

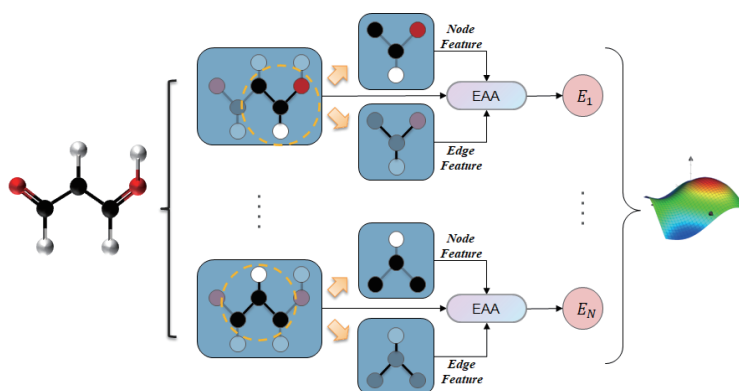
ARTICLE

Molecular Potential Energy Computation via Graph Edge Aggregate Attention Neural Network

Jian Chang^a, Yiming Kuai^{a,b}, Xian Wei^b, Hui Yu^{b,c}, Hai Lan^{b,c*}^a. College of Software, Liaoning Technical University, Huludao 125105, China^b. Fujian Institute of Research on the Structure of Matter, Chinese Academy of Sciences, Fuzhou 362200, China^c. Fujian Science & Technology Innovation Laboratory for Optoelectronic Information of China, Fuzhou 350108, China

(Dated: Received on September 16, 2022; Accepted on December 6, 2022)

Accurate potential energy surface (PES) calculation is the basis of molecular dynamics research. Using deep learning (DL) methods can improve the speed of PES calculation while achieving competitive accuracy to *ab initio* methods. However, the



performance of DL model is extremely sensitive to the distribution of training data. Without sufficient training data, the DL model suffers from overfitting issues that lead to catastrophic performance degradation on unseen samples. To solve this problem, based on the message passing paradigm of graph neural networks, we innovatively propose an edge-aggregate-attention mechanism, which specifies the weight based on node and edge information. Experiments on MD17 and QM9 datasets show that our model not only achieves higher PES calculation accuracy but also has better generalization ability compared with Schnet, which demonstrates that edge-aggregate-attention can better capture the inherent features of equilibrium and non-equilibrium molecular conformations.

Key words: Potential energy surface, Message passing paradigm, Attention mechanism, Graph deep learning

I. INTRODUCTION

The potential energy surface (PES) is the correspondence between the potential energy of a molecular system and its atomic coordinates. The precise construction of molecular PES is an essential tool in physics, chemistry, biology, and materials science research. Ac-

cording to the Born-Oppenheimer approximation, due to the negligible mass of the electrons, the energy of the molecule can be regarded as a function of the nuclear coordinates. That is, given a fixed molecular configuration, the corresponding unique potential energy can be calculated via computational chemistry methods. The mainstream taxonomy groups traditional PES calculation into two categories. One methodology is *ab initio* calculation which is based on first principles, such as density functional theory (DFT) [1] and Hartree-Fock calculations, which is widely used in quantum chem-

* Author to whom correspondence should be addressed. E-mail: lanhai09@fjirsm.ac.cn

istry [2]. *Ab initio* calculation has high accuracy but is computationally expensive, therefore it is limited to large-scale molecules system. The other methodology is empirical potential, that is, numeric simulation of experimental data [3–6], these methods can provide rapid access to PES and calculate millions of atoms in progress. However, the accuracy of empirical potential relies heavily on choosing the appropriate fitting functions which requires prior knowledge and lacks of portability. To tackle this problem, using machine learning algorithms to construct empirical potential has been taking increased efforts to improve the computing accuracy and generalization capacity [7–9]. Among numerous machine learning models, neural network (NN) becomes the first choice for PES calculation due to its powerful nonlinear fitting ability. Most research works focus on designing an effective molecular descriptor for the input of NN. Behler and Parrinello proposed a generalized neural network framework in which a symmetric function was designed to handle the input with different sizes [10], and this method achieved *ab initio* accuracy while it was several orders of magnitude faster. With further exploration of invariance regarding the particular molecule permutation group, Guo *et al.* proposed PIP-NN method [11] which used the permutation invariant polynomial (PIP) [12] as the input of the neural network and accurately reproduced the analytical potential energy surfaces for $\text{H} + \text{H}_2$ and $\text{Cl} + \text{H}_2$ systems. To reduce the number of polynomial inputs in PIP-NN, fundamental invariant neural network (FINN) [13, 14] introduced an efficient representation of molecular permutation symmetry and broadened the application range for larger molecular systems. Deep potential (DP) [15], a simple deep neural network representation of PES, eliminated *ad hoc* approximation of the above work by taking each atom into a single sub-network while respecting the symmetry of the system. Embedded atom neural network (EANN) [16] replaced the scalar embedded atom density in empirical embedded atom method (EAM) [17] with a Gaussian-type orbital based density vector which was sent into NN to obtain atomic energy. Gaussian moment neural network [18] used an extendable invariant local molecular descriptor constructed from geometric moments as input of NN to get a high accuracy and efficiency model.

From the perspective of AI developers and data scientists, this hand-crafted input of NN is less intuitive and violates the end-to-end paradigm of deep

learning. Therefore, graph is adopted to encode the molecules and graph neural network (GNN) is used as the backbone to follow the data-driven manner [19–22]. Deep tensor neural network (DTNN) [23] treats a molecule as a graph, and the molecular structure is encoded with an inter-atomic distance matrix and an atomic number vector so that the model can capture the interaction between atoms in a pairwise manner. A general framework named message passing neural networks (MPNN) [24] is distilled by reformulating existing models to exploit extra variations. Substantial refinement has been made on the basis of MPNN to improve the performance while reducing the model size and inference time, such as polarizable atom interaction neural network (PAINN) [25], (REANN) [26], and SpookyNet [27]. On the other hand, following the success of attention mechanism in natural language processing (NLP) and computer vision (CV), graph attention network [28] introduced the attention mechanism into GNN, which made it possible to distinguish the importance of neighbors and its own nodes during feature aggregation, rather than the averaging representation of adjacent nodes in MPNN scheme. Many works involved graph attention mechanism for molecular representation and demonstrated that the attention mechanism could effectively extract the nonlocal intramolecular interactions [29–32]. However, in the graph attention mechanism of previous research works, only node features that represent atoms were aggregated and edge features that represent bonds were neglected, which made the interaction between atoms could not be captured and updated during the feature extraction and impeded the complete representation of molecule.

Therefore, we propose a novel graph attention mechanism with a complete molecular graph, named edge-aggregate-attention (EAA), which characterizes the precise atomic environment by incorporating global node features and captures the inherent interactions between atoms by involving edge information iteratively. In this way, we are capable of embedding the core interactions among nuclei and electrons into the molecular graph efficiently. It is demonstrated our model not only achieves state-of-the-art performances on PES calculation for molecular dynamic simulation without a significant impact on the calculation speed, but also shows promising potential comparable with other current deep learning approaches on molecular properties prediction tasks.

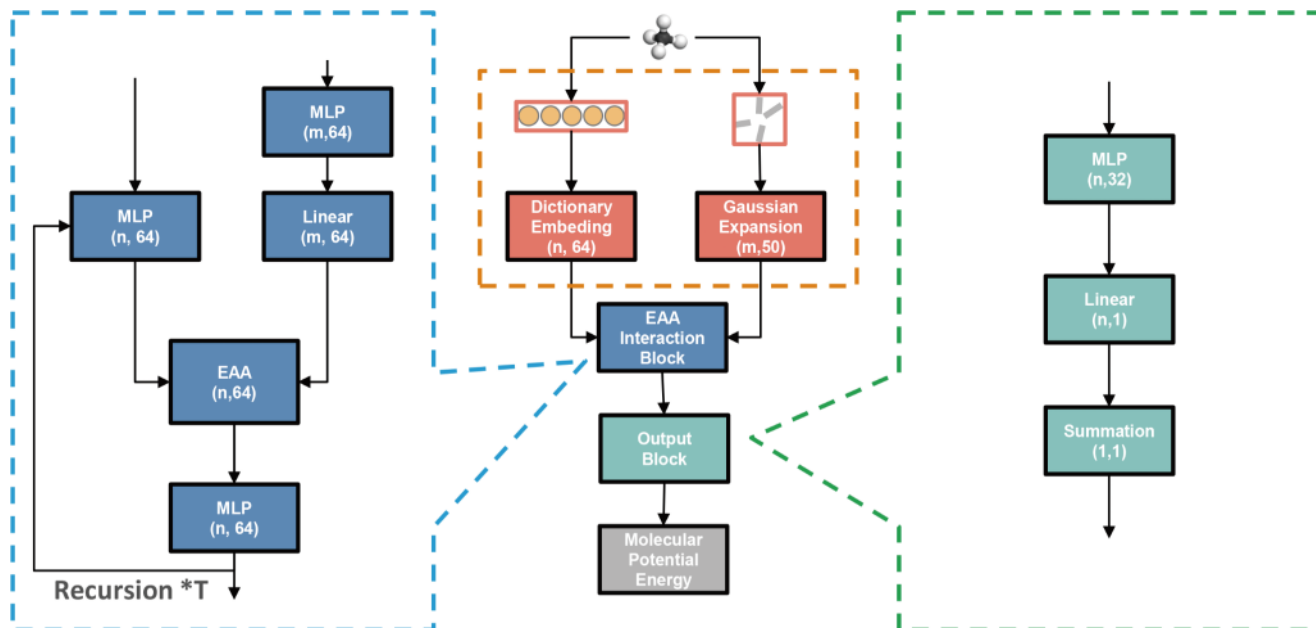


FIG. 1 Framework of our model. MPNN is used as backbone (middle). The framework consists of feature embedding block (orange dotted box), EAA interaction block (blue dotted box), output block (green dotted box).

II. METHODOLOGY

In this section, we describe the framework of EAA model and use MPNN as our backbone. As shown in FIG. 1, our network framework is comprised of feature embedding block, EAA interaction block, and output block. We use the same model training scheme as that in Schnet [33]. For the training of energies and forces of molecular dynamics trajectories, we use a combined L1 loss which includes the total energy E as well as forces F_i .

$$L1 = |\hat{E} - E| + \rho \frac{1}{N} \sum_{i=1}^N \left| F_i - \left(-\frac{\partial E}{\partial r_i} \right) \right| \quad (1)$$

where $\hat{E}$ denotes the ground truth of molecular potential energy; E denotes the potential energy calculated by deep learning model; N denotes the number of atoms in the molecule; $\mathbf{r}$ denotes the three-dimensional coordinates of atom; and ρ is a trade-off hyperparameter between energy and force loss. In our experiments, ρ is set to 100.

A. Feature embedding block

Graph is an ideal representation for non-euclidean data, which can capture the dependencies and interrelationships between various entities. A graph can be described as a set of V vertices (nodes) and a set of E edges (links). Using these two sets, a graph is denoted

as $G = (V, E)$.

There are undirected graph, directed graph, heterogeneous graph, dynamic graph, and attributed graph for different tasks. Undirected graph is the most widely used data structure in which edges are assumed to have equal weights from two directions. The tasks based on graph can be divided into three types: node level task, edge level task, graph level task, which focus on node classification, link prediction, and graph classification respectively. PES calculation [10, 23] can be regarded as a node regression task by embedding every atom in the molecule into the node feature to compute atomic contribution to the total energy.

Specifically, we treat every atom in the molecule as a node of graph, then embed the atomic number of the i -th atom in the molecule to the vector h_i as a node feature by a learnable dictionary. Meanwhile, Euclidean distance between atoms within a molecule is calculated as edge feature e^{ij} . And e^{ij} is expanded by a set of Gaussian kernel function [33] to generate a high dimensional vector that represents the different properties of interactions at different distances.

$$\begin{aligned} \mathbf{e}_{\text{gs}}^{ij} &= \mathbf{G}(\mathbf{e}^{ij}) \\ &= \left[\exp \left(-\frac{[e^{ij} - (u_{\min} + k\Delta u)]}{2d^2} \right) \right]_{0 \leq k \leq u_{\max}/\Delta u} \quad (2) \end{aligned}$$

where Δu is the offset value of the Gaussian functions, and d is the width value of the Gaussian function. In

our experiment, $u_{\min}$ and $u_{\max}$ are set to be 0 Å and 5 Å, k is 50. With Gaussian kernel function, the edge feature can be non-linearly mapped into a feature space where effective learning and generalization can be guaranteed.

B. EAA interaction block

MPNN is one of the most promising methods for predicting molecular chemical properties [24]. Therefore we adopt MPNN as the framework of EAA interaction block and elaborate a graph attention mechanism to replace the aggregation operation (*e.g.*, sum, average, or max of the neighborhood information) in the original MPNN algorithm [23, 24, 33, 34], which confine the ability of the model to capture spatial information by preventing identical weight assignment to different neighbors during the aggregation. As shown in FIG. 1, in EAA interaction block, Gaussian edge feature e_{gs}^{ij} is embedded with a multilayer perceptron (MLP) layer which consists of a linear layer and a soft-plus activation function $\sigma(x) = \ln(e^x + 1) - \ln 2$ employed for non-linearity, followed by a linear transformation. And node feature h_i is embedded with a MLP layer only, then both h_i and e_{gs}^{ij} are sent into EAA module and updated through T times interaction steps, each interaction consists of a message function $\mathbf{M}_t$ and an update function $\mathbf{U}_t$.

$$\begin{aligned} \mathbf{m}_i^{t+1} &= \sum_{j \in N(i)} \mathbf{M}_t(h_i^t, h_j^t, e_{\text{gs}}^{ij}) \\ \mathbf{h}_i^{t+1} &= \mathbf{U}_t(\mathbf{m}_i^{t+1}, h_j^t) \end{aligned} \quad (3)$$

As illustrated in FIG. 2, to capture the interactions between atoms by integrating edge information and assigning importance factors to different neighbor nodes, we propose a novel graph attention mechanism EAA by designing the message function $\mathbf{M}_t$ and the update function $\mathbf{U}_t$, which can be written as below:

$$\begin{aligned} \mathbf{M}_t(h_i^t, h_j^t, e_{\text{gs}}^{ij}) &= \text{MLP}(A \otimes h_j^t \otimes e_{\text{gs}}^{ij}) \\ \mathbf{U}_t(m_i^{t+1}, h_j^t) &= m_i^{t+1} \oplus h_j^t \end{aligned} \quad (4)$$

where $\otimes$ denotes the element-wise multiplication, $\oplus$ denotes the element-wise addition, A denotes the attention matrix.

C. Output block

The key role in EAA is the generation of attention

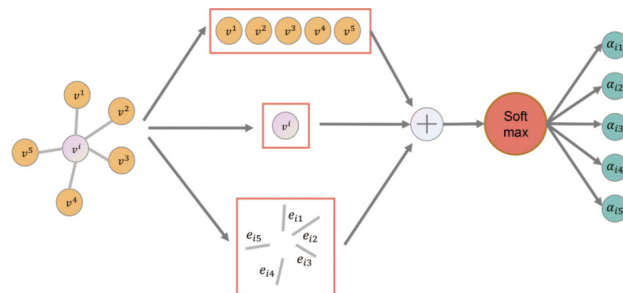


FIG. 2 Schematic diagram of EAA.

matrix A . Inspired by the multi-head attention mechanism in graph attention network [28], for each attention head A^h , three trainable weight matrices W_{node} , W_{nb} , $W_{\text{edge}} \in F^{R \times R}$ are employed for linear projection of node, neighbor node and edges respectively. We extend the attention mechanism to capture the high-dimensional information of edges. Each element A_{ij}^h in a single attention head A^h can be calculated by:

$$a_{ij}^h = W_{\text{node}}(h_i) \oplus W_{\text{node}}(h_j) \oplus W_{\text{edge}}(e_{\text{gs}}) \quad (5)$$

Then a softmax function is applied for nonlinearity:

$$A^h = \text{Softmax}(A^h) = \frac{\exp(a_{ij}^h)}{\sum_{k \in N(i)} \exp(a_{ik}^h)} \quad (6)$$

For each head, we calculate the $h_{i,h}^{t+1}$ by message function $\mathbf{M}_t$ and update function $\mathbf{U}_t$, and the output $h_{i,h}^{t+1}$ of all heads are subsequently concatenated into $h_{i,h}^{t+1} \in R^{hF}$, then a MLP is employed to reduce the dimension of $h_{i,h}^{t+1}$ back to F . The EAA interaction block can be stacked for better expressivity. In our work, a 6-layer stack is set as default if not specifically mentioned.

For each atomic node in the graph of molecule, the output block aims to simulate the energy contribution E_i to the total potential energy [10]. The molecular potential energy can be expressed by summing all the individual atomic potential energy as follows:

$$\begin{aligned} E &= \sum_{i=1}^N E_i \\ E_i &= \text{Linear}(\text{MLP}(h_i^T)) \end{aligned} \quad (7)$$

where N is the number of atoms in the molecule, and h_i^T denotes the output of the EAA interaction block after T times rounds of message passing. A MLP layer is applied and followed by a linear transformation to fit

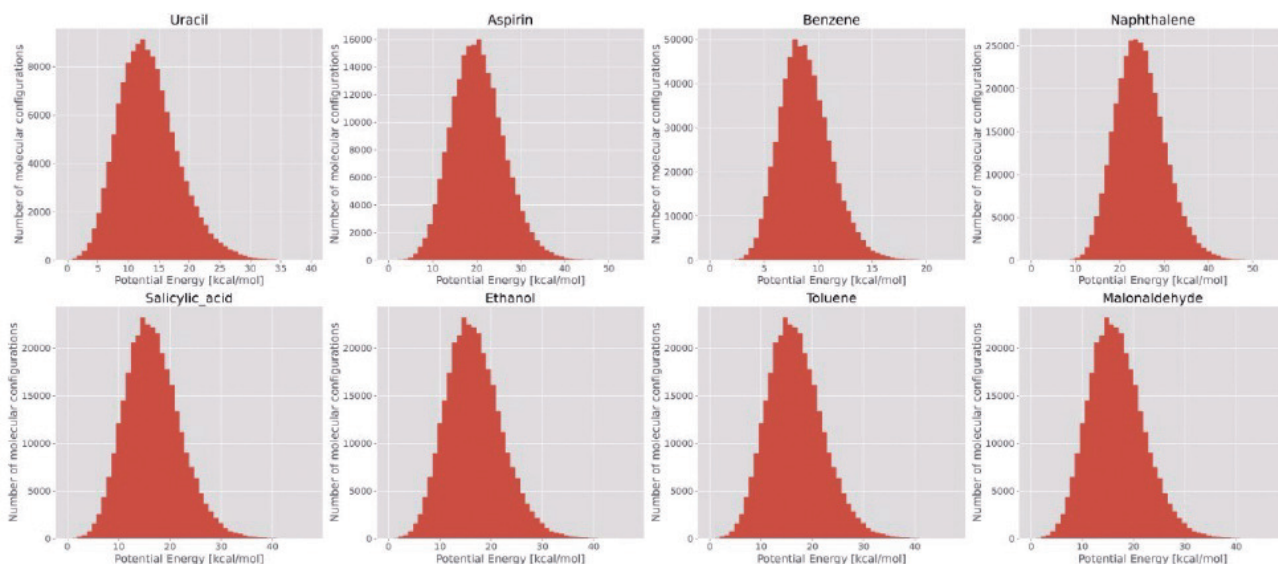


FIG. 3 Potential energy distribution of eight organic molecules in MD17 dataset.

the contribution from each atom to obtain the total molecular potential energy.

III. RESULTS AND DISCUSSION

A. Experimental configuration

The hardware environment used in the experiment is Core i7-6700 CPU@3.40 GHz, GeForce GTX1080Ti GPU. The algorithm is developed by Python 3.8, and PyTorch 1.10 is used as the backend. We use ADAM [35] optimizer for model training with an initial learning rate 1×10^{-4} which is adjusted by cosine decay schedule [36]. The mini-batch size is set to 100, and the number of message passing times T is set to 6.

B. Dataset

In order to verify the effectiveness of the proposed algorithm for PES calculation in this work, we choose two public datasets, MD17 [37] and QM9 [38, 39], to conduct experiments. The molecular dynamics (MD17) datasets provide both the energy and atomic forces of eight small organic molecules (benzene, toluene, malonaldehyde, salicylic acid, aspirin, ethanol, uracil, naphthalene) as well as the corresponding atom coordinates of the thermalized system. The datasets range in size from 1.5×10^6 to nearly 1.0×10^6 conformational geometries. All trajectories are calculated at temperature of 500 K with a resolution of 0.5 fs. The ground truth data are computed via molecular dynamics simulations using PBE+vdW-TS electronic structure method. In our experiment, all eight kinds of organic molecules are

trained with the goal of providing highly accurate energy and force predictions. The energy distributions of eight molecules are shown in FIG. 3, where the horizontal axis is the potential energy value and the vertical axis is the amount of data. Another dataset QM9 contains 133,885 stable small organic molecules with up to 7 heavy atoms (C, O, N, F) and up to 29 atoms in total including H. The ground truth is 12 quantum chemistry target properties, which are calculated by *ab initio* quantum chemistry methods at the B3LYP/6-31G(2df,p) level.

C. Comparisons with previous works

We randomly choose 1000 and 50000 molecular configurations as training set and the remaining data as test set in MD17 dataset respectively. Mean absolute error (MAE) between predictions and ground truth per molecule is applied as evaluation metrics. Firstly, we compare our model with four benchmark methods (Schnet [33], EANN [16], GMNN [18], Physnet [40]). The EAA model is trained on $N=1000$ and $N=50000$ samples. The results of experiments, as shown in Table I, demonstrate EAA performs best or at least equal to other models on 9 out of 16 targets with 50000 training samples and on half targets with 1000 training samples. Notably, for energy calculation, EAA achieves state-of-the-art performances on all the organic molecules except Benzene with 50000 training samples. When the number of training samples is decreased to 1000, EAA still outperforms others on most molecules datasets, which fully demonstrates the effectiveness of

TABLE I Mean absolute errors for energy and force prediction (in kcal/mol and kcal/(mol·Å)) for different training samples on MD17 dataset. GM-sNN [18], EANN [16], SchNet [33], PhysNet [40] results are compared. EANN does not provide results on benzene molecule. The lowest error is emphasized in bold.

Organic molecule		Mean absolute error							
		$N=1000$				$N=50000$			
		Schnet	EANN	GM-sNN	EAA	Schnet	PhysNet	GM-sNN	EAA
Benzene	Energy	0.08		0.08	0.06	0.07	0.07	0.07	0.10
	Force	0.31		0.21	0.34	0.17	0.15	0.14	0.17
Toluene	Energy	0.12	0.11	0.15	0.11	0.09	0.10	0.14	0.07
	Force	0.57	0.38	0.34	0.42	0.09	0.03	0.10	0.05
Malonaldehyde	Energy	0.13	0.14	0.12	0.10	0.08	0.07	0.12	0.07
	Force	0.66	0.62	0.45	0.40	0.08	0.04	0.08	0.04
Salicylic acid	Energy	0.20	0.14	0.19	0.16	0.10	0.11	0.19	0.09
	Force	0.85	0.51	0.49	0.52	0.19	0.04	0.14	0.09
Aspirin	Energy	0.37	0.33	0.38	0.32	0.12	0.12	0.19	0.11
	Force	1.35	0.99	0.69	0.99	0.33	0.06	0.26	0.14
Ethanol	Energy	0.08	0.10	0.10	0.07	0.05	0.05	0.05	0.04
	Force	0.39	0.47	0.33	0.22	0.05	0.03	0.06	0.02
Uracil	Energy	0.14	0.11	0.12	0.11	0.10	0.10	0.10	0.10
	Force	0.56	0.35	0.33	0.36	0.11	0.03	0.07	0.04
Naphthalene	Energy	0.16	0.12	0.17	0.15	0.11	0.12	0.13	0.08
	Force	0.58	0.27	0.36	0.36	0.11	0.04	0.13	0.05

EAA in the calculation of PES.

D. Generalizability study

Secondly, to fully explore the generalizability of our algorithm on small datasets, we choose two molecules, malonaldehyde with the most configurations in the dataset (993, 237) and uracil with the fewest configurations (133, 770) from MD17 dataset, and further decrease the number of training samples to (200, 400, 600, 800, 1000), while keeping test set fixed. Schnet, as the backbone of our model, is involved for comparison to prove the generalizability of EAA mechanism. The MAE evaluated on the test set is shown in Table II. The results demonstrate that EAA mechanism can significantly improve the generalization ability of the model for PES calculation. In both malonaldehyde and uracil datasets, EAA consistently outperforms the backbone model with different sizes of training sets, which reveals the outstanding generalizability of our model. These results corroborate that EAA mechanism can capture the inherent principle of interaction in the molecule with fewer training samples and provide an effective tool for PES calculation.

In addition, we visualize the fitting errors on malonaldehyde and uracil datasets to reveal the correlation

TABLE II Results with different training samples N . Scores are given by mean absolute errors of energy (kcal/mol) and force (kcal/(mol·Å)) prediction.

Organic molecule	N	Mean absolute error			
		Schnet		EAA	
		Energy	Force	Energy	Force
Malonaldehyde	200	0.48	1.25	0.28	1.44
	400	0.27	0.92	0.18	0.89
	600	0.21	0.78	0.14	0.66
	800	0.17	0.71	0.12	0.52
	1000	0.13	0.66	0.10	0.40
Uracil	200	0.40	1.45	0.20	1.12
	400	0.36	1.09	0.15	0.58
	600	0.28	0.89	0.13	0.50
	800	0.20	0.74	0.12	0.42
	1000	0.14	0.56	0.11	0.36

between the model performance and the data distribution. As shown in FIG. 4, even in the interval with rare data samples, most outputs of Schnet diffuse beyond ± 0.25 kcal/mol error range, while our model can still control the error of outputs in the range. This result provides a succinct explanation that our algorithm can still maintain exceptional performance without sufficient training data.

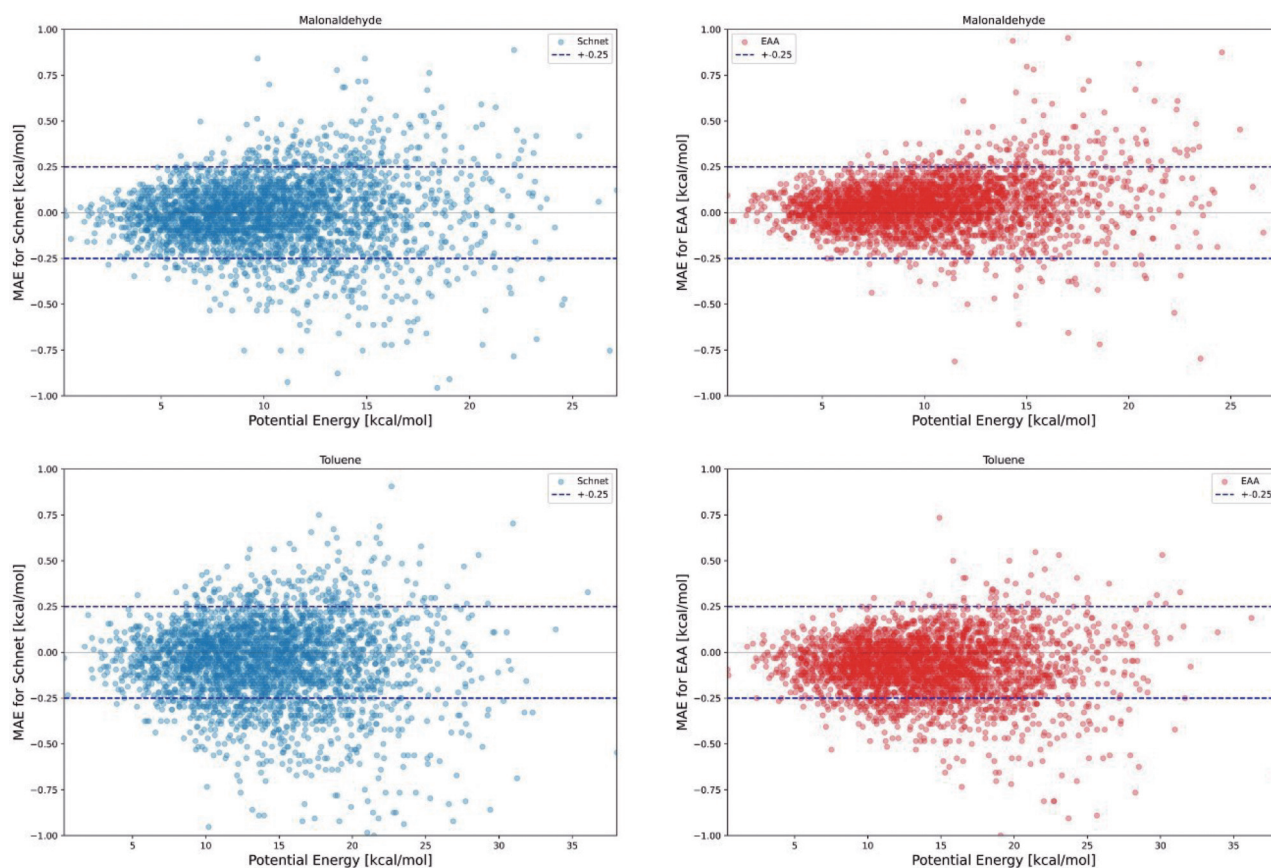


FIG. 4 The fitting errors for the malonaldehyde (upper), uracil (bottom) testing dataset with Schnet (left) and EAA (right). The horizontal axis denotes the distribution of ground truth energies of a given molecule dataset.

TABLE III MAE results with different attention mechanisms. Scores are given by mean absolute errors of energy (kcal/mol) and force (kcal/(mol·Å)) prediction.

Attention on node	Attention on edge	Malonaldehyde		Uracil	
		Energy	Force	Energy	Force
W/O	W/O	0.13	0.66	0.14	0.56
W	W/O	0.14	0.49	0.12	0.44
W	W	0.10	0.44	0.11	0.36

E. Ablation study

Thirdly, to further analyze which component takes effect in our model, we carry out ablation experiments. The MAE results are shown in Table III. We notice that, compared to traditional GNN, applying attention mechanism on the node substantially enhances the performance on force prediction but provides a trivial improvement on energy prediction, while involving the attention on the edge can remarkably boost the accuracy on both tasks. The experiment results confirm that neglecting the edge features will restrain the generalizability of the model. Therefore, we speculate that the intrinsic relationship captured by EAA mechanism is

more proximate to the first principles.

Furthermore, we vary the number of heads in the EAA block to investigate the influence of the width and depth of the model. As shown in FIG. 5(a), we enlarge the model width by increasing the number of heads. For large dataset (malonaldehyde), the performance improves, then soon saturates; while there is no substantial improvement for small dataset (uracil). Meanwhile, FIG. 5(b) shows the variation with model depth by changing the number of EAA layers can give rise to better performance on both datasets. However, a trade-off should be considered because stacking too many layers will consume more computational resources and training time.

F. Predict other quantum chemical properties

To evaluate the model performance in predicting other molecular quantum chemistry properties, we compare our work with other mainstreaming approaches on QM9. The dataset is split into 110,000 molecules for training, 10,000 for validation, and the rest for test. The descriptions of 12 quantum chemical properties and ex-

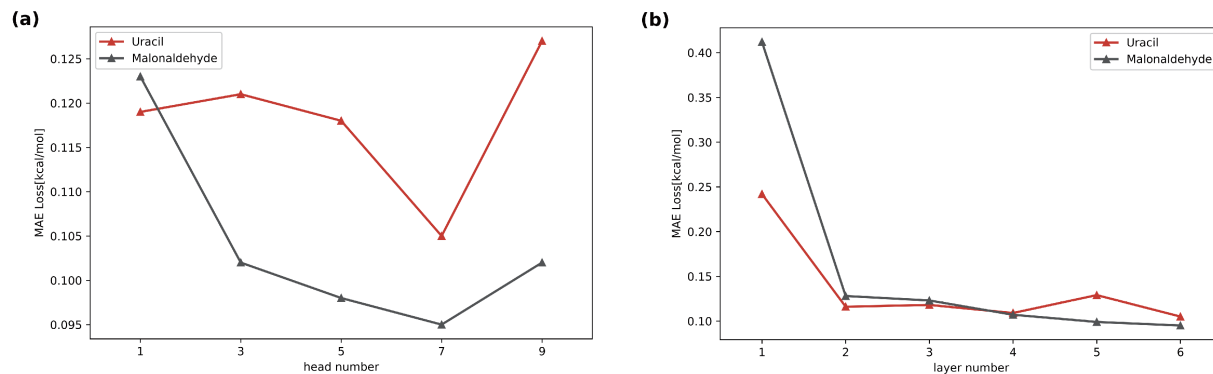


FIG. 5 (a) Performance comparison of models with different head numbers. (b) Performance comparison of models with different layer numbers.

TABLE IV Mean absolute error per molecule of predictions for different target properties of the QM9 dataset using 110,000 training examples. The lowest error is emphasized in bold. Provably powerful graph networks (PPGN) [19], SchNet [33] and enn-s2s [24] results are compared.

Parameters	Description	Unit	PPGN	SchNet	enn-s2s	EAA
ϵ_{homo}	Energy of highest occupied molecular orbital	meV	40	41	43	33
ϵ_{lumo}	Energy of lowest unoccupied molecular orbital	meV	33	34	37	27
$\Delta\epsilon$	Difference between LUMO and HOMO	meV	60	63	69	54
ZPVE	Dipole moment	meV	3.12	1.7	1.5	1.6
μ	Dipole moment	Debye	0.047	0.033	0.030	0.042
α	Isotropic polarizability	Bohr ³	0.131	0.235	0.092	0.086
$\langle R^2 \rangle$	Electronic spatial extent	Bohr ²	0.592	0.073	0.180	0.241
U_0	Internal energy at 0 K	meV	37	14	19	12
U	Internal energy at 298.15 K	meV	37	19	19	15
H	Enthalpy at 298.15 K	meV	36	14	17	14
G	Free energy at 298.15 K	meV	36	14	19	14
C_v	Heat capacity at 298.15 K	cal·mol ⁻¹ ·K ⁻¹	0.055	0.033	0.040	0.030

perimental results in QM9 are shown in Table IV. Compared with all the other approaches, our model has achieved a noteworthy improvement on the accuracy of ϵ_{homo} , ϵ_{lumo} , $\Delta\epsilon$, α , U_0 , U , and C_v (description shown in Table IV). This exhibits the outstanding capability of the EAA for the prediction of molecular chemical properties in thermal equilibrium.

IV. CONCLUSION AND OUTLOOK

In this work, we proposed EAA for PES calculation which could better capture the inherent feature of molecule. According to the experiment results, our approach can improve both accuracy and generalizability. It provides an essential tool and a new perspective for computational chemistry. In future work, we will aim to use different attention mechanisms to design explainable deep learning architectures for molecular representation. Our work will be of great value in the cross-field of data science and chemistry.

V. ACKNOWLEDGMENTS

This work is supported by Fujian Science & Technology Innovation Laboratory for Optoelectronic Information of China (No.2021ZZ120).

- [1] W. Kohn and L. J. Sham, *Phys. Rev.* **140**, A1133 (1965).
- [2] R. Car and M. Parrinello, *Phys. Rev. Lett.* **55**, 2471 (1985).
- [3] F. H. Stillinger and T. A. Weber, *Phys. Rev. B* **31**, 5262 (1985).
- [4] M. S. Daw and M. I. Baskes, *Phys. Rev. B* **29**, 6443 (1984).
- [5] A. D. MacKerell Jr., D. Bashford, M. Bellott, R. L. Dunbrack Jr., J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E.

- Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiórkiewicz-Kucze-
ra, D. Yin, and M. Karplus, *J. Phys. Chem. B* **102**,
3586 (1998).
- [6] A. C. Van Duin, S. Dasgupta, F. Lorant, and W. A. Goddard, *J. Phys. Chem. A* **105**, 9396 (2001).
- [7] B. Jiang, J. Li, and H. Guo, *J. Phys. Chem. Lett.* **11**, 5120 (2020).
- [8] S. Manzhos and T. Carrington Jr., *Chem. Rev.* **121**, 10187 (2020).
- [9] P. L. Kang, C. Shang, and Z. P. Liu, *Acc. Chem. Res.* **53**, 2119 (2020).
- [10] J. Behler and M. Parrinello, *Phys. Rev. Lett.* **98**, 146401 (2007).
- [11] B. Jiang and H. Guo, *J. Chem. Phys.* **139**, 054112 (2013).
- [12] Z. Xie and J. M. Bowman, *J. Chem. Theory Comput.* **6**, 26 (2010).
- [13] R. Chen, K. Shao, B. Fu, and D. H. Zhang, *J. Chem. Phys.* **152**, 204307 (2020).
- [14] L. Xiaoxiao, C. Shang, L. Li, R. Chen, B. Fu, X. Xu, and D. Zhang, *Nat. Commun.* **13**, 4427 (2022).
- [15] J. Han, L. Zhang, R. Car, and E. Weinan, *Commun. Comput. Phys.* **23**, 629 (2018).
- [16] Y. Zhang, C. Hu, and B. Jiang, *J. Phys. Chem. Lett.* **10**, 4962 (2019).
- [17] S. Foiles, M. Baskes, and M. S. Daw, *Phys. Rev. B* **33**, 7983 (1986).
- [18] V. Zaverkin and J. Kästner, *J. Chem. Theory Comput.* **16**, 5410 (2020).
- [19] H. Maron, H. Ben-Hamu, H. Serviansky, and Y. Lipman, in *Advances in Neural Information Processing Systems 32*, Vancouver, BC, Canada, (2019).
- [20] K. Liu, X. Sun, L. Jia, J. Ma, H. Xing, J. Wu, H. Gao, Y. Sun, F. Boulnois, and J. Fan, *Int. J. Mol. Sci.* **20**, 3389 (2019).
- [21] W. Jin, R. Barzilay, and T. Jaakkola, *International Conference on Machine Learning*, PMLR, 2323–2332 (2018).
- [22] T. Nguyen, H. Le, T. P. Quinn, T. Nguyen, T. D. Le, and S. Venkatesh, *Bioinformatics* **37**, 1140 (2021).
- [23] K. T. Schütt, F. Arbabzadah, S. Chmiela, K. R. Müller, and A. Tkatchenko, *Nat. Commun.* **8**, 1 (2017).
- [24] W. Jin, R. Barzilay, and T. Jaakkola, in *Proceedings of the 35th International Conference on Machine Learning*, Long Beach, California, USA, (2018).
- [25] K. Schütt, O. Unke, and M. Gastegger, *International Conference on Machine Learning*, PMLR, 9377–9388 (2021).
- [26] Y. Zhang, J. Xia, and B. Jiang, *J. Chem. Phys.* **156**, 114801 (2022).
- [27] O. T. Unke, S. Chmiela, M. Gastegger, K. T. Schütt, H. E. Sauceda, and K. R. Müller, *Nat. Commun.* **12**, 1 (2021).
- [28] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, *International Conference on Learning Representations*, (2018).
- [29] Z. Xiong, D. Wang, X. Liu, F. Zhong, X. Wan, X. Li, Z. Li, X. Luo, K. Chen, H. Jiang, and M. Zheng, *J. Med. Chem.* **63**, 8749 (2020).
- [30] C. W. Coley, W. Jin, L. Rogers, T. F. Jamison, T. S. Jaakkola, W. H. Green, R. Barzilay, and K. F. Jensen, *Chem. Sci.* **10**, 370 (2019).
- [31] A. Y. T. Wang, S. K. Kauwe, R. J. Murdock, and T. D. Sparks, *Npj Comput. Mater.* **7**, 1 (2021).
- [32] S. Y. Louis, Y. Zhao, A. Nasiri, X. Wang, Y. Song, F. Liu, and J. Hu, *Phys. Chem. Chem. Phys.* **22**, 18141 (2020).
- [33] K. Schütt, P. J. Kindermans, H. E. Sauceda Felix, S. Chmiela, A. Tkatchenko, and K. R. Müller, in *Advances in Neural Information Processing Systems 30*, Long Beach, CA, USA ((2017)).
- [34] K. T. Schütt, H. E. Sauceda, P. J. Kindermans, A. Tkatchenko, and K. R. Müller, *J. Chem. Phys.* **148**, 241722 (2018).
- [35] D. P. Kingma and J. Ba, in *3rd International Conference on Learning Representations*, San Diego, CA, USA, (2015).
- [36] I. Loshchilov and F. Hutter, in *5th International Conference on Learning Representations*, Toulon, France, (2017).
- [37] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K. R. Müller, *Sci. Adv.* **3**, e1603015 (2017).
- [38] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. Von Lilienfeld, *Sci. Data* **1**, 1 (2014).
- [39] L. Ruddigkeit, R. Van Deursen, L. C. Blum, and J. L. Reymond, *J. Chem. Inf. Model.* **52**, 2864 (2012).
- [40] O. T. Unke and M. Meuwly, *J. Chem. Theory Comput.* **15**, 3678 (2019).