Gravitational Wave Detection and Parameter Estimation Using Deep Neural Networks

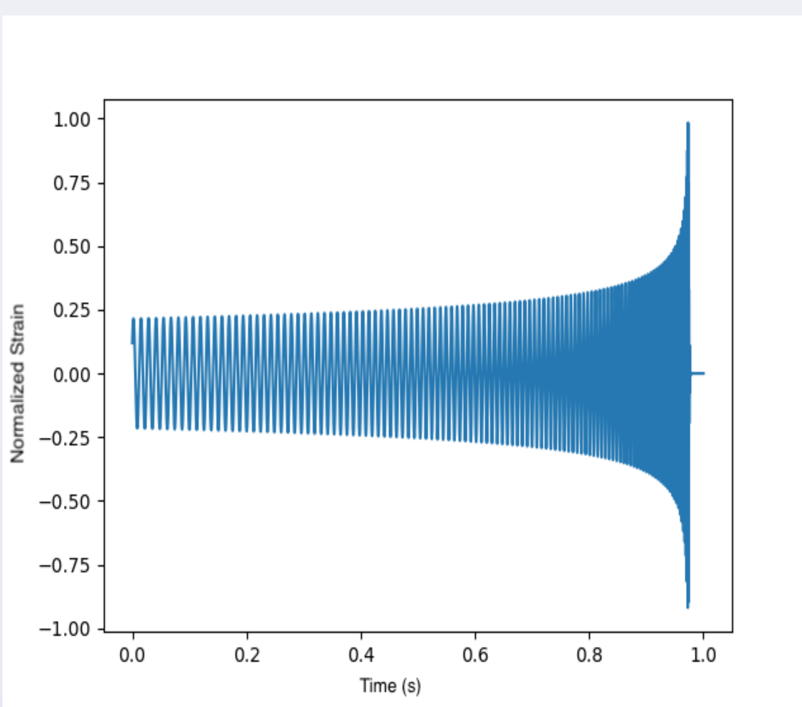
Rui Lan, Jiaxi Nie, Dongwei Shi

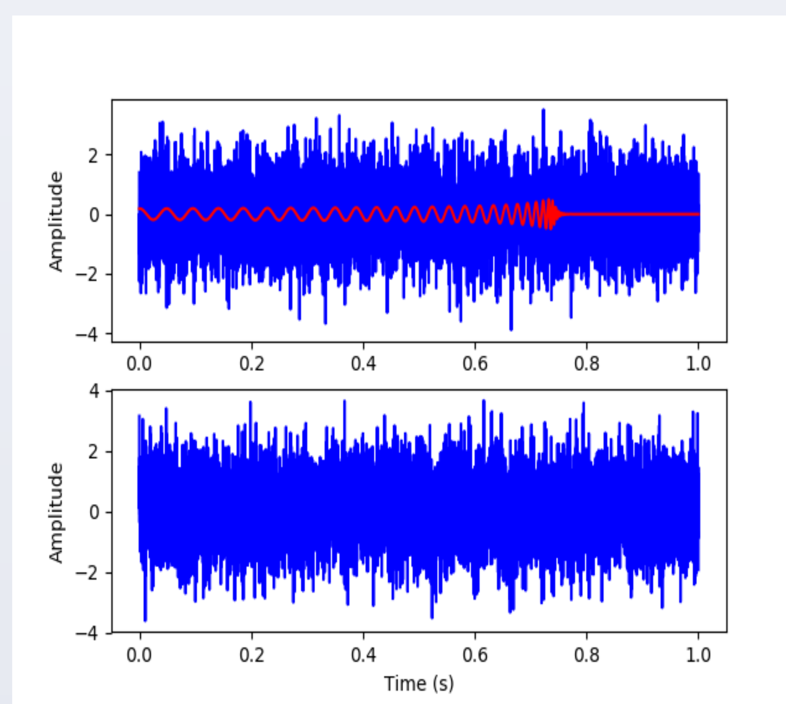
University of Illinois at Urbana-Champaign

Abstract

Gravitational wave (GW), predicted by Albert Einstein in 1916 based on his General Theory of Relativity, was first detected in 2015 by the Laser Interferometer Gravitational-Wave Observatory (LIGO). This year, the leading scientists in the Royal Swedish Academy of Sciences won the Nobel Prize in physics based on their contributions to GW detection of merging black holes. These all make GW a promising and exciting research area. In this report, we show how deep neural network can be applied to GW detection and parameter estimation. We also introduce our multi-GPU training approach which can significantly reduce the training time of deep neural networks while preserving the accuracy.

Gravitational Wave Dataset





Left: A simulated GW time-series, randomly picked in our train dataset.

Right: The upper plot shows a GW injected with Gaussian noise (blue) and the real GW (red), generated with signal-to-noise (SNR) of 0.5. The lower plot shows the pure Gaussian noise (blue). We observe that the noisy GW highly resembles the pure Gaussian noise, and we show below that deep learning approach is able to recover the GW and estimate its parameters (masses and spins of the two black holes.)

Methods

GW Detection

We construct a deep neural network to detect incoming Gravitational Waves. The neural network is called "classifier" in our system. During the inference, the classifier takes signal as input and generate the label to show whether the signal is GW or not.

	Input	vector	(size: 8192)
1	Reshape	matrix	(size: 1 × 8192)
2	Convolution	matrix	(size: 64 × 8177)
3	Pooling	matrix	(size: 64 × 2044)
4	ReLU	matrix	(size: 64 × 2044)
5	Convolution	matrix	(size: 128 × 2014)
6	Pooling	matrix	(size: 128 × 503)
7	ReLU	matrix	(size: 128 × 503)
8	Convolution	matrix	(size: 256 × 473)
9	Pooling	matrix	(size: 256 × 118)
10	ReLU	matrix	(size: 256 × 118)
11	Convolution	matrix	(size: 512 × 56)
12	Pooling	matrix	(size: 512 × 14)
13	ReLU	matrix	(size: 512 × 14)
14	Flatten	vector	(size: 7168)
15	Linear Layer	vector	(size: 128)
16	ReLU	vector	(size: 128)
17	Linear Layer	vector	(size: 64)
18	ReLU	vector	(size: 64)
19	Linear Layer	vector	(size: 2)
	Output	vector	(size: 2)

The left table shows the architecture of our deep neural network. We add another softmax layer at the final stage of the network and use it to generate Boolean outputs, either [0, 1] or [1, 0].

During the training, we inject Gaussian noise with SNR equals to 0.25, which is an SNR lower than typical GW. We also randomly shift the time-series to simulate the real-life situation where the detected GW may not be a signal contain full information for whole one second. The learning rate we use is 0.001.

GW Parameter Estimation

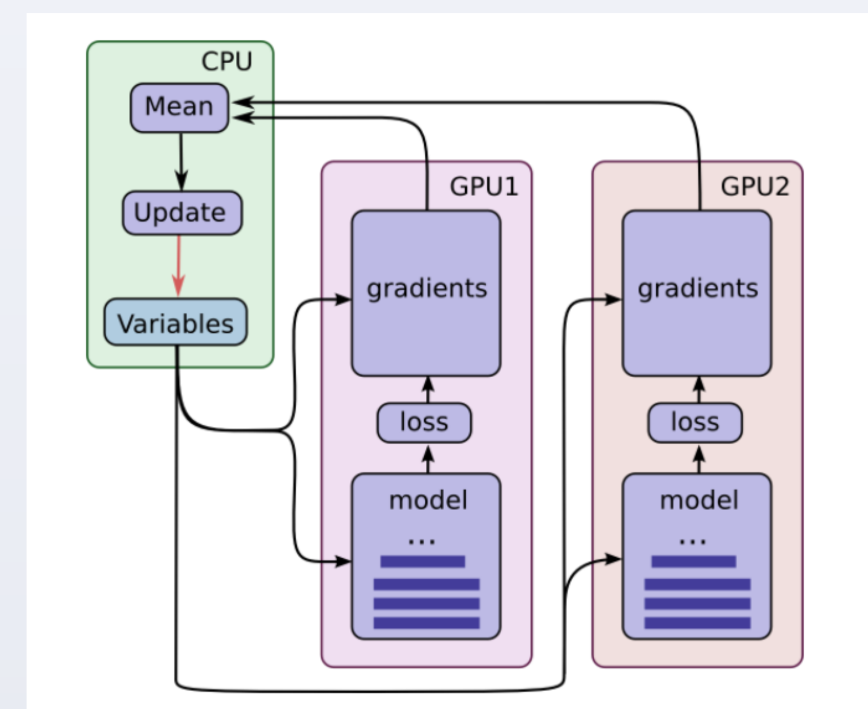
We also construct another deep neural network to extract the information of detected GW. The information includes masses and spins of two merging black holes. In our project, we name this network as "predictor". The predictor is a 4-parameter regression problem which requires deeper neural network and more powerful training strategy.

In this project, we use another neural network which is similar to our classifier except for the size of filters is larger than the classifier and there is no softmax layer at last stage. We construct several different architectures, and this one performs better. As the nature of regression problem requires more training resources and typically hard to converge, we make some changes during training compared to the classifier.

We first reduce the learning rate to 0.0001. Also, the neural network trained by using a fixed SNR cannot work well in an extensive range of SNR during the testing stage. Thus, we construct a function to decrease SNR from 100 to 3 as a function of training step. We expect this method can help our predictor remember the signals at a wide range of SNR so that it can perform better when testing.

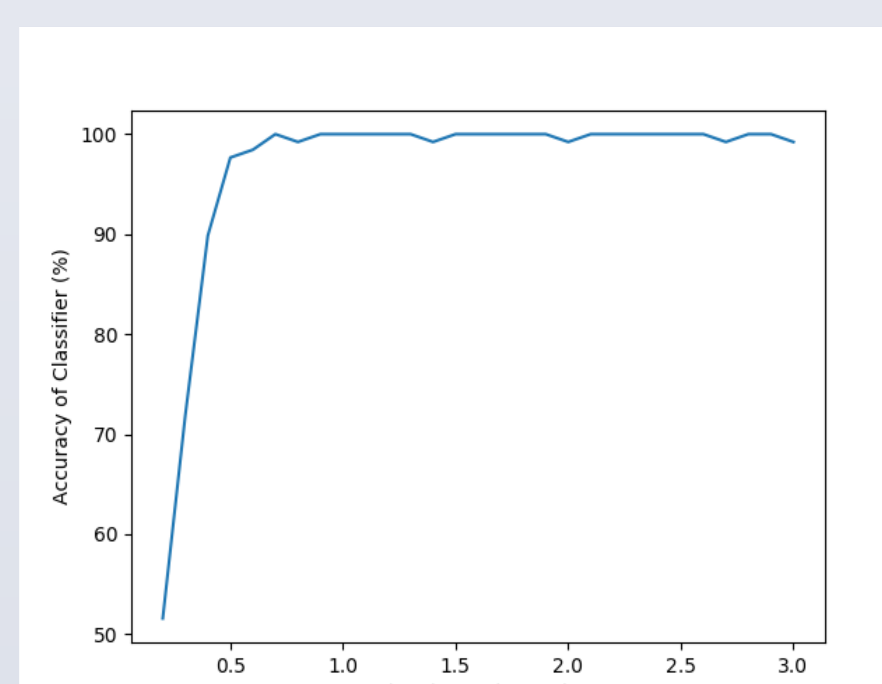
Multi-GPU Training

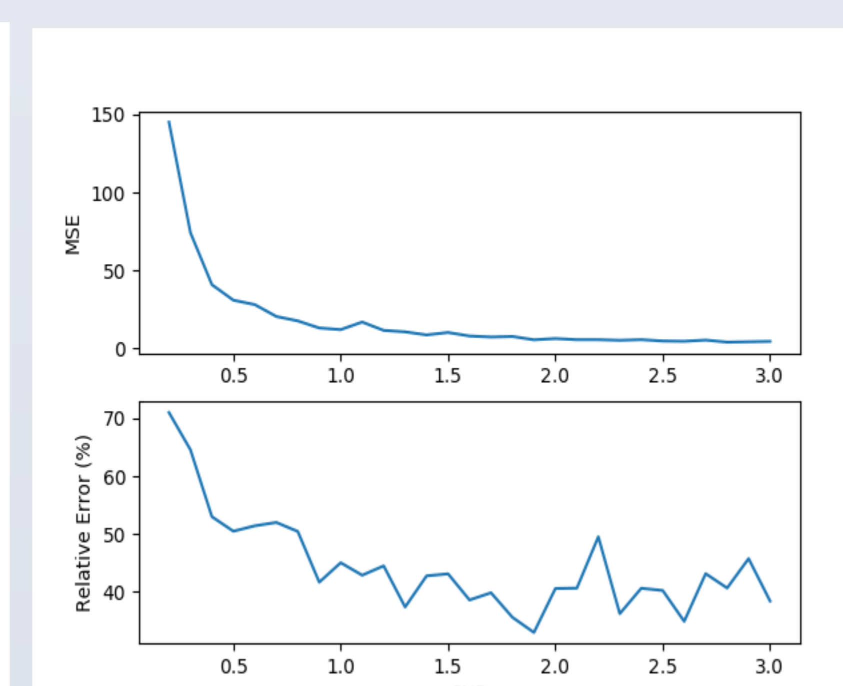
It has shown that the deep learning approach of GW detection significantly reduce the testing time complexity during the inference compared with traditional matched filtering approach used by LIGO. However, the time required to train a deep neural network is still too long. For example, it takes us 17 hours to train a predictor for 10,000 steps with a batch size equals to 128. In this project, we also demonstrate a multi-GPU code which can join multiple GPU workers to reduce the training time.



As the diagram on the above shows, we first distribute the same amount of data into two different workers with two sets of trainable variables. After they finish their work, we compute the average gradient and update the parameters on CPU. By using this data-parallel method, we demonstrate a higher training efficiency while keeping the same accuracy.

Results and Conclusion



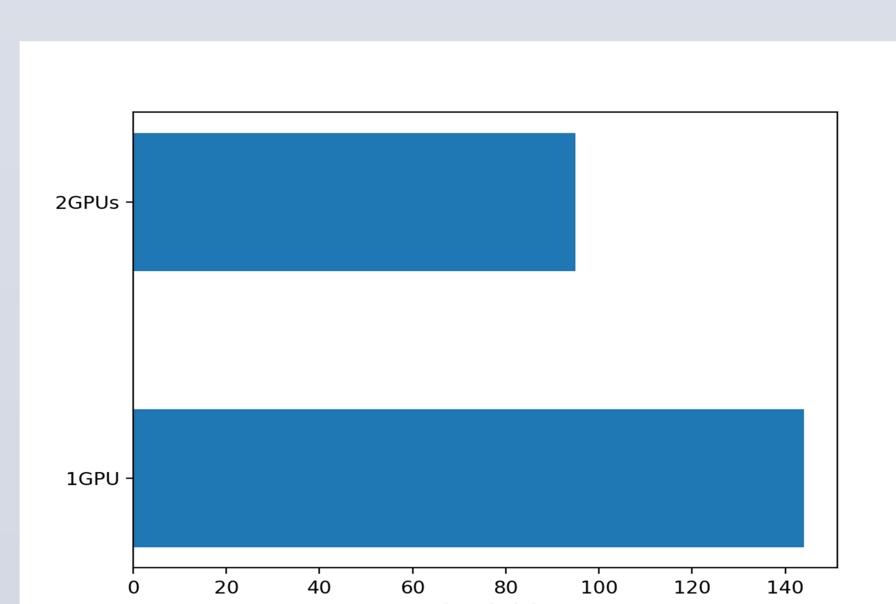


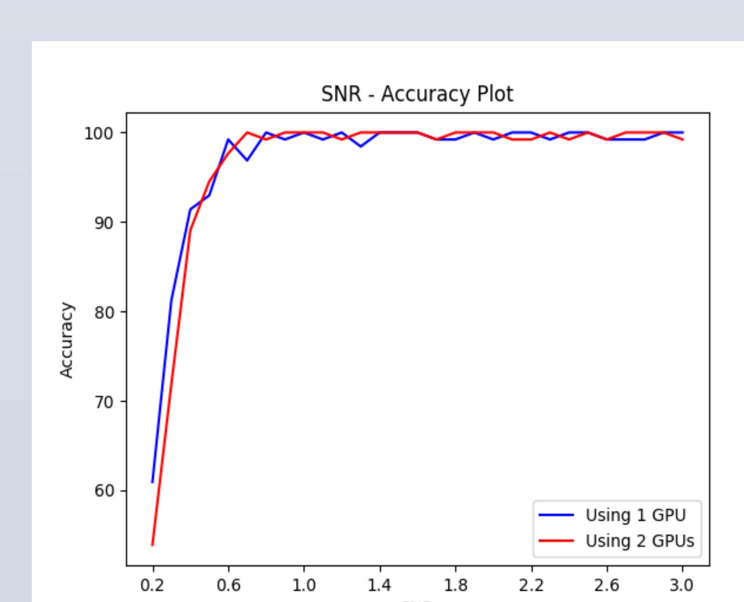
GW Detection (Left)

The result shows our classifier can achieve nearly 100% accuracy when GW's SNR higher than 0.5.

GW Parameter Estimation (Right)

The diagram on the right shows our predictor give our a low MSE when SNR is above 0.5. It demonstrates the decreasing SNR method can make the predictor perform well within a certain range of testing SNR.





Multi-GPU Training

The above result shows our multi-GPU code can reduce the training time to half of the original training time while keep the same accuracy during the testing, if we use 2 GPUs.

Acknowledgements

This project is one of the on-going projects at Deep Learning Team of Gravitational Wave Group at NCSA. We thank Prof. Zhizhen (Jane) Zhao and Daniel George for their guidance and help. We also express our gratitude to Dr. Eliu Huerta for granting us the access to Campus Cluster GPUs. Plus, We thank Prof. Paris Smaragdis and the TAs for giving us a great opportunity to study machine learning and signal processing techniques.