A Study on Application of RB-ARQ Considering Probability of Occurrence and Transition Probability for P300 Speller

Eri Samizo, Tomohiro Yoshikawa, and Takeshi Furuhashi

Nagoya University, Nagoya 464-8603, Japan

Abstract. Brain-Computer Interfaces (BCIs) control a computer or a machine based on the information of the signal of human's brain. P300 speller is one of the BCI communication tools, which uses P300 as the feature quantity and allows users to select letters just by thinking. Because of the low signal-to-noise ratio of the P300, signal averaging is often performed to improve the spelling accuracy instead of the degradation of the spelling speed. In texts, there is variability in occurrence probabilities and transition probabilities between lettersDThis paper proposes P300 speller considering the occurrence probabilities and the transition probabilities as the prior probabilities in RB-ARQ. It shows that the spelling speed and then the Utility were improved by the proposed method comparing with the conventional method.

1 Introduction

Brain-Computer Interface is the system that controls a computer or a machine based on the information of signals from human's brain[1]. It is expected to be developed as a communication tool for seriously paralyzed patients like those with amyotrophic lateral sclerosis (ALS). Electroencephalogram (EEG) is most likely used for BCIs because it is noninvasive and inexpensive. P300 speller that is first introduced by Farwell et al. is one of the communication tools using P300 as a feature[2]. P300 is one of the event-related potential (ERP) and it is elicited when a stimulus that a user attends to is provided. A user can choose and input letters just by his/her thoughts using P300 speller. It generally uses a letter matrix interface with visual stimulus. Each row or column flashes in random order one by one for a certain times. While they are flashing, the user concentrates on the desired letter by counting how many times it flashes. Thereby, P300 is elicited when the row or column that contains the desired letter is flashed. Then the system discriminates the letter that includes P300 most likely as the target one.

However, signal-to-noise ratio of the P300 is small. Thus, averaging signals is needed[3][4], which improves the spelling accuracy instead of degrading the spelling speed. Practically, it is needed to input letters correctly in a short time to reduce user's burden. Conventionally, the number of flashing times, i.e., the number of stimuli, is fixed. To reduce the number of stimuli, Reliability-Based

D.D. Schmorrow and C.M. Fidopiastis (Eds.): AC/HCII 2013, LNAI 8027, pp. 727–733, 2013.

Automatic Repeat reQuest (RB-ARQ) has been proposed[5]. It is shown that RB-ARQ can reduce spelling speed with keeping spelling accuracy[6].

In RB-ARQ, the prior probability, the likelihood of each letter to be the target before the presentation of stimuli, is set equally for all letters. On the other hand, there is variability in occurrence probabilities and transition probabilities between letters in texts. In the area of understanding texts or voice recognition, the transition probabilities between letters are used for letter correction or the support of input and recognition[7][8].

In this paper, we propose a new P300 speller that considers the ocurrence probabilities and the transition probabilities between letters as the prior probability in RB-ARQ. The experiments are done by three subjects with Japanese interface of P300 speller and the result shows the improvement of spelling speed and then the Utility, which is the performance index of spelling considering accuracy and discrimination time at once, by the proposed method comparing with the conventional one.

2 Reliability-Based Automatic Repeat reQuest

RB-ARQ is a method that presents stimuli randomly and sets the number of stimuli dynamically based on the maximum posterior probability[5][6]. Suppose $\boldsymbol{x}_t$ denotes a feature vector from EEG data at time t, and let $X_T = \{\boldsymbol{x}_t | t = 1, 2, ..., T\}$ be a set of data at time T, the posterior probability at time T can be calculated as follows:

$$P(k|X_T) = \frac{P(k)\prod_t p(\boldsymbol{x}_t|k)}{\sum_{l\in K} P(l)\prod_t p(\boldsymbol{x}_t|l)} \tag{1}$$

In this equation, let K be a set of candidate letters and $k \in K$. And $P(k)$ is the prior probability that $\boldsymbol{x}$ belongs to label k before the stimulus presentation, and they are set equally. The posterior probability is obtained by multiplying the prior probability and likelihood. Maximum posterior probability at time T is defined as Eq.(2) using the posterior probability $P(k|X_T)$.

$$\lambda_T = \max_k P(k|X_T) \tag{2}$$

The maximum posterior probability is equivalent to the discrimination accuracy, which can be regarded as the reliability of data. λ is set as the threshold of reliability, and a user keeps thinking until λ_T becomes larger than λ.

3 Proposed Method

As mentioned above, the prior probability of RB-ARQ is set equally to every letter in the conventional method. This paper proposes a method to consider the occurrence probability and the transition probability of letters in text as the prior probability. Transition probability is the frequency of a letter in texts after the given preceding letter(s), and it is given by the occurrence rate of N-gram character in an enormous quantity of text data. N-gram is every contiguous sequence of n characters in a given text[9]. Therefore, the prior probability is defined as below with n=1,2,...

$$P(X_i) = \frac{N(X_i)}{\sum_{l \in K} N(X_i)} \qquad (n = 1) \qquad (3)$$

$$P(X_i | X_{i-n+1}^{i-1}) = \frac{N(X_{i-n+1}^{i})}{N(X_{i-n+1}^{i-1})} \qquad (n \geq 2) \qquad (4)$$

Let X_i^j be a part of string from ith letter to jth letter in the character string $X_1X_2,...X_M$. $P(X_i|X_{i-n+1}^{i-1})$ is the conditional probability that ith letter becomes X_i when a string from $\{i-(n-1)\}$th letter to $(i-1)$th letter is given. $N(X_i^j)$ denotes the occurrence frequency of a string from ith letter to jth letter. When n is 1, the prior probability is simply represents the probability of the occurrence of each letter, and this paper calls iteUni-gram.fWhen n is 2 or 3, they represent the transition probability between letters. This paper calls themeBi-gramfandeTri-gram,frespectively. Using these probabilities, it is expected to improve the performance of inputting text in RB-ARQ. It is thought that the time until the posterior probability exceeds the threshold λ becomes shorter, because the letters with high occurrence rate have high prior probability.

4 Experiment

4.1 Data Description and Preprocessing

This experiment used a recorded dataset which contained EEG data measured by three subjects (Sub A, Sub B and Sub C) performed the P300 speller. EEG data was recorded with sampling frequency of 1000Hz using Polymate AP216 (Digitex lab. co., ltd., Tokyo), from 5electrodes: Fz, Cz ,Pz, O1 and O2, referenced to the linked ears, A1 and A2 (Fig.1). The stimulus onset asynchrony (SOA) was 175 ms: each stimulus was presented for 100 ms with an inter-stimulus interval (ISI) of 75 ms. In this experiment, the 7-by-10 letter matrix interface containing Japanese characters shown in Fig.2 was employed. An input of one letter consisted of ten sequences, while one sequence contained 17 (10 rows and 7 columns) stimuli. Then, the EEG signals were down-sampled to 20Hz, 14 data points corresponding to 0s to 0.65s after each stimulus was extracted. The extracted data were classified using Linear Discriminant Analysis (LDA). 20 letters were utilized for the learning session.

In this interface, <> is used when the user wants to input small letters. For example, when the user wants to input <>, he/she needs to select <> before <>. And <BS> means backspace which deletes the preceding letter. The prior probability employed in the proposed method was calculated based on the web corpus of Japanese[10]. In the calculation of Eq.(3)(4), sonant marks and p-sounds were regarded as one character, and the prior probability of <> was calculated using the number of the appearance of small letters.

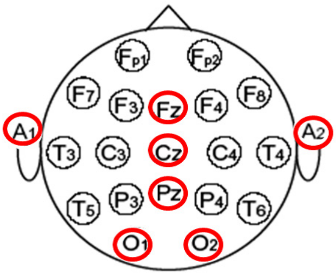

Fig. 1. Used electrodes

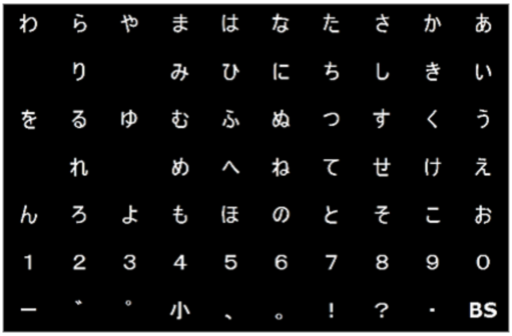

Fig. 2. User-interface

4.2 Experimental Settings and Performance Index

The experiment employed three long sentences, which had about 200 letters extracted from Web blog, essay and novel, respectively. They were inputted for 200 times in each test, and the results were averaged.

When non-target letter was inputted, that is, discrimination result was wrong, <BS> would be selected at the next target to input full sentence correctly. The threshold of RB-ARQ was set to 0.9, 0.95 and 0.99. We conducted the following two experiments.

Exp.1: We set the prior probability by Uni-gram, Bi-gram and Tri-gram and compared with the conventional method, the prior probability was set equally caled eEqual.'

Exp.2: We set the prior probability by Uni-gram, Bi-gram and Tri-gram, and when <BS> and a letter except for <BS> were repeated, we set every prior probability equal.

This paper employed the performance index for the comparison as the average of the discrimination accuracy, the number of stimuli and the discrimination (input) time per a letter. Each performance index is determined below.

$$Discrimination\ accuracy = \frac{\sharp\ of\ correct\ letters}{\sharp\ of\ inputted\ letters} \tag{5}$$

$$\sharp\ of\ stimuli\ per\ a\ letter = \frac{\sharp\ of\ all\ stimuli}{\sharp\ of\ inputted\ letters} \tag{6}$$

$$Discrimination\ time\ per\ a\ letter = \sharp\ of\ stimuli\ per\ letter \times SOA \tag{7}$$

This paper also uses gUtility[11]hdefined in Eq.(8) to evaluate the accuracy and the discrimination time at once.

$$U = \frac{(2P-1)\log_2(C-1)}{d} \tag{8}$$

where C is the number of classes (in this experiment, N=70, 10 rows 7 columns), P is the accuracy, and d is the discrimination time per a letter. Note that if P <0.5, U=0. Utility corresponds to the information transfer rate when the spelling is done perfectly by using <BS> that can delete incorrect characters. Thus, it is thought to be a practical performance measure for the P300 speller.

5 Result

Table 1 shows the discrimination accuracy, the number of stimuli and the discrimination time in Exp.1, and Table 2 shows those in Exp.2. Figure 3 shows the Utility in Exp.1, and Fig.4 shows that in Exp.2. In these figures, the values on the horizontal axis mean the thresholds of RB-ARQ and vertical axis shows the value of Utility.

Table 1. Performance indexes in Exp.1

	Threshold	Equal	Uni-gram	Bi-gram	Tri-gram
Accuracy	0.9	0.81	0.794	0.818	0.65
	0.95	0.869	0.87	0.891	0.819
	0.99	0.918	0.933	0.95	0.939
♯ of Stimuli	0.9	78.9	68	55.2	41.9
	0.95	90.6	80	66.2	57.7
	0.99	110.8	100.6	85.4	83
Time[s]	0.9	13.8	11.9	9.7	7.3
	0.95	15.9	14	11.6	10.1
	0.99	19.4	17.6	14.9	14.5

Table 1 shows that the accuracy of Uni-gram and Bi-gram were almost equal or better than the conventional method (Equal), while, the number of stimuli of Uni-gram and Bi-gram were smaller than Equal. On the other hand, the number of stimuli of Tri-gram at threshold 0.9 was also reduced, however, the accuracy decreased at the same time. Especially in Tri-gram, the prior probability widely varied depending on the next selectable letters comparing with other methods. Threfore, the discrimination time was largely decreased when the letter with high prior probability was selected as the target letter. On the other hand, when the letter with low prior probability was chosen as the target and the threshold in RB-ARQ was low, a non-target letter with high prior probability was tend to be selected incorrectly. When a non-target letter was inputted, the subject needed to input <BS> for the correction. Then, the discrimination accuracy became low because of the repetition of inputting <BS> and a non-target letter. Thus in the threshold higher than 0.9, the accuracy of Tri-gram improved because this repetition happened less frequently. As the result, Fig.3 shows that the

Table 2. Performance indexes in Exp.2

	Threshold	Equal	Uni-gram	Bi-gram	Tri-gram
Accuracy	0.9	0.81	0.797	0.805	0.777
	0.95	0.869	0.869	0.88	0.859
	0.99	0.918	0.933	0.944	0.941
♯ of Stimuli	0.9	78.9	64.7	58.8	54.1
	0.95	90.6	76.5	69.8	63.6
	0.99	110.8	98.9	91	83.2
Time[s]	0.9	13.8	11.3	10.3	9.5
	0.95	15.9	13.4	12.2	11.1
	0.99	19.4	17.3	15.9	14.6

performance in Utility of Uni-gram was better than Equal, Bi-gram was superior to Uni-gram and Tri-gram at threshold 0.95 or 0.99 was better than Bi-gram, while that of Tri-gram at threshold 0.9 was the worst.

On the other hand, Table 2 shows the improvement of the accuracy of Tri-gram at every threshold in Exp.2, which has the avoidance of the repitition by setting every prior probability equal. In this experiment, test sentences consisted of about 200 letters including a lot of particles which had poor connection with next letter. Thus, there were many times that the letter with low prior probability was to be selected as the target. In Exp.1, the prior probability was set to N-gram at all times, therefore, the repetition in Tri-gram was affected largely, which was improved by using the equal prior probability effectively with N-gram. In the comparison of Equal with Tri-gram, the discrimination time was shorten from 4 to 5 seconds per a letter. Thus in Fig.4, the performance in Utility of Tri-gram was superior to Bi-gram at every threshold.

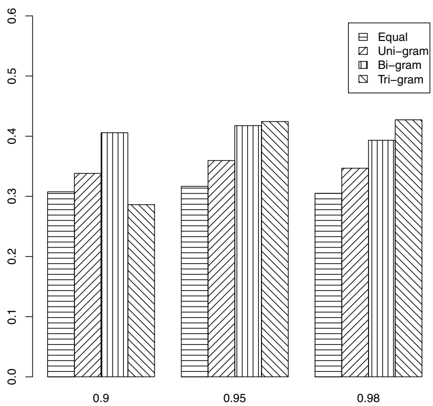

Fig. 3. Utility in Exp.1

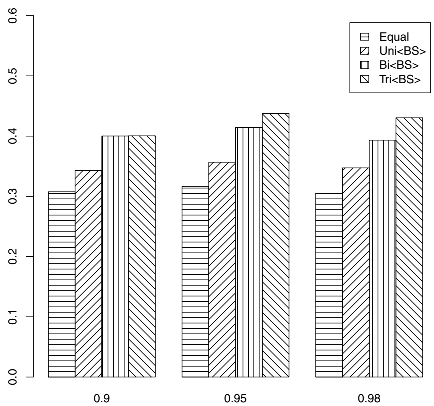

Fig. 4. Utility in Exp.2

There were the significant differences in Utility between the conventional method and every proposed method by the paired t-test at the significant level of $\alpha = 0.017$ (0.05/3; Boneferroni correction) considering multiple comparison. This result showed that considering the occurrence probability and the transition probability by the proposed method improved the inputting performance.

6 Conclusion

This paper proposed P300 speller that considering the occurrence probabilities and the transition probabilities between letters as the prior probability in RB-ARQ. The experiments were done by three subjects with Japanese interface of P300 speller and the result showed the improvement of spelling speed keeping high accuracy by the proposed method comparing with the conventional one. We will more investigate the proposed method through the online experiment.

References

[1] Wolpaw, J., Birbaumer, N., McFarland, D., Pfurtscheller, G., Vaughan, T., et al.: Brain-computer interfaces for communication and control. Clinical Neurophysiology 113(6), 767–791 (2002)

[2] Farwell, L.A., Donchin, E.: Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. Electroencephalography and Clinical Neurophysiology 70(6), 510–523 (1988)

[3] Hoffmann, U., Vesin, J.M., Ebrahimi, T., Diserens, K.: An efficient p300-based brain–computer interface for disabled subjects. Journal of Neuroscience Methods 167(1), 115–125 (2008)

[4] Scherer, R., Muller, G.R., Neuper, C., Graimann, B., Pfurtscheller, G.: An asynchronously controlled eeg-based virtual keyboard: improvement of the spelling rate. IEEE Transactions on Biomedical Engineering 51(6), 979–984 (2004)

[5] Takahashi, H., Yoshikawa, T., Furuhashi, T.: A study on application of reliability based automatic repeat request to brain computer interfaces. In: Proc. 15th Int. Conf. Neural Information Processing, pp. 1013–1020 (2009)

[6] Kaneda, Y., Takahashi, H., Yoshikawa, T., Furuhashi, T.: A study on application of reliability-based automatic repeat reQuest to p300 speller. IEICE Technical Report 109(280), 19–22 (2009)

[7] Kukich, K.: Techniques for automatically correcting words in text. ACM Computing Surveys (CSUR) 24(4), 377–439 (1992)

[8] Ullmann, J.R.: A binary n-gram technique for automatic correction of substitution, deletion, insertion and reversal errors in words. The Computer Journal 20(2), 141–147 (1977)

[9] Cavnar, W., Trenkle, J., et al.: N-gram-based text categorization, pp. 161–175, Citeseer, 48113 (1994)

[10] http://s-yata.jp/corpus/nwc2010/ngrams

[11] Dal Seno, B., Matteucci, M., Mainardi, L.T.: The utility metric: a novel method to assess the overall performance of discrete brain–computer interfaces. IEEE Transactions on Neural Systems and Rehabilitation Engineering 18(1), 20–28 (2010)