Instructions for completing the reproducible results

YC Wong

1 Introduction

This document serves as an independent guide to reproduce some numerical results presented in the report titled "Machine Learning Multiscale Simulation of Strongly Correlated Oxides." The focus is on verifying the performance and accuracy of machine learning models trained with MACE, particularly in predicting the energies and forces of various bulk Strontium Titanate (STO) structures. MACE is the essential tool utilised throughout this exercise for model training and evaluation.

The guide is designed to be rather self-sufficient, allowing readers to follow the steps without relying heavily on the main report, though references to specific figures will be made where necessary.

This guide includes two main tasks. Both tasks simply involve the calculation of the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) of the atomic **forces**. No energies evaluation are required. The first task is to reproduce the numerical results, which focusses on replicating the results of Figures 6, 7, and 8 from the main report, which detail the model's accuracy in energy and force predictions. The second task is uncertainty quantification, although it is not directly discussed in the report, this task will involve evaluating the uncertainty in the model predictions, providing additional insight into model robustness.

2 Files given in the GitHub repo

1. Pre-trained machine learning models (2x):
   - (a) MACE-opt: **STO-MACE-opt.model**
   - (b) MACE-ph: **STO-phonopy.model**
   - (c) MACE-MP: See the installation section below
2. Dataset for evaluating the performance of models:
   - (a) Test dataset used for verifying numerical result and performing uncertainty quantification in section 4: **final_test_set.xyz**
   - (b) Molecular dynamics dataset used for evaluating the performance of the model between the MACE-opt and MACE-MP models in section 5: **STO_MD_dataset.xyz**

3 Setup and Software Installations

1. Login to the SCRTP theory ohm node for GPU usage:
   `ssh username@ohm.theory.warwick.ac.uk`
2. First change to your desired working directory, and do
   `git clone https://github.com/ycwong-chess/UQ_peer_ex`

3. Installing mace through the below commands (upgrade pip first if installation not working)
```
git clone https://github.com/ACEsuit/mace.git
pip install ./mace
```

4. Now open the jupyter notebook which will provide the basic instructions. Feel free to work directly on the jupyter notebook or make a copy of it.

4 Task 1: Reproducing the exact numerical result for the three models

In this part of the exercise, you should be able to calculate the forces MAE and RMSE of the 3 models, **MACE-opt**, **MACE-ph** (phonopy model) **MACE-MP** model evaluated on the test set **final_test_set.xyz**. The test set contains bulk STO configuration with strain and AiMD data from literature. You can refer to section IV.C figure 6-8 in the main report for more information if necessary. For your convenience, the forces MAE and RMSE are given as below (all in unit eV/Å):

	MACE-opt	Phonopy	MACE-MP
MAE	0.0136	0.284	0.0457
RMSE	0.0284	0.574	0.0953

5 Task 2: Compare the uncertainties of the models against DFT using molecular dynamics Data

In this part of the exercise, you are expected to perform an uncertainty quantification on the trained ML models. This part is not for replicating any results in the main report, but rather act as as a supportive analysis.

You are given a new MD dataset (not seen in the model's training dataset) that contains 50 configurations that have been evaluated using DFT calculations. You are expected to split the MD dataset into five subsets with roughly an equal number of configurations in each set (each set should contain $10 \times 625 \times 3 = 18750$ force data points), and try to evaluate the mean and standard deviation of the forces MAE and RMSE respectively using the MACE-opt and MACE-MP model. This should give a relatively good estimate of the model's performance based on the mean and the spread, and you should be able to conclude which model is performing better.