Enhancing the Machine Learning-based Detection of Long-lasting Continuous Gravitational Waves

Lorin A. Bernard & Dhan Lord B. Fortela*
Department of Chemical Engineering, University of Louisiana at Lafayette, Lafayette, Louisiana 70504, United States
*Email: dhanlord.fortela@louisiana.edu

Purpose:

The detection of gravitational waves (GW) has become a growing target of space exploration and research. Gravitational wave detectors such as the National Science Foundation's Laser Interferometer Gravitational-Wave Observatory (LIGO) detectors of 2015 pioneered the detection pathway (Figure 1). Recently, the focus has been shifted to the more informative task of detecting low-frequency range GW signals [1,2]. These waves would be long-lasting in theory however the difficulty ensues with isolating noise signals from the GW signals.

A key task in the goal of continuous GW detection includes establishing data analytics pipeline to train machine learning models for the detection of continuous GW signals based on models of ground-based detectors LIGO and Virgo. The European Gravitational Observatory (EGO) group hosted an international competition titled "G2Net Detecting Continuous Gravitational Waves" resulting in the highest model performance of around 84.9% prediction accuracy on the target signals [1].

The aim of this research project is to contribute to efforts in refining data analytics to increase the prediction accuracy (beyond 84.9%) by implementing advanced machine learning techniques.

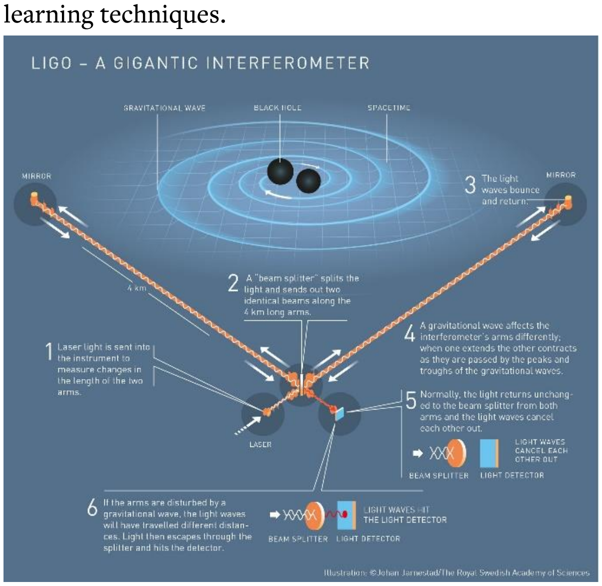


Figure 1: A general schematic of the LIGO Interferometer used in ground-based (Earth-based) detection of GW. International collaborations are currently being done to improve ground-based detectors and to establish space-based satellite detectors for GW. Image source: [4].

Acknowledgements:

This research project is sponsored by the Louisiana Space Grant Consortium (LaSPACE) and NASA under the LURA Program 2023-2024.

Objectives:

- Collection and organization of simulated continuous GW signals based on on-ground detector models by G2Net collaboration group.
- Collection of empirical raw GW signals from readily available online repositories of LIGO and EGO.
- Python codes implementation of machine learning are written in Jupyter Notebooks data analysis workflow to (Obj. 1) train machine learning models, and (Obj. 2) assess prediction performance of trained machine learning models on the test data subset of continuous GW.

Methodology:

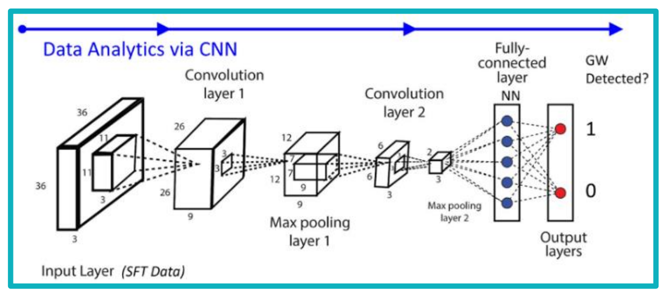
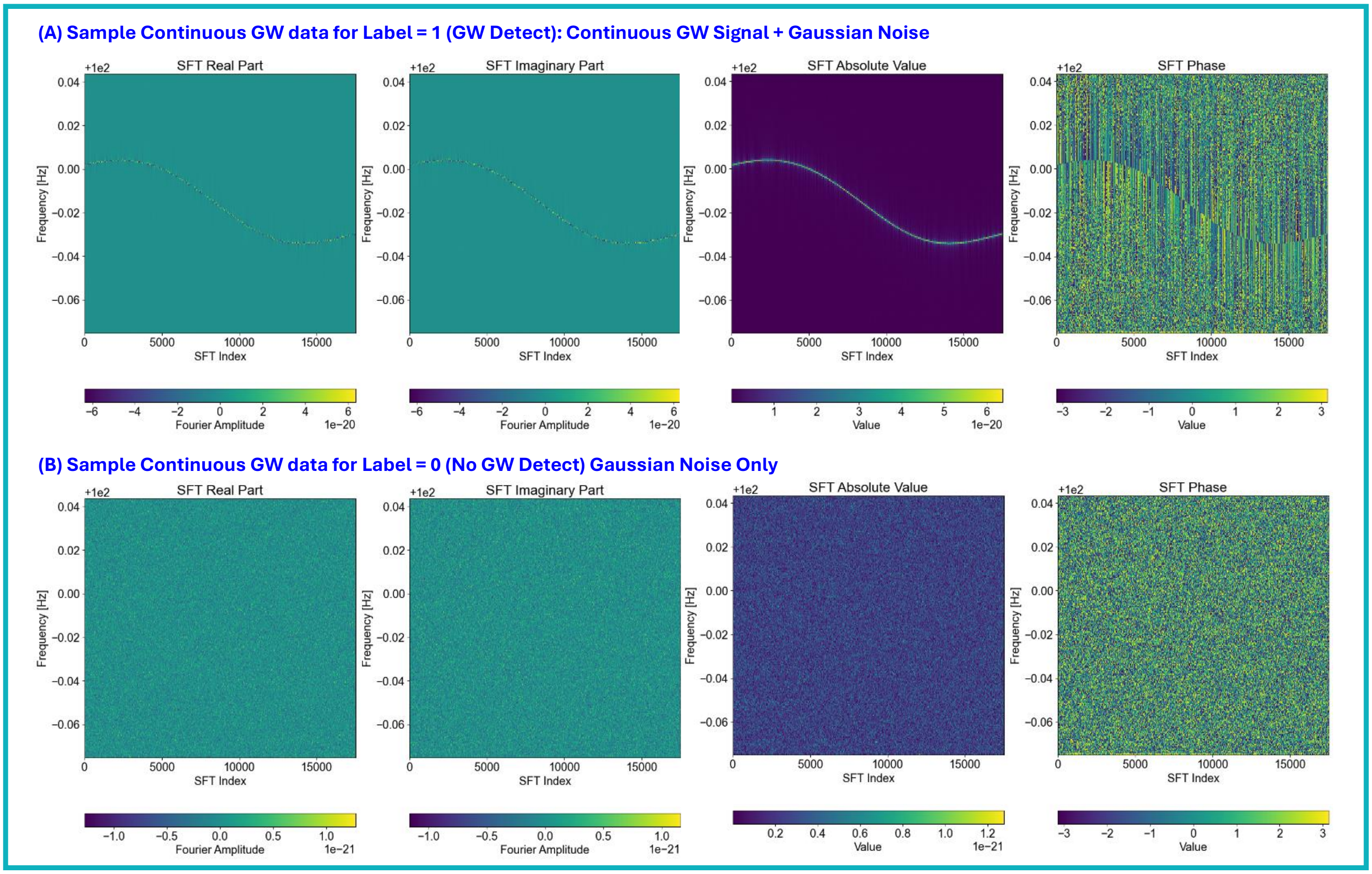


Figure 2: Training of a Binary Classifier using Convolutional Neural Network (CNN) model for the extraction of features from the GW data followed by forward Deep Neural Network (DNN) [3]. The CNN model predicts two class levels: 1 = GW Detected, and 0 = No GW Detected; hence, the model is a Binary Classifier.

1. Generate GW signal covering a 1-year (365 days) duration using PyFstat and LALSuite library [1].
2. Extract GW wave signal into Short Fourier Transform (SFT) wavelets [1].
3. Use SFT wavelet properties as Input Data to a CNN machine learning model for training and testing of a Binary Classifier [2,3].
4. Tune candidate CNN models and select best candidates using classification metrics Accuracy, Receiver Operating Characteristic (ROC) Curve, and Area Under the ROC Curve (AUC) [3].
5. Organize the Python codes used for the data analytics workflow and for the definition of CNN model configurations evaluated in the project.
6. Create an online repository of the Python code files and key results via GitHub. Set the online repository as public repo to allow open-access by interested developers who will use the project materials.

Results:

Figure 3: *Example training and testing data of Continuous GW signals generated with LALSuite GW library and PyFstat.*



Results 1: Using 2-layer CNN Model

The results of implementing a 2-layer CNN model as schematically depicted in Figure 2 are shown in Figure 3 and Figure 4 and Table 1. The classification performance of these 2-layer CNN exceeded the performance target of 84.9% (Table 1 and Figure 4). Other CNN models were then evaluated to further improve the detection performance.

Table 1*: Classification performance of the 2-layer CNN models (Figure 2) implemented during the preliminary evaluation phase. The corresponding ROC curves are shown in Figure 4.*

Model	AUC	Accuracy
CNN1b	**0.9866**	**86.9% (> 84.9% target)**
CNN2a	**0.9783**	**85.4% (> 84.9% target)**
CNN1a	0.9346	82.1% (< 84.9% target)

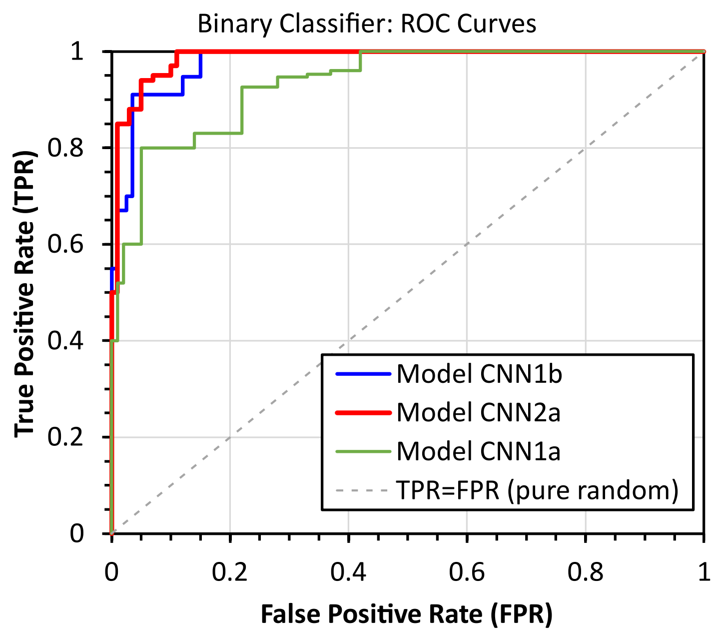


Figure 4: *Prediction performance of the trained CNN models on the test data set (50 data samples). PyTorch NN module was used for CNN model training and testing [3].*

Results 2: Using 3-layer and 4-layer CNN Models

Two (2) more advanced CNN models were tested to further improve the detection performance of the machine learning models. The CNN modeling results are shown in Figure 5 and Figure 6. The model configurations for these models can be accessed via their corresponding Python code files as provided in the GitHub repository of the project: https://github.com/dhanfort/LaSPACE_LURA2324_DetectContinuousGW.git. Some portions of the Python codes used were based on the codes developed by [5].

These 3-Layer and 4-Layer CNN models further increased the detection accuracy to 98.5% (Figure 5) and 97.1% (Figure 6), respectively, which are well above the target accuracy of 84.9% (baseline metric of performance for this LURA project).

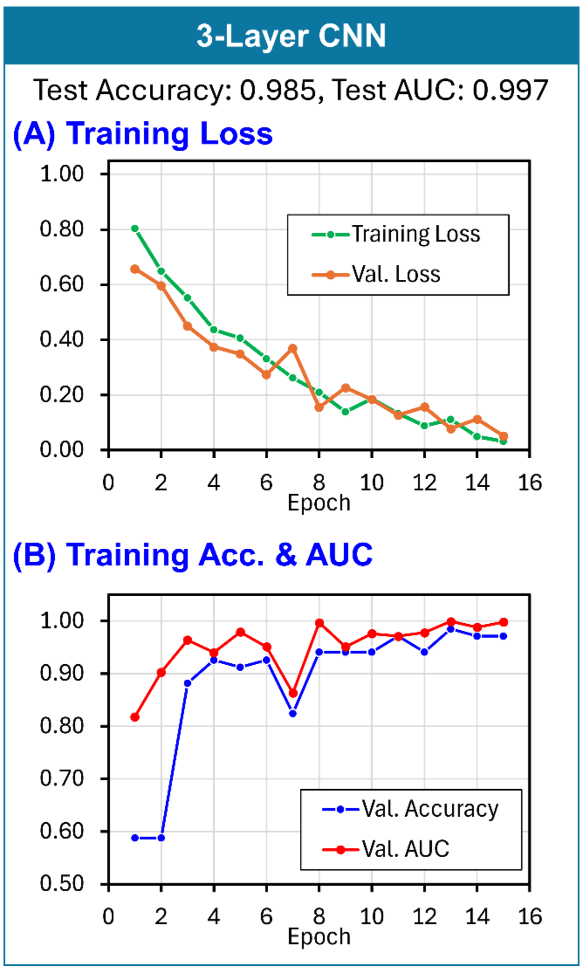


Figure 5: *Performance of the 3-Layer CNN showing Test Accuracy of* ***0.985*** *(>0.849 target) and AUC = 0.997. (A) Model loss values for the Training data subset and Validation (Val.) data subset decreasing as the 3-Layer CNN model is iteratively trained as number of Epoch increases. A loss value close to zero(0) is an indication of good model fitting to the data. (B) model prediction performance parameters Accuracy and AUC of the 3-Layer CNN using the Validation data subset. Accuracy and AUC values close to 1.0 are indications of model prediction performance.*

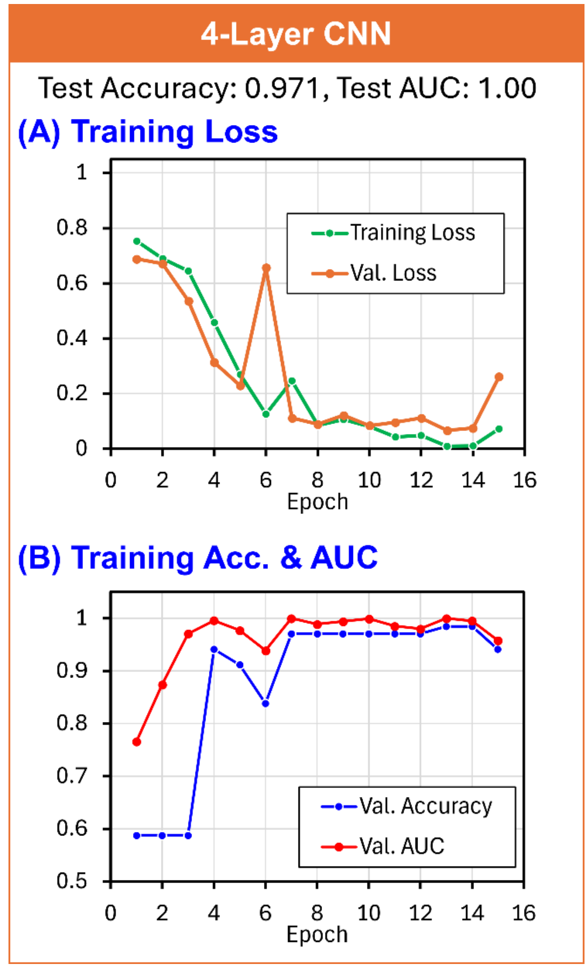


Figure 6: *Performance of the 4-Layer CNN showing Test Accuracy of* ***0.971*** *(>0.849 target) and AUC = 1.00. (A) Model loss values for the Training data subset and Validation (Val.) data subset decreasing as the 4-Layer CNN model is iteratively trained as number of Epoch increases. A loss value close to zero(0) is an indication of good model fitting to the data. (B) model prediction performance parameters Accuracy and AUC of the 4-Layer CNN using the Validation data subset. Accuracy and AUC values close to 1.0 are indications of model prediction performance.*

Conclusion:

The machine learning-based detection of long-lasting continuous gravitational waves can be enhanced using the Convolutional Neural Network (CNN) architecture.

References:

[1] G2Net (2022) URL:https://www.kaggle.com/competitions/g2net-detecting-continuous-gravitational-waves

[2] EGO (2024) URL: https://www.ego-gw.it/

[3] PyTorch (2024) URL: https://pytorch.org/

[4] Johan Jarnestad, The Royal Swedish Academy of Sciences (2023) URL: https://www.nobelprize.org/prizes/physics/2023/press-release/

[5] LucasRr (2023) URL: https://github.com/LucasRr/G2Net_gravitational_wave_detection.git