Proceedings of 2019 IEEE Region 10 Symposium (TENSYMP)

Analyzing the difference between deep learning and machine learning features of EEG signal using clustering techniques

Anushri Saha[1*], Sachin Singh Rathore[2], Shivam Sharma[2], and Debasis Samanta[1]
[1]Computer Science and Engineering, [2]Department of Mathematics and Computing
Indian Institute of Technology Kharagpur
Kharagpur, India
Email: *anusahaa2z@gmail.com

***Abstract*—The performance of many important Brain-Computer Interface (BCI) applications depends on its classification system. In former decades of BCI, machine learning classification algorithms serve this purpose. But in the recent picture of BCI, deep learning replaces the position of conventional machine learning algorithms. The research papers of BCI, where deep learning is used for the classification, claim to get better accuracy for the system compares to the use of machine learning. Though the machine learning classification algorithms are specifically designed for the purpose of classification, deep learning achieves better results by using a single layer of Softmax for the job of classification. The key behind this achievement of deep learning is its extracted features from several non-linear hidden layers. These features are far better for the task of classification than the handcrafted features used in the machine learning algorithms. In this work, we have analyzed the difference between these two different kinds of features with the support of clustering techniques.**

***Index Terms*—brain-computer interface, deep learning, machine learning, clustering**

I. Introduction

In the last few decades Brain-Computer Interface (BCI) become an emerging popular field of research due to its various applications like communicating for motor disable people [1], [2], preventing and recovering from neurological disease [3], [4], workload or fatigue monitoring [5], [6], adaptive e-learning [7], brain fingerprinting [8], [9], gaming [10] etc. The main aim of BCI is to establish a communication link between the user's brain and an external device [11]. The whole system constitutes a sequential process as brain signal preprocessing, feature extraction, classification and passing the adequate command to the external device [11].

The performance of the BCI system relies on the classification algorithm. Most of the BCI research works are mainly focused on the searching for a suitable classification algorithm which can improve the performance of the system. In earlier decades of BCI, researchers used machine learning classification algorithms for this purpose [12]–[15].

Nowadays deep learning techniques have gradually gained attention in the field of BCI. In [16]–[19], the authors claimed to get better accuracy by using deep learning techniques instead of machine learning.

All the machine learning classification algorithms are specifically built for the purpose of classification. But in the case of deep networks, only the last layer of Softmax achieves the job of classification and also with enhanced precision. The main differences between these two techniques are the features on which the classification is done. In the occasion of machine learning, handcrafted features are used, which are extracted from the signal using some statistical formulas. Some example of these kinds of features is the coefficient of variation, Hjorth parameters, band power ratio, etc [12]. But in the matter of deep learning, features are generated through the nonlinear hidden layers of the deep network.

In this work, we studied the differences between these two kinds of features with the aid of clustering techniques. We have used three different well-known clustering techniques [20] which are centroid based PAM (Partitioning Around Medoids) [20], hierarchical based BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) [20] and density based DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [20]. We have extracted some handcrafted features from our collected EEG data. Deep network features are also generated from unsupervised data using Stacked Denoising Autoencoder (SDAE) [21]. The two different sets of features are clustered separately using these three clustering techniques. The qualities of the resultant clusters are measured using different intrinsic and extrinsic clustering validation metrics to analyze the difference in the discrimination power of those two sets of features.

II. Work done

An outline of this work is presented in Fig 1. The tasks involved in this work are described in the following subsections:

A. Data acquisition

We have collected our own EEG signal for three different levels of cognitive loads. The EEG signal was recorded using 64 channels RMS India device in the BCI-HCI Lab of IIT Kharagpur. The dataset consists of data from four subjects where each subject data comprises two trials. The detail

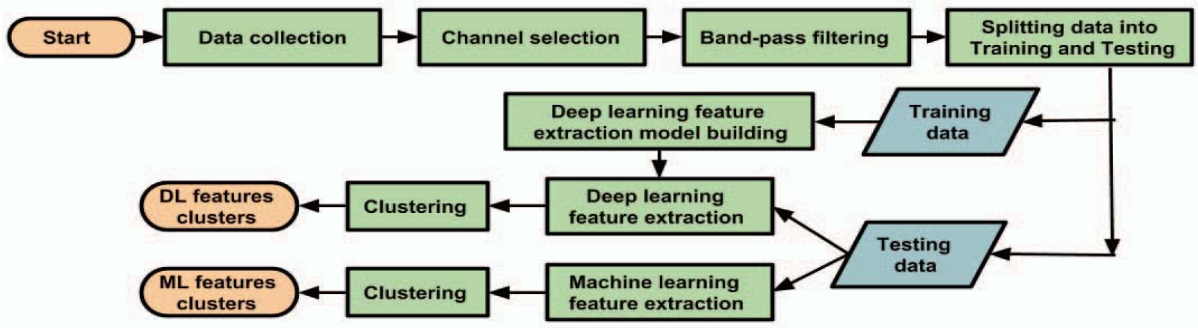


Figure 1: Flowchart for Feature sets Clustering

description of the experimental setup for this data collection can be found in our previous work [19]. After the data collection, the class information is removed from the dataset to make it unsupervised before further processing.

B. Channel selection and Band-pass filtering

Channel selection and filtering are two important steps to enhance the performance of the whole system. Explanations of these two steps are already discussed in our previous work [19].

C. Data splitting for training and testing

In our dataset, there is all total of four subjects data and for each subject, there are two different trials and each trial contains data related to three different cognitive loads. Both the trials data of every subject are divided into training and testing parts separately, using Monte Carlo cross-validation [22]with four splits. Monte Carlo cross-validation technique is used because of its higher optimization over K-fold and holdout cross-validation. Using this cross-validation technique, each split is separated randomly as 60:40 training and testing datasets. BCI systems should be designed in such a way that it should work for any subject. In order to accomplish this, the entire training dataset for all subjects and all trials are considered together for the training of the deep learning feature extraction model. But each of the testing dataset for a single trial and specific subject is used for the feature extraction, for both the handcrafted and deep learning features, distinctly. All clustering results are obtained by averaging over the four splits.

D. Machine learning feature extraction

We have extracted various kinds of features, some of which are statistical features, some are from the time domain, some are from the frequency domain, some are from the wavelet domain and some are auto-regressive (AR) model features. The window size was fixed at 256 which is the sampling frequency of the RMS India EEG device. We have extracted five statistical features which are Mean, Median, Standard Deviation, Skewness, and Kurtosis. In the time domain, we have extracted Hjorth parameters, Zero crossing, Coefficient of Variation, Slope Mean, Slope Variance, Fractal Dimension, First Differential Mean, First Differential Max, Second Differential Mean, and Second Differential Max. In the frequency domain, Bandpower ratio Theta/Alpha, Bandpower ratio (Delta+Theta)/Alpha and Shanon Entropy are extracted. In wavelet domain Mean, Standard Deviation, Energy and Entropy are extracted for both the detailed and approximate wavelet coefficients. We have implemented an AR model of order 3 using the Burg algorithm [23] and used the AR coefficients as features. The descriptions of all these features are given in [12] except Fractal Dimension and Shanon Entropy which can be found in [24] and [25] respectively.

E. Deep learning feature extraction

There are two steps for this feature extraction. At the first step, a deep learning network for feature extraction has to be trained using training dataset. In the second step, the trained deep learning model is used for feature extraction from testing dataset. In this work, a very well-known deep network, Stacked Denoising Autoencoder (SDAE) [21] is used.

The basic building block of an SDAE is Denoising Autoencoder (DAE). A DAE takes an input x, and through a nonlinearity function maps it to a feature representation z as $z = f(Wx + b)$. This part is known as the encoding. This feature representation z is then mapped back into a reconstruction y in the same shape of x as $y = f(W'z + b')$, known as decoding. If two or more DAEs are stacked, then it gives a deep neural network, known as SDAE [21], shown in Fig 2. Generally, the weight matrix of reverse mapping W', is set as $W' = W^T$, known as tied weights. The main intention of SDAE is to learn the latent features from the input data while avoiding the influence of noise and if necessary to reproduce the input data back from the latent features. Here, the latent features are the output of the encoder part and reconstructed input is the output of the decoder part of the network.

The parameters of this deep network are trained using training dataset such that the reconstruction error between the given input and reconstructed output is minimized. This error is measured using the squared error. After the training completion, the whole decoder part is removed from the network and the remaining encoder part of the network is used for the feature extraction from testing dataset.

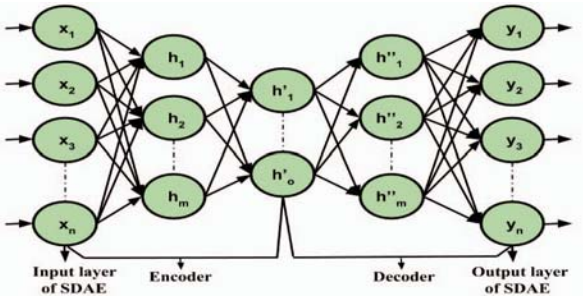


Figure 2: Stacked Denoising Autoencoder

F. Clustering the features

In this work, we have used three clustering techniques to separately cluster the two different set of features discussed above.

1) PAM: Partitioning Around Medoids (PAM) [20] is a centroid-based clustering approach where the statistical value called medoid of a cluster is considered as its cluster center. It initiates by choosing k number of data points as the initial medoids to represent k clusters. Then the other data points get assigned to its closest medoid. The algorithm iterates to alter the medoids locations for the improvement of cluster quality. Accordingly, the other data points also get relocated to their nearest medoids. This process continues until there is no change in the medoids location. The main advantage of PAM is its robustness against noise and outliers.

2) BIRCH: Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) [20] is a hierarchical clustering strategy, able to incrementally and dynamically cluster incoming data points for large databases with effective use of memory. It represents a cluster by a three-dimensional vector known as *Cluster feature* (CF) defined as $CF =< n, LS, SS >$. Here, n is the number of data points in the cluster, LS is the linear sum of data points and SS is the square sum of the data points. BIRCH works in two phases. In the first phase, it builds a CF tree and in the second phase, it clusters the leaf nodes of the tree. This clustering technique has two main user-defined parameters *threshold*(th) and *branching factor*(bf). They define the maximum number of entries in each leaf node and in each non-leaf node of the CF tree respectively. It is robust against noise and outliers.

3) DBSCAN: Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [20] is a density-based clustering algorithm which introduces two concepts, *core object*, and *density reachability*. If within a radius (say ϵ) of a given data point contains at least a minimum number (say $MinPts$) of data points then the given data point is considered as core object. A data point (say p) is *directly density-reachable* from another data point (say q) if p is within the radius of ϵ of q and q is a core object. A data point (say p) is density-reachable from another data point (say q) if there is a sequence of data points starting from q and ending at p, where every data point in this sequence is directly reachable from its predecessor. DBSCAN identifies a cluster in two phases. First, it hunts for a core point and then draws all the data points which are density-reachable from it forming a cluster including the core point. The main advantage of this technique is that it can discover arbitrarily shaped clusters.

III. Experimental Results

We have done clustering of both handcrafted and deep learning features separately for each trial data of every subject using the above mentioned three clustering techniques. The qualities of the subsequent clusters are measured using three intrinsic clustering validation metrics such as Silhouette score, Dunn index, and Davies-Bouldien index and as well as with three extrinsic clustering validation metrics such as Rand index, Jaccard index and Fowlkes-Mallow index [26].
Due to the page limit, clustering results for single trial data of one male and one female subjects are given in this paper. The parameters used for PAM, BIRCH and DBSCAN clustering and the consequent obtained validation metrics for both the machine learning (ML) and deep learning (DL) feature sets are tabulated in Table I, Table II and Table III respectively along with their corresponding scatter plots shown in Figure 3, Figure 4 and Figure 5 respectively. In the case of PAM clustering, we fixed the parameter ”number of clusters” at 3 for all the subjects as we have collected the data with respect to three different cognitive loads.

IV. Discussion

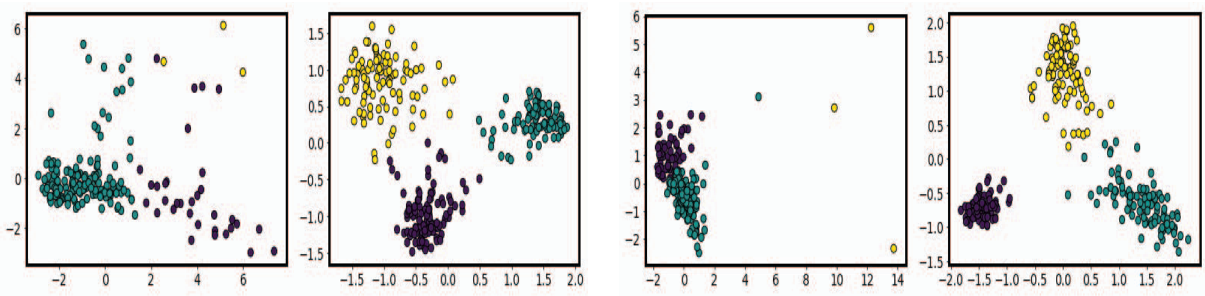
In this work, both the feature extraction process takes place in the unsupervised way that is without any class information of the data. The process of handcrafted feature extraction does not require class information. The deep network we used for feature extraction, that is SDAE, is an unsupervised network which does not require any class information for the training. It can train itself from the error generated between the given input and resultant output (the reconstructed version of the input). After that, the encoder part of the network becomes capable of the task of feature extraction. Thus, in the time of feature extraction, it can not ensure the relevancy of the extracted features with respect to the class. But the above-mentioned clustering results and scatter plots are showing that the deep network features are more relevant than the handcrafted features. A feature set is more relevant to the class information means that it has more discrimination power among different classes and within the same class, they are more similar to each other. From the different clustering indices and scatter plots of the formed clusters, it is clearly visible that the qualities of the deep network feature clusters are better than that of the handcrafted feature clusters. Thus, it verifies the fact that deep network features are the more relevant one comparison to the handcrafted ones. Consequently, when the deep network features are used for classification, it performs better than the handcrafted features.

V. Conclusion

In this study, we have analyzed the differences between deep learning features and handcrafted features used in machine

Table I: PAM clustering results

Subject No.	Feature type	Silhouette score	Dunn index	Davies Bouldien index	Rand index	Jaccard index	Fowlkes Mallows index
$M1_{t1}$	ML	0.27	0.27	1.39	0.03	0.23	0.50
	DL	0.33	0.30	1.25	0.83	0.94	0.89
$F1_{t1}$	ML	0.07	0.24	2.5	0.15	0.54	0.52
	DL	0.42	0.37	1.05	0.94	0.98	0.96

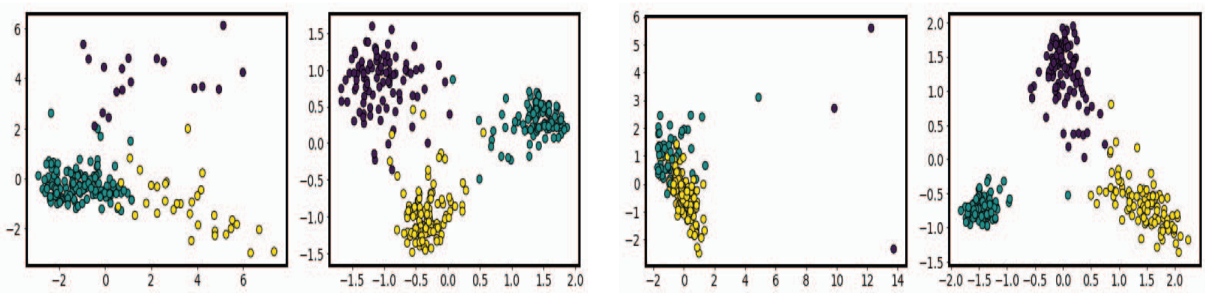


(a) $M1_{t1} - ML - PAM$ (b) $M1_{t1} - DL - PAM$ (c) $F1_{t1} - ML - PAM$ (d) $F1_{t1} - DL - PAM$

Figure 3: Scatter plots obtained for PAM clustering

Table II: BIRCH clustering results

Subject No.	Feature type	Parameters (th,bf)	Silhouette score	Dunn index	Davies Bouldien index	Rand index	Jaccard index	Fowlkes Mallows index	Scatter plot
$M1_{t1}$	ML	0.04,56	0.24	0.31	2.43	0.03	0.32	0.45	0
	DL	0.84,91	0.33	0.33	2.26	0.86	0.35	0.91	0
$F1_{t1}$	ML	0.75,66	0.07	0.25	2.30	0.11	0.25	0.50	0
	DL	0.93,42	0.42	0.36	1.06	0.93	0.97	0.95	0

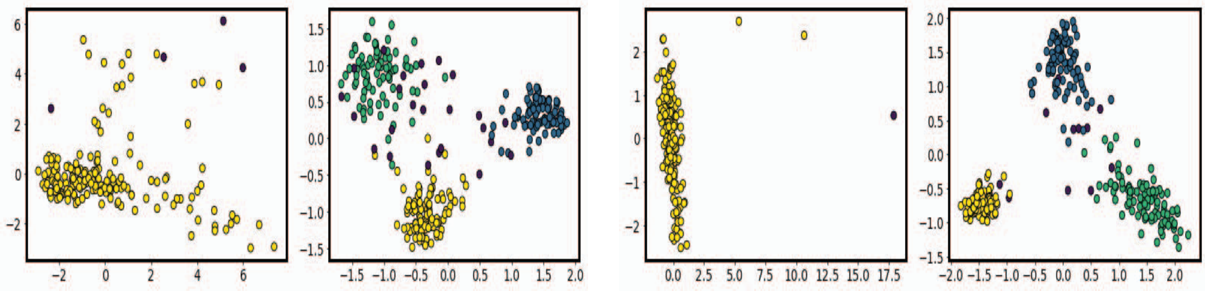


(a) $M1_{t1} - ML - BIRCH$ (b) $M1_{t1} - DL - BIRCH$ (c) $F1_{t1} - ML - BIRCH$ (d) $F1_{t1} - DL - BIRCH$

Figure 4: Scatter plots obtained for BIRCH clustering

Table III: DBSCAN clustering results

Subject No.	Feature type	Parameters (e,MinPts)	Silhouette score	Dunn index	Davies Bouldien index	Rand index	Jaccard index	Fowlkes Mallows index
$M1_{t1}$	ML	7.64,20	0.32	0.5	1.03	0.05	0.45	0.56
	DL	1.71,11	0.47	0.61	1.00	0.78	0.78	0.85
$F1_{t1}$	ML	12.56,13	0.37	0.55	1.00	0.03	0.34	0.57
	DL	1.46,11	0.67	0.88	1.00	0.87	0.4	0.91



(a) $M1_{t1} - ML - DBSCAN$ (b) $M1_{t1} - DL - DBSCAN$ (c) $F1_{t1} - ML - DBSCAN$ (d) $F1_{t1} - DL - DBSCAN$

Figure 5: Scatter plots obtained for DBSCAN clustering

learning classification algorithms with the help of three clustering methods PAM, BIRCH, and DBSCAN. The two different feature sets are clustered separately using these three clustering techniques. The results of all the three clustering techniques have shown that the deep network feature clusters quality is better than of the handcrafted feature clusters. Deep network feature clusters are more compact and dissimilarity between two different clusters is more comparable to the handcrafted feature clusters. This demonstrates the fact that deep network features are more relevant than the handcrafted ones with respect to the class information of data. Thus its performance for the task of classification in a BCI system is superior to the conventional machine learning algorithms.

References

[1] R. H. Abiyev, N. Akkaya, E. Aytac, I. Günsel, and A. Çağman, "Brain-Computer Interface for Control of Wheelchair Using Fuzzy Neural Networks," *BioMed Research International*, vol. 2016, 2016.

[2] S. Sreeja, V. Joshi, S. Samima, A. Saha, J. Rabha, B. S. Cheema, D. Samanta, and P. Mitra, "BCI Augmented Text Entry Mechanism for People with Special Needs," in *International Conference on Intelligent Human Computer Interaction*. Springer, 2016, pp. 81–93.

[3] M. Bamdad, H. Zarshenas, and M. A. Auais, "Application of BCI systems in neurorehabilitation: a scoping review," *Disability and Rehabilitation: Assistive Technology*, vol. 10, no. 5, pp. 355–364, 2015.

[4] K. K. Ang and C. Guan, "Brain-Computer Interface for Neurorehabilitation of Upper Limb After troke," *Proceedings of the IEEE*, vol. 103, no. 6, pp. 944–953, 2015.

[5] A. Riccio, F. Leotta, L. Bianchi, F. Aloise, C. Zickler, E. Hoogerwerf, A. Kübler, D. Mattia, and F. Cincotti, "Workload measurement in a communication application operated through a P300-based Brain-Computer Interface," *Journal of Neural Engineering*, vol. 8, no. 2, p. 025028, 2011.

[6] H. S. AlZu'bi, W. Al-Nuaimy, and N. S. Al-Zubi, "EEG-based Driver Fatigue Detection," in *Developments in eSystems Engineering (DeSE), 2013 Sixth International Conference on*. IEEE, 2013, pp. 111–114.

[7] R. Adams, R. Comley, and M. Ghoreyshi, "The Potential of the BCI for Accessible and Smart e-Learning," in *International Conference on Universal Access in Human-Computer Interaction*. Springer, 2009, pp. 467–476.

[8] J. CHANDRA *et al.*, "The Brain Fingerprinting Through Digital Electroencephalography Signal Technique," *International Journal on Computer Science and Engineering*, vol. 3, no. 3, pp. 1086–1090, 2011.

[9] A. Borse *et al.*, "Brain Fingerprinting," *International Journal of Electronics, Electrical and Computational System*, vol. 4, 2015.

[10] M. Ahn, M. Lee, J. Choi, and S. C. Jun, "A Review of Brain-Computer Interface Games and an Opinion Survey from Researchers, Developers and Users," *Sensors*, vol. 14, no. 8, pp. 14 601–14 633, 2014.

[11] R. A. Ramadan, S. Refat, M. A. Elshahed, and R. A. Ali, "Basics of Brain Computer Interface," in *Brain-Computer Interfaces*. Springer, 2015, pp. 31–50.

[12] J. Rabha, K. Nagarjuna, D. Samanta, P. Mitra, M. Sarma *et al.*, "Motor Imagery EEG Signal Processing and Classification Using Machine Learning Approach," in *2017 International Conference on New Trends in Computing Sciences (ICTCS)*. IEEE, 2017, pp. 61–66.

[13] A. Sivakami, S. S. Devi *et al.*, "Analysis of EEG for Motor Imagery Based Classification of Hand Activities," *International Journal of Biomedical Engineering and Science*, vol. 2, no. 3, pp. 11–22, 2015.

[14] P. Zarjam, J. Epps, and F. Chen, "Spectral EEG features for evaluating cognitive load," in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*. IEEE, 2011, pp. 3841–3844.

[15] D. Das, D. Chatterjee, and A. Sinha, "Unsupervised approach for measurement of cognitive load using EEG signals," in *Bioinformatics and Bioengineering (BIBE), 2013 IEEE 13th International Conference on*. IEEE, 2013, pp. 1–6.

[16] L. Vidyaratne, A. Glandon, M. Alam, and K. M. Iftekharuddin, "Deep recurrent neural network for seizure detection," in *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016, pp. 1202–1207.

[17] N. Lu, T. Li, X. Ren, and H. Miao, "A Deep Learning Scheme for Motor Imagery Classification based on Restricted Boltzmann Machines," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 6, pp. 566–576, 2017.

[18] H. Xu and K. N. Plataniotis, "Affective states classification using EEG and semi-supervised deep learning approaches," in *Multimedia Signal Processing (MMSP), 2016 IEEE 18th International Workshop on*. IEEE, 2016, pp. 1–6.

[19] A. Saha, V. Minz, S. Bonela, S. Sreeja, R. Chowdhury, and D. Samanta, "Classification of EEG signals for Cognitive Load Estimation Using Deep Learning Architectures," in *International Conference on Intelligent Human Computer Interaction*. Springer, 2018, pp. 59–68.

[20] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*. Elsevier, 2011.

[21] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*. MIT press Cambridge, 2016, vol. 1.

[22] W. Dubitzky, M. Granzow, and D. P. Berrar, *Fundamentals of Data Mining in Genomics and Proteomics*. Springer Science & Business Media, 2007.

[23] J. P. Burg, "A New Analysis Technique for Time Series Data," *Paper presented at NATO Advanced Study Institute on Signal Processing, Enschede, Netherlands, 1968*, 1968.

[24] K. Ansari-Asl, G. Chanel, and T. Pun, "A channel selection method for EEG classification in emotion assessment based on synchronization likelihood," in *2007 15th European Signal Processing Conference*. IEEE, 2007, pp. 1241–1245.

[25] D. Q. Phung, D. Tran, W. Ma, P. Nguyen, and T. Pham, "Using Shannon Entropy as EEG Signal Feature for Fast Person Identification," in *ESANN*, vol. 4, no. 1, 2014, pp. 413–418.

[26] Z. Ansari, M. Azeem, W. Ahmed, and A. V. Babu, "Quantitative Evaluation of Performance and Validity Indices for Clustering the Web Navigational Sessions," *arXiv preprint arXiv:1507.03340*, 2015.