



Deep contrastive learning of molecular conformation for efficient property prediction

Received: 20 March 2023

Accepted: 31 October 2023

Published online: 04 December 2023

 Check for updatesYang Jeong Park ^{1,2,3}, HyunGi Kim¹, Jeonghee Jo^{1,2} & Sungroh Yoon ^{1,2,4} ✉

Data-driven deep learning algorithms provide accurate prediction of high-level quantum-chemical molecular properties. However, their inputs must be constrained to the same quantum-chemical level of geometric relaxation as the training dataset, limiting their flexibility. Adopting alternative cost-effective conformation generative methods introduces domain-shift problems, deteriorating prediction accuracy. Here we propose a deep contrastive learning-based domain-adaptation method called Local Atomic environment Contrastive Learning (LACL). LACL learns to alleviate the disparities in distribution between the two geometric conformations by comparing different conformation-generation methods. We found that LACL forms a domain-agnostic latent space that encapsulates the semantics of an atom's local atomic environment. LACL achieves quantum-chemical accuracy while circumventing the geometric relaxation bottleneck and could enable future application scenarios such as inverse molecular engineering and large-scale screening. Our approach is also generalizable from small organic molecules to long chains of biological and pharmacological molecules.

Recent advances of machine learning (ML)-based optimization methods, such as reinforcement learning^{1,2}, active learning³ and deep generative models^{4,5}, have attracted research interest for inverse material design and drug discovery. To quickly predict the quantum-chemical properties of unknown molecules with lower computational costs in these applications, graph neural networks (GNNs) have emerged as a popular and successful model^{6–12}. In GNNs, atoms are typically represented as nodes in the molecular graph, with edges connecting all atoms within a certain distance from the atom's position. By exchanging and updating messages containing information about the local two-body or three-body interactions between atoms across multiple layers, GNN models have shown competitive performance in predicting various properties^{7–9} such as dipole moment, bandgap, internal energy and other properties. These techniques have also been

effective in calculating higher-level properties such as solubility and protein docking^{13–15}.

To train ML models effectively, high-quality datasets have been released, such as the QM9 dataset¹⁶ consisting of 134,000 small organic molecules, with all properties calculated using density functional theory (DFT) with the B3LYP (refs. 17,18)/6-31G(2df,p) (refs. 19–22) quantum-chemistry level. In large-scale inference scenarios such as high-throughput screening, preparing the input molecular geometry by DFT, which is time-consuming and expensive to converge, acts as a bottleneck in using the trained model. Conformations calculated with computationally efficient Merck molecular force field (MMFF)²³ optimization methods or ML-based conformation generative models^{24–29} can be considered as alternatives. However, in this case, the ML model suffers from the domain shift that occurs as it deviates from the

¹Department of Electrical and Computer Engineering, Seoul National University, Seoul, Republic of Korea. ²Institute of New Media and Communications, Seoul National University, Seoul, Republic of Korea. ³Department of Nuclear Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA. ⁴Interdisciplinary Program in Artificial Intelligence, Seoul National University, Seoul, Republic of Korea. ✉e-mail: parkyj@mit.edu; sryoon@snu.ac.kr

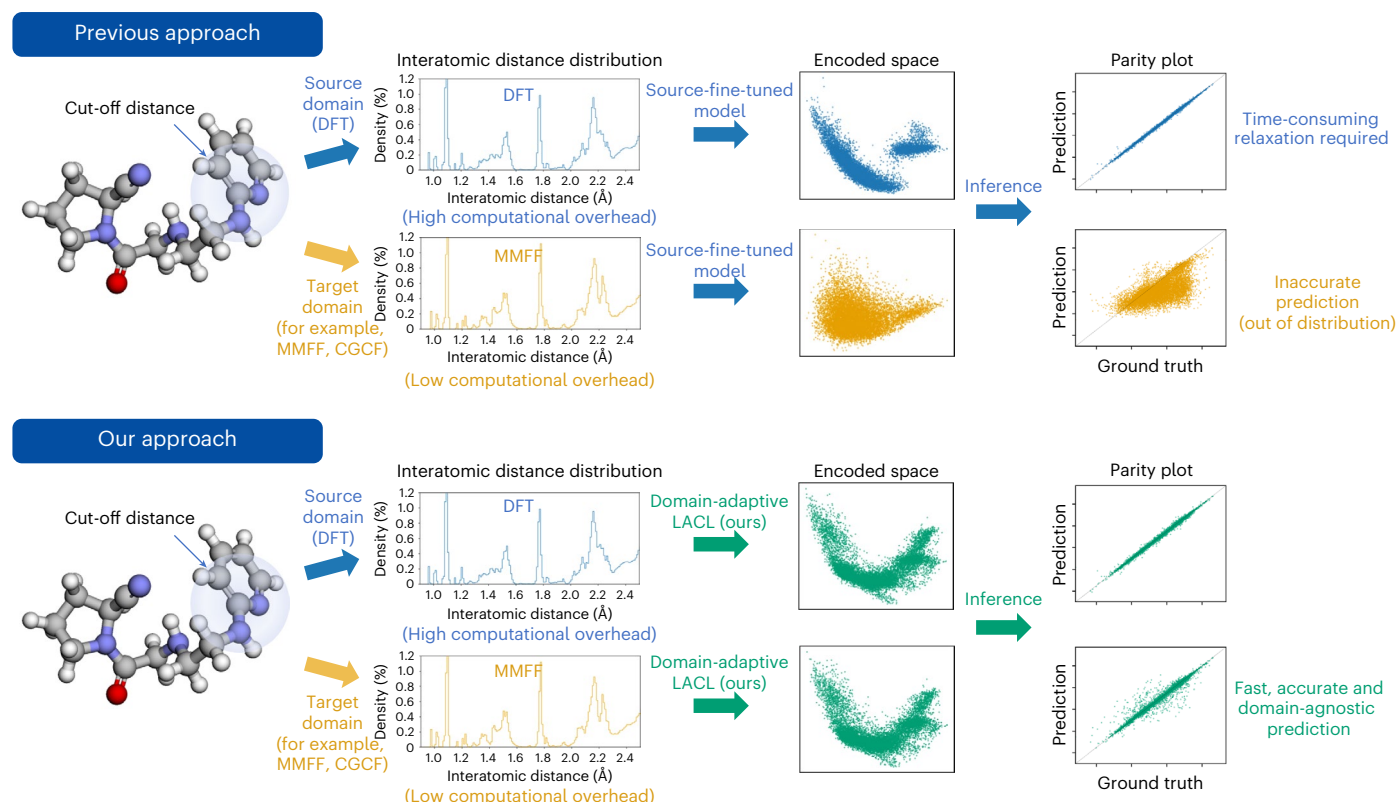


Fig. 1 | Comparison of molecular prediction methodologies between the previous approach and our proposed method. Illustration of the molecular property prediction process. Atoms within a specified cut-off radius are considered neighbors. The hydrogen, carbon, nitrogen and oxygen atoms in molecules are indicated as white, gray, purple and red, respectively. The ‘interatomic distance distribution’ visualizes Euclidean distances for modeling interatomic interactions that depend on the geometric domain. The ‘encoded space’ indicates the hidden representation encoded by each approach. In previous methods, a mismatch between two domains in the encoded space

means a decrease in performance in the target domain. In our method, the hidden expressions match in both domains. The ‘parity plot’ provides the performance of the model by comparing the final prediction inferred from each approach with the ground truth. Previous methods are trained on only the source domain, so during inference, they must perform time-consuming geometric relaxation to match the input data to the source domain, or suffer domain shifts. Our approach is trained by contrasting the source and target domains, thus bypassing computational bottlenecks of geometric relaxation by performing domain-agnostic predictions.

distribution of the previously learned training data computed by DFT, as illustrated in Fig. 1.

In this study, we introduce a Local Atomic environment representation learning model based on deep Contrastive Learning (LACL), specifically designed to tackle the domain-shift problem in molecular data. Our model captures the similarity between molecular data using a computationally efficient geometric relaxation method and DFT molecular geometry data. In this way, LACL leverages the full potential of quantum-chemical data and bypasses the computational bottleneck associated with ab initio geometric relaxation. We validate the domain-adaptation performance of our model against multiple baselines using the QM9 and QMugs³⁰ molecular property prediction benchmarks. LACL accurately predicts molecular properties from low-fidelity geometries, reducing computational costs and inference time while maintaining quantum-chemical accuracy.

In unsupervised domain adaptation³¹, the term ‘source domain’ is the dataset or domain that provides the initial knowledge or information for building an ML model while the term ‘target domain’ is the dataset or domain to which you want to apply the trained model. Unlike the source domain, the target domain may have different characteristics, distributions or data structures. Here we define the term ‘geometric domain’ as the statistical distribution of molecular geometric conformation, including its interatomic distances or triplet angles, generated by certain methods. In this study, we consider the conformations calculated by ab initio calculation methods, which contain the initial knowledge present in the existing benchmark

data, as the source domain. In addition, we regard the conformations obtained from computationally efficient force fields or ML-based conformation-generation models as the target domain. The main goal is to bridge the gap between the source and target domains, allowing the model to generalize its learned knowledge from the source domain to make accurate predictions in the target domain, despite domain shifts.

Figure 2 shows an overview of our model, LACL. To capture subtle differences between two geometric domains, we explicitly model three-body interactions by modifying an atomistic line graph neural network (ALIGNN)¹¹ model which utilizes line graph framework³². Details for encoding graph representations of our model are written in ‘Molecular graph building’ in Methods. Our proposed contrastive learning method compares augmentations of local atomic environments, represented by nodes, rather than augmentations of the entire molecule. LACL is developed based on the bootstrapped graph latents (BGRL)³³ framework, which is a contrastive learning method for graphs that is relatively free from the limitations of other contrastive learning methodologies, such as the need for multiple strategies to select the correct negative pair and large batch sizes. This is an advantage given the large computational memory occupied by the edge features of the molecular line graph. LACL is trained end to end through the entire pipeline by simultaneously minimizing both the BGRL loss and the target property prediction loss to prevent collapse. This training strategy allows for an efficient way to learn molecular graph representations for predicting properties from different views of molecules.

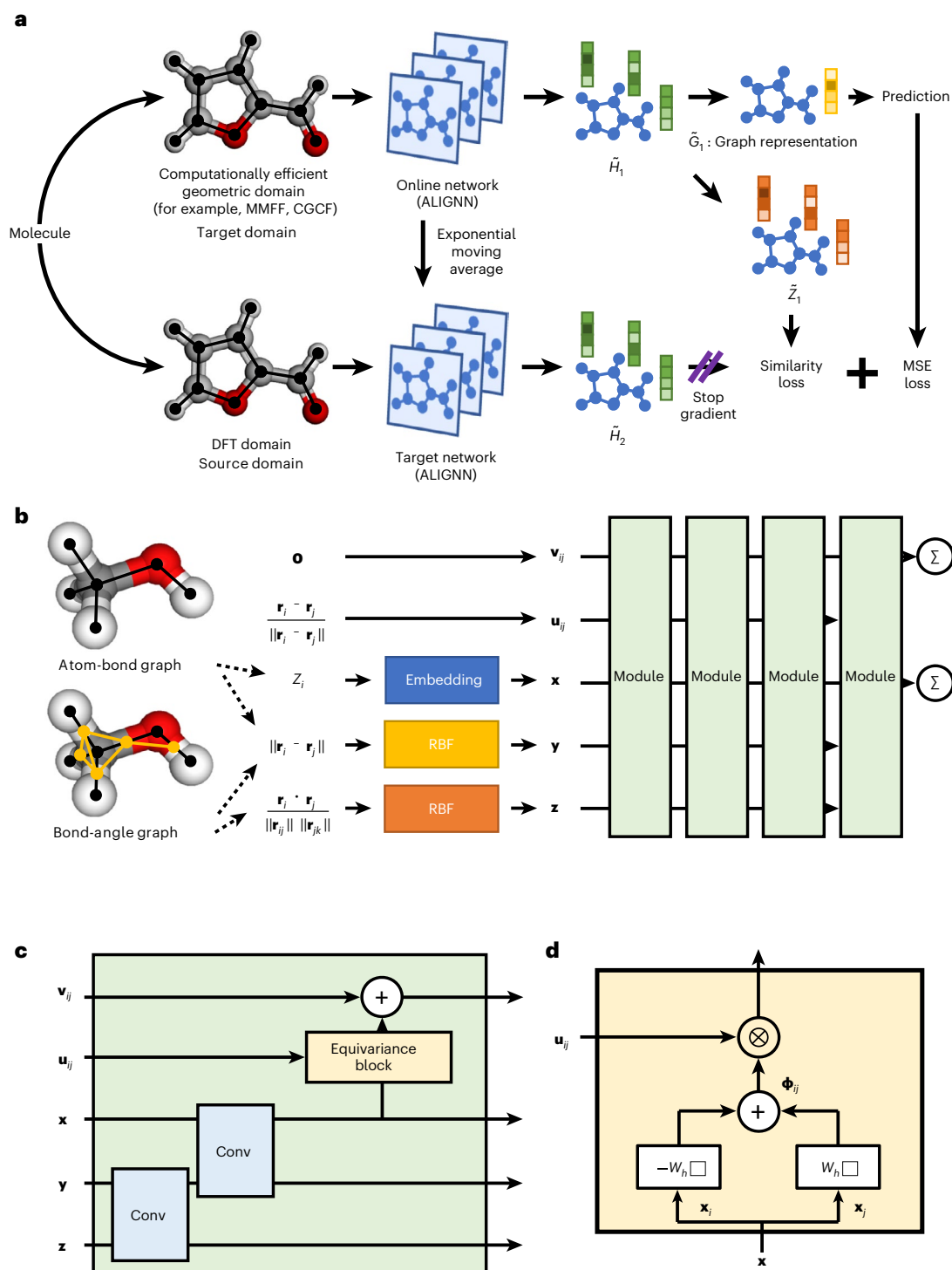


Fig. 2 | Overall LACL schematic. **a**, LACL aligns the source and target representation while inferring the target molecular property. The molecules in the figure show an example of generating two graph augmentations of the same molecule using two different conformation-generation methods. The hidden representations of the two graphs, \tilde{H}_1 and \tilde{H}_2 , are aligned through projection and used for final property prediction through pooling. \tilde{Z}_1 indicates online nonlinear projection of \tilde{H}_1 and \tilde{G}_1 indicates graph-level representation. **b**, Modified ALIGNN as an encoder of the LACL model. The example molecule shows a traditional molecular graph as well as a line graph that defines atomic bonds as nodes and the angles of atomic triplets as edges. The atomic number passes through an embedding layer and is converted into an atom feature \mathbf{x} . The bond feature \mathbf{y} and angle features \mathbf{z} are calculated by applying the RBF to the interatomic distance and angle, respectively. The latent vector representation \mathbf{v}_{ij} between atom i and atom j is learned through the training process using the initial zero vector \mathbf{o} as

input. The unit vector \mathbf{u}_{ij} of the displacement vector \mathbf{r}_{ij} is input into the module to compute the latent vector representation. The hydrogen, carbon and oxygen atoms in molecules are indicated in white, gray and red, respectively. **c**, The bond features are updated by applying graph convolution (Conv) to the bond-angle graph, and the atom features are updated sequentially by applying graph convolution to the atom-bond graph. The updated atom features are then utilized to calculate the latent vector representation. **d**, The latent vector representation is updated from atom features and unit vectors of displacement vectors. The atom feature of the source node and the atom feature of the target node are each processed by linear transformation and added. W_h is the weight matrix for atom features. A latent vector representation is defined using the outer product (\otimes) of the obtained vector Φ_{ij} and the unit vector. The vector properties are predicted as the sum (Σ) of the final latent vector representations.

To highlight the benefits of using the contrastive learning architecture to address the domain-adaptation problem, we compared the LACL model with the original source-fine-tuned ALIGNN¹¹ model. Both models were trained on the QM9 benchmark dataset and the QMugs dataset³⁰, which encompasses 665,000 biologically and pharmacologically relevant molecules. From the QMugs dataset, we selected a total of 68,206 conformations with less than or equal to 20 heavy atoms (QMugs20), each representing the lowest energy conformation.

To explore the impact of the molecular geometric domain on the learning of a properties prediction model, we carefully chose two geometric domains. For comparison with the DFT geometry data distribution, we chose a deep learning-based molecular conformation-generation model, conditional graph continuous-flow conformation-generation method (CGCF-confgen)³⁴ as well as MMFF relaxation. For details of explanation and implementation on obtaining data for each domain, see 'Data preparation and geometric domain distribution' in Methods. We define the terms 'DFT domain', 'MMFF domain' and 'CGCF domain' to indicate the molecular geometry data obtained from each method as DFT, MMFF and CGCF-confgen, respectively.

We attempted to explore the prediction performance of our model on various properties among both datasets (Supplementary Tables 1 and 2). Source fine-tune, indicating the ALIGNN model trained on DFT geometric domain conformations, results in substantial prediction errors when inferring the target domain's conformations, as the data distributions between the two geometries are substantially different. However, we can consider a scenario of giving up the advantages of an information-rich source domain to avoid domain-shift problems and training solely in the target domain. The target-fine-tune ALIGNN model, which is based on the same type of geometry for both the training and testing sets (MMFF geometric domain), shows a higher likelihood of model generalization and accurate predictions on the test set. Here LACL demonstrates its capability to leverage information from the DFT geometric domain to enhance predictions of MMFF geometric domain conformations. This improvement is meaningful as it suggests the potential to achieve quantum-chemical accuracy (less than 1 kcal mol⁻¹ error) with only MMFF-level relaxation without additional optimization. These results provide an opportunity to find an optimal conformation-generation method between accuracy and computational efficiency. More investigation of the prediction performance of LACL according to the type of source or target domain is described in Supplementary Sections 1 and 2.

We visualized the domain-shift problem that the model faces when receiving geometry input through a parity plot (Extended Data Fig. 1a). The LACL model can substantially reduce the difference between the two geometry inputs. Our model further improves the mean absolute error (MAE), indicating that it has properly learned the semantics of the local atomic environment. However, the degree of improvement varies depending on the property being predicted. Properties that are sensitive to electronic characteristics, such as bandgap, show a greater improvement compared with energy properties such as internal energy at 0 K (U0) and free energy at 298.15 K (G). This is consistent with the fact that these energy properties of neutral species at ground state are less sensitive to molecular geometric information. In the case of dipole moments, where molecular geometry is an important factor, not only is there a large difference in prediction performance between models trained on the different geometric domains but also there is a marginal degree of improvement through contrastive learning. Overall, these results demonstrate the importance of carefully considering the properties being predicted and the appropriate data distribution when training ML models for molecular prediction tasks.

To demonstrate the computational efficiency of our model, we measured the runtime required to compute the properties of molecules as the number of heavy atoms increases (Extended Data Fig. 1b). The details of comparison are described in 'Computational efficiency of LACL' in Methods. As expected, our model is much faster at predicting

properties because it skips the computationally expensive DFT relaxation step. The gap in runtime becomes larger as the size of the molecule increases due to the cubic complexity of DFT. These benefits can be further enhanced by utilizing graphics processing unit (GPU)-based distributed computation in large-scale screening scenarios especially when using deep learning-based models such as CGCF-confgen.

We evaluated the generalization ability of our trained model for open and compact conformers (Supplementary Fig. 4 and Supplementary Table 4). We utilized the test set of QMugs20 to generate compact conformers and open conformers. We selected 12 molecules among them and we unfolded the molecules to 180° to obtain open conformers. Thus, while compact conformers exist in the original dataset, open conformers can be considered newly synthesized data. Even considering the small number of test molecules, the results align well with the trends observed with the previous 1,706 test molecules and, overall, LACL shows superior prediction performance. Particularly noteworthy is its robust performance in open conformers, obtained by manipulating the original data. This quantitative experiment suggests that our research direction in finding domain-agnostic representations may expand to more complex systems such as proteins and polypeptides.

To investigate the meaning of the learned local atomic environments, that is node-level embedding, we used *t*-distributed stochastic neighbor embedding (*t*-SNE)³⁵ to visualize the relationships between these environments in two-dimensional space (Extended Data Fig. 2a). Highly similar local atomic environment vectors are positioned close to each other in the *t*-SNE embedding, allowing us to observe the clustering of atoms with similar local atomic environments. The result suggests that the local atomic environment is less dependent on the atomic number of the atom and that atoms with similar structural features form clusters, rather than grouping based on the properties of the molecule itself. For example, in the molecule '2-methyl-propyl-cyanoloxirane', we can see that each atom belongs to various clusters. The oxygen atom shown in the purple circle represents the oxygen atom in the pentagonal ring or oxirane, while the nitrogen atom shown in the green circle represents a cyanyl group. After atoms with similar local atomic environments are trained to have similar characteristics, the properties of each atom are combined to predict overall molecular properties.

Extended Data Fig. 2b shows the distribution of both node-level and graph-level embedding of each molecule learned in the ALIGNN model and the LACL model in the latent space using *t*-SNE in bi-dimensional space. It can be seen that the distribution of embeddings in the latent space changes substantially when the geometric domain changes for the ALIGNN model, which is the cause of the model's performance deteriorating when it is out of the trained domain. Embedding points between clusters mean the data that the model never faced during training, leading to performance degradation. However, the proposed LACL model maps both geometric domain inputs to the same location in the latent space, transferring information about the ground state to the CGCF domain. Therefore, the test data show similar embedding distributions and improved performance even in the CGCF domain. The embeddings of the encoder formed as a result of LACL training show that contrastive semantic alignment between the two domains has been successfully achieved.

However, LACL still has some limitations. It is worth mentioning that the characteristics of the latent space learned by the model vary depending on the predicted properties. While there is a tendency to learn continuous latent variables when learning properties (such as dipole moment) that depend heavily on the geometry itself or electronic band structure properties (such as bandgaps), discrete latent variables are learned when learning properties such as U0 and energy-related properties. When geometry is important information for predicting properties, continuous features are obtained as the local atomic environment is mapped continuously on the latent space according to subtle changes in interatomic distances. This tendency is more

pronounced in dipole moments compared with electron band structure properties and is a cause of difficulties in improving performance through contrastive learning, and addressing these challenges will be an important focus for future study. The features generated in electron band structure properties are more discrete than in dipole moments and more continuous than energy-related properties, and the greatest improvement in performance through contrastive learning occurs in these properties. Energy-related properties are more dependent on the molecular graph's topology or atomic number rather than subtle information about interatomic distances and are learned as discrete latent variables.

Our approach can be a viable alternative for minimizing the additional optimization process of complex molecular geometries in the computation of ground-state quantum-chemical properties. The recent rapid advancements in generative artificial intelligence have led to the emergence of molecular conformation-generation models. Nevertheless, achieving a data distribution equivalent to the *ab initio* conformation such as DFT remains a substantial challenge, highlighting the importance of domain-adaptation strategies. Our work opens opportunities for fast and accurate prediction of quantum-chemical properties.

Methods

Data preparation and geometric domain distribution

In this section, we explain the data acquisition process and the characteristics of each molecular geometric domain data. We utilize the text-based simplified molecular input line entry system (SMILES)^{36,37} representation from the QM9 dataset to obtain molecular conformations with low computational costs. A typical classical method for obtaining initial conformations is the experimental-torsion basic knowledge distance geometry (ETKDG)³⁸ method, which is a probabilistic search method with a lower computational cost than DFT. Recently, several methods^{25–29,34} have been proposed for generating molecular conformations using deep learning, which takes the advantage of utilizing the parallel processing capabilities of GPUs over classical rule-based methods to create molecular conformations much faster.

We choose molecular geometry data obtained from the ETKDG, CGCF-confgen and MMFF relaxation for comparison with the DFT geometry data distribution. The ETKDG domain data were obtained by applying ETKDG's RDKit³⁹ implementation to the SMILES representation, and the CGCF domain data were obtained by applying CGCF-ConfGen's official implementation. MMFF optimization was performed using RDKit's implementation.

Molecules in each geometric domain compared their different conformations with the Tanimoto distance implemented in the RDKit library. Each conformation was aligned according to the RDKit implementation before the Tanimoto distance measurement. The Tanimoto distance, a widely used metric in the field of drug discovery, can be used to quantitatively measure the distance between different conformations of the same molecule. If the conformations are completely identical, the Tanimoto distance is zero and the distance becomes larger as the conformations become different.

In the QM9 benchmark, we use 110,000 molecules for training, 10,000 for validation and the remainder for testing. The QMugs20 dataset was divided into training, validation and testing sets, consisting of 65,000, 1,500 and 1,706 conformations, respectively.

Molecular graph building

The ALIGNN¹¹ introduces an intuitive and simple line graph framework to model three-body interactions to incorporate embeddings of bond angles into edge features for richer representation and has inspired several studies on molecular properties predictions^{40–42}. The line graph of a given graph is constructed by treating the edges of the original graph as nodes, and connecting two nodes in the line graph whenever corresponding edges in the original graph share a common node.

We employed an atom-bond graph \mathcal{G} in which edges denote connections between atoms that are within a fixed cut-off radius of each other, regardless of whether there is an explicit bond between them. The state of atom i at layer l is represented as a hidden state node feature vector $\mathbf{h}_i^l \in \mathbb{R}^d$, and each edge has a representation \mathbf{e}_{ij}^l referred to as an edge feature. In one convolution layer, each feature is sequentially updated along the line graph and the original graph. At the final layer L of the network, the central node state \mathbf{h}_i^L is updated by gathering information from atoms that are within the receptive field, meaning those closer than the cut-off. During the readout step, the target value is computed from the node features of the entire graph by utilizing the readout function R , which remains consistent regardless of any permutations of the nodes. The bond-angle graph $L(\mathcal{G})$ is derived as a line graph of the atom-bond graph, such that bonds in the original graph correspond to nodes in the line graph. The latent representations of both edges in the original graph and nodes in the line graph are shared. This transformation allows for the modeling of higher-order structural features of a molecule, such as bond angles, which are not captured by the atom-bond representation.

We utilize the 'CGCNN atomic feature'⁴³ that includes hot-encoded 92 features implemented in JARVIS-Tools⁴⁴ as initial atom representations for our experiments. The interatomic distance is calculated as the difference between the positions of two atoms. The bond angle of a triplet of atoms, denoted as α_{ijk} , is initially calculated by taking the inner product of the unit vectors of the bond pairs. To further prepare initial edge features in the atom-bond graph and bond-angle graph, we apply radial basis function (RBF) expansion. We represent the atom, bond and angle features as \mathbf{x} , \mathbf{y} and \mathbf{z} , respectively. To update both the node and edge features, we utilize edge-gated graph convolution (EGGC). In each graph convolution layer, the features are sequentially updated along the line graph and the original graph.

$$\mathbf{m}^l, \mathbf{z}^l = \text{EGGC}(L(\mathcal{G}), \mathbf{y}^{l-1}, \mathbf{z}^{l-1}) \quad (1)$$

$$\mathbf{x}^l, \mathbf{y}^l = \text{EGGC}(\mathcal{G}, \mathbf{x}^{l-1}, \mathbf{m}^l) \quad (2)$$

where \mathbf{m}^l means messages at layer l and $L(\mathcal{G})$ means a line graph of a graph \mathcal{G} .

It is often necessary to ensure that molecular properties are invariant under transformations in the Euclidean group $E(3)$. While many GNN models guarantee the invariance of their predicted properties by operating solely on invariant inputs, this is not sufficient for vector properties such as dipole moments. Instead, these properties must satisfy equivariance, which means that the hidden representation must be transformed appropriately when the atomic geometry is transformed. A function $\phi: X \rightarrow Y$ is said to be equivariant to a group G if it satisfies the following condition:

$$\phi(T_g(x)) = S_g(\phi(x)) \quad \forall g \in G, \forall x \in X \quad (3)$$

where T_g is the transformation on the input space X for the abstract group g , and S_g is the transformation on the output space Y (ref. 45). The function ϕ is invariant, meaning that the output is unchanged by the transformation T_g when an operator S_g is an identity operator on Y . To address this issue, equivariant GNNs^{45,46} have been developed recently by utilizing displacement vectors. As ALIGNN, which we used as our backbone model, does not have an equivariant feature, we added it in this study. The output of the message function of our LACL model is fixed as a vector that is invariant under transformations. Transform equivariance can be achieved by multiplying this vector output by the coordinate embedding. To avoid the potential risk of learning a meaningless representation, we enforce symmetry with respect to the edge pair of the undirected molecular graph by applying different signs for source \mathbf{h}_i and target \mathbf{h}_j node embeddings.

$$\Phi_{ij} = W_h \mathbf{h}_i - W_h \mathbf{h}_j \quad (4)$$

$$\mathbf{v}_{ij} = \Phi_{ij} \otimes \mathbf{u}_{ij} \quad (5)$$

where W_h is the weight matrix for node embeddings and \mathbf{u}_{ij} is the unit vector of atomic dislocation vector \mathbf{r}_{ij} . \mathbf{v}_{ij} is the latent vector representation and \otimes is the outer product between two vectors. \mathbf{v}_{ij} is always the same as \mathbf{v}_{ji} as both Φ_{ij} and \mathbf{u}_{ij} change their sign when the source and target node are switched. The effect of this symmetric edge embedding is experimentally proved as shown in Supplementary Fig. 5.

Loss function

The two augmented views of a molecular graph \mathcal{G} : $\mathcal{G}_1 = (\tilde{X}_1, \tilde{A}_1)$ and $\mathcal{G}_2 = (\tilde{X}_2, \tilde{A}_2)$ are encoded by the online network and the target network, as hidden representation \tilde{H}_1 and \tilde{H}_2 , where \tilde{X} is node features and \tilde{A} is adjacency matrix. Each augmented view of the molecule corresponds to each geometric domain obtained from different relaxation methods. The model is trained to maximize the cosine similarity between online nonlinear projection \tilde{Z}_1 and target representation \tilde{H}_2 . The weight of the target encoder is updated with an exponential moving average of the weights of the online encoder. Through a bootstrapping process that predicts the embedding of the DFT domain from other domain inputs, the encoder transplants ground-state quantum-chemical information into other molecular geometric domains obtained by computationally efficient conformation-generation methods. To predict target property, node-wise hidden representation \tilde{H}_1 is pooled to graph representation \tilde{C}_1 . The entire model is trained end to end through the entire pipeline consisting of data preparation, and GNN models considered augmented nonlinear projection head \tilde{Z}_1 and BGRL loss and decoder³³, and target prediction loss. The BGRL loss optimized by following this neural network architecture performs contrastive semantic alignment between the two domains.

$$\mathcal{L}_{\text{contrastive}}(\theta, \phi) = -\frac{2}{N} \sum_{i=0}^{N-1} \frac{\tilde{Z}_{1,i} \tilde{H}_{2,i}^T}{\|\tilde{Z}_{1,i}\| \|\tilde{H}_{2,i}\|} \quad (6)$$

$$\mathcal{L}_{\text{prediction}}(\mathbf{y}_{\text{label}}, \hat{\mathbf{y}}) = \text{MSE}(\mathbf{y}_{\text{label}}, \hat{\mathbf{y}}) \quad (7)$$

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contrastive}} + \alpha \mathcal{L}_{\text{prediction}} \quad (8)$$

$$\phi \leftarrow \tau \phi + (1 - \tau) \theta \quad (9)$$

where θ is the parameters of the online network, ϕ is the parameters of the target network, MSE is mean squared error, α is a hyperparameter and τ is a decay rate. For vector properties such as dipole moment (μ), we use vector norm to calculate the loss.

$$\mathcal{L}_{\text{prediction}}(\mu_{\text{label}}, \hat{\mu}) = \sqrt{\frac{1}{N} \sum_i^N (\mu_{\text{label}} - \|\hat{\mu}\|)^2} \quad (10)$$

Computational efficiency of LACL

We first prepare the SMILES representation of the molecule and generate the initial conformation by applying the ETKDG and MMFF optimizations sequentially. We then perform DFT relaxation using the Python-based Simulations of Chemistry Framework (PySCF)⁴⁷, a popular quantum-chemistry computation library, and use the resulting geometry as input to the trained ALIGNN model. In contrast, we use the initial conformation as input to our model.

Investigation for long-chain conformers

To test the conformation change effect in long chains, we selected 390 molecules which include 20 heavy atoms each from the test data.

Finally, we heuristically selected 12 molecules among them following the rules: molecules that have rotatable bonds as close as possible to the center of mass; and molecules with their rotatable bonds that are folded less than 100°.

The selected molecules are considered compact conformers. In addition, we unfolded the molecules to 180° to obtain open conformers. The 12 pairs of molecules are optimized using MMFF to generate their conformations, which are inputs of the trained model. To label these conformers, we relaxed them further using the extended semiempirical tight-binding (XTB) calculator⁴⁸ and obtained their dipole moments using the atomic simulation environment (ASE) package⁴⁹.

Implementation and training

Our model was implemented using the PyTorch⁵⁰ and deep graph library (DGL)⁵¹. The AdamW optimizer with normalized weight decay of 10^{-5} was used. A learning rate reduction strategy during plateaus was employed and training was conducted for 500 epochs with early stopping applied if no improvement was observed. All the training process was carried out using an NVIDIA RTX 3090 24 GB GPU.

Data availability

The preprocessed data in this work for reproducing the results are available on figshare at <https://doi.org/10.6084/m9.figshare.24445129> (ref. 52). The model checkpoints used in this work for reproducing the results are available on GitHub at <https://github.com/parkymit/LACL> and figshare at <https://doi.org/10.6084/m9.figshare.24456802> (ref. 53). Source data are provided with this paper.

Code availability

The Python code capsule of this work including the training script for reproducing the results is available on GitHub at <https://github.com/parkymit/LACL> and figshare at <https://doi.org/10.6084/m9.figshare.24456802> (ref. 53).

References

- Olivecrona, M., Blaschke, T., Engkvist, O. & Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminform.* **9**, 48 (2017).
- Jeon, W. & Kim, D. Autonomous molecule generation using reinforcement learning and docking to develop potential novel inhibitors. *Sci. Rep.* **10**, 22104 (2020).
- Reker, D. & Schneider, G. Active-learning strategies in computer-assisted drug discovery. *Drug Discov. Today* **20**, 458–465 (2015).
- De Cao, N. & Kipf, T. MolGAN: an implicit generative model for small molecular graphs. Preprint at *arXiv* <https://doi.org/10.48550/arXiv.1805.11973> (2018).
- Guo, M. et al. Data-efficient graph grammar learning for molecular generation. Preprint at *arXiv* <https://doi.org/10.48550/arXiv.2203.08031> (2022).
- Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry. In *Proc. 34th International Conference on Machine Learning, Proc. Machine Learning Research* Vol. 70 (eds Precup, D. & Teh, Y. W.) 1263–1272 (PMLR, 2017).
- Unke, O. T. & Meuwly, M. PhysNet: a neural network for predicting energies, forces, dipole moments, and partial charges. *J. Chem. Theory Comput.* **15**, 3678–3693 (2019).
- Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet—a deep learning architecture for molecules and materials. *J. Chem. Phys.* **148**, 241722 (2018).
- Klicpera, J., Groß, J. & Günnemann, S. Directional message passing for molecular graphs. In *International Conference on Learning Representations* Vol. 8 (2020).

10. Klicpera, J., Giri, S., Margraf, J. T. & Günnemann, S. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. Preprint at <https://arxiv.org/abs/2011.14115> (2020).
11. Choudhary, K. & DeCost, B. Atomistic line graph neural network for improved materials property predictions. *npj Comput. Mater.* **7**, 185 (2021).
12. Schütt, K., Unke, O. & Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *Proc. 38th International Conference on Machine Learning, Proc. Machine Learning Research* Vol. 139 (eds Meila, M. & Zhang, T.) 9377–9388 (PMLR, 2021).
13. Lim, J. et al. Predicting drug–target interaction using a novel graph neural network with 3D structure-embedded graph representation. *J. Chem. Inf. Model.* **59**, 3981–3988 (2019).
14. Fout, A., Byrd, J., Shariat, B. & Ben-Hur, A. Protein interface prediction using graph convolutional networks. *Adv. Neural Inf. Process. Syst.* **30**, 6530–6539 (2017).
15. Tang, B. et al. A self-attention based message passing neural network for predicting molecular lipophilicity and aqueous solubility. *J. Cheminform.* **12**, 15 (2020).
16. Ramakrishnan, R., Dral, P. O., Rupp, M. & Von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data* **1**, 140022 (2014).
17. Becke, A. D. Density-functional thermochemistry. I. The effect of the exchange-only gradient correction. *J. Chem. Phys.* **96**, 2155–2160 (1992).
18. Lee, C., Yang, W. & Parr, R. G. Development of the Colle–Salvetti correlation-energy formula into a functional of the electron density. *Phys. Rev. B* **37**, 785 (1988).
19. Ditchfield, R., Hehre, W. J. & Pople, J. A. Self-consistent molecular-orbital methods. IX. An extended Gaussian-type basis for molecular-orbital studies of organic molecules. *J. Chem. Phys.* **54**, 724–728 (1971).
20. Frisch, M. J., Pople, J. A. & Binkley, J. S. Self-consistent molecular orbital methods 25. Supplementary functions for Gaussian basis sets. *J. Chem. Phys.* **80**, 3265–3269 (1984).
21. Hehre, W. J., Ditchfield, R. & Pople, J. A. Self-consistent molecular orbital methods. XII. Further extensions of Gaussian-type basis sets for use in molecular orbital studies of organic molecules. *J. Chem. Phys.* **56**, 2257–2261 (1972).
22. Krishnan, R., Binkley, J. S., Seeger, R. & Pople, J. A. Self-consistent molecular orbital methods. XX. A basis set for correlated wave functions. *J. Chem. Phys.* **72**, 650–654 (1980).
23. Halgren, T. A. MMFF VI. MMFF94s option for energy minimization studies. *J. Comput. Chem.* **20**, 720–729 (1999).
24. Lemm, D., von Rudorff, G. F. & von Lilienfeld, O. A. Machine learning based energy-free structure predictions of molecules, transition states, and solids. *Nat. Commun.* **12**, 4468 (2021).
25. Xu, M. et al. GeoDiff: a geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations* Vol. 10 (2022).
26. Luo, S., Shi, C., Xu, M. & Tang, J. Predicting molecular conformation via dynamic graph score matching. *Adv. Neural Inf. Process. Syst.* **34**, 19784–19795 (2021).
27. Zhu, J. et al. Direct molecular conformation generation. *Transactions on Machine Learning Research* (2022).
28. Lemm, D., von Rudorff, G. F. & von Lilienfeld, O. A. Machine learning based energy-free structure predictions of molecules, transition states, and solids. *Nat. Commun.* **12**, 4468 (2021).
29. Mansimov, E., Mahmood, O., Kang, S. & Cho, K. Molecular geometry prediction using a deep generative graph neural network. *Sci. Rep.* **9**, 20381 (2019).
30. Isert, C., Atz, K., Jiménez-Luna, J. & Schneider, G. QMugs, quantum mechanical properties of drug-like molecules. *Sci. Data* **9**, 273 (2022).
31. Ganin, Y. & Lempitsky, V. Unsupervised domain adaptation by backpropagation. In *Proc. 32nd International Conference on Machine Learning, Proc. Machine Learning Research* Vol. 37 (eds Bach, F. & Blei, D.) 1180–1189 (PMLR, 2015).
32. Chen, Z., Li, X. & Bruna, J. Supervised community detection with line graph neural networks. In *International Conference on Learning Representations* Vol. 5 (2017).
33. Thakoor, S. et al. Large-scale representation learning on graphs via bootstrapping. In *International Conference on Learning Representations* Vol. 10 (2022).
34. Xu, M., Luo, S., Bengio, Y., Peng, J. & Tang, J. Learning neural generative dynamics for molecular conformation generation. In *International Conference on Learning Representations* Vol. 9 (2021).
35. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
36. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
37. Weininger, D., Weininger, A. & Weininger, J. L. SMILES. 2. Algorithm for generation of unique smiles notation. *J. Chem. Inf. Comput. Sci.* **29**, 97–101 (1989).
38. Riniker, S. & Landrum, G. A. Better informed distance geometry: using what we know to improve conformation generation. *J. Chem. Inf. Model.* **55**, 2562–2574 (2015).
39. Landrum, G. RDKit: open-source cheminformatics. <http://www.rdkit.org> (2006).
40. Hsu, T. et al. Efficient and interpretable graph network representation for angle-dependent properties applied to optical spectroscopy. *npj Comput. Mater.* **8**, 151 (2022).
41. Kaundinya, P. R., Choudhary, K. & Kalidindi, S. R. Prediction of the electron density of states for crystalline compounds with atomistic line graph neural networks (ALIGNN). *JOM* **74**, 1395–1405 (2022).
42. Fang, X. et al. Geometry-enhanced molecular representation learning for property prediction. *Nat. Mach. Intell.* **4**, 127–134 (2022).
43. Xie, T. & Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys. Rev. Lett.* **120**, 145301 (2018).
44. Choudhary, K. et al. The joint automated repository for various integrated simulations (JARVIS) for data-driven materials design. *npj Comput. Mater.* **6**, 173 (2020).
45. Satorras, V. G., Hoogeboom, E. & Welling, M. E(n) equivariant graph neural networks. In *Proc. 38th International Conference on Machine Learning, Proc. Machine Learning Research* Vol. 139 (eds Meila, M. & Zhang, T.) 9323–9332 (PMLR, 2021).
46. Batzner, S. et al. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 2453 (2022).
47. Sun, Q. et al. PySCF: the Python-based simulations of chemistry framework. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **8**, e1340 (2018).
48. Bannwarth, C., Ehlert, S. & Grimme, S. GFN2-xtTB—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. *J. Chem. Theory Comput.* **15**, 1652–1671 (2019).
49. Larsen, A. H. et al. The atomic simulation environment—a Python library for working with atoms. *J. Phys. Condensed Matter* **29**, 273002 (2017).
50. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **32**, 8026–8037 (2019).

51. Wang, M. Y. Deep graph library: towards efficient and scalable deep learning on graphs. In *International Conference on Learning Representations* Vol. 7 (2019).
52. Park, Y. J., Kim, H., Jo, J. & Yoon, S. sharedata-to-reproduce-lacl. *figshare* <https://doi.org/10.6084/m9.figshare.24445129> (2023).
53. Park, Y. J., Kim, H., Jo, J. & Yoon, S. LACL. *figshare* <https://doi.org/10.6084/m9.figshare.24456802> (2023).

Acknowledgements

Y.J.P. was supported by a grant from the National Research Foundation of Korea (NRF) funded by the Korean government, Ministry of Science and ICT (MSIT) (no. 2021R1A6A3A01086766). The O5-Neuron supercomputer was provided by the Korea Institute of Science and Technology Information (KISTI) National Supercomputing Center for Y.J.P. Y.J.P., H.K., J.J. and S.Y. were supported by Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea government (MSIT) (2021-O-01343: Artificial Intelligence Graduate School Program (Seoul National University)), National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2022R1A3B1077720) and the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2023. We express our gratitude to J. Im at the Chemical Data-driven Research Center in the Korea Research Institute of Chemical Technology (KRICT) for his valuable insights and discussion on the content of this paper.

Author contributions

Y.J.P. conceived the study. Y.J.P. and S.Y. supervised the research. Y.J.P. designed and implemented the deep learning framework. Y.J.P., H.K. and J.J. conducted benchmarks and case studies. All authors participated in the preparation (writing and drawing) of the paper and the analysis of experimental results. All authors reviewed and edited the submitted version of the paper.

Competing interests

The authors declare no competing interests.

Additional information

Extended data is available for this paper at <https://doi.org/10.1038/s43588-023-00560-w>.

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s43588-023-00560-w>.

Correspondence and requests for materials should be addressed to Yang Jeong Park or Sungroh Yoon.

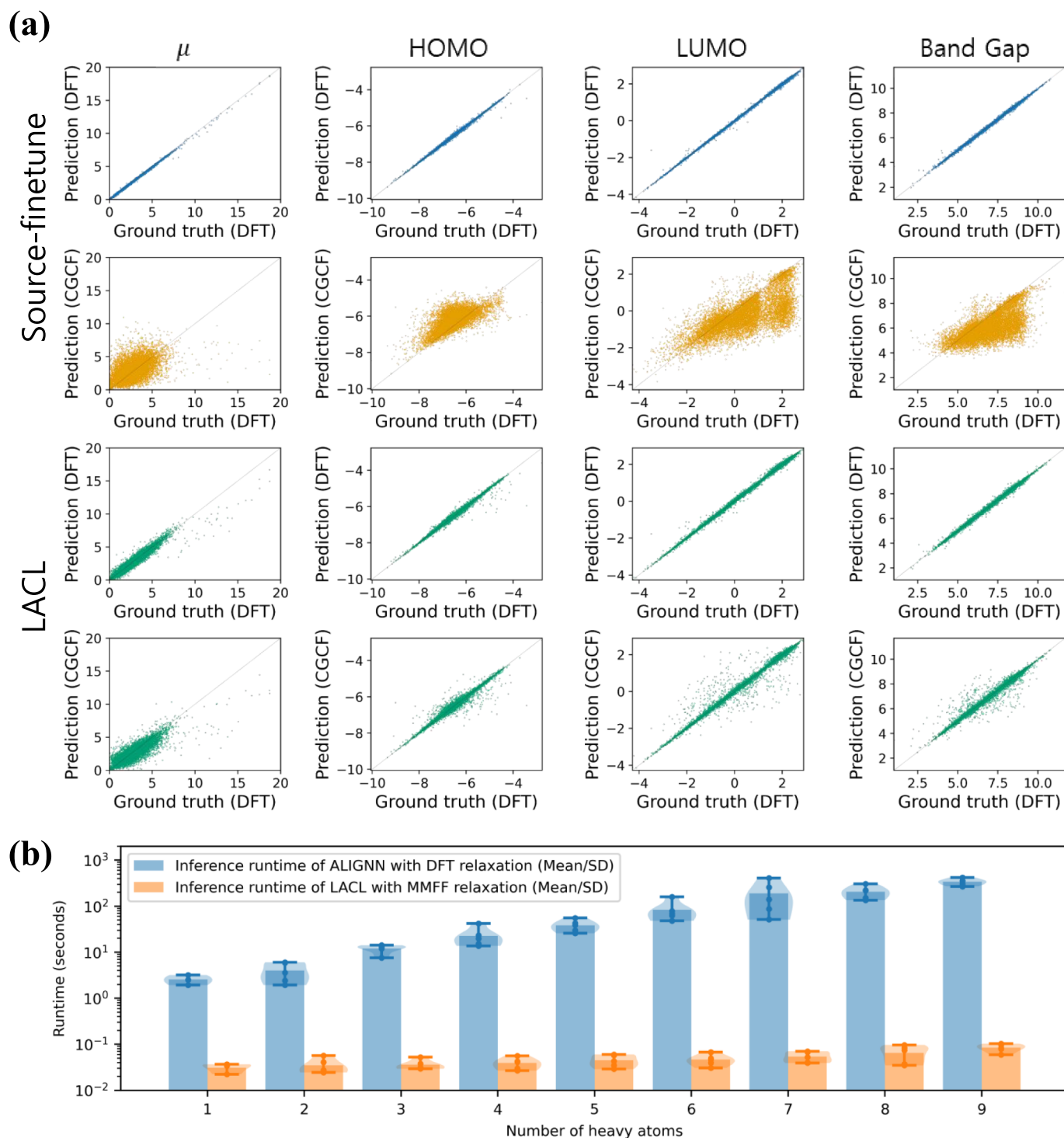
Peer review information *Nature Computational Science* thanks the anonymous reviewers for their contribution to the peer review of this work. Primary Handling Editor: Kaitlin McCardle, in collaboration with the *Nature Computational Science* team.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

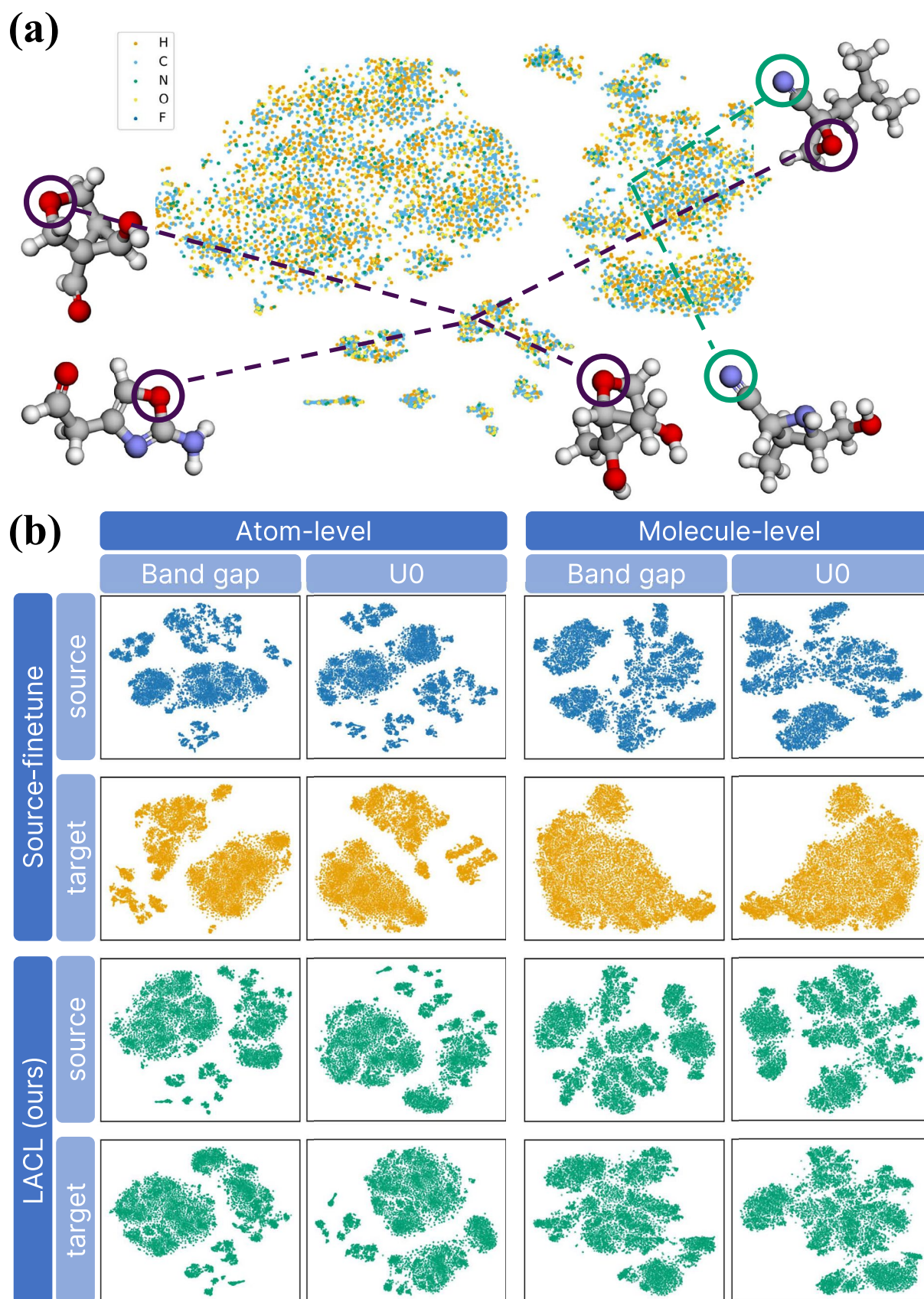
Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

© The Author(s), under exclusive licence to Springer Nature America, Inc. 2023



Extended Data Fig. 1 | Comparison between a backbone ALIGNN and the LACL model from the point of view of accuracy and inference time. (a) Parity plot of trained ALIGNN and LACL model for various regression targets in QM9 dataset. Molecular conformation data from both the DFT domain and the CGCF domain is used as a test dataset. μ is the dipole moment. (b) Comparison of computation time between an ALIGNN model with DFT geometric relaxation and the LACL

model with MMFF geometric relaxation. Geometric relaxations running on two 24-core Intel Cascade Lake i9 CPUs and GNN architectures running on a single NVIDIA RTX3090 graphics processing unit (GPU). The bar indicates the mean of computation time and the error bar indicates the standard deviation. Five samples were collected for runtime calculation, except When the number of heavy atoms was one (three samples).



Extended Data Fig. 2 | See next page for caption.

Extended Data Fig. 2 | 2-D t-SNE visualization of trained node and graph representations from both ALIGNN and LACL model for bandgap and internal energy at 0 K (U0) regression in QM9 dataset. (a) 2-D t-SNE visualization of trained representations of the local atomic environment from LACL model for bandgap regression in QM9 dataset. Molecular conformation data from both DFT and CGCF domains are used as the test dataset. Orange, sky blue, green, yellow, and blue-colored point indicates hydrogen, carbon, nitrogen, oxygen, and fluorine, respectively. To visualize the node representation of

different molecules, several example molecules are shown. The atom surrounded by a green circle is a nitrogen atom belonging to the cyanyl group. The atom surrounded by the purple circle is the oxygen atom included in the ring. The hydrogen, carbon, nitrogen, and oxygen atoms in molecules are indicated as white, gray, purple, and red color, respectively. (b) t-SNE visualization for trained node (atom-level) and graph (molecule-level) representations. Representations are visualized for each level, feature, model, and domain.