Classification of Fatigue in Consumer-grade EEG Using Entropies as Features

Muhammad Azam, Derek Jacoby, and Yvonne Coady

Department of Computer Science, University of Victoria,
Victoria, Canada

Abstract. Electroencephalogram (EEG) records electrical activity at different locations in the brain. It is used to identify abnormalities and support the diagnoses of different disease conditions. The accessibility of low-cost EEG devices has seen the analysis of this data become more common in other research domains. In this work, we assess the performance of using Approximate entropy, Sample entropy, and Reyni entropy as features in the classification of fatigue from EEG data captured by a MUSE 2 headset. We test 5 classifiers: Naive Bayes, Radial Basis Function Network, Support Vector Machine, K-Nearest Neighbor, and Best First Decision Tree. We achieved the highest accuracy of 77.5% using the Support Vector Machine classifier and present possible enhancements to improve this.

Keywords: EEG, Electroencephalogram, Approximate entropy, Sample entropy, Reyni entropy, Fatigue detection, Automatic classification, MUSE 2

1 Introduction

Fatigue has become a prominent issue in our society and more people are starting to suffer from 'burnout'. It is defined as a physiological state of reduced mental or physical capability [1]. When a person experiences a fatigue condition and is operating at a reduced capability, this can lead to accidents and mistakes which can have very real consequences. This is apparent in fatigue-related traffic accidents, where research has shown that 24 hours of prolonged wakefulness has similar adverse effects on cognitive performance to a blood alcohol level of 100 mg per 100 ml of blood [2]. This exceeds the legal limit in British Columbia, Canada by 25%.

Hospitals are known to be stressful and demanding work environments. Staff often work long shifts, which can sometimes be back-to-back. The effects of reduced cognitive and physical performance in such a workplace, can have serious consequences for patient health, especially as workers experiencing constant acute fatigue will likely exhibit more residual fatigue over time.

Yet with all of the adverse effects, fatigue is not monitored, diagnosed, or treated in the same way as other health risks such as obesity or diabetes. Part of the reason for this could be determining and monitoring it is not straightforward. For diabetics, test strips provide a fast and convenient way to measure blood sugar

David C. Wyld et al. (Eds): ICAITA, CDKP, SAI, NCO, CMC, SOFT, MLT, AdNLP - 2020
pp. 13-26, 2020. CS & IT - CSCP 2020 DOI: 10.5121/csit.2020.100902

levels, allowing for effective management. For patients experiencing obesity, there are numerous ways of measurement, and well researched practices for weight loss.

For fatigue, there is no simple device that one can go out and buy, and no straightforward method that can be used at home to help manage it. In order for such a device to be effective, it would be critical that it not be invasive, be able to accurately detect fatigue levels, track trends, and perhaps be adaptable for a treatment plan.

This work looks at addressing the detection component. The first step is to create a computerized and automated process that can consume EEG data and provide a binary classification of whether the condition is fatigued or normal. If successful, this methodology may be adaptable to determine not only the current condition, but also the level of fatigue accumulation. A reduction in cognitive performance would likely alter the electrical activity in the brain, making EEG signals ideal for this analysis. In addition, non-clinical devices such as the MUSE 2 headset used to collect the data used in this work, are becoming more common and cost-effective to acquire.

The approach taken adapts a method used in [3] to determine focal and non focal EEG signals for identification of epileptic disease conditions using entropies.

This paper is organized as follows: Section 2 provides references to related work that were consulted, Section 3 provides a detailed description of the data used, a brief overview of the entropies used as features, as well as some information about the classifiers used in the training process. Section 4 presents the results that were obtained through this process. Section 5 provides a discussion on the performance, as well as possibilities for improving the classification process. Finally, the conclusion is given in Section 6.

2 Related Work

To determine an effective approach in detection of fatigue in EEG signals, a number of works aimed at producing features from EEG signals and classifying them were consulted. In [3], three entropy features, Approximate entropy, Sample entropy, and Reyni entropy, were extracted from the Bern Barcelona EEG dataset, and 5 classifiers were trained using a subset of the input data to detect focal and non-focal EEG nodes to detect underlying epileptic conditions. The performance of the classifiers was then tested with novel data and determined to be over 90%, with the best performance of 98% being for a Non-Nested Generalized Exemplars (NNGE) classifier. The dataset used in [3] is described in detail in [4]. Srinivasan et al. have also proposed using Approximate entropy, but in conjunction with artificial neural networks to classify epileptic conditions in EEG data [5]. Other methods using neural networks [6] as well as signal decomposition techniques [7] have also been used in epilepsy detection.

Further examples of detecting epileptic conditions in EEG are also available. In [8], Acharya et al. perform analysis on EEG data using Approximate entropy, Sample entropy, and Phase entropies (S1 and S2) as features. Additional classifiers are used in this work, with the highest performance in accuracy of 98.1% being attributed to the Fuzzy Sugeno Classifier. Research has also been conducted on quantifying depth of anesthesia [9] as well as coma [10] and unresponsive [11] states, using a combination of EEG data and extracted entropy features.

In the domain of sleep stage classification, Rodríguez-Sotelo et al. [12] show that unsupervised learning techniques to automatically classify sleep stages using Shannon entropy, Approximate entropy, Sample entropy, and Multiscale entropy (in addition to a number of features that draw from the complexity of a signal) work well. A comparison of classification accuracy on the features showed that Approximate and Sample entropy performed at 74% and 73% respectively.

Driver fatigue detection and correction is also a common application of fatigue detection in EEG data. Huang et al. [13] go a step beyond the detection of fatigue in simulated driving conditions, and at the onset of detected fatigue, provide a warning signal to the driver. This has the effect of improved behavioural performance following the signal, but a reduction in efficacy is also shown with repeated signals. To detect the onset of fatigue, the authors analyze the EEG power (alpha band) and a high value exceeding a pre-determined threshold signifies that the driver is approaching a fatigue state. The threshold was determined experimentally through baseline measurements for each participant.

Application of wireless headsets for detecting and providing real-time feedback on high-speed train driver fatigue are shown in [14]. The method presented uses a Fast Fourier Transform (FFT) feature extraction combined with a Support Vector Machine (SVM) classifier to feed the detection model. Alerts are then presented if the onset of fatigue is detected. Classification accuracy is shown to achieve between 77% and 96% accuracy. FFT features are widely used in the literature to extract features from signals for classification models.

Much of the literature showcases examples of classification using different techniques/features but with the use of medical grade EEG devices. In this work, we evaluate the use of entropy as features in automatically detecting a fatigue condition using a device that has limited nodes and configuration options, but is more suitable for general at-home use. Our goal is to evaluate whether reliable enough performance can be achieved, that make the MUSE 2 headset a viable fatigue tracking tool.

3 Proposed Approach

This section describes the dataset used for this work, the features extracted, as well as a brief overview of the classifiers used.

3.1 Data

The data used for this work was obtained from Krigolson Labs at the University of Victoria [15]. The data was captured using a MUSE 2 headset from participants before and after working a 12 hour shift in a hospital environment. The data collection was performed in 2018 for 15 participants, and in 2019 for 33 participants (2 observations are missing in the dataset for the fatigue condition for this year). EEG recordings taken with the headset before the shift are ground truth labeled as 'normal' condition, and recordings after the shift was complete were classified as 'fatigue' condition. The recordings all vary in length from 41,988 - 129,132 samples. At the sampling frequency of 256 Hz, these recordings fall between approximately 2.7 - 8.4 minutes of signal data. The MUSE 2 headset has 4 electrodes and, since these are fixed in the device with no options for different device sizes, the locations of readings are likely non-standard [16].

The raw data was contained in .csv format and an example signal for normal condition is shown in Figure 1. Preprocessing was required before data was appropriately usable, including header corrections, fixing junk values, format transformations, and mean removal. No noise removal or filtering techniques were applied to the data.

An additional step of data preparation was necessary before the feature extraction process. This involved converting each 4 channel signal into 4 individual signals. In related literature, readings from the electrode of an EEG device are often taken as individual signals for classification purposes. An example of this is shown in [3]. Segmentation was also applied to the data to transform the signals into a fixed length of 10,240 samples, or 40 seconds at the sampling frequency. Each signal that was divided into the subsets also had the same ground truth label applied. The effect of this preparation step greatly increased the dataset size to 1,244 signals of fatigue condition and 1,360 signals of normal condition.

3.2 Feature Extraction

In machine learning, pattern recognition, and image processing tasks, feature extraction is a key component to training a system well so that it is able to recognize and classify novel data accurately. Features are mathematical descriptors of the data that lend well to the application of classifier networks that use statistical techniques, or in the more complex case, deep learning.

Some simpler features that can be extracted from signals include mean, standard deviation, minimums, and maximums. The frequency information of a signal can also be used to extract quality features. It is often the case that raw data values are simply too large to process. For example, in the case of each signal used here there would be 10,240 features, so through careful feature extraction and selection the number of features describing our data is reduced down to just three. In this work,

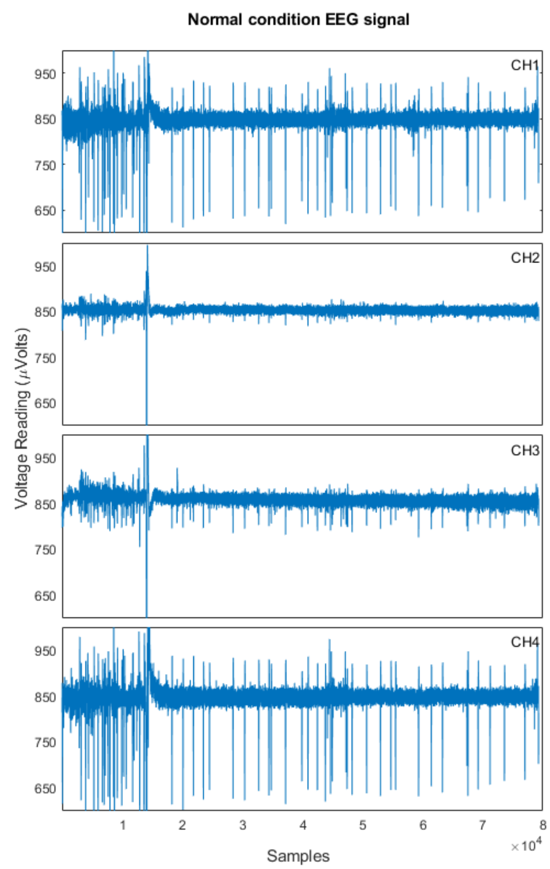

Fig. 1. Normal condition raw signal data

we will use three entropies as features of the data: Approximate entropy, Sample entropy, and Reyni entropy. Entropy have been used extensively in the literature to perform classification of signals in various domains.

3.3 Approximate entropy

Approximate entropy is a statistical technique that is used to measure the regularity of a signal. Low Approximate entropy is a reflection of the persistent, repetitive, and predictive nature of a series with apparent patterns that repeat themselves [17]. The value of using this entropy is that it can distinguish between series that have the same moment statistics (mean, standard deviation, etc.) but may be more different in random probability. A completely random and unpredictable signal will produce a high value of Approximate entropy, i.e. it is chaotic. A more regular and predictable series will yield a low value. To calculate this value, an integer m defines the length of the window size and r specifies the comparison filtering level. The algorithm to determine the approximate entropy for a given time series is as follows:

Step 1: Form a time series of data with length **N**:

$$u(t) = u(1), u(2), u(3), ..., u(N) \tag{1}$$

Step 2: From the time series, form a sequence of vectors each of length m:

$$\vec{x(1)}, \vec{x(2)}, \vec{x(3)}, ..., x(N - \vec{m} + 1)$$

where

$$\vec{x(1)} = \begin{bmatrix} u(1) \\ u(2) \\ ... \\ u(m) \end{bmatrix}, \vec{x(2)} = \begin{bmatrix} u(2) \\ u(3) \\ ... \\ u(m+1) \end{bmatrix}, \vec{x(3)} = \begin{bmatrix} u(3) \\ u(4) \\ ... \\ u(m+2) \end{bmatrix} ...$$

and generally

$$\vec{x(i)} = \begin{bmatrix} u(i) \\ u(i+1) \\ ... \\ u(i+m-1) \end{bmatrix} \tag{2}$$

Step 3: Calculate the maximum distance between a vector $\vec{x(i)}$ and all other $\vec{x(j)}$ in the set to create a distance value vector:

$$\vec{d(i)} = \begin{bmatrix} max(dist|\vec{x(i)}, \vec{x(1)}|) \\ max(dist|(\vec{x(i)}, \vec{x(2)}|) \\ ... \\ max(dist|\vec{x(i)}, (x(N - \vec{m} + 1)|) \end{bmatrix} \tag{3}$$

Step 4: Calculate the number of values in each distance vector that are within the filtering criterion m to get a ratio C for each:

$$C_i^m(r) = \frac{\sum (\vec{d(i)} \leq m)}{N - m + 1} \tag{4}$$

Step 5: Define ϕ as the negative logarithmic sum of ratio values:

$$\phi^m(r) = \frac{\sum_{i=1}^{N-m+1} \log C_i^m(r)}{N - m + 1} \tag{5}$$

Step 6: Increase m by 1 and repeat the process to get:

$$\phi^{m+1}(r)$$

Step 7: Approximate entropy (AE) is defined as the difference of these two:

$$AE(m,r) = \phi^{m}(r) - \phi^{m+1}(r) \tag{6}$$

Typically, the initial value of m is taken as 2 or 3, and the value of r greatly depends on the data and application domain. In this work, we set m equal to 2, and the filtering parameter r equal to 3. A full detailed description and analysis of Approximate entropy is provided in [17].

There are two important implications to using Approximate entropy: the first is that relative consistency is not guaranteed as a different filtering parameter of r may yield entirely different results. The second is that this entropy calculation depends heavily on the length of the series. Approximate entropy is also biased towards suggesting more regularity than there actually is in the signal.

3.4 Sample entropy

Sample entropy is a modification of Approximate entropy that addresses the issues mentioned above with Approximate entropy. This statistical technique excludes self-counting, maintains relative consistency and is mostly independent of the length of the series [17].

The Sample entropy algorithm requires as input a window size m and filtering parameter r. It is calculated through the following steps:

Step 1: Form a time series of data with length **N**:

$$u(t) = u(1), u(2), u(3), ..., u(N) \tag{7}$$

Step 2: Form two sets of vectors as follows:

$$\vec{x_m}(i) = \begin{bmatrix} u(1) & ... & u(N-m) \\ ... & ... & ... \\ u(m) & ... & u(N-m+1) \end{bmatrix}$$

$$\vec{x_m}(j) = \begin{bmatrix} u(1) & ... & u(N-m+1) \\ ... & ... & ... \\ u(m) & ... & u(N) \end{bmatrix}$$

Step 3: Calculate the number of vector pairs at a distance less than r as:

$$B = [\sum_{1}^{N-m} count(dist|\vec{x_m}(i), \vec{x_m}(j)|) < r] - 1 \tag{8}$$

Step 4: Increase m by 1, and repeat the process to get:

$$A = [\sum_{1}^{N-m+1} count(dist|\vec{x_{m+1}}(i), \vec{x_{m+1}}(j)|) < r] - 1 \tag{9}$$

Step 5: The sample entropy (SE) is then defined as:

$$SE(m,r) = -\log \frac{A}{B} \tag{10}$$

Both A and B represent probabilities that two sequences are similar. Detailed analysis and comparison of Approximate and Sample entropy is provided in [17].

3.5 Reyni entropy

Reyni entropy (RE) is a generalization on a number of other entropies, specifically the Hartley entropy, Shannon entropy, the collision entropy, and the min-entropy. It is given by:

$$RE(\alpha) = -\frac{\alpha}{1-\alpha} \sum \log p_k^\alpha \tag{11}$$

where

$$\alpha \geq 0, \alpha \neq 0$$

With specific values of α, Reyni entropy simplifies to one of the above mentioned types.

3.6 Classification

Once features have been extracted from the data, we use a subset to train 5 different classifiers. This training process yields a parameter-based model for each that can be used to predict values on novel data. The remaining unused data is then fed into each of the classifiers, and the predicted output labels are matched to ground truth labels to determine performance. A brief overview of the classifiers used is provided in this section.

3.7 Naive Bayes (NB)

The Naive Bayes family of classifiers works by assuming that each of the input features are independent variables: that is, that each input feature is not dependent on any other feature. While this is generally a poor assumption to make since features often have some relationship or dependence describing the problem, these classifiers work surprisingly well in practice. The Bayes model works on conditional probability, determining an outcome class based on some number of independent variables (the input features) [18].

3.8 Radial Basis Function Network (RBF)

A Radial Basis Function is a function whose value depends only on the distance between the input and some fixed point. The RBF network is a simple single-layer artificial neural network where the RBF is used as the activation function. As RBF's operate on distance approximation, the data was modified to represent the fatigue and normal states as absolute distance values. Since the output of this network produces a real-valued distance approximation, results of the classified test data had to be converted back to labels.

3.9 K-Nearest Neighbour (KNN)

This classifier works on grouping incoming data points into the member group of the nearest neighbour. The computation uses distance values to determine which nearest neighbour to group to. Different types of distances can be used, but experimentally we determined Euclidean distance to perform optimally.

3.10 Support Vector Machines (SVM)

Support Vector Machine classifiers work by attempting to draw an optimal hyper-plane that separates clusters of data points belonging to the same member groups (classification types). Support vectors are then drawn around the plane. The performance of this classifier is highly dependent on if the drawn hyper-plane can effectively separate points representing different classifications, and by the width of the support vectors around this plane. A larger width means that the data is effectively separable.

3.11 Best First Decision Tree (BFDT)

A decision tree is a set of conditional control statements where each node represents a "test" on an attribute. The branch from each node represents the outcome of the node. The paths from root to leaf (end node) represent the classification rules used in this classifier [19].

3.12 Performance Measures

Performance on novel test data for each of the classifiers is determined by 6 different measures. These measures are calculated from Type I and II errors (false positive (FP) and negative(FN)) as well as true positive (TP) and negative (TN) predicted outcomes. Here, positive refers to the fatigue condition and negative refers to the normal condition. The measures are:

Accuracy (ACC):

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

Sensitivity (SEN):

$$\frac{TP}{TP + FN} \tag{13}$$

Specificity (SPF):

$$\frac{TN}{TN + FP} \tag{14}$$

Positive Predictive Value (PPV):

$$\frac{TP}{TP + FP} \tag{15}$$

Negative Predictive Value (NPV):

$$\frac{TN}{TN + FN} \tag{16}$$

Matthews Correlation Coefficient (MCC):

$$\frac{(TP * TN) - (FN * FP)}{\sqrt{(TP + FN)(TP + FP)}\sqrt{(TN + FN)(TN + FP)}} \tag{17}$$

4 Experimental Results

Calculations of Approximate, Sample, and Reyni entropy values were obtained directly from the prepared 2,604 signals each of length 10,240 samples. Each of resulting 3 x 1 feature vectors per signal had a corresponding ground truth label of 'Normal' or 'Fatigue' condition attached. This was the final state of the data before the training process. To train the classifiers, the input data rows were randomized, and separated into a training set comprising 80% of the data, and a test set which had the remaining 20%.

Each of the classifiers was given the same set of randomly prepared training data, and hyper-parameter optimization was used to determine the best performing parameter settings. This process yielded a trained model for each classifier. Finally, each of the classifiers was given the 3 x 1 feature vectors from the earlier prepared test data, yielding predicted labels for each. These predicated labels were compared to the ground truth values for that data to determine the level of performance. Resulting performance of each classifier is shown in Table 1.

Table 1
Results of classifiers in detecting 'Fatigue' condition

Classifier	ACC	SEN	SPF	PPV	NPV	MCC
Naive Bayes	72.6	46	99.6	99.1	64.4	0.54
Radial Basis Network	70.1	76.7	64.4	64.8	76.4	0.41
Support Vector Machine	77.4	47.5	99.7	99.1	71.7	0.58
K-Nearest Neighbor	77.5	63	90.1	86.3	72.9	0.56
Best First Decision Tree	68.9	53.9	82.5	73.5	66.5	0.38

The results show that the classification accuracy is highest for the SVM and KNN classifiers at about 77%. Values for the other measures are mixed, and show that the classifiers are achieving lower than expected performance.

5 Discussion

In [3], the authors show that using these three entropies as features yields results of over 90% for all of these classifiers in detecting focal and non-focal EEG. That is, EEG channel readings where an underlying epileptic condition exists (focal) and where it does not (non-focal). In theory, this method should apply well to the analysis done here. There are likely a few issues preventing the classifiers from performing better.

The data used for this work was of raw EEG data containing 4 channels. Without additional preprocessing, especially noise removal, it is likely that the entropy algorithms were negatively affected. As shown in Figure 1, significant noise and artefacts appear in the signal which may be removed or suppressed through the application of signal processing techniques. The dataset used in [8] was obtained from the Department of Epileptology at the University of Bonn and artefact free, further providing evidence that EEG data cannot be used in its raw recorded format effectively for feature extraction.

Table 2
Mean and variance of entropy features

Feature	Fatigue	Normal
Approximate Entropy	1.223 ± 0.4823	1.079 ± 0.5581
Sample Entropy	1.148 ± 0.5184	0.951 ± 0.5680
Reyni Entropy	10.85 ± 6.477	13.68 ± 5.669

A statistical analysis of mean and variance of the calculated features is shown in Table 2. As is evident from this, while the mean value is different for each of the conditions in each of the entropy values, the standard deviation is too large for an effective linearly separable solution. This is likely a major contributor to the performance outcomes seen. When calculating the features, the parameters used can also have significant impact on the outcome value. In the case of Approximate and Sample entropy, while the dimensional parameter is usually kept the same, the filtering parameters depend heavily on the application and signal type. It is possible that the values used during calculations were not appropriate and may require further research to determine.

Data quality issues are likely also present. The MUSE headset operates differently than clinical EEG devices. The headset has fixed contact locations, which may not be ideal for getting optimal readings. The manner in which the headset was worn by the participant could also have affected the recordings. In [16], the authors show some of the challenges involved with data collection using a MUSE headset which can extend to even the amount of hair a participant has affecting readings. Participants themselves could have started the shift already fatigued at the time of the normal condition recording. Different participants may have had varying levels of residual fatigue affecting the observation. Research-grade EEG devices often have many more electrodes available and can be manually adjusted to get optimal readings. This is not the case with the MUSE headset. The utilization of the 4 EEG channels differently could also lead to better results as the process of converting 4 channels to 4 signals required a significant assumption to be made: that electrical activity in the brain should change everywhere under fatigue conditions. This could be a false assumption, and it is possible that certain locations in the brain may be better markers of fatigue than others.

Finally, it is possible that these entropy features are not well suited to detecting fatigue condition from EEG signals. While entropies have been used in the literature to detect various brain conditions such as epilepsy, adaptations may be required for the problem of fatigue detection. Further research using other features such as instantaneous frequency, spectral entropy, and wavelet transformations may yield better results. In [12], power and complexity analysis techniques such as Fractal Dimension actually show a better performance in unsupervised classification tasks. These may also improve the performance of fatigue detection. With the remarkable performance of deep learning networks, using a 2-dimensional spectrogram of the signal in conjunction with a convolutional neural network may also yield better results. Techniques such as this have been performed on audio signals for applications such as music classification [20].

6 Conclusion

Fatigue is becoming more common in society, and as we become more aware in the impacts it has on our daily lives, we are also looking for better ways to manage or reduce it. For non-invasive and easy to use devices that can be used at home, effective detection and measurement of fatigue is critical. In this work, we used entropy features to train classifiers for detecting fatigue conditions in EEG signals automatically. We achieved the highest performance of 77.5% accuracy using the Support Vector Machine classifier. This performance can likely be improved through better signal processing on the raw data to remove noise and artefacts, as well as by tuning the entropy feature calculations using domain-specific parameters.

Further research on the work presented here may look at binary classification of fatigue using different devices or by comparing the effectiveness of different features. Deep learning techniques may also perform well. Following the development of a binary classification model, the method may be adapted to determine levels of fatigue, given individual baselines. More extensive data collection and individual level baselines may also aid in determining the effect of fatigue on EEG data. Greater accuracy can then be combined with inexpensive devices and connected apps that help individuals monitor and understand their fatigue levels. This can also have significant impacts on stressful work environments where long hours can impact performance.

7 Acknowledgements

The authors would like to thank Krigolson Labs at the University of Victoria for providing the dataset used in this work.

References

1. Lock, A. M., D. L. Bonetti, and A. D. K. Campbell. "The psychological and physiological health effects of fatigue." Occupational medicine 68.8 (2018): 502-511.
2. Dawson, Drew, and Kathryn Reid. "Fatigue, alcohol and performance impairment." Nature 388.6639 (1997): 235-235.
3. Arunkumar, N., et al. "Classification of focal and non focal EEG using entropies." Pattern Recognition Letters 94 (2017): 112-117.
4. Andrzejak, Ralph G., Kaspar Schindler, and Christian Rummel. "Nonrandomness, nonlinear dependence, and nonstationarity of electroencephalographic recordings from epilepsy patients." Physical Review E 86.4 (2012): 046206.
5. Srinivasan, Vairavan, Chikkannan Eswaran, and Natarajan Sriraam. "Approximate entropy-based epileptic EEG detection using artificial neural networks." IEEE Transactions on information Technology in Biomedicine 11.3 (2007): 288-295.
6. Yuan, Qi, et al. "Epileptic EEG classification based on extreme learning machine and nonlinear features." Epilepsy research 96.1-2 (2011): 29-38.
7. Subasi, Abdulhamit, and M. Ismail Gursoy. "EEG signal classification using PCA, ICA, LDA and support vector machines." Expert systems with applications 37.12 (2010): 8659-8666.
8. Acharya, U. Rajendra, et al. "Automated diagnosis of epileptic EEG using entropies." Biomedical Signal Processing and Control 7.4 (2012): 401-408.
9. Liang, Zhenhu, et al. "EEG entropy measures in anesthesia." Frontiers in computational neuroscience 9 (2015): 16.
10. Gosseries, Olivia, et al. "Automated EEG entropy measurements in coma, vegetative state/unresponsive wakefulness syndrome and minimally conscious state." Functional neurology 26.1 (2011): 25.

11. Thul, Alexander, et al. "EEG entropy measures indicate decrease of cortical information processing in Disorders of Consciousness." Clinical Neurophysiology 127.2 (2016): 1419-1427.
12. Rodríguez-Sotelo, Jose Luis, et al. "Automatic sleep stages classification using EEG entropy features and unsupervised pattern analysis techniques." Entropy 16.12 (2014): 6573-6589.
13. Huang, Kuan-Chih, et al. "An EEG-based fatigue detection and mitigation system." International journal of neural systems 26.04 (2016): 1650018.
14. Zhang, Xiaoliang, et al. "Design of a fatigue detection system for high-speed trains based on driver vigilance using a wireless wearable EEG." Sensors 17.3 (2017): 486.
15. The Theoretical and Applied Neuroscience Laboratory at the University of Victoria https://www.krigolsonlab.com/
16. Krigolson, Olave E., et al. "Choosing MUSE: Validation of a low-cost, portable EEG system for ERP research." Frontiers in neuroscience 11 (2017): 109.
17. Delgado-Bonal, Alfonso, and Alexander Marshak. "Approximate entropy and sample entropy: A comprehensive tutorial." Entropy 21.6 (2019): 541.
18. Rish, Irina. "An empirical study of the naive Bayes classifier." IJCAI 2001 workshop on empirical methods in artificial intelligence. Vol. 3. No. 22. 2001.
19. Decision Trees: A simple way to visualize a decision https://medium.com/greyatom/decision-trees-a-simple-way-to-visualize-a-decision-dc506a403aeb
20. Costa, Yandre MG, Luiz S. Oliveira, and Carlos N. Silla Jr. "An evaluation of convolutional neural networks for music classification using spectrograms." Applied soft computing 52 (2017): 28-38.

Authors

Muhammad Azam received his Bachelor degree in Electrical Engineering from the University of Victoria. Currently he is pursuing his Master's in Computer Science. His research interests include Artificial Intelligence, Computer Vision, and Computer Graphics.

Derek Jacoby obtained his PhD in Computer Science from the University of Victoria. He has more than 10 years of experience at Microsoft, leading teams in Windows system administration tools, and research on speech recognition. He founded Canada's first community biology lab at the Victoria Makerspace.

Yvonne Coady is a professor of computer science at the University of Victoria where she works with the MOD(ularity) Squad to improve the modularity of system infrastructure software. Her research interests include mixed reality systems, citizen science, advanced modularity across the software stack and distributed clouds, and new programming paradigms and pedagogy for immersive applications.