2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

# LSTM based GAN Networks for Enhancing Ternary Task Classification Using fNIRS Data

Sajila D.Wickramaratne[1] and MD Shaad Mahmud[2]

***Abstract*— Brain activation patterns vary according to the tasks performed by the subject. Neuroimaging techniques can be used to map the functioning of the cortex to capture brain activation patterns. Functional near-infrared spectroscopy (fNIRS) is a neuroimaging technique increasingly used for task classification based on brain activation patterns. fNIRS can be widely used in population studies due to the technology's economic,non-invasive, and portable nature. The multidimensional and complex nature of fNIRS data makes it ideal for deep learning algorithms for classification. Most deep learning algorithms need a large amount of data to be appropriately trained. Generative networks can be used in such cases where a substantial amount of data is required. Still, the collection is complex due to various constraints. Conditional Generative Adversarial Networks (CGAN) can generate artificial samples of a specific category to improve the deep learning classifier's accuracy when the sample size is insufficient. The proposed system uses an LSTM based CGAN with an LSTM classifier to enhance the accuracy through data augmentation. The system can determine whether the subject's task is a Left Finger Tap, Right Finger Tap, or Foot Tap based on the fNIRS data patterns. The authors obtained a task classification accuracy of 90.2% for the LSTM based GAN combination.**

***Clinical relevance*— Acquiring medical data present practical difficulties due to time, money, labor, and economic cost. The deep learning-based model can better perform medical image classification than hand-crafted features when dealing with many data. GAN-based networks can be valuable in the medical field where collecting extensive data is not feasible. GAN-generated synthetic data can be used to improve the classification accuracy of classification systems.**

## I. INTRODUCTION

Functional near-infrared spectroscopy (fNIRS) is a neuroimaging technology for mapping the functioning human cortex, which uses near-infrared spectroscopy [1]. This mapping is done by measurements and images of local brain changes caused by the modulation of cerebral blood flow and oxygen metabolism by neural activity[2]. fNIRS is a non-invasive, repeatable, portable, high temporal resolution, and economical technology with widespread use. fNIRS is more suited for the populations and studies for which other imaging modalities are limited, including infants and children, procedures involving mobility and interactivity, and clinical environments[2].

Neuroimaging techniques used such as EEG and fNIRS have emerged to be the most widely used modalities for task classification[3]. EEG and fNIRS are suitable for studies with population since it is inexpensive and not harmful to repeated use. Most of the existing task classification systems use conventional machine learning methods for classification. The traditional machine learning methods are used frequently due to their simplicity of implementation. On the drawbacks, these traditional methods require a significant amount of data preprocessing and feature extraction. Additionally, traditional machine learning methods do not fully use complex patterns of neural signals. The accuracy of the conventional classifiers depends predominantly on the features selected for training the model. The extraction and selection of optimal features can be a challenge with neural signals. The complexity and multi-dimensionality of fNIRS data make it much more suitable for deep learning methods.

There are currently some successful deep learning classifiers with neuroimaging modalities such as EEG and fNIRS. One of the main issues with implementing deep learning-based classifiers is the insufficient sample size in instances where collecting data is difficult. Acquiring medical data present practical difficulties due to time, money, labor, and economic cost, sometimes resulting in smaller sample sizes. The models tend to overfit with small sample sizes, create difficulty generalizing the model, and underperform testing data. Data augmentation is a method that enables researchers to increase the diversity of training data available for models without additional data collection. In the health field, obtaining high-quality labeled data for deep learning algorithms can be costly and time-consuming; generative networks can help. One way which can be used for data augmentation with a deep learning algorithm is known as General Adversarial Networks(GAN). Researchers have found that data augmentation with GAN networks has improved the classification accuracy[4]. GAN networks have been used successfully to generate EEG signals using GAN networks for data augmentation[5], [6] . However, the most challenging task of a GAN-based signal is signal verification. Although with the visual output, it may be easier to verify visually.

The authors propose a classification system based on a hybrid CGAN-LSTM network to classify images derived from fNIRS signals. The proposed deep learning system will not be affected by the relatively smaller number of samples due to the ability of the CGANs to augment the data. This proposed system can be used when training a deep learning classifier with a relatively smaller number of samples. The proposed system can both generate artificial samples and classify actual data as well. The proposed system obtained a classification accuracy of 90.2% and AUROC of 0.95.

[1] Sajila D.Wickramaratne is with the Department of Electrical Computer Engineering, University of New Hampshire, Durham NH 03824, United States sdw1014 at wildcats.unh.edu

[2]MD Shaad Mahmud is with the Department of Electrical Computer Engineering, University of New Hampshire, Durham NH 03824, United States mdshaad.mahmud at wildcats.unh.edu

This proposed system exceeded the accuracy obtained for the same dataset using the traditional machine learning classifiers. It was also able to match the performance of the deep learning classifiers that were used to classify the data earlier.

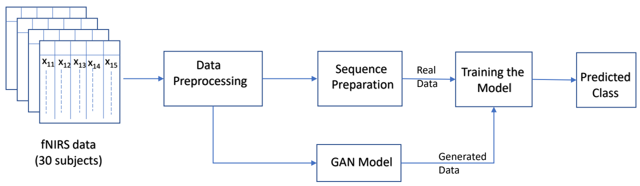


Fig. 1. System Overview

## A. System Overview

Several architectures were initially considered to determine the final configuration of the system design. The final model was chosen based on the best performance metrics. Fig.1 illustrates the Overview of the complete system. The acquired raw fNIRS data were initially preprocessed to remove the disturbances. Neural signals have highly correlated variables that should be removed before being fed into the model. Therefore dimension reduction techniques were performed on preprocessed data to remove the highly correlated variables. Afterward, the data is sent to the sequence preparation phase. The sequences are made according to the model input parameters. The data set is then divided into the test and training sets. The training set is sent to the CGAN model, where artificial sequences are generated for the three categories. Both real data and CGAN generated data are used to train the deep learning classifier based on LSTM. The test set is fed directly to the classifier and used to determine the performance. Finally, the result is obtained with the task being classified into Right Hand Tap(RHT), Left Hand Tap(LHT), or Foot Tap(FT).

## B. Data

The data used for the training of the classifier was obtained from an open database[3]. A more detailed description of the data can be found in the original publication. Thirty volunteers participated in this study. The fNIRS data were recorded by a multichannel fNIRS system consisting of eight light sources and detectors.

A single trial included an introduction period and a task period, followed by an inter-trial break. The inter-trial interval was 30 s on average. Out of RHT, LHT, and FT, a specific task type was displayed randomly, which volunteers were required to perform. For RHT/LHT tasks, the volunteers performed unilateral complex finger-tapping at a rate of 2 Hz. For FT, the participants tapped their foot at a 1 Hz rate.

## C. Pre-processing Data

The concentration changes of the oxygenated and reduced hemoglobin (ΔHbO/R) were band-pass filtered. This was done using a zero-order filter implemented by the third-order Butterworth filter with a 0.01-0.1 Hz passband. Filtering is of the signal is essential to remove the physiological noises and DC offset. The ΔHbO/R values were segmented into epochs ranging from -2 to 28 s relative to the task onset. Baseline correction was done for each epoch by subtracting the average value within the reference interval(-1 to 0s).

After the data was preprocessed, the data should be prepared in sequences before being fed to the model. All the sequences have the same length and are scaled to [-1,1] range. Then the data is sent through a kernel principal component unit to extract the most significant component. Hence the number of data sequences from each subject was reduced to 1. This reduction was made to make the GAN models less complicated. Each sequence had a length of 350. These sequences were used to train the GAN network and then for the training of the LSTM based classifier. Although the GAN-generated data was used to train the network, the testing was done only on original data.

## D. Model

GANs are an innovative way of training a generative model by framing the problem as a supervised learning problem using deep learning models. GANs can automatically discover the pattern in input data. The GAN architecture was first proposed in the 2014 paper by Ian Goodfellow[7]. GANs can generate new samples that appear to belong to the original dataset[8].

There are two sub-models in a GAN called the Generator(G) and Discriminator(D). The generator model is used to train to generate new examples and the discriminator model to classify samples real or fake generated[7]. GANs are the two models behind the training motivation trying to achieve the Nash equilibrium of Game Theory. A non-cooperative game solution must be reached between two adversaries to achieve Nash equilibrium. Each player already knows all the other player's strategies. Therefore, no player gains anything by modifying their strategy[7]. Any function that can be differentiated can be used as the function for equations of Generator and Discriminator.

The technique of Conditional GAN(CGAN) has similarities to the GAN network. Both the Generator and Discriminator have conditioned an extra input(y) which can be auxiliary information. The conditioning can be performed by feeding into both the Discriminator and Generator as the additional input layer. In the proposed model, class labels are considered as the "y" parameter[9]. The cost function for CGAN is shown in Eq.1. By adding additional information in a form of a condition, both the generator G and the discriminator D learn to operate in specific modes.

$$\begin{aligned} \min_G \max_D V(G,D) = \min_G \max_D E_{x\sim p_{data}}[log(D(x|y))] \\ +E_{z\sim p_z}[log(1-D(G(z|y)))] \end{aligned} \tag{1}$$

The generator model takes a fixed-length random vector as input and generates a sample in the domain. From a Gaussian distribution, a random vector is drawn to initiate the generative process. After training, points in this multidimensional vector space will correspond to points in the problem domain to form a compressed representation of the data distribution. This vector space is referred to as a latent space. Latent variables are not directly observable.

The proposed model is based on LSTM units. Most of the existing GAN models are based on the CNN architecture. The purely CNN-based architectures are most often used for image generation. The LSTM based modules are much more suitable for handling time series data. The LSTM units can retain the data that can be useful in memorizing the sequence's temporal dynamics. Time series generation using GANs is a challenging task. There are temporal and global dynamics that the generated sequences should preserve. Throughout literature, there are not a lot of sequence-based GANs which were able to converge successfully. Neural signals-based GANs are even harder to design due to their complicated design. In this design, only a single channel of uncorrelated data is used for training the GAN. This approach can be scaled for a multichannel system. Apart from the LSTM model, the authors attempted to train a 1D CNN model, which was unsuccessful in converging for the fNIRS data. In literature, however, there are several 1D models which were successfully trained for sequence generation, although the verification methods were not well explored.

The Input for the GAN system was the preprocessed fNIRS data. For simplicity, only a single channel was chosen. Yet, there are ways to expand the process to a multi-channel approach. However, such an approach can increase the time taken for converging or may not converge at all. Both generated and actual data were considered for training. However, for testing, only the actual data was considered. From the final classifier, the task was determined. The final classifier was also an LSTM based model, and the authors attempted to increase the accuracy of the classifier only through data augmentation.

## II. Results

The results for the classifier can be analyzed in several different stages. In the first step, the results obtained by the traditional classifiers are analyzed. In the second stage, the results are analyzed considering the performance of the LSTM based classifier. In the final phase, with the GAN network's data, the classifier's performance is observed when the training data is augmented. The main performance parameters that were considered are classification accuracy and the Area Under the ROC curve. Further confusion matrices were also analyzed to find out which classes were easier to detect.

## III. Performance evaluation and discussions

In the first step of the study, the traditional classifiers were used to obtain classification accuracy. The best performing traditional classifier was the SVM classifier, based on the data set's original research. The traditional classifiers require the feature selection process. The accuracy of the model greatly depends on the features that are selected. Two deep learning models were used, one for classification and the other for data augmentation. In the second phase, further analysis was done to determine the data generated through the data augmentation process. The deep learning classifier proposed in this study is LSTM-based, which is trained using the CGAN network's data. The LSTM classifier, which was trained using only real data, obtained an accuracy of 78.9%. The maximum accuracy obtained is 90.2%, which was obtained using the original data augmented with 110% of generated data. The generated data is given as a percentage of the original data.

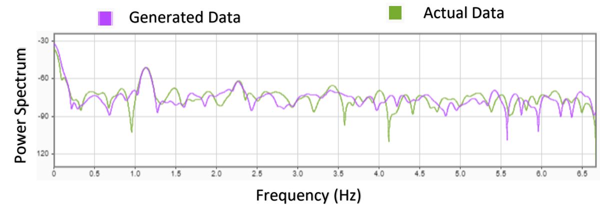


Fig. 2. The comparison between frequency power spectrum generated for GAN generated data for the first subject

The comparison between frequency power spectrum generated for GAN generated data for the first subject is shown in Fig.2. The generated data samples start at 10% of the original data and increases by 10% each step. The traditional performance metrics, such as accuracy and AUROC, and the classification accuracy are calculated for each step. Table II shows how the classification accuracy and average AUROC vary with each step.

As expected, the classification accuracy improved with more data. The general architecture of the classifier was not changed. However, several regularization parameters were changed along with the increase of data. Initially, there was a strict regularization scheme due to the small size of data. The regularization was periodically relaxed to prevent under-fitting. The classifier obtained a maximum classification accuracy of 95.6%, which was trained with actual data and 150% of generated data. Further improvement of the classification accuracy required the data changes to the data architecture. Since the authors intended to improve the model's performance through data augmentation, only expanding the training data was not considered. Another vital fact to be considered is preserving data integrity if more generated data will be used than actual data.

A common result during the training phase of a GAN is mode collapse. In mode collapse, the Generator learns a single solution that deceives the discriminator. As the discriminator processes each point independently, it cannot know how different each Generator solution is from the other. Thus, all outputs generated become the single point that the discriminator believes is realistic. The authors initially encountered this problem and solved it by adding a minibatch discrimination layer for the discriminator. Minibatch discrimination examines solutions in combination and penalizes the discriminator if the solutions are similar. Evaluation of

TABLE I
COMPARISON BETWEEN THE PERFORMANCE OF TRADITIONAL CLASSIFIERS FOR THE ORIGINAL DATA SET

| Model | Classification Accuracy% | Average AUROC |
|---|---|---|
| Logistic Regression | 58.6 | 0.51 |
| Random Forest | 64.8 | 0.59 |
| SVM | 70.6 | 0.72 |
| XGBoost | 65.4 | 0.60 |

the distribution of the samples generated by the Generator is a problem. However, researchers are coming up with new approaches. Visual inspection of generated samples can help identify apparent failures and mode collapse. Visual Inspection cannot give any quantitative information about the variance of generated samples and how similar they are to the training data. A common approach to providing information about the quality of the trained Generator is to use the inception score (IS) mainly for image data.

One difficulty that the authors came across was the verification of the generated signals. When GANs are used for image generation, even a visual comparison can indicate how good the generated samples are compared with the real samples. There are no definite visible markers to compare with the real samples through visual inspection of fNIRS data. One area of improvement that future researchers can consider is verification. Further, since this study intends to diversify the samples, a proper verification method that can verify the samples do belong to different individuals can help future studies. If this verification process can be successful, the generated data can be used for other medical-related purposes such as simulation besides classification.

More prominence should be given to a proper verification process in future studies since it seems the traditional verification metrics do not make any sense in this context. With an appropriate verification method, these models can expand roles in neural simulations and assist in both medical training and classification algorithms. Further, the ideal amount of data required to train the model properly will vary according to the situation. There may be a generalized way to determine the amount of data that has to be generated.

## IV. CONCLUSION

The authors propose a ternary classification system combined with a GAN system. The GAN system can be appropriately trained to generate synthetic samples of the training data. GAN facilitates data augmentation, which can increase the accuracy of the classification model. For many medical field-related classifiers, the lack of adequately labeled data is a significant concern. GAN networks can generate data that can be used for network training to increase performance metrics. GANs are currently focused on image generation. The sequence generation models still suffer from the difficulty of convergence. Further verification methods regarding generated sequences can also be improved. In order to generalize the model a proper verification method is essential, and should be considered as a future research direction. For medicine and health science, where collecting data from many people is practically difficult due to economic or time constraints, an adequately trained GAN network can generate artificial data to improve classification accuracy.

TABLE II
TRAINING DATA COMPOSITION AND PERFORMANCE METRICS FOR THE CLASSIFIER

| Data | Classification Acc.% | Avg. AUROC |
|---|---|---|
| Real Data+0% Gen. Data | 78.2% | 0.69 |
| Real Data+10% Gen. Data | 80.4 % | 0.72 |
| Real Data+20% Gen. Data | 81.6% | 0.7 |
| Real Data+30% Gen. Data | 81.8 % | 0.75 |
| Real Data+40% Gen. Data | 82.4% | 0.77 |
| Real Data+50% Gen. Data | 84.2% | 0.80 |
| Real Data+60% Gen. Data | 85.8 % | 0.80 |
| Real Data+70% Gen. Data | 86.4% | 0.82 |
| Real Data+80% Gen.Data | 88.2% | 0.84 |
| Real Data+90% Gen. Data | 88.6% | 0.86 |
| Real Data+100% Gen. Data | 89.4% | 0.85 |
| Real Data+110% Gen. Data | 90.2% | 0.88 |
| Real Data+120% Gen. Data | 92.2% | 0.90 |
| Real Data+130% Gen. Data | 92.8% | 0.88 |
| Real Data+140% Gen. Data | 94.4% | 0.92 |
| Real Data+150% Gen. Data | 95.6% | 0.92 |

## REFERENCES


[1] M. Ferrari and V. Quaresima, "A brief review on the history of human functional near-infrared spectroscopy (fnirs) development and fields of application," *Neuroimage*, vol. 63, no. 2, pp. 921–935, 2012.

[2] M. A. Yücel, J. J. Selb, T. J. Huppert, M. A. Franceschini, and D. A. Boas, "Functional near infrared spectroscopy: enabling routine functional brain imaging," *Current opinion in biomedical engineering*, vol. 4, pp. 78–86, 2017.

[3] S. Bak, J. Park, J. Shin, and J. Jeong, "Open-access fnirs dataset for classification of unilateral finger-and foot-tapping," *Electronics*, vol. 8, no. 12, p. 1486, 2019.

[4] A. Antoniou, A. Storkey, and H. Edwards, "Data augmentation generative adversarial networks," *arXiv preprint arXiv:1711.04340*, 2017.

[5] K. G. Hartmann, R. T. Schirrmeister, and T. Ball, "Eeg-gan: Generative adversarial networks for electroencephalograhic (eeg) brain signals," *arXiv preprint arXiv:1806.01875*, 2018.

[6] S. M. Abdelfattah, G. M. Abdelrahman, and M. Wang, "Augmenting the size of eeg datasets using generative adversarial networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–6.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, "How generative adversarial networks and their variants work: An overview," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–43, 2019.

[9] J. Gauthier, "Conditional generative adversarial nets for convolutional face generation," *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester*, vol. 2014, no. 5, p. 2, 2014.