An Interpretable Deep Learning Framework for Health Monitoring Systems: A Case Study of Eye State Detection using EEG Signals

Amirhessam Tahmassebi†, Jennifer Martin¶, Anke Meyer-Baese†, and Amir H. Gandomi*
†Department of Scientific Computing, Florida State University, Tallahassee, FL, USA
¶Department of Biostatistics and Medical Informatics, University of Wisconsin–Madison, Madison, WI, USA
*Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia
atahmassebi@fsu.edu, jlmartin22@wisc.edu, ameyerbaese@fsu.edu, gandomi@uts.edu.au

***Abstract*—Effective monitoring and early detection of deterioration in patients play an essential role in healthcare. This includes minimizing the number of emergency encounters, reducing the length of hospitalization stay, re-admission rates of the patients, and etc. Cutting-edge methods in artificial intelligence (AI) have the ability to significantly improve outcomes. However, the struggle to interpret these black box models presents a serious problem to the healthcare industry. When selecting a model, the decision to sacrifice accuracy for interpretability must be made. In this paper, we propose an interpretable framework with the ability of real-time prediction. To demonstrate the predictive power of the framework, a case study on eye state detection using electroencephalogram (EEG) signals was employed to investigate how a deep neural network (DNN) model makes a prediction, and how that prediction can be interpreted. The promising results can be used to employ more advanced models in healthcare solutions without any concern of sacrificing the interpretation.**

I. Introduction

United States healthcare spending grew 4.6 percent in 2018, reaching $3.6 trillion or $11,172 per person. As a share of the nation's Gross Domestic Product (GDP), health spending accounted for 17.7 percent. This would promise implementation of more impactful solutions in healthcare via cutting-edge approaches. Real-time health monitoring and early detection of deterioration in patients are excellent subjects to be explored for possible novel approaches. For example, research on cardiac diseases shows that high level of Troponin (above 0.40 ng/ml) in blood could lead to an acute myocardial infarction event [1]. This feature is a result of a blood test from lab which can be used for an early prediction of one the most important adverse events according to US Centers for Medicare Medicaid Services (CMS). Similarly, there are several indicators and physiological signs that can be employed for early warning of serious illnesses and deterioration including airway, breathing, circulation, etc. In addition to this, Internet of Things (IoT) and wearable technologies provide a competent and structured approach to improve the healthcare services in terms of social benefits and penetration as well as cost-efficiency [2]. To this end, implementation of real-time health monitoring systems are developed based on the conclusion of several studies suggesting that better interventions and responses can be employed using early detection of deterioration in patients [3]. As shown in Fig. 1, Cerner Corporation was the market leader with 10.5% market share in license, maintenance and subscription revenues, followed by Microsoft, athenahealth, Allscripts and Oracle. These companies build various predictive models based on their structured and non-structured electronic medical records (EMR). The EMR data include demographics information, historical encounters, medications, clinical conditions, and lab test results, etc. Most of the clinical data requires standardization and transformation via ontology solutions. In this paper, we investigate a case study on eye state detection using EEG signals which are more sophisticated EMR data.

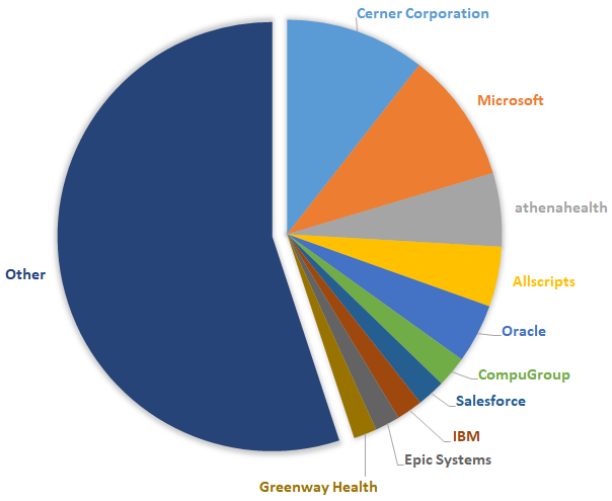


Fig. 1. Healthcare applications market shares split by top 10 healthcare vendors and others in 2018.

In 1875, Richard Caton [4] observed electric activity in the exposed brains of rabbits and monkeys. This sparked use of EEG data in various fields of research. The greatest advantage of EEG data is its high temporal resolution, which can be used to determine the relative strengths and positions of

978-1-7281-2547-3/20/$31.00

electrical activity in different brain regions. Recent years have seen increased interest in using human brain activities as the input of various applications such as sport performance [5], smart health applications [6], neuroergonomics applications [7], and epileptic seizure detection [8]. In addition, designing an implementation setup to employ EEG signals to predict a task has been a point of attention in publications. Various methods including autoregressive and bispectral analysis, common independent component analysis (ICA) (e.g. InfoMax, FastICA, SOBI, JADE) [9], and Fourier-based transformations can be used as an approach for preparing the raw EEG signals and extracting salient features [10]. Preprocessing the EEG signals is necessary due to the existence of noise created by muscle artifact, skin artifact, electrode movement, eye movement, respiration artifact, etc [5].

EEG signals are non-stationary, causing the conventional method of frequency analysis to be less successful in diagnostic classification. Therefore, more complex models and feature engineering techniques are needed to improve the precision and accuracy of the classification [11]. These complex models are commonly known as black box models because they are difficult to interpret. Cutting edge methods involving black box models have the ability to significantly improve outcomes, however the trade-off between accuracy and interpretability is a significant challenge in the field of machine learning. Hastie et al. [12] has shown multiples ways, including Friedman's partial dependence plot [13] and Pearl's back-door adjustment [14], to determine causal interpretation of black box models. A black box model receives the input variables and produces the response variables. We can name two important elements of a black box: (1) the information that can be algorithmically extracted, and (2) the noise [12]. Some common types of models, such as generalized linear models, subscribe to a data modeling culture which assumes the input variables are in a parametric form. Black box models are non-parametric and work to maximize predictive accuracy by approximating the input variables using a high-dimensional and highly nonlinear function with many interactions. These black box models often perform significantly better than the parametric models (in terms of prediction) and have achieved tremendous success in applications across many fields [15].

In this paper, we use a case study on eye state detection using EEG data to investigate how the proposed framework makes predictions using a DNN model. The main objective here to demonstrate how the proposed framework is able to explain the outcomes while using a black-box model. Therefore, the results were compared to a gradient boosted tree model to evaluate the accuracy and interpretabilty of each of the models. This paper is structured as follows: The introduction and background is presented in section I. Section II includes the details of the modeling and interpretation framework followed by section III with the details of the case study including the experimental data and developed models. Results and discussion are presented in section IV and the summary and conclusions of the study are recapitulated in section V.

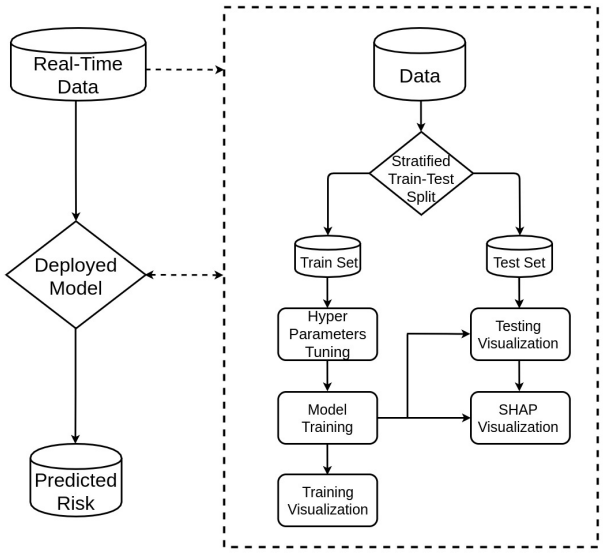


Fig. 2. Modeling and interpretation framework flowchart.

II. Modeling & Interpretation Framework

Modeling and interpretation framework includes four main components: (1) extract, transform, load (ETL) process, (2) training , (3) deployment, (4) prediction. Fig. 2 illustrates the flowchart of the framework. As shown, the overall pipeline flow begins with loading the real-time data and ETL process, applying the deployed trained model, and calculate the prediction, while each step includes various details and specific challenges. For instance, the ETL process itself includes extracting the data from in-house database or cloud storage, applying the required transformation to have the data in tidy format, and loading it back to the server for the next steps of the pipeline. The box with the dashed line in Fig. 2 illustrates the training process that can applied to any model to be deployed via the proposed framework. As the training step goes after feature extraction/engineering (let's say for a classification problem), the data is splitted into train/test sets in a stratified fashion. This would a be crucial task to improve the generalization results due to any imbalanced problem. In the training step, the feature standardization/scaling can be fitted to the train data and scaler object should be applied to the test set to transform the test data into the scaled train data subspace (no fitting for test data). Next, the hyper-parameters of the model should be tuned. Fail to tune the hyper-parameter values is one of the most common reasons for training a biased/over-fitted model. There are various methods including exhaustive grid-search, random search, and Bayesian optimization [16]. Grid search provides an exhaustive search over specified parameter values. All the possible combinations of the specified hyper-parameters will be checked. In contrast to grid search, in random search not all parameter values would be tried out, but rather a fixed number of parameter settings will be sampled from the specified distributions. This is quite interesting since not all

the alternative values in each array for hyper-parameters play an important role in the outcomes. Therefore, by random sampling, the most important hyper-parameters can be determined and the other hyper-parameters settings can be kept fixed. In this way, the same results would not be replicated anymore and the learning slope will be positive. Employing machine learning to predict what combinations are likely to work well could help to rescue from the huge computational time. It requires to predict the regions of the hyper-parameter space that might give better outcomes. It also requires to predict how well a new combination will do and model the uncertainty of that prediction using Gaussian Process models. Gaussian processes provide a principled, practical, and probabilistic approach in machine learning. Gaussian processes simply have an essential assumption that similar inputs give similar outputs. This simple and weak prior are actually very sensible for the effects of hyper-parameters. Bayesian optimization, is a constrained global optimization approach built upon Bayesian inference and Gaussian process models to find the maximum value of an unknown function in the most efficient ways (less iterations) [16]. After tuning the hyper-parameters, the model can be trained and the fitness metrics can be evaluated using both training and testing data sets. Finally, the visualization modules of the training stage including the evolution of the performance metrics on both training and testing data sets, visualization of the trained model, and the feature importance can be employed. The testing stage begins with running the testing visualization modules including receiver operating characteristic (ROC) curves and confusion matrix. Finally, Shapley values can be calculated for the testing data and SHAP visualization modules can be applied on the testing data. As show, the training process has a feedback loop to the deployment method. The most common approach for real-time risk prediction in healthcare is batch-processing. In principle, the real-time data goes through the deployed model and after the feature extraction/engineering the risk can be predicted. However, everyday the model encounters new patients and the ability to predict the new encounters is vital. Therefore, the new data after the prediction step can be joined back to the training data via a scheduled job and re-training process of the model and deployment of the new version of the model can be done. The interval of this step totally depends on the use-case.

III. Case Study

A. EEG Data

The goal of EEG is to non-invasively record the voltage differences in scalp potentials that result from the electrical activity of neurons. In principle, these potential differences are caused by summed post-synaptic potentials from pyramidal cells that create diploes between soma and apical dendrites. However, the recorded potentials can also be the result of noise signals due to different sources of artifacts and movements. The EEG electrodes on the scalp amplify these microscopic signals, which are usually sampled at 256 Hz or higher, to provide a high temporal resolution [5].

TABLE I
Layer settings for DNN Model.

Layer Type	Number of Neurons	Activation Function	# of Params
Input	14	ReLU	-
Dense	10	ReLU	150
Dense	9	ReLU	99
Dense	8	ReLU	80
Dense	7	ReLU	63
Dense	4	ReLU	32
Dense	2	ReLU	10
Output	1	Sigmoid	3

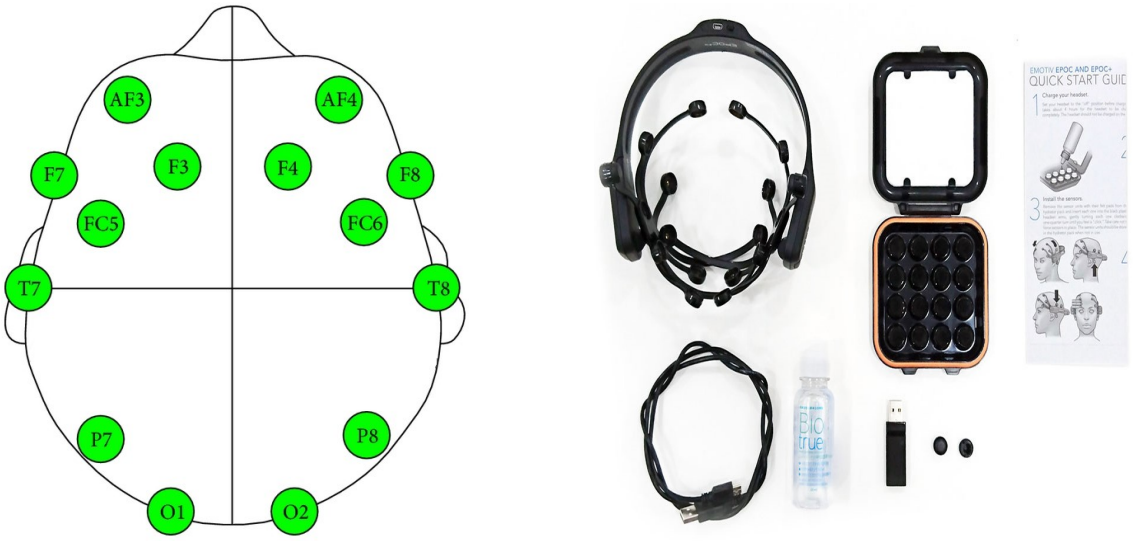
Algorithm 1: Adam

Input: Training data S, learning rate η, weights w, fuzz factor ϵ, learning rates decay over each update r_1 and r_2, exponential decay rates β_1=0.9 and β_2=0.999

Output: Updated weights w

$\epsilon \leftarrow \epsilon_0 \approx 10^{-8}$;
$w \leftarrow w_0$;
$r_1 \leftarrow 0$;
$r_2 \leftarrow 0$;
$t \leftarrow 0$;
while *stopping criterion is not met* **do**
  Randomly shuffle the training data S ;
  Sample a minibatch of size m:$\{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\} \in S$;
  for $i \in \{1, \ldots, m\}$ **do**
    $\hat{G} \leftarrow \frac{\partial}{\partial w_i} cost(w, (x^{(i)}, y^{(i)}))$; Gradient calculation
    $t \leftarrow t + 1$;
  end
  $r_1 \leftarrow \beta_1 r_1 + (1 - \beta_1)\hat{G}$;
  $r_2 \leftarrow \beta_2 r_2 + (1 - \beta_2)\hat{G} \odot \hat{G}$;
  $\hat{r_1} \leftarrow \frac{r_1}{1-\beta_1^t}$;
  $\hat{r_2} \leftarrow \frac{r_2}{1-\beta_2^t}$;
  $w \leftarrow w - \eta \frac{\hat{r_1}}{\epsilon + \sqrt{\hat{r_2}}}$;
end

All of the experiments were conducted in a quiet room. The participant was told to sit in a relaxed position and change their eye state at will. Data was recorded as one continuous EEG measurement obtained from the 14 electrodes placed in the regions of interest (ROIs), as shown in Fig. 3a with the Emotiv EEG Neuro-headset (shown in Fig. 3b). These ROIs were located in the frontal (F), temporal (T), central (C), occipital (O), and parietal (P) lobes. Odd numbers represent the left hemisphere and even numbers represent the right hemisphere. The duration of the measurement was 117 seconds and the data was saved in chronological order to be able to analyze temporal dependencies using forward-chain cross-validation. The eye state was recorded via camera during the EEG



(a) ROIs

(b) Emotiv Neuro-headset

Fig. 3. EEG measurement elements: (a) ROIs for EEG measurement, (b) Emotiv EEG Neuro-headset.

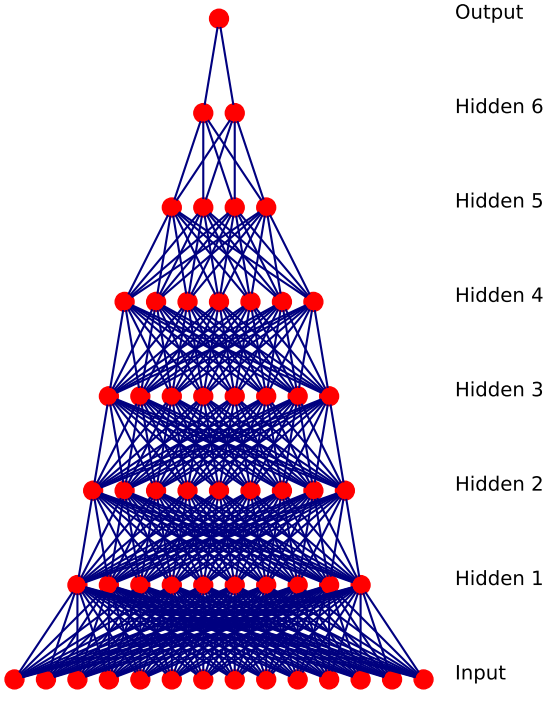


Fig. 4. The architecture of the DNN model.

measurement and manually added to the file after analyzing the video frames [17], [18]. Neurons' activities generate different wave patterns, including (1) δ (< 4 Hz), (2) θ ($4-7$ Hz), (3) α ($7-12$ Hz), (4) β ($12-30$ Hz), and (5) γ ($30-100$ Hz), which were the main inputs in this research [19]. The data includes the measurements of the 14 ROIs, shown in Fig. 3 [17], [18], for 14977 instances. Of the instances, 55% correspond to open-eye state and 45% correspond to closed-eye state.

B. Models

Since EEG signals are non-stationary, the conventional method of frequency analysis is not highly successful in diagnostic classification [8]. Neurons in the perceptual system represent features of the sensory input. The brain has a deep architecture and learns to extract many layers of features. Features in one layer represent combinations of simpler features in the layer below and so on. This is referred to as feature hierarchy [16]. Based on this idea, we have developed a DNN model (shown in Fig. 4) with seven fully-connected layers. The architecture of the model was chosen after using a rapid Bayesian optimization, along with the rules of thumb presented in Table I. To benchmark the results from the DNN model, a gradient boosting model, implemented using the XGBoost API in Python [20], was also trained on the same training set.

The main purpose of this paper is to show how the results from a DNN model can be as interpretable as a gradient boosting model while exceeding the accuracy. The acceptance of the interpretability of complex models would open new avenues to employ more sophisticated models for various applications, including diagnosis and early detection of illnesses. In principle, deep learning models involve optimization to find the best weights at each layer of the DNN in order to decrease the cost on the entire training set [21]. Optimization is an arduous and time-consuming problem even without the complexity that non-convex models, such as DNNs. In contrast to the traditional optimization methods in which the optimization of the pure objective function is the direct goal, finding some weights that minimize the cost function and reduce the expected generalization error respectively in DNN is indirect.

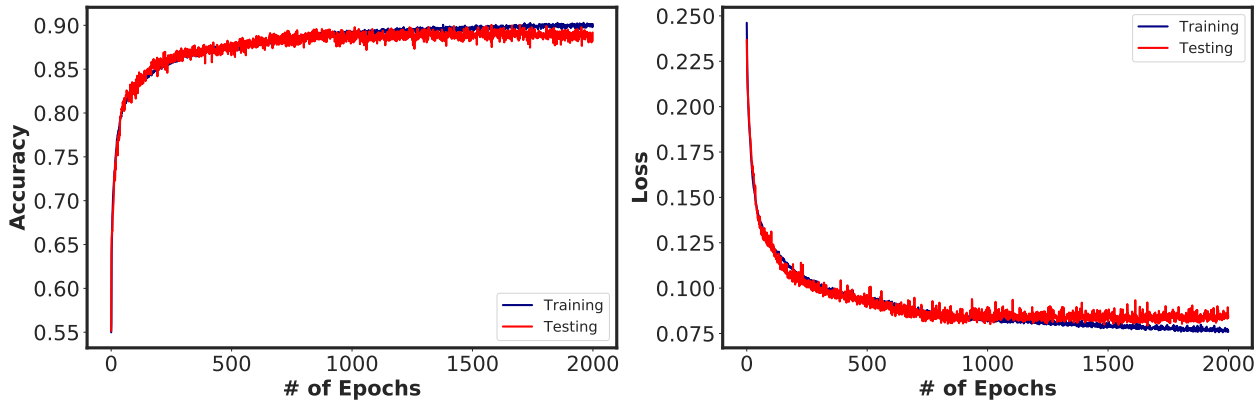
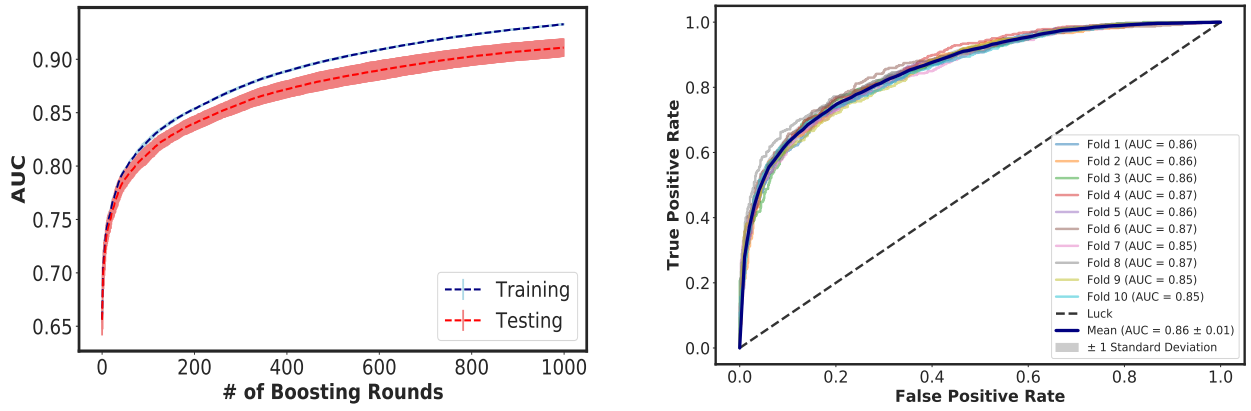


Fig. 5. The evolution of the loss (binary cross entropy) and accuracy for the training/testing sets through different epochs for the trained deep model.



(a) Evolution

(b) ROC

Fig. 6. The trained XGBoost model performance: (a) evolution of the AUC through the boosting rounds with 10-folds CV in training, and (b) the ROC curves with 10-folds CV.

Additionally, in machine learning tasks the true distribution of the training data set is unknown. To overcome this challenge in our study, before optimization algorithms were applied the true distribution of the training data was substituted with the empirical distribution defined by the training data. There are various optimization strategies, including grid search and random search, as well as advanced strategies such as Bayesian optimization, which can be employed along with optimization algorithms with adaptive learning rates for the training phase of the DNN model [16].

In this paper, the DNN model was trained for 2000 epochs with a batch size of 100. The Adam algorithm, as shown in Algorithm 1 (with default built-in hyper-parameters in Keras API and TensorFlow backend [22]), was chosen as the optimizer with binary cross entropy as the loss function.

IV. Results & Discussion

Both models were trained on the same training set and were tested on the same testing set. Fig. 5 illustrates the loss/accuracy evolution of the DNN model through 2000 epochs for training/testing sets. Similarly, Fig. 6a presents the evolution of the area under ROC curve (AUC) through 1000 boosting rounds for training/testing sets, while Fig. 6b presents the forward-chain 10-fold cross-validation ROC curves with confidence intervals of ± 1 standard deviation of the AUC in 10 folds. In addition to this, Fig. 7 illustrates the feature importance of the trained model based on the total gain metric. The gain implies the relative contribution of the corresponding feature to the trained model calculated by taking each feature's contribution for each tree in the model. A higher value of this metric when compared to another feature implies that the corresponding feature has more impact for generating a prediction. In principle, total gain is the total improvement in evaluation metric (AUC here) brought by a feature with respect to all features to the branches it is on. In fact, before adding a new split on a feature X to the branch, there were some wrongly classified elements, after adding the split on this feature, there are two new branches, and each of these branches would be more accurate (one branch saying if your observation is on this branch, then it should be classified as 1, and the other branch saying the exact opposite and it should be classified as 0).

The basic idea of interpretability comes from the simplicity of the model: the simpler model, the more explainable. For

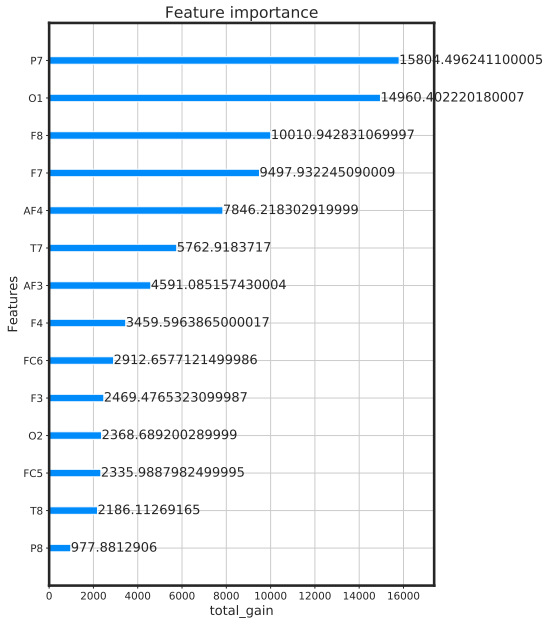
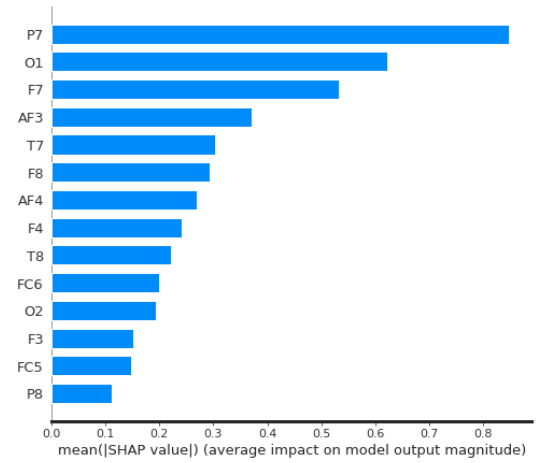


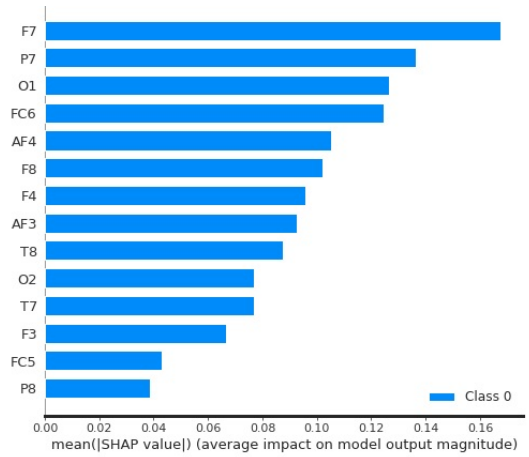
Fig. 7. The feature importance of the trained XGBoost model using total gain.

complex models such as DNNs and ensemble methods including gradient boosting, simplicity is not attainable. Therefore, an approach is needed to replace the complex model with an interpretable approximation of the original model. There are several approaches to improve the explainability of a model: (1) LIME [23], (2) DeepLIFT [24], (3) Layer-Wise Relevance Propagation [25], (4) Shapley Regression Values [26], and (5) Shapley Sampling Values [27]. In this paper, we have used SHAP values to explain the importance of the features based on the trained DNN model. SHAP (SHapley Additive exPlanations) is a unified approach created to explain the output of any machine learning model through connecting game theory with local explanations. SHAP unifies several of the previous methods and presents the only possible consistent and locally accurate additive feature attribution method based on expectations [28], [29]. For example, Deep explainer of SHAP is inspired by DeepLIFT. [24]. DeepLIFT implies that what we care about is not the gradient, which describes how y changes as x changes at the point x, but the slope, which describes how y changes as x differs from the baseline.

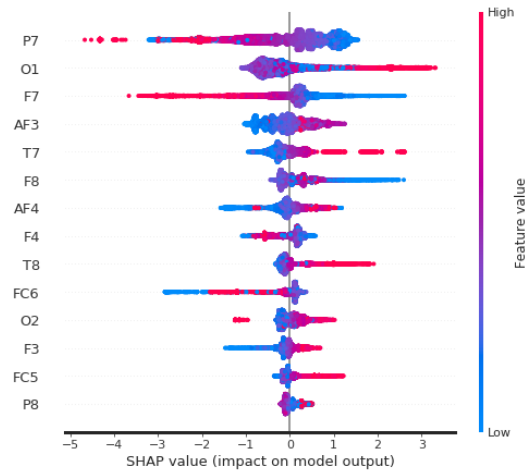
Fig. 8 illustrates the SHAP summary plots and feature importance of the trained XGBoost model using tree explainer and DNN models using deep explainer. The summary plot combines the feature importance with feature effect, as seen in Fig. 8c. For each of the features, the SHAP values and their impacting contribution to the model (high as red, low as blue) are shown. While SHAP values can have both positive and negative values, for the sake of comparison, the average of absolute Shapley values are used in Fig. 8a and Fig. 8b to compare the global average impact on model output magnitude between XGBoost and DNN models ($I_j = \sum_{i=1}^{n} |\phi_j^{(i)}|$). The idea behind SHAP feature importance is simple: features with large absolute Shapley values are important. As seen, the top three important features (P7, O1, and F7) are the same



(a) XGBoost



(b) DNN



(c) XGBoost

Fig. 8. (a) and (b) illustrate the SHAP feature importance for the trained XGBoost, and the DNN models while (c) presents the SHAP summary plot for the trained XGBoost model.

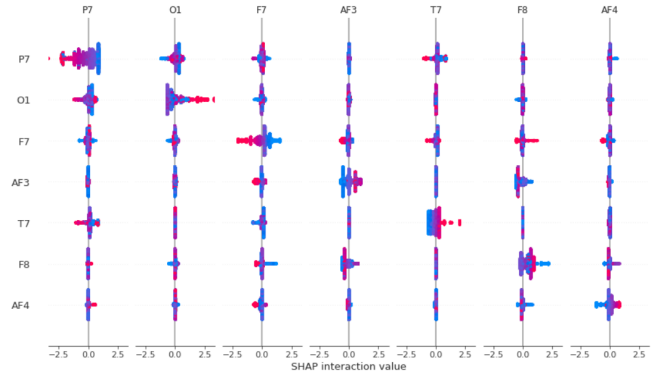


Fig. 9. Summary plot for SHAP interaction values based on the trained model using XGBoost.

for both models with slightly different impacts. However, a point of disparity in the models is the contribution of FC6, which is high in the DNN model but much lower in the XGBoost model. It is always recommended to compare the feature importance of the features during training with their SHAP values. As seen in Fig. 7, F8 gained slightly more importance over F7 region, while F8 (SHAP $\approx$ 0.3) had about 45% less average impact on model output magnitude in comparison to F7 (SHAP ≈ 0.55). Similarly, this pattern is repeated for the DNN model as F7 has the highest impact on the model output magnitude based on the absolute value of SHAP (SHAP ≈ 0.16), while F8 (SHAP ≈ 0.10) has roughly 40% less impact on the model output magnitude. The other interesting pattern that can be seen is the higher contribution of the low-impact regions based on the XGBoost model in the DNN model. This is due to the non-linearity of the neural network layers which lead to more complex components in the DNN, while the XGBoost model with a tree depth of two can just lead to second-order interaction between features [30]. This interaction is shown in Fig. 9 and is similar to a pair-wise correlation matrix in that the interactions in the training phase would be considered between each pair's features. It should be noted that, the tree depth can be increased, however as the tree depth gets larger, so does the risk of over-fitting the model.

V. Summary & Conclusions

In this paper, we have discussed why it is beneficial to have a real-time framework for health monitoring systems using models such as DNNs, as well as some of the explainability complications involved in such models. Through our case study we have illustrated that by using SHAP values, these models can be interpretable, at least in the area of feature importance. It should be noted the input signals can always be noisy, especially in more complex datasets and increase of the number participants, which requires more advanced preprocessing techniques to produce explainable results.

Although the results were promising, it was not clear whether the results were statistically significant since there was only one participant. As the availability of data increases, so does the opportunity for machine learning algorithms to discover solutions to real-world problems. However, much of this data is unstructured and complex, just as the EEG data was in this case study. These challenges have pushed many to forego traditional methods and innovate new, cutting-edge approaches, many of which fall into the category of black-box models. Many consumers of machine learning models will not trust the results if they cannot understand the method. While the mechanism and math of a black box model is still a difficult concept to grasp, we hope that supplementing predictions with understandable feature importance results will go a long way in fostering trust in these methods. If this trust cannot be gained, the benefit of cutting-edge methods in machine learning is largely lost. Developing a framework with even more complex algorithms such as convolutional neural networks with the main objective of building trust in consumers could open new avenues for future studies.

Acknowledgment

The authors would like to thank Persia Behbahani for the careful revision of the final version of the manuscript.

Conflict of Interest

No conflict of interest has been declared by the authors.

Data Accessibility

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] T. Keller, T. Zeller, D. Peetz, S. Tzikas, A. Roth, E. Czyz, C. Bickel, S. Baldus, A. Warnholtz, M. Fröhlich *et al.*, "Sensitive troponin i assay in early diagnosis of acute myocardial infarction," *New England Journal of Medicine*, vol. 361, no. 9, pp. 868–877, 2009.

[2] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier, "The internet of things for ambient assisted living," in *2010 seventh international conference on information technology: new generations*. Ieee, 2010, pp. 804–809.

[3] A. Anzanpour, I. Azimi, M. Götzinger, A. M. Rahmani, N. TaheriNejad, P. Liljeberg, A. Jantsch, and N. Dutt, "Self-awareness in remote health monitoring systems using wearable electronics," in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*. IEEE, 2017, pp. 1056–1061.

[4] R. Caton, "Electrical currents of the brain," *The Journal of Nervous and Mental Disease*, vol. 2, no. 4, p. 610, 1875.

[5] T. Thompson, T. Steffert, T. Ros, J. Leach, and J. Gruzelier, "Eeg applications for sport and performance," *Methods*, vol. 45, no. 4, pp. 279–288, 2008.

[6] A. Z. Al-Marridi, A. Mohamed, A. Erbad, A. Al-Ali, and M. Guizani, "Efficient eeg mobile edge computing and optimal resource allocation for smart health applications," in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*. IEEE, 2019, pp. 1261–1266.

[7] W. Karwowski, M. Rahman, P. Hancock, and M. Fafrowicz, "Applications of electroencephalography in physical neuroergonomics: A systematic literature review," *Frontiers in Human Neuroscience*, vol. 13, p. 182, 2019.

[8] A. Subasi and E. Ercelebi, "Classification of eeg signals using neural network and logistic regression," *Computer methods and programs in biomedicine*, vol. 78, no. 2, pp. 87–99, 2005.

[9] G. Sahonero-Alvarez and H. Calderón, "A comparison of sobi, fastica, jade and infomax algorithms," in *Proceedings of the 8th International Multi-Conference on Complexity, Informatics and Cyberneti, Orlando, FL, USA*, 2017, pp. 21–24.

[10] T. Ning and J. Bronzino, "Autoregressive and bispectral analysis techniques: Eeg applications," *IEEE Engineering in Medicine and Biology Magazine*, vol. 9, no. 1, pp. 47–50, 1990.

[11] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[12] Q. Zhao and T. Hastie, "Causal interpretations of black-box models," *Journal of Business & Economic Statistics*, no. just-accepted, pp. 1–19, 2019.

[13] J. H. Friedman and J. J. Meulman, "Multiple additive regression trees with application in epidemiology," *Statistics in medicine*, vol. 22, no. 9, pp. 1365–1381, 2003.

[14] J. Pearl, "Interpretation and identification of causal mediation." *Psychological methods*, vol. 19, no. 4, p. 459, 2014.

[15] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005.

[16] A. Tahmassebi, "ideeple: Deep learning in a flash," in *Disruptive Technologies in Information Sciences*, vol. 10652. International Society for Optics and Photonics, 2018, p. 106520S.

[17] O. Rösler and D. Suendermann, "A first step towards eye state prediction using eeg," *Proc. of the AIHLS*, 2013.

[18] O. Roesler, L. Bader, J. Forster, Y. Hayashi, S. Heßler, and D. Suendermann-Oeft, "Comparison of eeg devices for eye state classification," *Proc. of the AIHLS*, 2014.

[19] A. Tahmassebi, A. H. Gandomi, and A. Meyer-Baese, "An evolutionary online framework for mooc performance using eeg data," in *2018 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2018, pp. 1–8.

[20] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[22] F. Chollet *et al.*, "Keras," 2015.

[23] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?: Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 2016, pp. 1135–1144.

[24] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3145–3153.

[25] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, p. e0130140, 2015.

[26] S. Lipovetsky and M. Conklin, "Analysis of regression in game theory approach," *Applied Stochastic Models in Business and Industry*, vol. 17, no. 4, pp. 319–330, 2001.

[27] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, no. 3, pp. 647–665, 2014.

[28] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, 2017, pp. 4765–4774.

[29] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent individualized feature attribution for tree ensembles," *arXiv preprint arXiv:1802.03888*, 2018.

[30] A. Tahmassebi, A. H. Gandomi, I. McCann, M. H. Schulte, A. E. Goudriaan, and A. Meyer-Baese, "Deep learning in medical imaging: fmri big data analysis via convolutional neural networks," in *Proceedings of the Practice and Experience on Advanced Research Computing*, 2018, pp. 1–4.