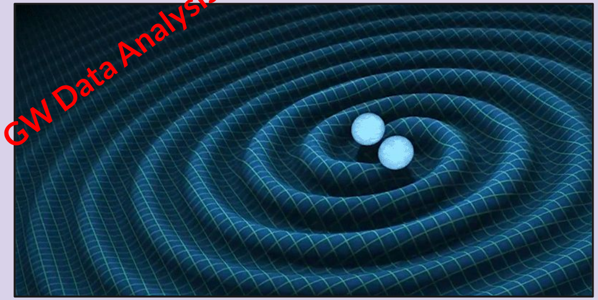
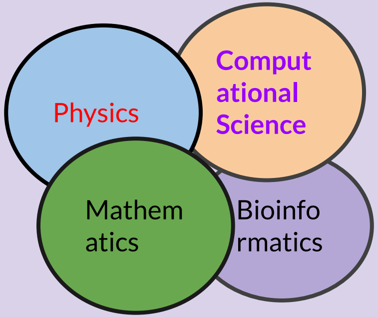
06-02-2023

Interview for Academy Scientist in Machine Learning in The Space Research Institute (IWF), Graz, Austria

Amit Reza
Nikhef, Amsterdam

What I do? Development of data driven models for Gravitational Waves Astronomy





Amit Reza
Postdoctoral Researcher,
Supervisor: Prof. Sarah Caudill
Joining Year - Feb 2021.

PhD in Gravitational Wave Data Analysis
Indian Institute of Technology Gandhinagar, Gujarat, India

Thesis Title: **Developing efficient search pipelines for detecting astrophysical gravitational wave signals using numerical linear algebra.**

Supervisor: Prof. Anand Sengupta (Physics discipline)
Co-supervisor: Prof. Anirban Dasgupta (Computer Science and Engineering)

Commonly Used Tools/ topics in my research projects

1. Signal Processing.
2. Numerical Linear Algebra.
3. Machine Learning / Deep Learning algorithms.
4. Bayesian Inference.
5. Scientific Coding (Python)

Deep Learning models

- CNN
- RNN
- GANs
- Auto-encoders
- Normalizing Flows
- Neural ODE
- PINNs

Flow of the presentation

- Part-I: My ML Background and possible links to the IWF (**10 minutes**)
    - My research domain: Gravitational Wave Data Analysis
    - My ML research projects: Application of CNN, RNN, GANs, Normalizing flow

- Part-II New Developments in ML (probably quantum computing) (**10 minutes**)
    - Quantum Computing: Quantum Machine Learning, Quantum Deep Learning
    - Physics informed Neural Network
    - Neural ODE/ PDE
    - Bayesian Neural Network

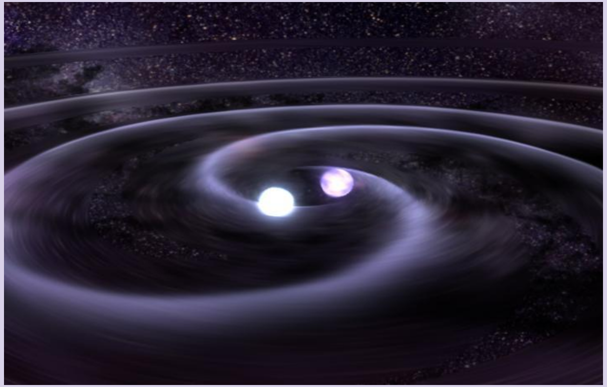
- Part-III: ML Test Problem: Tackling incomplete metal-oxide cluster data (**10 - 15 minutes**)
    - Problem Statement
    - Density Functional Theory
    - DFT and ML : ML for time dependent Schrodinger Equation

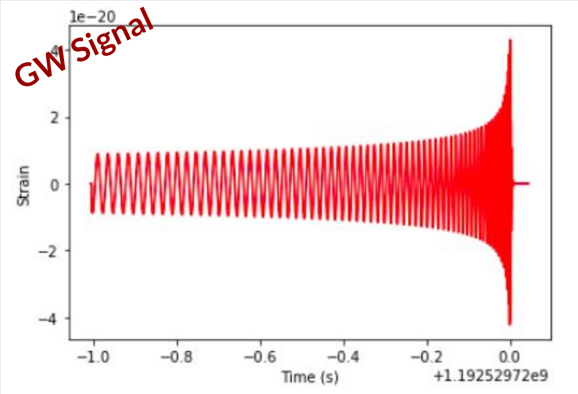
Part-I: ML background and possible links to the IWF

Duration: within 10 mins.

Gravitational wave searches from compact binary coalescence

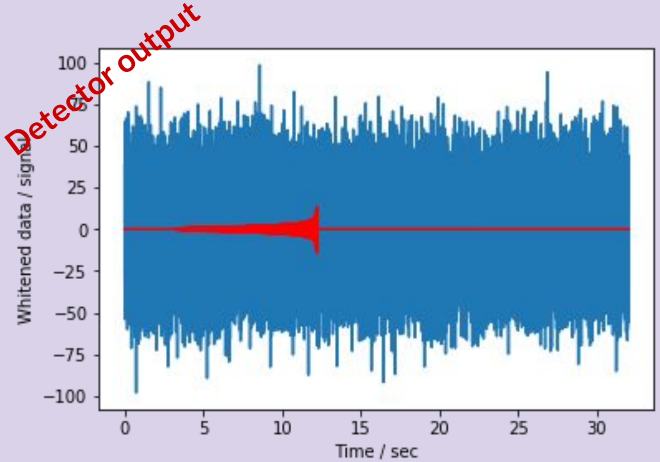
How to extract a faint GW signal buried in a noise?





Modeled searches

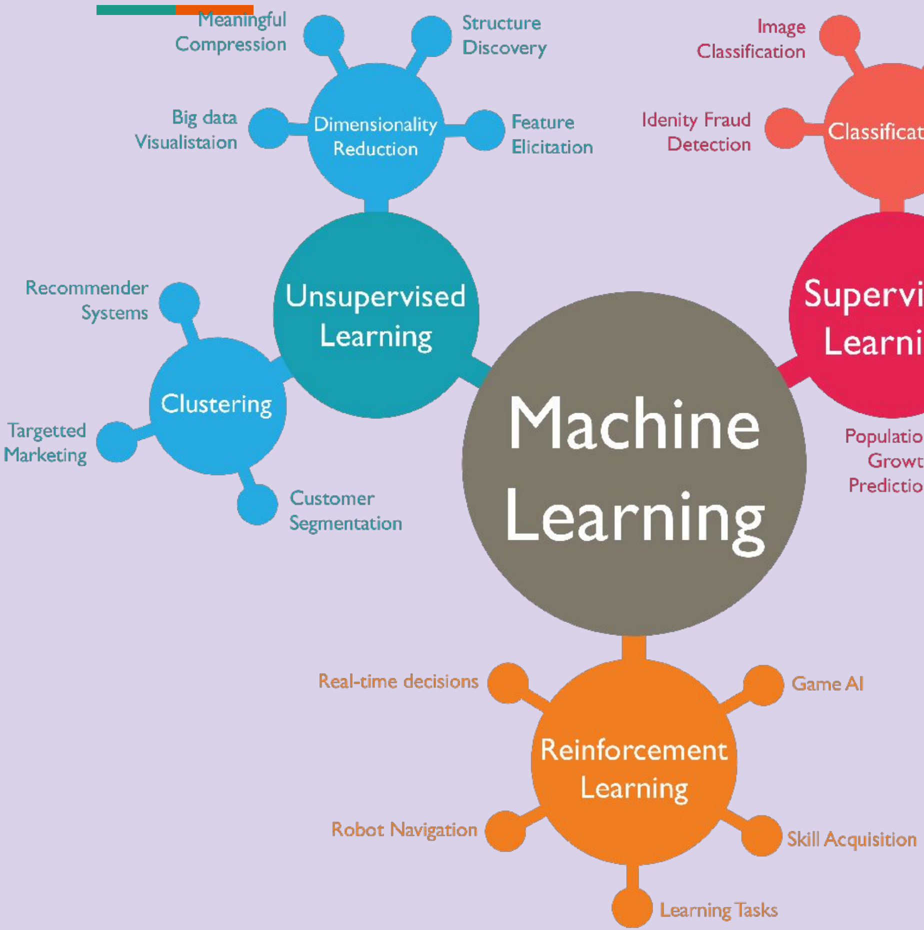
- Search for known signal shape in detector noise.
- Signal parameters are unknown.

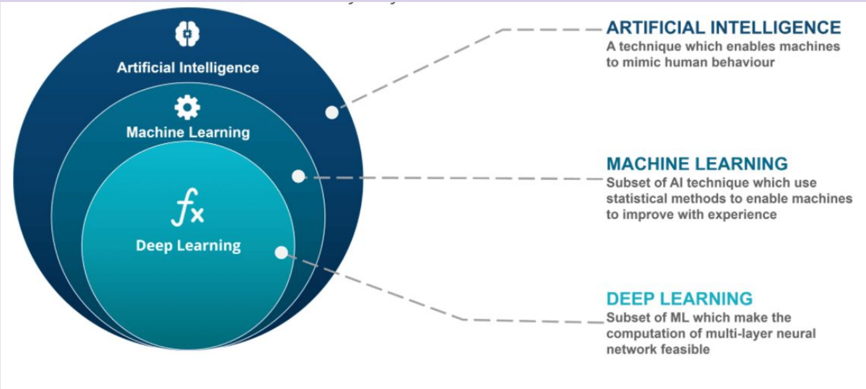


$$\vec{d} = \vec{s} + \vec{n}$$

Nearly real time analysis of the data is required.

Types of ML Algorithms:





My Contribution

- Dimension Reduction based fast computation of SNR time-series via Random Projection. PRD 99, 101503 (R) 2019.
- Randomized dictionary learning for signal representation. arxiv

- Fast ML based Bayesian Inference for rapid parameter estimation. arxiv
- Fast Multi-detector consistency Test via covariance matrix structure and KD Tree based NN search. PRD 101, 022003, 2020

- Classification of GW signals using CNN. arxiv

- Early warning framework for detecting BNS signal using CNN. PRD 103, 102003, 2021

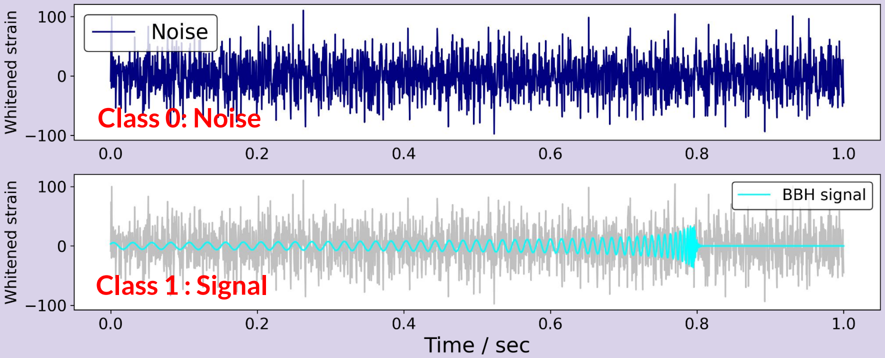
- Simulating Blip Glitches using Generative Adversarial Networks (GANs). PRD 106, 023027

- ML-Bayesian framework for identifying glitches to improve GW data quality. MIP

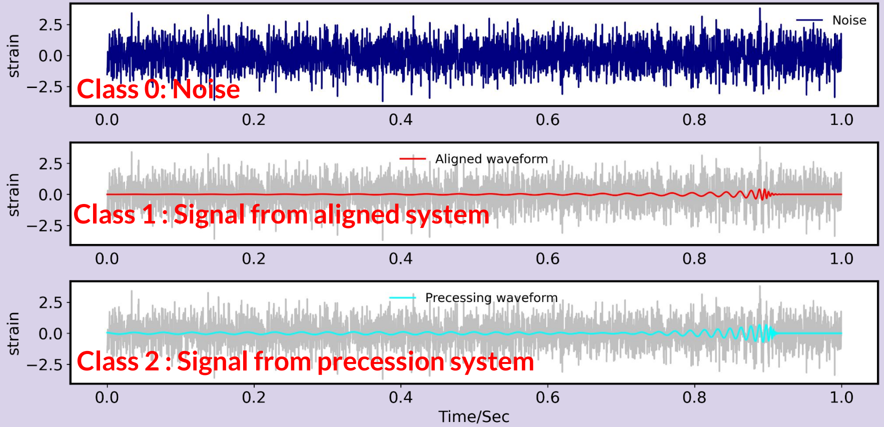
Detection of GW signal from precession system : Classification problem

arXiv:2206.12673

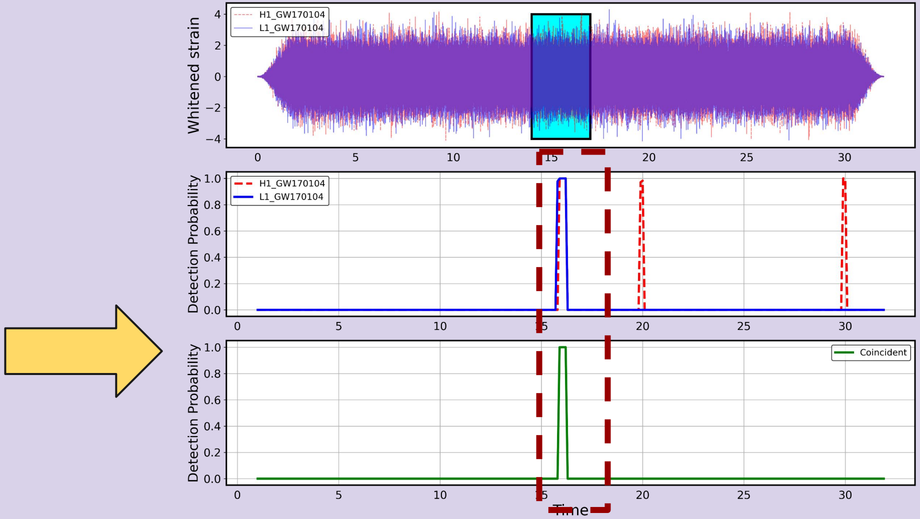
Binary classification

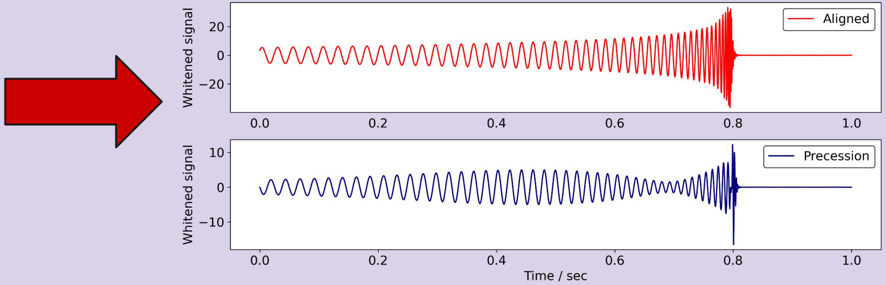


Multiclass classification



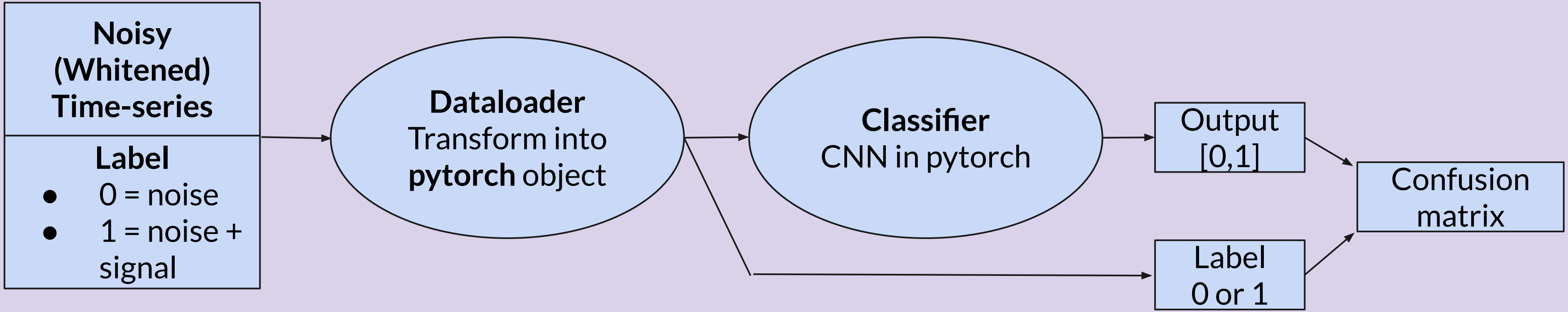
Multi-detector classification





CNN based search pipeline for the detection of GW signals

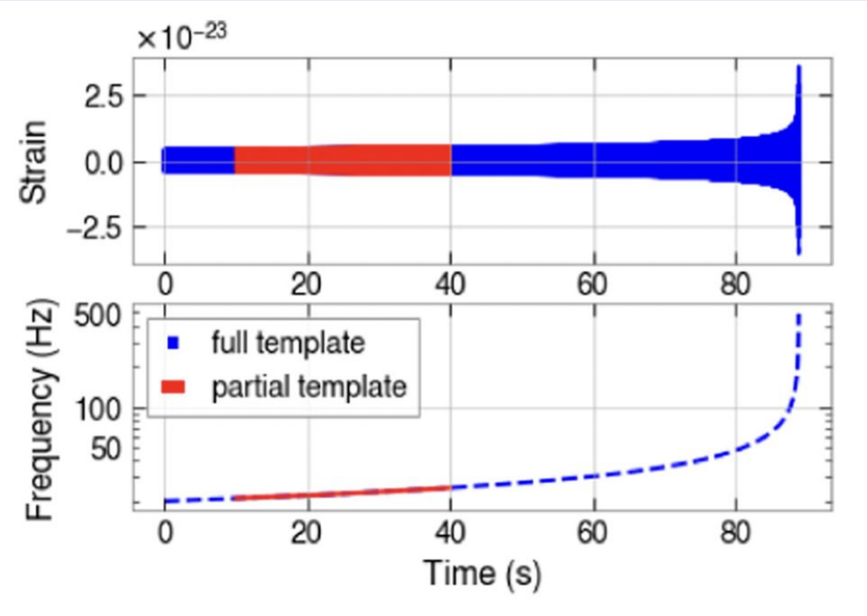
Our proposed pipeline : classification task between noise and noise + signal

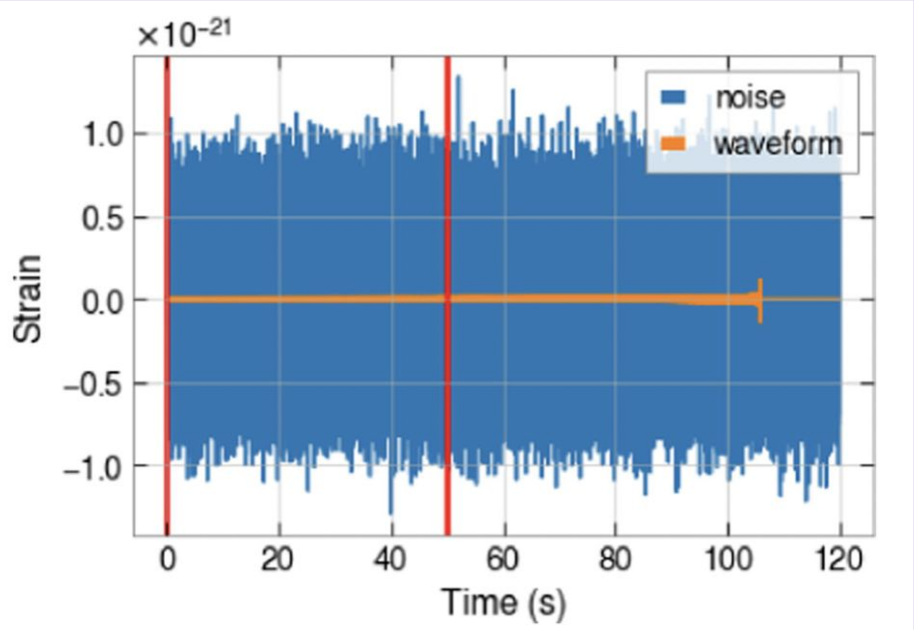


We used a threshold :
If output value > threshold
→ CNN claims a detection
If output value < threshold
→ CNN claims only noise

Early inspiral of a gravitational-wave signal

Objective: Detection of early inspiral GW signals from BNS system



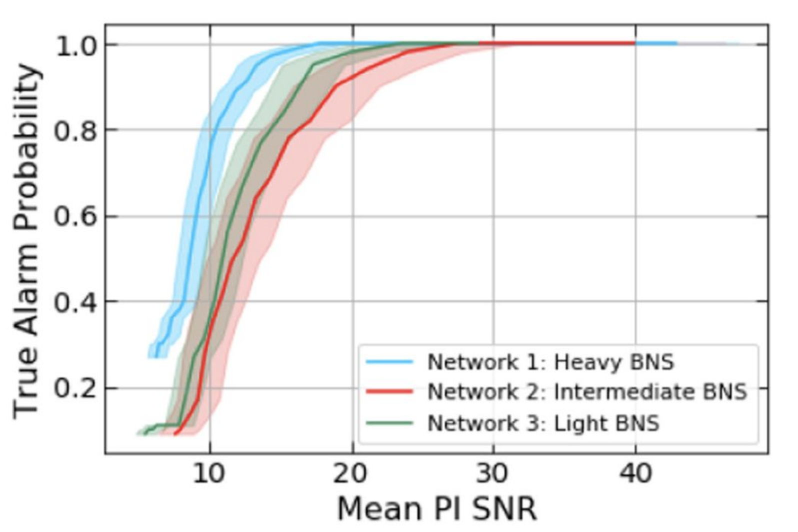


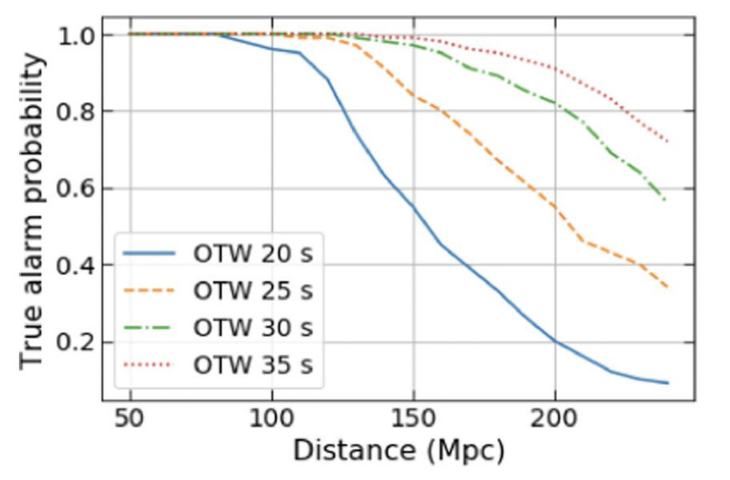
- Three different architectures are used.
- OTW is different for these architectures.

G.Baltus et. al. PRD 103, 102003 (2021)

Results

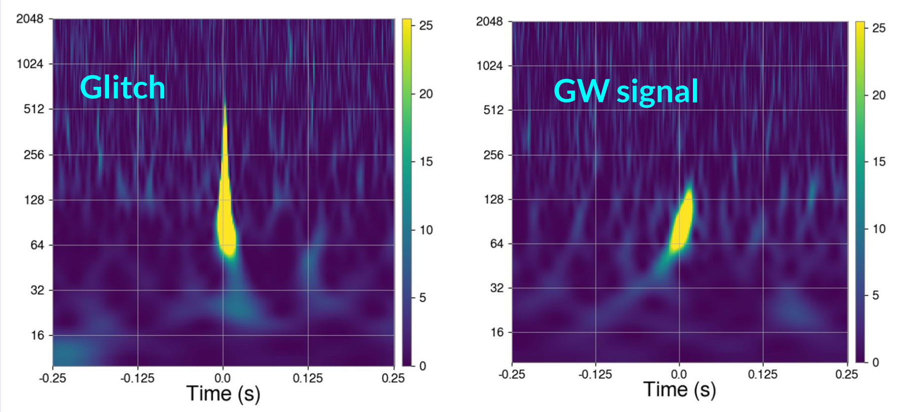
BNS	light	intermediate	heavy
$\mathcal{M}_c$ ($M_\odot$)	1.13 - 1.56	1.56 - 2.09	2.09 - 2.61
f_{low} (Hz)	20	20	20
Duration (s)	100 - 180	65 - 100	45 - 65
OTW (s)	80	50	30
Fraction of signal	0.44 - 0.8	0.5 - 0.77	0.46 - 0.66
Early alert before merger (s)	20 - 100	15 - 50	15 - 35



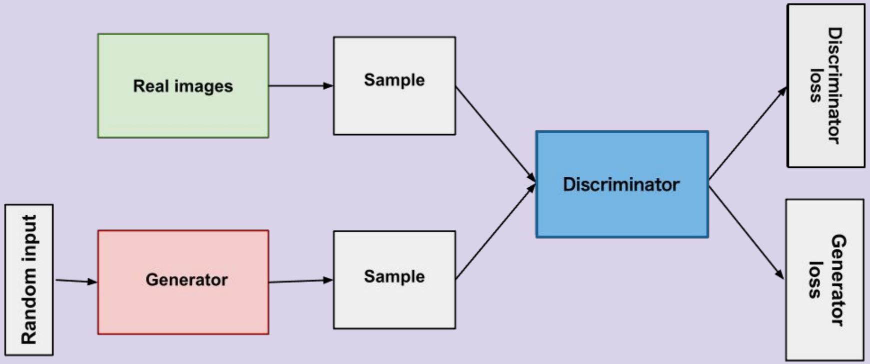


- Performance of the CNN trained on the heavy BNS systems for different OTWs.
- A longer window gives a higher number of detections.
- The detection accuracy can improve closer to the merger.

Simulating Transient Noise Bursts in LIGO with Generative Adversarial Networks

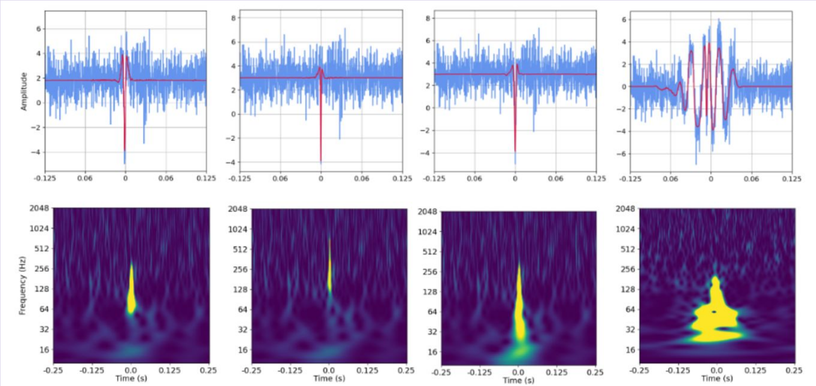


Glitch mimics GW signal



Generative Adversarial Networks:

- Used to learn the underlying distribution of the data
- Generate family of blip glitches.

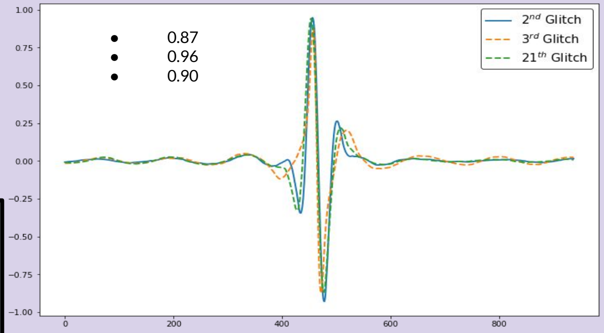


a) Glitch A b) Glitch B c) Glitch C d) Glitch D

[PRD.106.023027](PRD.106.023027)

Glitch Set -> How to use a glitch set for improving data quality?

The proposed framework



‘Gengli’: blip glitch generator → Filter data with glitches → Bayesian analysis:

It can produce tons of blip of glitches: “glitch bank”

$$\langle s(t), g(t)\rangle = 4\,\mathrm{Re}\int_0^\infty \frac{\tilde{s}(f)\,\tilde{g}^*(f)}{S_n(f)}\,df\,,$$

Obtain glitch triggers based on threshold on SNR

SVD based matched filtering can be adapted

$$g(\vec{\lambda}) = g_j A e^{-2\pi i f \delta t - i\phi}, \quad \text{where} \quad g_j \in G$$

For estimating
(a) Phase, (c) overall amplitude, (d) time.

$$\mathcal{L}(d \mid g(\vec{\lambda})) = -\frac{1}{2}(d - g(\vec{\lambda})|\, d - g(\vec{\lambda}))$$

How my experience can help projects related to IWF

- **ML for Space Plasma Physics**
  - Estimation of the short- term solar irradiance from sky images
  - Classification of Auroral Images From the Oslo Auroral THEMIS (OATH) Data Set, and extraction of local cloud cover information from all-sky-camera images using random forest classifier.

- **ML for magnetospheric physics**
  - For the identification of magnetic reconnection
  - T.M.Garton et al. (2021) developed a new NN model for identifying the reconnection signature in Saturn's magnetosphere through spherical magnetic field measurements.
  - Yeakel et al. (2022) implemented an RNN-LSTM-based neural architecture to learn sixty minutes of magnetometer data to classify Cassini's orbit regions as the magnetosphere, magnetosheath, or solar wind.

- **ML for the space weather forecasting**
  - Bailey et al. 2021) forecasted the geomagnetically induced currents (GICs) and geoelectric fields using a recurrent neural networks and LSTM architecture-based trained models to predict the geoelectric field components and GICs.

- **ML for Exoplanet characterization and evolution**
  - ML is used to infer the interior structure of low-mass exoplanets.
  - Bayesian Deep Learning algorithms are proposed to study the atmospheric composition of the exoplanets.

Part-II: New Developments in ML (probably in Quantum Computing)

Duration: 10 minutes

Quantum Machine Learning

Quantum Circuit:

- The fundamental element of quantum computing is the quantum circuit.
- Any quantum program can be represented by a sequence of quantum circuits and classical near-time computation.
- A quantum circuit is a computational routine consisting of coherent quantum operations on quantum data, such as **qubits**.

Qubit states:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

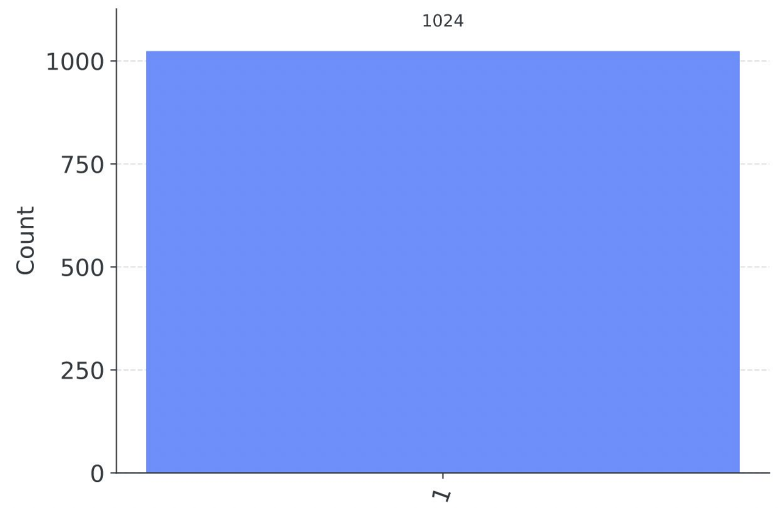
The states $|0\rangle$ and $|1\rangle$ form an orthonormal basis, we can represent any 2D vector with a combination of these two states.

$$|q_0\rangle = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} \end{bmatrix} \rightarrow |q_0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle$$

Quantum computing using Qiskit

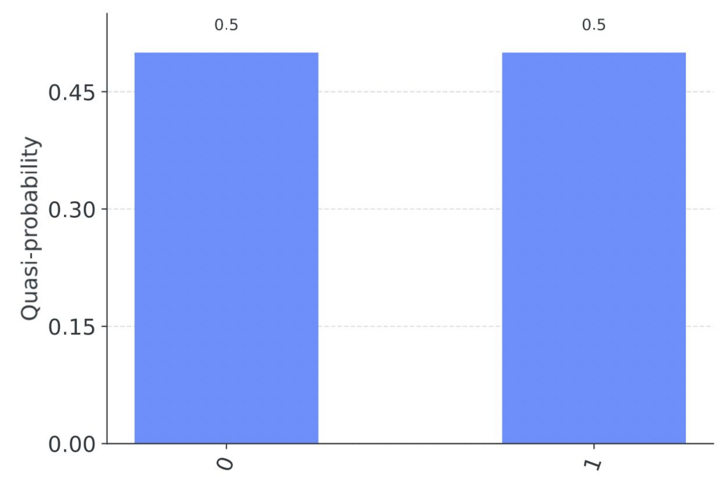
```
qc = QuantumCircuit(1)  # Create a quantum circuit with one qubit
initial_state = [0,1]   # Define initial_state as |1>
qc.initialize(initial_state, 0) # Apply initialisation operation to the 0th
qc.draw()  # Let's view our circuit
```

Source: qiskit



100% chance of measuring $|1\rangle$

initial_state = [1/sqrt(2), 1j/sqrt(2)]



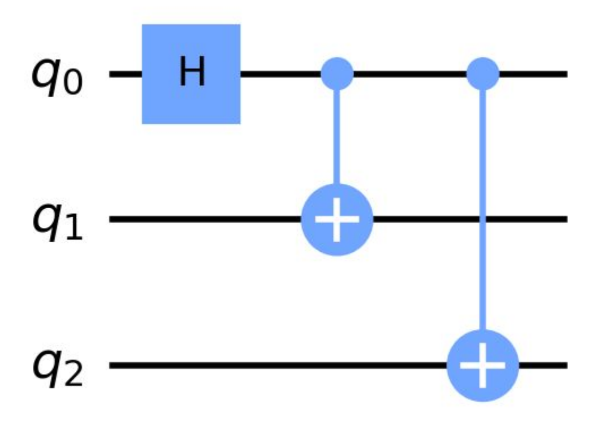
Equal probability of measuring either $|0\rangle$ or $|1\rangle$

Quantum Circuit

- A **Greenberger-Horne-Zeilinger (GHZ)** state is an entangled quantum state having extremely non-classical properties.

$$|GHZ\rangle = \frac{|0\rangle^{\otimes n} + |1\rangle^{\otimes n}}{\sqrt{2}}.$$

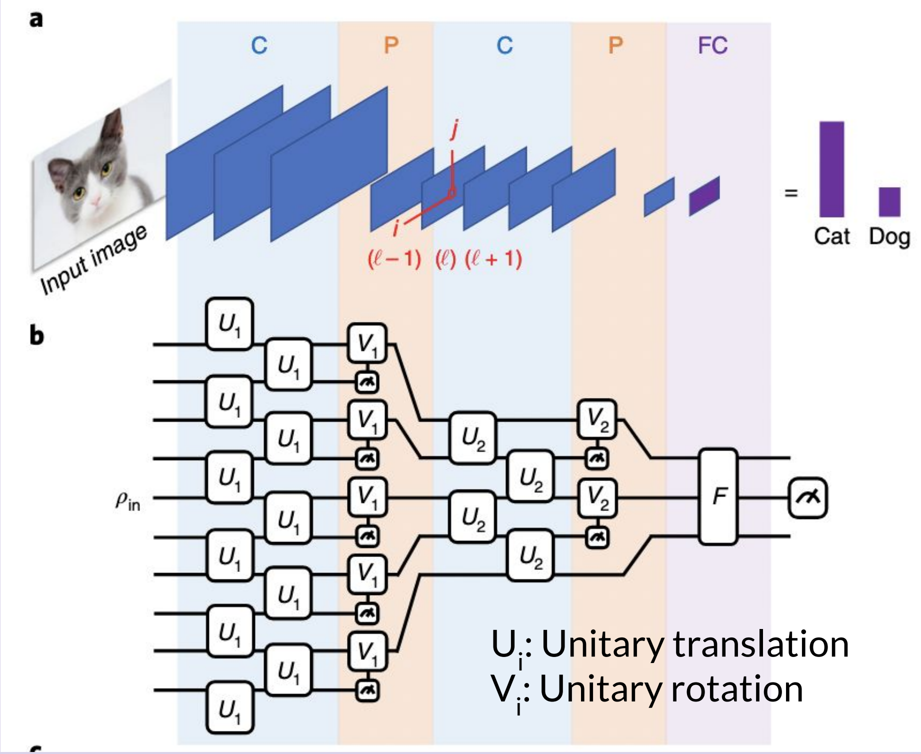
A simplest 3-qubit GHZ state: $$|GHZ\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$$



- H gate on qubit 0, putting this qubit in a superposition of |0> + |1>.
- A CX (CNOT) gate on control qubit 0 and target qubit 1 generating a Bell state.
- CX (CNOT) gate on control qubit 0 and target qubit 2 resulting in a GHZ state.

Source: qiskit

Quantum Convolutional Neural Network



U_i: Unitary translation
V_i: Unitary rotation

- Classical data => N-qubit input states => Quantum circuit representation.
- Error:

$$\mathrm{MSE} = \frac{1}{2M}\sum_{\alpha=1}^{M}\left(y_i - f_{\{U_i,V_j,F\}}(|\psi_\alpha\rangle)\right)^2$$

Binary classification outputs

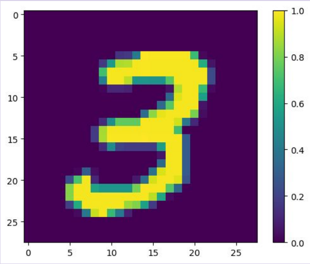
Expected QCNN output value

Input states

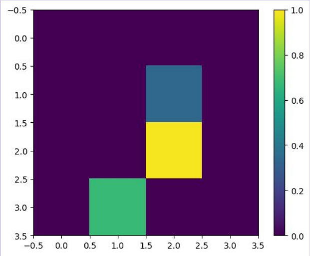
Source: I Cong et al. Nature Physics (2019)

Quantum CNN using tensorflow

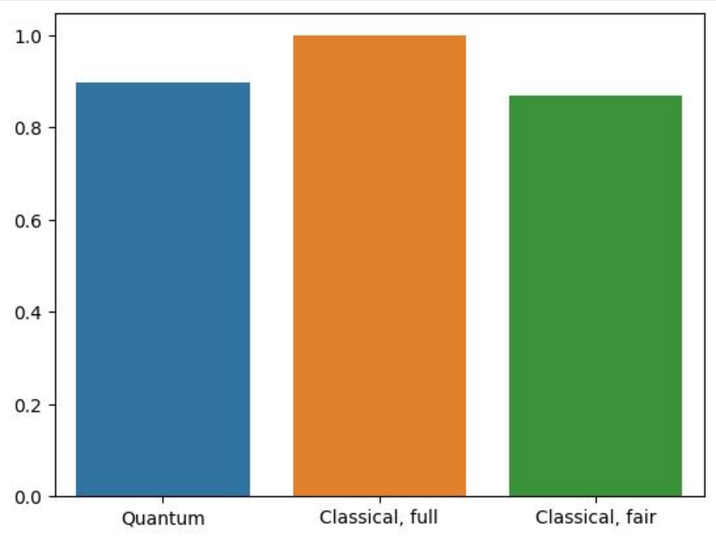
- Step -1: **Encode the data as quantum circuits**
- Step-2: **Quantum neural network**
    - Build the model circuit
    - Wrap the model-circuit in a tfq-keras model
    - Train the quantum model



4 x 4 grid



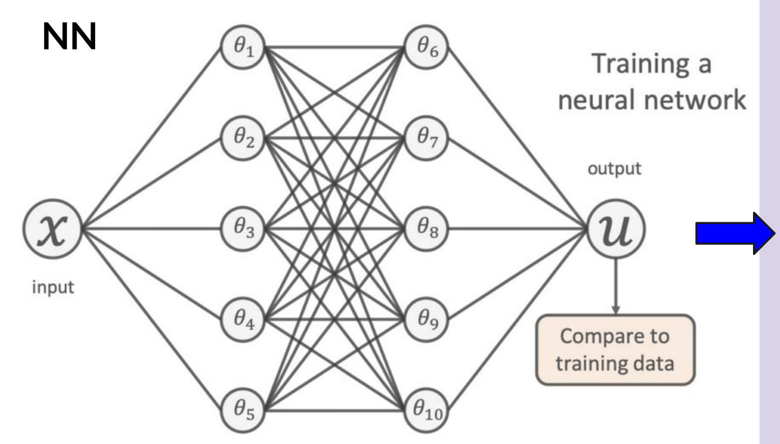
```python
def convert_to_circuit(image):
    """Encode truncated classical image into quantum datapoint."""
    values = np.ndarray.flatten(image)
    qubits = cirq.GridQubit.rect(4, 4)
    circuit = cirq.Circuit()
    for i, value in enumerate(values):
        if value:
            circuit.append(cirq.X(qubits[i]))
    return circuit


x_train_circ = [convert_to_circuit(x) for x in x_train_bin]
x_test_circ = [convert_to_circuit(x) for x in x_test_bin]
```



Source: quantum tensorflow

Physics informed Neural Network

NN

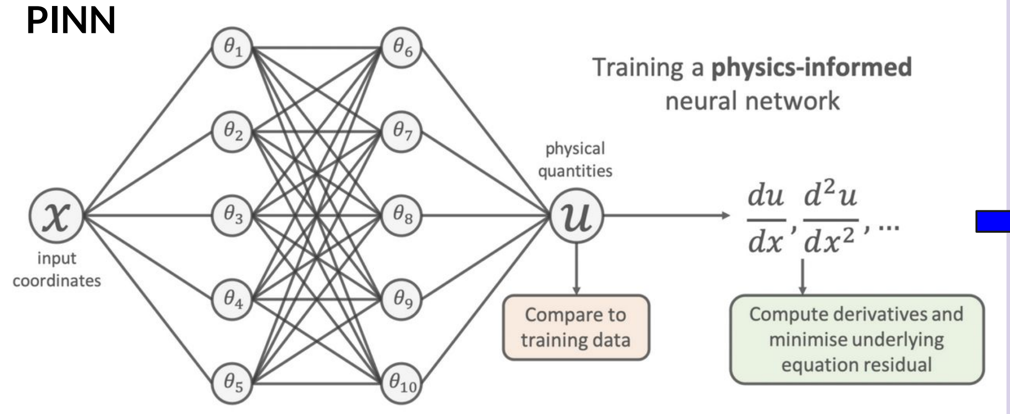


$$\min \frac{1}{N}\sum_{i}^{N}\left(u_{\mathrm{NN}}(x_i;\theta)-u_{\mathrm{true}}(x_i)\right)^2$$

Damped harmonic oscillator

$$m\frac{d^2u}{dx^2}+\mu\frac{du}{dx}+ku=0$$

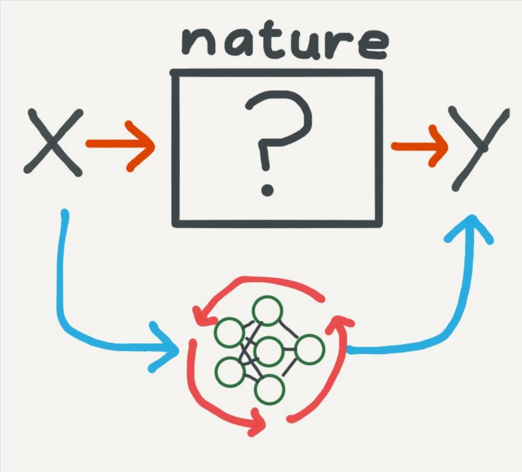
PINN

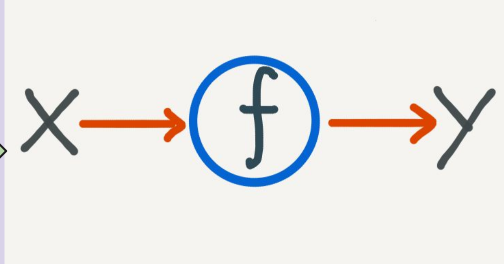


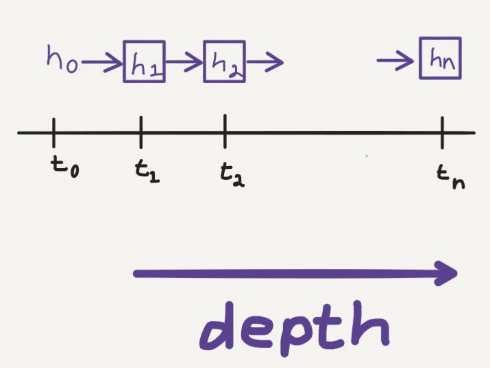
$$\min \frac{1}{N}\sum_{i}^{N}\left(u_{\mathrm{NN}}(x_i;\theta)-u_{\mathrm{true}}(x_i)\right)^2$$
$$+\frac{1}{M}\sum_{j}^{M}\left(\left[m\frac{d^2}{dx^2}+\mu\frac{d}{dx}+k\right]u_{\mathrm{NN}}(x_j;\theta)\right)^2$$

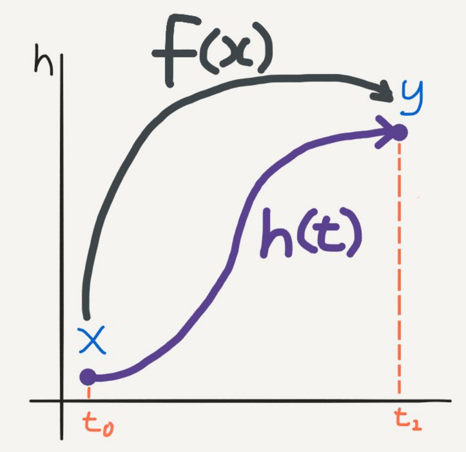
Source: Ben Moseley

Neural ODE



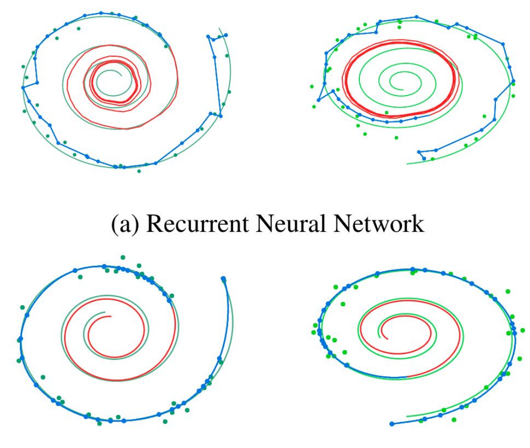






Assumption: Continuous hidden state dynamics over time

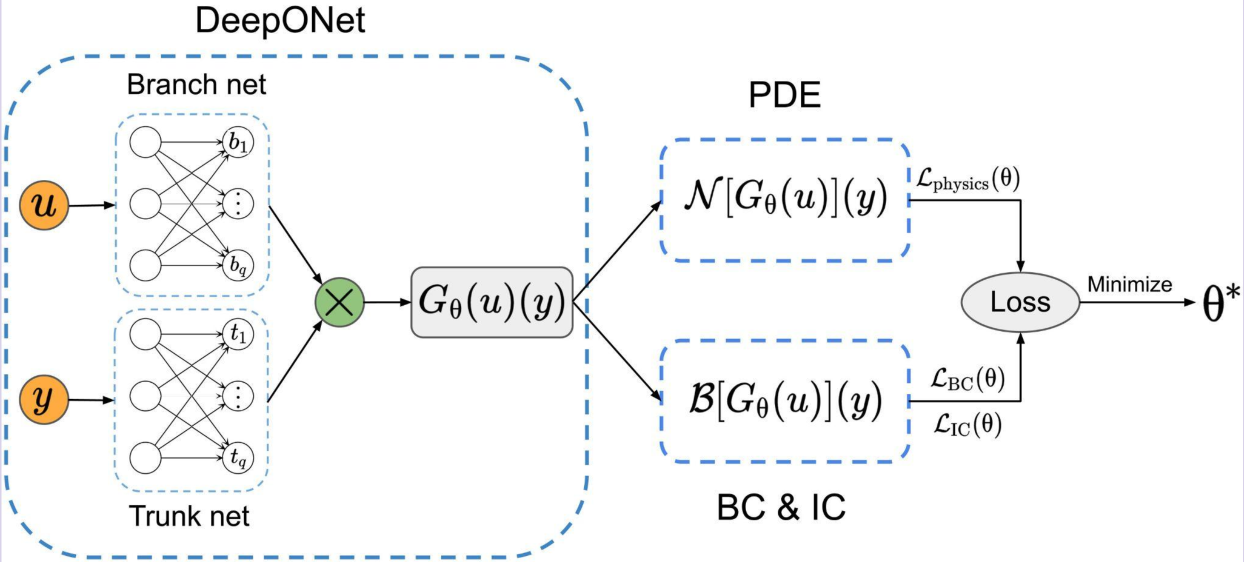
Outcome: Visualization of hidden state dynamics



(a) Recurrent Neural Network

(b) Latent Neural Ordinary Differential Equation

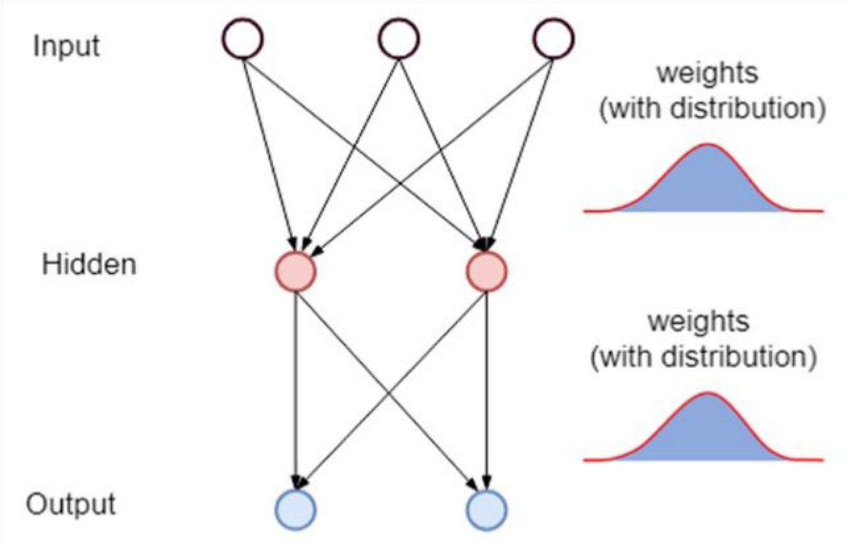
Neural PDE Solvers:



- Branch: Approximates functions related to the input.
- Trunk: Approximates functions related to the output.
- By combining the outputs, it can learn to approximate an operate that effectively solves PDEs.

Image source: S. Wang et al. 2021

Bayesian Neural Network



- Bayesian neural networks provide a probabilistic implementation of a standard neural network.
- The weights and biases are represented via posterior probability distributions rather than single point estimates

- The challenge of Bayesian inference is in sampling to approximate (learn) the posterior distribution of neural network weights and biases.
- The inference procedure begins by setting prior distributions over the weights and biases.
- Then the sampling scheme (e.g. MCMC/ Nested Sampling) employs a likelihood function that takes into account the training data accepting or rejecting a proposed sample.

Bayesian Inference (Bayes' theorem)

$$\underbrace{p(\vec{\Lambda} \mid \boldsymbol{d})}_{\text{Posterior}} = \frac{\overbrace{p(\boldsymbol{d} \mid \vec{\Lambda})}^{\text{Likelihood}}\,\underbrace{p(\vec{\Lambda})}_{\text{Prior}}}{\underbrace{p(\boldsymbol{d})}_{\text{Evidence}}}$$

Part-III: ML Test problem on Tackling incomplete metal-oxide cluster data

Duration: 10-15 minutes

Problem Statement:

- The transition from the gas to a solid phase may be described as a sequence of thermally stable clusters of sizes N = 1, ..., 150.
- The N = 1 cluster may be TiO_2, N = 2 $(TiO_2)_2$ etc.
- Only the smallest clusters N = 1...15 are known from expensive QM-Simulations, but >50 isomer structures have been derived by detailed simulations for each of these cluster sizes. The number of isomers per N varies, however, and only one isomer per size N is the most stable cluster.

The challenge is that we ideally wish to know the full chain N = 1 ... 150 of thermally stable clusters without having to conduct the expensive quantum mechanical calculations for all sizes N>15, but **instead use the knowledge we have for the N=1 ... 15 ensembles to intelligently guess (via ML methods) the data for the missing clusters.**

Density Functional Theory (DFT)

Background:

The electronic ground state of solids determines the **minimum energy equilibrium structure** and characteristics of bonding between nuclei.

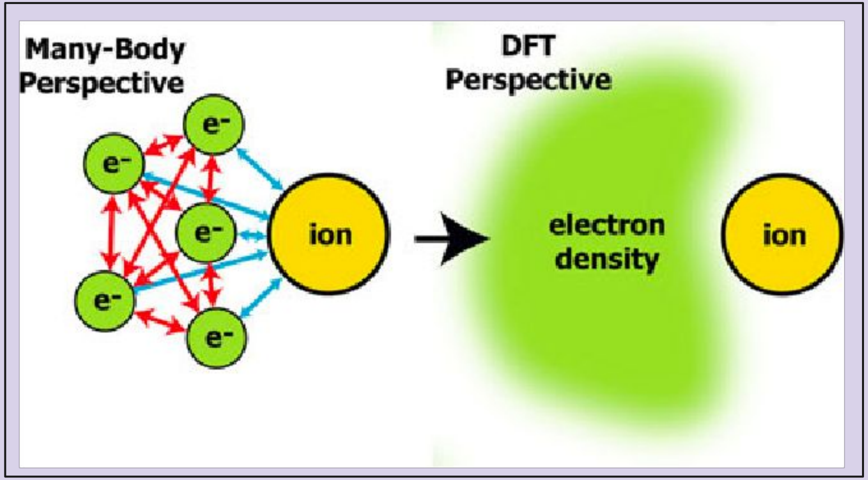
Hence, a electrons or '**electron density**' plays a huge role in determining the properties of real materials.

Finding the electronic ground state problem can be solved for a **single electron using Schrödinger's equation. However, the problem arises for many-body problems, e.g. molecules, atoms, clusters.**

It is very hectic to solve Schrödinger's equation for many body system.

DFT provides an approximate solution to the N-body Schrödinger's equation.

"DFT is quantum mechanical modeling method to investigate the electronic ground state of many body system in Physics, Chemistry and material sciences."



Source: M.T.Lusk et al. Springer, 0.1557/mrs.2011.30 (2011)

Density Functional Theory

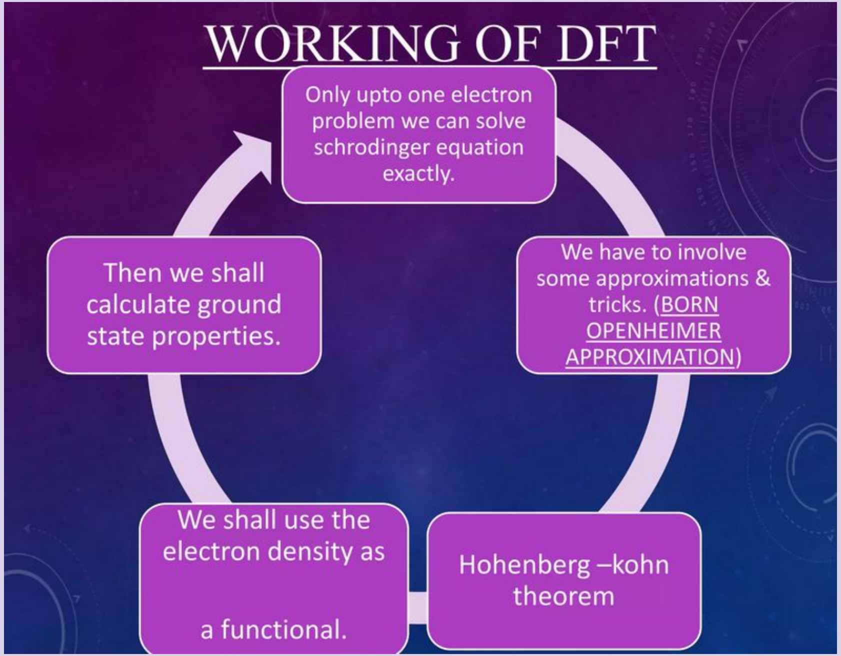
- Instead of wave-functions DFT uses functionals.

  Φ^2 = n(r): Electron density functional

- And solve: $$\hat{H}\phi = E\phi$$

- Uses approximations and theorems to solve the above equations
  - Bohn-Oppenheimer Approx (Separate the nuclei and electron dynamics)
  - Hartree-Fock Apprxn. (Convert many electron problem to many one electron problems)
  - Uses Hohenberg-Kohn Theorems
  - Uses **Kohn-Sham Scheme**

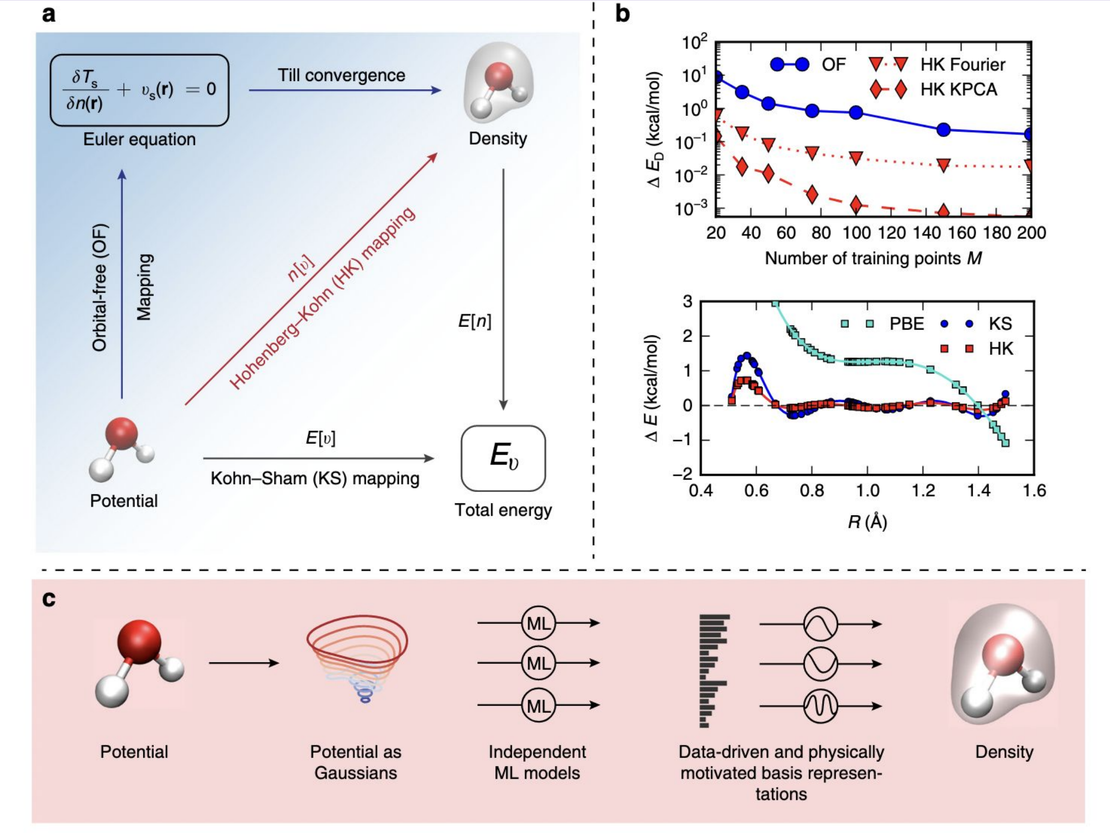
- An electron density corresponding to ground state is obtained.



Source: google image

Packages for DFT calc:
VASP, Quantum Espresso, ABINIT, SIESTA

Kohn-Sham DFT and Deep Learning



- The ground-state energy is found by solving KS equations given the external potential, v.
- E[n] is the total energy density functional.
- The red arrow is the HK map n[v] from external potential to its ground state density.
- Errors in the PBE energies (relative to exact values) and the ML maps (relative to PBE) as a function of interatomic spacing, R, for H2 with M = 7.
- The molecular geometry is represented by Gaussians; many independent Kernel ridge regression models predict each basis coefficient of the density.

Source: F.Brockherde et al. Nature communications, 2017

PINN for Time-Dependent Schrödinger Equation

TDSE: $$i\frac{\partial \phi(\mathbf{r},t)}{\partial t} - \hat{H}\phi(\mathbf{r},t) = 0$$

Source: K Shah et al. NeurIPS (2022)

Assumption: Non-interacting and spinless particles in a quantum harmonic oscillator in one spatial dimension with Hamiltonian

$$\hat{H}_x = -\frac{1}{2}\frac{\partial^2}{\partial x^2} + \frac{\omega^2}{2}x^2$$

Considering an analytical eigenstates of the harmonic oscillator $\phi(x,t) = u + iv,$

The TDSE can then be written as in terms of u and v as

$$\left(-\frac{\partial v}{\partial t} + \frac{1}{2}\frac{\partial^2 u}{\partial x^2} - \frac{\omega^2}{2}x^2\right) + i\left(\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial^2 v}{\partial x^2} - \frac{\omega^2}{2}x^2\right) = 0$$

- A PINN can be constructed with inputs (x, t, ω) and outputs (unet, vnet).
- The error of the predictions was quantified on the probability density: $|\phi(x,t)|^2 = u(x,t)^2 + v(x,t)^2$

Machine-learned interatomic potentials (MLIPs)

- Parameterized by fitting to a set of training data.
- Sometimes slower than classical interatomic potentials by an order of magnitude or more, but MLIPs are generally more accurate and are still orders of magnitude faster than ab initio calculation

Examples:

- Neural network potentials
- Gaussian approximation potentials (GAP)
- Spectral neighbor analysis potentials
- **Moment tensor potentials (MTP)**

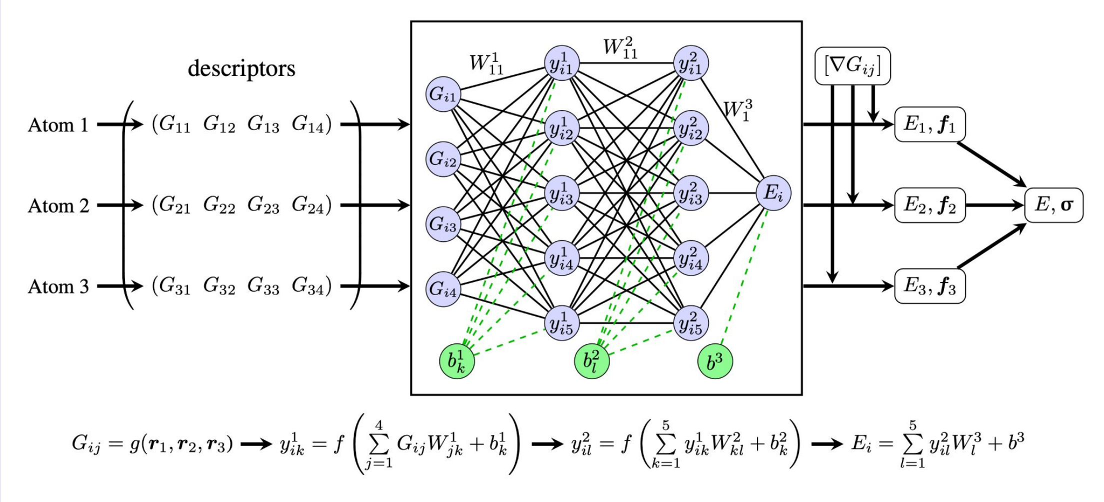
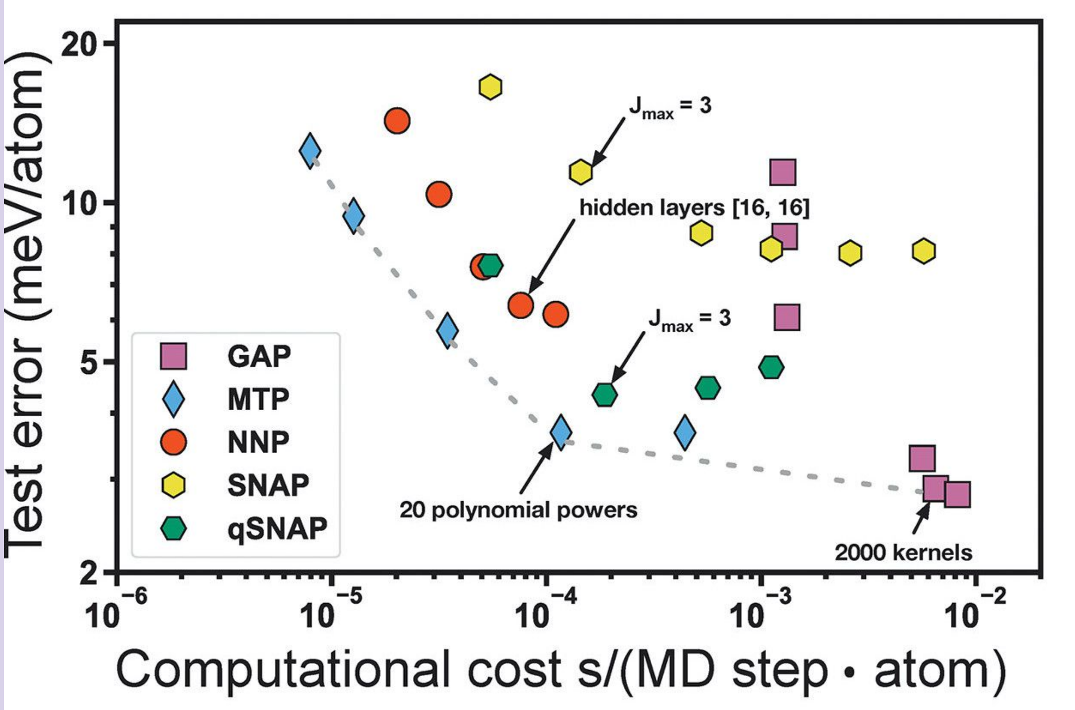


Image source: W. Gao et al. 2021

Comparison of several ML potentials



ML potentials on a range of crystal structures with a single CPU core

- The figure shows the "optimal" MTP, NNP and SNAP models tend to be 2 orders of magnitude less computationally expensive than the "optimal" GAP model, and better accuracy can only be attained at the price of greater computational cost.

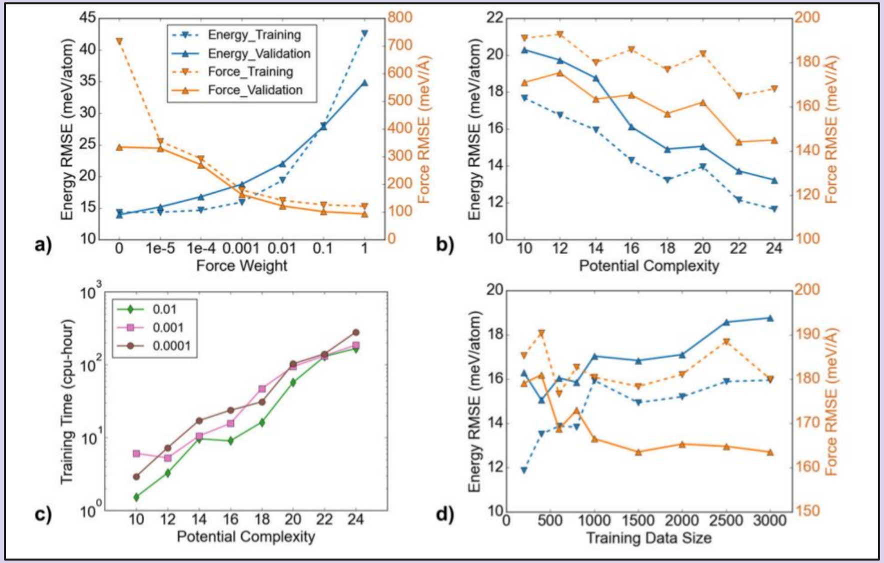
- At the end, there is no absolute winner among these methods but MTP can provide better accuracy.

Estimating MTP via ML

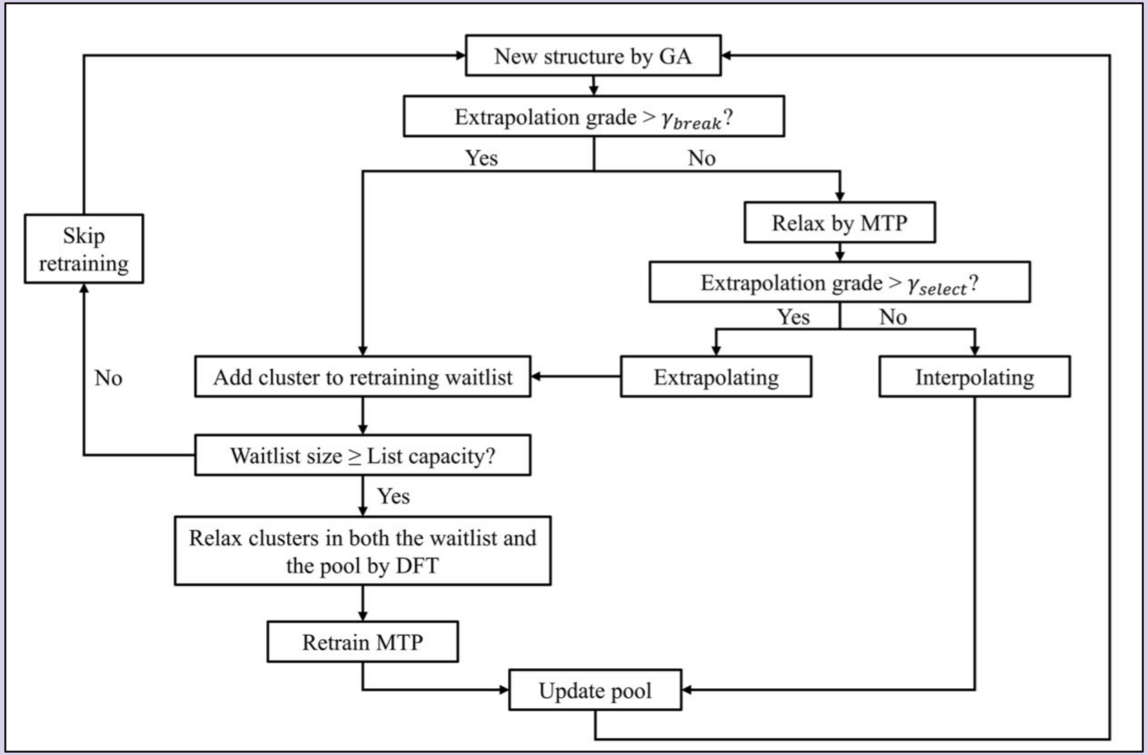
Y. Wang et al., npj computational materials, springer (2022)

- Hyperparameter selection for moment tensor potentials
- Prediction of structures for Al clusters with 21–40 atoms
- Size-transferable interatomic potentials for nanoclusters (odd/ even / mixed cluster)
- Prediction of structures for Al clusters with 41–55 atoms

  - Hyperparameters optimization for MTP
  - Hyperparameters: Force weight, Potential complexity and training data size
  - Minimizing energy and force RMSE of MTP



Genetic Algorithm with Active Learning (GA_AL):



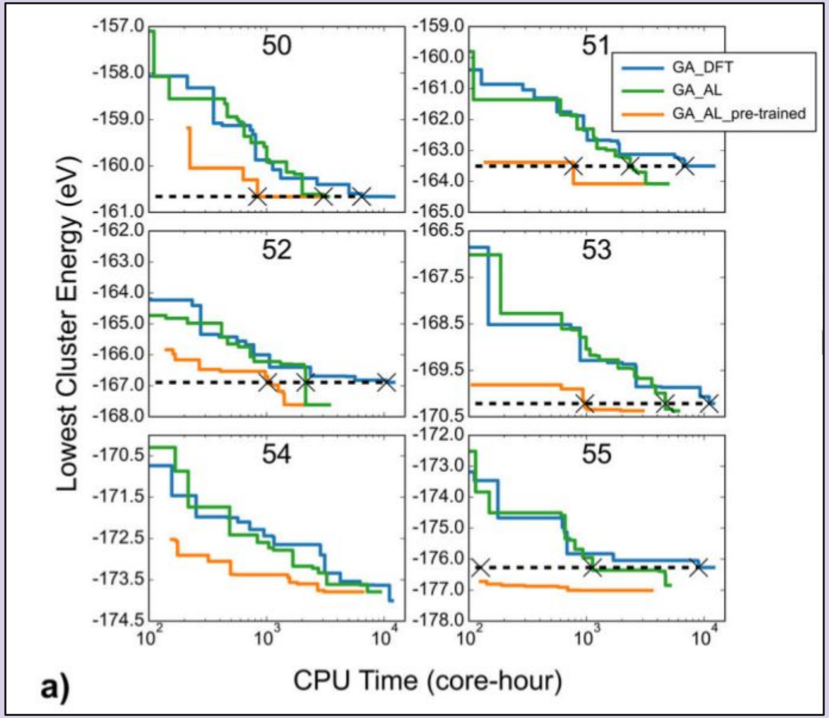
- For a given cluster of size N, initialized with MTP.
- Choose a new (random) structure.
- Compute the extrapolation grade: Similarity between new structure and existing structures in the training pools.
- MTP => Least energy structure

newly-generated cluster

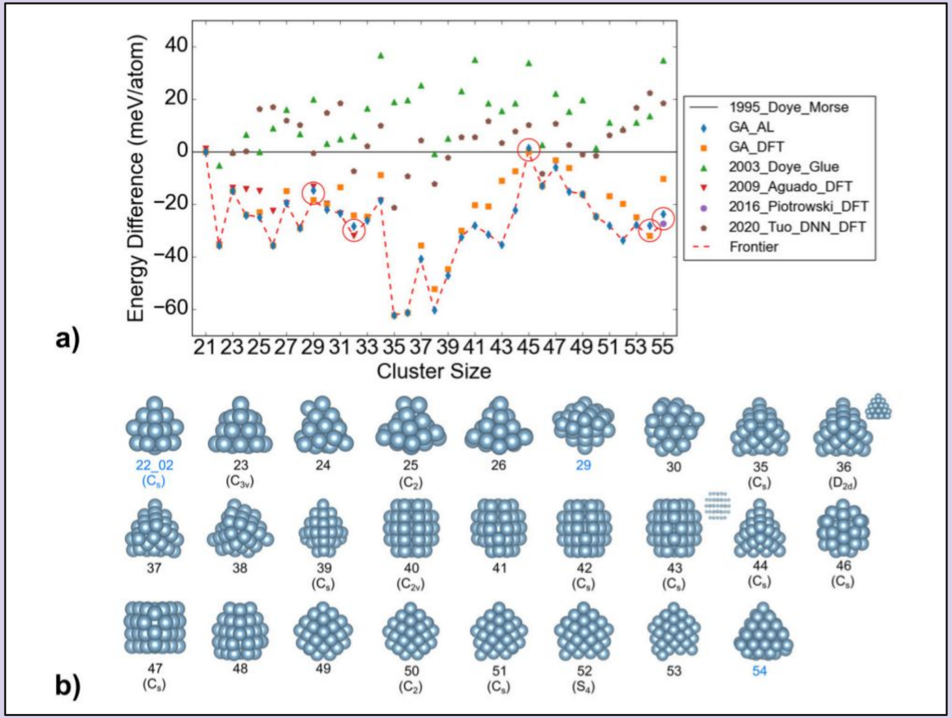
Extrapolation grade =>Retrain MTP => Update pool

Interpolation grade => Update pool

Results



50
51
52
53
54
55
GA_DFT
GA_AL
GA_AL_pre-trained
Lowest Cluster Energy (eV)
CPU Time (core-hour)
a)



Energy Difference (meV/atom)
Cluster Size
1995_Doye_Morse
GA_AL
GA_DFT
2003_Doye_Glue
2009_Aguado_DFT
2016_Piotrowski_DFT
2020_Tuo_DNN_DFT
Frontier
a)
b)
22_02 (C_s)
23 (C_{3v})
24
25 (C_2)
26
29
30
35 (C_s)
36 (D_{2d})
37
38
39 (C_s)
40 (C_{2v})
41
42 (C_s)
43 (C_s)
44 (C_s)
46 (C_s)
47 (C_s)
48
49
50 (C_2)
51 (C_s)
52 (S_4)
53
54

Thank you