nature communications

Article

https://doi.org/10.1038/s41467-025-57101-4

Online test-time adaptation for better generalization of interatomic potentials to out-of-distribution data

Received: 9 June 2024

Accepted: 4 February 2025

Published online: 22 February 2025

Check for updates

Taoyong Cui[1,2,5], Chenyu Tang[1,5], Dongzhan Zhou[1], Yuqiang Li[1], Xingao Gong[3,4], Wanli Ouyang[1], Mao Su[1] ✉ & Shufei Zhang[1] ✉

Machine learning interatomic potentials (MLIPs) enable more efficient molecular dynamics (MD) simulations with ab initio accuracy, which have been used in various domains of physical science. However, distribution shift between training and test data causes deterioration of the test performance of MLIPs, and even leads to collapse of MD simulations. In this work, we propose an online Test-time Adaptation Interatomic Potential (TAIP) framework to improve the generalization on test data. Specifically, we design a dual-level self-supervised learning approach that leverages global structure and atomic local environment information to align the model with the test data. Extensive experiments demonstrate TAIP's capability to bridge the domain gap between training and test dataset without additional data. TAIP enhances the test performance on various benchmarks, from small molecule datasets to complex periodic molecular systems with various types of elements. TAIP also enables stable MD simulations where the corresponding baseline models collapse.

Molecular Dynamics (MD) simulation serves as a crucial technique across various disciplines including biology, chemistry, and material science[1–4]. MD simulations are typically based on interatomic potential functions that characterize the potential energy surface of the system, with atomic forces derived as the negative gradients of the potential energies. Subsequently, Newton's laws of motion are applied to simulate the dynamic trajectories of the atoms. In ab initio MD simulations[5], the energies and forces are accurately determined by solving the equations in quantum mechanics. However, the computational demands of ab initio MD limit its practicality in many scenarios. By learning from ab initio calculations, machine learning interatomic potentials (MLIPs) have been developed to achieve much more efficient MD simulations with ab initio-level accuracy[6–8].

Despite their successes, the crucial challenge of implementing MLIPs is the distribution shift between training and test data. When using MLIPs for MD simulations, the data for inference are atomic structures that are continuously generated during simulations based on the predicted forces, and the training set should encompass a wide range of atomic structures to guarantee the accuracy of predictions. However, in fields such as phase transition[9,10], catalysis[11,12], and crystal growth[13,14], the configurational space that needs to be explored is highly complex. This complexity makes it challenging to sample sufficient data for training and easy to make a potential that is not smooth enough to extrapolate to every relevant point. Consequently, a distribution shift between training and test datasets often occurs, which causes the degradation of test performance and leads to the emergence of unrealistic atomic structures, and finally the MD simulations collapse[15]. Although strategies such as active learning[16–18] and pretraining[19–23] have been developed to alleviate this challenge, they still struggle to explore unknown atomic structures and inevitably consume more computational resources. Therefore, a method that address the distribution shift without exploring additional atomic structures is desired.

Test-time adaptation[24–27] emerges as a promising solution that tackles the issue of distribution shifts by fine-tuning the models during the testing phase. Different from the aforementioned methods, test-time adaptation does not require exploring additional atomic structures or

[1]Shanghai Artificial Intelligence Laboratory, Shanghai, China. [2]The Chinese University of Hong Kong, Hong Kong, China. [3]Key Laboratory for Computational Physical Sciences (MOE), State Key Laboratory of Surface Physics, Department of Physics, Fudan University, Shanghai, China. [4]Shanghai Qi Zhi Institute, Shanghai, China. [5]These authors contributed equally: Taoyong Cui and Chenyu Tang. ✉e-mail: sumao@pjlab.org.cn; zhangshufei@pjlab.org.cn

extra training data, instead opting for on-the-fly model adjustments to the model in response to the characteristic of test data with only a modest increase in computational overhead. Test-time adaptation has been proven effective in various domains such as image classification[26–29], semantic segmentation[30–33], and object detection[34–37]. Nevertheless, its application in predicting interatomic potentials remains unexplored. For a successful implementation of test-time adaptation in MLIP, it is crucial to devise task-specific strategies which account for the specific characteristics of atomic structure data. A well-crafted test-time adaptation tailored for MLIP holds substantial promise in improving the accuracy of MLIP, as well as the stability and reliability of MD simulations driven by MLIPs.

In this work, we propose an online Test-time Adaptation strategy for Interatomic Potentials (TAIP) aimed at mitigating the impact of distribution shifts on MLIP applications. We design a dual-level self-supervised learning scheme that help extract both global and local structural information using an encoder. During training, the encoder is trained by the combined losses from both MLIP and self-supervised learning tasks. At the inference stage, the encoder is updated once per test sample by minimizing the self-supervised learning loss, subsequently yielding the final energy and force predictions. This fine-tuning process during inference allows the encoder to extract more adaptive features for test data. We test the accuracy of MLIPs on four datasets, including MD17, ISO17, water, and electrolyte solutions. Compared to baselines, TAIP reduces the prediction errors by an average of 30% without using any additional data. Moreover, we assess the influence of TAIP on the MD simulation stability using periodic water and electrolyte solution datasets and find that TAIP enables stable MD simulations throughout even under conditions where baselines collapse. Finally, visual analysis of feature distributions confirms that TAIP curtails the distribution shifts between training and test datasets.

Results

Preliminaries

MLIPs are used to predict the potential energy and forces within a given atomic system. Various model architectures have been developed, including descriptor-based methods with neural networks[38], linear models[39,40], and Gaussian process regressions[41,42]. Recently, graph neural networks have emerged as a particularly powerful architecture for MLIPs[43–47]. In a typical graph neural network MLIP, a molecular configuration can be represented as n atoms with atomic numbers $Z = \{z_i\}_{i=1}^n$ and coordinates $R = \{\mathbf{r_i}\}_{i=1}^n$. The atom features $X^0 = \{\mathbf{x_i^0}\}_{i=1}^n$ are first initialized with a embedding function $f_{\text{emb}}: \mathbb{N}_+ \mapsto \mathbb{R}^D$ to obtain a D-dimensional feature vector for each atom. The atom features are updated iteratively for L times through a message-passing scheme. The updated atom features X^L carry information about the local environment in which the atoms are located, and are used to predict energy E and forces $F = \{\mathbf{f}_i\}_{i=1}^n$, as well as for the self-supervised learning tasks proposed in this work.

TAIP framework

Our framework comprises a main task of energy and force prediction (Fig. 1a) and three self-supervised learning tasks, which includes noise intensity prediction, atom feature recovery, and pseudo force recovery (Fig. 1b and c). All the tasks share an encoder that transform the input atomic structures into atom features, leveraging the architecture from existing MLIPs like SchNet and PaiNN. As shown in Fig. 1a, the main task block aims to predict the energy E and force $\mathbf{f}_i$ on each atom i. Here, the atom features are updated by the encoder and then aggregated to compute energy and force. The self-supervised learning tasks are designed to improve the performance of the main task by utilizing both local and global structural information. We will discuss these three tasks in detail in the following.

The noise intensity prediction task is shown in Fig. 1b. For each sample, we perturb the atomic coordinates using Gaussian noises with a mean of zero and a variance of t^2 (we call t the noise intensity) to obtain the noisy configuration $\{Z, \tilde{R}\}$. The encoder takes the noisy configuration as input and produces noisy atom features and then aggregated to noisy graph feature $\tilde{\mathbf{u}}$. Meanwhile, the noiseless configuration $\{Z, R\}$ is transformed into clean graph features $\mathbf{u}$ in the same way. Subsequently, the clean and noisy graph features are concatenated to jointly estimate the encoding of the noise intensity $\mathbf{I}$. A noisy configuration with a smaller noise intensity is considered closer to the clean one, so that the noise intensity reflects the dissimilarity between noisy and clean data. Instead of directly predicting the precise value of noise, we categorize the intensity into several bins and determine the bin to which the combined graph feature corresponds. By predicting the noise intensity, the model tries to group graph features of configurations with similar noise intensities, thereby smoothing the configuration representation space.

The atom feature recovery task is shown in the branch containing decoder 1 in Fig. 1c. For each sample, we randomly select a set of atoms C_1 and mask their atom types with a special token [MASK] to obtain the masked atom types $\tilde{Z}$. The masked configuration $\{\tilde{Z}, R\}$ is then transformed into masked atom features $\tilde{X}^L$ by the encoder. These masked atom features $\tilde{X}^L$ are subsequently fed into a decoder (Decoder 1), which is a multi-layer perceptron to predict the original atom type encodings $\{\mathbf{v_i}\}_{i \in C_1}$. The model is trained by computing the cosine similarity between the output encoding and the original one-hot encoding of the masked atoms. By restoring the masked atoms, we can infer the most probable type of neighboring atoms, thereby enhancing the extraction of local information.

The pseudo force recovery task is shown in the branch containing decoder 2 in Fig. 1c. In the main task, the atom features are aggregated to obtain the energy E. In the same way, the masked atom features $\tilde{X}^L$ can be used to calculate a pseudo energy E^{P}. We can also compute the negative gradient of the pseudo forces F^{P} by taking the negative gradient of E^{P} with respective to the coordinates. Then, the pseudo forces are masked with a special token [FMASK] and fed into a decoder (Decoder 2) to reconstruct the masked pseudo forces. By restoring the pseudo forces, the model can capture the relationship among the forces on neighboring atoms, which enhances its understanding of the local atomic environment and helps with the prediction of atomic forces.

The three stages of TAIP including training, test-time adaptation, and inference are shown in Fig. 1d–f.

During the training stage, we utilize a multi-task learning scheme to update all parameters in the model based on the supervised learning loss of the main task, namely energy and force prediction, as well as three self-supervised learning task losses. By leveraging the rich information provided by both supervised and self-supervised learning objectives, this strategy enables the model to learn more expressive feature representations.

During the test-time adaptation stage, since the labels of energy and force are absent, the model parameters can only be adapted to the test sample by minimizing the self-supervised learning losses. Here, we only update the parameters in the encoder once, while keeping other parameters, including those in the two decoders and MLIPs outside of the encoder, fixed. In this way, the model is able to adapt to the test data and achieves better performance.

During the inference stage, the test sample passes through the encoder, which has been updated in the test-time adaptation stage, to predict energy and forces. The other branches of the model regarding the self-supervised learning tasks are not involved.

Experiments

We conduct a series of experiments on multiple datasets to evaluate the performance of TAIP on the widely used MLIPs of SchNet[43] and PaiNN[45], which are invariant and equivariant, respectively. First, we

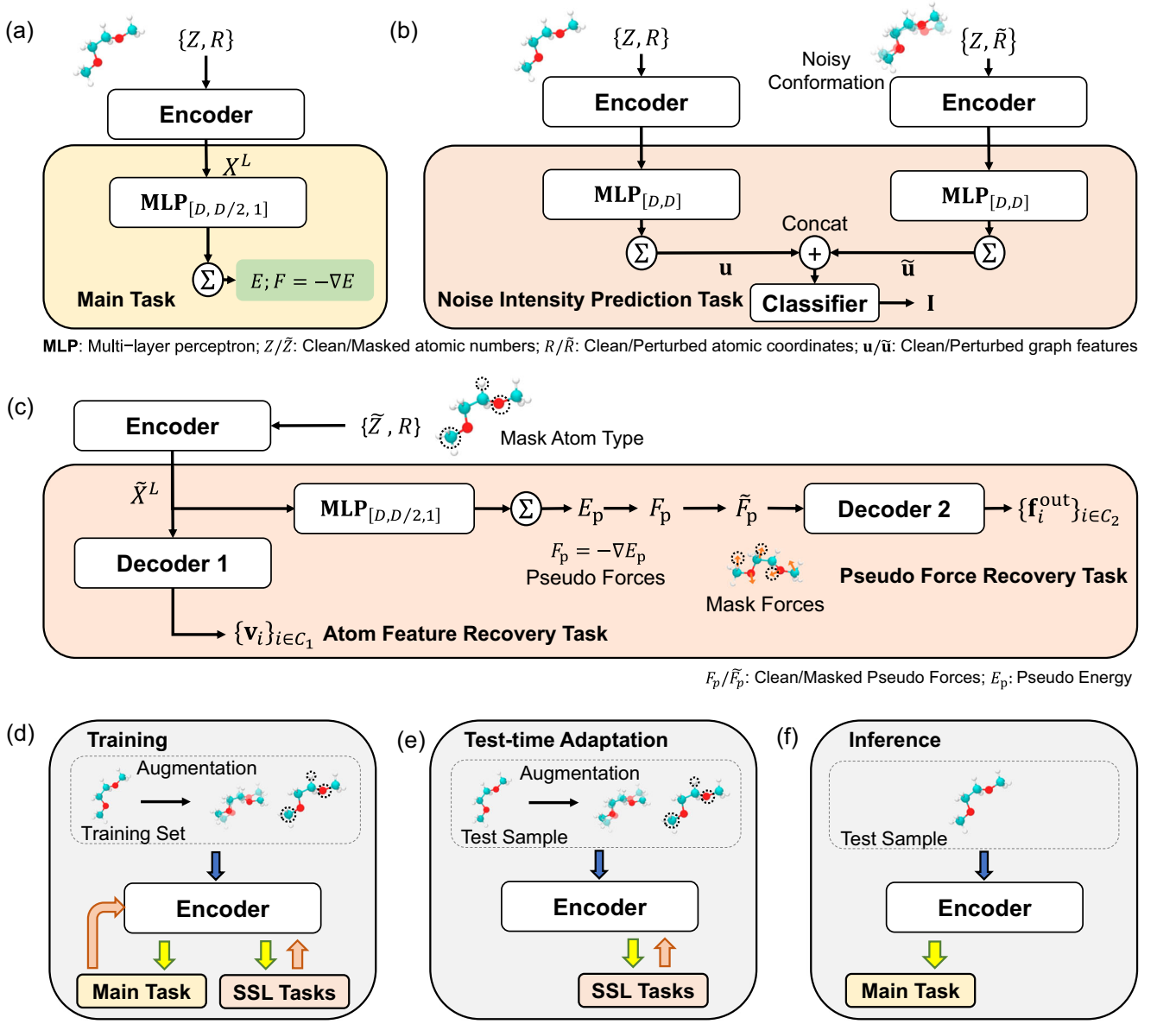

Fig. 1 | Overview. a The main task of energy and force prediction. **b** The noise intensity prediction task. **c** The atom feature recovery task and the pseudo force recovery task. **d**–**f** The three stages of online Test-time Adaptation Interatomic Potential (TAIP): training, test-time adaptation, and inference. The main task and the self-supervised learning (SSL) tasks losses are minimized simultaneously in the training stage. The encoder is updated once per test sample in the test-time adaptation stage by minimizing the SSL tasks losses. The test sample then passes through the updated encoder to predict energy and forces.

demonstrate the superior performance of our method on the MD-17 dataset[48]. Then, we extend our evaluation to the ISO17[43] dataset and complex systems featuring periodic boundary conditions, including water and electrolyte solutions. The test data from these datasets are designed to have a larger distribution shift from the training data, as indicated in Supplementary Fig. S1, and therefore are used to investigate the improvement in generalizability achieved by TAIP. We then perform MD simulations with MLIPs on the aforementioned complex systems to assess the practical utility of TAIP. Moreover, we explore the impact of TAIP on feature distributions through dimensionality reductions using the t-SNE method[49], illustrating the mechanism behind the improvement from the perspective of feature embedding.

MD-17. We evaluate the performance of TAIP on the widely used MD-17 dataset, which contains small organic molecules with reference values of energies and forces generated by ab initio MD simulations. Table 1 compares the mean absolute errors (MAEs) of the predicted energies and forces for different molecules using different models. The results show that the TAIP-based models outperform the corresponding baseline models for all eight molecules, demonstrating the generalization ability of our method.

ISO17. The ISO17 dataset contains trajectories of isomers of $C_7O_2H_{10}$. We consider two scenarios with the ISO17 dataset. In the first scenario (known molecules/unknown conformation), the isomers in the test set are also present in the training set. In the second scenario (unknown molecules/unknown conformation), the test set contains a different subset of isomers. This task is more challenging and is used to evaluate TAIP in the case where training and test data are drawn from different distributions. Table 2 shows that, for energy and force MAEs on the ISO17 dataset, TAIP achieves an average reduction of 40% and 31% in scenario 1 and scenario 2, respectively. We have also studied the performance of TAIP as a function of training size on the ISO17 dataset.

Table 1 | Results on MD17 dataset

		SchNet, a)[43]	SchNet, b)	SchNet-TAIP	PaiNN, a)[45]	PaiNN, b)	PaiNN-TAIP
Aspirin	Energy	0.37	0.425 ± 0.032	**0.322 ± 0.041**	0.159	0.244 ± 0.006	**0.144 ± 0.011**
	Force	1.35	1.017 ± 0.042	**0.853 ± 0.045**	0.376	0.347 ± 0.012	**0.330 ± 0.012**
Benzene	Energy	0.08	0.135 ± 0.007	**0.059 ± 0.019**	0.101	0.161 ± 0.008	**0.067 ± 0.012**
	Force	0.31	0.309 ± 0.017	**0.251 ± 0.013**	0.226	0.205 ± 0.009	**0.186 ± 0.007**
Ethanol	Energy	0.08	0.122 ± 0.005	**0.057 ± 0.020**	0.086	0.114 ± 0.009	**0.061 ± 0.015**
	Force	0.39	0.371 ± 0.015	**0.272 ± 0.035**	0.230	0.191 ± 0.007	**0.180 ± 0.010**
Malonaldehyde	Energy	0.13	0.142 ± 0.010	**0.118 ± 0.011**	0.100	0.118 ± 0.005	**0.091 ± 0.011**
	Force	0.67	0.655 ± 0.020	**0.612 ± 0.023**	0.319	0.301 ± 0.007	**0.297 ± 0.009**
Naphthalene	Energy	0.16	0.224 ± 0.015	**0.110 ± 0.017**	0.113	0.118 ± 0.006	**0.093 ± 0.005**
	Force	0.58	0.481 ± 0.018	**0.318 ± 0.033**	0.079	0.101 ± 0.004	**0.072 ± 0.003**
Salicylic acid	Energy	0.20	0.303 ± 0.019	**0.154 ± 0.026**	0.114	0.138 ± 0.005	**0.105 ± 0.004**
	Force	0.85	0.682 ± 0.023	**0.671 ± 0.036**	0.209	0.250 ± 0.002	**0.193 ± 0.003**
Toluene	Energy	0.12	0.320 ± 0.015	**0.103 ± 0.010**	0.119	0.275 ± 0.007	**0.109 ± 0.003**
	Force	0.57	0.500 ± 0.019	**0.461 ± 0.023**	0.102	0.126 ± 0.005	**0.090 ± 0.004**
Uracil	Energy	0.14	0.142 ± 0.004	**0.109 ± 0.006**	0.104	0.226 ± 0.008	**0.090 ± 0.006**
	Force	0.56	0.571 ± 0.012	**0.493 ± 0.013**	0.143	0.157 ± 0.003	**0.130 ± 0.003**

Energy and force Mean Absolute Errors (MAEs) are reported in units of kcal mol^{-1} and kcal mol^{-1} $Å^{-1}$, respectively. Standard deviations are calculated from five independent experiments. MAEs of online Test-time Adaptation strategy for Interatomic Potentials (TAIP)-based models are consistently less than the corresponding baselines and are shown in bold.
*Results for SchNet a) and PaiNN a) are directly taken from the original papers, and standard deviations are not provided therein. SchNet a) was trained using the L1 loss for energy and forces with a ratio of 1: 100. PaiNN a) was trained using the L2 loss for energy and forces with a ratio of 0.05: 0.95. Meanwhile, baseline models b) and TAIP-based models were trained using the L1 loss for energy and forces with a ratio of 1: 1000, consistent with the models trained for MD simulations.

The results are shown in Supplementary Table S4, where the MAEs decreased with an increasing amount of training data, and TAIP leads to an increase in performance in all settings.

Liquid Water and Ice. Water has complex phase behaviors that pose considerable challenges for computational studies. As shown in Fig. 2a and (b), there exists a substantial difference between the microscopic structures of liquid water and ice. In the liquid state, water molecules form a highly dynamic network through hydrogen bonding. In contrast, water molecules form a stable hexagonal lattice structure in an ice crystal. The structural differences between different phases can lead to a decrease in prediction accuracy. Thus, we train the models only with liquid water, using a training set of 1000 frames and a validation set of 100 frames, and report the test accuracy on randomly sampled 500 liquid water and ice structures from the remaining dataset, respectively. The results of MAE on liquid water (known phase/unknown configuration) and on ice (unknown phase/unknown configuration) are shown in Table 2. The TAIP method reduces errors in force and energy predictions by an average of 40% for unknown configurations across both known and unknown phases.

Electrolyte solutions of different concentrations. We examine TAIP on the electrolyte solution dataset, developed in our previous work[23], to further analyze its compatibility with complex systems. The dataset consists of charged ions and organic solvents, including more elements, stronger electrostatic forces, and more complicated interatomic interactions, thus exhibiting a higher degree of complexity than water. Training MLIPs for such complex systems and generalizing the trained models to different concentrations are both challenging. Here, we train the models on 1 M (mol L^{-1}) electrolyte solutions and test the generalization across configurational space at different concentrations. The atomic configurations of 1 M and 4 M electrolyte solutions are illustrated in Fig. 2c and d, respectively. The MAE results for the test set of 1 M solutions (known concentration/unknown configuration) and 4 M solutions (unknown concentration/unknown configuration) are presented in Table 2, showing that TAIP substantially enhances the precision of energy and force predictions for unknown configurations, irrespective of whether the concentration is known or unknown.

MD simulations. The MAEs of predicted forces and potential energies on fixed datasets are not enough to characterize the performance of MLIPs on long-time MD simulations[15]. During MD simulations, new molecular structures are constantly produced, and the false prediction of the interatomic forces on a single atom may cause the collapse of the entire simulation system, as illustrated in Supplementary Fig. S2. Therefore, we perform MD simulations with and without TAIP on periodic systems (water and electrolyte solutions) to further analyze how TAIP can affect the predicted forces and, thus, the quality of MD simulations.

We select four individual systems for MD simulations, including the liquid water, the hexagonal ice, 1 M, and 4 M electrostatic solutions. For the SchNet baseline and SchNet-TAIP models, the potential energies, kinetic energies, and the maximum absolute force of these trajectories are shown on Fig. 3. As indicated, the simulation for water exhibits undesired force and energy instability after 250 ps, and that for ice collapses at 248 ps. The simulations for electrolyte solutions with concentrations of 1 M and 4 M collapse due to losing several atoms at the step of 258 (~0.1 ps) and of 1843 (~0.9 ps), respectively. In contrast, all the simulations with TAIP can run stably for more than 500 ps. Specifically, the simulation trajectories for ice and 4 M electrolyte solution can be seen in Supplementary Videos S1–S4. We have also performed MD simulations using the PaiNN baseline and PaiNN-TAIP models, where the results of the simulations are provided in Supplementary Fig. S3.

It is evident that the failure of MD simulations is caused by the divergence of predicted potential energies and forces. Instability is a fundamental issue for MLIP-based MD simulations, as the accuracy of MLIPs without TAIP is crucially compromised when encountered with unknown configurations. The inclusion of the TAIP method enhances the overall performance of MILP-based MD simulations when dealing with unknown molecular structures, and it can be suggested as an applicable solution to MD instability.

Table 2 | Results on ISO17, water, and electrolyte datasets

			SchNet, a)[43]	SchNet, b)	SchNet-TAIP	PaiNN, b)	PaiNN-TAIP
ISO17	known molecules/	Energy	0.39	0.323 ± 0.015	**0.291 ± 0.013**	0.322 ± 0.085	**0.111 ± 0.079**
	unknown conformation	Force	1.00	0.872 ± 0.077	**0.554 ± 0.079**	0.263 ± 0.015	**0.202 ± 0.011**
ISO17	unknown molecules/	Energy	2.40	1.973 ± 0.127	**1.130 ± 0.123**	0.927 ± 0.055	**0.662 ± 0.054**
	unknown conformation	Force	2.18	1.955 ± 0.121	**1.783 ± 0.111**	0.895 ± 0.045	**0.684 ± 0.048**
Water	known phase/	Energy	-	0.283 ± 0.027	**0.236 ± 0.023**	0.216 ± 0.034	**0.090 ± 0.017**
	unknown configuration	Force	-	0.076 ± 0.005	**0.066 ± 0.004**	0.027 ± 0.004	**0.021 ± 0.004**
Water	unknown phase/	Energy	-	1.193 ± 0.237	**0.423 ± 0.204**	0.585 ± 0.233	**0.042 ± 0.084**
	unknown configuration	Force	-	0.083 ± 0.010	**0.055 ± 0.005**	0.023 ± 0.003	**0.018 ± 0.003**
Electrolyte	known concentration/	Energy	-	0.263 ± 0.031	**0.203 ± 0.010**	0.238 ± 0.014	**0.199 ± 0.006**
	unknown configuration	Force	-	0.064 ± 0.012	**0.043 ± 0.008**	0.049 ± 0.010	**0.037 ± 0.003**
Electrolyte	unknown concentration/	Energy	-	9.459 ± 0.459	**4.816 ± 0.274**	9.872 ± 0.423	**4.914 ± 0.307**
	unknown configuration	Force	-	0.341 ± 0.043	**0.197 ± 0.032**	0.143 ± 0.025	**0.113 ± 0.012**

These datasets are used to evaluate TAIP in the case of large distribution shift. For ISO17 dataset, energy and force MAEs are reported in units of kcal mol^{-1} and kcal mol^{-1} Å^{-1}, respectively. For water and electrolyte dataset, energy and force MAEs are reported in units of eV and eV Å^{-1}, respectively. Standard deviations are calculated from five independent experiments. MAEs of TAIP-based models are consistently less than the corresponding baselines and are shown in bold.

*Results for SchNet a) are directly taken from the original papers, and standard deviations are not provided therein. SchNet a) was trained using the L1 loss for energy and forces with a ratio of 1: 100. The ISO17 dataset was not evaluated in the original paper of PaiNN. Meanwhile, baseline models b) and TAIP-based models were trained using the L1 loss for energy and forces with a ratio of 1: 1000, consistent with the models trained for MD simulations.

Feature visualization. To visually identify the effect of TAIP. We perform a series of dimensionality reductions using the t-SNE method[49] to map the feature embeddings of the training dataset and the testing dataset on benchmark models and TAIP models. In Fig. 4, we can notice that the feature embeddings of the test dataset in the TAIP models are much closer to that of the training dataset than the feature embeddings in the baseline models. It indicates that the TAIP enables the MLIPs to process unknown molecular structures with less uncertainty, which helps the energies and forces predictions to harness more accuracy and generalizability.

Discussion

One of the primary challenges for MLIPs in practical applications lies in their ability to make accurate predictions on test data that exhibit a distribution shift from the training set. Specifically, a feasible MLIP must maintain sufficient accuracy for the new atomic structures that continuously emerge during simulations. This study demonstrates that our TAIP framework enhances the performance of MLIPs on test datasets, which are designed to exhibit substantial distribution shifts from the training data, without requiring any extra data. The efficacy of TAIP is attributed to self-supervised learning tasks including noise intensity prediction, atom feature recovery, and pseudo force recovery. These tasks are trained concurrently with the main task of energy and force prediction during training. Before inference, the model parameters are refined according to self-supervised learning losses. This approach effectively bridges the domain gap between training and testing datasets, as evidenced by t-SNE visualizations of the feature space. The proposed self-supervised learning tasks are complementary for the improvement of performance, as indicated in the ablation studies in Supplementary Table S5. Beyond improving the accuracy of predicting energy and force, TAIP also enables stable MLIP-based MD simulations, outperforming baseline models that otherwise fail. The experiments across various molecular systems underscore the vast potential of our method.

Methods

Dataset

MD-17. The MD17 dataset is composed of ab initio molecular dynamics trajectories of eight small molecules[48]. Obtained from http://www.sgdml.org/#datasets, we use the training set of 1000, the validation set of 1000, and the test set of 1000 conformations for each small molecule.

ISO17. ISO17 dataset consists of ab initio MD trajectories of 129 isomers whose energies and forces are calculated by Density Functional Theory (DFT)[43]. We adopted the same splitting strategy reported in the original literature[43], using 80% trajectory steps of the 80% isomers for training and validation (totaling 404,000 molecular conformations for training and 4000 for validation). We further set up two test datasets to evaluate the effectiveness of TAIP: (1) The other 20% unseen trajectory steps of the isomers included in the training set (totaling 101,000 conformations) and (2) all of the remaining 20% isomers not included in the training and validation set (totaling 130, 000 conformations).

Liquid water and ice. The liquid water dataset consists of a training set of 1000 configurations, a validation set of 100 configurations, and a test set of 500 configurations of liquid water. In addition to the liquid water dataset, 500 ice configurations in the form of hexagonal ice crystals are sampled to be the other test set. All of the liquid water and ice configurations are generated through the classical MD simulations using LAMMPS software package[50] and SPC/E force field parameter[51]. The MD simulations are set up with a time step of 1 fs using the Nose-Hoover thermostat[52] and the Parrinello-Rahman barostat[53] as the coupling method. The sampling simulation undergoes 10 ns under a canonical (NVT) ensemble at 300K after one 50 ps NVT MD simulation at 300K and one 50 ps isothermal-isobaric (NPT) MD simulation to equilibrate the system at 300K and 1 atm. Each configuration contains 96 molecules or 288 atoms, sampled every 10 ps during the simulations.

The energy and forces for each sampled configuration are calculated by DFT using the cp2k package[54]. The DFT calculations are conducted using PBE exchange-correlation functional[55] with the Projector Augmented-Wave (PAW) pseudo-potential[56]. The DFT-D3 method is used for the vdW-dispersion energy correction[57]. A single gamma k-point is used to sample the Brillouin zone, the cutoff energy for the plane-wave-basis set is set to be 400 eV, and the electronic self-consistency is considered to be achieved if the change of total energy between two steps is smaller than 10^{-6} eV.

Electrolyte solutions. The electrolyte solutions dataset is taken from our previous work[23]. Here, we used the combination of [$Li^+PF_6^-$DME], [$Na^+PF_6^-$DME], [$Li^+Tf_2N^-$DME], [$Na^+Tf_2N^-$DME], [$Li^+PF_6^-$EC+DMC], [$Na^+PF_6^-$EC+DMC], [$Li^+Tf_2N^-$EC+DMC], and [$Na^+Tf_2N^-$EC+DMC] as electrolyte solutions with ionic concentrations of 1 M and 4 M. The training and validation data are randomly selected from the 1 M solutions. The training set contains 1000 samples and the validation set

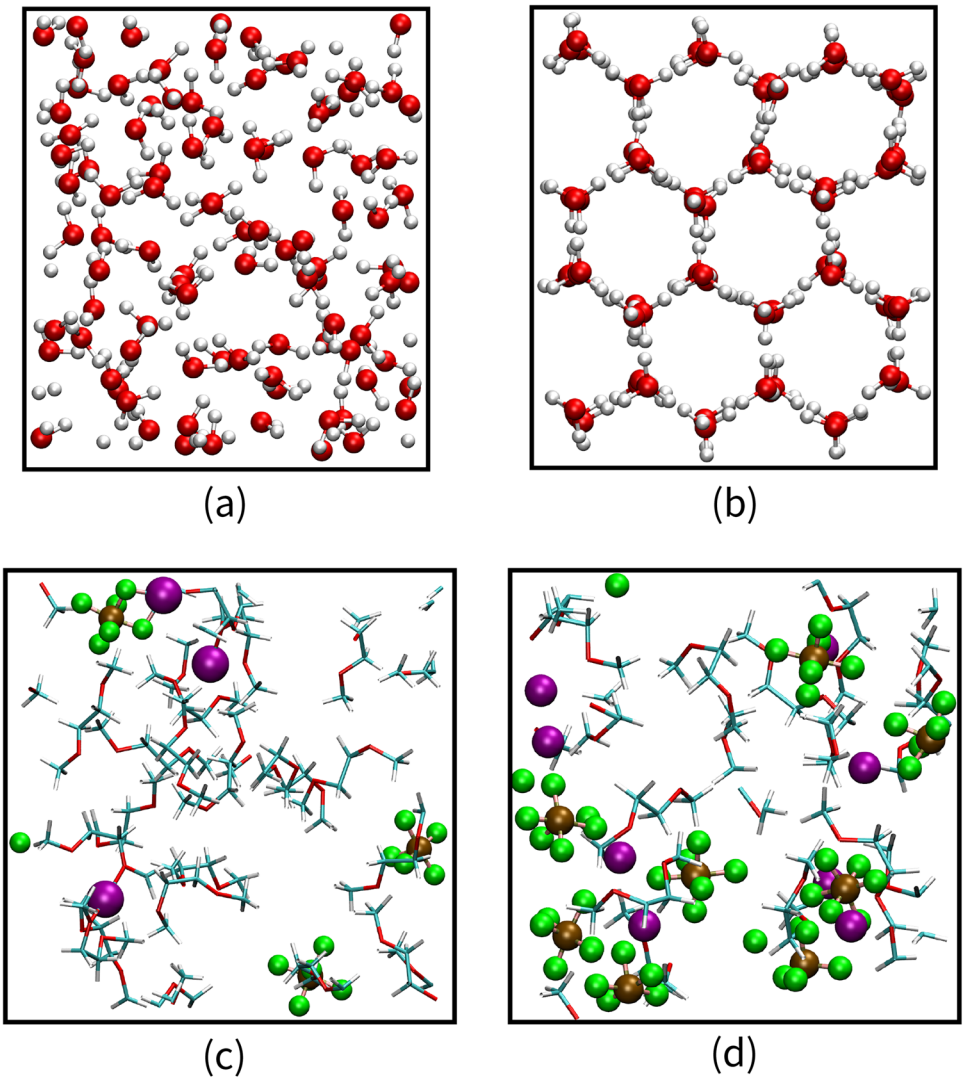

Fig. 2 | Visualization of atomic structures. a Liquid Water. **b** Ice. **c** 1 M (mol L^{-1}) electrolyte solution. **d** 4 M electrolyte solution. The snapshots in (**a**) and (**c**) are taken from the first configuration in their respective training datasets, while the snapshots in (**b**) and (**d**) are taken from the first configuration in their respective test datasets. The electrolyte solution consists of lithium hexafluorophosphate (LiPF6) in 1,2-dimethoxyethane (DME) solvents. Color scheme: H-white; O-red; C-cyan; Li-purple; P-ocher; F-green. The DME solvents are displayed in line style to highlight the difference between 1 M and 4 M solutions.

contains 500 samples. We construct two different test sets to evaluate TAIP: one is 1000 samples randomly sampled from the remaining 1 M solutions and the other is 1000 samples sampled from the 4 M solutions.

Model implementations

Encoder. Encoders are used to transform the input atomic structure information into atom feature vectors. In this work, we use encoders from the invariant model SchNet[43] and the equivariant model PaiNN[45] respectively to validate our method. In an encoder, the feature vector associated with each atom is first initialized using an embedding function $f_{\text{emb}}: \mathbb{N}_+ \mapsto \mathbb{R}^D$ to obtain the initial atom feature:

$$\mathbf{x}_i^0 = f_{\text{emb}}(z_i) \tag{1}$$

where z_i is the atomic number of atom i.

For the encoder of SchNet, continuous-filter convolutional layers are leveraged as message passing layers. The atom features are updated by a message function:

$$\mathbf{x}_i^{l+1} = \sum_{j \in N_i} f_{\text{m}}\left(\mathbf{x}_i^l, \mathbf{x}_j^l, \mathbf{r}_{ij}\right), \tag{2}$$

where l denotes the l^{th} layer, N_i denotes the neighbors of atom i, $\mathbf{x}_i^l \in \mathbb{R}^D$ is the feature of atom i at l^{th} layer and $\mathbf{r}_{ij} = \mathbf{r}_j - \mathbf{r}_i$ is the relative position between atom i and j.

For the encoder of PaiNN, a kind of equivariant message passing strategy is used that maintains separate tracks for scalar and vector features. It uses a similar equation as Equation (2) for scalar features update. Additionally, it also leverages continuous-filter convolutions to update equivariant vector features[45]. Here, we refer to the scalar features as the atom features.

For both SchNet and PaiNN encoders, the atom features within a configuration are updated for L times to obtain $\{\mathbf{x}_i^L\}$. We further define a graph feature for each configuration, which will be used in the noise intensity prediction task, as follows:

$$\mathbf{u} = \sum_i \mathbf{MLP}_{[D,D]}\left(\mathbf{x}_i^L\right), \tag{3}$$

https://doi.org/10.1038/s41467-025-57101-4

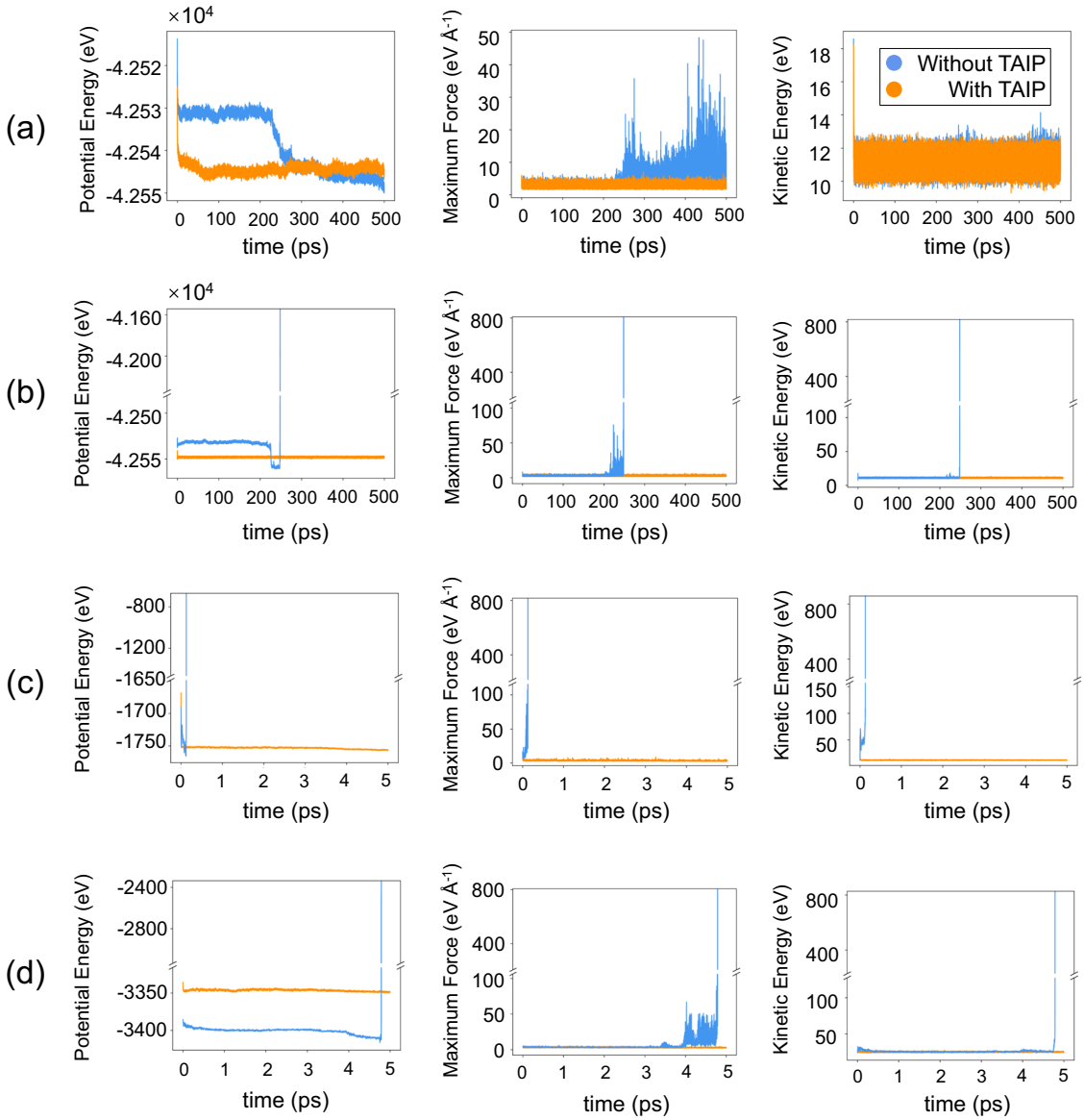

Fig. 3 | Results of molecular dynamics (MD) simulations. a Liquid water. **b** ice. **c** 1 M (mol L^{-1}) electrolyte solutions. **d** 4 M electrolyte solutions. The MD simulations with baseline Machine learning interatomic potentials (MLIPs) (blue) breakdown in a short time, whereas those with online test-time adaptation strategy for interatomic potentials (TAIP) (yellow) are stable throughout the simulation time. Source data are provided as a Source Data file.

where $\mathbf{MLP}_{[D_1,\ldots,D_L]}$ denotes a multilayer perceptron network (MLP) of L layers with layer dimension D_i and SiLU nonlinearity.

In the main task, the updated atom features $\{\mathbf{x}_i^L\}$ are used to compute the potential energy:

$$E = \sum_i \mathbf{MLP}_{[D,D/2,1]}\left(\mathbf{x}_i^L\right). \tag{4}$$

Self-supervised learning tasks. *Noise intensity prediction*. For each configuration, we randomly choose a noise intensity t, which is a positive integer, from a uniform distribution $U(1, T)$. Subsequently, we generate random noises $\varepsilon \in \mathbb{R}^{n\times 3}$ from a Gaussian distribution with a mean of zero and a variance of t^2 to perturb the corresponding coordinates and obtain the perturbed coordinates of all atoms in the configuration $\tilde{R} = R + \varepsilon$. The perturbed configuration $\{Z, \tilde{R}\}$ is fed into the encoder to generate the noisy graph feature $\tilde{\mathbf{u}}$ according to Equation (3). A noiseless graph feature $\mathbf{u}$ is also generated from the original configuration. Then, the noisy and noiseless graph features are concatenated and pass through an MLP network to produce the output logits:

$$\mathbf{I} = \sigma\left(\mathbf{MLP}_{[D,T]}(\mathbf{u}||\tilde{\mathbf{u}}\right), \tag{5}$$

where $\sigma(\cdot)$ is the Softmax function, and $||$ denotes concatenation. The output $\mathbf{I}$ is a vector with dimension T, and its t^{th} element $\mathbf{I}_t$ indicates the likelihood of the applied noisy intensity being t. The corresponding loss function can be formulated as follows:

$$\mathcal{L}_{\text{ni}} = -\sum_{t=1}^{T} \hat{\mathbf{I}}_t \log(\mathbf{I}_t) \tag{6}$$

where $\hat{\mathbf{I}}_t$ is the t^{th} logit of one-hot noise intensity label $\hat{\mathbf{I}}$.

Atom feature recovery. We randomly sample a subset of atoms C_1 and mask their atom types with a special token [MASK] to obtain the

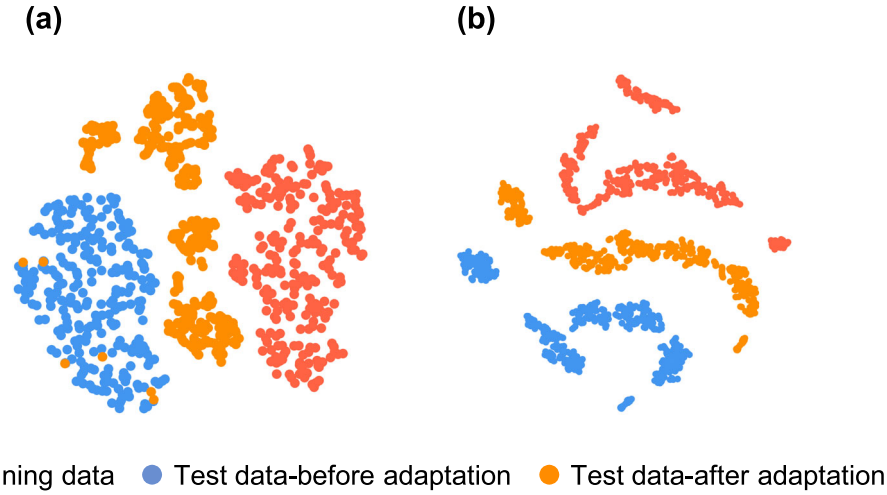

Fig. 4 | Visualization of feature space. a Liquid water (train) and ice (test). **b** 1 M (mol L^{-1}) (train) and 4 M (test) electrolyte solutions. The feature embeddings of test data are closer to those of training data.

masked configuration $\tilde{Z} = \{\tilde{z}_i\}_{i=1}^n$, where

$$\tilde{z}_i = \begin{cases} [\text{MASK}], & i \in C_1 \\ z_i, & i \notin C_1 \end{cases} \tag{7}$$

The masked configuration $\{\tilde{Z}, R\}$ are fed into the encoder to get the masked node features $\tilde{X}^L = \{\tilde{\mathbf{x}}_i^L\}_{i=1}^n$. Then, the masked node features pass through the Decoder 1 to recover the original atom types:

$$\mathbf{v}_i = \mathbf{MLP}_{[D,N_z]}\left(\tilde{\mathbf{x}}_i^L\right), \tag{8}$$

where N_z is the maximum atomic number. The output $\mathbf{v}_i$ is a vector with dimension N_z. The corresponding atom type label is represented using the one-hot encoding $\hat{\mathbf{v}}_i \in \mathbb{R}^{N_z}$. We utilize the cosine error with a power of γ in the loss function, as shown below:

$$\mathcal{L}_{\text{ma}} = \frac{1}{|C_1|}\sum_{i\in C_1}\left(1 - \frac{\mathbf{v}_i^{\mathrm{T}}\cdot\hat{\mathbf{v}}_i}{\|\mathbf{v}_i\|\|\hat{\mathbf{v}}_i\|}\right)^{\gamma}, \gamma \geq 1, \tag{9}$$

which is averaged over all masked atoms. The scaling factor γ is a hyperparameter, and a power of $\gamma > 1$ can down-weight easy sample's contribution in training[58].

Pseudo force recovery. We utilize the masked atom features $\{\tilde{\mathbf{x}}_i^L\}_{i=1}^n$ obtained in the atom feature recovery task to compute the pseudo energy E^{p}:

$$E^{\text{p}} = \sum_i \mathbf{MLP}_{[D,D/2,1]}\left(\tilde{\mathbf{x}}_i^L\right), \tag{10}$$

where the parameters are shared with those in Equation (4). Then we compute the negative gradient of the pseudo energy E^{p} with respect to atomic coordinates R to obtain the pseudo forces $F^{\text{p}} = \{\mathbf{f}_i^{\text{p}}\}_{i=1}^n$.

Then randomly sample a subset of atoms C_2 and mask their pseudo forces with a special token [FMASK] to obtain the masked pseudo forces $\tilde{F}^{\text{p}} = \{\tilde{\mathbf{f}}_i^{\text{p}}\}_{i=1}^n$, where

$$\tilde{\mathbf{f}}_i^{\text{p}} = \begin{cases} [\text{FMASK}], & i \in C_2 \\ \mathbf{f}_i^{\text{p}}, & i \notin C_2 \end{cases} \tag{11}$$

We try to recover the masked pseudo forces F^{p} by feeding $\tilde{F}^{\text{p}}$ into the decoder 2. To utilize the information from neighboring atoms, we employs a continuous-filter convolution layer[43] to reconstruct the original pseudo forces from the masked pseudo forces. The output of decoder 2 $\mathbf{f}_i^{\text{out}}$ is then given by:

$$\mathbf{f}_i^{\text{out}} = \mathbf{MLP}_{[D,3]}\left(\sum_{j\in N_i}\tilde{\mathbf{f}}_j^{\text{p}} \circ W(\mathbf{r}_{\text{i}} - \mathbf{r}_{\text{j}})\right), \tag{12}$$

where N_i denotes neighbors of atom i, $\circ$ represents the element-wise multiplication, and W is a filter-generating network $W : \mathbb{R}^3 \mapsto \mathbb{R}^D$[43].

We utilize the L1 loss as the criterion for reconstructing pseudo force:

$$\mathcal{L}_{\text{mf}} = \frac{1}{|C_2|}\sum_{i\in C_2}\left\|\mathbf{f}_i^{\text{out}} - \mathbf{f}_i^{\text{p}}\right\|. \tag{13}$$

Experimental settings

The TAIP framework is implemented based on PyTorch 1.8.0. The experiments are conducted with NVIDIA GeForce RTX 3090 GPU. The models are trained using the Adam optimizer[59], employing single-GPU training for efficient processing. The hyperparameters are provided in Supplementary Table S1–S3.

Training. We incorporate both energy E and forces $\{\mathbf{f}_i\}$ into the criterion for training:

$$\mathcal{L}_{\text{TAIP}} = |E - \hat{E}|^2 + \frac{\lambda}{n}\sum_{i=1}^n\left\|\left(-\frac{\partial E}{\partial \mathbf{r}_i} - \hat{\mathbf{f}}_i\right)\right\|^2 + \mathcal{L}_{\text{ni}} + \mathcal{L}_{\text{ma}} + \mathcal{L}_{\text{mf}}, \tag{14}$$

where $\hat{E}$ represents the ground-truth energy, n is the number of atoms in each sample, and $\mathbf{r}_i$ represents the coordinate of atom i. The weight $\lambda = 1000$ is used in all experiments in this work. The $\mathcal{L}_{\text{ni}}$, $\mathcal{L}_{\text{ma}}$ and $\mathcal{L}_{\text{mf}}$ are self-supervised learning losses corresponding to noise intensity prediction, masked atom reconstruction and masked pseudo force reconstruction, respectively.

For the periodic systems including water and electrolyte solutions, we employ the atomization energy as the target energy by shifting the original potential energy according to the energy of each single atom:

$$E^{\text{shift}} = E - \sum_{i=1}^n E_i^{\text{ref}} \tag{15}$$

where E_i^{ref} is the potential energy of a single atom i in vacuum.

Test-time adaptation and inference. When the model training is completed, we perform an adaptation operation before inference. For each input test data, we are able to compute the losses associated with three self-supervised learning tasks without the need for labels. Subsequently, we fine-tune the encoder parameters using the gradient descent method:

$$w^{\text{adapted}} = w^{\text{trained}} - \eta_{\text{a}} \frac{\partial(\mathcal{L}_{\text{ni}} + \mathcal{L}_{\text{ma}} + \mathcal{L}_{\text{mf}})}{\partial w^{\text{trained}}}, \tag{16}$$

where w represent a learnable parameter in the encoder, the superscript "trained" and "adapted" denote the parameters before and after adaptation, respectively. The learning rate η_{a} is an adjustable hyperparameter. Note that, instead of minimizing the self-supervised learning losses, the parameters are updated only once during adaptation. Then, the model with the updated encoder parameters is used to make energy and force predictions.

The specific strategy of adaptation will have an impact on the results. For the fixed test datasets, i.e., those in Tables 1 and 2, the adaptation is not permanent. This means that the adapted parameters are not saved, and for each new test data, the adaptation starts from the trained parameters.

For MD simulations, we have tested several test-time adaptation strategies, including simulate with non-saving adaptation, with saved adaptation every simulation step, with saved adaptation every 100 simulation steps, with saved adaptation when the self-supervised learning loss is larger than a certain threshold, and without any adaptation, shown in Supplementary Information Figure S4. Since the stability of an MD simulation can be affected by many random factors, a generally optimal strategy is hard to determine. Finally, we recommend using the simplest non-saving strategy with a learning rate of 10^{-5} or 10^{-6} in MD simulation, as this constantly outperforms the baselines. The results shown in Fig. 3 and Supplementary Fig. S3 adopt the non-saving strategy.

MD simulations

The MD simulations for evaluating the performance of TAIP are conducted using the Atomic Simulation Environment (ASE) Python library[60]. SchNet and PaiNN are used, respectively, as the baseline models to produce the potential energy and interatomic forces.

The initial structures of liquid water, hexagonal ice, and electrolyte solutions are randomly sampled from the corresponding test dataset. For each system, we performed 10 different simulations to ensure the reliability of the experiments. All simulations are set up with a timestep of 0.5 fs under canonical (NVT) ensembles, using the Berendsen thermostat[61] as the temperature coupling method with a coupling temperature of 300 K and a decaying time constant τ of 100 fs. The velocity of each atom is initialized according to the Boltzmann distribution at 300 K. We use the the liquid water training set to train the models for simulations on liquid water and hexagonal ice systems and the 1 M electrolyte solution training set to train the models for simulations on 1 M and 4 M electrolyte solutions.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The MD17 and ISO17 datasets are publicly available (see "Methods"). Other datasets used in this work (water/ice and electrolyte solutions), raw data files of MD simulation trajectories (Fig. 3, Supplementary S3, and Supplementary Fig. S4) are available at[62] and at https://doi.org/10.6084/m9.figshare.27115468. Source data are provided with this paper.

Code availability

The source code for reproducing the findings in this paper is available at[63] and https://github.com/TaoyongCui/TAIP-codes. It is licensed under the `Apache License 2.0`, which allows users to use, modify, and distribute the code freely, provided that proper attribution is given to the original authors. This open source approach improves the reproducibility of our results and facilitates further research in this area.

References

1. Hospital, A., Goñi, J. R., Orozco, M. & Gelpí, J. L. Molecular dynamics simulations: advances and applications. *Adv. Appl. Bioinform. Chem.* **8**, 37–47 (2015).
2. Senftle, T. P. et al. The reaxff reactive force-field: development, applications and future directions. *npj Computational Mater.* **2**, 1–14 (2016).
3. Karplus, M. & Petsko, G. A. Molecular dynamics simulations in biology. *Nature* **347**, 631–639 (1990).
4. Yao, N., Chen, X., Fu, Z.-H. & Zhang, Q. Applying classical, *Ab Initio*, and machine-learning molecular dynamics simulations to the liquid electrolyte for rechargeable batteries **122**, 10970–11021 (2022).
5. Car, R. & Parrinello, M. Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.* **55**, 2471 (1985).
6. Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O. & Walsh, A. Machine learning for molecular and materials science **559**, 547–555 (2018).
7. Noé, F., Tkatchenko, A., Müller, K.-R. & Clementi, C. Machine learning for molecular simulation **71**, 361–390 (2020).
8. Unke, O. T. et al. Machine learning force fields **121**, 10142–10186 (2021).
9. Zhang, L., Wang, H., Car, R. & Weinan, E. Phase diagram of a deep potential water model. *Phys. Rev. Lett.* **126**, 236001 (2021).
10. Niu, H., Bonati, L., Piaggi, P. M. & Parrinello, M. Ab initio phase diagram and nucleation of gallium. *Nat. Commun.* **11**, 2654 (2020).
11. Li, J.-L., Li, Y.-F. & Liu, Z.-P. In situ structure of a mo-doped pt–ni catalyst during electrochemical oxygen reduction resolved from machine learning-based grand canonical global optimization. *JACS Au* **3**, 1162–1175 (2023).
12. Chen, D. et al. Square-pyramidal subsurface oxygen [ag4oag] drives selective ethene epoxidation on silver. *Nature Catalysis* **7**, 536–545 (2024).
13. Su, M., Yang, J.-H., Liu, Z.-P. & Gong, X.-G. Exploring large-lattice-mismatched interfaces with neural network potentials: the case of the cds/cdte heterostructure. *J. Phys. Chem. C.* **126**, 13366–13372 (2022).
14. Zhang, D., Yi, P., Lai, X., Peng, L. & Li, H. Active machine learning model for the dynamic simulation and growth mechanisms of carbon on metal surface. *Nat. Commun.* **15**, 344 (2024).
15. Fu, X. et al. Forces are not Enough: Benchmark and Critical Evaluation for Machine Learning Force Fields with Molecular Simulations. *Trans. Mach. Learn. Res.* (2023).
16. Zhang, Y. et al. Dp-gen: a concurrent learning platform for the generation of reliable deep learning based potential energy models. *Computer Phys. Commun.* **253**, 107206 (2020).
17. Kulichenko, M. et al. Uncertainty-driven dynamics for active learning of interatomic potentials. *Nat. Computational Sci.* **3**, 230–239 (2023).
18. Yuan, X. et al. Active learning to overcome exponential-wall problem for effective structure prediction of chemical-disordered materials. *npj Computational Mater.* **9**, 12 (2023).
19. Zhang, D. et al. DPA-1: Pretraining of attention-based deep potential model for molecular simulation. *arXiv* https://doi.org/10.48550/arXiv.2208.08236 (2022).
20. Zhang, D. et al. DPA-2: a large atomic model as a multi-task learner. *npj Comput. Mater.* **10**, 293 (2024).

21. Wang, Y. et al. Denoise pretraining on nonequilibrium molecules for accurate and transferable neural potentials. *J. Chem. Theory Comput.* **19**, 5077–5087 (2023)
22. Gardner, J. L., Baker, K. T. & Deringer, V. L. Synthetic pre-training for neural-network interatomic potentials. *Mach. Learn.: Sci. Technol.* **5**, 015003 (2024).
23. Cui, T. et al. Geometry-enhanced pretraining on interatomic potentials. *Nature Machine Intelligence* **6**, 428–436 (2024).
24. Liang, J., Hu, D. & Feng, J. *Do We Really Need To Access The Source Data? Source Hypothesis Transfer For Unsupervised Domain Adaptation*, 6028–6039 (PMLR, 2020).
25. Sun, Y. et al. *Test-time Training With Self-supervision For Generalization Under Distribution Shifts*, 9229–9248 (PMLR, 2020).
26. Wang, D. et al. Tent: Fully Test-Time Adaptation by Entropy Minimization. *Int. Conf. Learn. Represent.* (2021).
27. Iwasawa, Y. & Matsuo, Y. Test-time classifier adjustment module for model-agnostic domain generalization. *Adv. Neural Inf. Process. Syst.* **34**, 2427–2440 (2021).
28. Boudiaf, M., Mueller, R., Ben Ayed, I. & Bertinetto, L. *Parameter-free Online Test-time Adaptation*, 8344–8353 (2022).
29. Wang, Q., Fink, O., Van Gool, L. & Dai, D. *Continual Test-time Domain Adaptation*, 7201–7211 (2022).
30. Voigtlaender, P. & Bastian, L. Online Adaptation of Convolutional Neural Networks for Video Object Segmentation. *Proc. of the British Machine Vision Conf.* (2017).
31. Karani, N., Erdil, E., Chaitanya, K. & Konukoglu, E. Test-time adaptable neural networks for robust medical image segmentation. *Med. Image Anal.* **68**, 101907 (2021).
32. Zhang, Y., Borse, S., Cai, H. & Porikli, F. *Auxadapt: Stable And Efficient Test-time Adaptation For Temporally Consistent Video Semantic Segmentation,* 2339–2348 (2022).
33. Shin, I. et al. *Mm-tta: Multi-modal Test-time Adaptation For 3d Semantic Segmentation*, 16928–16937 (2022).
34. Veksler, O. et al. *Test Time Adaptation With Regularized Loss For Weakly Supervised Salient Object Detection*, 7360–7369 (2023).
35. Kim, D., Park, S. & Choo, J. *When Model Meets New Normals: Test-time Adaptation For Unsupervised Time-series Anomaly Detection*, **38**, 13113–13121 (2024).
36. Gao, Z., Yan, S. & He, X. Atta: Anomaly-aware test-time adaptation for out-of-distribution detection in segmentation. *Adv. Neural Inf. Process. Syst.* **36**, 45150–45171 (2023).
37. Segu, M., Schiele, B. & Yu, F. *Darth: Holistic test-time adaptation for multiple object tracking*, 9717–9727 (2023).
38. Behler, J. & Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces **98**, 146401 (2007).
39. Kovács, D. P. et al. Linear atomic cluster expansion force fields for organic molecules: beyond rmse. *J. Chem. Theory Comput.* **17**, 7696–7711 (2021).
40. Chen, C. et al. Accurate force field for molybdenum by machine learning large materials data. *Phys. Rev. Mater.* **1**, 043603 (2017).
41. Bartók, A. P., Payne, M. C., Kondor, R. & Csányi, G. Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons **104**, 136403 (2010).
42. Vandermause, J. et al. On-the-fly active learning of interpretable Bayesian force fields for atomistic rare events. *npj Computational Mater.* **6**, 20 (2020).
43. Schütt, K. et al. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems* **30**, 992–1002 (2017).
44. Batzner, S. et al. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 2453 (2022).
45. Schütt, K., Unke, O. & Gastegger, M. *Equivariant Message Passing For The Prediction Of Tensorial Properties And Molecular Spectra*, 9377–9388 (PMLR, 2021).
46. Batatia, I., Kovacs, D. P., Simm, G., Ortner, C. & Csányi, G. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Adv. Neural Inf. Process. Syst.* **35**, 11423–11436 (2022).
47. Wang, Y. et al. Enhancing geometric representations for molecules with equivariant vector-scalar interactive message passing. *Nat. Commun.* **15**, 313 (2024).
48. Chmiela, S. et al. Machine learning of accurate energy-conserving molecular force fields. *Sci. Adv.* **3**, e1603015 (2017).
49. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Machine Learning Res.* **9**, 2579–2605 (2008).
50. Thompson, A. P. et al. Lammps-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Phys. Commun.* **271**, 108171 (2022).
51. Berendsen, H. J., Grigera, J. R. & Straatsma, T. P. The missing term in effective pair potentials. *J. Phys. Chem.* **91**, 6269–6271 (1987).
52. Hoover, W. G. & Holian, B. L. Kinetic moments method for the canonical ensemble distribution. *Phys. Lett. A* **211**, 253–257 (1996).
53. Parrinello, M. & Rahman, A. Polymorphic transitions in single crystals: a new molecular dynamics method. *J. Appl. Phys.* **52**, 7182–7190 (1981).
54. Kühne, T. D. et al. Cp2k: an electronic structure and molecular dynamics software package-quickstep: Efficient and accurate electronic structure calculations. *J. Chemical Phys.* **152**, 194103(2020).
55. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865 (1996).
56. Blöchl, P. E. Projector augmented-wave method. *Phys. Rev. B* **50**, 17953 (1994).
57. Grimme, S., Antony, J., Ehrlich, S. & Krieg, H. A consistent and accurate ab initio parametrization of density functional dispersion correction (dft-d) for the 94 elements h-pu. *J. Chemical Phys.* **132**, 154104 (2010).
58. Hou, Z. et al. *Graphmae: Self-supervised Masked Graph Autoencoders*, 594–604 (2022).
59. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. *3rd Int. Conf. Learn. Represent., San Diego* (2015).
60. Larsen, A. H. et al. The atomic simulation environment—a Python library for working with atoms. *J. Phys.: Condens. Matter* **29**, 273002 (2017).
61. Berendsen, H. J., Postma, J. V., Van Gunsteren, W. F., DiNola, A. & Haak, J. R. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.* **81**, 3684–3690 (1984).
62. Cui, T. et al. *Datasets and Trajectories for Online Test-time Adaptation for Better Generalization of Interatomic Potentials to Out-of-distribution Data*. Zenodo. https://doi.org/10.6084/m9.figshare.27115468 (2025).
63. Cui, T. et al. *TAIP-codes: version one*. Zenodo. https://zenodo.org/records/14712539 (2025).

Acknowledgements

This work was supported by Shanghai Artificial Intelligence Laboratory. M.S. was partially supported by Shanghai Committee of Science and Technology, China (Grant No. 23QD1400900). T.C. and C.T. did this work during their internship at Shanghai Artificial Intelligence Laboratory.

Author contributions

S.Z. and M.S. conceived the idea and led the research. T.C. developed the codes and trained the models. C.T. performed experiments and analyzes. Y.L. and X.G. contributed technical ideas for datasets and experiments. D.Z. and W.O. contributed technical ideas for self-

supervised methods. T.C., C.T., D.Z., M.S., and S.Z. wrote the paper. All authors discussed the results and reviewed the manuscript.

Competing interests
The Authors declare no competing interests.

Additional information
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-025-57101-4.

Correspondence and requests for materials should be addressed to Mao Su or Shufei Zhang.

Peer review information *Nature Communications* thanks the anonymous reviewer(s) for their contribution to the peer review of this work. A peer review file is available.