TITLE:

P300-Based Brain-Computer Interface Speller Performance Estimation with Classifier-Based Latency Estimation

AUTHORS:

Nazmun N. Khan[1]*, Taylor Sweet[1], Chase A. Harvey[1], Seth Warschausky[2], Jane E. Huggins[3,4], David E. Thompson[1]

[1]Brain and Body Sensing Lab, Mike Wiegers Department of Electrical & Computer Engineering, Kansas State University, Manhattan, KS, United States
[2]Adaptive Cognitive Assessment Laboratory, Department of Physical Medicine and Rehabilitation, University of Michigan, Ann Arbor, MI, United States
[3]Direct Brain Interface Laboratory, Department of Physical Medicine and Rehabilitation, University of Michigan, Ann Arbor, MI, United States
[4]Direct Brain Interface Laboratory, Department of Biomedical Engineering, University of Michigan, Ann Arbor, MI, United States

Email addresses of the co-authors:

Taylor Sweet	(tasweet@ksu.edu)
Chase A. Harvey	(caharvey@ksu.edu)
Seth Warschausky	(sethaw@med.umich.edu)
Jane E. Huggins	(janeh@med.umich.edu)
David E. Thompson	(davet@ksu.edu)

Email address of the corresponding author:

Nazmun N. Khan	(nkhan1@ksu.edu)

SUMMARY:

This article presents a method for estimating same-day P300 speller Brain-Computer Interface (BCI) accuracy using a small testing dataset.

ABSTRACT:

Performance estimation is a necessary step in the development and validation of Brain-Computer Interface (BCI) systems. Unfortunately, even modern BCI systems are slow, making collecting sufficient data for validation a time-consuming task for end users and experimenters alike. Yet without sufficient data, the random variation in performance can lead to false inferences about how well a BCI is working for a particular user. For example, P300 spellers commonly operate around 1–5 characters per minute. To estimate accuracy with a 5% resolution requires 20 characters (4–20 min). Despite this time investment, the confidence bounds for accuracy from 20 characters can be as much as ±23% depending on observed accuracy. A previously published method, Classifier-Based Latency Estimation (CBLE), was shown to be highly correlated with BCI accuracy. This work presents a protocol for using CBLE to predict a user's P300 speller accuracy from relatively few characters (~3–8) of typing data. The resulting confidence bounds are tighter

than those produced by traditional methods. The method can thus be used to estimate BCI performance more quickly and/or more accurately.

INTRODUCTION:

Brain-computer interfaces (BCIs) are a noninvasive technology that allows individuals to communicate through machines directly without regard for physical limitations imposed by the body. BCI can be utilized as an assistive device operated directly by the brain. BCI uses the brain activity of a user to determine if the user intends to choose a certain key (letter, number, or symbol) displayed on the screen[1]. In a typical computer system, a user physically presses the intended key on a keyboard. However, in a BCI system with a visual display, the user needs to focus on the desired key. Then, BCI will select the intended key by analyzing the measured brain signals[1]. The activity of the brain can be measured using various techniques. Though there are competing BCIs technologies, electroencephalogram (EEG) is considered a leading technique due to its noninvasive nature, high temporal resolution, reliability, and relatively low cost[2].

Applications of BCI include communication, device control, and also entertainment[3,4,5,6]. One of the most active BCI application areas is the P300 speller, which was introduced by Farwell and Donchin[7]. The P300 is an event-related potential (ERP) produced in response to the recognition of a rare but relevant stimulus[8]. When a person recognizes their target stimulus, they automatically produce a P300. The P300 is an effective signal for a BCI because it conveys the participant's recognition of the target event without requiring an outward response[9].

The P300 BCI has attracted researchers from computer science, electrical engineering, psychology, human factors, and various other disciplines. Advances have been made in signal processing, classification algorithms, user interfaces, stimulation schemes, and many other areas[10,11,12,13,14,15]. However, regardless of the research area, the common thread in all of this research is the necessity of measuring the BCI system performance. This task typically requires the generation of a test data set. This necessity is not limited to research; eventual clinical application as an assistive technology will likely require individual validation sets for each end user to ensure the system can generate reliable communication.

Despite the considerable research applied toward the P300 BCI, the systems are still quite slow. While the majority of people are able to use a P300 BCI[16], most P300 Spellers produce text on the order of 1–5 characters per minute. Unfortunately, this slow speed means that generating test data sets requires substantial time and effort for participants, experimenters, and eventual end users. Measuring BCI system accuracy is a binomial parameter estimation problem, and many characters of data are necessary for a good estimate.

To estimate the presence or absence of the P300 ERP, most classifiers use a binary classification model, which involves assigning a binary label (e.g., "presence" or "absence") to each trial or epoch of EEG data. The general equation used by most classifiers can be expressed as:

$$\hat{\boldsymbol{y}}(x) = w^T . f(x) + b$$

where $\hat{y}$ is called the classifier's score which represents the probability of the P300 response being present, x is the feature vector extracted from the EEG signal, and b is a bias term[17]. The function f is a decision function that maps the input data to the output label, and is learned from a set of labeled training data using a supervised learning algorithm[17]. During training, the classifier is trained on a labeled dataset of EEG signals, where each signal is labeled as either having a P300 response or not. The weight vector and bias term are optimized to minimize the error between the predicted output of the classifier and the true label of the EEG signal. Once the classifier is trained, it can be used to predict the presence of the P300 response in new EEG signals.

Different classifiers can use different decision functions, such as linear discriminant analysis (LDA), step-wise linear discriminant analysis (SWLDA), least squares (LS), logistic regression, support vector machines (SVM), or neural networks (NNs). The least squares classifier is a linear classifier that minimizes the sum of squared errors between the predicted class labels and the true class labels. This classifier predicts the class label of a new test sample using the following equation:

$$\hat{y}(x) = sign(\hat{\boldsymbol{w}} * x) \tag{1}$$

where the sign function returns +1 if the product is positive and -1 if it is negative and weight vector $\hat{\boldsymbol{w}}$ is obtained from the feature set of the training data, (x) and class labels (y) using the below equation:

$$\widehat{\boldsymbol{W}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T y \tag{2}$$

In earlier research, we argued that Classifier-Based Latency Estimation (CBLE) can be used to estimate BCI accuracy[17,18,19]. CBLE is a strategy for evaluating latency variation by exploiting the classifier's temporal sensitivity[18]. While the conventional approach to P300 classification involves using a single time window that is synchronized with each stimulus presentation, the CBLE method involves creating multiple time-shifted copies of the post-stimulus epochs. Then it detects the time shift that results in the maximum score in order to estimate the latency of the P300 response[17,18]. Here, this work presents a protocol that estimates BCI performance from a small dataset using CBLE. As a representative analysis, the number of characters is varied to make predictions of overall performance of an individual. For both example datasets, the root mean square error (RMSE) for vCBLE and actual BCI accuracy were computed. The results indicate that the RMSE from vCBLE predictions, using its fitted data, was consistently lower than the accuracy derived from 1 to 7 tested characters.

We developed a Graphical User Interface (GUI) called "CBLE Performance Estimation" for the implementation of the proposed methodology. Example code is also provided (**Supplementary Coding File 1**) that operates on the MATLAB platform. The example code performs all of the steps applied in the GUI, but the steps are provided to assist the reader with adapting to a new dataset. This code employs a publicly available dataset "Brain Invaders calibration-less P300-based BCI using dry EEG electrodes Dataset (bi2014a)" to evaluate the proposed method[20].

Participants played up to three game sessions of Brain Invaders, each session having 9 levels of the game. The data collection continued until all levels were completed or the participant lost all control over the BCI system. The Brain Invaders interface included 36 symbols that flashed in 12 groups of six aliens. According to the Brain Invaders P300 paradigm, a repetition was created by 12 flashes, one for each group. Out of these 12 flashes, two flashes contained the Target symbol (known as Target flashes), while the remaining 10 flashes did not contain the Target symbol (known as non-Target flashes). More information on this paradigm can be found in the original reference[20].

The CBLE approach was also implemented on a Michigan dataset, which contained data from 40 participants[18,19]. Here, the data of eight participants had to be discarded because their tasks were incomplete. The whole study required three visits from each participant. On the first day, each participant typed a 19-character training sentence, followed by three 23-character testing sentences on Days 1, 2, and 3. In this example, the keyboard included 36 characters which were grouped into six rows and six columns. Each row or column was flashed for 31.25 milliseconds with an interval of 125 milliseconds between flashes. Between characters, a 3.5 s pause was provided.

Figure 1 shows the block diagram of the proposed method. The detailed procedure is described in the protocol section.

PROTOCOL:

The “CBLE Performance Estimation” GUI was applied on two datasets: “BrainInvaders” dataset and Michigan dataset. For the “BrainInvaders” dataset, the data collection was approved by the Ethical Committee of the University of Grenoble Alpes[20]. Michigan data were collected under the University of Michigan Institutional Review Board approval[19]. Data were analyzed under Kansas State University exempt protocol 7516. If collecting new data, follow the user's IRB-approved process for collecting informed consent. Here, the proposed protocol is evaluated using offline analysis of previously-recorded, de-identified data and therefore did not require additional informed consent.

The graphical user interface (GUI) included in this paper is proficient in managing two distinct dataset formats. The first format is associated with the BCI2000 software, while the second format is referred to as the "BrainInvaders" dataset. In order to utilize the “Brain Invaders” format, data must be pre-processed as described in step 1 of the protocol section. However, when dealing with the “BCI2000” dataset format, step 1 can be omitted.

1. Data preparation

1.1. BrainInvaders only: Generate the input data file in ".mat" file format that can be used with the "CBLE performance Estimation" graphical user interface (GUI). For a sample script, refer to **Supplementary Coding File 2**.

NOTE: Each datafile consists of a two-dimensional matrix comprised of rows that represent observations recorded at distinct time samples. The matrix columns numbered from 2 to 17 are recordings derived from 16 EEG electrodes. The first column of the matrix denotes the timestamp of each observation, while column 18 encompasses information related to experimental events. In column 19, there are mostly zeros, but when a non-Target (or Target) flash starts, the numbers change to one (or two) at that specific time. A detailed description may be found in the reference[20].

2. Download and install the GUI package

2.1. Download and install the "CBLE Performance Estimation" GUI.

3. Store the dataset in a subfolder of the GUI location

3.1. Ensure that the dataset folder remains within the same directory as the GUI.

3.2. For instance, create a new folder and place the "CBLE Performance Estimation" GUI inside it. Keep all the datasets in a subfolder within "CBLE GUI" named "Dataset."

4. Open the installed GUI

4.1. Open MATLAB, change the current directory to folder where you placed the GUI, click "**APPS**" tab and select "**MY APPS**".

4.2. Under the "**MY APPS**" tab, select "CBLE Performance Estimation".

5. Choose dataset format

5.1. Select a dataset format from the dropdown "Select dataset format".

6. Load EEG data file

6.1. Click on the "**Select input folder**" button to choose the directory where the dataset is located.

6.2. Observe the count of data files present in that selected folder.

NOTE: In the "Brain Invaders" format, each participant is represented by a single data file. Therefore, the total number of data files indicates the number of participants in the study. However, this is not the case for the "BCI2000" format, as each participant may have multiple train and test files.

7. Set the parameters

7.1. Type the number of participants that the user intends to use for the estimation process in the “No. of subjects” text box.

7.2. BrainInvaders only: Specify the sampling rate of the dataset.

NOTE: BCI2000 files include the sample rate.

7.3. Choose a decimation value to downsample the dataset to approximately 20 Hz in order to improve classification performance[21]. For example, if the sampling frequency is 256 Hz, then select a decimation value of 13.

7.4. Specify the time window for the classification in milliseconds.

NOTE: The recommended initial window size is specified, allowing the starting point to vary from 0 to 100 ms and the ending point from 700 to 800 ms. However, it is important to avoid making the window size excessively large to prevent overlapping with another P300 event.

7.5. Define the shift window for CBLE in milliseconds.

NOTE: The term 'shift window' refers to the window of new epochs of EEG test data which are extracted by shifting the window forward one sample in time. This shift window size must be larger than the original window, as it indicates the number of shifts that CBLE can detect. The difference between the shift window and classification window should be less than 100 ms from each side.

7.6. BC2000 only: Enter the length of the subject ID indicated in the dataset files within the "ID length" field.

NOTE: The GUI expects the first sub_len characters of the filenames to encode the subject ID.

7.7. BCI2000 only: In the "Channel ID" field, indicate either the total number of channels or specify the specific channel numbers to be used for the analysis.

7.8. Click the “**Set parameters**” button to set all the parameters required for the analysis.

8. BrainInvaders only: Split the dataset into training and test set

8.1. Select a number of targets that represents the size of the training set. The remaining portion of the dataset will be considered as the test dataset.

NOTE: To ensure proper training of the model, it is essential to have a sufficiently large training sample. The recommended minimum training sample size is 20, though this may vary depending on the overall dataset size. If regression errors occur during training session, it is advisable to increase the training sample size.

8.2. Press the "**Split the dataset**" button to divide the dataset into the training and test sets.

NOTE: Each participant will have an equal amount of training data. However, the number of test data may not be equal for all participants due to the possibility of multiple attempts during the task. Consequently, the total number of targets or flashes presented may vary from person to person.

9. Train a model with training dataset

Note: Step 9.1 is applicable for "Brain Invaders" format and step 9.2 is applicable for "BCI2000" format.

9.1. BrainInvaders only: Click on the "**Train a model**" button to apply linear regression on the training dataset using equation 2 for training a classifier model.

9.2. BCI2000 only: Indicate training and testing filenames along with their data format (.dat) to distinguish the training and testing files from all files. Then, click on the "**Train a model**" button to apply linear regression on the training dataset.

10. Predict the accuracy of the test set

10.1. Click on the "**Predict accuracy**" to apply the trained classifier model to the test feature set and predict the accuracies using equation 1.

11. Get X-target accuracies

11.1. Select a maximum target number, X, to consider in the test set.

11.2. BCI2000 only: Select a test file number if user has multiple test files.

11.3. Press "**Find X target accuracy**" to get the accuracies.

12. Calculate vCBLE

12.1. Click on the "**Find vCBLE**" button to get the vCBLE for all targets.

13. Calculate Root mean square error (RMSE) of BCI accuracy and vCBLE

13.1. Click on the "**Calculate RMSE**" button to calculate the RMSE between both predictions based on vCBLE with BCI accuracy, and X-target accuracy with BCI accuracy.

14. Visualize the results of the analysis

14.1. Click on the "**Accuracy vs vCBLE**" button to observe the relation between total accuracy and total vCBLE for all participants.

14.2. Press on the "**RMSE of BCI & vCBLE**" button to show the RMSE curve of BCI accuracy and vCBLE.

15. Prediction of the performance of an individual participant

15.1. To predict the accuracy of an individual participant, place the subject id in "**Sub ID**".

NOTE: Here, the dataset of all participants, excluding the test participant, will be used to train a linear regression model. The vCBLE scores of all other participants and their corresponding test accuracies will be utilized as predictors and labels, respectively, for the classifier.

15.2. Select a target number, n. The prediction will be made based on the accuracy of n testing characters.

15.3. Click the "**Predic**t" button to get the predicted accuracy of the test participant.

REPRESENTATIVE RESULTS:

The proposed protocol has been tested on two different datasets: "BrainInvaders" and the Michigan dataset. These datasets are already introduced briefly in the Introduction section. The parameters used for this two datasets are mentioned in **Table 1**. **Figures 2–4** depict the findings obtained using the "BrainInvaders" dataset, whereas **Figures 5–7** demonstrate the results achieved from the Michigan dataset.

The "BrainInvaders" dataset has 64 participants. **Figure 2** presents the relationship between BCI accuracy and vCBLE of all 64 participants. It shows that vCBLE is highly negatively correlated with BCI accuracy, although a few outliers are observed. **Figure 3** illustrates the RMSE of vCBLE and actual accuracy when the prediction was made based on the accuracy of testing characters. It shows the evidence that the RMSE for this prediction, based on the fit obtained by vCBLE, was lower than the accuracy based on any number of testing characters from 1 to 10. For the "BrainInvaders" dataset, vCBLE is capable of predicting BCI accuracy using only 7 characters. In **Figure 4**, the prediction was made from the vCBLE of test sets with 2, 5, 7, and 10 characters, respectively. Here, a leave-one-participant-out approach was employed in the regression analysis to predict the accuracy of each individual participant. BCI accuracy and vCBLE were estimated over 100 repetitions. The lower and upper bounds are ±2 standard deviations from the mean. All four conditions indicate that there is minimal variance observed when the number of participants in the training set exceeds 10. We conclude that about 10 individuals are required to build the regression model for the relationship between vCBLE and accuracy for a particular experimental paradigm.

In the second example, the Michigan dataset has 32 participants, in which all typed one training sentence on Day 1 and three testing sentences on Days 1, 2, and 3. The test sentences were 23 or 24 characters in length, and many participants made additional selections to correct errors made during online operation. In **Figure 5**, it can be seen that the vCBLE model performed better when the training and testing datasets were collected on the same day. In fact, this prediction resulting from the fit provided by vCBLE resulted in a lower RMSE than the accuracy based for any number of testing characters from 1-20 when the training data and testing data were collected on the same day. **Figure 6** shows that overall, the vCBLE fit had a lower RMSE when the test included less than six characters. Additionally, it can be seen from **Figure 7** that the RMSE of the vCBLE accuracy estimation only decreases about 0.025 between three characters and the optimal number of characters used. This implies that there is not much benefit to collecting more than three characters for the small test set.

FIGURE AND TABLE LEGENDS:

Figure 1: Block diagram of the proposed protocol. (a) Data preprocessing and feature extraction. (b) P300 classification. (c) Evaluation of vCBLE. (d) Predicting the accuracy of an individual.

Figure 2: Accuracy vs. vCBLE. BCI accuracy plotted against vCBLE using "bi2014a" dataset. It shows a high negative correlation between accuracy and vCBLE.

Figure 3: RMSE of BCI accuracy and vCBLE. The RMSE of vCBLE and accuracy were plotted against different test dataset sizes (1-10) using "bi2014a" dataset. Overall, vCBLE performs better than BCI accuracy.

Figure 4: Comparison of models using RMSE. These models are built while predictions are performed from different sizes of test characters. Top left: 2 targets; top right: 5 targets; bottom left: 7 targets; bottom right: 10 targets.

Figure 5: RMSE Values of vCBLE Models. A separate model was built to predict accuracy over three different days using the Michigan dataset. The RMSE values for models built using different test dataset sizes are shown.

Figure 6: Model comparison. The mean of the RMSE over three days was computed for the vCBLE and the accuracy models using Michigan data.

Figure 7: Mean RMSE difference from the best model. For each day, the minimum RMSE value was subtracted from each character's RMSE value. The mean was calculated over the three days. This graph represents the average performance of a model using a certain data set size compared to the best model.

TABLE 1: Standard parameters for "BrainInvaders" and Michigan datasets.

DISCUSSION:

This article outlined a method for estimating BCI accuracy using a small P300 dataset. Here, the current protocol was developed based on the “bi2014a” dataset, although the efficacy of the protocol was confirmed on two different datasets. To successfully implement this technique, it is crucial to establish certain variables, such as the epoch window for the original data, the window for time shifting, the down-sampling ratio, and the size of both the training and testing datasets. These variables are determined by the characteristics of the dataset being used, including the number of targets or characters, the number of sequences, and the total number of participants.

The findings of the "bi2014a" dataset indicate that vCBLE's prediction exhibits superior performance compared to character-level BCI accuracy for all test conditions (less than 10 characters), which involve test datasets containing one to ten characters. However, when the test dataset comprises more than seven targets, the performance of vCBLE shows minimal variance. Results from the Michigan data suggest that using vCBLE to predict same-day performance will outperform the accuracy-based estimation if the test data set is less than six characters. Interestingly, increasing the amount of data used to build this model only improves marginally after the first few characters of data. Overall, this would imply that it is not necessary to collect large amounts of data to predict same-day accuracy.

According to the outcomes of the “bi2014a” dataset, it can be suggested that a minimum of 10 participants is necessary to construct a classifier model that can forecast an individual's BCI accuracy. However, this also depends on the number of characters or the number of sequences used in both the training and testing phases. The " bi2014a " dataset includes several participants who had a relatively small number of total targets. It is worth mentioning that the vCBLE prediction method has already been successfully tested on small-size datasets consisting of 32 and 9 participants, respectively, and has demonstrated effective performance[17,18]. However, these datasets have a relatively larger number of total targets, such as 19 characters in the training session and a minimum of 23 characters in the testing session.

There are a few limitations to be aware of when applying this method. From the analysis of the Michigan dataset, the vCBLE model seems to perform worse when the training and test data are collected on different days. Also, this method requires multiple participants to build a custom model for a given dataset. Moreover, the proposed method has been tested on four classifiers, including a least-squares classifier, stepwise linear discriminant analysis, support vector machine (SVM), and space autoencoder (SAE)[17,18]. However, the protocol should be applicable to any time-sensitive classifier. Despite these limitations, the potential time savings to the research and clinical communities warrant further investigation and application.

ACKNOWLEDGMENTS:

The data used for representative results were collected from the work supported by the National Institute of Child Health and Human Development (NICHD), the National Institutes of Health (NIH) under Grant R21HD054697, and the National Institute on Disability and Rehabilitation Research (NIDRR) in the Department of Education under Grant H133G090005 and Award Number

H133P090008. The rest of the work was funded in part by the National Science Foundation (NSF) under award #1910526. Findings and opinions within this work do not necessarily reflect the positions of NICHD, NIH, NIDRR or NSF.

DISCLOSURES:

All authors declare they do not have any conflicts of interest.

REFERENCES:

1. Rezeika A, Benda M, Stawicki P, Gembler F, Saboor A, Volosyak I. Brain-Computer Interface spellers: A review. *Brain Sci.* **8**(4), (2018).

2. Gannouni S, Aledaily A, Belwafi K, Aboalsamh H. Em9otion detection using electroencephalography signals and a zero-time windowing-based epoch estimation and relevant electrode identification. *Sci. Rep.* **11**(1), 7071(2021).

3. Daly JJ, Wolpaw JR. Brain-computer interfaces in neurological rehabilitation. *Lancet Neurol.* **7(**11), 1032–1043(2008).

4. Birbaumer N. Breaking the silence: brain-computer interfaces (BCI) for communication and motor control. *Psychophysiology.* **43(**6), 517–532(2006).

5. Riccio A, Simione L, Schettini F, Pizzimenti A, Inghilleri M, Belardinelli MO, et al. Attention and P300-based BCI performance in people with amyotrophic lateral sclerosis. *Front. Hum. Neurosci.* **7**, 732(2013).

6. Finke A, Lenhardt A, Ritter H. The MindGame: a P300-based brain-computer interface game. *Neural Netw.* **22**, (9), 1329–1333(2009).

7. Farwell LA, 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).

8. Li Q, Lu Z, Gao N, Yang J. Optimizing the performance of the visual P300-speller through active mental tasks based on color distinction and modulation of task difficulty. *Front. Hum. Neurosci.* **13**, 130(2019).

9. McFarland DJ, Sarnacki WA, Townsend G, Vaughan T, Wolpaw JR. The P300-based brain-computer interface (BCI): effects of stimulus rate. *Clin. Neurophysiol.*, **122**(4), 731–737(2011).

10. Krusienski DJ, Sellers EW, Cabestaing F, Bayoudh S, McFarland DJ, Vaughan TM, et al. A comparison of classification techniques for the P300 Speller. *J. Neural Eng.* **3(**4), 299–305(2006).

11. Sellers EW, Donchin E. A P300-based brain-computer interface: initial tests by ALS patients. *Clin. Neurophysiol.* **117** (3),538–548(2006).

12. Donchin E, Spencer KM, Wijesinghe R. The mental prosthesis: assessing the speed of a P300-based brain-computer interface. *IEEE Trans. Rehabil. Eng.* **8** (2), 174–179(2000).

13. Höhne J, Schreuder M, Blankertz B, Tangermann M. A novel 9-class auditory ERP paradigm driving a predictive text entry system. *Front. Neurosci.* **5**, 99(2011).

14. Acqualagna L, Treder MS, Blankertz B. Chroma Speller: Isotropic visual stimuli for truly gaze-independent spelling. *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, (2013).

15. Townsend G, LaPallo BK, Boulay CB, Krusienski DJ, Frye GE, Hauser CK, et al. A novel P300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns. *Clin. Neurophysiol.* **121(**7), 1109–1120(2010).

16. Guger C, Daban S, Sellers E, Holzner C, Krausz G, Carabalona R, et al. How many people are able to control a P300-based brain-computer interface (BCI)?, *Neurosci. Lett.* **462**(1), 94–98(2009).

17. Mowla MR, Gonzalez-Morales JD, Rico-Martinez J, Ulichnie DA, Thompson DE. A comparison of classification techniques to predict Brain-computer interfaces accuracy using classifier-based latency estimation. Brain Sci. 10(10), (2020).

18. Thompson, D. E., Warschausky, S., & Huggins, J. E}. Classifier-based latency estimation: a novel way to estimate and predict BCI accuracy. *Journal of Neural Engineering.* **10**(1), (2012).

19. Thompson DE, Gruis KL, Huggins JE. A plug-and-play brain-computer interface to operate commercial assistive technology. *Disabil. Rehabil. Assist. Technol.* **9**(2), 144–150(2014).

20. Korczowski, L., Ostaschenko, E., Andreev, A., Cattan, G., Coelho Rodrigues, P. L., Gautheret, V., & Congedo, M. (2019). Brain Invaders calibration-less P300-based BCI using dry EEG electrodes Dataset (bi2014a) [Data set]. Zenodo.

21. Krusienski DJ, Sellers EW, Cabestaing F, Bayoudh S, McFarland DJ, Vaughan TM, Wolpaw JR. A comparison of classification techniques for the P300 Speller. *J Neural Eng*. **3**(4), (2006).

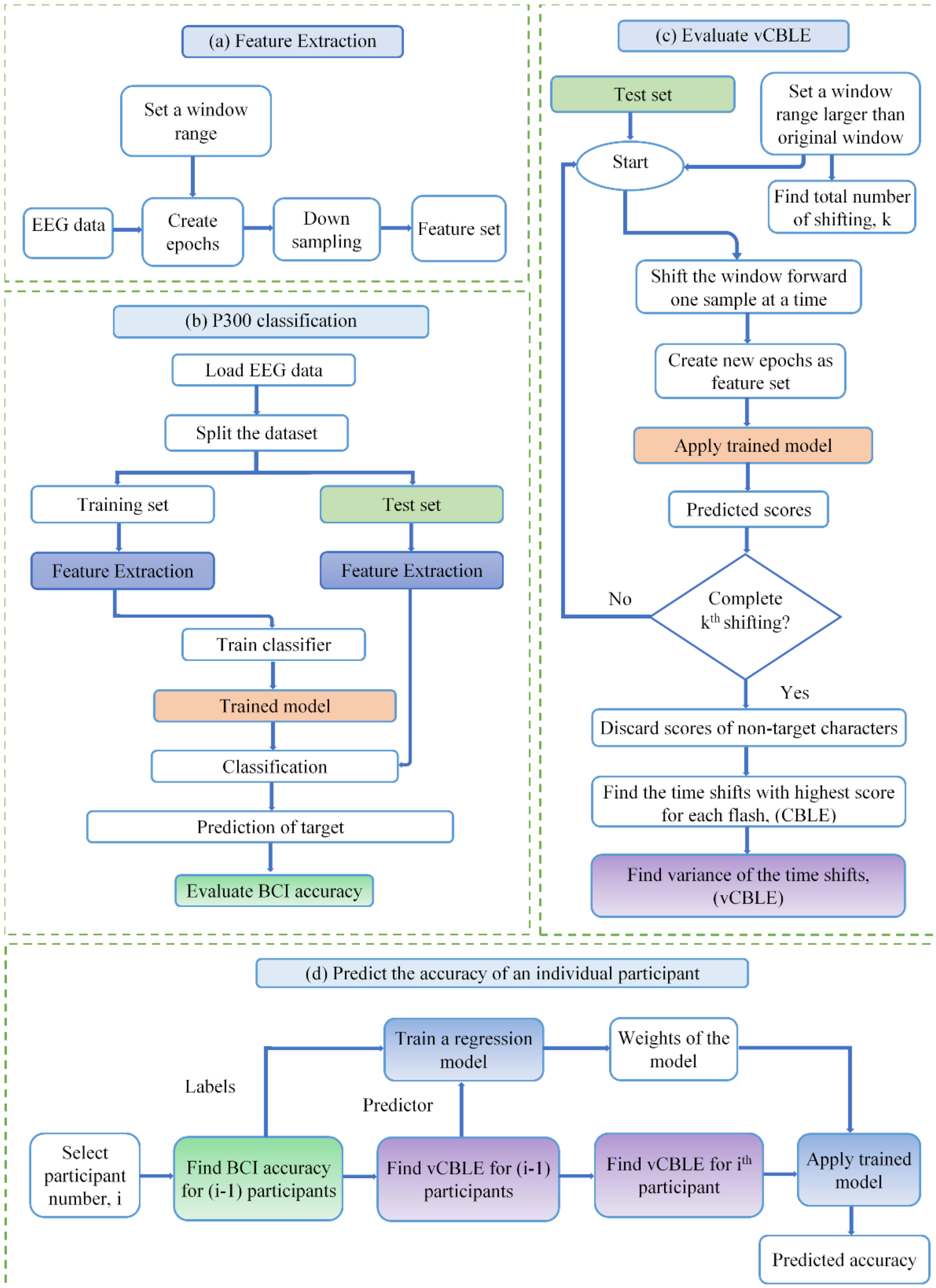

Figure 1: Block diagram of the proposed protocol. (a) Data preprocessing and feature extraction. (b) P300 classification. (c) Evaluation of vCBLE. (d) Predicting the accuracy of an individual.

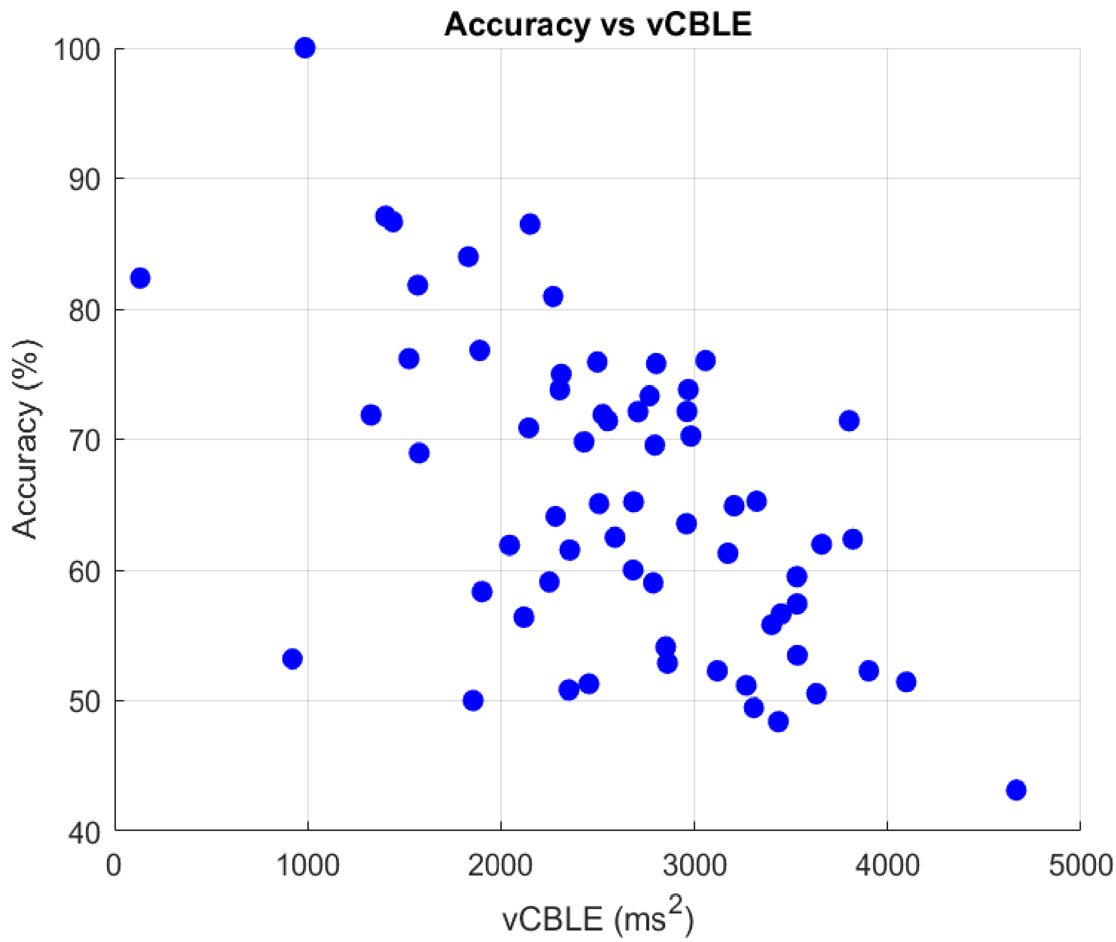

Figure 2: Accuracy vs. vCBLE. BCI accuracy plotted against vCBLE using “bi2014a” dataset. It shows a high negative correlation between accuracy and vCBLE.

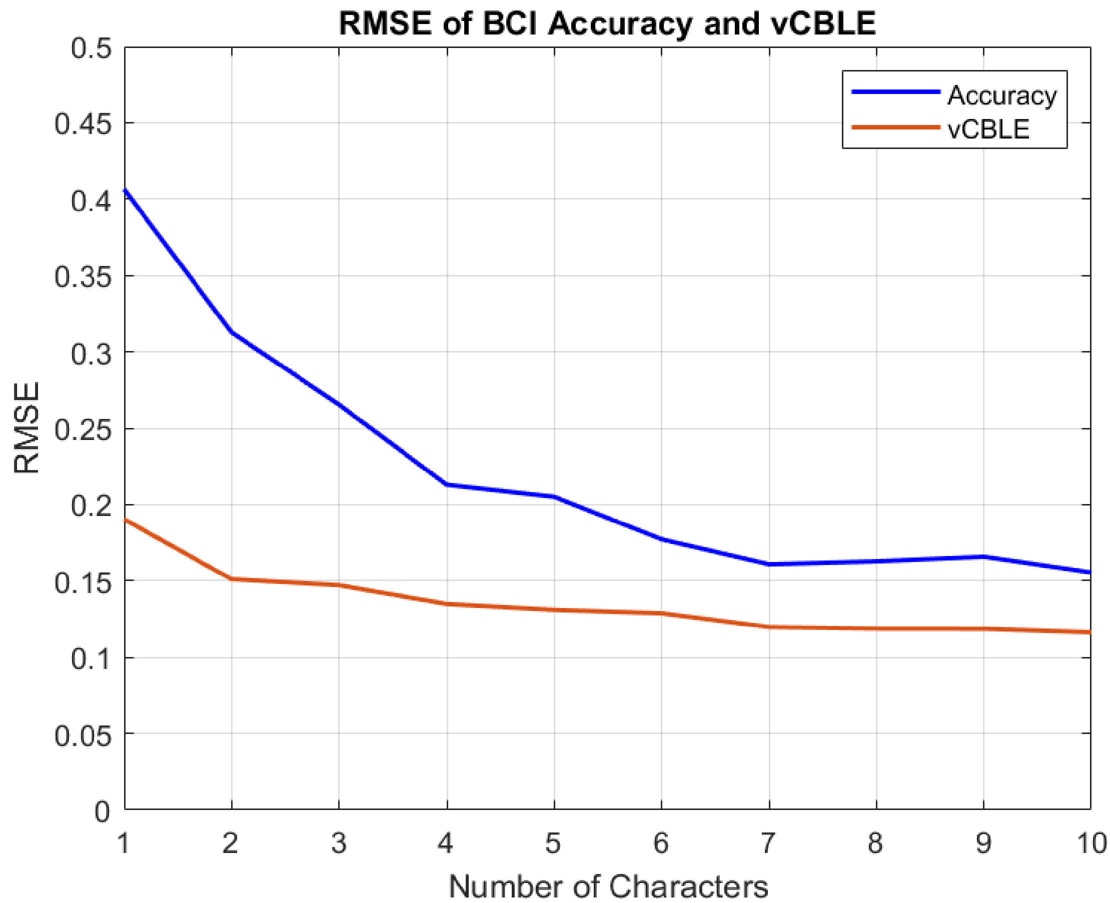

Figure 3: RMSE of BCI accuracy and vCBLE. The RMSE of vCBLE and accuracy were plotted against different test dataset sizes (1-10) using “bi2014a” dataset. Overall, vCBLE performs better than BCI accuracy.

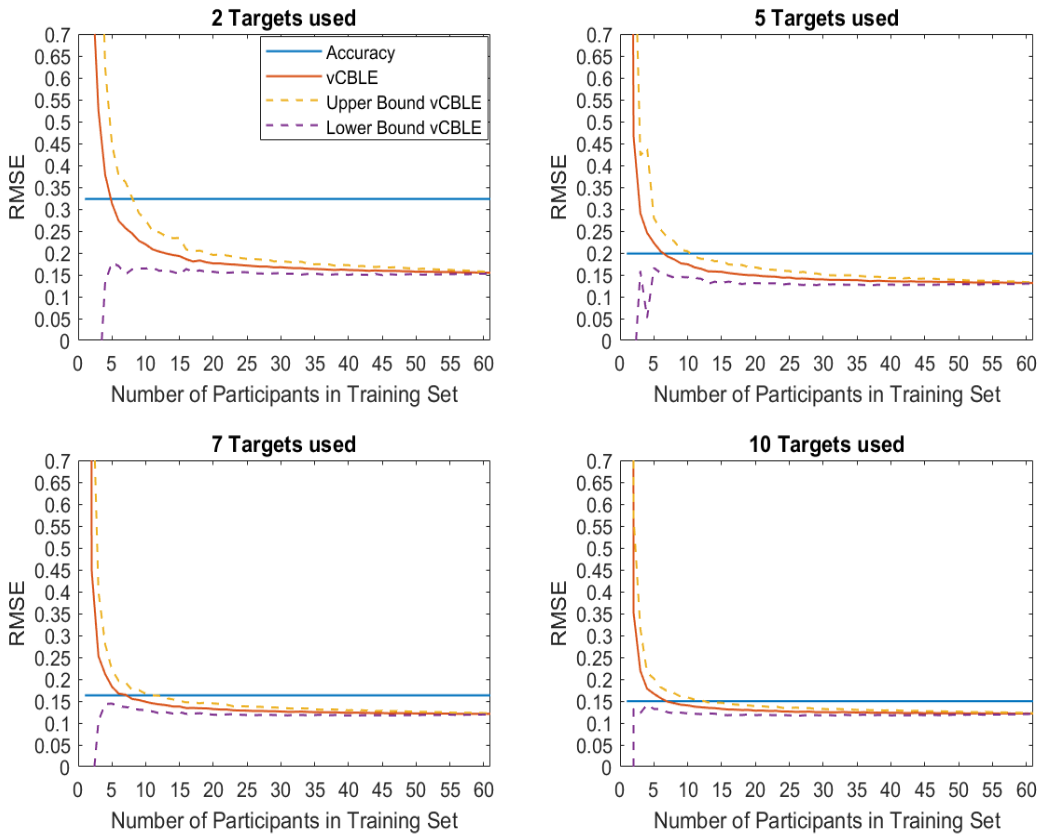

Figure 4: Comparison of models using RMSE. These models are built while predictions are performed from different sizes of test characters. Top left: 2 targets; top right: 5 targets; bottom left: 7 targets; bottom right: 10 targets.

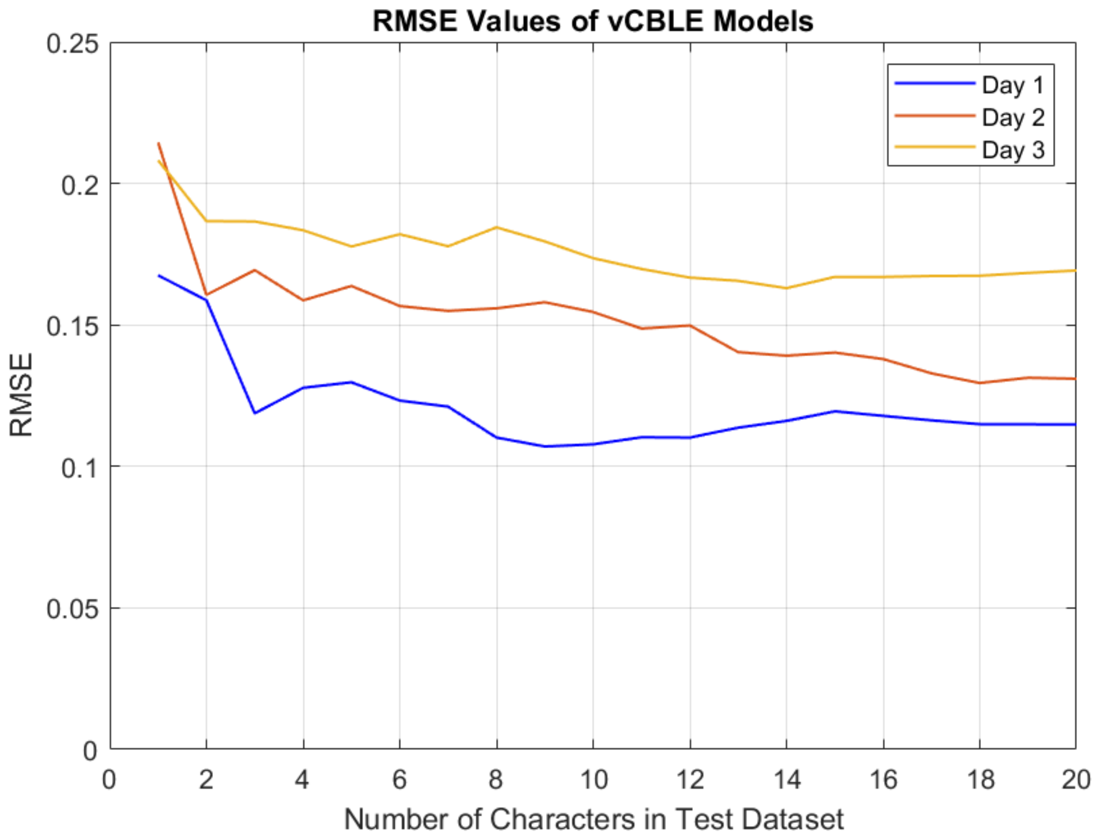

Figure 5: RMSE Values of vCBLE Models. A separate model was built to predict accuracy over three different days using the Michigan dataset. The RMSE values for models built using different test dataset sizes are shown.

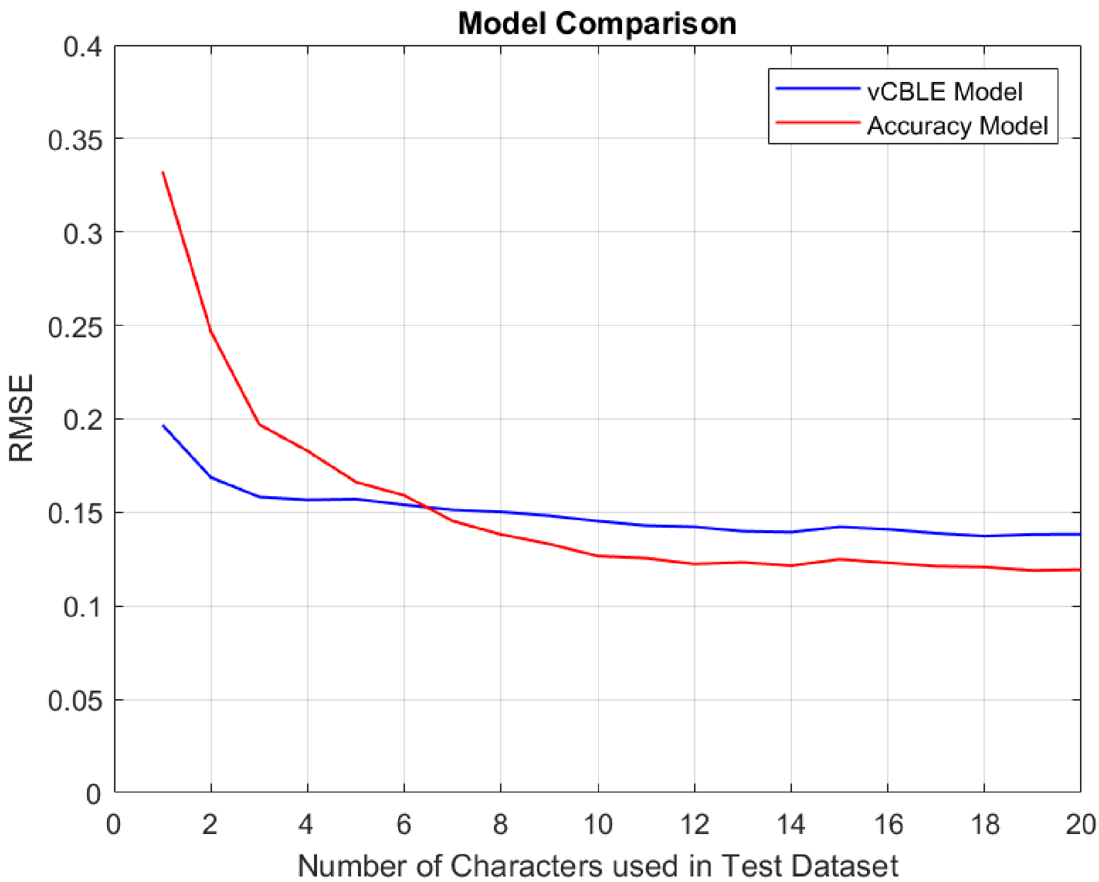

Figure 6: Model comparison. The mean of the RMSE over three days was computed for the vCBLE and the accuracy models using Michigan data.

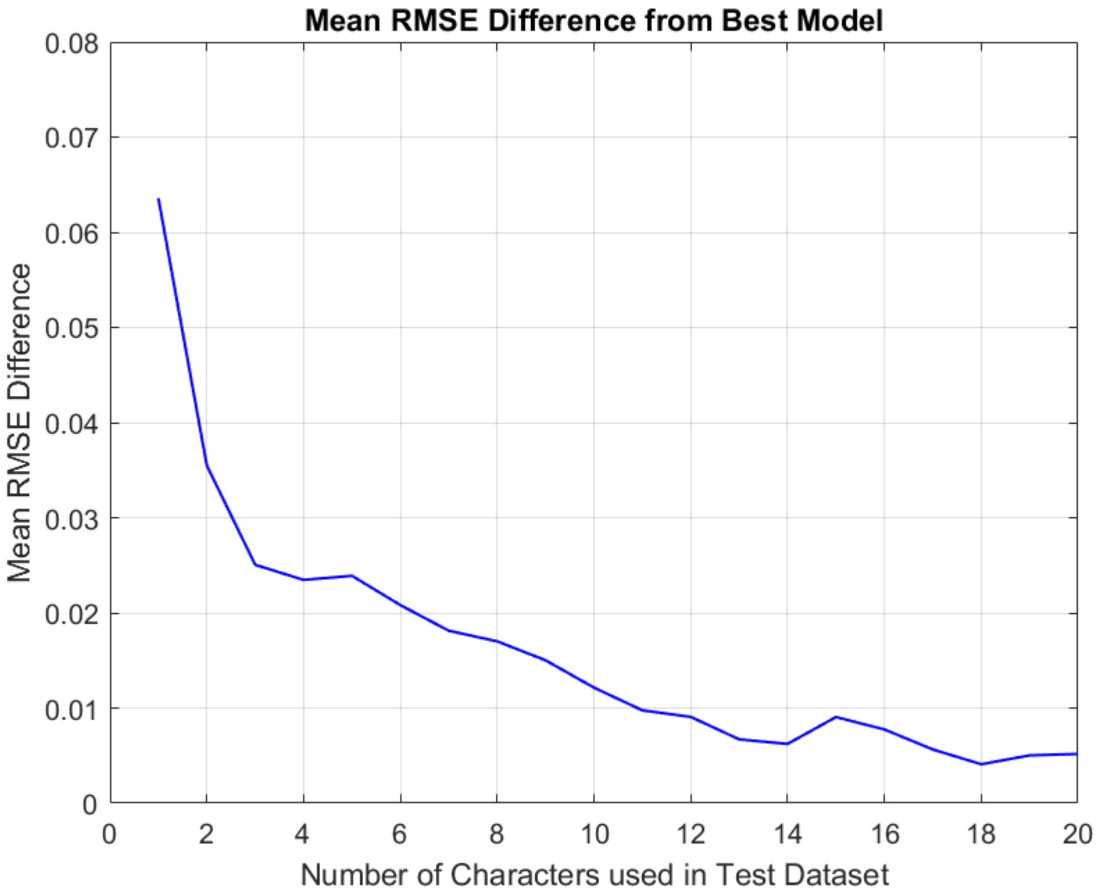

Figure 7: Mean RMSE difference from the best model. For each day, the minimum RMSE value was subtracted from each character's RMSE value. The mean was calculated over the three days. This graph represents the average performance of a model using a certain data set size compared to the best model.

Table 1: Standard parameters for "BrainInvaders" and Michigan datasets

Dataset name	Subject number	ID length	Channel ID	Sampling rate	Decimation value	Original window	CBLE window	Training sample no	Target number, X
Brain Invaders	64	N/A	[1:16]	512	26	[100, 600]	[0, 700]	20	10
Michigan	32	4	[1:16]	256	13	[4, 804]	[-100, 900]	N/A	20