A Robust Classification Approach for Character Detection Using P300-Based Brain-Computer Interface

Deepthi Hitesh Mehta (deepthihmehta14499@gmail.com)
National Institute of Technology, Raipur, India

Mohit Sarin (mohitsarin98@gmail.com)
National Institute of Technology, Raipur, India

Praveen Kumar Shukla (praveenlnct98@gmail.com)
VIT Bhopal University, India

Shrish Verma (shrishverma@nitrr.ac.in)
National Institute of Technology, Raipur, India

Rahul Kumar Chaurasiya (rkchaurasiya@nitrr.ac.in)
National Institute of Technology, Raipur, India

ABSTRACT

Researchers have been contributing to the brain-computer interface (BCI), which acts as a direct connection between the human brain and the computer that uses the P300 speller paradigm to decode the response of the brain by stimulating a subject, involving no muscular movements. This research uses BCI Competition III Dataset II, which uses a 6x6 character matrix paradigm for data collection purposes for two healthy subjects. The ensembles of support vector machine (SVM) method of classification has been proposed to surpass the problem of false detection, which is preceded by empirical mode decomposition (EMD) as the preprocessing technique and the use of a stacked autoencoder for feature extraction and covariate shift adaptation by normalized principal components as the feature selection method for better accuracy of the detected character. The experiment yields a better result than many existing methods; it produces an average accuracy of 98.75%.

INTRODUCTION

Brain-Computer Interface (BCI) is direct communication between the human brain and the computer machine. BCI involves the decoding of brain responses without any involvement of muscular motions. BCI is a proven excellent mode of communication for people with neurological disorders who cannot communicate their emotions and feelings through handwriting, speaking, or typing (R.K. Chaurasiya et al., 2016). One of the primarily used techniques among all the BCI techniques is Electroencephalographic(EEG) because of its non-invasive recording technique and cost-benefit ratio. The communication in EEG is based on an event-related potential (ERP)which is generated, recorded, and analyzed, resulting in the event that visually stimulates a subject. An elicited component that is the response to ERP is the P300 signal. P300 ERP is a natural endogenous response that varies with stimulation type, subject matter, and expectations, but it is independent of stimulation's physical characteristics. P300 speller consists of the following technical aspects-recording of EEG signals, preprocessing, feature extraction, and classification (Akcakaya M et al., 2014).

P300 potential is observed when rare, and expected events occur at the central locations of EEG measurement. P300 is the user's brain signature which typically happens around 300 ms after the unusual occurrence (Wolpaw JR et al., 2002). A 6x6 matrix of characters is shown in the P300 speller. The job of the user is to focus his attention on the characteristics of a predefined sentence, one character at a time. But due to the proposed paradigm, the user faces problems with crowding, exhaustion, and thus resulting in false detection. So, to surpass this problem, the Ensembled Support Vector Machine (ESVM) method of classification has been proposed in this research which is preceded by Empirical Mode Decomposition (EMD) as the preprocessing technique and use of a stacked autoencoder and have applied Covariate Shift Adaptation by normalized principal components as an optimization technique for selection of extracted features for better accuracy after classification.

LITERATURE REVIEW

This research uses BCI Competition III Dataset II, which consists of P300 signals, and for the feature extraction, the research makes use of an autoencoder. Any autoencoder takes input in the form of images, so there is a need to convert the P300 signals into an image. In their paper (Azad et al., 2019), has converted the 1D signals/ vibrational signals to 2-D signals using Empirical Mode Decomposition (EMD) and utilizing the energy esteems. The main reasons for the use of EMD for the conversion are because of its adaptive nature and since it allows the projection of a non-stationary signal onto a time-frequency plane using mono-component signals.

There are many ways to extract features; this research uses an autoencoder for feature extraction as the autoencoder is capable of removing redundant information. It doesn’t require labeled information of the data to create a model for feature extraction (Md Shopon et al., 2016). There are many autoencoders available for feature extraction. In their research (S. Kundu et al., 2019), have used sparse and stacked sparse autoencoder for feature extraction, which yielded good results but took more time for training and had less consistency. (Minmin Chen, 2014) proposed a Marginalized Denoising autoencoder that learns the mapping from input to the output but doesn’t learn representation. This autoencoder can only modify the domain but cannot be used for representational learning and related problems. Stacked Autoendocer has recently evolved to provide a version of the raw data with very comprehensive and promising features to train a classifier in a particular context and find more consistency than raw information training (Venkata Krishna Jonnalagadda, 2018).

To increase the accuracy, this research uses Principal Component Analysis (PCA) along with Covariate Shift Adaption (CSA).PCA is used for extracting the principal components (Jolliffe, I. T et al., 2016). (Spüler, M et al., 2012) paper has made use of the mentioned technique and has described that the technique extracts principle components using PCA and CSA is used to normalize extracted components to the effect of non-stationarity by displacing the window over the results. There are various covariate shift adaption methods. A. Satti in (A. Satti et al., 2010) proposed a covariate shift adaption method that uses a polynomial function to adapt the data accordingly resulting from the estimation of the covariate shift of the subsequent trial, for that it uses a polynomial of order 3.

(Chaurasiya et al., 2015) have proposed an efficient P300 speller. They have performed their experiments on the Devnagiri script. The design and architecture of the speller are much similar to that of BCI Competition III Dataset II. Classification of P300 signals was performed using Ensembles of SVM in this research. They were successful in minimizing problems related to multi-trial. Taking it one step further, (G. B. Kshirsagar et al., 2020) proposed a Single-trial Character Detection using the Ensembles of a Deep Convolutional Network. They have achieved an accuracy of 96.2% on Devnagiri based P300 speller.

DATASET DESCRIPTION

The dataset used is the BCI Competition III Dataset II. It is a complete record of P300 evoked potentials, recorded with BCI20001 by Donchin et al. in 2000 and originally Farwell and Donchin in 1988 using a paradigm. The dataset contained data collected from 2 subjects collected in 5 sessions. The user was shown a 6x6 matrix of English alphabets and digits (0-9) and was asked to focus on the single character out of the 36 characters. The frequencies were increased successively and at random at 5.7Hz, for all rows and columns on this matrix. Out of the 12 intensified rows and columns, a particular row and column contain the desired character. The reactions evoked by these two of the 12 stimuli which contain the desired character differ from the stimuli which did not contain the desired character and are similar to those described in Farwell and Donchin, 1988; (Donchin et al., 2000).

The data was collected in 5 sessions, each having many runs. The subject focuses on a series of characters in each run. The user display was as follows for the run of each character epoch: the matrix was seen for 2.5 s, and each character had the same intensity during that time. Each row and column was intensified for 100ms, and then the matrix was left blank for 75ms. There were a total of 12 intensifications; each of them was repeated 15 times for each character epoch, so there were a total of 180 intensifications (12 x 15=180) for each character epoch. An epoch of each character followed for a period of 2.5sec, and then the matrix was kept blank. During this period, the user was told of the completion of this character, and the emphasis was on the next character in the word on the top of the screen. The dataset had 4 data, two for each subject, one of two was training data, and the other was the test data of that subject. The training data was used to predict the character sequence in the test data. In the end, the signals were passed through a bandpass filter from 0.1-60 Hz (Farwell LA et al., 1988).

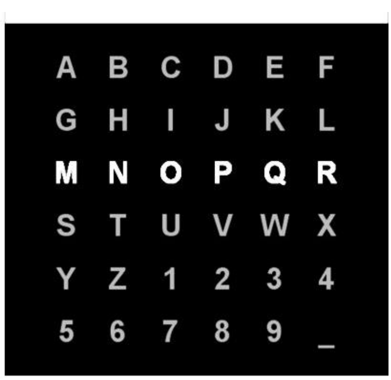

Figure 1: The P300 Speller Paradigm for data collection

PROPOSED METHODOLOGY

The research makes use of BCI Competition III Dataset II, which consists of data with signals in binary matrix form which need to be converted into 2-D grayscale images; thus, the research makes use of Empirical Mode Decomposition (EMD) as the preprocessing technique. The application of EMD results in Intrinsic Mode Functions (IMFs); out of all the IMFs, those with higher frequencies are only considered. Then the energy of the signals used for converting into 2-D images is calculated. Then the energy values are normalized, and the energy spectrum is converted into a 2-D image. The research uses a stacked autoencoder for feature extraction from 2-D images, and classification is done using the Support Vector Machine (SVM) method.

FILTERING & PREPROCESSING

To avoid noise and to restrict our observation to a particular range of frequencies, we have used the 8th order Chebyshev Bandpass filter with a lower cut-off frequency of 0.1 Hz and a higher cut-off frequency of 10 Hz. We have chosen this range because, as the study (Zhihua Yang et al., 2006) suggests, the P300 signal lies predominantly in this range of frequencies.

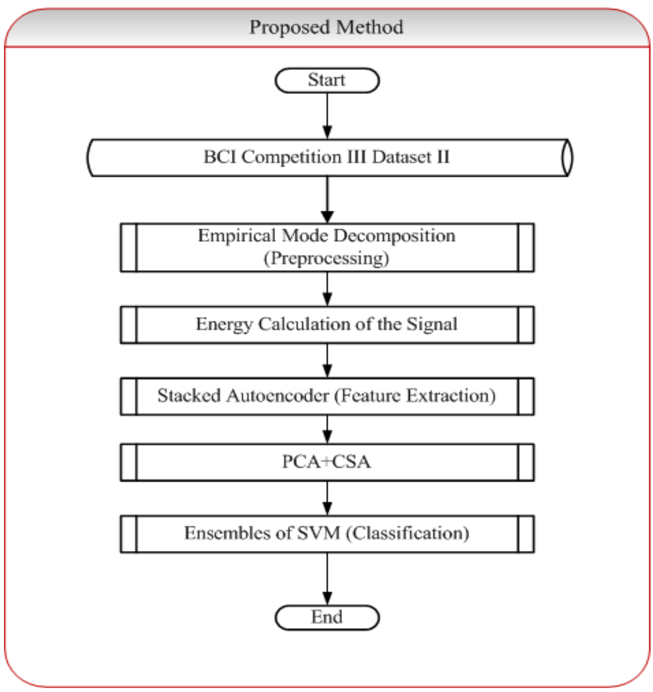

Figure 2: Proposed Method

EMD

This research makes use of autoencoders which take inputs only in the form of images. The dataset used for the study is the BCI Competition III Dataset II which consists of the data in the form of signals which is not suitable as an input for the autoencoder. So, there arises a need to preprocess the data to make that suitable as an input for the autoencoder.

EMD is a time-space adaptive analysis approach suitable for non-stationary, non-linear, and stochastic signal processing. EMD involves breaking down a signal without leaving the time domain and was proposed as an integral part of the Hilbert–Huang transform (HHT). Hilbert-Huang transform (HHT) works in two stages. In the first stage, EMD breaks up non-linear and non-stationary data into a band-limited component summary called Intrinsic Mode Functions (IMF). The second stage involves producing an instantaneous frequency spectrum by applying the Hilbert transform to the results obtained in the first stage.
The energy of the wave signal has then been used to describe the essence of the signal more effectively than using the numerical values of the time field of the signal. Then the energy values are used to convert into a 2-D image.

1. Segmentation of the signal into subparts called Frames.
2. Calculating the size of the frame which is the multiplication of duration of the frame and signal's frequency rate.
   Size of the frame= frame duration x Frequency rate of the signal.
   Defining the size of the matrix-
   The height of the matrix is equal to the size of the frame (M) and width equates to the number of frames (N) produced after segmentation.
   Size of the matrix= M x N
3. Keeping the energy values into the cells of the matrix and transferring the frame values vertically to the matrix. The first column holds the energy values of the first frame, the first frame value is transferred to the first cell of the first row, the second frame value is transferred to the first cell of the second row and so on, and the last frame value of the first frame is transferred to the first cell of the last row. This process continues for further frames. Since the height of the matrix equates to the size of the frame, all the frame values get fit in the matrix and since the width of the matrix is equal to the number of frames produced after segmentation, all the energy values get fit in the matrix.
4. The values in the matrix are normalized in the range 0-255, thus reducing the noise.
   2-D representation preserves time-domain signal characteristics, and even 2-D signal texture characteristics can be extracted to detect the signal [4].

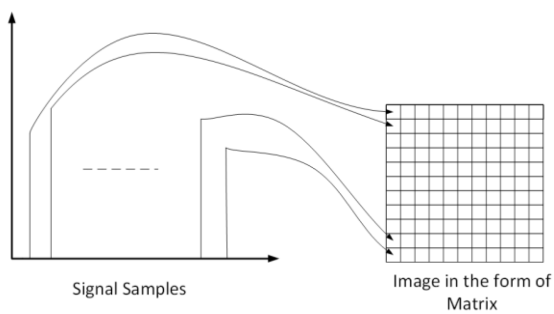

Figure 3: Conversion into a 2-D matrix with the energy values of the sample

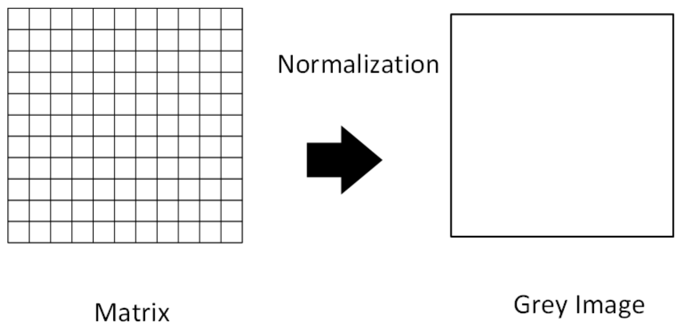

Figure 4: Producing an image from the 2-D matrix

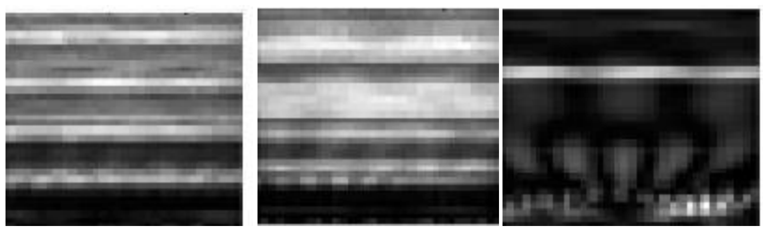

Figure 5: Conversion of signal into a grey image.

Feature Extraction Autoencoder

Stacked Autoencoder is prominently used for denoising images and compression of extracted features to a minimum so that only the most important features will remain. It is a feed-forward network that is used to reproduce the result in the output.
The hidden layer will maintain all the useful information, as the bottleneck part is much smaller than the input layer; hence it is used for data compression. The weights between the hidden layer and the reconstructed output layer, which is weighted towards the decoder side, are tied to the weights used in encoder sides. “Tied” means that the weights towards the decoder side are the transpose of weights used for the encoder side.

In this research, the output obtained after converting the P300 signals into a 2-D grayscale image is given as input to the stacked autoencoder. Initially, 30 percent of the features for the test and 70 percent for training were chosen to analyze classification problems better. This was achieved to ensure that the model knows the P300 stimulus and can develop the parameters. And then, all the training sets were passed together for better learning levels and precision after the parameters and the model had been found.
MATLAB neural network was used to implement this project. Parameters such as the number of iterations, number of layers, and neurons of the stacked autoencoders were set empirically. Initially, the project began with two layers and 1000 iterations. Still, as the work continued, the layers were increased to improve the accuracy, and the iterations were reduced to 300 after examination to face the over-fitting condition. The investigation resulted in an optimum number of layers as follows:

1. The inputs with 420 features were passed to the autoencoder's first layer.
2. The inputs with 420 features were shortened to 210 by the first layer of autoencoder, which was trained to 210.
3. Next, there was a 100 neuron autoencoder.
4. The 100 features were passed on to the next autoencoder with 50 neurons.
5. And then, the 50 features were passed on to the next autoencoder with 20 neurons.
6. The 20 features were passed to the last autoencoder with ten neurons.

This clearly shows that to reduce the features from 420 to 10 there were 5 layers which were a combination of 5 autoencoders. Besides, other parameters such as L2Weight Regularization and Sparsity Regularization were set in the MATLAB default configuration.

Further, the ten features were passed on to the softmax matrix with 200 iterations. So, stacked autoencoders with the structure: 420-210-100-50-10 were achieved at the end (Vařeka et al., 2017)

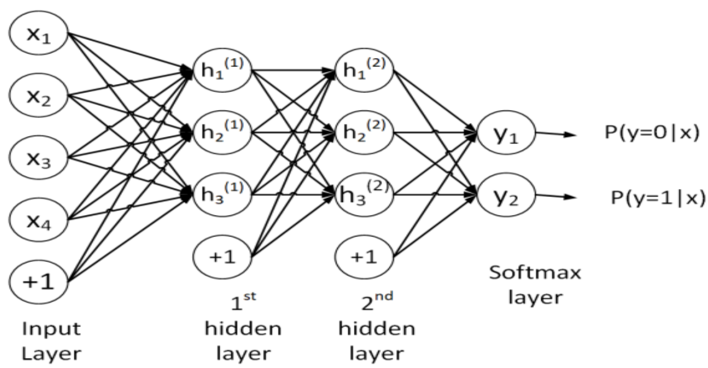

Figure 6: Architecture of Stacked Autoencoder

CSA+PCA

The uncorrelated variables obtained by orthogonal transformation of correlated variables are called principal components, and this method is called the PCA method (Andrzej Maćkiewicz et al., 1993). The sorting of the principal components depends on the variance, and the first major component reflects the highest variance of the original data. After extracting the features, when the power spectrum has been estimated for each channel, the normalization method is applied. The dataset contains a matrix D with dimensions n to p in the case of n testing of training, and the numbers of p= (channels. bins) and D (i, j) features are the value of tests i and j.

Firstly, a PCA is employed to minimize dimensionality and remove non-stationary components for the covariate shift adaptation, followed by the selection of m principal components with the highest variance, thus resulting in a transformation matrix W(pxm) and a P=D.W matrix representing the principal components of m. Further, a rectangular window is defined with length w, which moves through the data and normalizes the value of P (i, j) in the previous w trial with-

$$\widehat{P_{(ij)}} = P_{(ij)} - mean(P_{(i-w,j)}, \ldots\ldots, P_{(w,j)})$$

$(P_{(1,j)}, \ldots\ldots, P_{(w,j)})$ is used for all $\widehat{P_{(ij)}}$ with $i \leq w$.

ESVM

In this research work, the features extracted are applied to the ensembles of the SVM classifier (Marc Claesen et al., 2014) as the input, and the final output is the sum of the results from each classifier which are normalized using min-max normalization. Different types of features selected for evaluation are then concatenated to represent the signal in a much better way in terms of the ratio of information it holds.

Min-Max

If f_p is the score for test data assigned by the pth classifier then the min-max normalization (Patro et al., 2015) function is given as follows-

$$f_{pnorm} = \frac{f_p - \min(f_p)}{\max(f_p) - \min(f_p)}$$

f_p is the normalized score.
The scaling takes place between 0 and 1 because of the min-max normalization. The final output of the classifier is given as-

$$S = \frac{1}{j}\frac{1}{p}\sum_{j=1}^{J}\sum_{p=1}^{P} f_{pnorm}$$

J-number of epochs.
P-number of classifiers.
The intersection of the row and column gives the desired character.

RESULTS

For every subject, they have provided test data that consists of 85 characters and training data that consists of 100 characters. So, we have approximately 46% training data and 54% testing data individually for both the subjects. Hence, in total, they have provided four .mat files in single-precision format. As we have worked on a newer version of MATLAB, we have converted a single-precision format to a double-precision format.

In this research, we used EMD as a preprocessing technique; and extracted features using stacked autoencoders. We have optimized the obtained results using PCA- Corporate Shift Analysis. For the final classification, we have used Ensembles of SVM. Further, we have used two different architectures of the model. The first is preprocessing, feature extraction using a stacked autoencoder, and classification using ESVM. In the second method, we have applied PCA+ CSA for effective feature selection and to improve the performance of our proposed model.

Figure 7 summarizes the results using the first architecture obtained after every epoch. For simplification, results are rounded off to the nearest one's place. It is evident from the results that classification accuracy for subject A is significantly lower for the first epoch, but it gained faster than subject B. At the end of 1st, 5th, 10th, 15th, and 18th epoch we have obtained the subject wise accuracies of A as 19.5%, 71.6%, 85.2%, 95.5%, and 98.0%, whereas for subject B the results are 32.0%, 76.9%, 89.0%, 95.0%, and 97.5% respectively. The average accuracy obtained after 18 epochs is 97.75%.
Figure 7 also compares results we have obtained after including PCA +CSA for feature selection before the classification using Ensembles of SVM with those obtained without CSA. We can see the results obtained using this architecture is around 1-1.5% better than the first architecture of the proposed model. Using PCA+ CSA, we have obtained subject-wise classification accuracies for Subject A for epochs 1st, 5th, 10th, 15th, and 18th as 21.0%, 72.1%, 93.2%,98.4%, and 99.3%. Similarly for Subject B it is 36.3%, 78.3%, 91.2%, 96.8%, and 98.2%respectively. We can see that the average classification accuracy here is increased to 98.75%.

Table 1: Confusion after 18 epochs without CSA

Subjects	Expected	Output
A	H	Z
	Q	P
B	T	Z

Table 2: Confusion after 18 epochs with CSA

Subjects	Expected	Output
A	Q	P
B	T	Z

Mainly, the alphabets misclassified after 18 epochs lie in the same row or same columns. It means either row or column is detected correctly for these misclassified symbols. This may be because some of the signals which have been recorded using different subjects are co-related.

In Table 1, H, which is misclassified as Z, is in the same column, Q, which is misclassified as P, is in the same row, similarly for T, which is misclassified as Z, by subject B. On the contrary, Table 4 summarizes the classification result obtained using the architecture that consists of feature selection using PCA + CSA. It's clear from the results that the feature selection technique helps to improve the model accuracy.

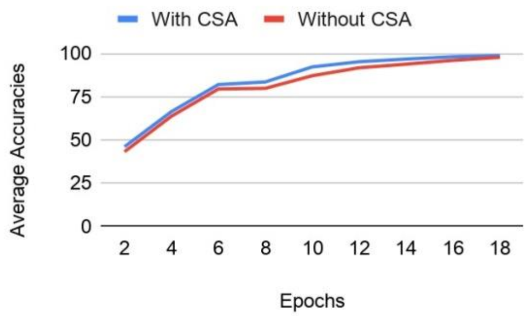

Figure 7: Graph showing results with CSA and without CSA

Table 3: Paired T-test

Method	P-value
CNN-1 (H. Cecotti et al., 2011)	$8.735\ 10^{-6}$
MCNN-1 (F. Reza et al.,2012)	$4.255\ 10^{-5}$
Temporal-ESVM (S. Kundu et al., 2018)	$2.78\ 10^{-3}$
PCA-EWSVM (V. Guigue, 2008)	$5.669\ 10^{-6}$
ESVM (M. J. Idaji et al., 2017)	$4.89\ 10^{-4}$

Hypothesis testing is one of the oldest procedures for checking data validity. The T-test is one of those statistical methods. In the beginning, the t-test implies a null hypothesis meaning the two data set means are identical. Based on the test's predefined method, specific values are calculated and compared with standard values, and the null hypothesis is finally accepted or rejected depending on the calculated values. If the null hypothesis appears to be dismissed, the collection of evidence is solid and is therefore not by mistake.
In this data, we have applied, paired T-test, and checked the value p-value. As it is clear from Table 3, that all of the results of the paired t-test give the p-value, which is significantly less than 0.05. Hence this set of data passes the t-test. This test also validates the performance of our model, as it shows improvement in character recognition compared to earlier reported techniques by other researchers.

DISCUSSIONS

For further analysis of the results obtained, we have compared our results with the results of our fellow researchers. Many great researchers have worked in this field and specifically on this dataset, they have used various methods for pre-processing, feature extraction, channel selection, and classification. Some have reported average classification accuracy, whereas few reported subject-wise accuracy. A few experiments have targeted artifact removal in pre-processing, whereas others targeted proper channel selection. In this section, we have compared feature extraction techniques, channel selection techniques, and classification methods.

Table 4 provides a detailed comparison of different methods that different researchers used to solve the proposed classification problem for BCI Competition III dataset II. Authors [25] used ensembles of SVM to classify this data; they have extracted features using a proposed deep neural network. He (H. Cecotti et al., 2011) has used CNN with 16 hidden layers, extracted deep features using CNN, and performed classification using the CNN network.
He (S. Kundu et al., 2018) used PCA as a feature extraction method and ensembles of SVM as a classification method and has obtained a classification accuracy of 99% & 97% for subjects A & B, respectively. In the (Mina Jamshidi Idaji, 2017)applied LDA for classification of these signals, and the results were comparatively less accurate than deep neural networks. Deep neural networks are more effective in this classification problem as they extract more relevant features. Moreover, the results are much more optimized because of the use of autoencoders for extracting features, as this would also include temporal features, and hence much better feature extraction. Autoencoders have already been used by (Liu et al., 2017) & (S. Kundu et al., 2019, and they have reported an average classification accuracy of 98% & 98.5%. We have extended his (S. Kundu et al., 2019) thoughts of using autoencoder but with autoencoder, we have also applied PCA+CSA for effective feature selection to improve classification results.

Table 4: Comparison of classification accuracy

Method	Subject A	Subject B	Mean
Proposed Method	99.3	98.2	98.75
CNN-1(H. Cecotti et al., 2011)	97.0	92.0	94.5
MCNN-1(H. Cecotti et al., 2011)	97.0	94.0	95.5
Temporal-ESVM (F. Reza et al., 2012)	99.0	97.0	98.0
PCA-EMSVM (V. Guigue et al., 2008)	99.0	97.0	98.0
ESVM (Mina et al.,2017)	97.0	96.0	96.5
HOSRDA&LDA (Mina et al.,2017)	96.0	97.0	96.5
GsBLDA (T. Yu et al., 2015)	99.0	95.0	97.0
BN3 (Liu et al., 2017)	98.0	95.0	96.5
SSAE-ESVM (S. Kundu et al., 2019)	99.0	98.0	98.5

CONCLUSION & FUTURE SCOPE

In this set of experiments, we have improved the classification accuracy of BCI Competition III dataset II. For this, we have implemented a stacked autoencoder for the extraction of features and PCA +CSA for feature selection and the classification was performed using the ensemble of SVM and the overall results are represented in normalized form using min-max normalization. Feature extraction using a Stacked autoencoder, we have obtained an average accuracy of 98.75 % when we have used this model architecture, compared to the accuracy of 97.75 % when PCA +CSA is not included in the architecture of the model. Approximately an increase of [1-1.5] % is recorded when we have used the feature selection compared to when not. Paired t-test with other proposed methods was also performed, and the results are satisfactory as all of them give a p-value less than 0.05. A detailed comparison of results obtained and that obtained by other research has been done, in terms of classification method and preprocessing techniques which they have used. Those alphabets are also identified, which have been misclassified after 18 epochs and have been listed in Table 4.

In the future, we can try improving the accuracy close to 100% by using different channel selection algorithms to reject those channels that have more noise. We will also try to extend this research to more than two subjects and compare the performance of our model in that case. Further, we plan to implement our proposed model in an online system. Also, we are working to improve accuracy using lower sequences r trails.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest regarding this work.

ACKNOWLEDGMENT

The authors extend their appreciation and gratitude to the Electronics and Communication Department, National Institute of Technology, Raipur, for supporting this research work.

REFERENCES

Rahul Kumar Chaurasiya, Narendra D. Londhe & Subhojit Ghosh. (2016). Binary DE-Based Channel Selection and Weighted Ensemble of SVM Classification for Novel Brain–Computer Interface Using Devanagari Script-Based P300 Speller Paradigm, *International Journal of Human–Computer Interaction*, 32:11, 861-877

Akcakaya M, Peters B, Moghadamfalahi M, Mooney AR, Orhan U, Oken B, Erdogmus D, Fried-Oken M. (2014). Noninvasive brain-computer interfaces for augmentative and alternative communication. *IEEE Rev Biomed Eng. 2014*;7:31-49.

Wolpaw JR, Birbaumer N, McFarland DJ, Pfurtscheller G, Vaughan TM. (2002) Brain-computer interfaces for communication and control. *Clin Neurophysiol.;*Page767-91.

Azad, M,Khaled, F,Pavel, Monirul.(2019). A novel approach to classify and convert 1D signal to 2D grayscale image implementing support vector machine and empirical mode decomposition algorithm.,*International Journal of Advanced Research*

Md Shopon and Nabeel Mohammed and Md. Anowarul Abedin. (2016). Bangla handwritten digit recognition using autoencoder and deep convolutional neural network, *International Workshop on Computational Intelligence* (IWCI)},64-68

Kundu, Sourav, Ari, Samit. (2019).P300 based character recognition using sparse autoencoder with ensemble of SVMs *Biocybernetics and Biomedical Engineering*, Volume 39

Minmin Chen, Kilian Weinberger, Fei Sha, Yoshua Bengio. (2014). Marginalized Denoising Auto-encoders for Nonlinear Representations, *Proceedings of the 31st International Conference on Machine Learning, PMLR* 32(2):1476-1484,.

Venkata Krishna Jonnalagadda. (2018). Sparse, Stacked and Variational Autoencoder, https://medium.com/

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, *374*(2065)

Spüler, M., Rosenstiel, W. & Bogdan. (2012). M. Principal component based covariate shift adaption to reduce non-stationarity in a MEG-based brain-computer interface. *EURASIP J. Adv. Signal Process.*

A. Satti, C. Guan, D. Coyle and G. Prasad. (2010). A Covariate Shift Minimisation Method to Alleviate Non-stationarity Effects for an Adaptive Brain-Computer Interface, *20th International Conference on Pattern Recognition*

R. K. Chaurasiya, N. D. Londhe and S. Ghosh. (2015). An efficient P300 speller system for Brain-Computer Interface, *International Conference on Signal Processing, Computing and Control (ISPCC)*

G. B. Kshirsagar and N. D. Londhe. (2020). Weighted Ensemble of Deep Convolution Neural Networks for Single-Trial Character Detection in Devanagari-Script-Based P300 Speller *IEEE Transactions on Cognitive and Developmental Systems*, vol. 12, no. 3, pp. 551-560,

Farwell LA, Donchin E. (1988). Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalogr Clin Neurophysiol.*

Thulasidas M, Guan C, Wu J. (2006). Robust classification of EEG signal for brain-computer interface. *IEEE Trans Neural Syst Rehabil Eng.*

Wadsworth BCI Dataset (P300 Evoked Potentials),Data Acquired Using BCI2000's P3 Speller Paradigm. (2004). (http://www.bci2000.org)

Zhihua Yang, Lihua Yang, Dongxu Qi, and Ching Y. Suen. (2006). An EMD-based recognition method for Chinese fonts and styles. *Pattern Recognition Letters*, *Volume 27*

Vařeka Lukáš, Mautner Pavel. (2017) Stacked Autoencoders for the P300 Component Detection, *Frontiers in Neuroscience*

Andrzej Maćkiewicz, Waldemar Ratajczak. (1993) Principal components analysis (PCA),*Computers & Geosciences,* Volume 19, Issue 3,Pages 303-342

Marc Claesen,Frank De Smet,Johan A.K. Suykens,Bart De Moor. (2014). EnsembleSVM: A Library for Ensemble Learning Using Support Vector Machines, *Journal of Machine Learning Research*, Page: 141-145

Patro, S Gopal Sahu, Dr-Kishore Kumar. (2015),Normalization: A Preprocessing Stage, *International Advanced Research Journal in Science*, *Engineering and Technology*

H. Cecotti and A. Graser,. (2011) Convolutional Neural Networks for P300 Detection with

Application to Brain-Computer Interfaces, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 33, no. 3, pp. 433-445

Fazel-Rezai Reza, Allison Brendan, Guger Christoph, Sellers Eric, Kleih Sonja, Kübler Andrea. (2012). P300 brain computer interface: current challenges and emerging trends, *Frontiers in Neuroengineering*, Volume-5

S. Kundu and S. Ari. (2018). P300 Detection Using Ensemble of SVM for Brain-Computer Interface Application, *9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*

A. Rakotomamonjy and V. Guigue. (2008) BCI Competition III: Dataset II- Ensemble of SVMs for BCI P300 Speller, *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 1147-1154

Mina Jamshidi Idaji , Mohammad B. Shamsollahi, Sepideh Hajipour Sardouie. (2017). Higher order spectral regression discriminant analysis (HOSRDA): A tensor feature reduction method for ERP detection, *Journal of Pattern Recognition, Elsevier*

T. Yu, Z. Yu, Z. Gu and Y. Li. (2015). Grouped Automatic Relevance Determination and Its Application in Channel Selection for P300 BCIs, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 6, pp. 1068-1077

Liu, Mingfei, Wu, Wei, Gu, Zhenghui, Yu, Zhuliang, Qi, FeiFei, Li, Yuanqing. (2017). Deep Learning Based on Batch Normalization for P300 Signal Detection, *Neurocomputing*, Volume 275, Pages 288-297

Sourav Kundu and Samit Ari. (2019). P300 based character recognition using sparse autoencoder with ensemble of SVMs, *Biocybernetics and Biomedical Engineering*

.