $\overline{w} = \arg\max_w P(w|\boldsymbol{l}, \boldsymbol{c}, \boldsymbol{x})$. Note that the model allows computing other marginals of interest (e.g,. letters) as well.

Description of Experiments: Two kinds of experiments are reported. In the first experiment, 8 subjects are instructed to spell a number of words one by one. In this scenario (which we call screening) the number of letters in the words typed in the testing session is known. In the second experiment 7 subjects write multiple words in a continuous fashion, using the character "-" to indicate the end of a word. The EEG signals were recorded at a sample frequency of 240 Hz using a cap embedded with 64 electrodes according to the 10-20 standard. All electrodes were referenced to the right earlobe and grounded to the right mastoid. From the total set of electrodes a subset of 16 electrodes in positions F3, Fz, F4, FCz, C3, Cz, C4, CPz, P3, Pz, P4, PO7, PO8, O1, O2, Oz were selected, motivated by the study presented in [6]. In total each subject spelled 32 letters (9 words). Training and testing sessions were held on different days for the same subjects. Signal segments of 600ms following the intensification of each row or column were calculated and filtered between 0.5Hz and 8Hz using a zero-phase IIR filter of order 8.

"The parameters $\theta_4, \theta_{3i}, \theta_{2j}$ in Equation 2 are independent of the brain signals and can be learned prior to any EEG data collection. The parameters θ_4 are learned by calculating the relative frequency of each word in the dictionary of interest. The parameters θ_{3i} take a positive or negative value depending of the presence of absence of a letter in the i-th position of a word. Statistics about the language were calculated using a text corpus with 450,000,000 words. The dictionary was then built using the 5000 words with highest frequency. The parameters θ_2 represent a set of decoding vectors that map rows and columns to letters in the spelling matrix. The parameters θ_1 are the set of parameters that maximize the potential function $\Psi_1(j, c_{i,j}, x_{i,j})$ in Equation 2 for each class (i.e., P300 vs. not P300).

3 Results

Figure 2 shows the results for the screening scenario and for continuous decoding where a 3-gram method and a classifier based in Stepwise LDA have been used for comparison. The results show that the proposed method performs better than the other methods in terms of classification accuracy and requires fewer numbers of repetitions to achieve a particular level of accuracy. In order to verify the results of correct classification accuracy, a statistical test

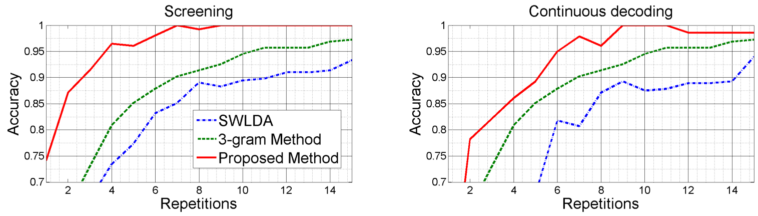

Figure 2: Average classification accuracy across subjects.

was performed. For the screening scenario, a repeated measures ANOVA on the performance results reveals significant difference ($F(2, 14), \epsilon = 0.56, p = 0.0041$) between the three compared methods. Using a post hoc Tukey-Kramer test, the proposed method performs significantly better ($p < 0.01$) than the 3-gram based method and than SWLDA. For the continuous decoding scenario the results are similar, the proposed method performs significantly better ($p < 0.01$) than the 3-gram based method and than SWLDA.

4 Conclusion

We present a probabilistic framework as well as an inference approach for P300-based spelling that exploits learned prior models of words. While language models at the level of letters have previously been proposed for BCI, word-level language modeling is new. The structure of the model we propose enables the use of efficient inference algorithms, making it possible to use our approach in real-time applications. While our approach can in principle be used with word prior models learned from any corpus, we expect it to be of special interest for applications involving the use of a limited vocabulary in a specific context.

Acknowledgments

This work was partially supported by Universidad del Norte (Colombia), the Scientific and Technological Research Council of Turkey under Grant 111E056, by Sabanci University under Grant IACF-11-00889, and by NIH grant EB00085605 (NIBIB).

References

[1] Tanja Mathis and Dennis Spohr. Corpus-Driven Enhancement of a BCI Spelling Component. In *RANLP'07: Int. Conf. on Recent Advances in Natural Language Processing*, September 2007.

[2] Umut Orhan et al. Offline analysis of context contribution to ERP-based typing BCI performance. *Journal of Neural Engineering*, 10(6), 2013.

[3] S.T. Ahi, H. Kambara, and Y. Koike. A Dictionary-Driven P300 Speller With a Modified Interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):6–14, 2011.

[4] William Speier et al. Natural language processing with dynamic classification improves P300 speller accuracy and bit rate. *Journal of Neural Engineering*, 9(1), 2012.

[5] C. Ulas and M. Cetin. Incorporation of a Language Model into a Brain Computer Interface based Speller through HMMs. In *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2013.

[6] D.J. Krusienski, E.W. Sellers, D.J. McFarland, T.M. Vaughan, and J.R. Wolpaw. Toward Enhanced P300 Speller Performance. *Journal of Neuroscience Methods*, 167(1):15 – 21, 2008.