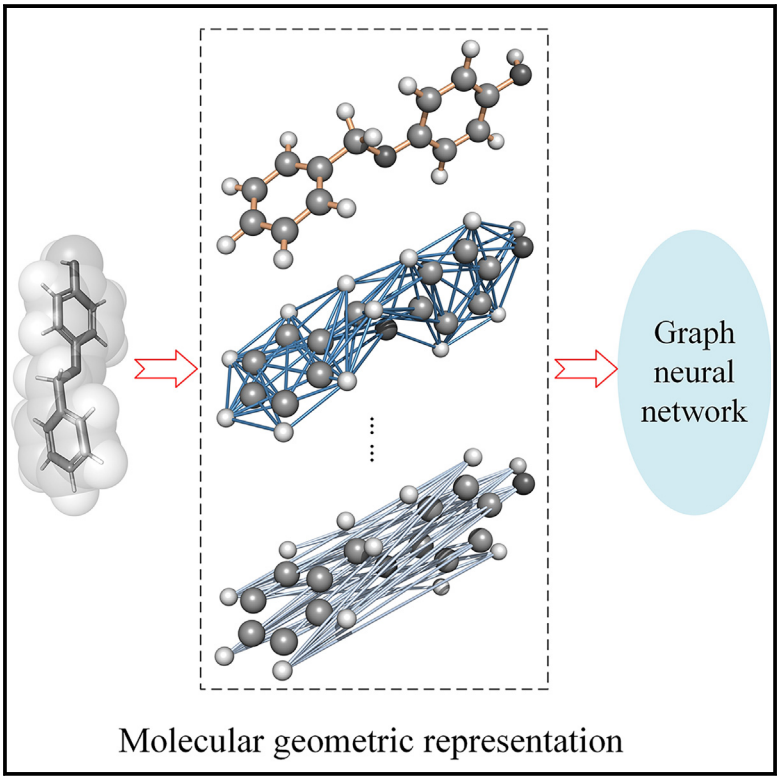
Cell Reports
Methods

Article

Molecular geometric deep learning

Graphical abstract



Highlights

- Mol-GDL predicts molecular properties considering covalent and non-covalent information
- Mol-GDL is benchmarked on 14 commonly used datasets
- Mol-GDL can achieve a better performance than state-of-the-art methods
- Mol-GDL demonstrates the role of non-covalent interactions in molecular models

Authors

Cong Shen, Jiawei Luo, Kelin Xia

Correspondence

luojiawei@hnu.edu.cn (J.L.),
xiakelin@ntu.edu.sg (K.X.)

In brief

Shen et al. propose molecular geometric deep learning (Mol-GDL), which builds more general molecular representation into GDL models. In Mol-GDL, molecular topology is modeled as a series of molecular graphs reflecting different scales of atomic interactions. Effective molecular property prediction by Mol-GDL highlights the role of non-covalent interactions.

Shen et al., 2023, Cell Reports Methods *3*, 100621
November 20, 2023
https://doi.org/10.1016/j.crmeth.2023.100621

CellPress

Cell Reports Methods

CellPress
OPEN ACCESS

Article

Molecular geometric deep learning

Cong Shen,[1,2] Jiawei Luo,[1,*] and Kelin Xia[2,3,*]
[1]College of Computer Science and Electronic Engineering, Hunan University, Changsha 410000, China
[2]School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371, Singapore
[3]Lead contact
*Correspondence: luojiawei@hnu.edu.cn (J.L.), xiakelin@ntu.edu.sg (K.X.)
https://doi.org/10.1016/j.crmeth.2023.100621

MOTIVATION Geometric deep learning (GDL) has demonstrated huge power and enormous potential in molecular data analysis. However, a great challenge still remains for highly efficient molecular representations. Currently, covalent-bond-based molecular graphs are the *de facto* standard for representing molecular topology at the atomic level. Here, we demonstrate that molecular graphs constructed only from non-covalent bonds can achieve similar or even better results than covalent-bond-based models in molecular property prediction. This demonstrates the great potential of novel molecular representations beyond the *de facto* standard of covalent-bond-based molecular graphs. Based on the finding, we propose molecular geometric deep learning (Mol-GDL). In our Mol-GDL, molecular topology is modeled as a series of molecular graphs, each focusing on a different scale of atomic interactions. In this way, both covalent interactions and non-covalent interactions are incorporated into the molecular representation on an equal footing.

SUMMARY

Molecular representation learning plays an important role in molecular property prediction. Existing molecular property prediction models rely on the *de facto* standard of covalent-bond-based molecular graphs for representing molecular topology at the atomic level and totally ignore the non-covalent interactions within the molecule. In this study, we propose a molecular geometric deep learning model to predict the properties of molecules that aims to comprehensively consider the information of covalent and non-covalent interactions of molecules. The essential idea is to incorporate a more general molecular representation into geometric deep learning (GDL) models. We systematically test molecular GDL (Mol-GDL) on fourteen commonly used benchmark datasets. The results show that Mol-GDL can achieve a better performance than state-of-the-art (SOTA) methods. Extensive tests have demonstrated the important role of non-covalent interactions in molecular property prediction and the effectiveness of Mol-GDL models.

INTRODUCTION

Artificial intelligence (AI)-based models have demonstrated huge power and enormous potential for the prediction of various molecular properties. In particular, AI-based drug design has achieved great success in various steps in virtual screening and has the potential to revolutionize the drug industry.[1–4] However, even with this great progress, designing efficient molecular representations and featurization remains a great challenge. In general, all AI-based molecular models can be classified into two types, i.e., molecular-descriptor-based machine learning models and end-to-end deep learning models.

The first type uses handcrafted molecular descriptors or fingerprints as input features for machine learning models. The generation of the handcrafted descriptors is known as featurization or feature engineering. Other than physical, chemical, and biological properties, such as atomic partial charge, hydrophobicity, electronic properties, steric properties, etc., the majority of the molecular features are obtained from molecular structural properties. In fact, more than 5,000 molecular structural descriptors have been developed and can be generalized as one-dimensional (1D), two-dimensional (2D), three-dimensional (3D), and four-dimensional (4D) features.[5,6] The 1D molecular descriptors include atom counts, bond counts, molecular weight, fragment counts, functional group counts, and other summarized general properties. The 2D molecular descriptors include topological indices, graph properties, combinatorial properties, molecular profiles, autocorrelation coefficients, etc. The 3D molecular descriptors include molecular surface properties, volume properties, autocorrelation descriptors, substituent constants, quantum chemical descriptors, etc. A related higher computational cost is usually required for the generation of 3D molecular descriptors. The 4D chemical descriptors characterize configuration changes in a dynamic process. These structural descriptors are widely used in

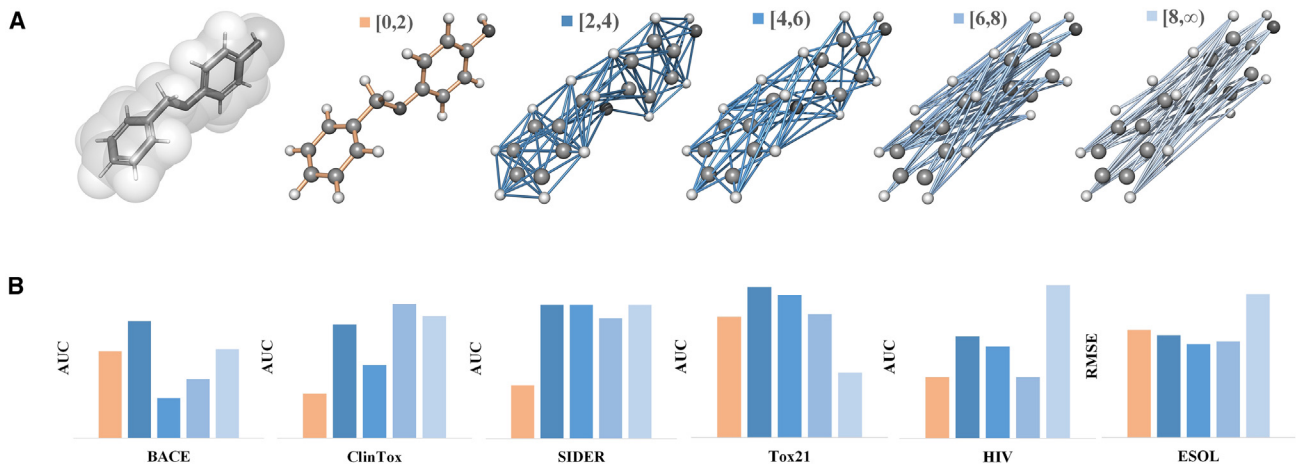


Figure 1. Illustration of different molecular graph representations and the performance of their GDLs in six commonly used datasets

The *de facto* standard of covalent-bond-based molecular graph representation has clear limitations and is inferior to models with only non-covalent interactions.

(A) Molecular graph representations for monobenzone molecule. The *de facto* standard of the covalent bond model is represented in orange, and the other four non-covalent-interaction-based graphs are in blue.

(B) The performance of GDLs with five different molecular representations on the six most commonly used datasets. The color of the bar (for each model) is the same as that of the corresponding molecular graph. For instance, the orange bars are for GDLs with covalent-bond-based molecular graphs.

quantitative structure-activity relationships (QSARs), quantitative structure-property relationships (QSPRs), and machine learning models. Recently, deep learning models, including autoencoder, convolutional neural network (CNN), and graph neural network (GNN), have also been used in molecular fingerprint generation.[7–11] For all machine learning models, their performance is directly related to the efficiency of these molecular descriptors.

The second type is end-to-end geometric deep learning (GDL) models. In these GDLs, molecules are represented as molecular graphs, density functions, or molecular surfaces, and various deep learning models, such as (3D) CNNs, GNNs, recurrent neural networks (RNNs), etc., can be used to automatically learn the molecular properties.[12–16] Among these different molecular representations, molecular graphs are the most popular one. In particular, covalent-bond-based molecular graphs are the *de facto* standard for representing molecular topology at the atomic level. Based on them, various GDL models have been proposed, including graph RNNs (GraphRNNs),[17] graph convolutional networks (GCNs),[18] graph autoencoders,[19] graph transformers,[20] and others. These GDLs have been widely used in molecular data analysis, in particular drug design.[21–23] Recently, non-covalent-interaction-based molecular descriptors have achieved great performance in the prediction of binding affinities of protein-ligand and protein-protein,[24,25] demonstrating potential new molecular graph representations beyond the *de facto* standard of covalent-bond-based graphs. In fact, the incorporation of geometric information, such as bond angles, periodicity, symmetry, rotation-translation invariance, equivalence, etc., into GDL models can help to significantly improve the learning performance,[26–29] especially for machine-learned-based force-field models. Efficient representations beyond covalent-bond-based molecular graphs provide a great promise for a better representation of molecular geometric information.

Here, we show, for the first time, that molecular representations with only non-covalent interactions can achieve similar or even better results than the *de facto* standard of covalent-bond-based models in molecular property prediction. More specifically, we systematically compare the performance of GDL models using two types of molecular representations, i.e., covalent interaction graphs and non-covalent interaction graphs, in several of the most commonly used benchmark datasets, including BACE, ClinTox, SIDER, Tox21, HIV, ESOL, etc. It has been found that GDL models using only non-covalent interactions have comparable or even superior performance than the *de facto* standard models. Further, we propose molecular GDL (Mol-GDL) to incorporate a more general molecular representation into GDLs. In our Mol-GDL, molecular topology is modeled as a series of molecular graphs, each focusing on a different scale of atomic interactions. In this way, both covalent interactions and non-covalent interactions are incorporated into the molecular representation on an equal footing. We systematically test Mol-GDL on fourteen commonly used benchmark datasets. The results show that our Mol-GDL can achieve a better performance than state-of-the-art (SOTA) methods.

RESULTS

Is covalent-bond-based molecular graph a *de facto* standard for GDL?

Currently, the *de facto* standard for molecular representation at the atomic level is covalent-bond-based molecular graphs. Here, we show that, in molecular property prediction, GDLs with molecular graphs constructed only from non-covalent interactions can achieve similar or even better results. We consider five different molecular graph representations and their GDL performance on the six most commonly used datasets, including BACE, ClinTox, SIDER, Tox21, HIV, and ESOL, as illustrated in Figure 1. Of the five different molecular graphs, one is the *de facto* standard of the covalent-bond-based model, and the other four are all constructed using only non-covalent interactions. Stated differently,

all the edges in these four molecular graphs represent only non-covalent information, and none of them are generated from covalent bonds. Mathematically, these four non-covalent molecular graphs are constructed by defining the edges only between atoms within a certain pre-defined Euclidean distance (larger than the covalent bond distances). We specify a certain domain I for each graph in such a way that the edge in this graph exists if and only if the distance between the corresponding two atoms is in the domain I. For instance, a graph $G(I)$ with $I = [4, 6)$ means that all the edges in graph $G(I)$ have lengths between 4 and 6 Å. Details for the molecular graph representation can be found in Mol-GDL.

It can be seen from Figure 1 that GDL with the *de facto* standard of covalent-bond-based molecular graphs does not have the best performance. In contrast, GDLs with non-covalent molecular graphs can not only have comparable results but can even outperform the covalent-bond-based model. In fact, for all six datasets, the non-covalent model with $I = [4, 6)$ has superior performance over the *de facto* standard model. More interestingly, even for the highly non-traditional molecular graph representation with $I = [8, \infty)$, in which edges only form between atoms that have Euclidean distances larger than 8 Å, the corresponding GDL model can still have a comparably good performance and even outperform the *de facto* standard model in the four test datasets, including BACE, ClinTox, SIDER, and HIV. This demonstrates the great potential of novel molecular representations beyond the *de facto* standard of covalent-bond-based molecular graphs.

Mol-GDL

Molecular graph representation for Mol-GDL

For a molecule with N atoms and atom coordinates denoted as $\mathbf{r}_1$, $\mathbf{r}_2$,…,$\mathbf{r}_N$, its molecular graph representation can be expressed as $G(I) = (V, E(I))$. Here, I is a certain interaction region that we used to define our graph representation. For simplicity, we only consider region in terms of $I = [x_{\min}, x_{\max})$ with $x_{\min}$ and $x_{\max}$ real values (that satisfy $0 \le x_{\min} < x_{\max}$). The set of nodes (or atoms) is denoted V, and the set of edges is denoted $E(I)$. Note that each atom is modeled as a single node (or vertex) in our molecular graph representation. The adjacent matrix $A(I) = (a(I)_{ij})_{1 \le i \le N; 1 \le j \le N}$ and its element $a(I)_{ij}$ are represented as

$$a(I)_{ij} = \begin{cases} 1, & x_{\min} \le \| \mathbf{r}_i - \mathbf{r}_j \| < x_{\max} \text{ and } i \neq j \\ 0, & \text{others.} \end{cases} \quad \text{(Equation 1)}$$

Geometrically, this means that an edge is formed between the i-th and j-th atoms if their Euclidean distance $\| \mathbf{r}_i - \mathbf{r}_j \|$ is within the domain of I in molecular graph $G(I)$. For instance, we can set $I_0 = [0, 2)$, and the corresponding molecular graph $G(I_0)$ is exactly the *de facto* standard molecular graph, which is constructed with only covalent bonds, as their bond lengths are all within the region of $I_0 = [0, 2)$. Mathematically, by the variation of the domain I, graph representations with dramatically different topologies can be generated. For instance, if we let $x_{\min} > 2$ (thus, $x_{\max} > 2$), the corresponding molecular graph will contain only non-covalent interactions.

In our Mol-GDL, instead of using only one molecular graph, a series of graphs $G(I_k)$ are systematically generated by selecting different regions I_k $(k > 0)$ (see STAR Method for details). Geometrically, for a certain graph $G(I_k)$, an edge is formed between two atoms only when their Euclidian distance is within the region of I_k. In this way, we have great flexibility to construct molecular graphs.

Geometric node features for Mol-GDL

The other important setting for our Mol-GDL is the distance-related node features. In contrast to traditional node features, our node features contain only atom types and distance information.

Mathematically, the node feature for the i-th vertex of molecular graph $G(I)$ is denoted as $f_i(I) = [f_i(I, \alpha_1), f_i(I, \alpha_2), \ldots, f_i(I, \alpha_m)]$. Note that here α_j (with $1 \le j \le m$) means the type of atoms, such as carbon (C), nitrogen (N), oxygen (O), hydrogen (H), sulfur (S), etc. The element $f_i(I, \alpha_j)$ is defined as the number (or frequency) of edges in $G(I)$ that are formed between the i-th node and any other nodes that are of atom type α_j. Mathematically, it is defined as follows:

$$f_i(I, \alpha_j) = \sum_{T_j = \alpha_j} \chi(x_{\min} \le \|\mathbf{r}_i - \mathbf{r}_j\| < x_{\max}). \quad \text{(Equation 2)}$$

Here, T_j is the atom type of the j-th atom, and the value of the indicator function χ is 1 if the following condition is satisfied and 0 otherwise. Geometrically, the node feature contains m descriptors, and each descriptor represents the total number of edges connecting to atoms of a specific atom type. Figure 2 illustrates the node features using a carbon atom as an example.

Further, a refined node feature can be generated through the subdivision of domain I. More specifically, we can divide I into L intervals $\{I^l = [x_{l-1}, x_l); l = 1, 2, .., L\}$ with $x_0 = x_{\min}$ and $x_L = x_{\max}$. The node feature element $f_i(I, \alpha_j)$ will be extended into a vector $[f_i(I^1, \alpha_j), f_i(I^2, \alpha_j), \ldots, f_i(I^L, \alpha_j)]$. Similar, the scale element $f_i(I^1, \alpha_j)$ is defined as

$$f_i(I^l, \alpha_j) = \sum_{T_j = \alpha_j} \chi(x_{l-1} \le \|\mathbf{r}_i - \mathbf{r}_j\| < x_l). \quad \text{(Equation 3)}$$

Note that our node features are solely determined by atom types and Euclidean distances between atoms.

The general framework for Mol-GDL

Our Mol-GDL provides a GDL architecture that can comprehensively learn the multi-scale information within a molecule. Figure 3 illustrates the flowchart of our Mol-GDL. Different from all previous GDLs, a series of molecular graphs focusing on different scales of interactions are systematically constructed. A common message-passing (MP) module is employed on each individual molecular graph. After that, two pooling operations are used. The first pooling is done at the atomic level. For each molecular graph, node features are aggregated together into a single molecular feature vector. The second pooling is done at the molecular graph level. Each molecular feature vector from the first pooling goes through a single-layer perceptron, and then they are concatenated into a single feature vector. Finally, the single feature vector, which contains information from all the molecular graphs, goes through multi-layer perceptron (MLP) to generate the final prediction.

Performance of Mol-GDL for molecular data analysis

In this section, we present the comparison results between our Mol-GDLs and SOTAs on different types of molecular data

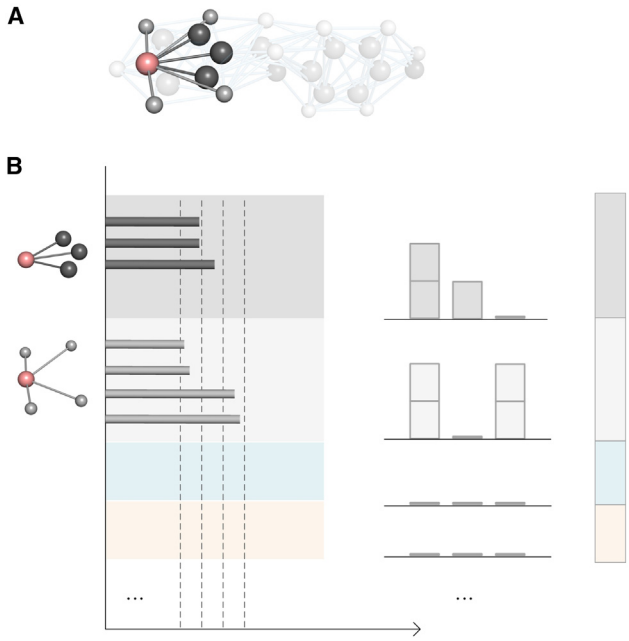


Figure 2. Illustration of geometric node features for Mol-GDL

Different from all previous models, our geometric node features are solely determined by atom types and Euclidean distances between atoms.

(A) The illustration of a carbon atom (in pink) and its neighboring carbon atoms (in dark black) and hydrogen atoms (in light gray) from a molecular graph for monobenzone.

(B) In our geometric node features, the neighboring atoms are grouped based on their atom types. Here, the two carbon atoms are classified into one group, and the four hydrogen atoms are classified into the other. For each group, the Euclidian distances between all the neighboring atoms to the carbon atoms are classified into several intervals. For each interval, we count the total number (or frequency) of distances within it. Here, three equal-sized intervals are considered. For carbon atoms, their frequencies in these intervals are (2, 1, 0), and for hydrogen atoms, their numbers are (2, 0, 2). These frequency numbers are then concatenated (in a pre-defined order according to atom types) into a fixed-length long vector, i.e., our geometric node feature.

analysis, including classification tasks on the molecular property, regression tasks on the molecular property, and tasks on molecular interaction prediction. For a fair comparison with SOTA models, we adopt the same scaffold split method to divide task datasets into train, valid, and test parts with a ratio of 8:1:1.[4] Our Mol-GDLs are trained individually on each task and then directly compared with SOTA models. The detailed information of datasets can be found in Table S1.

Mol-GDL for classification tasks on molecular property

In the classification task, seven commonly used datasets[30] are considered. These datasets can be roughly classified into two categories: one for biophysical properties (BACE, HIV, and MUV) and the other for physiological properties (BBBP, Tox21, SIDER, and ClinTox). More specifically, the BACE dataset provides quantitative binding results for inhibitors of human β-secretase 1 (BACE-1).[31] Its average number of atoms per molecule is around 65, which is the largest of the seven datasets. The HIV dataset is for the study of the molecule's ability to inhibit HIV replication. The MUV dataset, which is screened from PubChem BioAssay by applying a nearest-neighbor analysis,[32] is devoted to the validation of virtual screening techniques. The BBBP dataset is for barrier permeability,[33] while SIDER is a database of marketed drugs and adverse drug reactions.[34] Both Tox21 and ClinTox datasets[35] are related to the toxicity of compounds, and they both contain multiply classification tasks. The measurement of area under the curve (AUC) is used for the evaluation of the results of the models. The detailed setting of our Mol-GDL parameters can be found in Table S2.

The overall performance of Mol-GDL on classification benchmarks along with SOTAs is shown in Table 1. It can be seen that our Mol-GDL has achieved the best results and consistently outperformed the SOTAs in all the tasks except only one from MUV. Our Mol-GDL on BACE is 0.863, which is better than the previous best result (AUC = 0.856). The AUC results of the Mol-GDL model on the two datasets of HIV and MUV are 0.808 and 0.675, respectively. The results of Mol-GDL on BBBP and SIDER are significantly better than SOTAs. Among them, the performance on the SIDER dataset is the most prominent, and the AUC value of the Mol-GDL model is 20.68% higher than the previous SOTA results. The performance of the Mol-GDL model on these two datasets of Tox21 and ClinTox is also much better than the comparison methods. It is worth noting that our model not only has the largest AUC value on the ClinTox dataset (AUC = 0.966) but that it also has a much lower standard deviation (SD = 0.002) than all other existing methods.

Mol-GDL for regression tasks on the molecular property

The commonly used datasets for the regression task on the molecular property are mainly divided into two categories.[30] One is for predicting the physical and chemical properties of molecules, including ESOL, FreeSolv, and Lipo. The other is in the field of quantum chemistry, including QM7, QM8, and QM9. The detailed setting of our Mol-GDL parameters can be found in Table S3.

For a better comparison with SOTAs,[30,41] we consider two metrics, root-mean-square error (RMSE) and mean absolute error (MAE), respectively, for the two types of regression tasks. Table 2 presents the results of the Mol-GDL model and SOTAs. Generally speaking, our Mol-GDL can achieve similar or even better results than SOTA. In particular, our Mol-GDL has the smallest RMSE values on the ESOL and FreeSolv tasks. On the QM7 dataset, the MAE value of the Mol-GDL model is 62.2, second only to GEM.[41] Although the performance of Mol-GDL on the three datasets (Lipo, QM8, and QM9) did not rank among the top methods, it still beats several classical prediction methods of molecular properties, such as $N-\text{Gram}_{RF}$[38], $N-\text{Gram}_{XGB}$[38], GROVE_{base},[40] and GROVE_{large}.[40]

Mol-GDL for molecular interaction analysis

In this task, we use the dataset from DeepDDS,[42] which contains 36 anticancer drugs, 31 human cancer cell lines, and 12,415 unique drug pair-cell line combinations. Six different validation metrics are used to measure the performance of models, including area under the curve (AUC), area under the curve of precision and recall (AUPR), recall, precision, and F1-score. Since this test involves the drug combination prediction (between two drugs) in different cell lines, a slightly different architecture is used. The representation of synergistic drug combination can be calculated as

$$a^0 = concat\left(R_{drug_1}, R_{drug_2}, MLP\left(R^0_{cellline}\right)\right), \qquad \text{(Equation 4)}$$

where R_{drug_1} and R_{drug_2} represent the representations of $drug_1$ and $drug_2$ obtained by Mol-GDL and $R^0_{cellline}$ is the expression

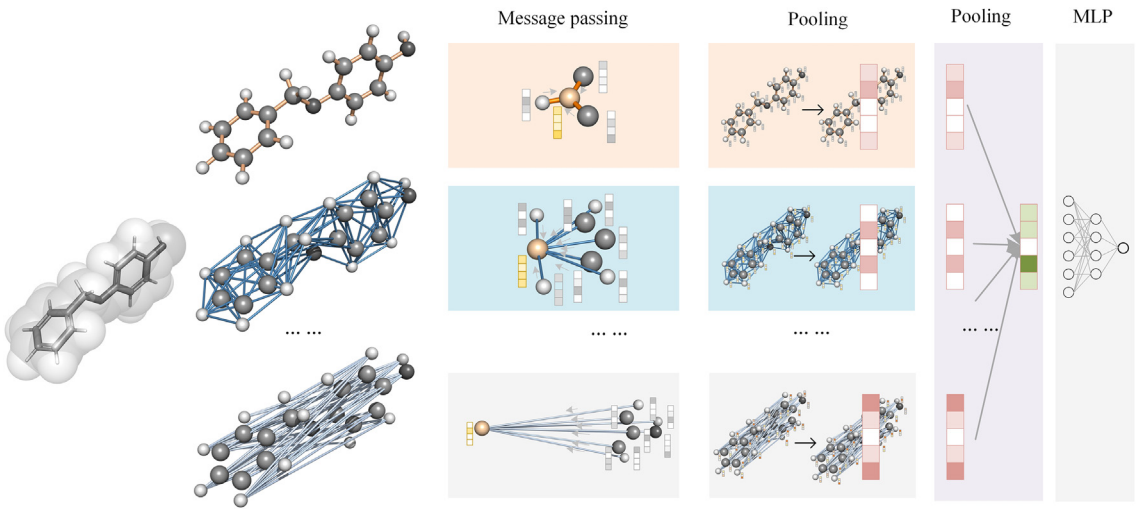


Figure 3. The flowchart of our Mol-GDL model

A set of molecular graphs are systematically constructed for each molecule. The geometric node features goes through the same message-passing module. The two pooling operations, one done within graphs and the other between groups, are used to aggregate individual node features into a single molecular feature vector. A multi-layer perceptron (MLP) is employed on the molecular feature vector to generate the final prediction.

profiling data for 954 genes, which were used as raw representations of cell lines. Here, a^0 is the representation vector concatenated from $drug_1$, $drug_2$, and cell line representations. It was then fed into an MLP to predict synergistic drug combinations

$$a^l = ReLU\left(W^l a^{l-1} + b^l\right),\ l = 1, 2, \ldots, L \qquad \text{(Equation 5)}$$

and

$$\hat{y} = softmax\left(W_{out} a^L + b_{out}\right), \qquad \text{(Equation 6)}$$

where $\hat{y}$ is the predicted value and other notations are the standard weight matrices and biases. Finally, we employ cross-entropy as the loss function to train the model.

Five deep-learning-based drug synergy prediction methods, including TranSynergy,[43] DeepSynergy,[44] deep tensor factorization (DTF),[45] $DeepDDS_{GAT}$,[42] and $DeepDDS_{GCN}$[42], and two machine learning methods, i.e., XGBoost[46] and random forest (RF),[47] are used as baselines. The detailed setting of our Mol-GDL parameters can be found in Table S4. Table 3 shows the overall performance of the Mol-GDL model in predicting synergistic drug combinations. It can be seen that our Mol-GDL model outperforms all the other methods on five validation metrics.

DISCUSSION

Incorporation of non-covalent bond information enhances performance

The role of non-covalent bonds in predicting molecular properties has long been underappreciated. In this work, we systematically compare the performance of GDLs from non-covalent-bond-based molecular graphs and covalent-bond-based molecular graphs in 13 datasets. Table S5 shows the performance of the Mol-GDL models with the molecular graphs constructed for different bond length ranges, as well as different MP architectures. It can be seen that the performance of our Mol-GDLs is better than that of the models that consider only covalent bond information on most datasets, indicating that non-covalent bonds provide a positive impact on improving the prediction performance of the model. Therefore, these results show the importance of non-covalent interactions in molecular property prediction.

Further, we consider some extreme cases to explore the effects of covalent and non-covalent bonds in molecular representations for GDLs. Three types of neighboring relations are used for the construction of molecular graphs, i.e., *n*-nearest neighbor, *n*-farthest neighbor, and *n*-random neighbor. Here, *n* is an integer number and can be taken from 1 to 7. The results are shown in Tables S6 and S7, and we have the following observations. First, as the number of neighbor nodes increases, the performance will get better and then tends to be stable. Second, with only a few farthest neighbors or random neighbors, i.e., n = 1 or 2, the model can still have comparably good prediction, in particular the classification tasks. Third, on some datasets, the prediction performance of molecular graphs constructed by the *n*-farthest neighbors is better than that of the *n*-nearest neighbors, in particular for BBBP, ClinTox, and QM7 datasets. These observations again demonstrate the importance of non-covalent interactions and the potential novel molecular representations beyond the *de facto* standard of covalent-bond-based molecular graphs.

Geometric node features enhance GDL performance

Note features are of key importance for GDLs. Here, we propose the geometric node features that are related only to atomic types and Euclidean distances. To verify the efficiency of our geometric node feature approach, we compare it with three different feature engineering methods from the learning models of AttentiveFP,[37] D-MPNN,[36,48] and DeepDDS.[42] Note that the features used in these three models are derived from the structural, physical, chemical, and biological properties, such as the

Table 1. The comparison of Mol-GDL with SOTAs on seven commonly used datasets, which contain only classification tasks on molecular properties

Dataset	BACE	BBBP	ClinTox	SIDER	tox21	HIV	MUV
No. molecules	1,513	2,039	1,478	1,427	7,831	41,127	93,087
No. average atoms	$65_{(19)}$	$46_{(21)}$	$50.58_{(31)}$	$65_{(93)}$	$36_{(23)}$	$46_{(24)}$	$43_{(10)}$
No. tasks	1	1	2	27	12	1	17
D-MPNN[36]	$0.809_{(0.006)}$	$0.710_{(0.003)}$	$0.906_{(0.006)}$	$0.570_{(0.007)}$	$0.759_{(0.007)}$	$0.771_{(0.005)}$	$0.786_{(0.014)}$
AttentiveFP[37]	$0.784_{(0.022)}$	$0.643_{(0.018)}$	$0.847_{(0.003)}$	$0.606_{(0.032)}$	$0.761_{(0.005)}$	$0.757_{(0.014)}$	$0.766_{(0.015)}$
N − $Gram_{RF}$[38]	$0.779_{(0.015)}$	$0.697_{(0.006)}$	$0.775_{(0.040)}$	$0.668_{(0.007)}$	$0.743_{(0.004)}$	$0.772_{(0.001)}$	$0.769_{(0.007)}$
N − $Gram_{XGB}$[38]	$0.791_{(0.013)}$	$0.691_{(0.008)}$	0.875	$0.655_{(0.007)}$	$0.758_{(0.009)}$	$0.787_{(0.004)}$	$0.748_{(0.002)}$
PretrainGNN[39]	$0.845_{(0.007)}$	$0.687_{(0.013)}$	$0.726_{(0.015)}$	$0.627_{(0.008)}$	$0.781_{(0.006)}$	$0.799_{(0.007)}$	$0.813_{(0.021)}$
$GROVE_{base}$[40]	$0.826_{(0.007)}$	$0.700_{(0.001)}$	$0.812_{(0.030)}$	$0.648_{(0.006)}$	$0.743_{(0.001)}$	$0.625_{(0.009)}$	$0.673_{(0.018)}$
$GROVE_{large}$[40]	$0.810_{(0.014)}$	$0.695_{(0.001)}$	$0.762_{(0.037)}$	$0.654_{(0.001)}$	$0.735_{(0.001)}$	$0.682_{(0.011)}$	$0.673_{(0.018)}$
GEM[41]	$0.856_{(0.011)}$	$0.724_{(0.004)}$	$0.901_{(0.013)}$	$0.672_{(0.004)}$	$0.781_{(0.001)}$	$0.806_{(0.009)}$	$\mathbf{0.817}_{(0.005)}$
Mol-GDL	$\mathbf{0.863}_{(0.019)}$	$\mathbf{0.728}_{(0.019)}$	$\mathbf{0.966}_{(0.002)}$	$\mathbf{0.831}_{(0.002)}$	$\mathbf{0.794}_{(0.005)}$	$\mathbf{0.808}_{(0.007)}$	$0.675_{(0.014)}$

Note that the subindex indicates standard deviation values. For instance, the element $65_{(19)}$ means the number of average atoms in BACE is 65, with 19 as its standard deviation. Bolding indicates best results.

number of atoms, normal charge, chirality, hybridization, aromaticity, etc. For a fair comparison, the same GNN architecture from our Mol-GDL is used. That is, the same Mol-GDL architecture equipped with four types of different initial node features on 11 datasets (MUV and QM9 datasets are omitted due to memory issues) is used. The results are listed in Figure 4 and Table S8. It can be seen that our geometric node features are superior to the three feature engineering approaches in almost all the datasets. In particular, our geometric node feature approach has obvious advantages in BACE, BBBP, FreeSolv, and QM7 datasets.

It should be noted that our geometric node feature approach is developed based on strong relations between molecular structures and their functions. Even though our approach contains only the atomic types and Euclidean distances, various physical, chemical, and biological information has been implicitly incorporated into it. In fact, the positions of all the atoms within a molecule are the direct reflection of the overall interactions within the molecule. Stated differently, two atoms are close to each other only when the overall interactions (from all the physical, chemical, and biological effects) between them are relatively strong. Since this interaction information is implicitly embodied into its 3D structure through the structure-function relationships, an efficient molecular representation can be obtained as long as we fully explore the structural information of the molecule. From another point of view, the length of chemical bonds is used as the node feature in our Mol-GDL model. This is essentially the process of storing the information of non-covalent bonds in the initial representation. The great performance of our approach again indicates the importance of non-covalent interactions in molecular representation.

General properties of Mol-GDL

With the highly efficient molecular representation, our Mol-GDL model achieves a better performance than SOTAs on most

Table 2. The comparison of Mol-GDL with SOTAs on six commonly used datasets, which contain only region tasks on molecular properties

	RMSE			MAE		
Dataset	ESOL	FreeSolv	Lipo	QM7	QM8	QM9
No. molecules	1,128	642	4,200	6,830	21,786	133,885
No. average atoms	$26_{(13)}$	$18_{(7)}$	$49_{(15)}$	$16_{(3)}$	$16_{(3)}$	$18_{(3)}$
No. tasks	1	1	1	1	12	12
D-MPNN[36]	$1.050_{(0.008)}$	$2.082_{(0.082)}$	$0.683_{(0.016)}$	$103.5_{(8.6)}$	$0.0190_{(0.0001)}$	$0.00814_{(0.00001)}$
AttentiveFP[37]	$0.877_{(0.029)}$	$2.073_{(0.183)}$	$0.721_{(0.001)}$	$72.0_{(2.7)}$	$0.0179_{(0.0001)}$	$0.00812_{(0.00001)}$
N − $Gram_{RF}$[38]	$1.074_{(0.107)}$	$2.688_{(0.085)}$	$0.812_{(0.028)}$	$92.8_{(4.0)}$	$0.0236_{(0.0006)}$	$0.01037_{(0.00016)}$
N − $Gram_{XGB}$[38]	$1.083_{(0.082)}$	$5.061_{(0.744)}$	$2.072_{(0.030)}$	$81.9_{(1.9)}$	$0.0215_{(0.0005)}$	$0.00964_{(0.00031)}$
PretrainGNN[39]	$1.100_{(0.006)}$	$2.764_{(0.002)}$	$0.739_{(0.003)}$	$113.2_{(0.6)}$	$0.0200_{(0.0001)}$	$0.00922_{(0.00004)}$
$GROVE_{base}$[40]	$0.983_{(0.090)}$	$2.176_{(0.052)}$	$0.817_{(0.008)}$	$94.5_{(3.8)}$	$0.0218_{(0.0004)}$	$0.00984_{(0.00055)}$
$GROVE_{large}$[40]	$0.895_{(0.017)}$	$2.272_{(0.051)}$	$0.823_{(0.010)}$	$92.0_{(0.9)}$	$0.0224_{(0.0003)}$	$0.00986_{(0.00025)}$
GEM[41]	$0.798_{(0.029)}$	$1.877_{(0.094)}$	$\mathbf{0.660}_{(0.008)}$	$\mathbf{58.9}_{(0.8)}$	$\mathbf{0.0171}_{(0.0001)}$	$\mathbf{0.00746}_{(0.00001)}$
Mol-GDL	$\mathbf{0.798}_{(0.024)}$	$\mathbf{1.809}_{(0.100)}$	$0.779_{(0.007)}$	$62.2_{(0.4)}$	$0.0205_{(0.0001)}$	$0.00952_{(0.00013)}$

Note that the subindex indicates standard deviation values. Bolding indicates best results.

Table 3. The comparison of the performance of Mol-GDL and SOTAs on synergistic drug combination dataset

Methods	Metrics				
	AUC	AUPR	Recall	Precision	F1
XGBoost[46]	$0.92_{(0.01)}$	$0.92_{(0.01)}$	$0.84_{(0.01)}$	$0.84_{(0.01)}$	$0.84_{(0.01)}$
RF[47]	$0.86_{(0.01)}$	$0.85_{(0.02)}$	$0.74_{(0.01)}$	$0.78_{(0.02)}$	$0.76_{(0.01)}$
TranSynergy[43]	$0.90_{(0.01)}$	$0.89_{(0.01)}$	$0.80_{(0.01)}$	$0.84_{(0.01)}$	$0.82_{(0.01)}$
DTF[45]	$0.89_{(0.01)}$	$0.88_{(0.01)}$	$0.77_{(0.03)}$	$0.82_{(0.01)}$	$0.80_{(0.02)}$
DeepSynergy[44]	$0.88_{(0.01)}$	$0.87_{(0.01)}$	$0.75_{(0.01)}$	$0.81_{(0.01)}$	$0.78_{(0.01)}$
$DeepDDS_{GAT}$[42]	$0.93_{(0.01)}$	$0.93_{(0.01)}$	$0.84_{(0.07)}$	$0.85_{(0.07)}$	$0.85_{(0.07)}$
$DeepDDS_{GCN}$[42]	$0.93_{(0.01)}$	$0.92_{(0.01)}$	$0.84_{(0.01)}$	$0.85_{(0.01)}$	$0.84_{(0.01)}$
Mol-GDL	$\mathbf{0.94}_{(0.01)}$	$\mathbf{0.94}_{(0.01)}$	$\mathbf{0.86}_{(0.01)}$	$\mathbf{0.86}_{(0.01)}$	$\mathbf{0.86}_{(0.01)}$

Note that the subindex indicates standard deviation values. Bolding indicates best results.

datasets. The performance of our Mol-GDL model is highly consistent and does not depend on the size of the datasets. In Tables 1 and 2, our Mol-GDL achieves the best performance in 8 of the 13 datasets. Especially on the classification task, the average performance is improved by about 3% compared with the best model GEM in the SOTA methods, and 6 of the 7 classification datasets are optimal. Note that the number of samples varies widely in datasets for these classification tasks. There are only about 1,400 molecules in ClinTox and SIDER datasets, while more than 40,000 molecules are included in the HIV dataset. Even with the huge size difference in the datasets, the performance of our Mol-GDL model is consistently better than SOTAs.

Further, the performance of our Mol-GDL model can be related to the size of the molecules. In particular, Mol-GDL may have slightly inferior results than SOTAs for small-sized molecules. In the regression task, although the Mol-GDL model has better results than most of the models, such as $N-Gram_{RF}$, $N-Gram_{XGB}$, $GROVE_{base}$, and $GROVE_{large}$, it has slightly inferior results than GEM on QM7, QM8, and QM9. The average numbers of atoms per molecule for these three datasets are about 16, 16, and 18, respectively. However, it should be noted that GEM is a pre-trained self-supervised model, and it uses a much larger dataset for the pre-training process. In contrast, our Mol-GDL uses only the regular training set, which is significantly smaller in size than the dataset for pre-training. It is worth mentioning that the other three models, i.e., PretainGNN, $GROVER_{base}$, and $GROVER_{large}$, also use the pre-training process, but their performance is still inferior to our Mol-GDL for

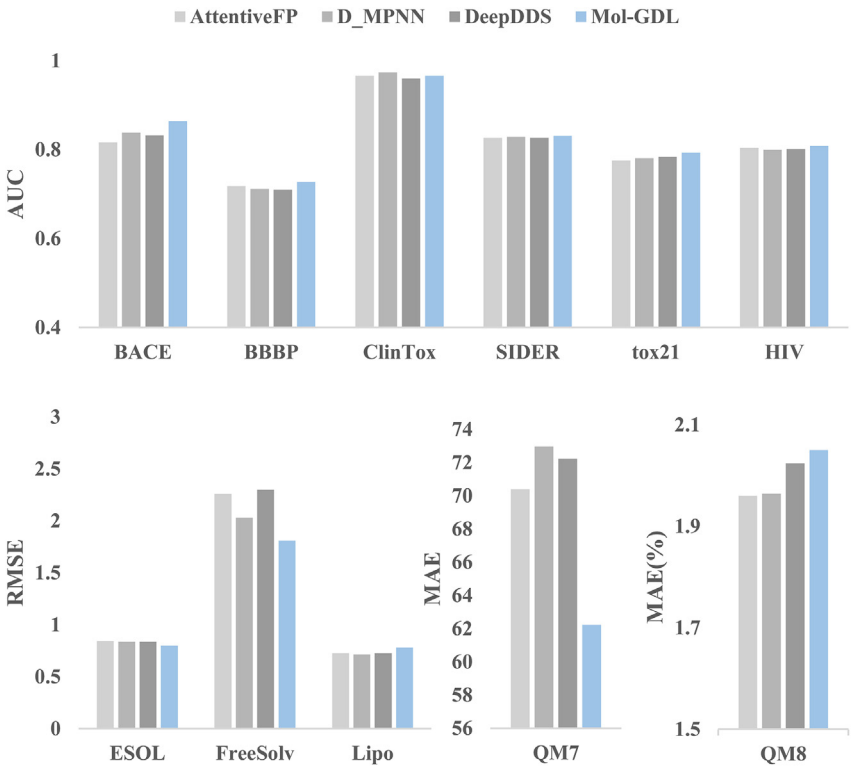


Figure 4. Comparison of different methods of calculating node features

all the tasks. To explore the effects of atomic coordinates on classification and regression tasks, we downloaded the 3D coordinates of the two datasets (Lipo and BBBP) from the website https://weilab.math.msu.edu/DataLibrary/3D/, and these were density functional theory (DFT) optimized. We replace the coordinates generated by Rdkit with these coordinates, and the results of the Mol-GDL model are shown in Table S9. It can be seen from the results that different coordinates have a greater impact on Mol-GDL.

Moreover, other than molecular property analysis, our Mol-GDL model can also achieve great performance in molecular interaction prediction. As shown in Table 3, our Mol-GDL outperforms SOTA methods for predicting synergistic drug combinations on all validation metrics. This indicates that efficient molecular graph representations are important to the analysis of not only properties at the single molecular level but also interactions between two or more molecules.

Finally, our Mol-GDL models can be further generalized or extended from several different aspects. First, we can use our Mol-GDL framework to generate a series of molecular fingerprints.[7–11] More specifically, a series of molecular descriptors can be obtained from the aggregation of node features. Second, Mol-GDL-based multi-task learning models can be constructed to further improve the performance of Mol-GDL. Third, distance GNN models have some intrinsic limitations to distinguish some special molecular structures.[49] Many-body descriptors alone are also incomplete for molecular structure representation.[50] The generalization of Mol-GDL into higher-order topological models, such as simplicial complex, polyhedron complex, hypergraph, etc., may provide ways for a more efficient and complete molecular representation.

Limitations of the study

In this article, we address the fundamental challenge of molecular representations for GDL. In particular, we demonstrate the fundamental limitations of the *de facto* standard of covalent-bond-based molecular graphs and propose novel molecular representations. However, molecular fingerprints learned from a specific task usually have a lower transferability. In our future works, we will consider multi-task learning models to enhance the transferability and further improve the performance of Mol-GDL.

STARMETHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Molecular graph representation for Mol-GDL
  - Graph neural network architecture in Mol-GDL
  - Geometric node features for Mol-GDL
  - Benchmark datasets and experimental setting
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.crmeth.2023.100621.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China (NSFC grant nos. 61873089 and 62032007), Nanyang Technological University SPMS Collaborative Research Award 2022, and the Singapore Ministry of Education Academic Research fund (Tier 2 grants MOE-T2EP20120-0013 and MOE-T2EP20221-0003), as well as the China Scholarship Council (CSC grant no. 202006130147).

AUTHOR CONTRIBUTIONS

K.X. designed the research, C.S. performed the research, K.X. and C.S. analyzed the data, C.S. wrote the initial manuscript draft, and K.X. and and J.L. subsequently revised the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: January 19, 2023
Revised: June 16, 2023
Accepted: September 28, 2023
Published: October 23, 2023

REFERENCES

1. Zhang, L., Tan, J., Han, D., and Zhu, H. (2017). From machine learning to deep learning: progress in machine intelligence for rational drug discovery. Drug Discov. Today *22*, 1680–1685.
2. Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., and Blaschke, T. (2018). The rise of deep learning in drug discovery. Drug Discov. Today *23*, 1241–1250.
3. Mak, K.K., and Pichika, M.R. (2019). Artificial intelligence in drug development: present status and future prospects. Drug Discov. Today *24*, 773–780.
4. Chan, H.C.S., Shan, H., Dahoun, T., Vogel, H., and Yuan, S. (2019). Advancing drug discovery via artificial intelligence. Trends Pharmacol. Sci. *40*. 801–604.
5. T. Puzyn, J. Leszczynski, and M.T. Cronin, eds. (2010). Recent Advances in QSAR Studies: Methods and Applications*vol. 8* (Springer Science & Business Media).
6. Lo, Y.C., Rensi, S.E., Torng, W., and Altman, R.B. (2018). Machine learning in chemoinformatics and drug discovery. Drug Discov. Today *23*, 1538–1546.
7. Merkwirth, C., and Lengauer, T. (2005). Automatic generation of complementary descriptors with molecular graph networks. J. Chem. Inf. Model. *45*, 1159–1168.
8. Duvenaud, D.K., Maclaurin, D., Iparraguirre, J., Bombarell, R., Hirzel, T., Aspuru-Guzik, A., and Adams, R.P. (2015). Convolutional networks on graphs for learning molecular fingerprints. Preprint at arXiv. https://doi.org/10.48550/arXiv.1509.09292.
9. Coley, C.W., Barzilay, R., Green, W.H., Jaakkola, T.S., and Jensen, K.F. (2017). Convolutional embedding of attributed molecular graphs for physical property prediction. J. Chem. Inf. Model. *57*, 1757–1772.
10. Xu, Y., Pei, J., and Lai, L. (2017). Deep learning based regression and multiclass models for acute oral toxicity prediction with automatic chemical feature extraction. J. Chem. Inf. Model. *57*, 2672–2685.
11. Winter, R., Montanari, F., Noé, F., and Clevert, D.A. (2019). Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. Chem. Sci. *10*, 1692–1701.

12. Wieder, O., Kohlbacher, S., Kuenemann, M., Garon, A., Ducrot, P., Seidel, T., and Langer, T. (2020). A compact review of molecular property prediction with graph neural networks. Drug Discov. Today Technol. *37*, 1–12.

13. Yu, Z., and Gao, H. (2022). Molecular graph representation learning via heterogeneous motif graph construction. Preprint at arXiv. https://doi.org/10.48550/arXiv.2202.00529.

14. Atz, K., Grisoni, F., and Schneider, G. (2021). Geometric deep learning on molecular representations. Nat. Mach. Intell. *3*, 1023–1032.

15. Li, S., Zhou, J., Xu, T., Dou, D., and Xiong, H. (2022). GeomGCL: geometric graph contrastive learning for molecular property prediction. Proc. AAAI Conf. Artif. Intell. *36*, 4541–4549.

16. Wang, Y., Wang, J., Cao, Z., and Barati Farimani, A. (2022). Molecular contrastive learning of representations via graph neural networks. Nat. Mach. Intell. *4*, 279–287.

17. You, J., Ying, R., Ren, X., Hamilton, W., and Leskovec, J. (2018). GraphRNN: Generating realistic graphs with deep auto-regressive models. In Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research PMLR, J. Dy and A. Krause, eds., pp. 5708–5717.

18. Kipf, T.N., and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. In International Conference on Learning Representations (ICLR 2017).

19. Kipf, T.N., and Welling, M. (2016). Variational Graph Auto-Encoders.. Preprint at arXiv. https://doi.org/10.48550/arXiv.1611.07308.

20. Yun, S., Jeong, M., Kim, R., Kang, J., and Kim, H.J. (2020). Graph transformer networks. Preprint at arXiv. https://doi.org/10.48550/arXiv.1911.06455.

21. Kotsias, P.C., Arús-Pous, J., Chen, H., Engkvist, O., Tyrchan, C., and Bjerrum, E.J. (2020). Direct steering of de novo molecular generation with descriptor conditional recurrent neural networks. Nat. Mach. Intell. *2*, 254–265.

22. Wang, J., Hsieh, C.Y., Wang, M., Wang, X., Wu, Z., Jiang, D., Liao, B., Zhang, X., Yang, B., He, Q., et al. (2021). Multi-constraint molecular generation based on conditional transformer, knowledge distillation and reinforcement learning. BMC Cancer *21*, 914–922.

23. Vamathevan, J., Clark, D., Czodrowski, P., Dunham, I., Ferran, E., Lee, G., et al. (2019). Applications of machine learning in drug discovery and development. Nat. Rev. Drug Discov. *18*, 463–477.

24. Wang, M., Cang, Z., and Wei, G.W. (2020). A topology-based network tree for the prediction of protein–protein binding affinity changes following mutation. Nat. Mach. Intell. *2*, 116–123.

25. Meng, Z., and Xia, K. (2021). Persistent spectral–based machine learning (PerSpect ML) for protein-ligand binding affinity prediction. Sci. Adv. *7*, eabc5329.

26. Schütt, K.T., Sauceda, H.E., Kindermans, P.J., Tkatchenko, A., and Müller, K.R. (2018). SchNet–a deep learning architecture for molecules and materials. J. Chem. Phys. *148*, 241722.

27. Gong, W., and Yan, Q. (2021). Graph-based deep learning frameworks for molecules and solid-state materials. Comp Mater Sci *195*, 110332.

28. Kim, K., Kang, S., Yoo, J., Kwon, Y., Nam, Y., Lee, D., et al. (2018). Deep-learning-based inverse design model for intelligent discovery of organic molecules. npj Comput. Mater. *4*, 67.

29. Batatia, I., Kovacs, D.P., Simm, G., Ortner, C., and Csányi, G. (2022). MACE: Higher order equivariant message passing neural networks for fast and accurate force fields. Adv. Neural Inf. Process. Syst. *35*, 11423–11436.

30. Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V., and Pande, V. (2018). MoleculeNet: a benchmark for molecular machine learning. Chem. Sci. *9*, 513–530.

31. Subramanian, G., Ramsundar, B., Pande, V., and Denny, R.A. (2016). Computational modeling of β-secretase 1 (bace-1) inhibitors using ligand based approaches. J. Chem. Inf. Model. *56*, 1936–1949.

32. Rohrer, S.G., and Baumann, K. (2009). Maximum unbiased validation (MUV) data sets for virtual screening based on pubchem bioactivity data. J. Chem. Inf. Model. *49*, 169–184.

33. Martins, I.F., Teixeira, A.L., Pinheiro, L., and Falcao, A.O. (2012). A bayesian approach to in silico blood-brain barrier penetration modeling. J. Chem. Inf. Model. *52*, 1686–1697.

34. Kuhn, M., Letunic, I., Jensen, L.J., and Bork, P. (2016). The sider database of drugs and side effects. Nucleic Acids Res. *44*, D1075–D1079.

35. Gayvert, K.M., Madhukar, N.S., and Elemento, O. (2016). A data-driven approach to predicting successes and failures of clinical trials. Cell Chem. Biol. *23*, 1294–1301.

36. Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., et al. (2019). Analyzing learned molecular representations for property prediction. J. Chem. Inf. Model. *59*, 3370–3388.

37. Xiong, Z., Wang, D., Liu, X., Zhong, F., Wan, X., Li, X., et al. (2020). Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. J. Med. Chem. *63*, 8749–8760.

38. Liu, S., Demirel, M.F., and Liang, Y. (2019). N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. Adv. Neural Inf. Process. Syst. *32*.

39. Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. (2019). Strategies for pre-training graph neural networks. Preprint at arXiv. https://doi.org/10.48550/arXiv.1905.12265.

40. Rong, Y., Bian, Y., Xu, T., Xie, W., Wei, Y., Huang, W., and Huang, J. (2020). Self-supervised graph transformer on large-scale molecular data. Adv. Neural Inf. Process. Syst. *33*, 12559–12571.

41. Fang, X., Liu, L., Lei, J., He, D., Zhang, S., Zhou, J., Wang, F., Wu, H., and Wang, H. (2022). Geometry-enhanced molecular representation learning for property prediction. Nat. Mach. Intell. *4*, 127–134.

42. Wang, J., Liu, X., Shen, S., Deng, L., and Liu, H. (2022). DeepDDS: deep graph neural network with attention mechanism to predict synergistic drug combinations. Brief. Bioinform. *23*, bbab390.

43. Liu, Q., and Xie, L. (2021). TranSynergy: Mechanism-driven interpretable deep neural network for the synergistic prediction and pathway deconvolution of drug combinations. PLoS Comput. Biol. *17*, e1008653.

44. Preuer, K., Lewis, R.P.I., Hochreiter, S., Bender, A., Bulusu, K.C., and Klambauer, G. (2018). DeepSynergy: predicting anti-cancer drug synergy with deep learning. Bioinformatics *34*, 1538–1546.

45. Sun, Z., Huang, S., Jiang, P., and Hu, P. (2020). DTF: deep tensor factorization for predicting anticancer drug synergy. Bioinformatics *36*, 4483–4489.

46. Chen, T., and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785–794.

47. Ho, T.K. (1995). Random decision forests. Proceedings of 3rd international conference on document analysis and recognition *1*, 278–282.

48. Stokes, J.M., Yang, K., Swanson, K., Jin, W., Cubillos-Ruiz, A., Donghia, N.M., et al. (2020). A deep learning approach to antibiotic discovery. Cell *180*, 688–702.e13.

49. Pozdnyakov, S.N., and Ceriotti, M. (2022). Incompleteness of graph neural networks for points clouds in three dimensions. Mach. Learn, Sci. Technol. *3*, 045020.

50. Pozdnyakov, S.N., Willatt, M.J., Bartók, A.P., Ortner, C., Csányi, G., and Ceriotti, M. (2020). Incompleteness of atomic structure representations. Phys. Rev. Lett. *125*, 166001.

51. Li, S., Wan, F., Shu, H., Jiang, T., Zhao, D., and Zeng, J. (2020). MONN: a multi-objective neural network for predicting compound-protein interactions and affinities. Cell Syst. *10*, 308–322.e11.

52. Chen, D., Gao, K., Nguyen, D.D., Chen, X., Jiang, Y., Wei, G.W., and Pan, F. (2021). Algebraic graph-assisted bidirectional transformers for molecular property prediction. Nat. Commun. *12*, 3521–3529.

STARMETHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
BACE	MoleculeNet[30]	https://moleculenet.org/datasets-1
BBBP	MoleculeNet[30]	https://moleculenet.org/datasets-1
ClinTox	MoleculeNet[30]	https://moleculenet.org/datasets-1
SIDER	MoleculeNet[30]	https://moleculenet.org/datasets-1
Tox21	MoleculeNet[30]	https://moleculenet.org/datasets-1
HIV	MoleculeNet[30]	https://moleculenet.org/datasets-1
MUV	MoleculeNet[30]	https://moleculenet.org/datasets-1
ESOL	MoleculeNet[30]	https://moleculenet.org/datasets-1
FreeSolv	MoleculeNet[30]	https://moleculenet.org/datasets-1
Lipophilicity	MoleculeNet[30]	https://moleculenet.org/datasets-1
QM7	MoleculeNet[30]	https://moleculenet.org/datasets-1
QM8	MoleculeNet[30]	https://moleculenet.org/datasets-1
QM9	MoleculeNet[30]	https://moleculenet.org/datasets-1
DeepDDS	Wang et al.[42]	https://github.com/Sinwang404/DeepDDS/tree/master
Software and algorithms		
D-MPNN	Yang et al.[36]	https://github.com/chemprop/chemprop
AttentiveFP	Xiong et al.[37]	https://github.com/OpenDrugAI/AttentiveFP
N − $Gram_{RF}$	Liu et al.[38]	https://github.com/chao1224/n_gram_graph
N − $Gram_{XGB}$	Liu et al.[38]	https://github.com/chao1224/n_gram_graph
PretrainGNN	Hu et al.[39]	http://snap.stanford.edu/gnn-pretrain
$GROVE_{base}$	Rong et al.[40]	https://github.com/tencent-ailab/grover
$GROVE_{large}$	Rong et al.[40]	https://github.com/tencent-ailab/grover
GEM	Fang et al.[41]	https://github.com/PaddlePaddle/PaddleHelix/tree/dev/apps/pretrained_compound/ChemRL/GEM
Mol-GDL	This paper	https://github.com/CS-BIO/Mol-GDL https://zenodo.org/record/8304176

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Kelin Xia (xiakelin@ntu.edu.sg).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- All data used in this study is publicly available. The accession numbers for the datasets are listed in the Key resources table.
- Original code generated for this study is available at Github [https://github.com/CS-BIO/Mol-GDL]. An archival DOI is listed in the Key resources table.
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

CellPress
OPEN ACCESS

METHOD DETAILS

Molecular graph representation for Mol-GDL

Graph neural networks can encode molecular structural information into a low-dimensional representation vector from a high-dimensional space. It is a popular method to regard the atoms in the molecule as nodes and the chemical bonds as edges in the graph, and then use the method of message passing to predict the properties of the molecule.[48,51] However, ignoring the quantitative distances between atoms in traditional graphs may result in the loss of a great deal of critical physical and chemical information in molecules. Molecular graphs reconstructed from specific distance intervals can capture some different physicochemical and biophysical patterns, such as hydrogen bonding and van der Waals interactions between different atoms.[25,52]

A molecule can be represented as a graph $G(I) = (V, E(I))$, where $E(I)$ is an edge set and V is a node set. I is a certain interaction region. Physically, the covalent bonds are usually within the interaction region of $[0, 2]$. Non-covalent interactions are comparably weak and are within a much larger range. For instance, the distance between donor and acceptor atoms in hydrogen bonds, is roughly between $[2.0, 4.0]$. van der Waals forces can cover a much larger range and play an very important role in the range $[4.0, 6.0]$. The electrostatic forces can go further to $[6.0, 8]$. To fully explore the physicochemical and biophysical patterns in non-covalent interactions, we divide the interval region greater than $2\mathring{A}$ into several non-overlapping regions. For each interaction region, a corresponding molecular graph is generated and its associated adjacency matrix is defined as Equation 1.

To balance the computational costs and model accuracy, a total of 4 segments are considered for non-covalent interactions, namely $I_1 = [2, 4), I_2 = [4, 6), I_3 = [6, 8), I_4 = [8, \infty)$. Note that the selection of interaction regions to build molecular graphs is not unique. For convenience, we let $I_0 = [0, 2)$ denotes the interval of covalent interactions. In this way, total 5 interaction regions are considered and a total of 5 molecular graphs, denoted as $G = \{G_1, G_2, G_3, G_4, G_5\}$, are generated for each molecular structure, in all of our three types of tasks except those from QM7, QM8, and QM9. Since the number of atoms in the three datasets QM7, QM8, and QM9 is relatively small, the number of non-covalent interactions greater than 6Å is very small. Therefore, we divide the non-covalent interaction interval for these three datasets into three segments, namely $I_1 = [2, 4), I_2 = [4, 6), I_3 = [6, \infty)$. Similarly, let $I_0 = [0, 2)$ represent the interval of covalent bonds. In this way, total 4 interaction regions are considered and a total of 4 molecular graphs, denoted as $G = \{G_1, G_2, G_3, G_4\}$, are generated for each molecular structure from QM7, QM8, and QM9.

Graph neural network architecture in Mol-GDL

For a molecular $G(I_k)$, denoted as G_k for short, we use message passing to learn the feature representation of each node,

$$\boldsymbol{h}_i^{G_k,(t)} = \sigma\left(\sum_{j \in \mathcal{N}(i)\cup\{i\}} \frac{1}{\sqrt{d(i)}\cdot\sqrt{d(j)}} \cdot \left(W_{G_k} \cdot \boldsymbol{h}_j^{G_k,(t-1)}\right)\right) \quad \text{(Equation 7)}$$

In the t-th iteration, the i-th node feature vector $h_i^{G_k,(t)}$ of a graph G_k is obtained by gathering node feature vectors of its neighbors, denoted as $\mathcal{N}(i)$, and itself, from $(t-1)$-th iteration. Here $d(\cdot)$ represents the node degree and W_{G_k} is the weight matrix (to be learned). Computationally, we usually repeat the process 1 to 3 times, and the final node feature is denoted as $\boldsymbol{h}_i^{G_k}$.

After the message passing, we aggregate all node features in a molecular graph through a pooling process,

$$\boldsymbol{h}_{G_k} = Pooling\left(\boldsymbol{h}_i^{G_k} | i \in [1, 2, \ldots, N]\right) \quad \text{(Equation 8)}$$

where $Pooling(\cdot)$ is a pooling function, such as max pooling, mean pooling, etc. Note that this pooling process is within the single molecular graph.

In our Mol-GDL, each molecule will have more than one molecular graph, we aggregate all the features from its molecular graphs,

$$\boldsymbol{h}'_{G'_k} = \sigma\left(W'_{G'_k}\boldsymbol{h}_{G_k} + b'_{G'_k}\right) \quad \text{(Equation 9)}$$

$$\boldsymbol{h}_G = READOUT\left(\boldsymbol{h}'_{G'_k} \middle| k \in \{1, 2, \ldots, 5\}\right) \quad \text{(Equation 10)}$$

where $READOUT(\cdot)$ denotes a pooling function, and we choose concatenation (or mean) in this study.

Finally, a multiply layer perceptron (MLP) is utilized for the final prediction,

$$\widehat{y} = \delta\left(W^1\sigma\left(W^0\boldsymbol{h}_G + b^0\right) + b^1\right) \quad \text{(Equation 11)}$$

where $\sigma(\cdot)$ and $\delta(\cdot)$ are $Relu(\cdot)$ and $Sigmoid(\cdot)$, respectively. For regression tasks, $\delta(\cdot)$ is a linear activation function. The ℓ_1 loss and cross-entropy loss are implemented for regression and classification tasks, respectively.

Geometric node features for Mol-GDL

The geometric node features are of great importance for GDLs. In our Mol-GDL geometric node features that contain only atomic types and Euclidean distance information. Computationally, we consider a total of 12 types of atoms, including *C*, *H*, *O*, *N*, *P*, *Cl*, *F*, *Br*, *S*, S_i, *I* and all the rest atoms as one type. These atoms are chosen due to their high frequencies in the molecules in our datasets. For the *i*-th atom, atom type α_j will contribute a component $f_i(I, \alpha_j)$ in the geometric node feature $f_i(I) = [f_i(I, \alpha_1), f_i(I, \alpha_2), \ldots, f_i(I, \alpha_{12})]$. Here $f_i(I, \alpha_j)$ means the number (or frequency) of all the neighboring atoms of type α_j for the *i*-th atom. Further, different subdivision of interaction region *I* is done for different types of atoms. More specifically, for *C* (and *H*), its node feature vector components are of the size 19 after subdivision. That is the component $f_i(I, C)$ has been extended to a vector $[f_i(I^1, C), f_i(I^2, C), \ldots, f_i(I^{19}, C)]$. Here $f_i(I^1, C)$ is the number of all the neighboring atoms of type α_j within the interaction subregion I^1 for the *i*-th atom. For *O* and *N*, their node feature components are 4 after subdivision. All the rest types of atoms have the same node feature components of size 2 after subdivision. The detailed information for the subdivision of the interaction region (based on different atom types) is listed in Table S10.

Benchmark datasets and experimental setting

In this work, the 13 datasets used for molecular property prediction are all derived from MoleculeNet.[30] The detailed information of these 13 datasets can be found in literature.[30,36] The general information of these 13 datasets is shown in Table S1. It is worth noting that No. atoms represents the average number of atoms in each molecule in the corresponding dataset, with the standard deviation in parentheses. In addition, the 3D coordinate indicates the source of the atomic coordinates in each dataset. As can be seen from the table, except for the three datasets QM7, QM8 and QM9, the coordinates of other datasets are obtained through RDKIT (https://www.rdkit.org/). It is worth noting that both the node features and the original graph in Mol-GDL heavily depend on the accuracy of the atomic 3D coordinates, so accurate atomic 3D coordinates play an important role in ensuring the performance of the Mol-GDL model. For these 13 benchmarks, we adopt the scaffold split method[39] to create a train, valid and test split with a ratio of 8:1:1. Unlike random split, the principle of scaffold split is to reasonably distribute the substructures of molecules in the train, valid and test set. This splitting method is more realistic and more challenging.

QUANTIFICATION AND STATISTICAL ANALYSIS

Further, different metrics are employed for model evaluation. More specifically, two metrics, i.e., RMSE and MAE are used for regression tasks. For classification tasks, five commonly used evaluation metrics, namely AUC, AUPR, Recall, Precision and F-score, are considered. The detailed explanation of these metrics can be found below.

RMSE is the abbreviation of root mean squared error, and its definition is as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad \text{(Equation 12)}$$

where y_i and $\hat{y}_i$ represent the true value and predicted value of ith sample respectively.

MAE is the abbreviation of mean absolute error, and its definition is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad \text{(Equation 13)}$$

where y_i and $\hat{y}_i$ represent the true value and predicted value of ith sample respectively.

For classification tasks, AUC and AUPR are the most common metrics used to evaluate the performance of classification tasks. If you want to calculate AUC, you must first calculate true positive rate (TPR) and false positive rate (FPR). The calculation process is as follows:

$$TPR = \frac{true\ \ positive}{true\ \ positive + false\ \ negative} \quad \text{(Equation 14)}$$

$$FPR = \frac{false\ \ positive}{false\ \ positive + true\ \ negative} \quad \text{(Equation 15)}$$

where the curve composed of TPR and FPR is called ROC, and the area under the receiver operating characteristic (ROC) curve, namely area under the receiver operating characteristic convex hull (AUC-ROC), also called AUC.

AUPR is the area under the curve composed of precision and recall, where precision and recall are calculated as follows:

$$precision = \frac{true\ \ positive}{true\ \ positive + false\ \ positive} \quad \text{(Equation 16)}$$

$$recall = \frac{true\ \ positive}{true\ \ positive + false\ \ negative} \quad \text{(Equation 17)}$$

where the precision and recall can calculate F1-score, which is also an important evaluation metric for evaluating classification performance. Its calculation method is as follows,

$$F1 - score = 2 * \frac{precision * recall}{precision + recall} \quad \text{(Equation 18)}$$