DPA-2: a large atomic model as a multi-task learner

Duo Zhang[*1,2,3], **Xinzijian Liu**[*1,2], **Xiangyu Zhang**[4,5], **Chengqian Zhang**[2,6], **Chun Cai**[1,2], **Hangrui Bi**[1,2], **Yiming Du**[4,5], **Xuejian Qin**[7,8], **Anyang Peng**[1], **Jiameng Huang**[2,9], **Bowen Li**[10], **Yifan Shan**[7,8], **Jinzhe Zeng**[11], **Yuzhi Zhang**[2], **Siyuan Liu**[2], **Yifan Li**[12], **Junhan Chang**[2,13], **Xinyan Wang**[2], **Shuo Zhou**[2,14], **Jianchuan Liu**[15], **Xiaoshan Luo**[16,17], **Zhenyu Wang**[17,18], **Wanrun Jiang**[1], **Jing Wu**[19], **Yudi Yang**[19], **Jiyuan Yang**[19], **Manyi Yang**[20], **Fu-Qiang Gong**[21], **Linshuang Zhang**[2], **Mengchao Shi**[2], **Fu-Zhi Dai**[1], **Darrin M. York**[11], **Shi Liu**[19,22], **Tong Zhu**[10,23,24], **Zhicheng Zhong**[7,8], **Jian Lv**[17], **Jun Cheng**[21,25,26], **Weile Jia**[4], **Mohan Chen**[1,6], **Guolin Ke**[2], **Weinan E**[1,27,28], **Linfeng Zhang**[1,2,†], and **Han Wang**[6,29,‡]

[1]*AI for Science Institute, Beijing 100080, P. R. China*
[2]*DP Technology, Beijing 100080, P. R. China*
[3]*Academy for Advanced Interdisciplinary Studies, Peking University, Beijing 100871, P. R. China*
[4]*State Key Lab of Processors, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100871, P.R. China*
[5]*University of Chinese Academy of Sciences, Beijing 100871, P.R. China*
[6]*HEDPS, CAPT, College of Engineering, Peking University, Beijing 100871, P.R. China*
[7]*Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo 315201, P.R. China*
[8]*CAS Key Laboratory of Magnetic Materials and Devices and Zhejiang Province Key Laboratory of Magnetic Materials and Application Technology, Chinese Academy of Sciences, Ningbo 315201, P.R. China*
[9]*School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, P.R. China*
[10]*Shanghai Engineering Research Center of Molecular Therapeutics & New Drug Development, School of Chemistry and Molecular Engineering, East China Normal University, Shanghai 200062, P.R. China*
[11]*Laboratory for Biomolecular Simulation Research, Institute for Quantitative Biomedicine and Department of Chemistry and Chemical Biology, Rutgers University, Piscataway, New Jersey 08854, USA*
[12]*Department of Chemistry, Princeton University, Princeton, New Jersey 08540, USA*
[13]*College of Chemistry and Molecular Engineering, Peking University, Beijing 100871, P.R. China*
[14]*Yuanpei College, Peking University, Beijing 100871, P.R. China*
[15]*School of Electrical Engineering and Electronic Information, Xihua University, Chengdu, 610039, P.R. China*
[16]*State Key Laboratory of Superhard Materials, College of Physics, Jilin University, Changchun 130012, P.R. China*
[17]*Key Laboratory of Material Simulation Methods & Software of Ministry of Education, College of Physics, Jilin University, Changchun, 130012, P.R. China*
[18]*International Center of Future Science, Jilin University, Changchun, 130012, P.R. China*
[19]*Key Laboratory for Quantum Materials of Zhejiang Province, Department of Physics, School of Science, Westlake University, Hangzhou, Zhejiang 310030, P.R. China*
[20]*Atomistic Simulations, Italian Institute of Technology, 16156 Genova, Italy*
[21]*State Key Laboratory of Physical Chemistry of Solid Surface, iChEM, College of Chemistry and Chemical Engineering, Xiamen University, Xiamen, 361005, P.R. China*
[22]*Institute of Natural Sciences, Westlake Institute for Advanced Study, Hangzhou, Zhejiang 310030, P.R. China*
[23]*NYU-ECNU Center for Computational Chemistry at NYU Shanghai, Shanghai 200062, P.R. China*
[24]*Institute for Advanced algorithms research, Shanghai, 201306, P.R. China*
[25]*Laboratory of AI for Electrochemistry (AI4EC), IKKEM, Xiamen, 361005, P.R. China*
[26]*Institute of Artificial Intelligence, Xiamen University, Xiamen, 361005, P.R. China*
[27]*Center for Machine Learning Research, Peking University, Beijing 100871, P.R. China*
[28]*School of Mathematical Sciences, Peking University, Beijing, 100871, P.R. China*
[29]*Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Fenghao East Road 2, Beijing 100094, P.R. China*

[*]These authors contributed equally to this work.
[†]linfeng.zhang.zlf@gmail.com
[‡]wang_han@iapcm.ac.cn

Preprint. Under review.

Abstract

The rapid advancements in artificial intelligence (AI) are catalyzing transformative changes in atomic modeling, simulation, and design. AI-driven potential energy models have demonstrated the capability to conduct large-scale, long-duration simulations with the accuracy of *ab initio* electronic structure methods. However, the model generation process remains a bottleneck for large-scale applications. We propose a shift towards a model-centric ecosystem, wherein a large atomic model (LAM), pre-trained across multiple disciplines, can be efficiently fine-tuned and distilled for various downstream tasks, thereby establishing a new framework for molecular modeling. In this study, we introduce the DPA-2 architecture as a prototype for LAMs. Pre-trained on a diverse array of chemical and materials systems using a multi-task approach, DPA-2 demonstrates superior generalization capabilities across multiple downstream tasks compared to the traditional single-task pre-training and fine-tuning methodologies. Our approach sets the stage for the development and broad application of LAMs in molecular and materials simulation research.

1 Introduction

An accurate interatomic potential energy surface (PES) is crucial for molecular modeling and simulations. Quantum mechanical (QM) methods, such as density functional theory (DFT) [1, 2], provide satisfactory accuracy in most applications. However, their computational complexity typically scales as the cubic order of the system size, thus limiting large-scale simulations. In contrast, empirical force fields (EFF) are way more efficient, but their accuracy is often deemed insufficient for various applications. Machine learning potentials (MLPs) have emerged as a powerful approach to modeling complex materials and molecules, bridging the gap between the high accuracy of QM methods and the computational efficiency of EFFs. This has enabled the study of large-scale molecular systems with QM-level accuracy across diverse applications, including drug discovery [3, 4], materials design [5–7], and catalysis [8, 9], etc.

In most MLP applications, the training data is generated from scratch either through brute force *ab initio* molecular dynamics simulations [10] or by using a concurrent learning (or active learning) scheme capable of automatically generating the most critical data for building uniformly accurate models [11–14]. In any case, DFT-calculated energies and forces are required for each configuration in the training dataset, resulting in a substantial amount of efforts spent on constructing DFT-labeled datasets. For instance, in the AlMgCu general-purpose ternary alloy MLP [15], more than 10 million CPU hours were spent on labeling the 141K training data points. Furthermore, MLPs often struggle to generalize to applications not covered by the training data [5], such as when additional elements are included in materials design or when crystal structures in a broader range of thermodynamic conditions need to be explored.

To further extend the application range of MLPs, efforts have been made to develop "universal" or "fundamental" models [16–21], referred to as large atomic models (LAMs), based on extensive density functional theory (DFT)-labeled datasets. However, the technical approach still requires further exploration, and a LAM-centric ecosystem remains to be established. The primary factors influencing this exploration process are the methods employed for model training and their subsequent application in various tasks.

During the model training stage, a single-task-based training strategy, i.e., training using consistently labeled data, remains dominant. Models generated in this way are typically expected to be directly applicable to downstream tasks in which the explored configurations are effectively covered by the training data. Some examples include models such as M3GNet [17], CHGNet [19] and MACE-MP-0 [20], which are all trained on snapshots from DFT relaxations of the Material Project [22] structures, with M3GNet utilizing 88K configurations across 89 chemical species and both CHGNet and MACE-MP-0 being trained on 1.58M inorganic crystal frames from the concurrently introduced MPtrj dataset [19]; GNoME [21], trained on a dataset of inorganic crystals also starting from MP, but nearly two orders of magnitude larger than MPtrj; PreFerred Potential (PFP), trained on approximately 9M frames of 45 elements [16]; and ALIGNN, trained on 307K data frames of 89 elements [18].

Several limitations exist in the single-task training strategy: (1) Simultaneously training multiple datasets from different application fields is not feasible due to the variations in labeling with different DFT settings. For instance, the MPtrj dataset, labeled by DFT calculations using PBE/PBE+U [23] exchange-correlation functional and plane-wave basis, cannot be concurrently trained with the ANI-1x dataset, labeled by DFT calculations using the ωB97x hybrid functional [24] and an atomic basis set, thus little possibility is left to improve the model's generalizability on molecular applications. (2) The requirements of downstream tasks might be difficult to satisfy. For instance, a task may require DFT accuracy at the meta-general gradient approximation (meta-GGA) level. A model trained with GGA-level DFT data would not be easily adapted to fulfill this requirement.

Multi-task pre-training, combined with various strategies for downstream tasks such as fine-tuning and distillation, has emerged as a promising alternative for the development of LAMs [25–28]. By employing the multi-task training strategy [29, 30], it becomes possible to jointly pre-train models using multiple datasets labeled with different DFT settings [27, 31]. During fine-tuning for downstream tasks, the model's backbone, which encodes the representation of configurational and chemical spaces, is preserved and connected to one or multiple task heads [32, 33]. As a result, the labeling methods for pre-training and fine-tuning datasets do not need to be identical. Furthermore, the downstream tasks can involve property predictions rather than PES modeling [31]. This scheme offers significant flexibility in downstream tasks and may lead to a much better generalization ability of a LAM.

Before proceeding further, let us list the requirements of a LAM that we consider to be fundamental: (1) highly generalizable, (2) extensive and respect the translational, rotational, and permutational symmetries, (3) conservative, and (4) continuous up to second-order derivatives. A model with high generalizability implies that when trained with the same amount of data, the model can achieve high accuracy [34]. The generalizability is critical in pre-training LAMs, considering that the DFT-labeled data are expensive and sparse in the configurational and chemical spaces. By conservative, we mean that the forces (and virial tensor, for periodic systems) are calculated by the derivatives of the model-predicted total energy of the system concerning atom coordinates (and cell tensor, respectively). The conservativeness and smoothness of the model are critical for energy conservation in MD simulations and are thus a compulsory requirement for calculating dynamic properties such as diffusion coefficient, viscosity, and thermal conductivity [35]. The requirements (1)–(4) are physical restraints imposed on a PES, thus they are necessary (but in general not sufficient) conditions for the generalizability of the LAMs.

In this context, the primary contribution of this work is the development of DPA-2, a multi-task pre-trained model that meets all the mentioned requirements and furnishes a representation suitable for a diverse array of multi-disciplinary applications, including alloys, semiconductors, battery materials, drug molecules, and more, while exhibiting a high degree of generalization for downstream tasks. The revelation of a remarkable correspondence between the learned representations by DPA-2 and existing chemical knowledge underscores the potential of the proposed model architecture and the multi-task training scheme. Furthermore, we emphasize the importance of an open and application-oriented model evaluation system for the molecular simulation community in the era of large atomic models.

1.1 Related work

Machine learning potential models In recent years, there has been rapid development in MLP models. While it is nearly impossible to provide a comprehensive list, some notable examples include the Behler-Parrinello neural network (BPNN) [36], ANI [37], deep tensor neural networks (DTNN) [38], weighted atom-centered symmetry functions (wACSF) [39], Deep Potential (DP) [40–42], Deep Potential with attention (DPA-1) [43], and embedded atom neural network (EANN) [44]. These models employ either hand-crafted or machine-learned descriptors of atomic environments, along with deep neural networks, to approximate potential energy. Other machine learning techniques, such as kernel ridge regression, are also widely used. Examples include the Gaussian approximation potential (GAP) [45], which uses a smooth overlap of atomic positions (SOAP) measure of distance between local environments [46], the Coulomb matrix [47], and gradient-domain machine learning (GDML) [48]. Some potential energy models, such as the spectral neighbor analysis method (SNAP) [49] and the moment tensor potential (MTP) [50], utilize linear regression for fitting the potential energy surface (PES).

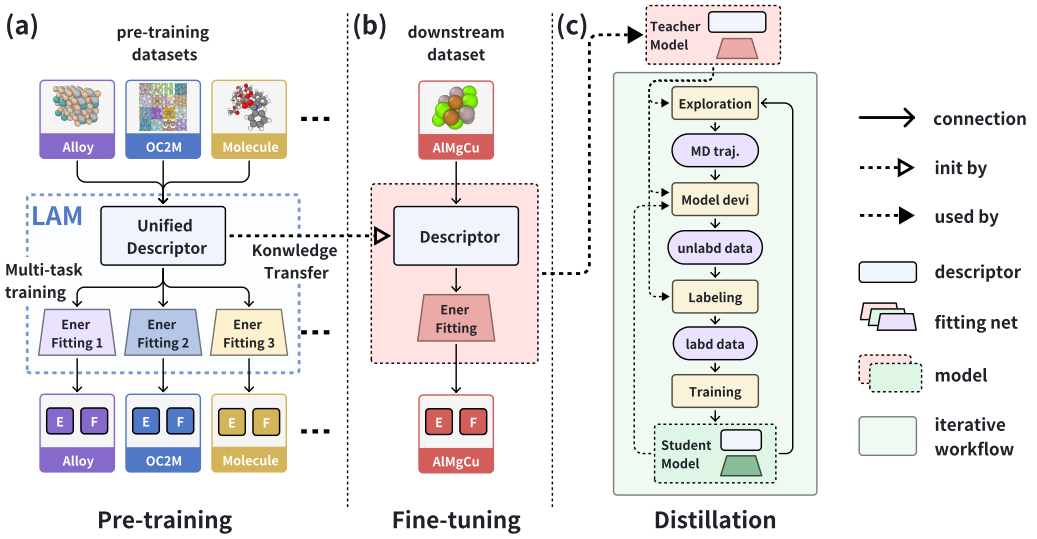

Figure 1: An overview of the proposed LAM workflow, (a) the multi-task pre-training process, in which different DFT-labeled data can be pre-trained together by sharing a single descriptor and having their unique fitting nets, with sampling according to their importance. This results in a unified descriptor. (b) The fine-tuning process on the downstream dataset, using the pre-trained unified descriptor and selecting a fitting net from upstream tasks or reinitializing the fitting net for the downstream dataset. (c) The distillation process uses the fine-tuned model as a teacher model, iteratively performing MD simulations and adding labeled data to the training set to train a high-efficiency student model, which is convenient for downstream applications.

Recently, there has been a surge in the development of equivariant graph neural networks (GNN) [51, 52], with examples including SchNet [53], Directional Message Passing Neural Network (DimeNet) [54], Polarizable Atom Interaction Neural Network (PaiNN) [55], Geometric Message Passing Neural Network (GemNet) [56], SpinConv [57], Spherical Channel Network (SCN) [58], Neural Equivariant Interatomic Potentials (NequIP) [59], MACE [60] and Equiformer/EquiformerV2 [61, 62]. These networks are based on message passing among node and edge equivariant representations and have demonstrated promising fitting accuracy. However, it has been noted that GNNs are not easily parallelizable, making them less ideal for large-scale molecular dynamics (MD) simulations [63].

Pre-trained models for molecular modeling Pre-training, or representation learning [64, 65], has shown significant success across various applications, including natural language processing [30, 66] and computer vision [67]. In the realm of molecular modeling, a primary objective of pre-trained models is to learn atomic representations of chemical species and 3D configurations of atoms.

One category of downstream tasks involves property prediction. Pre-trained models can be trained in an unsupervised manner by recovering masked atomic types and perturbed coordinates [68–72], by undertaking generative tasks [69], or by engaging in supervised learning tasks such as regression and classification [73–75, 31].

Another category of downstream tasks focuses on the modeling of PESs. The model can be pre-trained through unsupervised tasks like denoising or chemical species restoration [28, 25], supervised learning of energy, force, or partial charge [76, 27], or a combination of both types of tasks [26]. Interestingly, most of these methods were developed for pre-training on molecule-in-vacuum systems, thus limiting the downstream tasks to such a class of tasks. Ref. [76] developed pre-trained models for condensed-phase carbon systems, but these models are unlikely to be generalizable to systems composed of chemical elements other than carbon. Zhang et al. [43] pre-trained the DPA-1 model on the OC2M dataset [77] and examined its performance on downstream tasks involving high entropy alloys and AlMgCu ternary alloys. However, the study did not investigate downstream tasks related to non-metallic systems.

2 Results

2.1 The workflow of LAM

The LAM workflow includes the phases of *pre-training*, *fine-tuning* for downstream tasks, and *knowledge distillation*, as schematically presented in Fig. 1. The LAM is constructed with a unified descriptor that encodes the symmetry-preserving representation of the chemical and configurational spaces of atomic systems. This descriptor is connected to the energy-fitting networks, each predicting the energy (E) and force (F) outputs based on the data used during the pre-training phase (see Fig. 1(a)).

The LAM employs a multi-task training strategy, as illustrated in Figure 1(a). Specifically, the network parameters within the unified descriptor are concurrently optimized through back-propagation using all pre-training datasets. In contrast, the parameters of the fitting network are updated exclusively with the specific pre-training dataset to which they are associated. This approach is fundamentally different from the single-task training paradigm, where all model parameters, encompassing those within both the descriptor and the fitting network, are refined using a singular training dataset. The inability to merge the pre-training datasets into a unified "super-dataset" stems from the fact that labels across different datasets are typically derived from DFT calculations subject to variable conditions, such as exchange-correlation functionals, basis sets, and energy cut-off radii, culminating in distinct PESs. We have shown that the multi-task training is as efficient as the single-task training scheme, see Sec. S3 of the Supplementary Materials. Therefore, the multi-task training delivers the possibility of training the atomic representation from the heterogeneously labeled pre-training datasets. It is noted that although a hybrid multi-task pre-training approach using both labeled and unlabeled data is technically feasible, we focus on supervised learning for pre-training in this work, and leave the investigation of hybrid multi-task pre-training in future studies.

The pre-trained descriptor and the fitting networks can be fine-tuned for specific downstream PES modeling tasks, as illustrated in Figure 1(b). In the downstream model, the descriptor is initialized with the pre-trained unified descriptor, while the fitting network may be initialized either randomly or with a fitting head akin to the one used in one of the pre-training tasks. Given that the pre-training dataset encodes the bulk of the information within the descriptor, the initialization method for the downstream fitting network is likely to be of minor importance. The training dataset for a downstream task might be pre-existing and ready for training, or it could be generated through concurrent learning schemes such as DP-GEN [14]. In this study, we present several ready-to-use downstream datasets to validate the effectiveness of our proposed methodology and defer the exploration of concurrent learning-based data generation to future research.

The fine-tuned model, while possessing a large number of parameters, may exhibit reduced efficiency when directly applied to applications like molecular dynamics (MD) simulations. To address this concern, we propose model distillation to create a streamlined version that retains the desired accuracy for downstream tasks while also enhancing processing speed and facilitating extensive simulations. Figure 1(c) depicts the distillation procedure, which employs an iterative learning loop. Within this framework, the original model, henceforth referred to as the "teacher", labels the data. In parallel, a "student" model, characterized by a simplified architecture (e.g. DPA-1 without any attention layer, which can be further compressed [78] to significantly enhance performance), is trained on this labeled data. The teacher model is then engaged in MD exploration, operating under conditions akin to those of the intended downstream application. This ensures that the chemical and physical parameters encountered during both the distillation process and the actual tasks are consistent, facilitating effective learning by the student model. Configurations from the MD trajectories are sampled, and the student model's predictions are compared against those of the teacher. If the discrepancy between their predictions surpasses a pre-established threshold, these configurations are appended to the training set for subsequent iterations. The cycle is reiterated until the student model's predictive accuracy either meets the preset standards or stabilizes without further improvement.

2.2 Datasets and DPA-2 descriptor

The primary goal in developing LAMs is to embed comprehensive knowledge within the multi-task pre-trained model by leveraging the pre-training dataset. Consequently, this embedded knowledge is anticipated to alleviate the intensive fine-tuning process required for specific downstream tasks. This objective necessitates two essential criteria during the pre-training phase: (1) the pre-training dataset

Table 1: Overview of pre-training and downstream datasets employed in the multi-task learning framework. The columns provide dataset name, coverage of the chemical space, number of training data points, number of test data points, the total data count, and assigned weight.

Pre-training datasets					
Name	element	#train	#test	#total	weight
Alloy	53	71,482	1,240	72,722	2.0
Cathode-P	Li,Na,O,Mn,Fe,Co,Cr,Ni	58,690	6,451	65,141	1.0
Cluster-P	Pd,Ru,Al,Au,Ag,Pt,Si,Cu,Ni	139,200	14,936	154,136	1.0
Drug	H, C, N, O, F, Cl, S, P	1,379,956	24,257	1,404,213	2.0
FerroEle-P	15	6,966	760	7,726	1.0
OC2M	56	2,000,000	999,866	2,999,866	2.0
SSE-PBE-P	Li, P, S, Si, Ge	15,019	755	15,774	1.0
SemiCond-P	14	136,867	14,848	151,715	1.0
H2O-PD	H, O	46,077	2,342	48,419	1.0
Ag∪Au-PBE	Ag, Au	16,696	812	17,508	0.2
Al∪Mg∪Cu	Al, Mg, Cu	24,252	1,145	25,397	0.3
Cu	Cu	14,596	770	15,366	0.1
Sn	Sn	6,449	276	6,725	0.1
Ti	Ti	10,054	474	10,528	0.1
V	V	14,935	738	15,673	0.1
W	W	42,297	2,100	44,397	0.1
C12H26	H, C	33,898	1,598	35,496	0.1
HfO2	O, Hf	27,660	917	28,577	0.1
sum	73	4,045,094	1,074,285	5,119,379	13.2
Downstream datasets					
Name	element	#train	#test	#total	weight
Cathode-D	Li, Na, O, Mn, Fe, Co, Cr	30,002	3,244	33,246	1.0
Cluster-D	Pd, Au, Ag, Pt, Cu, Ni	4,218	395	4,613	1.0
FerroEle-D	15	7,521	597	8,118	1.0
SSE-PBE-D	Li, P, S, Sn	2,563	131	2,694	0.5
SSE-PBESol	Li, P, S, Si, Ge, Sn	7,502	384	7,886	0.5
SemiCond-D	P, N, Al, Te, In, Se, Sb, B, As	78,614	8,495	87,109	1.0
ANI-1x	H, C, N, O	4,872,049	83,956	4,956,005	1.0
Transition-1x	H, C, N, O	7,632,328	967,454	8,599,782	1.0
H2O-DPLR	H, O	557	46	603	0.5
H2O-SCAN0	H, O	7,002	347	7,349	0.5
H2O-PBE0TS	H, O	133,000	7,000	140,000	0.5
H2O-PBE0TS-MD	H, O	38,000	2,000	40,000	0.5
AgAu-PBED3	Ag, Au	64,239	2,256	66,495	0.3
AlMgCu-D	Al, Mg, Cu	113,942	2,820	116,762	0.2
In2Se3	In, Se	11,621	568	12,189	0.2
sum	39	13,003,158	1,079,693	14,082,851	9.0

must encompass a broad spectrum of chemical and configurational spaces to prepare the model for potential scenarios in downstream applications; and (2) the DPA-2 model, pre-trained in a multi-task manner, is expected to exhibit a strong ability to generalize to downstream tasks, provided that the chemical and configurational space relevant to these tasks overlaps to some extent with the scope of the datasets used during pre-training.

For the first criterion, the datasets utilized in this study are summarized in Table 1. Detailed descriptions are provided in Section S1 of the Supplementary Materials. Some datasets are newly generated in this work, including metallic alloys (Alloy), cathode materials (Cathode), metal nano-clusters (Cluster), and drug-like molecules (Drug). Some datasets are contributed by the DeepModeling community [4], including the ferroelectric perovskite (FerroEle), solid-state-electrolyte (SSE), semiconductors (SemiCond), H_2O, metallic material datasets (e.g. Sn, AgAu and AlMgCu), and the pyrolysis of n-dodecane (C12H26). Additionally, we have the open catalyst 20 [77] (OC2M) that is formed by AIMD trajectories of molecular chemical reactions catalyzed by metallic substrates. These datasets are labeled with various DFT software like the VASP [79, 80], Gaussian [81], and ABACUS [82, 83]. In addition, They are divided into two groups, the pre-training and the downstream datasets, as

[4] See `https://github.com/deepmodeling/AIS-Square/tree/main/datasets`

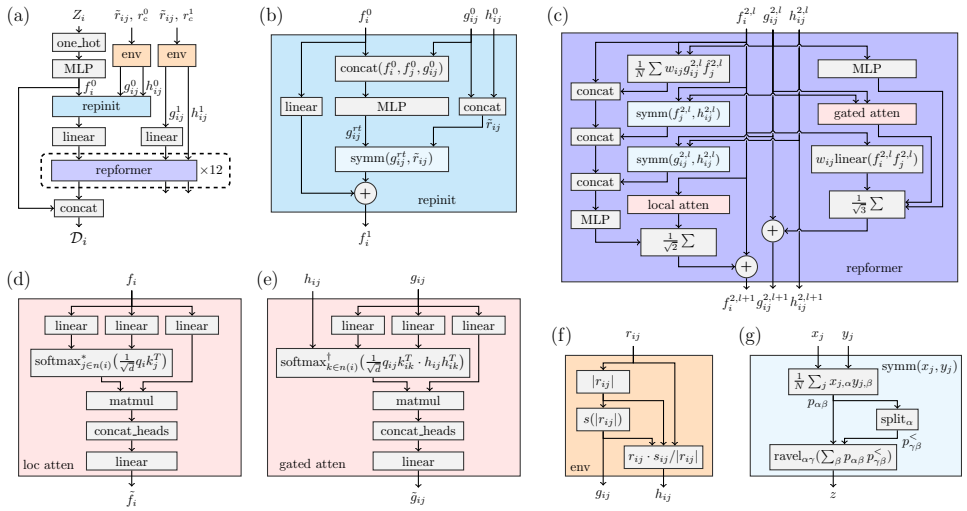

Figure 2: (a) Detailed architecture of the DPA-2 descriptor, which includes two primary components: repinit and repformer. (b) Structure of repinit. (c) Structure of repformer. (d-g) Substructures referenced in subsequent sections.

detailed in Sec. S1 of the Supplementary Materials. It is noted that the division is only to demonstrate the effectiveness of the workflow of LAM. For production purposes, all the datasets listed in Tab. 1 should be used to pre-train a LAM.

In the last column of Tab. 1, weights are assigned to each pre-training dataset. These weights are based on relevance, diversity in both chemical and configurational spaces, and data volume. The weight of a dataset is proportional to its selection probability during multi-task training, meaning that datasets with higher weights are favored in each training iteration. These weights also play a crucial role in calculating the weighted average of errors across all datasets, as shown in Tab. 2 and Tabs. S2–S3 of the Supplementary Materials, which helps to provide an assessment of the model's overall accuracy.

For the second criterion, we propose the DPA-2 model with full details of the model architecture explained in Section 4. The descriptor of the model, which is supposed to encode the representation of the chemical and configurational spaces of the pre-training dataset, is schematically demonstrated in Fig. 2. The chemical and configurational spaces are represented by a single-atom channel f_i, a rotationally invariant pair-atom channel g_{ij} and a rotationally equivariant pair-atom channel h_{ij}. The pair-atom representations are initialized by the environment matrix (operator env in Fig. 2), which encodes the relative positions of the near neighbors within a certain cut-off radius (r_c^0 and r_c^1), and smoothly decays to zero at the cut-off radius. The single-atom representations f_i is initialized by a repinit (representation initializer) layer. Then the single- and pair-atom representations are subsequently updated by the representation transformer (repformer) layers, which are stacked 12 times and communicate information in a message-passing manner between the layers. In each of the repformer layer, f_i is updated by convolution, symmetrization, MLP, and localized self-attention operators, while g_{ij} is updated by MLP, dot-product, and gated self-attention operators (see Fig. 2(c) and Sec. 4.2.3 for more details). The contribution of different building blocks to the model accuracy is investigated by an ablation study in Sec. S7 of the Supplementary Materials.

The DPA-2 model is designed to be extensible and inherently respects translational, rotational, and permutational symmetries. Moreover, it is conservative, as it predicts atomic forces by computing the negative gradient of the system's energy with respect to the atomic positions, $F_i = -\nabla_{r_i} E$, and calculates the virial tensor as $\Xi_{\alpha\beta} = \sum_\gamma (-\nabla_{h_{\gamma\alpha}} E) h_{\gamma\beta}$, where E represents the energy, r_i denotes the position of atom i, and $h_{\alpha\beta}$ is the βth component of the αth basis vector of the simulation cell. Furthermore, all components of the DPA-2 model are continuous up to the second-order derivative, ensuring energy conservation. Numerical examples demonstrating the energy conservation properties of the DPA-2 model can be found in Supplementary Material Sec. S8.

Table 2: Comparison on the zero-shot generalization errors on downstream tasks. The MACE-MP-0 (MACE) and DPA-2 pre-trained on MPtrj dataset, the DPA-2 pre-trained by single-task (ST) and multi-task (MT) approaches are compared. The DPA-2 ST is trained by the pre-training datasets listed in the second column of the Table, while the DPA-2 MT is trained by all the pre-training datasets listed in Tab. 1. The energy and force RMSEs on the downstream test datasets are reported. The weighted averaged RMSEs (WARMSE) with the weights presented in Tab. 1 is given in the first row of the table. The standard deviations of energy and force labels in the test set are also provided. If the RMSE is smaller than the corresponding standard deviation, the model shows the ability of zero-shot generalization, on the other hand, the model cannot be generalized to downstream tasks without downstream data.

Downstream	Pre-train (only for ST)	Energy RMSE [meV/atom] ↓					Force RMSE [meV/Å] ↓				
		data std.	MACE (MPtrj)	DPA-2 (MPtrj)	DPA-2 ST	DPA-2 MT	data std.	MACE (MPtrj)	DPA-2 (MPtrj)	DPA-2 ST	DPA-2 MT
WARMSE		**121.4**	**104.0**	**68.3**	**100.2**	**50.1**	**1405.4**	**575.6**	**516.6**	**628.0**	**238.8**
AgAu-PBED3	AgAu-PBE	906.9	1812.8	268.9	222.9	192.3	878.0	683.2	293.3	236.9	63.6
AlMgCu-D	AlMgCu	383.8	33.8	32.0	254.3	41.2	1229.5	240.1	245.3	663.7	111.8
AlMgCu-D	Alloy	383.8	33.8	32.0	74.9	48.4	1229.5	240.1	245.3	122.3	112.8
ANI-1x	Drug	198.9	52.3	61.7	67.2	56.6	2124.6	636.1	700.1	738.7	346.7
Cathode-D	Cathode-P	42.2	15.8	29.7	39.8	43.8	641.9	288.4	613.9	339.7	273.9
Cluster-D	Cluster-P	636.0	323.7	262.7	41.4	40.5	3605.4	2230.8	1193.6	238.4	190.5
FerroEle-D	FerroEle-P	43.0	12.5	14.5	6.3	3.9	881.3	191.3	194.2	282.7	115.1
H2O-DPLR	H2O-PD	15.6	2.1	2.0	9.1	9.3	825.2	94.4	99.7	263.5	263.4
H2O-H2O	H2O-PD	47.0	4.9	7.2	4.9	4.7	1941.0	381.0	382.7	58.8	64.4
H2O-PBE0TS-MD	H2O-PD	3.3	1.1	1.5	0.5	0.6	816.1	330.8	314.4	37.6	40.8
H2O-SCAN0	H2O-PD	12.6	3.2	3.8	1.1	0.7	2163.2	387.5	385.2	409.2	162.9
In2Se3	SemiCond-P	120.5	31.9	24.5	160.6	38.9	611.1	190.2	188.0	1544.1	341.6
SemiCond-D	SemiCond-P	587.6	49.8	70.9	486.2	175.7	1755.4	470.7	534.9	1439.4	439.3
SSE-PBE-D	SSE-PBE-P	79.0	33.7	39.4	40.7	6.2	789.5	222.1	249.9	635.6	162.4
SSE-PBESol	SSE-PBE-P	84.3	32.5	37.4	26.1	8.3	810.9	231.8	260.4	425.0	115.3
Transition-1x	Drug	139.8	56.4	55.1	48.2	45.8	368.1	518.6	618.3	1298.6	363.8

2.3 Generalizability of the multi-task pre-trained DPA-2 model

Before moving to a discussion on the generalizability of the multi-task training scheme, we test the model of DPA-2 by using single-task benchmarks, which are directly comparable to the state-of-the-art model architectures. In the first benchmark, the ANI-1x dataset, the DPA-2 shows superior test accuracy compared with the ANI-1x model reported in Ref. [11], see Tab. S1 in the Supplementary Materials. In the second benchmark, the accuracy of the DPA-2 model is comparable to GemNet-OC [84] and higher than Equiformer V2 [62], NequIP [59], Allegro [63] and MACE [60] models on the pre-training datasets, see Tab. S2 in the Supplementary Materials.

Next, we train the DPA-2 model on all the pre-training datasets by the multi-task scheme. The details of the training protocol, the test accuracy of these datasets, and a discussion on the effectiveness of the multi-task scheme are given in Sec. S3 of the Supplementary Materials.

We investigate the generalizability of the multi-task pre-trained DPA-2 model to downstream tasks by testing the model directly on downstream datasets. This approach is known as zero-shot generalization because no data from the downstream tasks are used to refine the pre-trained model before testing. In an ideal scenario, a perfectly generalizable model—that is, one that encapsulates the chemical knowledge of the periodic table and all relevant configurations for a given downstream task—would exhibit a zero-shot generalization error comparable to, or potentially lower than, the test error of a model specifically trained from scratch for that task.

The zero-shot generalizability of the multi-task pre-trained DPA-2 model is presented in Table 2 and compared with its single-task pre-trained counterpart, MPtrj-trained DPA-2, and MACE-MP-0. For all cases, the single-task DPA-2 models are exclusively trained on the datasets specified in the second column, whereas the multi-task DPA-2 model undergoes pre-training on the entire corpus of pre-training datasets (see Table 1). The multi-task DPA-2 model then employs the fitting head indicated in the second column to initialize the fitting procedure for downstream tasks. All model variants are evaluated on their respective downstream datasets without any additional training. The results demonstrate that multi-task training substantially enhances generalizability compared to the single-task pre-trained DPA-2 and the MPtrj-trained models. The comparable performance between the MPtrj-trained MACE-MP-0 and DPA-2 suggests that the improvement is primarily due to the multi-task pre-training scheme rather than differences in model architecture.

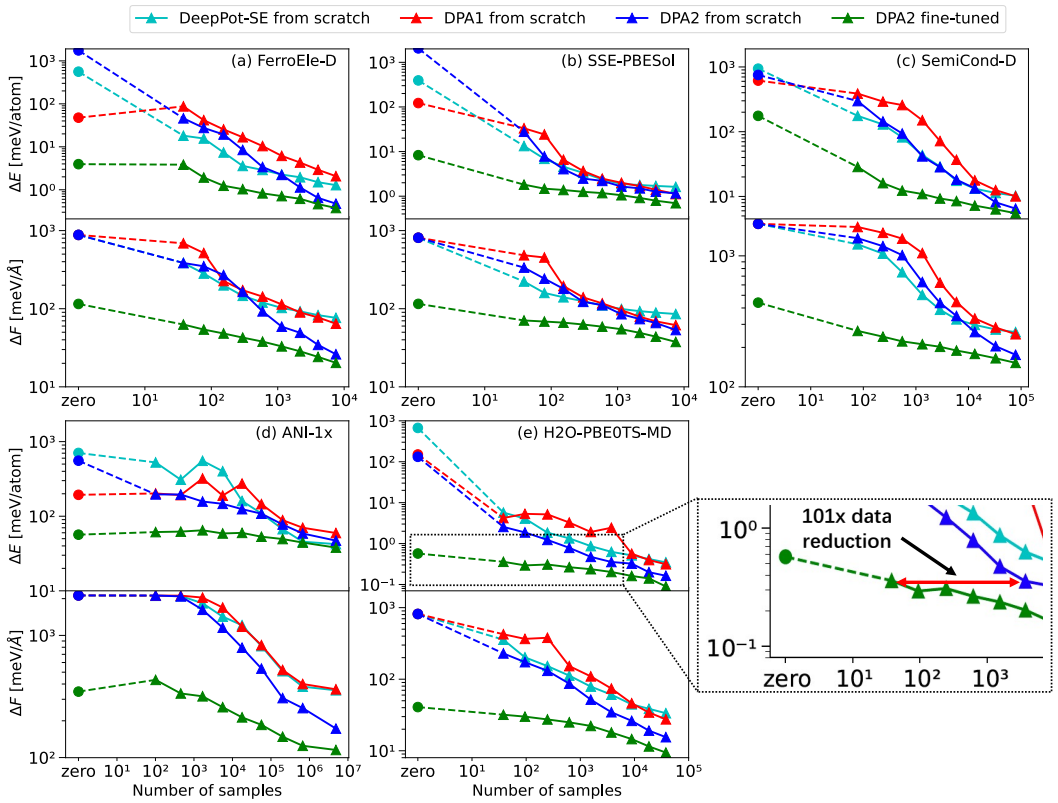

Figure 3: Comparative analysis of sample efficiency on downstream tasks. The horizontal axis represents the volume of downstream data required, while the vertical axis depicts the RMSE convergence in energy or force predictions. For a uniform assessment across models, the number of training epochs per model for each downstream task is normalized to a standard value, derived by dividing 1 million by the number of downstream samples.

2.4 Fine-tuning on downstream tasks

Although zero-shot generalizability is often observed to a certain extent, a gap from perfect generalization typically remains. To bridge this gap, we fine-tune the models using data from the downstream tasks. A stronger generalizability in a pre-trained model implies that less data is required during fine-tuning, leading to higher sample efficiency. The reduction in sample size relative to training a model from scratch quantifies the advantage of employing a multi-task pre-trained model.

The sample efficiency of the pre-trained DPA-2 on downstream tasks was evaluated by comparing it against various other DP models that were trained from scratch. Fig. 3 showcases a selection of downstream tasks, with a comprehensive comparison available in Section S4 of the Supplementary Materials. The figure illustrates the convergence trends of the energy and force RMSEs in relation to the expanding sample size used for downstream training.

To draw distinctions between the fine-tuned DPA-2 and the from-scratch DPA-2 models, it is important to realize that both models share identical architectures. However, the fine-tuned model begins with parameters derived from a multi-task pre-trained model, whereas the from-scratch model starts with randomly initialized parameters. The fine-tuned DPA-2 model consistently achieves lower error curves compared to the DPA-2 model trained from scratch, particularly when the available downstream data is scarce. This translates to a considerable reduction in the amount of data needed to reach equivalent levels of accuracy. Taking the H2O-PBE0TS-MD task for example, two orders of magnitudes of training data are saved to reach the same energy accuracy, see the zoomed-in of Fig. 3. As the sample size grows, the performance disparity between the fine-tuned and from-scratch DPA-2 models diminishes. This outcome is anticipated, given that both models possess the same capacity and, theoretically, their accuracy should converge as the dataset approaches an infinite size. When

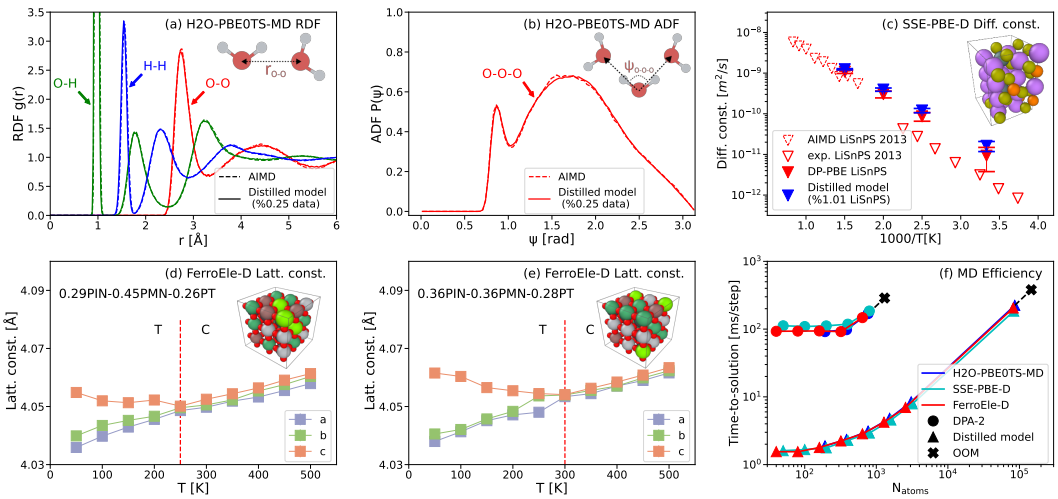

Figure 4: Evaluation of the distilled model across various downstream applications. (a-b) Comparison of the radial distribution function (RDF) and angular distribution function (ADF) for the H2O-PBE0TS-MD dataset between the reference AIMD results [85] and the distilled model. The model is distilled from a DPA-2 model fine-tuned from merely 0.25% of DFT-labeled data. (c) A comparison of diffusion constants for the solid-state electrolyte $Li_{10}SnP_2S_{12}$. The constants were determined using various methods: the distilled model, DPMD as reported in Huang et al. (2021) [86], AIMD simulations from the studies by Mo et al. (2012) and Marcolongo et al. (2017) [87, 88], and experimental findings from solid-state nuclear magnetic resonance (NMR) as documented by Kuhn et al. (2013) [89]. The distilled model is trained from a DPA-2 model fine-tuned by 1.01% of the SSE-PBE-D data. (d-e) The temperature-dependent lattice constants for the ternary solid solution ferroelectric perovskite oxides $Pb(In_{1/2}Nb_{1/2})O_3$–$Pb(Mg_{1/3}Nb_{2/3})O_3$–$PbTiO_3$ (PIN-PMN-PT). The NPT MD simulations using the distilled model are conducted for two concentrations, 0.29PIN–0.45PMN–0.26PT and 0.36PIN-0.36PMN-0.28PT [90]. The model is distilled from a DPA-2 model fine-tuned with the complete FerrEle-P dataset and 7.86% of the FerrEle-D data. (f) Computational efficiency assessment for the aforementioned three systems, showcasing the time-to-solution as a function of the system size in the number of atoms (N_{atoms}).

comparing DeepPot-SE (DP-SE), DPA-1, and DPA-2 models trained from scratch, the DPA-2 model exhibits superior performance over the other architectures. While the convergence patterns of the DPA-1 and DP-SE models are somewhat parallel, the DP-SE model reaches a performance plateau more rapidly than the DPA-1 in the FerroEle-D, SSE-PBESol, and SemiCond-D tasks.

2.5 Model distillation and evaluation

The fine-tuned DPA-2 model typically suffers from computational inefficiency due to its extensive parameter set, as illustrated in Fig. 4(f). To address this, we employed a knowledge distillation approach, transferring insights from the fine-tuned DPA-2 models to compressed DPA-1 models without attention layers. We evaluated the performance of these distilled models in terms of efficiency and accuracy on three benchmark downstream tasks: H2O-PBE0TS-MD, SSE-PBE-D, and FerroEle-D. Notably, in all the cases, the fine-tuned models are exposed to only a small portion (0.25%–7.86%, see Tab. S4) of the downstream dataset, and are used to generate the distillation training datasets that sufficiently cover the relevant configuration spaces. In the FerroEle-D task, we append the full FerroEle-P to a small (7.86%) portion of the FerroEle-D dataset for the training of the fine-tuned model. The FerroEle-D that contains solid solution perovskite oxides was generated by the concurrent learning scheme starting from the FerroEle-P dataset that contains unitary perovskite (see Ref. [90] and Supplementary Materials Sec. S1). Consequently, the FerroEle-D dataset alone does not provide a comprehensive basis for training a fully capable potential model.

After distillation, the time-to-solution and the maximal system size that can be simulated on a single GPU card improved by nearly two orders of magnitude, as shown in Fig. 4(f). Moreover, the accuracy of the distilled models is on par with that of the fine-tuned DPA-2 models, as detailed in Tab. S4. The

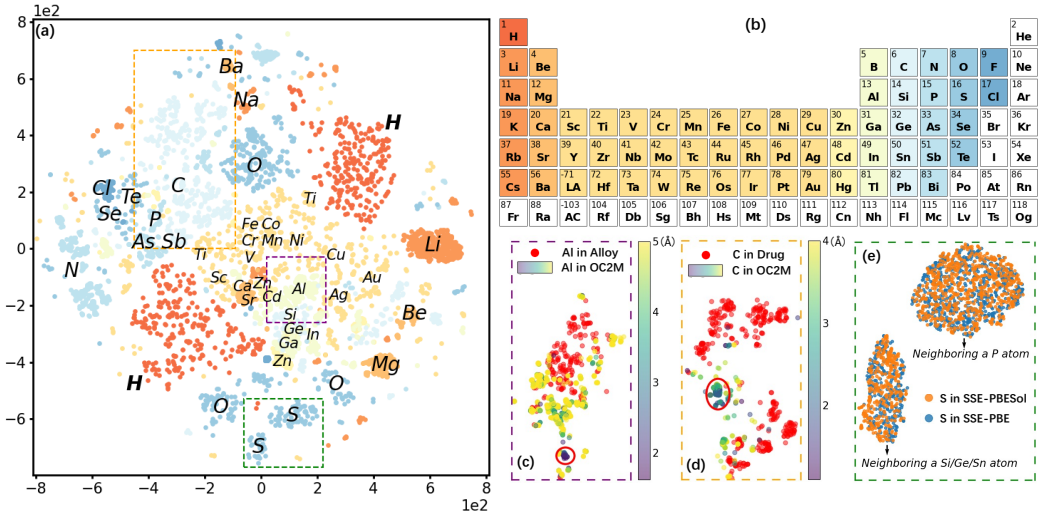

Figure 5: t-SNE visualizations of the DPA-2 single-atom representation of the chemical and configurational space. (a-b) Different colors correspond to the different groups in the periodic table. From group IA to group VII, red gradually transitions to blue. (c) The representations of aluminum in Alloy and OC2M datasets. Red points represent aluminum in Alloy dataset.The gradient colors represent different shortest distances of aluminum in catalyst materials from the adsorbates in the OC2M dataset. (d) The representations of carbon in Drug dataset and in adsorbates of the OC2M dataset. Red points represent carbon in Drug dataset. The gradient colors represent different shortest distances of carbon in adsorbates from the catalyst materials in OC2M dataset. (e)The representations of sulfur in SSE-PBE and SSE-PBESol datasets.

distilled models appear to have reached the peak of their performance, given that their accuracies closely match those of the DPA-1 models (without an attention layer) when trained on the complete downstream datasets.

Finally, to validate the reliability of the distilled models beyond the energy and force RMSEs, we have conducted various application tests on the aforementioned three systems, as reported in Fig. 4(a-e). In the downstream task of H2O-PBE0TS-MD, we observe that the radial distribution functions (RDFs) and the angular distribution function (ADF) of the distilled model are in almost perfect agreement with those obtained from the AIMD simulation, see Fig. 4(a-b). In the downstream task of SSE-PBE-D, the diffusion constants of Lithium ions in the $Li_{10}SnP_2S_{12}$ system under different temperature conditions are calculated. The distilled model presents satisfactory agreement with the previously reported MD simulations using DP-PBE LiSnPS model and DFT (i.e. AIMD simulations) [87, 88], see Fig. 4(c). The discrepancy between the simulation and the experimental results [89] may be attributed to the approximation error of the density functional and finite size effects, as discussed in Ref. [86]. In the downstream task of FerroEle-D, we investigated the temperature-driven phase transition in the solid solution ferroelectric perovskite $Pb(In_{1/2}Nb_{1/2})O_3$–$Pb(Mg_{1/3}Nb_{2/3})O_3$–$PbTiO_3$ (PIN-PMN-PT), see Fig. 4(d-e). Tetragonal-cubic (T-C) transitions are observed at $\sim$ 250 and $\sim$ 300 K for two concentrations 0.29PIN–0.45PMN–0.26PT and 0.36PIN-0.36PMN-0.28PT, repectively. The fact that the transition temperature raises for $\sim$ 50 K due to the increment in the PIN ($Pb(In_{1/2}Nb_{1/2})O_3$) portion from 29% to 36% is in line with the experimental observations [91, 92].

2.6 The representation learned by the DPA-2 model

We present a visualization of the update of single-atom representations by the final repformer layer using a 2-dimensional t-SNE plot [93], as depicted in Fig.5. In Fig.5(a), colors denote distinct groups in the periodic table, as annotated in Fig.5(b). Notably, Fig.5(a) reveals that representations of identical chemical species tend to form cohesive clusters in the t-SNE latent space. The distribution of these representations distinctly aligns with known chemistry: The elements in groups IA and IIA

are clustered at the top right of the t-SNE plot; The non-metals cluster predominantly at the top left and bottom; The transition metals, typically positioned at the middle of the periodic table, are accordingly situated in the central region of the t-SNE figure. However, hydrogen (H) presents an exception, exhibiting two clusters: one aligned with metals, primarily in water datasets, and another near non-metals, particularly in molecule datasets such as Drug, ANI-1x, and Transition-1x.

Elements such as Copper (Cu), Silver (Ag), and Gold (Au) in group IB exhibit a tendency to cluster closer to Lithium (Li) than other transition metals due to their shared possession of one s-electron in the outermost electron shell. Similarly, representations of group IIA elements like Calcium (Ca) and Strontium (Sr) closely associate with those of group IIB elements such as Zinc (Zn) and Cadmium (Cd) owing to their shared possession of two s-electrons in the outermost electron shell. Additionally, there's a discernible trend for elements from the same group in the periodic table to cluster together, as evident with Phosphorus (P), Arsenic (As), and Antimony (Sb) from group VII, and Selenium (Se) and Tellurium (Te) from group VIII.

The DPA-2 representation effectively distinguishes between various chemical and configurational environments, as showcased in Fig.5(c-e). In Fig.5(c), representations of Aluminum (Al) atoms from the Alloy and OC2M datasets are depicted. The color gradient from purple to yellow indicates the distance of the Al atom from the closest adsorbate in the OC2M dataset, while Al atoms from the Alloy dataset (all-metal environment) are colored red. Notably, Al atoms distanced from adsorbates closely resemble those in the Alloy dataset, indicative of similar chemical and configurational environments, whereas those in proximity to adsorbates exhibit discernible differences (see the red-circled blue cluster). Similarly, Fig.5(d) illustrates representations of Carbon (C) atoms in the Drug and OC2M datasets. Carbon atoms in adsorbates closer to catalyst materials are positioned farther away in latent space from representations in the Drug dataset due to more pronounced differences in their chemical and configurational environments.

Moreover, the DPA-2 representation shows insensitivity to DFT labeling accuracy. As demonstrated in Fig. 5(e), representations of sulfur (S) in SSE-PBE (labeled with PBE exchange correlation functional) and SSE-PBESol (labeled with PBE-Sol exchange correlation functional) datasets exhibit mutual overlap. The S atoms form two clusters, with one cluster indicating a phosphorus neighboring atom and the other representing a neighboring Si/Ge/Sn atom.

In summary, our analysis reveals that atoms sharing similar chemical and configurational environments are closer in the representation space learned by the DPA-2 model. Thus, the DPA-2 representation emerges as a promising candidate for encoding chemical and configurational information in molecular and condensed-phase applications.

3 Discussion

In this work, we introduce DPA-2, a Large Atomic Model (LAM), supported by a comprehensive pipeline that includes multi-task pre-training, fine-tuning, knowledge distillation, and practical deployment. The principal findings concerning DPA-2 are as follows: (1) DPA-2 demonstrates exceptional ability for generalization, primarily due to the multi-task pre-training approach, which utilizes 18 datasets covering 73 chemical elements. These datasets would not typically be merged in a single-task pre-training scenario due to differing labeling methodologies, such as exchange-correlation functionals, energy cutoffs, and k-space grid spacing. (2) In downstream tasks, the multi-task pre-training approach enables a reduction in data requirements by approximately 1–2 orders of magnitude without sacrificing accuracy. These results suggest that the DPA-2 model, along with the proposed workflow, stands as a promising framework for molecular and materials simulation.

It is evident that the existing pre-training datasets for the DPA-2 model are insufficient. For example, the datasets currently in use are notably deficient in information on 2-D materials, which significantly limits the model's generalizability to such systems. As a result, the development of LAMs like DPA-2 must be considered a long-term endeavor. This process necessitates the ongoing collection of diverse training data, the incorporation of application-specific test cases, and the establishment of automated workflows for data preprocessing, model training, model evaluation, and version updates. In recognition of these needs, we underscore the importance of fostering LAMs within an open and collaborative ecosystem. Such an approach would enable the molecular simulation community to both benefit from and contribute to the evolution of LAMs. Reflecting our commitment to this vision,

we have launched the OpenLAM Initiative [5]. Updates on this initiative will be regularly posted on the AIS Square platform [6]. We cordially invite readers to participate in this project in any capacity they deem fit.

4 Methods

4.1 Formulation

In this study, we examine a system consisting of N atoms, where the atomic numbers are represented by the list $\mathcal{Z} = (Z_1, \ldots, Z_i, \ldots, Z_N)$, and the atomic coordinates are denoted by the list $\mathcal{R} = (r_1, \ldots, r_i, \ldots, r_N)$. The potential energy surface (PES) of the system is symbolized by E, a function dependent on elemental types and coordinates, expressed as $E = E(\mathcal{X})$, $\mathcal{X} := (\mathcal{R}, \mathcal{Z})$. The potential energy surface can be further decomposed into the following equation:

$$E = \sum_i E_i, \tag{1}$$

where E_i signifies the atomic energy contributions originating from atom i. The atomic force exerted on atom i, represented as F_i, is defined as the negative gradient of the total energy with respect to the coordinate:

$$F_i = -\nabla_{r_i} E. \tag{2}$$

For periodic systems, the virial tensor can be obtained as follows:

$$\Xi_{\alpha\beta} = -\sum_\gamma \frac{\partial E}{\partial h_{\gamma\alpha}} h_{\gamma\beta}, \tag{3}$$

where $\Xi_{\alpha\beta}$ corresponds to the $\alpha\beta$ component of the virial tensor, and $h_{\alpha\beta}$ yields the β-th component of the α-th cell vector.

4.2 The DPA-2 model

4.2.1 The overall architecture of the DPA-2 model

The DPA-2 is a model that predicts the atomic energy contribution based on the atomic numbers $\mathcal{Z}$ and the coordinates $\mathcal{R}$. It consists of two parts,

$$E_i = \mathcal{F}\Big(\mathcal{D}_i\big(\mathcal{R}, \mathcal{Z}\big)\Big), \tag{4}$$

where $\mathcal{D}_i$ represents the descriptor of atom i. The descriptor must be a smooth mapping from the atomic numbers and coordinates to a hidden representation that remains invariant under translational, rotational, and permutational (only among atoms with the same atomic number) operations.

The fitting network $\mathcal{F}$ is usually modeled by a standard multiple-layer perceptron (MLP) composed of an energy-biasing layer,

$$\mathcal{F}(\mathcal{D}_i) = e_{\text{bias}}\Big(\text{MLP}(\mathcal{D}_i)\Big). \tag{5}$$

The energy bias layer "e_{bias}" adds a constant bias to the atomic energy contribution according to the atomic number, i.e., $e_{\text{bias}}(Z_i)(\text{MLP}(\mathcal{D}_i)) = \text{MLP}(\mathcal{D}_i) + e_{\text{bias}}(Z_i)$. Ideally, the energy bias e_{bias} should be taken as the energy of an atom in a vacuum. In practice, the energy bias may be determined by a least-square fitting of the energies in the training data. More precisely, suppose we have M data frames, and within the m-th frame, we have c_{mz} atoms with atom number z, and the DFT labeled energy of the frame is denoted by E_m^*. Then the linear system

$$\sum_z c_{mz} e_{\text{bias}}(z) = E_m^*, \quad m = 1, \ldots, M, \tag{6}$$

is solved in the least-square sense. Here we assume that the number of independent equations in system Eq. (6) is equal to or smaller than the number of frames M.

[5] `https://deepmodeling.github.io/blog/openlam/`

[6] `https://www.aissquare.com/openlam`

The DPA-2 descriptor is graphically illustrated in Fig. 2, specifically,

$$\mathcal{D}_i = \mathrm{concat}(f_i^0, f_i^2), \tag{7}$$

where f_i^0 and f_i^2 denote the single-atom representations of atom i. The requirements for smoothness and symmetry preservation in single-atom representations are identical to those for the descriptor. The representation f_i^0 is defined as

$$f_i^0 = \mathrm{MLP}\left(\mathrm{one\_hot}(Z_i)\right). \tag{8}$$

The atomic number, Z_i, is initially converted into a one-hot representation and subsequently embedded by an MLP. The output f_i^0 is the single-atom hidden representation with dimension n_1^0. The single-atom representation is updated by the $\mathrm{repinit}$ (representation-initializer) layer that encodes the information of local configuration, expressed by the pair-atom representations g_{ij}^0 a and h_{ij}^0, into the single-atom representation.

$$f_i^1 = \mathrm{repinit}(f_i^0, g_{ij}^0, h_{ij}^0). \tag{9}$$

The feature f_i^2 is mapped from single-atom representation and pair-atoms representations g_{ij}^0, h_{ij} by a multiple-layer structure,

$$f_i^2 = \underbrace{\mathrm{repformer} \circ \cdots \circ \mathrm{repformer}}_{\times 12}\Big(\mathrm{linear}(f_i^1), \mathrm{linear}(g_{ij}^1), h_{ij}^1\Big), \tag{10}$$

where the single- and pair-atom representations are updated by $\mathrm{repformer}$ (representation-transformer) layers. The $\mathrm{repformer}$ is designed in a way that the input and output representations share the shape dimension, thus they are stacked 12 times. The "$\circ$" in Eq. (10) thus denotes the layer composition (or mathematically the function composition). The linear mappings are used to change the dimension of f_i^1 and g_{ij}^1 to match the shape requirement of $\mathrm{repformer}$. The pair-atom representations g_{ij}^0, h_{ij}^0, g_{ij}^1 and h_{ij}^1 will be introduced shortly later. It is assumed that the $\mathrm{repinit}$ and $\mathrm{repformer}$ layers only require the information of i's neighboring atoms, i.e., all atoms falling within a sphere centered at atom i with a radius r_c. This radius is commonly referred to as the cut-off radius. We thus introduce the notation $N_{r_c}(i)$, which represents the set of all neighbors of i, i.e., $N_{r_c}(i) = \{j : j \neq i, |r_j - r_i| < r_c\}$. The maximum possible number of neighbors for the atoms in the system is denoted by $N_{r_c}^m$, so we have $|N_{r_c}(i)| \leq N_{r_c}^m, \forall i$.

To define the pair-atom representations, g_{ij}^0, h_{ij}^0, we consider the local configuration of atom i represented by the augmented environment matrix with shape $N_{r_c^0}^m \times 4$, where r_c^0 is the cut-off radius used to compute the pair-atom representations. The j-th row of the environment matrix, being a 4-dimensional vector, is defined by

$$\tilde{r}_{ij} = s(r_{ij}) \times \left(1, \frac{x_{ij}}{|r_{ij}|}, \frac{y_{ij}}{|r_{ij}|}, \frac{z_{ij}}{|r_{ij}|}\right), \tag{11}$$

where (x_{ij}, y_{ij}, z_{ij}) are the Cartesian coordinates of the relative position $r_{ij} = r_i - r_j$. In most cases, the number of neighbors is smaller than $N_{r_c}^m$, so the environment matrix only has $|N_{r_c}(i)|$ rows defined by Eq. (11), and the remaining positions are filled with zeros. The switched inverse distance function s in Eq. (11) is defined by

$$s\left(r_{ij}\right) = \frac{w_{ij}}{|r_{ij}|}, \quad w_{ij} = w(|r_{ij}|). \tag{12}$$

The switch function w takes the value 0 outside the cut-off radius r_c, and 1 inside a starting point of switching, denoted by r_{cs}. In between r_{cs} and r_c, the switch function smoothly changes from 1 to 0. It is required that w has a continuous second-order derivative on $\mathbb{R}$. One possible implementation of w is provided as

$$w\left(|r_{ij}|\right) = \begin{cases} 1 & \text{if } r_{ij} < r_{cs}, \\ u^3\left(-6u^2 + 15u - 10\right) + 1 & \text{if } r_{cs} \leq r_{ij} < r_c, \\ 0 & \text{if } r_c \leq r_{ij}, \end{cases} \tag{13}$$

where $u = (|r_{ij}| - r_{cs})/(r_c - r_{cs})$ and $r_{cs} < r_c$ is the starting point of the smooth switch.

The first column of the augmented environment matrix is defined as the rotationally invariant pair-atom representation, while the remaining three columns are denoted by the rotationally equivariant pair-atom representation, i.e.

$$g_{ij}^0 = s(r_{ij}), \tag{14}$$

$$h_{ij}^0 = s(r_{ij}) \times \left(\frac{x_{ij}}{|r_{ij}|}, \frac{y_{ij}}{|r_{ij}|}, \frac{z_{ij}}{|r_{ij}|} \right). \tag{15}$$

The procedure for calculating pair-atom representations is graphically illustrated in Fig. 2(f). The representations g_{ij}^1 and h_{ij}^1 are established in precisely the same manner as g_{ij}^0 and h_{ij}^0, with the only potential variation being the selection of a distinct cut-off radius, denoted as r_c^1.

4.2.2 The repinit layer

The $\mathrm{repinit}$ layer only updates the single-atom f_i^0 and pair-atom g_{ij}^0 representations, and does not update the equivariant pair-atom representation h_{ij} that is of dimension 3. The $\mathrm{repinit}$ layer first embeds the concatenated single- and pair-atom representations to update the pair-atom representation

$$g_{ij}^{rt} = \mathrm{MLP}(\mathrm{concat}(f_i^0, f_j^0, g_{ij}^0)), \quad \forall j \in N_{r_c^0}(i) \tag{16}$$

Then, we concatenate the g_{ij}^0 and h_{ij} pair-atom representations to recover the environment matrix and update single-atom representation using a symmetrization operation

$$f_i^1 = \mathrm{linear}(f_i^0) + \mathrm{symm}(g_{ij}^{rt}, \tilde{r}_{ij}). \tag{17}$$

The symmetrization operator, first introduced by Ref. [41], has the general form of $\mathrm{symm}(x_j, y_j)$, where x_j and y_j are neighbor indexed vectors. It is assumed that x_j is rotationally invariant, while y_j is not, but the inner product is rotationally invariant. The symmetrization operator is defined by

$$\mathrm{symm}(x_j, y_j) = \underset{\alpha\gamma}{\mathrm{flatten}}\Big(\sum_{\beta} p_{\alpha\beta}\, p_{\gamma\beta}^{<} \Big), \tag{18}$$

$$p_{\alpha\beta} = \frac{1}{N_{r_c^0}^m} \sum_{j \in N_{r_c^0}(i)} w_{ij}\, x_{j,\alpha}\, y_{j,\beta}, \tag{19}$$

$$p_{\alpha\beta}^{<} = \underset{\alpha}{\mathrm{split}}(p_{\alpha\beta}). \tag{20}$$

In Eq. (18), the matrix dimensions α and γ are flattened to form a vector. In Eq. (19), the summation is taken over the index of neighbors j, making the matrix p permutationally invariant. When an atom comes into the neighborhood of atom i, the quantities x_j and y_j generally do not smoothly switch from 0. To prevent the discontinuous jump, the switch w_{ij} is multiplied. In Eq. (20), the matrix $p_{\alpha\beta}$ is split along the α dimension, and the first certain number of elements are taken and assigned with notation $p^{<}$. It can be proven that the symmetrization operator is invariant with respect to rotational operations and permutational operations over atoms of the same atomic number [41].

4.2.3 The repformer layer

The $\mathrm{repformer}$ layer maintains the input and output dimensions of the single- and pair-atom representations, allowing it to be stacked to enhance its representational capabilities. However, the output of $\mathrm{repinit}$ may not necessarily satisfy the dimension requirements of the $\mathrm{repformer}$ layer. To address this issue, the representations are first projected to the desired shape using a linear layer, as follows:

$$f_i^{2,0} = \mathrm{linear}(f_i^1), \tag{21}$$

$$g_{ij}^{2,0} = \mathrm{linear}(g_{ij}^1), \tag{22}$$

$$h_{ij}^{2,0} = h_{ij}^1. \tag{23}$$

Subsequently, these representations are updated by the $\mathrm{repformer}$ layers. The dimensions of the single- and pair-atom representations are denoted by n_1^2 and n_2^2, respectively. In the subsequent discussion, the input representations for the l-th $\mathrm{repformer}$ layer are denoted by $f_i^{2,l}$ and $g_{ij}^{2,l}$.

In each repformer layer, the single-atom representation is updated by

$$f_i^{2,l+1} = \frac{1}{\sqrt{3}}\Big(f_i^{2,l} + \mathrm{MLP}\big(\tilde{f}_i^{2,l}\big) + \mathrm{loc\_attn}\big(f_i^{2,l}\big) \Big). \tag{24}$$

The intermediate representation $\tilde{f}_i^{2,l}$ is defined by

$$\tilde{f}_i^{2,l} = \mathrm{concat}\Big(f_i^{2,l},\ \frac{1}{N_{r_c^1}^m} \sum_{j \in N_{r_c^1}(i)} w_{ij} g_{ij}^{2,l} \hat{f}_j^{2,l},\ \mathrm{symm}\big(f_j^{2,l}, h_{ij}^{2,l}\big),\ \mathrm{symm}\big(g_{ij}^{2,l}, h_{ij}^{2,l}\big) \Big), \tag{25}$$

where $\hat{f}_j^{2,l}$ is a linearly transformed $f_j^{2,l}$ that has the same dimension as the equivariant pair-atom channel, i.e. $\hat{f}_j^{2,l} = \mathrm{linear}(f_j^{2,l})$. The last term in Eq. (24) is the local multi-head self-attention, defined by

$$\mathrm{loc\_attn}\big(f_i^{2,l}\big) = \underset{\beta,h \to n_1^2}{\mathrm{linear}}\Big(\sum_{j \in N_{r_c^1}(i),\alpha} B_{ij}^{l,\eta} f_{j,\alpha}^l \hat{V}_{\alpha,\beta}^{l,\eta} \Big), \tag{26}$$

with the attention map B given by

$$\hat{q}_{i,\gamma}^{l,\eta} = \sum_\alpha f_{i,\alpha}^l \, \hat{Q}_{\alpha,\gamma}^{l,\eta}, \quad \hat{k}_{j,\gamma}^{l,\eta} = \sum_\beta f_{j,\beta}^l \, \hat{K}_{\beta,\gamma}^{l,\eta}, \tag{27}$$

$$B_{ij}^{l,\eta} = \underset{j \in N_{r_c^1}(i)}{\mathrm{softmax}^*}\Big(\frac{1}{\sqrt{\hat{d}}} \sum_\gamma \hat{q}_{i,\gamma}^{l,\eta} \hat{k}_{j,\gamma}^{l,\eta} \Big). \tag{28}$$

Here, $\hat{d}$ denotes the hidden dimension of the local self-attention, and the $\hat{Q}$, $\hat{K}$, and $\hat{V}$ are trainable matrices. The "$*$" over the softmax operator indicates that the softmax used in Eq. (28) is modified to guarantee the smoothness of the attention map. The definition will be introduced in Sec. 4.2.4.

In each layer, the rotationally invariant pair-atom representation is updated by

$$g_{ij}^{2,l+1} = \frac{1}{\sqrt{4}}\Big(g_{ij}^{2,l} + \mathrm{MLP}(g_{ij}^{2,l}) + w_{ij} \underset{n_1^2 \to n_2^2}{\mathrm{linear}}(f_i^{2,l} \odot f_j^{2,l}) + \mathrm{gated\_attn}\big(g_{ij}^{2,l}, h_{ij}\big) \Big), \tag{29}$$

where the last term in Eq. (29) is the gated multi-head self-attention, which is defined by

$$\mathrm{gated\_attn}\big(g_{ij}^{2,l}, h_{ij}\big) = \underset{\beta,h \to n_2^2}{\mathrm{linear}}\Big(\sum_{k \in N_{r_c^1}(i),\alpha} A_{ijk}^h g_{ik,\alpha}^{2,l} V_{\alpha,\beta}^{l,\eta} \Big). \tag{30}$$

In Eq. (30), the attention map A is given by

$$q_{ij,\gamma}^{l,\eta} = \sum_\alpha g_{ij,\alpha}^{2,l} \, Q_{\alpha,\gamma}^{l,\eta}, \quad k_{ik,\gamma}^{l,\eta} = \sum_\beta g_{ik,\beta}^{2,l} \, K_{\beta,\gamma}^{l,\eta}, \tag{31}$$

$$A_{ijk}^{l,\eta} = \underset{k \in N_{r_c^1}(i)}{\mathrm{softmax}^\dagger}\Big(\Big(\frac{1}{\sqrt{d}} \sum_\gamma q_{ij,\gamma}^{l,\eta} k_{ik,\gamma}^{l,\eta} \Big) \Big(\sum_\delta h_{ij,\delta} h_{ik,\delta} \Big) \Big), \tag{32}$$

where d denotes the hidden dimension of the self-attention, the Q, K, and V are trainable matrices, and η is the index of the attention heads. The gate term $h_{ij} h_{ik}^T$ is proved to be critical to the generalization ability of the model [43]. As detailed in Sec. 4.2.4, the $\dagger$ over the softmax operator indicates that the softmax used in Eq. (32) is modified to guarantee the smoothness.

We notice that it is fully valid to update the rotationally equivariant representation h_{ij} in a similar way, e.g.,

$$h_{ij}^{2,l+1} = \frac{1}{\sqrt{2}}\Big(h_{ij}^{2,l} + \underset{h}{\mathrm{linear}}\Big(\sum_{k \in N_{r_c^1}(i)} A_{ijk}^h h_{ik}^{2,l} \Big) \Big). \tag{33}$$

However, we find such an update would not improve the accuracy and often make the training procedure unstable. Therefore, we choose not to update h_{ij} in the current version of the DPA-2 model.

4.2.4 Smoothness of the softmax operation

The standard softmax is defined by

$$\mathrm{softmax}(x_{ij}) = \frac{e^{x_{ij}}}{\sum_k e^{x_{ik}}}, \tag{34}$$

which introduces discontinuity in the attention maps in Eqs. (28) and (32). Simply multiplying a switch to the attention maps does not fix the problem. Suppose that one atom comes into the cut-off; the denominator of Eq. (34) changes in a discontinuous way, thus all $\mathrm{softmax}(x_{ij})$ change discontinuously, no matter whether j is the new neighbor or not.

To fix this issue, we define the $\mathrm{softmax}^*$ by

$$\mathrm{softmax}^*(x_{ij}) = w_{ij}\,\mathrm{softmax}\big(w_{ij}(x_{ij}+s^*)-s^*\big). \tag{35}$$

Similarly, the $\mathrm{softmax}^\dagger$ is given by

$$\mathrm{softmax}^\dagger(y_{ijk}) = w_{ij}w_{ik}\,\mathrm{softmax}\big(w_{ij}w_{ik}(y_{ijk}+s^\dagger)-s^\dagger\big). \tag{36}$$

It is assumed that the shifting constants s^* and $s^\dagger$ are chosen a magnitude larger than x_{ij} and y_{ijk}, respectively. In practice, the magnitude of both x_{ij} and y_{ijk} in Eqs. (35) and (36) are of order 1, so we set $s^* = s^\dagger = 20$.

4.3 Single-task training

Suppose that we have a training dataset T of size M, and denote the DFT-labeled energy and force for any configuration $\mathcal{X}_m$, $1 \le m \le M$, by E^*_m and $\{F^*_{i,m}\}$, respectively. The dataset T yields

$$T = \{(\mathcal{X}_1, E^*_1, \{F^*_{i,1}\}), \ldots, (\mathcal{X}_M, E^*_M, \{F^*_{i,M}\})\}. \tag{37}$$

We denote the trainable parameters of the descriptor by θ, and those of the fitting network by ξ. When necessary, the parameters are placed as superscripts of the corresponding notation, i.e., we have $\mathcal{D}^\theta_i$ and $\mathcal{F}^\xi$ for the descriptor and fitting network, respectively. The PES model is thus rewritten as $E = E^{\theta,\xi}(\mathcal{X})$. The loss function at training step t is written as

$$\mathcal{L}(\theta,\xi,B,t) = \frac{1}{|B|}\sum_{m\in B}\Big(\frac{p_e(t)}{N}\left|\Delta E^{\theta,\xi}_m\right|^2 + \frac{p_f(t)}{3N}\sum_i\left|\Delta F^{\theta,\xi}_{i,m}\right|^2\Big), \tag{38}$$

$$\Delta E^{\theta,\xi}_m = E^{\theta,\xi}(\mathcal{X}_m) - E^*_m, \tag{39}$$

$$\Delta F^{\theta,\xi}_{i,m} = F^{\theta,\xi}_i(\mathcal{X}_m) - F^*_{i,m}, \tag{40}$$

where B, a randomly sampled subset of $\{1,\ldots,M\}$, represents the minibatch of the training dataset. $p_e(t)$ and $p_f(t)$ are the energy and force prefactors, respectively. If the learning rate at step t is denoted by $\gamma(t)$, then the prefactors are defined by

$$p_\xi(t) = p^{\mathrm{start}}_\xi\frac{\gamma(t)}{\gamma(0)} + p^{\mathrm{limit}}_\xi\Big(1-\frac{\gamma(t)}{\gamma(0)}\Big), \quad \xi\in\{e,f\}. \tag{41}$$

At the beginning of the training, the prefactor p_ξ is set to a hyperparameter p^{start}_ξ, and it linearly decays with respect to the learning rate. If the learning rate decays to zero, i.e., $\lim_{t\to\infty}\gamma(t) = 0$, the prefactor converges to the hyperparameter p^{limit}_ξ at the infinite training step. We have adopted the Adam stochastic gradient descent method [94] to minimize the loss function with respect to the model parameters θ and ξ. Virial errors, which are omitted here, can be added to the loss for training if available.

4.4 Multi-task training protocol

For various datasets labeled with different DFT calculation parameters, it is infeasible to merge them directly into a single training set for model training. However, these DFT datasets should inherently share a significant amount of commonality, and we expect they can mutually promote each other's training, thus benefiting the overall model capacity.

In this work, to fully utilize various sources of DFT calculated data, we propose a novel *multi-task* training strategy using a unified model framework for simultaneous training on data calculated with different DFT parameters, as illustrated in Fig. 1(a). We first group all the training data into K training datasets, denoted as $\mathcal{T} = \{T_1, \ldots, T_K\}$, where each dataset contains configurations labeled with identical DFT parameters. The configurations and labels in the k-th training dataset are represented by:

$$T_k = \{(\mathcal{X}_{k1}, E^*_{k1}, \{F^*_{i,k1}\}), \ldots, (\mathcal{X}_{kM}, E^*_{kM}, \{F^*_{i,kM}\})\}. \tag{42}$$

We establish a DPA-2 model with the unified descriptor and K fitting networks, and the k-th model is given by:

$$E = E^{\theta,\xi_k}(\mathcal{X}), \tag{43}$$

where ξ_k represents the network parameters of the k-th fitting network. The k-th fitting network is trained by the k-th training dataset, while the unified descriptor (with parameters θ) is *simultaneously* trained by all datasets, and the loss function is given by

$$\mathcal{L}(\theta, \{\xi_k\}, S, \{B\}, t) = \frac{1}{|S|} \sum_{k \in S} \frac{1}{|B_k|} \sum_{m \in B_k} \Big(\frac{p_e(t)}{N_m} \left| \Delta E^{\theta,\xi_k}_{km} \right|^2 + \frac{p_f(t)}{3N_m} \sum_i \left| \Delta F^{\theta,\xi_k}_{i,km} \right|^2 \Big), \tag{44}$$

$$\Delta E^{\theta,\xi_k}_{km} = E^{\theta,\xi_k}(\mathcal{X}_{km}) - E^*_{km}, \tag{45}$$

$$\Delta F^{\theta,\xi_k}_{i,km} = F^{\theta,\xi_k}_i(\mathcal{X}_{km}) - F^*_{i,km}. \tag{46}$$

At each training step, a subset of the training datasets is sampled from $\mathcal{T}$, and the indices of the sampled datasets are denoted by S. B_k represents the minibatch of the training dataset T_k. It should be noted that there is a significant degree of freedom in designing the sampling strategy for S. Sampling can be conducted with a uniform probability or with a bias towards certain systems. Furthermore, sampling may be performed with or without replacement. In our implementation, larger and more complex datasets are assigned a higher probability, and sampling with replacement is employed.

4.5 Pre-training and fine-tuning

By utilizing multi-task training on all available training datasets, the configurational and elemental knowledge shared among the datasets is expected to be encoded in the descriptor $\mathcal{D}^{\theta_p}$, with θ_p denoting the converged model parameters. The fitting networks are expected to encode system-specific knowledge. The multi-task training scheme provides the possibility of training with a large number of training datasets (most likely labeled with distinct DFT parameters). Therefore, when trained with a sufficiently large dataset that covers a wide range of configurations and elements for future applications, it is expected that much less training data would be needed to train a new system with the help of the encoded knowledge. The multi-task *pre-trained* model can be used to improve the accuracy and data efficiency in *downstream tasks*. It is worth noting that the downstream task can be either constructing a PES, or a property prediction task, and in this work, we only discuss the PES as a downstream task. The procedure of training a model for downstream tasks from a pre-trained model is called *fine-tuning*.

Given a downstream task training dataset, we may initialize the descriptor of our downstream task model with θ_p to boost the performance compared to a random initialization of the descriptor parameters. Furthermore, if the downstream dataset shares similar configurational and elemental information with any of the fitting networks, then the fitting network of the model could also be initialized with the pre-trained fitting network. The energy bias of the downstream task is determined by the downstream training dataset, rather than by those used in the pre-training stage.

4.6 Model distillation

The fine-tuned model possesses a large number of parameters, which might result in low efficiency when directly applied to production scenarios, such as MD simulations. To mitigate this issue, we can distill the model into a more compact version that maintains accuracy on downstream tasks while concurrently achieving speed enhancements and enabling large-scale simulations. The distillation process, illustrated in Fig. 1(c), consists of an iterative concurrent learning loop. The model prior

to distillation, denoted as the teacher model, is used for data labeling, whereas a student model featuring a simpler model structure (e.g., DPA-1 without any attention layer, which can be further compressed [78] to significantly enhance performance) is trained on the labeled data. Subsequently, the teacher model is utilized for MD exploration, adopting simulation settings similar to those of downstream tasks, ensuring that the elemental and configurational spaces explored during distillation and downstream tasks exhibit overlap. Configurations are sampled from the simulated MD trajectories, and the inference deviations between the teacher and student models on those samples are assessed. Samples with model deviation exceeding a predetermined threshold are added to the training dataset for the next iteration. This procedure is repeated until the student model's accuracy satisfies our criteria or no longer changes.

5 Data and Code Availability

The datasets and models used in this study, as detailed in Sec. S1 of the Supplementary Materials, are all available on AIS Square (`https://www.aissquare.com`). The codes, datasets and input scripts are all available on zenodo (`https://doi.org/10.5281/zenodo.13342300`). Finally, to test the models, users are welcome to consider going through this Bohrium Notebook (`https://nb.bohrium.dp.tech/detail/18475433825`), and explore the DP Combo web server (`https://app.bohrium.dp.tech/dp-combo`).

6 Acknowledgements

We gratefully acknowledge the support received for this work. The work of Han Wang is supported by the National Key R&D Program of China (Grant No. 2022YFA1004300) and the National Natural Science Foundation of China (Grant No. 12122103). The work of Weinan E is supported by the National Key Research and Development Project of China (Grant No. 2022YFA1004302) and the National Natural Science Foundation of China (Grants No. 92270001 and No. 12288101). The work of Jinzhe Zeng and Darrin M. York is supported by the National Institutes of Health (Grant No. GM107485 to D.M.Y.) and the National Science Foundation (Grant No. 2209718 to D.M.Y.). The work of Shi Liu is supported by the Natural Science Foundation of Zhejiang Province (Grant No. 2022XHSJJ006). The work of Tong Zhu is supported by the National Natural Science Foundation of China (Grants No. 22222303 and No. 22173032). The work of Zhicheng Zhong is supported by the National Key R&D Program of China (Grants No. 2021YFA0718900 and No. 2022YFA1403000). The work of Jian Lv is supported by the National Natural Science Foundation of China (Grants No. 12034009 and No. 91961204). The work of Jun Cheng is supported by the National Science Fund for Distinguished Young Scholars (Grant No. 22225302), Laboratory of AI for Electrochemistry (AI4EC), and IKKEM (Grants No. RD2023100101 and No. RD2022070501). The work of Mohan Chen is supported by the National Natural Science Foundation of China (Grants No. 12122401, No. 12074007, and No. 12135002). The work of Yifan Li is supported by the "Chemistry in Solution and at Interfaces" (CSI) Center funded by the United States Department of Energy under Award No. DE-SC0019394. Lastly, the computational resource was supported by the Bohrium Cloud Platform at DP Technology and Tan Kah Kee Supercomputing Center (IKKEM).

References

[1] Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical review*, 136(3B):B864, 1964.

[2] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical review*, 140(4A):A1133, 1965.

[3] Magd Badaoui, Pedro J Buigues, Dénes Berta, Gaurav M Mandana, Hankang Gu, Tamas Foldes, Callum J Dickson, Viktor Hornak, Mitsunori Kato, Carla Molteni, et al. Combined free-energy calculation and machine learning methods for understanding ligand unbinding kinetics. *Journal of chemical theory and computation*, 18(4):2543–2555, 2022.

[4] Jinzhe Zeng, Yujun Tao, Timothy J Giese, and Darrin M York. Qdπ: A quantum deep potential interaction model for drug discovery. *Journal of Chemical Theory and Computation*, 19(4):1261–1275, 2023.

[5] Albert P Bartók, James Kermode, Noam Bernstein, and Gábor Csányi. Machine learning a general-purpose interatomic potential for silicon. *Physical Review X*, 8(4):041048, 2018.

[6] Volker L Deringer, Miguel A Caro, and Gábor Csányi. A general-purpose machine-learning force field for bulk and nanostructured phosphorus. *Nature communications*, 11(1):1–11, 2020.

[7] Tongqi Wen, Linfeng Zhang, Han Wang, Weinan E, and David J Srolovitz. Deep potentials for materials science. *Materials Futures*, 1(2):022601, 2022.

[8] Sicong Ma and Zhi-Pan Liu. Machine learning for atomic simulation and activity prediction in heterogeneous catalysis: current status and future. *ACS Catalysis*, 10(22):13213–13226, 2020.

[9] Manyi Yang, Umberto Raucci, and Michele Parrinello. Ammonia decomposition on lithium imide surfaces: A new paradigm in heterogeneous catalysis. *ChemRxiv*, 2022.

[10] Roberto Car and Michele Parrinello. Unified approach for molecular dynamics and density-functional theory. *Physical review letters*, 55(22):2471, 1985.

[11] Justin S Smith, Ben Nebgen, Nicholas Lubbers, Olexandr Isayev, and Adrian E Roitberg. Less is more: Sampling chemical space with active learning. *The Journal of chemical physics*, 148(24):241733, 2018.

[12] Elena Uteva, Richard S Graham, Richard D Wilkinson, and Richard J Wheatley. Active learning in gaussian process interpolation of potential energy surfaces. *The Journal of chemical physics*, 149(17):174114, 2018.

[13] Linfeng Zhang, De-Ye Lin, Han Wang, Roberto Car, and Weinan E. Active learning of uniformly accurate interatomic potentials for materials simulation. *Physical Review Materials*, 3(2):023804, 2019.

[14] Yuzhi Zhang, Haidi Wang, Weijie Chen, Jinzhe Zeng, Linfeng Zhang, Han Wang, and Weinan E. Dp-gen: A concurrent learning platform for the generation of reliable deep learning based potential energy models. *Computer Physics Communications*, 253:107206, 2020.

[15] Wanrun Jiang, Yuzhi Zhang, Linfeng Zhang, and Han Wang. Accurate deep potential model for the al–cu–mg alloy in the full concentration space. *Chinese Physics B*, 30(5):050706, 2021.

[16] So Takamoto, Chikashi Shinagawa, Daisuke Motoki, Kosuke Nakago, Wenwen Li, Iori Kurata, Taku Watanabe, Yoshihiro Yayama, Hiroki Iriguchi, Yusuke Asano, et al. Towards universal neural network potential for material discovery applicable to arbitrary combination of 45 elements. *Nature Communications*, 13(1):2991, 2022.

[17] Chi Chen and Shyue Ping Ong. A universal graph deep learning interatomic potential for the periodic table. *arXiv preprint arXiv:2202.02450*, 2022.

[18] Kamal Choudhary, Brian DeCost, Lily Major, Keith Butler, Jeyan Thiyagalingam, and Francesca Tavazza. Unified graph neural network force-field for the periodic table: solid state applications. *Digital Discovery*, 2(2):346–355, 2023.

[19] Bowen Deng, Peichen Zhong, KyuJung Jun, Janosh Riebesell, Kevin Han, Christopher J Bartel, and Gerbrand Ceder. Chgnet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence*, 5(9):1031–1041, 2023.

[20] Ilyes Batatia, Philipp Benner, Yuan Chiang, Alin M. Elena, Dávid P. Kovács, Janosh Riebesell, Xavier R. Advincula, Mark Asta, William J. Baldwin, Noam Bernstein, Arghya Bhowmik, Samuel M. Blau, Vlad Cărare, James P. Darby, Sandip De, Flaviano Della Pia, Volker L. Deringer, Rokas Elijošius, Zakariya El-Machachi, Edvin Fako, Andrea C. Ferrari, Annalena Genreith-Schriever, Janine George, Rhys E. A. Goodall, Clare P. Grey, Shuang Han, Will Handley, Hendrik H. Heenen, Kersti Hermansson, Christian Holm, Jad Jaafar, Stephan Hofmann, Konstantin S. Jakob, Hyunwook Jung, Venkat Kapil, Aaron D. Kaplan, Nima Karimitari, Namu Kroupa, Jolla Kullgren, Matthew C. Kuner, Domantas Kuryla, Guoda Liepuoniute, Johannes T. Margraf, Ioan-Bogdan Magdău, Angelos Michaelides, J. Harry Moore, Aakash A. Naik, Samuel P. Niblett, Sam Walton Norwood, Niamh O'Neill, Christoph Ortner, Kristin A. Persson, Karsten Reuter, Andrew S. Rosen, Lars L. Schaaf, Christoph Schran, Eric Sivonxay, Tamás K. Stenczel, Viktor Svahn, Christopher Sutton, Cas van der Oord, Eszter Varga-Umbrich, Tejs Vegge, Martin Vondrák, Yangshuai Wang, William C. Witt, Fabian Zills, and Gábor Csányi. A foundation model for atomistic materials chemistry, 2023.

[21] Amil Merchant, Simon Batzner, Samuel S Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Dogus Cubuk. Scaling deep learning for materials discovery. *Nature*, pages 1–6, 2023.

[22] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, et al. The materials project: A materials genome approach to accelerating materials innovation, apl mater. 2013.

[23] John P Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Physical review letters*, 77(18):3865, 1996.

[24] Jeng-Da Chai and Martin Head-Gordon. Systematic optimization of long-range corrected hybrid density functionals. *The Journal of chemical physics*, 128(8), 2008.

[25] Taoyong Cui, Chenyu Tang, Mao Su, Shufei Zhang, Yuqiang Li, Lei Bai, Yuhan Dong, Xingao Gong, and Wanli Ouyang. Gpip: Geometry-enhanced pre-training on interatomic potentials. *arXiv preprint arXiv:2309.15718*, 2023.

[26] Rui Feng, Qi Zhu, Huan Tran, Binghong Chen, Aubrey Toland, Rampi Ramprasad, and Chao Zhang. May the force be with you: Unified force-centric pre-training for 3d molecular conformations. *arXiv preprint arXiv:2308.14759*, 2023.

[27] Leif Jacobson, James Stevenson, Farhad Ramezanghorbani, Steven Dajnowicz, and Karl Leswing. Leveraging multitask learning to improve the transferability of machine learned force fields. *ChemRxiv*, 2023.

[28] Y. Wang, C. Xu, Z. Li, and A. B. Farimani. Denoise pre-training on non-equilibriummolecules for accurate and transferable neuralpotentials. *arXiv preprint arXiv:2303.02216*, 2023.

[29] Iasonas Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6129–6138, 2017.

[30] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[31] Nima Shoghi, Adeesh Kolluru, John R Kitchin, Zachary W Ulissi, C Lawrence Zitnick, and Brandon M Wood. From molecules to materials: Pre-training large generalizable models for atomic property prediction. *arXiv preprint arXiv:2310.16802*, 2023.

[32] Justin S Smith, Benjamin T Nebgen, Roman Zubatyuk, Nicholas Lubbers, Christian Devereux, Kipton Barros, Sergei Tretiak, Olexandr Isayev, and Adrian E Roitberg. Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning. *Nature communications*, 10(1):1–8, 2019.

[33] Adeesh Kolluru, Nima Shoghi, Muhammed Shuaibi, Siddharth Goyal, Abhishek Das, C Lawrence Zitnick, and Zachary Ulissi. Transfer learning using attentions across atomic systems with graph neural networks (taag). *The Journal of Chemical Physics*, 156(18):184702, 2022.

[34] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[35] Mark Tuckerman. *Statistical Mechanics: Theory and Molecular Simulation*. OUP Oxford, 2010.

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

[37] Justin S Smith, Olexandr Isayev, and Adrian E Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chemical science*, 8(4):3192–3203, 2017.

[38] Kristof T Schütt, Farhad Arbabzadah, Stefan Chmiela, Klaus R Müller, and Alexandre Tkatchenko. Quantum-chemical insights from deep tensor neural networks. *Nature communications*, 8(1):1–8, 2017.

[39] Michael Gastegger, Ludwig Schwiedrzik, Marius Bittermann, Florian Berzsenyi, and Philipp Marquetand. wacsf—weighted atom-centered symmetry functions as descriptors in machine learning potentials. *The Journal of chemical physics*, 148(24), 2018.

[40] Linfeng Zhang, Jiequn Han, Han Wang, Roberto Car, and Weinan E. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. *Physical review letters*, 120(14):143001, 2018.

[41] Linfeng Zhang, Jiequn Han, Han Wang, Wissam Saidi, Roberto Car, et al. End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems. *Advances in Neural Information Processing Systems*, 31, 2018.

[42] Jinzhe Zeng, Duo Zhang, Denghui Lu, Pinghui Mo, Zeyu Li, Yixiao Chen, Marián Rynik, Li'ang Huang, Ziyao Li, Shaochen Shi, et al. Deepmd-kit v2: A software package for deep potential models. *The Journal of Chemical Physics*, 159:054801, 2023.

[43] Duo Zhang, Hangrui Bi, Fu-Zhi Dai, Wanrun Jiang, Linfeng Zhang, and Han Wang. Dpa-1: Pretraining of attention-based deep potential model for molecular simulation. *arXiv preprint arXiv:2208.08236*, 2022.

[44] Yaolong Zhang, Ce Hu, and Bin Jiang. Embedded atom neural network potentials: Efficient and accurate machine learning with a physically inspired representation. *The journal of physical chemistry letters*, 10(17):4962–4967, 2019.

[45] Albert P Bartók, Mike C Payne, Risi Kondor, and Gábor Csányi. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Physical review letters*, 104(13):136403, 2010.

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

[47] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O Anatole Von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Physical review letters*, 108(5):058301, 2012.

[48] Stefan Chmiela, Alexandre Tkatchenko, Huziel E Sauceda, Igor Poltavsky, Kristof T Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science advances*, 3(5):e1603015, 2017.

[49] 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.

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

[51] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

[52] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. Se (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *arXiv preprint arXiv:2101.03164*, 2021.

[53] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

[54] Johannes Gasteiger, Shankari Giri, Johannes T Margraf, and Stephan Günnemann. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. *arXiv preprint arXiv:2011.14115*, 2020.

[55] Kristof T Schütt, Oliver T Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. *arXiv preprint arXiv:2102.03150*, 2021.

[56] Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34:6790–6802, 2021.

[57] Muhammed Shuaibi, Adeesh Kolluru, Abhishek Das, Aditya Grover, Anuroop Sriram, Zachary Ulissi, and C Lawrence Zitnick. Rotation invariant graph neural networks using spin convolutions. *arXiv preprint arXiv:2106.09575*, 2021.

[58] Larry Zitnick, Abhishek Das, Adeesh Kolluru, Janice Lan, Muhammed Shuaibi, Anuroop Sriram, Zachary Ulissi, and Brandon Wood. Spherical channels for modeling atomic interactions. *Advances in Neural Information Processing Systems*, 35:8054–8067, 2022.

[59] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):1–11, 2022.

[60] Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Advances in Neural Information Processing Systems*, 35:11423–11436, 2022.

[61] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. *arXiv preprint arXiv:2206.11990*, 2022.

[62] Yi-Lun Liao, Brandon Wood, Abhishek Das, and Tess Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. *arXiv preprint arXiv:2306.12059*, 2023.

[63] Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics. *arXiv preprint arXiv:2204.05249*, 2022.

[64] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[65] Daokun Zhang, Jie Yin, Xingquan Zhu, and Chengqi Zhang. Network representation learning: A survey. *IEEE transactions on Big Data*, 6(1):3–28, 2018.

[66] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[67] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[68] Gengmo Zhou, Zhifeng Gao, Qiankun Ding, Hang Zheng, Hongteng Xu, Zhewei Wei, Linfeng Zhang, and Guolin Ke. Uni-mol: A universal 3d molecular representation learning framework. *ChemRxiv*, 2022.

[69] Jinhua Zhu, Yingce Xia, Lijun Wu, Shufang Xie, Tao Qin, Wengang Zhou, Houqiang Li, and Tie-Yan Liu. Unified 2d and 3d pre-training of molecular representations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2626–2636, 2022.

[70] S. Zaidi, M. Schaarschmidt, J. Martens, H. Kim, Y. W. Teh, A. Sanchez-Gonzalez, P. Battaglia, R. Pascanu, and J. Godwin. Pre-training via denoisingfor molecular property prediction. *arXiv preprint arXiv:2206.00133*, 2022.

[71] S. Lu, Z. Gao, D. He, L. Zhang, and G. Ke. Highly accurate quantum chemical propertyprediction with uni-mol+. *arXiv preprint arXiv:2303.16982*, 2023.

[72] S. Feng, Y. Ni, Y. Lan, Z. Ma, and W.-Y. Ma. Fractional denoising for 3d molecular pre-training. *arXiv preprint arXiv:2307.10683*, 2023.

[73] Rui Jiao, Jiaqi Han, Wenbing Huang, Yu Rong, and Yang Liu. 3d equivariant molecular graph pretraining. *arXiv preprint arXiv:2207.08824*, 2022.

[74] D. Beaini and et.al. Towards foundational models for molecularlearning on large-scale multi-task datasets. *arXiv:2310.04292*, 2023.

[75] K.L.K. Lee and et.al. Towards foundation models for materials science: The open matsci ml toolkit. *arXiv:2310.07864*, 2023.

[76] John LA Gardner, Kathryn T Baker, and Volker L Deringer. Synthetic pre-training for neural-network interatomic potentials. *Machine Learning: Science and Technology*, 2023.

[77] Lowik Chanussot, Abhishek Das, Siddharth Goyal, Thibaut Lavril, Muhammed Shuaibi, Morgane Riviere, Kevin Tran, Javier Heras-Domingo, Caleb Ho, Weihua Hu, et al. Open catalyst 2020 (oc20) dataset and community challenges. *ACS Catalysis*, 11(10):6059–6072, 2021.

[78] Denghui Lu, Wanrun Jiang, Yixiao Chen, Linfeng Zhang, Weile Jia, Han Wang, and Mohan Chen. Dp compress: A model compression scheme for generating efficient deep potential models. *Journal of chemical theory and computation*, 18(9):5559–5567, 2022.

[79] Georg Kresse and Jürgen Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational materials science*, 6(1):15–50, 1996.

[80] Georg Kresse and Jürgen Furthmüller. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Physical review B*, 54(16):11169, 1996.

[81] MJ Frisch, GW Trucks, H Bernhard Schlegel, GE Scuseria, MA Robb, JR Cheeseman, G Scalmani, VPGA Barone, GA Petersson, HJRA Nakatsuji, et al. Gaussian 16, 2016.

[82] Mohan Chen, GC Guo, and Lixin He. Systematically improvable optimized atomic basis sets for ab initio calculations. *Journal of Physics: Condensed Matter*, 22(44):445501, 2010.

[83] Pengfei Li, Xiaohui Liu, Mohan Chen, Peize Lin, Xinguo Ren, Lin Lin, Chao Yang, and Lixin He. Large-scale ab initio simulations based on systematically improvable atomic basis. *Computational Materials Science*, 112:503–517, 2016.

[84] Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ulissi, C Lawrence Zitnick, and Abhishek Das. Gemnet-oc: developing graph neural networks for large and diverse molecular simulation datasets. *arXiv preprint arXiv:2204.02782*, 2022.

[85] Robert A DiStasio, Biswajit Santra, Zhaofeng Li, Xifan Wu, and Roberto Car. The individual and collective effects of exact exchange and dispersion interactions on the ab initio structure of liquid water. *The Journal of chemical physics*, 141(8), 2014.

[86] Jianxing Huang, Linfeng Zhang, Han Wang, Jinbao Zhao, Jun Cheng, and Weinan E. Deep potential generation scheme and simulation protocol for the li10gep2s12-type superionic conductors. *The Journal of Chemical Physics*, 154(9):094703, 2021.

[87] Yifei Mo, Shyue Ping Ong, and Gerbrand Ceder. First principles study of the li10gep2s12 lithium super ionic conductor material. *Chemistry of Materials*, 24(1):15–17, 2012.

[88] Aris Marcolongo and Nicola Marzari. Ionic correlations and failure of nernst-einstein relation in solid-state electrolytes. *Physical Review Materials*, 1(2):025402, 2017.

[89] Alexander Kuhn, Jürgen Köhler, and Bettina V Lotsch. Single-crystal x-ray structure analysis of the superionic conductor li 10 gep 2 s 12. *Physical Chemistry Chemical Physics*, 15(28):11620–11622, 2013.

[90] Jing Wu, Jiyuan Yang, Yuan-Jinsheng Liu, Duo Zhang, Yudi Yang, Yuzhi Zhang, Linfeng Zhang, Shi Liu, et al. Universal interatomic potential for perovskite oxides. *Physical Review B*, 108(18):L180104, 2023.

[91] Dawei Wang, Maosheng Cao, and Shujun Zhang. Phase diagram and properties of pb (in1/2nb1/2) o3–pb (mg1/3nb2/3) o3–pbtio3 polycrystalline ceramics. *Journal of the European Ceramic Society*, 32(2):433–439, 2012.

[92] Qian Li, Sergey Danilkin, Guochu Deng, Zhengrong Li, Ray L Withers, Zhuo Xu, and Yun Liu. Soft phonon modes and diffuse scattering in pb (in1/2nb1/2) o3-pb (mg1/3nb2/3) o3-pbtio3 relaxor ferroelectrics. *Journal of Materiomics*, 4(4):345–352, 2018.

[93] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[94] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.