Performance and Cost Assessment of Machine Learning Interatomic Potentials

Term Project Report

CH5650
Molecular Data Science and Informatics

Guide: Dr. Tarak Kumar Patra

Author: Ayesha Ulde
Roll No: MM19B021

Indian Institute of Technology Madras
Chennai - 600036

May 12, 2022

Contents

1 Introduction

A fundamental input for atomistic simulations of materials is potential energy surface (PES), which can be described as a function of atomic positions using DFT. Such quantum mechanical descriptions are accurate and transferable across chemistries but their high cost and poor scaling (typically $\mathcal{O}(n_e^3)$ or higher, where n_e is the number of electrons)[18][19][10] limits simulations to ∼1000 atoms and hundreds of picoseconds. Large and long-time simulations traditionally rely on interatomic potentials (IAPs) which to date are in most cases empirical parametrizations of the PES. The computational cost of IAPs scales linearly with the number of atoms at the cost of accuracy and transferability.
A modern alternative to IAPs are **machine-learning IAPs (MLIAPs)**[20], which express the PES as a sum of atomic energies. These atomic energies are described as a function of local environment descriptors that are invariant to translation, rotation and exchange of equivalent atoms. These descriptors are called **fingerprint functions**.[3] [1] A machine learning (ML) model is then used to map these descriptors/fingerprints to the PES. Examples of MLIAPs include the high-dimensional neural network potential (NNP)[4][2], the Gaussian approximation potential (GAP)[1][14][8], the **spectral neighbor analysis potential (SNAP)**[15][5][11][7], and moment tensor potentials (MTP)[13][12][9].
SNAP uses the **coefficients of the bispectrum** of the atomic neighbor density functions[1] as descriptors. The original formulation of SNAP uses a linear model to map the bispectrum components to the energies.[15] Recently, a quadratic model (denoted as qSNAP in [20])[17] has been developed, which extends the linear SNAP model to include all distinct pairwise products of bispectrum components. The SNAP and qSNAP models have both demonstrated great success in transition metals[15][5][11][17] as well as binary systems.[11][7][16].

2 Objective

The objective of the term project is to attempt to reproduce the SNAP results in [20] for Si (diamond group IV semiconductor), Mo (BCC metal), Ni (FCC metal) and Cu (FCC metal).

3 Methodology

The approach to reproduce SNAP results in [20] was as follows :

- maml[6][20] Installation
- Data Loading
- Feature Analysis
- Feature Space Visualization
- ML Model training
- SNAP Development
- Materials Property Predictions

3.1 maml Installation

As part of [20], high-level Python interfaces for MLIAPs development and LAMMPS material properties calculators were also developed and published in Python package **maml**. It can be pip installed via PyPI but the developers recommend the following installation via command line:

```
$ git clone https://github.com/materialsvirtuallab/maml
$ cd maml
$ pip installed -e .
```

To run the potential energy surface (pes), Lammps installation is required which can be installed from the source or from conda using the command:

```
$ conda install -c conda-forge/label/cf202003 lammps
```

The SNAP potential comes with this lammps installation.

3.2 Data Loading

Data given in the mlearn GitHub repository (maml was previously called mlearn) was used to reproduce the results in [20]. It is given in json file format, which can be read using loadfn function in the monty Python package.
It includes the number of atoms, the density functional theory (DFT) computed energy and forces, and the type of calculation (Vacancy, Surface, AIMD or Elastic as described in [20]) for each structure.

3.3 Feature Analysis

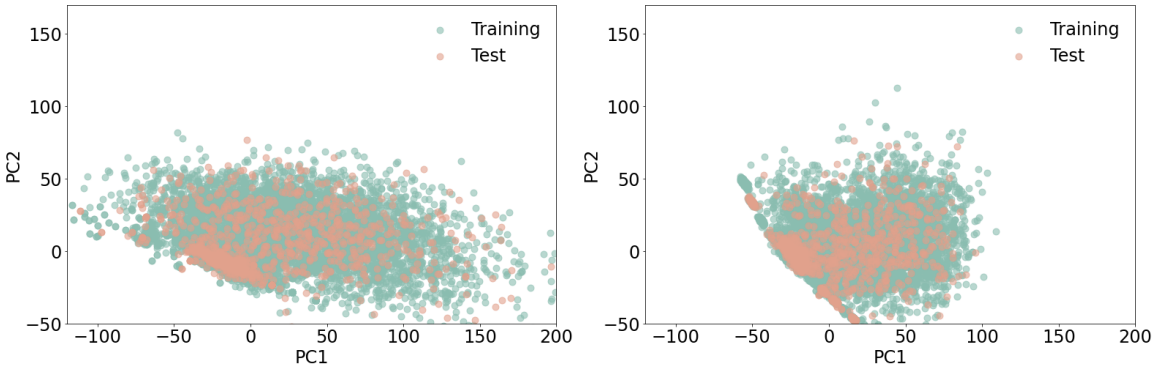
The bispectrum coefficients, used to describe the local environment of each atom, are calculated by defining descriptors by setting the hyperparameters of the BispectrumCoefficients class in maml. There are two options available to define the descriptors:

- Transform each atom in each structure to one feature array by setting pot_fit = False. The resulting describer is called per_atom_describer.
- Transform each structure to one feature for structure and one feature for each atomic force component by setting pot_fit = True. The resulting describer is called per_force_describer.

The hyperparameters given in the supplementary material with [20] were used to define the descriptors.

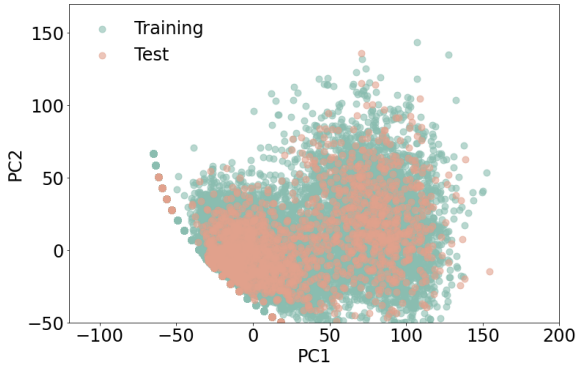
3.4 Feature Space Visualization

The feature space, that is, the bispectrum coefficients from the per_force_describer were visualized by reducing their dimensions to two using Principal Component Analysis (PCA). The bispectrum coefficients for Si, Mo, Ni and Cu were visualized in the reduced dimensions in Figure 1.

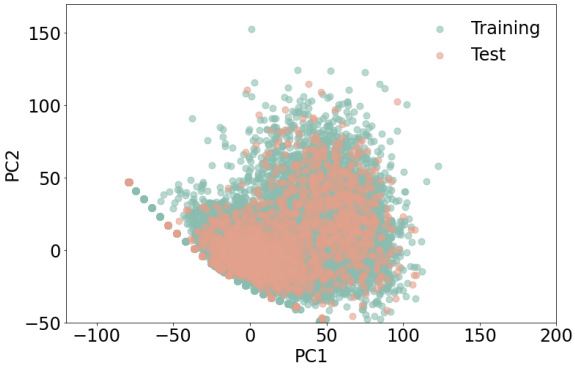


(a) Si

(b) Mo



(c) Ni



(d) Cu

Figure 1: Visualizing the bispectrum coefficients of (a) Si, (b) Mo, (c) Ni, and (d) Cu in the reduced dimensions obtained by PCA.

3.5 ML Model training

DFT computed energy and forces were used to train as targets to train linear regression models for each element using scikit-learn.
The input to the model were the bispectrum coefficients generated using the per_force_describer and the targets were the DFT computed energy and forces, which are pooled together using the pool_from function in maml and converted into a pandas dataframe using the convert_docs function in maml. This dataframe has three columns :

- d_type - Data type, force or energy
- y_orig - value of the DFT computed quantity in eV and eV/Å for energy and force respectively
- n - the number of atoms in the structure when d_type = energy, and 1 when d_type = force

A simple and a weighted regression model are fitted to the training data for each element. Since the number of forces are overwhelming, the weighted model assigns the weights 10000 and 1 for d_type = energy and d_type = force respectively. The mean absolute errors (MAEs) of energy and forces in the simple and weighted models for each element are listed below in Table 1.
From Table 1, it can be observed that :

Element		Mo	Ni	Cu	Si
Simple Model MAE	Energy meV/atom	29.171	3.117	4.552	54.924
	Forces (eV/Å)	0.176	0.031	0.026	0.129
Weighted Model MAE	Energy (meV/atom)	7.480	1.170	1.004	6.009
	Forces (eV/Å)	0.183	0.031	0.027	0.135

Table 1: MAEs of energy and forces obtained from the simple and weighted models for Mo, Ni, Cu, and Si.

- MAE of energy obtained from simple model for all elements is higher than that obtained from weighted models. However, the MAE of forces obtained from simple model for all elements is more or less same as that obtained from weighted models.
- MAE of energy and forces obtained from both simple and weighted models for Mo and Si are much higher than those obtained for Ni and Cu. This suggests that bispectrum coefficients are able to capture the local atomic environments in FCC metals much better than BCC metal and diamond cubic semiconductors.

3.6 SNAP Development

For each element, as done in section 3.5, the energies and forces of all the structures were pooled together using the pool_from function in maml and converted into a pandas dataframe using the convert_docs function in maml. Weights were assigned to the structures according to Table S8 given in the supplementary material of [20]. The bispectrum coefficients describer used was the per_force_describer used in the previous sections. The hyperparameters were set according to Table S8 in the supplementary material of [20].
Linear regression model from the scikit-learn package and the bispectrum coefficients describer were wrapped together using the SKLModel class in maml. The resulting model was then used to train the SNAP using SNAPotential class in maml on the energies, forces and structures of the training data and the weights set.
The fitted SNAP was then used to evaluate the energy and forces errors. The errors are given in Table 2 below. The parity plots of energy and forces are provided for each element in their respective jupyter notebook.

Element	Mo	Ni	Cu	Si
Energy MAE (meV/atom)	16.365	2.825	2.633	25.224
Forces MAE (eV/Å)	0.179	0.037	0.031	0.108

Table 2: MAEs of energy and forces of SNAP for Mo, Ni, Cu, and Si.

From the Table 2, it can be observed that:

- The MAE of forces is less than that of energy for all elements because the training data available for forces is more and the magnitude of forces is also smaller than that of energies.

- MAE of energy and forces of Mo and Si are significantly higher than those of Ni and Cu. This was also observed in ML model trained on DFT data in section 3.5.

3.7 Materials Property Predictions

3.7.1 Lattice Constants

Using the SNAP developed, lattice constants were predicted using the LatticeConstant class in maml. The lattice constants predicted and those obtained from DFT, as reported in [20], are given below in Table 3.

Lattice Constant (Å)	Mo	Ni	Cu	Si
Predicted	3.146 (-0.7%)	3.517 (0.3%)	3.631 (0.3%)	5.450 (-0.3%)
DFT	3.168	3.508	3.621	5.469

Table 3: DFT and predicted lattice constants of Mo, Ni, Cu, and Si. Error percentages with respect to DFT values are reported in the brackets.

Error in predicting lattice constants is small and within ±1 of the DFT value.

3.7.2 Elastic Constants

Similar to lattice constants, elastic constants were predicted using the ElasticConstant class in maml. Elastic constant prediction is computationally very expensive, and took 1 hour for Mo, 8 hours for Ni, 9 hours for Cu and 8 hours for Si on one CPU core. The elastic constants and bulk modulus predicted and those obtained from DFT, as reported in [20], are given below in Table 4.

Elastic Constant (GPa)		Mo	Ni	Cu	Si
Predicted	c_{11}	471 (-0.2%)	276 (0.0%)	145 (-19.3%)	-598
	c_{12}	200 (0.3%)	166 (4.4%)	101 (-24.0%)	-280
	c_{44}	75 (-29.2%)	117 (-11.4%)	67 (-23.8%)	-385
	B	290 (10.3%)	202 (2%)	113 (-22.6%)	-364
DFT	c_{11}	472	276	173	116
	c_{12}	158	159	133	65
	c_{44}	106	132	88	76
	B	263	198	146	95

Table 4: DFT and predicted elastic constants and bulk modulus of Mo, Ni, Cu, and Si. Error percentages with respect to DFT values are reported in the brackets.

From the table above, it can be observed that:

- Elastic constant prediction errors of Mo are relatively low, which is contrasts the high MAE of energy and forces of Mo obtained from the developed SNAP.
- Elastic constants obtained for Si are negative, which is expected given the high MAE of energy and forces of Si obtained from the developed SNAP

3.7.3 Vacancy Formation Energy

Using the SNAP developed, vacancy formation energy was predicted using the DefectFormation class in maml. The vacancy formation energy predicted and that obtained from DFT, as reported in [20], are given below in Table 5.

Vacancy Formation Energy (eV)	Mo	Ni	Cu	Si
Predicted	1.46 (-45.9%)	1.28 (-14.1%)	0.99 (-13.9%)	2.08 (-36.0%)
DFT	2.70	1.49	1.15	3.25

Table 5: DFT and predicted vacancy formation energy of Mo, Ni, Cu, and Si. Error percentages with respect to DFT values are reported in the brackets.

From the table above, it can be observed that:

- SNAP developed for all the elements underestimates the vacancy formation energy significantly.
- Vacancy formation energy errors for Mo and Si are higher than those of Ni and Cu.

4 Comparing Results Obtained With Paper

On comparing the results obtained with those published in [20], the following observations can be made:

- Using the same training and test data and hyperparameters does not give the same results.
- Prediction errors obtained for Si and Mo are much higher than those reported in the paper.

5 Concluding Remarks

Despite being computionally expensive, SNAP captures the local environment of FCC metals remarkably and can be extended to more complex systems like interfaces.

6 Acknowledgements

I would like to thank Prof. Tarak Kumar Patra for guiding me throughout this project and giving me an opportunity to work on it.

7 References

[1] Albert P Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. *Physical Review B*, 87(18):184115, 2013.

[2] Jörg Behler. Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Physical Chemistry Chemical Physics*, 13(40):17930–17955, 2011.

[3] Jörg Behler. Perspective: Machine learning potentials for atomistic simulations. *The Journal of chemical physics*, 145(17):170901, 2016.

[4] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical review letters*, 98(14):146401, 2007.

[5] Chi Chen, Zhi Deng, Richard Tran, Hanmei Tang, Iek-Heng Chu, and Shyue Ping Ong. Accurate force field for molybdenum by machine learning large materials data. *Physical Review Materials*, 1(4):043603, 2017.

[6] Chi Chen, Ye Weike Ji Qi Zuo, Yunxing, and Shyue Ping Ong. Maml - materials machine learning package. `https://github.com/materialsvirtuallab/maml`, 2020.

[7] Zhi Deng, Chi Chen, Xiang-Guo Li, and Shyue Ping Ong. An electrostatic spectral neighbor analysis potential for lithium nitride. *npj Computational Materials*, 5(1):1–8, 2019.

[8] Daniele Dragoni, Thomas D Daff, Gábor Csányi, and Nicola Marzari. Achieving dft accuracy with a machine-learning interatomic potential: Thermomechanics and defects in bcc ferromagnetic iron. *Physical Review Materials*, 2(1):013808, 2018.

[9] Konstantin Gubaev, Evgeny V Podryabinkin, Gus LW Hart, and Alexander V Shapeev. Accelerating high-throughput searches for new alloys with active learning of interatomic potentials. *Computational Materials Science*, 156:148–156, 2019.

[10] Hyunjun Ji, Yihan Shao, William A Goddard, and Yousung Jung. Analytic derivatives of quartic-scaling doubly hybrid xygj-os functional: Theory, implementation, and benchmark comparison with m06-2x and mp2 geometries for nonbonded complexes. *Journal of chemical theory and computation*, 9(4):1971–1976, 2013.

[11] Xiang-Guo Li, Chongze Hu, Chi Chen, Zhi Deng, Jian Luo, and Shyue Ping Ong. Quantum-accurate spectral neighbor analysis potential models for ni-mo binary alloys and fcc metals. *Physical Review B*, 98(9):094104, 2018.

[12] Evgeny V Podryabinkin and Alexander V Shapeev. Active learning of linearly parametrized interatomic potentials. *Computational Materials Science*, 140:171–180, 2017.

[13] Alexander V Shapeev. Moment tensor potentials: A class of systematically improvable interatomic potentials. *Multiscale Modeling & Simulation*, 14(3):1153–1173, 2016.

[14] Wojciech J Szlachta, Albert P Bartók, and Gábor Csányi. Accuracy and transferability of gaussian approximation potential models for tungsten. *Physical Review B*, 90(10):104108, 2014.

[15] Aidan P Thompson, Laura P Swiler, Christian R Trott, Stephen M Foiles, and Garritt J Tucker. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. *Journal of Computational Physics*, 285:316–330, 2015.

[16] Mitchell A Wood, Mary A Cusentino, Brian D Wirth, and Aidan P Thompson. Data-driven material models for atomistic simulation. *Physical Review B*, 99(18):184305, 2019.

[17] Mitchell A Wood and Aidan P Thompson. Extending the accuracy of the snap interatomic potential form. *The Journal of chemical physics*, 148(24):241721, 2018.

[18] Igor Ying Zhang, Xin Xu, Yousung Jung, and William A Goddard. A fast doubly hybrid density functional method close to chemical accuracy using a local opposite spin ansatz. *Proceedings of the National Academy of Sciences*, 108(50):19896–19900, 2011.

[19] Ying Zhang, Xin Xu, and William A Goddard. Doubly hybrid density functional for accurate descriptions of nonbond interactions, thermochemistry, and thermochemical kinetics. *Proceedings of the National Academy of Sciences*, 106(13):4963–4968, 2009.

[20] Yunxing Zuo, Chi Chen, Xiangguo Li, Zhi Deng, Yiming Chen, Jorg Behler, Gábor Csányi, Alexander V Shapeev, Aidan P Thompson, Mitchell A Wood, et al. Performance and cost assessment of machine learning interatomic potentials. *The Journal of Physical Chemistry A*, 124(4):731–745, 2020.