Path Complex Neural Network for Molecular Property Prediction

Longlong Li[1 2 3] **Xiang Liu**[4] **Guanghui Wang**[1] **Yuguang Wang**[5] **Kelin Xia**[3]

Abstract

Enormous power has been demonstrated by geometric deep learning (GDL) in molecular data analysis. However, there are still challenges in achieving high efficiency and expressivity in molecular representations, which are fundamental for the success of GDL. In this work, we introduce path complex neural network (PCNN) model for molecular property prediction. The essential idea is to use path complices to characterize various types of molecular interactions specified in molecular dynamic (MD) force field. We propose a path complex message-passing module to allow the communication of simplex features within/between different dimensions. Our model has been extensively validated on benchmark datasets and can achieve the state-of-the-art results.

1. Introduction

Effectively predicting molecular properties is of paramount importance in the fields of drug design (Zhang et al., 2017; Chen et al., 2018; Mak & Pichika, 2019; Chan et al., 2019), biology (Townshend et al., 2021; Jamasb et al., 2022), chemistry (Qiao et al., 2022), and materials (Vlassis et al., 2020). As Geometric Deep Learning (GDL) has demonstrated tremendous potential and power in molecular sciences, an increasing number of studies have employed GDL models for effective representation learning of molecules (Bronstein et al., 2017; Atz et al., 2021; Ingraham et al., 2023). Due to its simplicity, flexibility, and efficiency, the molecular graph (Wieder et al., 2020; Yu & Gao, 2022; Atz et al., 2021; Li et al., 2022; Wang et al., 2022b) has become the most popular among the three types of representations (topological, geometric, and functional) used to describe molecules in GDL models. However, relying solely on graphical representations fails to capture the many-body interactions present in many complex systems, and the expressiveness of this approach has been proven to be limited (Bodnar et al., 2021).

In a common GNN model, the covalent-bond based graph is used as the de facto molecular graph. Node features are usually selected from atomic properties and further updated with aggregating information from neighboring nodes (Huang et al., 2020; Shindo & Matsumoto, 2019; Shui & Karypis, 2020a; Schütt et al., 2017; Unke & Meuwly, 2019). To enhance GNN performance, three major approaches have been proposed. The first approach is to devise more complicated molecular graphs to incorporate non-covalent interactions. The most popular way is to use a cutoff distance and edges are generated between any two atoms within the cutoff distance. Further, molecule-based line graph model has been developed with nodes representing atom bonds and edges representing bond angles (Choudhary & DeCost, 2021). The second approach is to consider global physical features and local geometric information. Global physical attributes such as temperature, pressure, entropy, etc, have been added into GNN architectures for a better characterization the molecular states and environments in MEGNet (Chen et al., 2019) and SphereNet (Liu et al., 2022). Local geometric features, in particular bond lengths, bond angles (Schütt et al., 2018; Flam-Shepherd et al., 2021), dihedral angles (Wang et al., 2022a), and torsion angles which are key to molecular properties, have been extensively considered in models such as DimNet (Gasteiger et al., 2020), GemNet (Gasteiger et al., 2021), ALIGNN (Choudhary & DeCost, 2021) and GEM (Fang et al., 2022). The third approach is to design efficient message-passing modules for invariant features, equivalent properties, and higher order tensors. The GNN expressivity is tightly related to message-passing modules used in the invariant/equivalent/higher-order-tenser layers. The above three approaches are synergistically integrated with each other.

In this work, we develop path complex-based molecular representation and path complex neural network (PCNN) model for molecular property analysis, as in Figure 1. Our path complices are specially designed, based on de facto

[1]School of Mathematics, Shandong University, Jinan 250100, China. [2]Data Science Institute, Shandong University Jinan 250100, China. [3]Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371, Singapore. [4]Chern Institute of Mathematics, Nankai University, Tianjin 300071, China [5]Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai 200240, China.. Correspondence to: Kelin Xia <xiakelin@ntu.edu.sg>.

Proceedings of the Geometry-grounded Representation Learning and Generative Modeling at 41st International Conference on Machine Learning, Vienna, Austria. PMLR Vol Number, 2024. Copyright 2024 by the author(s).

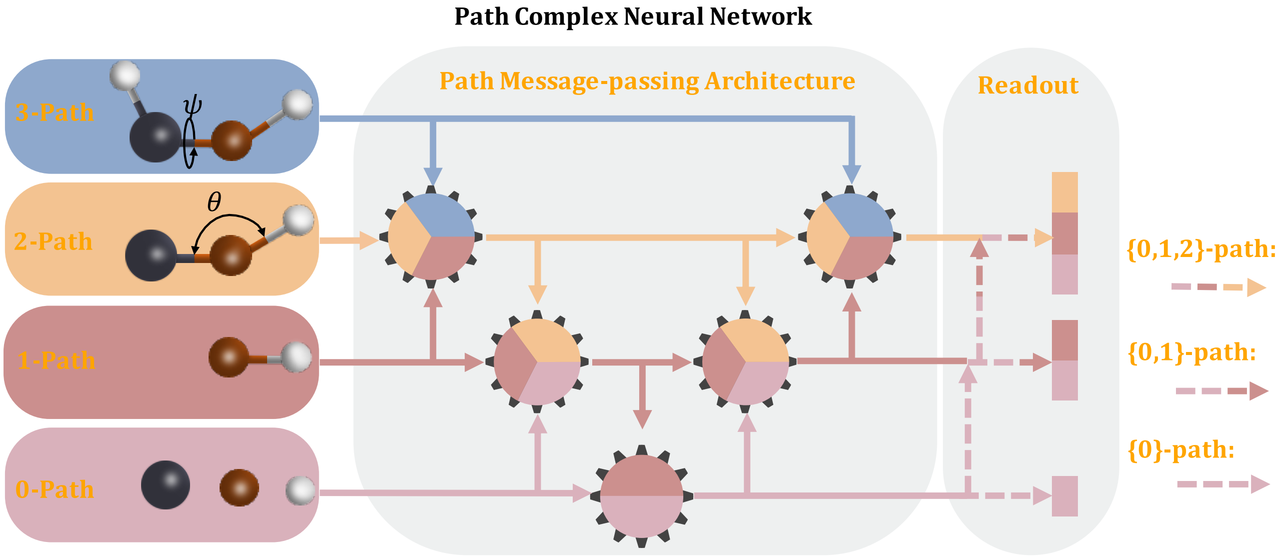

Figure 1. Architecture of PCNN. Each gear component updates its representation in the current dimension by integrating both high-order and low-order information. Up to 3-path is included to Three different "Readout" modules are denoted as $\mathbf{R_1}$, $\mathbf{R_2}$, and $\mathbf{R_3}$. Depending on the model setting, one of the "Readout" module will be selected.

covalent-bond molecular graph, to characterize different types of energy specified in molecular dynamic (MD) force field. More specifically, the MD potential energy (Mayo et al., 1990; González, 2011; Leach, 2001) contains bond term (E_B, two-body), bond-angle term (E_A, three-body), and dihedral-angle term (E_T, four-body), which are well characterized by our 1-path, 2-path, and 3-path features, respectively. We design efficient path-complex message-passing to allow the information passing between simplex features (at different dimensions), and use the aggregated information to predict molecular properties. Testing on benchmark datasets has shown promising performance. Our contributions are as follows:

1. For the first time, we develop path complex-based molecular representation that can explicitly characterize different terms in molecular dynamic (MD) force field.

2. We propose a path Weisfeiler-Lehman (PWL) test for distinguishing non-isomorphic path complexes, based on the theoretical results, we developed path complex neural network (PCNN) model for molecular property analysis. We then show that PCNN is as powerful as PWL and strictly better than WL test.

3. Our PCNN model has been extensively tested and validated on benchmark datasets. It has been found that our model can achieve state-of-the-art results.

2. Related Work

2.1. Graph Neural Networks for Molecular Property Prediction

Graph neural network models have played an pivotal role in molecular data analysis. Traditional GNN models represent molecules as the de factor covalent-bond-based molecular graphs, and use major GNN architectures, such as GIN (Xu et al., 2018), GAT (Velickovic et al., 2017), GCN (Kipf & Welling, 2016), SGCN (Danel et al., 2020) and GTtransformer (Rong et al., 2020), to learn molecular properties (Yang et al., 2019; Xiong et al., 2019; Choudhary & DeCost, 2021; Fang et al., 2022). With the importance of non-covalent bonds, cutoff-distance-based molecular graph representations have been widely employed in GNN models, such as DimeNet (Gasteiger et al., 2020), HMGNN (Shui & Karypis, 2020b), GeoGNN (Fang et al., 2022), Mol-GDL (Shen et al., 2023), etc. Further, higher-order interactions (beyond pair-wise forces) has been explicitly incorporated into GNN models, including ALIGNN (Choudhary & DeCost, 2021), GEM (Fang et al., 2022), DimeNet (Gasteiger et al., 2020), GemNet (Gasteiger et al., 2021), etc, by the consideration of bond angles, dihedral angles, torsion angles, and other local geometric information. In particular, these higher-order terms can be directly related to MD force field information (Halgren, 1996; Choudhary et al., 2018). Finally, pre-training process has been adopted to further improve the accuracy of GNN models, such as N-Gram (Liu et al., 2019), PretrainGNN (Hu et al., 2019), GEM (Fang et al., 2022), MolCLR (Wang et al., 2022b), DMP (Zhu

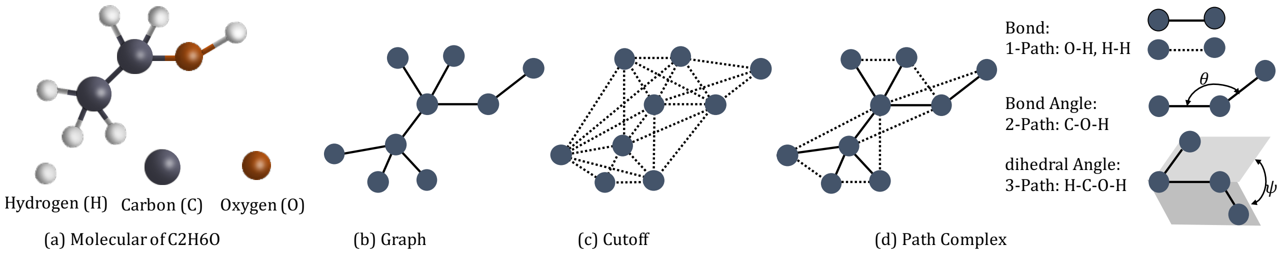

Figure 2. Different Representations of the C2H6O Molecule. Figure (a) displays the molecular structure of C2H6O, including the oxygen (O), carbon (C), and hydrogen (H) atoms. Figure (b) shows the graph representation based on chemical bonds; figure (c) illustrates the nearly fully connected graph generated based on a distance threshold (cutoff); and figure (d) presents the representation using the path complex method and its physical implications. In the diagrams, solid lines represent chemical bonds, while dashed lines represent cutoff connections.

et al., 2023), etc.

2.2. Topological Deep Learning (TDL)

Topological Deep Learning (TDL) (Hajij et al., 2022; Bodnar, 2022) leverages novel topological tools to characterize data with complicated higher-order structures. Different from graph-based data representation, TDL uses topological representations from algebraic topology, including simplicial complexes (Bodnar, 2022; Schaub et al., 2022), cell complexes (Hajij et al., 2020; Roddenberry et al., 2022; Giusti et al., 2023), sheaves (Hansen & Ghrist, 2019; Bodnar et al., 2021), hypergraphs (Feng et al., 2019; Kim et al., 2020; Bai et al., 2021), and combinatorial complexes (Hajij et al., 2022) to model not only pair-wise interactions (as in graphs), but also higher-order interactions among three or more elements. In fact, these algebraic topology-based molecular representations have already achieved great success in molecular data analysis, including protein flexibility and dynamic analysis (Xia & Wei, 2014; Sverrisson et al., 2021), drug design (Cang & Wei, 2017), virus analysis (Chen et al., 2022), materials property analysis (Reiser et al., 2022; Townsend et al., 2020). Further, TDL uses a generalized message-passing mechanism thus enables the communication of information from simplices of different dimensions. In contrast to GNNs, where information is passing among nodes or edges, TDL allows information to propagate through any neighborhood relation (Roddenberry et al., 2021).

Recently, path complex and its related models, including path homology (Grigor'yan et al., 2018), persistent path homology (Chowdhury & Mémoli, 2018; Liu et al., 2023; Chen et al., 2023), and path Laplacian (Wang & Wei, 2023), have been developed and demonstrated great potential for the analysis of molecular structures. However, there is still a lack of effective models that integrate path (simplex) complex with deep learning architecture in molecular domain.

3. Path Complex Neural Network

Path complex was originally developed on directed graph (or digraph) and set, by Grigoryan, Lin, Muranov and Yau in 2012 (Grigor'yan et al., 2012). They also proposed a new homology theory for path complex, called path homology, and use it to explore topological invariant information of digraphs (Grigor'yan et al., 2014). Mathematically, path homology provides a novel framework to systematically explore intrinsic topological information of more general structures (Grigor'yan et al., 2019; Grigor'yan et al., 2020).

3.1. Generalized Path Complex

Definition 3.1 (Path)**.** Given a simple undirected graph $G = (V, E)$ over the vertex set V, an n-path σ_n of G is defined as any sequence of $n+1$ vertices $v_0 v_1 \cdots v_n (v_i \in V)$ such that every two vertices are distinct and every two adjacent vertices form an edge.

Note that for each n-path $\sigma_n = v_0 v_1 \cdots v_n$, $\sigma'_n = v_n \cdots v_1 v_0$ is also an n-path, we identify these two paths as the same one. For an n-path $\sigma_n = v_0 \cdots v_n$, the $(n-1)$-paths by removing the first or last vertex, denoted by $\partial^L_{\sigma_n}$ and $\partial^R_{\sigma_n}$ respectively, are called the faces of σ_n. The n-path τ_n is called a coface of $(n-1)$-path σ_{n-1} if σ_{n-1} is a face of τ_n. Two n-paths are upper adjacent if they are faces of a common $(n+1)$-path, lower adjacent if they have a common $(n-1)$-path as face. For an n-path σ_n, let $\mathcal{B}(\sigma_n)$ be the set of faces of σ_n, $\mathcal{C}(\sigma_n)$ be the set of cofaces of σ_n, $\mathcal{N}_{\uparrow}(\sigma_n)$ be the set of n-paths that are upper adjacent with σ_n, $\mathcal{N}_{\downarrow}(\sigma_n)$ be the set of n-paths that are lower adjacent with σ_n. We can use the above four relations to define the neighbors of an n-path σ_n.

3.2. Path WL Test

The deep theoretical connection between the Weisfeiler-Lehman (WL) graph isomorphism test and message-passing

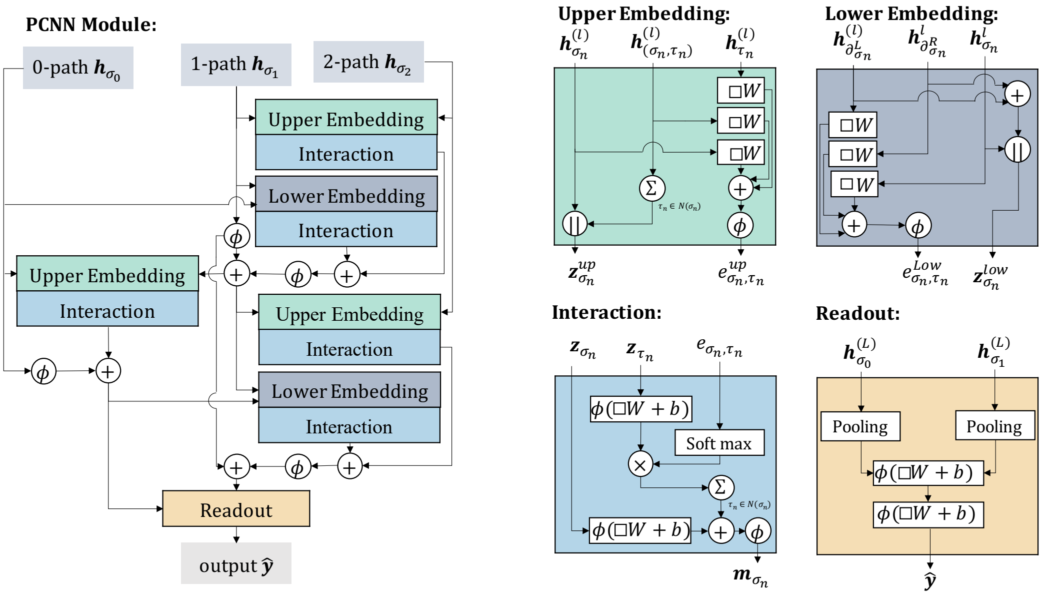

Figure 3. The PCNN Module. □ denotes the layer's input, || concatenation, and ϕ a non-linearity. Upper embedding and Upper interaction refers to utilizing high-order path features to update low-order path features, while Lower embedding and Lower interaction refers to using low-order path features to update high-order path features.

graph neural networks (GNNs) is well-documented (Xu et al., 2018). Leveraging this relationship, we introduce a path complex version of the WL test, aiming to create a message-passing procedure that maintains the test's expressive power. We term this approach the Path Weisfeiler Lehman (PWL) Test, with details provided in Appendix B.

Definition 3.2 (PWL)**.** The steps of general PWL are as follows:

1. Given a path complex P, all the paths of P are initialized with the same color.

2. For the color c^t_σ of path σ at iteration t, the color c^{t+1}_σ of σ at the next iteration is computed by perfectly hashing the color multi-set of the neighbors of σ.

3. The algorithm stops once a stable coloring is reached. Two path complexes are considered non-isomorphic if their color histograms are different at some dimensions.

Based on the four neighbor definitions, including face neighbor $\mathcal{B}(\sigma)$, coface neighbor $\mathcal{C}(\sigma)$, upper adjacent neighbor $\mathcal{N}_\uparrow(\sigma)$ and lower adjacent neighbor $\mathcal{N}_\downarrow(\sigma)$, we have four types of neighbor color multi-sets. Let c^t be the coloring of PWL for path complex P at iteration t, four types of color multi-sets are as follows

1. $c^t_{\mathcal{B}}(\sigma) = \{\{c^t_\tau | \tau \in \mathcal{B}(\sigma)\}\}$
2. $c^t_{\mathcal{C}}(\sigma) = \{\{c^t_\tau | \tau \in \mathcal{C}(\sigma)\}$
3. $c^t_\uparrow(\sigma) = \{\{(c^t_\tau, c^t_{\sigma \cup \tau}) | \tau \in \mathcal{N}_\uparrow(\sigma)\}$
4. $c^t_\downarrow(\sigma) = \{\{(c^t_\tau, c^t_{\sigma \cap \tau}) | \tau \in \mathcal{N}_\downarrow(\sigma)\}$

Having the neighbor color multi-sets, we obtain the following update rule that contains all four types of neighbors:

$$c^{t+1}_\sigma = \mathrm{HASH}\{c^t_\sigma, c^t_{\mathcal{B}}(\sigma), c^t_{\mathcal{C}}(\sigma), c^t_\uparrow(\sigma), c^t_\downarrow(\sigma)\}$$

Actually, certain neighbors can be removed without affecting the expressive power of PWL test in terms of path complex that can be differentiated.

Theorem 3.3. *PWL with* $\mathrm{HASH}\{c^t_\sigma, c^t_{\mathcal{B}}(\sigma), c^t_\uparrow(\sigma)\}$ *is as powerful as PWL with the updating strategy* $\mathrm{HASH}\{c^t_\sigma, c^t_{\mathcal{B}}(\sigma), c^t_{\mathcal{C}}(\sigma), c^t_\uparrow(\sigma), c^t_\downarrow(\sigma)\}$.

Theorem 3.4. *PWL is strictly more powerful than WL.*

3.3. Molecular PCNN model

Molecular Path Complex Representation An illustration of our path simplices and their relations with bond terms can be found in Figure 2. Here we consider the C_2H_6O molecule and its path simplices. Table 2 (in the Appendix A.1) presents a comprehensive listing of our 1-path, 2-path, and 3-path features.

General Path Complex Neural Network We introduce a comprehensive Path Complex Neural Network (PCNN) that employs the specified message-passing operations. For detailed information on the modules within the PCNN, please

Table 1. Comparison with GNN architectures. The best performance is indicated as **bold**, and the subindex indicates standard deviation values. * indicates that the result is not available for the model.

Method		**QM7**	**QM9**	**Tox21**	**HIV**	**MUV**
GNN	GIN	$110.3_{(7.2)}$	$0.00886_{(0.00005)}$	*	*	*
	GAT	$103.0_{(4.4)}$	$0.01117_{(0.00018)}$	*	*	*
	GCN	$100.0_{(3.8)}$	$0.00923_{(0.00019)}$	*	*	*
	D-MPNN	$103.5_{(8.6)}$	$0.00812_{(0.00009)}$	$0.759_{(0.007)}$	$0.771_{(0.005)}$	$0.786_{(0.014)}$
	Attentive FP	$72.0_{(2.7)}$	$0.00812_{(0.00001)}$	$0.761_{(0.005)}$	$0.757_{(0.014)}$	$0.766_{(0.015)}$
	GTransformer	$161.3_{(7.1)}$	$0.00923_{(0.00019)}$	*	*	*
	SGCN	$131.3_{(11.6)}$	$0.01459_{(0.00055)}$	*	*	*
	DimNet	$95.6\ _{(4.1)}$	$0.01031_{(0.00076)}$	*	*	*
	HMGNN	$101.6\ _{(3.2)}$	$0.01239_{(0.00001)}$	*	*	*
	Mol-GDL	$62.2_{(0.4)}$	$0.00952_{(0.00013)}$	$0.791_{(0.005)}$	$0.808_{(0.007)}$	$0.675_{(0.014)}$
Pretrain_GNN	N-Gram$_{\text{RF}}$	$92.8_{(4.0)}$	$0.01037_{(0.00016)}$	$0.758_{(0.009)}$	$0.787_{(0.004)}$	$0.748_{(0.002)}$
	N-Gram$_{\text{XGB}}$	$81.9_{(1.9)}$	$0.00964_{(0.00031)}$	$0.758_{(0.009)}$	$0.787_{(0.004)}$	$0.748_{(0.002)}$
	PretrainGNN	$113.2_{(0.6)}$	$0.00922_{(0.00004)}$	$0.781_{(0.006)}$	$0.799_{(0.007)}$	$0.813_{(0.021)}$
	GROVER$_{\text{base}}$	$94.5_{(3.8)}$	$0.00986_{(0.00055)}$	$0.743_{(0.001)}$	$0.625_{(0.009)}$	$0.673_{(0.018)}$
	GROVER$_{\text{large}}$	$92.0_{(0.9)}$	$0.00986_{(0.00025)}$	$0.735_{(0.001)}$	$0.682_{(0.011)}$	$0.673_{(0.018)}$
	MolCLR	*	*	*	$0.750\ _{(0.002)}$	$0.796\ _{(0.019)}$
	GEM	$58.9_{(0.8)}$	$0.00746_{(0.00001)}$	$0.781_{(0.001)}$	$0.806_{(0.009)}$	$0.817_{(0.005)}$
	DMP	$74.4_{(1.2)}$	*	$0.791_{(0.004)}$	$0.814_{(0.004)}$	*
	SMPT	*	*	$0.797\ _{(0.001)}$	$0.812\ _{(0.001)}$	$0.822\ _{(0.008)}$
PCNN		$\mathbf{53.9}_{(2.1)}$	$\mathbf{0.00685}_{(0.00005)}$	$\mathbf{0.801}_{(0.002)}$	$\mathbf{0.823}_{(0.004)}$	$\mathbf{0.827}_{(0.015)}$

refer to Figure 3. For a path σ in P, we have

$$m_{\mathcal{B}}^{t+1}(\sigma) = AGG_{\tau \in \mathcal{B}(\sigma)}(M_{\mathcal{B}}(h_{\sigma}^{t}, h_{\tau}^{t})) \tag{1}$$

$$m_{\uparrow}^{t+1}(\sigma) = AGG_{\tau \in \mathcal{N}_{\uparrow}(\sigma)}(M_{\uparrow}(h_{\sigma}^{t}, h_{\tau}^{t}, h_{\sigma \cup \tau}^{t})) \tag{2}$$

Then, the updating function considers these two types of messages and the previous color of σ:

$$h^{t+1}(\sigma) = U(h_{\sigma}^{t}, m_{\mathcal{B}}^{t}(\sigma), m_{\uparrow}^{t}(\sigma)) \tag{3}$$

After L layers of the message passing process, the readout function takes the color multi-sets at all dimensions as input:

$$h_P = \text{READOUT}(\{\{h_{\sigma}^{L}\}\}_{dim(\sigma)=0}, \cdots, \{\{h_{\tau}^{L}\}\}_{dim(\tau)=p}) \tag{4}$$

Theorem 3.5. *A Path Complex Neural Network (PCNN) with sufficient layers and injective neighborhood aggregators achieves the same expressive power as the PWL.*

4. Experiments

4.1. Results

The comparison of our PCNN with existing models, on benchmark datasets are illustrated in Table 1. Detailed Benchmark Models and Hyperparameters can be found in Section A.3 (in Appendix A). Our PCNN model demonstrates significant performance advantages on the most datasets, primarily due to its advanced feature expression capabilities and enhanced recognition of complex molecular structures. The message-passing mechanism within the PCNN is structured into two distinct layers: the upper embedding, which considers upper adjacent neighbors, and the lower embedding, which focuses on upper adjacent neighbors and face neighbors. This dual-layered approach enables the integration of path information from multiple perspectives, theoretically improving the model's proficiency in managing both local and global structures of the graph. Each path is not only updated based on the features of its constituent nodes but also incorporates information from both higher-order and lower-order connected paths. This sophisticated mechanism aids the model in detecting subtle structural variations within molecules that are typically challenging to differentiate.

5. Conclusion

In this study, we presented the PCNN model, a novel molecular structure representation based on path complexes, designed for predicting molecular properties. This approach integrates force fields with path complexes, enhancing our understanding of the molecular structure-function relationship, and supports theoretical and practical applications in molecular design and materials science. The PCNN model utilizes 0-paths for atomic properties, 1-paths for pairwise interactions, 2-paths for bond angle terms, and 3-paths for dihedral angle information. It leverages these paths to compute attention scores, facilitating effective message propagation and feature integration across different informational levels. Validation on benchmark datasets confirmed PCNN's superior predictive abilities.

References

Atz, K., Grisoni, F., and Schneider, G. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3(12):1023–1032, 2021.

Bai, S., Zhang, F., and Torr, P. H. Hypergraph convolution and hypergraph attention. *Pattern Recognition*, 110: 107637, 2021.

Blum, L. C. and Reymond, J.-L. 970 million druglike small molecules for virtual screening in the chemical universe database gdb-13. *Journal of the American Chemical Society*, 131(25):8732–8733, 2009.

Bodnar, C. *Topological Deep Learning: Graphs, Complexes, Sheaves*. PhD thesis, Apollo - University of Cambridge Repository, 2022. URL https://www.repository.cam.ac.uk/handle/1810/350982.

Bodnar, C., Frasca, F., Wang, Y., Otter, N., Montufar, G. F., Lio, P., and Bronstein, M. Weisfeiler and lehman go topological: Message passing simplicial networks. In *International Conference on Machine Learning*, pp. 1026–1037. PMLR, 2021.

Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4): 18–42, 2017.

Cang, Z. and Wei, G.-W. Topologynet: Topology based deep convolutional and multi-task neural networks for biomolecular property predictions. *PLoS computational biology*, 13(7):e1005690, 2017.

Chan, H. S., Shan, H., Dahoun, T., Vogel, H., and Yuan, S. Advancing drug discovery via artificial intelligence. *Trends in pharmacological sciences*, 40(8):592–604, 2019.

Chen, C., Ye, W., Zuo, Y., Zheng, C., and Ong, S. P. Graph networks as a universal machine learning framework for molecules and crystals. *Chemistry of Materials*, 31(9): 3564–3572, 2019.

Chen, D., Liu, J., Wu, J., Wei, G.-W., Pan, F., and Yau, S.-T. Path topology in molecular and materials sciences. *The journal of physical chemistry letters*, 14(4):954–964, 2023.

Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., and Blaschke, T. The rise of deep learning in drug discovery. *Drug discovery today*, 23(6):1241–1250, 2018.

Chen, J., Qiu, Y., Wang, R., and Wei, G.-W. Persistent laplacian projected omicron ba. 4 and ba. 5 to become new dominating variants. *Computers in biology and medicine*, 151:106262, 2022.

Choudhary, K. and DeCost, B. Atomistic line graph neural network for improved materials property predictions. *npj Computational Materials*, 7(1):185, 2021.

Choudhary, K., DeCost, B., and Tavazza, F. Machine learning with force-field-inspired descriptors for materials: Fast screening and mapping energy landscape. *Physical review materials*, 2(8):083801, 2018.

Chowdhury, S. and Mémoli, F. Persistent path homology of directed networks. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1152–1169. SIAM, 2018.

Danel, T., Spurek, P., Tabor, J., Śmieja, M., Struski, Ł., Słowik, A., and Maziarka, Ł. Spatial graph convolutional networks. In *International Conference on Neural Information Processing*, pp. 668–675. Springer, 2020.

Fang, X., Liu, L., Lei, J., He, D., Zhang, S., Zhou, J., Wang, F., Wu, H., and Wang, H. Geometry-enhanced molecular representation learning for property prediction. *Nature Machine Intelligence*, 4(2):127–134, 2022.

Feng, Y., You, H., Zhang, Z., Ji, R., and Gao, Y. Hypergraph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 3558–3565, 2019.

Flam-Shepherd, D., Wu, T. C., Friederich, P., and Aspuru-Guzik, A. Neural message passing on high order paths. *Machine Learning: Science and Technology*, 2(4):045009, 2021.

Gasteiger, J., Groß, J., and Günnemann, S. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

Gasteiger, J., Becker, F., and Günnemann, S. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34: 6790–6802, 2021.

Giusti, L., Battiloro, C., Testa, L., Di Lorenzo, P., Sardellitti, S., and Barbarossa, S. Cell attention networks. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2023.

González, M. A. Force fields and molecular dynamics simulations. *École thématique de la Société Française de la Neutronique*, 12:169–200, 2011.

Grigor'yan, A., Lin, Y., Muranov, Y., and Yau, S.-T. Homologies of path complexes and digraphs. *arXiv preprint arXiv:1207.2834*, 2012.

Grigor'yan, A., Jimenez, R., Muranov, Y., and Yau, S.-T. Homology of path complexes and hypergraphs. *Topology and its Applications*, 267:106877, 2019.

Grigor'yan, A., Lin, Y., Muranov, Y., and Yau, S.-T. Homotopy theory for digraphs. *Pure and Applied Mathematics Quarterly*, 10(4):619–674, 2014.

Grigor'yan, A., Jimenez, R., Muranov, Y., and Yau, S.-T. On the path homology theory of digraphs and eilenberg–steenrod axioms. *Homology, Homotopy and Applications*, 20(2):179–205, 2018.

Grigor'yan, A., Lin, Y., Muranov, Y., et al. Path complexes and their homologies. *Journal of Mathematical Sciences*, 248:564–599, 2020. doi: 10.1007/s10958-020-04897-9. URL https://doi.org/10.1007/s10958-020-04897-9.

Hajij, M., Istvan, K., and Zamzmi, G. Cell complex neural networks. In *TDA* {\&} *Beyond*, 2020.

Hajij, M., Zamzmi, G., Papamarkou, T., Miolane, N., Guzmán-Sáenz, A., Natesan Ramamurthy, K., Birdal, T., Dey, T. K., Mukherjee, S., Samaga, S. N., et al. Topological deep learning: Going beyond graph data. *arXiv e-prints*, pp. arXiv–2206, 2022.

Halgren, T. A. Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of computational chemistry*, 17(5-6):490–519, 1996.

Hansen, J. and Ghrist, R. Toward a spectral theory of cellular sheaves. *Journal of Applied and Computational Topology*, 3(4):315–358, 2019.

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations*, 2019.

Huang, K., Fu, T., Glass, L., Zitnik, M., Xiao, C., and Sun, J. Deeppurpose: a deep learning library for drug–target interaction prediction. *Bioinformatics*, 36:5545 – 5547, 2020. URL https://api.semanticscholar.org/CorpusID:220496219.

Ingraham, J. B., Baranov, M., Costello, Z., Barber, K. W., Wang, W., Ismail, A., Frappier, V., Lord, D. M., Ng-Thow-Hing, C., Van Vlack, E. R., et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

Jamasb, A., Viñas Torné, R., Ma, E., Du, Y., Harris, C., Huang, K., Hall, D., Lió, P., and Blundell, T. Graphein-a python library for geometric deep learning and network analysis on biomolecular structures and interaction networks. *Advances in Neural Information Processing Systems*, 35:27153–27167, 2022.

Jiang, X., Tan, L., Cen, J., and Zou, Q. Molbench: A benchmark of ai models for molecular property prediction. In *International Symposium on Benchmarking, Measuring and Optimization*, pp. 53–70. Springer, 2023.

Kim, E.-S., Kang, W. Y., On, K.-W., Heo, Y.-J., and Zhang, B.-T. Hypergraph attention networks for multimodal learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14581–14590, 2020.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Leach, A. R. *Molecular modelling: principles and applications*. Pearson education, 2001.

Li, S., Zhou, J., Xu, T., Dou, D., and Xiong, H. GeomGCL: geometric graph contrastive learning for molecular property prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 4541–4549, 2022.

Li, Y., Wang, W., Liu, J., and Wu, C. Pre-training molecular representation model with spatial geometry for property prediction. *Computational Biology and Chemistry*, 109: 108023, 2024.

Liu, J., Chen, D., Pan, F., and Wu, J. Neighborhood path complex for the quantitative analysis of the structure and stability of carboranes. *Journal of Computational Biophysics and Chemistry*, 22(04):503–511, 2023.

Liu, S., Demirel, M. F., and Liang, Y. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *Advances in neural information processing systems*, 32, 2019.

Liu, Y., Wang, L., Liu, M., Lin, Y., Zhang, X., Oztekin, B., and Ji, S. Spherical message passing for 3d molecular graphs. In *International Conference on Learning Representations (ICLR)*, 2022.

Mak, K.-K. and Pichika, M. R. Artificial intelligence in drug development: present status and future prospects. *Drug discovery today*, 24(3):773–780, 2019.

Mayo, S. L., Olafson, B. D., and Goddard, W. A. Dreiding: a generic force field for molecular simulations. *Journal of Physical chemistry*, 94(26):8897–8909, 1990.

Qiao, Z., Christensen, A. S., Welborn, M., Manby, F. R., Anandkumar, A., and Miller III, T. F. Informing geometric deep learning with electronic interactions to accelerate quantum chemistry. *Proceedings of the National Academy of Sciences*, 119(31):e2205221119, 2022.

Ramakrishnan, R., Hartmann, M., Tapavicza, E., and Von Lilienfeld, O. A. Electronic spectra from tddft and machine learning in chemical space. *The Journal of chemical physics*, 143(8), 2015.

Ramsundar, B., Eastman, P., Walters, P., and Pande, V. *Deep learning for the life sciences: applying deep learning to genomics, microscopy, drug discovery, and more*. " O'Reilly Media, Inc.", 2019.

Reiser, P., Neubert, M., Eberhard, A., Torresi, L., Zhou, C., Shao, C., Metni, H., van Hoesel, C., Schopmans, H., Sommer, T., et al. Graph neural networks for materials science and chemistry. *Communications Materials*, 3(1): 93, 2022.

Roddenberry, T. M., Glaze, N., and Segarra, S. Principled simplicial neural networks for trajectory prediction. In *International Conference on Machine Learning*, pp. 9020–9029. PMLR, 2021.

Roddenberry, T. M., Schaub, M. T., and Hajij, M. Signal processing on cell complexes. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8852–8856. IEEE, 2022.

Rong, Y., Bian, Y., Xu, T., Xie, W., Wei, Y., Huang, W., and Huang, J. Self-supervised graph transformer on large-scale molecular data. *Advances in Neural Information Processing Systems*, 33:12559–12571, 2020.

Ruddigkeit, L., Van Deursen, R., Blum, L. C., and Reymond, J.-L. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling*, 52(11):2864–2875, 2012.

Schaub, M. T., Seby, J.-B., Frantzen, F., Roddenberry, T. M., Zhu, Y., and Segarra, S. Signal processing on simplicial complexes. In *Higher-Order Systems*, pp. 301–328. Springer, 2022.

Schütt, K., Kindermans, P.-J., Sauceda Felix, H. E., Chmiela, S., Tkatchenko, A., and Müller, K.-R. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A., and Müller, K.-R. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24), 2018.

Shen, C., Luo, J., and Xia, K. Molecular geometric deep learning. *Cell Reports Methods*, 3(11), 2023.

Shindo, H. and Matsumoto, Y. Gated graph recursive neural networks for molecular property prediction. *ArXiv*, abs/1909.00259, 2019. URL https://api.semanticscholar.org/CorpusID:202541698.

Shui, Z. and Karypis, G. Heterogeneous molecular graph neural networks for predicting molecule properties. *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 492–500, 2020a. URL https://api.semanticscholar.org/CorpusID:221971188.

Shui, Z. and Karypis, G. Heterogeneous molecular graph neural networks for predicting molecule properties. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 492–500. IEEE, 2020b.

Sverrisson, F., Feydy, J., Correia, B. E., and Bronstein, M. M. Fast end-to-end learning on protein surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15272–15281, 2021.

Townsend, J., Micucci, C. P., Hymel, J. H., Maroulas, V., and Vogiatzis, K. D. Representation of molecular structures with persistent homology for machine learning applications in chemistry. *Nature communications*, 11(1): 3230, 2020.

Townshend, R. J., Eismann, S., Watkins, A. M., Rangan, R., Karelina, M., Das, R., and Dror, R. O. Geometric deep learning of rna structure. *Science*, 373(6558):1047–1051, 2021.

Unke, O. T. and Meuwly, M. Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges. *Journal of chemical theory and computation*, 15 (6):3678–3693, 2019.

Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., et al. Graph attention networks. *stat*, 1050 (20):10–48550, 2017.

Vlassis, N. N., Ma, R., and Sun, W. Geometric deep learning for computational mechanics part i: Anisotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering*, 371:113299, 2020.

Wang, L., Liu, Y., Lin, Y., Liu, H., and Ji, S. Comenet: Towards complete and efficient message passing for 3d molecular graphs. *Advances in Neural Information Processing Systems*, 35:650–664, 2022a.

Wang, R. and Wei, G.-W. Persistent path laplacian. *Foundations of data science (Springfield, Mo.)*, 5(1):26–55, 2023.

Wang, Y., Wang, J., Cao, Z., and Barati Farimani, A. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence*, 4(3):279–287, 2022b.

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

Wu, Z., Ramsundar, B., Feinberg, E. N., Gomes, J., Geniesse, C., Pappu, A. S., Leswing, K., and Pande, V. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

Xia, K. and Wei, G.-W. Persistent homology analysis of protein structure, flexibility, and folding. *International journal for numerical methods in biomedical engineering*, 30(8):814–844, 2014.

Xiong, Z., Wang, D., Liu, X., Zhong, F., Wan, X., Li, X., Li, Z., Luo, X., Chen, K., Jiang, H., et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *Journal of medicinal chemistry*, 63(16):8749–8760, 2019.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388, 2019.

Yu, Z. and Gao, H. Molecular graph representation learning via heterogeneous motif graph construction. *arXiv preprint arXiv:2202.00529*, 2022.

Zhang, L., Tan, J., Han, D., and Zhu, H. From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug discovery today*, 22(11):1680–1685, 2017.

Zhu, J., Xia, Y., Wu, L., Xie, S., Zhou, W., Qin, T., Li, H., and Liu, T.-Y. Dual-view molecular pre-training. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3615–3627, 2023.

A. supplemental material

A.1. Initialization Features

Table 2. MD Encoder for Path Features

Features Type		Description	Type	Size
1-Path (bond)	Bond Directionality	None, Beginwedge, Begindash, etc.	One-Hot	7
	Bond Type	Single, Double, Triple, or Aromatic.	One-Hot	4
	Bond Length	Numerical length of the bond.	Float	1
	In Ring	Indicates if the bond is part of a chemical ring.	One-Hot	2
1-Path (non-bond) cutoff=3	Atom charges	Atoms charges in Molecular $(q_i, q_j, q_i \times q_j)$	Float	3
	Distance between atoms	Distance between atoms $(1/d_{ij}, 1/d_{ij}^6, 1/d_{ij}^{12})$	Float	3
2-Path	Centroid	Centroid position of the triangle formed by 2-path	Float	3
	Distance	Three bond lengths (two for covalent bond and one for non-covalent bond)	Float	3
	Area	Triangle area spanned by 2-path	Float	1
	Bond Angle	Bond angle for 2-path	Float	1
3-Path	Volume	Volume spanned by 3-path	Float	1
	Dihedral	Dihedral angle for 3-path	Float	1
	Total Area	Total Area of the corresponding four triangles	Float	1
	Bond Length	Non-covalent bond length ($\{v_1v_3\}$,$\{v_2v_4\}$, $\{v_1v_4\}$)	Float	3

A.2. Dataset details, Min-Max Scaling, Splitting Metho and Mean Absolute Err

In this study, we analyzed three key quantum chemistry datasets from MoleculeNet (Wu et al., 2018) and MolBench (Jiang et al., 2023): QM7 (Blum & Reymond, 2009), QM8 (Ramakrishnan et al., 2015), QM9 (Ruddigkeit et al., 2012), Tox21, HIV and MUV, all of which are publicly available on the MoleculeNet website: https://moleculenet.org/datasets-1. Details about these datasets are in Table 3.

Table 3. The details of the datasets. Note that the subindex indicates standard deviation values.

Dataset	QM7	QM9	Tox21	HIV	MUV
No. molecules	6,830	133,885	7831	41127	93808
No. average atoms	$16_{(3)}$	$18_{(3)}$	$36_{(23)}$	$46_{(24)}$	$43_{(10)}$
No. tasks	1	3	12	1	17
Task type	Regression	Regression	Classification	Classification	Classification

Note that the subindex indicates standard deviation values. For instance, the element $16_{(13)}$ means the number of average atoms in QM7 is 16, with 13 as its standard deviation. The QM7 dataset is a subset of the GDB-13 database (Blum & Reymond, 2009), which contains approximately 1 billion organic molecules with up to seven ”heavy” atoms (C, N, O, S). The QM7 dataset comprises 7,160 molecules along with their corresponding atomization energies. The QM9 dataset, a subset of the GDB-17 database, provides twelve properties, encompassing geometric, energetic, electronic, and thermodynamic properties. This dataset consists of 133,865 molecules. Tox21 is qualitative toxicity measurements on 12 biological targets, including nuclear receptors and stress response pathways. HIV is experimentally measured abilities to inhibit HIV replication. MUV is subset of PubChem BioAssay by applying a refined nearest neighbor analysis, designed for validation of virtual screening techniques.

Min-Max Scaling Given that QM7 and QM9 involve regression, we applied min-max normalization to scale target values between 0 and 1. In multiple-target regression tasks, Min-Max Scaling is commonly used to normalize the targets. This technique linearly transforms the target values to a specified range between a minimum and maximum value. The transformation follows the formula:

$$\overline{y} = \frac{y - y_{\text{min}}}{y_{\text{max}} - y_{\text{min}}}, \quad y_{\text{scal}} = y_{\text{max}} - y_{\text{min}} \tag{5}$$

Here, $\overline{y}$ represents the normalized target value, y is the original target value, y_{min} is the minimum value of the target, and y_{max} is the maximum value of the target.

During prediction, the normalized predictions obtained from the model need to be transformed back to the original scale of the target values. The transformation is performed using the formula:

$$\tilde{y} = \hat{y} \cdot y_{\text{scal}} + y_{\text{min}}, \quad y = \overline{y} \cdot y_{\text{scal}} + y_{\text{min}} \tag{6}$$

where $\hat{y}$ is the model output, and $\tilde{y}$ and y are used for loss function computation and evaluation.

This normalization process ensures that all target values are scaled within a fixed range, typically between 0 and 1. It facilitates better convergence during model training and helps in handling targets with varying scales effectively. Furthermore, Min-Max Scaling maintains the relative relationships between target values while bringing them into the desired range, making it a suitable choice for multiple-target regression tasks.

A.3. Benchmark Models and Hyperparameters

For an extensive validation of our PCNN model, we consider three widely-used benchmark datasets from MoleculeNet (Wu et al., 2018). In data preprocessing, we utilize Merck molecular force field (MMFF94) function from RDKit to generate molecular 3D structures. Following the work of Bharath Ramsundar (Ramsundar et al., 2019), we employed scaffold splitting to partition all datasets. This method segments molecules based on their scaffolds (molecular substructures). Scaffold splitting is a more challenging partitioning approach that can better evaluate a model's generalization ability on out-of-distribution data samples. To ensure a fair comparison with other models, we adopted the same scaffold splitting method to divide the task datasets into training, validation, and test sets with a ratio of 8:1:1.

We have compared our PCNN model with state-of-the-art GNN models with and without pre-training process. The compared GNN models without pre-training process include (1) the commonly used GNN architectures, GIN (Xu et al., 2018), GAT (Velickovic et al., 2017) and GCN (Kipf & Welling, 2016); (2) recent works incorporating three-dimensional molecular geometry, SGCN (Danel et al., 2020), DimeNet (Gasteiger et al., 2020) and HMGNN (Shui & Karypis, 2020b); (3) the architectures specially designed for molecular representation, D-MPNN (Yang et al., 2019), AttentiveFP (Xiong et al., 2019) Mol-GDL (Shen et al., 2023). Additionally, the compared GNN models with pre-training process include, N-Gram (Liu et al., 2019), PretrainGNN (Hu et al., 2019), GROVER (Rong et al., 2020), GEM (Fang et al., 2022), DMP (Zhu et al., 2023) and SMPT (Li et al., 2024).

MAE (Mean Absolute Error) The Mean Absolute Error (MAE) is defined as:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{7}$$

where y_i and $\tilde{y}_i$ represent the true value and predicted value of the i^{th} sample respectively. MAE is a commonly used metric for evaluating regression performance. A lower MAE value indicates higher prediction accuracy, with a decrease in MAE typically suggesting improved model performance.

Hyperparameters setup **Hyperparameters** We have set up a set of hyperparameters for training the model are summarized in Table 4. Inaddition, the optimizer selected as ADAM, and the loss function chosen as L1. All models are trained using NVIDIA RTX A6000 48GB GPUs.

Table 4. Hyperparameters set up.

Dataset	QM7	QM9	Tox21	HIV	MUV
Learning rate	1e-4	1e-3	1.5e-4	1e-3	1e-4
Batch size	512	64	36	512	512
No.heads	1	6	6	2	1
No.layers	2	2	2	2	2
Train/Valid/Test	8:1:1	8:1:1	8:1:1	8:1:1	8:1:1
Loss function	L1	L1	BCE	BCE	BCE
Optimizer	ADAM	ADAM	ADAM	ADAM	ADAM
Epochs	500	500	1000	1000	1000
Seed	42	42	42	42	42

B. Path Weisfeiler Lehman (PWL) Test

B.1. Path Complex

Definition B.1 (Path Complex Isomorphism)**.** Given two path complexes P_1, P_2 over the vertices V_1, V_2. P_1 and P_2 are called isomorphic if there is a map $f : V_1 \to V_2$ such that $\sigma_n = v_0 v_1 \cdots v_n \in P_1 \iff f(\sigma) = f(v_0)f(v_1)\cdots f(v_n) \in P_2$.

Theorem B.2. *Given two graphs G_1, G_2, let P_{G_1}, P_{G_2} be the path complexes derived from G_1, G_2 respectively. We have*

$$G_1 \cong G_2 \iff P_{G_1} \cong P_{G_2}$$

B.2. Path Complex Coloring

Definition B.3 (Path Coloring)**.** A path coloring is a map c such that for each path complex P and any path σ of P, $c(\sigma)$ is a color from a fixed color table. We denote this color by c_σ^P.

We will often omit P in the subscript when the underlying path complex is arbitrary.

Definition B.4. Given two path complexes P_1, P_2 and a path coloring c. P_1 and P_2 are called c-similar, denoted by $c^{P_1} = c^{P_2}$, if for any dimension n, we have the color multi-sets equality

$$\{\{c_\sigma^{P_1} | dim(\sigma) = n, \sigma \in P_1\}\} = \{\{c_\tau^{P_2} | dim(\tau) = n, \tau \in P_2\}\}$$

Definition B.5 (PWL)**.** We give a path complex version of the WL test to derive a message passing procedure that can retain the expressive power of the test. We call this the Path WL (PWL), the steps of general PWL are as follows:

1. Given a path complex P, all the paths of P are initialized with the same color.
2. For the color c_σ^t of path σ at iteration t, the color c_σ^{t+1} of σ at the next iteration is computed by perfectly hashing the color multi-set of the neighbors of σ.
3. The algorithm stops once a stable coloring is reached. Two path complexes are considered non-isomorphic if their color histograms are different at some dimensions.

Neighbor Color Multi-set Based on the four neighbor definitions, we have four types of neighbor color multi-sets. Let c^t be the coloring of PWL for path complex P at iteration t, four types of color multi-sets are as follows

1. $c_{\mathcal{B}}^t(\sigma) = \{\{c_\tau^t | \tau \in \mathcal{B}(\sigma)\}\}$
2. $c_{\mathcal{C}}^t(\sigma) = \{\{c_\tau^t | \tau \in \mathcal{C}(\sigma)\}$
3. $c_\uparrow^t(\sigma) = \{\{(c_\tau^t, c_{\sigma\cup\tau}^t) | \tau \in \mathcal{N}_\uparrow(\sigma)\}$
4. $c_\downarrow^t(\sigma) = \{\{(c_\tau^t, c_{\sigma\cap\tau}^t) | \tau \in \mathcal{N}_\downarrow(\sigma)\}$

Having the neighbor color multi-sets, we obtain the following update rule that contains all four types of neighbors:

$$c_\sigma^{t+1} = \mathrm{HASH}\{c_\sigma^t, c_\mathcal{B}^t(\sigma), c_\mathcal{C}^t(\sigma), c_\uparrow^t(\sigma), c_\downarrow^t(\sigma)\}$$

Actually, certain neighbors can be removed without affecting the expressive power of PWL test in terms of path complex that can be differentiated.

Theorem B.6. *PWL with* $\mathrm{HASH}\{c_\sigma^t, c_\mathcal{B}^t(\sigma), c_\uparrow^t(\sigma)\}$ *is as powerful as PWL with the four-neighbor-updating strategy* $\mathrm{HASH}\{c_\sigma^t, c_\mathcal{B}^t(\sigma), c_\mathcal{C}^t(\sigma), c_\uparrow^t(\sigma), c_\downarrow^t(\sigma)\}$.

Theorem B.7. *PWL is strictly more powerful than WL.*

Theorem B.8. *PWL is no less powerful than SWL (Bodnar et al., 2021) with the clique complex lifting.*

B.3. Path Complex Neural Network

We propose a general Path Complex Neural Network (PCNN) using the following messages passing operations. For a path σ in P, we have

$$m_\mathcal{B}^{t+1}(\sigma) = AGG_{\tau\in\mathcal{B}(\sigma)}(M_\mathcal{B}(h_\sigma^t, h_\tau^t)) \tag{8}$$

$$m_\uparrow^{t+1}(\sigma) = AGG_{\tau\in\mathcal{N}_\uparrow(\sigma)}(M_\uparrow(h_\sigma^t, h_\tau^t, h_{\sigma\cup\tau}^t)) \tag{9}$$

Then, the updating function considers these two types of messages and the previous color of σ:

$$h^{t+1}(\sigma) = U(h_\sigma^t, m_\mathcal{B}^t(\sigma), m_\uparrow^t(\sigma)) \tag{10}$$

After L layers of the message passing process, the readout function takes the color multi-sets at all dimensions as input:

$$h_P = \mathrm{READOUT}(\{\{h_\sigma^L\}\}_{dim(\sigma)=0}, \cdots, \{\{h_\tau^L\}\}_{dim(\tau)=p}) \tag{11}$$

Theorem B.9. *PCNN with sufficient layers and injective neighborhood aggregators are as powerful as PWL.*

B.4. Proof of Main Results

In order to prove the main results, we give some notations.

Definition B.10 (Path Coloring Refinement)**.** A path coloring c refines a path coloring d, denoted by $c \sqsubseteq d$, if for any path complex P_1, P_2 and $\sigma \in P_1$, $\tau \in P_2$, $c_\sigma^{P_1} = c_\tau^{P_2}$ implies $d_\sigma^{P_1} = d_\tau^{P_2}$. Additionally, if $d \sqsubseteq c$, we say that c and d are equivalent.

Lemma B.11. *Given two path complexes* P_1, P_2 *with* $A \subset P_1$, $B \subset P_2$*. Assume* c *and* d *are two path coloring such that* $c \sqsubseteq d$*. If* $\{\{d_\sigma^{P_1} | \sigma \in A\}\} \neq \{\{d_\tau^{P_2} | \tau \in B\}\}$*, then* $\{\{c_\sigma^{P_1} | \sigma \in A\}\} \neq \{\{c_\tau^{P_2} | \tau \in B\}\}$.

Proof. Let $C_1 = \{\{c_\sigma^{P_1} | \sigma \in A\}\}$, $C_2 = \{\{c_\tau^{P_2} | \tau \in B\}\}$. Assume $C_1 = C_2$, then there is a bijection $f : A \to B$ such that $\forall \sigma \in A$, $\tau = f(\sigma)$, we have $c_\sigma^{P_1} = c_\tau^{P_2}$. From $c \sqsubseteq d$ we know $d_\sigma^{P_1} = d_\tau^{P_2}$. Consequently, $\{\{d_\sigma^{P_1} | \sigma \in A\}\} = \{\{d_{f(\sigma)}^{P_2} | \sigma \in A\}\} = \{\{d_\tau^{P_2} | \tau \in B\}\}$, which contradicts with the condition that $\{\{d_\sigma^{P_1} | \sigma \in A\}\} \neq \{\{d_\tau^{P_2} | \tau \in B\}\}$. Hence the assumption is wrong. □

Corollary B.12. *Given two path colorings* c *and* d *such that* $c \sqsubseteq d$*. If* $d^{P_1} \neq d^{P_2}$*, then* $c^{P_1} \neq c^{P_2}$.

Proof. This follows by replacing the subsets A, B by the sets of n-paths of P_1 and P_2 respectively in the proof of Lemma B.11. □

The above corollary B.12 means that if c refines d, then c is able to distinguish all the path complex pairs that d can distinguish. In this sense, we can say that c is at least as powerful as d. If c and d are equivalent, we say they have the same expressive power.

Proof of Theorem B.2. It is easy to see that if $G_1 \cong G_2$, then $P_{G_1} \cong P_{G_2}$. The inverse statement follows from the fact that any graph is a subcomplex of its derived path complex by considering the 0-paths and 1-paths. □

Proof of Theorem B.6. Let a^t be the coloring at iteration t of the updating startegy

$$\mathrm{HASH}\{a_\sigma^t, a_\mathcal{B}^t(\sigma), a_\mathcal{C}^t(\sigma), a_\uparrow^t(\sigma), a_\downarrow^t(\sigma)\}$$

b^t be the coloring at iteration t of the updating strategy

$$\mathrm{HASH}\{b_\sigma^t, b_\mathcal{B}^t(\sigma), b_\uparrow^t(\sigma), b_\downarrow^t(\sigma)\}$$

c^t be the coloring at iteration t of the updating strategy

$$\mathrm{HASH}\{c_\sigma^t, c_\mathcal{B}^t(\sigma), c_\uparrow^t(\sigma)\}$$

We firstly prove that a^t and b^t are equivalent, then prove that b^t and c^t are equivalent.

1. a^t and b^t are equivalent. We have $a^t \sqsubseteq b^t$ because a^t contains additional colors of its coface neighbors in the color updating rule. It suffices to prove that $b^t \sqsubseteq a^t$. We do this by induction. The base case holds since all the paths are initialized with the same color. Assume the result holds for $t = k$, we prove that $b^{k+1} \sqsubseteq a^{k+1}$. Let $\sigma \in P_1$ and $\tau \in P_2$ be two n-paths from two arbitrary path complexes, suppose $b_\sigma^{k+1} = b_\tau^{k+1}$, we prove that $a_\sigma^{k+1} = a_\tau^{k+1}$.

   The equation $b_\sigma^{k+1} = b_\tau^{k+1}$ means that the hash function at iteration t+1 have the same arguments. Consequently, $b_\sigma^k = b_\tau^k$, $b_\mathcal{B}^k(\sigma) = b_\mathcal{B}^k(\tau)$, $b_\uparrow^k(\sigma) = b_\uparrow^k(\tau)$, $b_\downarrow^k(\sigma) = b_\downarrow^k(\tau)$. We prove that $b_\mathcal{C}^k(\sigma) = b_\mathcal{C}^k(\tau)$.

   We have $b_\uparrow^k(\sigma) = b_\uparrow^k(\tau)$ and

   $$b_\uparrow^k(\sigma) = \{\{(b_e^k, b_{\sigma\cup e}^k) | e \in \mathcal{N}_\uparrow(\sigma)\}\}, b_\uparrow^k(\tau) = \{\{(b_e^k, b_{\tau\cup e}^k) | e \in \mathcal{N}_\uparrow(\tau)\}\} \tag{12}$$

   Replacing the first component of the tuple by the same color, we have

   $$\{\{(-, b_{\sigma\cup e}^k) | e \in \mathcal{N}_\uparrow(\sigma)\}\} = \{\{(-, b_{\tau\cup e}^k) | e \in \mathcal{N}_\uparrow(\tau)\}\} \tag{13}$$

   By the definition of upper adjacency and coface we have

   $$b_\mathcal{C}^k(\sigma) = \{\{b_w^k | w \in \mathcal{C}(\sigma)\}\} = \{\{b_{\sigma\cup e}^k | e \in \mathcal{N}_\uparrow(\sigma)\}\} \tag{14}$$

   $$b_\mathcal{C}^k(\tau) = \{\{b_w^k | w \in \mathcal{C}(\tau)\}\} = \{\{b_{\tau\cup e}^k | e \in \mathcal{N}_\uparrow(\tau)\}\} \tag{15}$$

   Combining Equation (12), (13), (14), (15), we have $b_\mathcal{C}^k(\sigma) = b_\mathcal{C}^k(\tau)$.

   From the induction hypothesis, we have $a_\sigma^k = a_\tau^k$, $a_\mathcal{B}^k(\sigma) = a_\mathcal{B}^k(\tau)$, $a_\mathcal{C}^k(\sigma) = a_\mathcal{C}^k(\tau)$, $a_\uparrow^k(\sigma) = a_\uparrow^k(\tau)$, $a_\downarrow^k(\sigma) = a_\downarrow^k(\tau)$, so $a_\sigma^{k+1} = a_\tau^{k+1}$.

2. b^t and c^t are equivalent. Similarly we have $b^t \sqsubseteq c^t$, we further prove that $c^{2t} \sqsubseteq b^t$. We do this by induction. The base case is obvious because all the paths are initialized with the same color. Assume the results holds for $t = k$, we prove that $c^{2k+2} \sqsubseteq b^{k+1}$. Let $\sigma \in P_1$ and $\tau \in P_2$ be two n-paths from two arbitrary path complexes, suppose $c_\sigma^{2k+2} = c_\tau^{2k+2}$, we prove that $b_\sigma^{k+1} = b_\tau^{k+1}$.

   For $c_\sigma^{2k+2} = c_\tau^{2k+2}$, by going back two steps of the hash function, we have $c_\sigma^{2k} = c_\tau^{2k}, c_\mathcal{B}^{2k}(\sigma) = c_\mathcal{B}^{2k}(\tau), c_\uparrow^{2k}(\sigma) = c_\uparrow^{2k}(\tau)$. We want to prove that $c_\downarrow^{2k}(\sigma) = c_\downarrow^{2k}(\tau)$.

   Assume $c_\downarrow^{2k}(\sigma) \neq c_\downarrow^{2k}(\tau)$, then there is a color pair (c_0, c_1) such that (c_0, c_1) appears more times in $c_\downarrow^{2k}(\sigma)$ (without loss of generality) than in $c_\downarrow^{2k}(\tau)$. For any path δ and λ, define

   $$A(\delta) = \{\{(c_\phi^{2k} = c_0, c_\delta^{2k} = c_1) | \phi \in \mathcal{C}(\delta)\}\} \tag{16}$$

   $$C_\lambda = \{\{|A(\delta)| | \delta \in \mathcal{B}(\lambda)\}\} \tag{17}$$

   Then we have

   $$C_\sigma = \{\{|A(\delta)| | \delta \in \mathcal{B}(\sigma)\}\} = \{\{|(c_\phi^{2k} = c_0, c_\delta^{2k} = c_1)| | \delta \in \phi \cap \sigma\}\} \tag{18}$$

   $$C_\tau = \{\{|A(\delta)| | \delta \in \mathcal{B}(\tau)\}\} = \{\{|(c_\phi^{2k} = c_0, c_\delta^{2k} = c_1)| | \delta \in \phi \cap \tau\}\} \tag{19}$$

   So $C_\sigma \neq C_\tau$.

Considering the path coloring $d(\delta) = |A(\delta)|$. For two n-paths δ_1, δ_2, if $d(\delta_1) \neq d(\delta_2)$, we can assume that $|A(\delta_1)| > |A(\delta_2)|$ without loss of generality, then the number of upper adjacent neighbors of δ_1 and δ_2 up to color pair (c_0, c_1) are different, which means $c_{\uparrow}^{2k}(\delta_1) \neq c_{\uparrow}^{2k}(\delta_2)$. So $c_{\delta_1}^{2k+1} \neq c_{\delta_2}^{2k+1}$, which means $c^{2k+1} \sqsubseteq d$.

Applying Lemma B.11 to $\mathcal{B}(\sigma)$ and $\mathcal{B}(\tau)$, we have

$$\{\{c_{\delta_1}^{2k+1} | \delta_1 \in \mathcal{B}(\sigma)\}\} \neq \{\{c_{\delta_2}^{2k+1} | \delta_2 \in \mathcal{B}(\tau)\}\} \tag{20}$$

The above multi-sets are exactly the color multi-sets of the faces of σ and τ, which means $c_{\mathcal{B}}^{2k+1}(\sigma) \neq c_{\mathcal{B}}^{2k+1}(\tau)$. Consequently, $c_{\sigma}^{2k+2} \neq c_{\tau}^{2k+2}$, which contradicts with the induction hypothesis, so $c_{\downarrow}^{2k}(\sigma) = c_{\downarrow}^{2k}(\tau)$.

From the induction hypothesis, we have $b_{\sigma}^{k} = b_{\tau}^{k}$, $b_{\mathcal{B}}^{k}(\sigma) = b_{\mathcal{B}}^{k}(\tau)$, $b_{\uparrow}^{k}(\sigma) = b_{\uparrow}^{k}(\tau)$, $b_{\downarrow}^{k}(\sigma) = b_{\downarrow}^{k}(\tau)$, so $b_{\sigma}^{k+1} = b_{\tau}^{k+1}$.

□

Proof of Theorem B.7. Given a path complex P, let a^t be the coloring of the vertices of P at iteration t of WL and b^t be the coloring of the same vertices at iteration t of PWL. We firstly prove that $b^t \sqsubseteq a^t$, then give a pair of graphs to show that they cannot be differentiated by WL but can be differentiated by PWL.

1. $b^t \sqsubseteq a^t$. We do this by induction. The base case holds because all vertices are initialized with the same color. Suppose the result holds for $t = k$, we prove that $b^{k+1} \sqsubseteq a^{k+1}$. Let v and w be two vertices of two arbitrary path complexes P_1, P_2, suppose $b_v^{k+1} = b_w^{k+1}$, we prove that $a_v^{k+1} = a_w^{k+1}$.

   Note that vertices only has upper adjacent neighbors, so we have $b_v^k = b_w^k, b_{\uparrow}^k(v) = b_{\uparrow}^k(w)$. The second equation means

   $$\{\{b_x^k | (b_x^k, -) \in b_{\uparrow}^k(v)\}\} = \{\{b_y^k | (b_y^k, -) \in b_{\uparrow}^k(w)\}\}$$

   This can be equivalently written as

   $$\{\{b_x^k | x \in \mathcal{N}_{\uparrow}(v)\}\} = \{\{b_y^k | y \in \mathcal{N}_{\uparrow}(w)\}\}$$

   From the induction hypothesis, we have $a_v^k = a_w^k$ and

   $$\{\{a_x^k | x \in \mathcal{N}_{\uparrow}(v)\}\} = \{\{a_y^k | y \in \mathcal{N}_{\uparrow}(w)\}\}$$

   These are the arguments of the hash function for WL to compute the colors of v and w in the next iteration, so $a_v^{k+1} = a_w^{k+1}$.

2. Considering the graphs in Figure 4, they cannot be differentiated by WL test. In PWL test, the path complex derived from the right graph has not any 3-path while the derived path complex from the left graph has 3-paths.

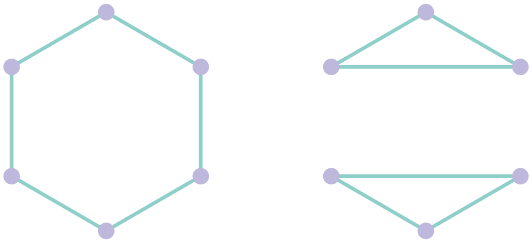

Figure 4. Two graphs that cannot be distinguished by WL but can be differentiated by PWL.

□

Proof of Theorem B.8. Considering the graphs in Figure 5, they cannot be differentiated by SWL test. In PWL test, the path complex derived from the right graph has not any 4-path while the derived path complex from the left graph has 4-paths.

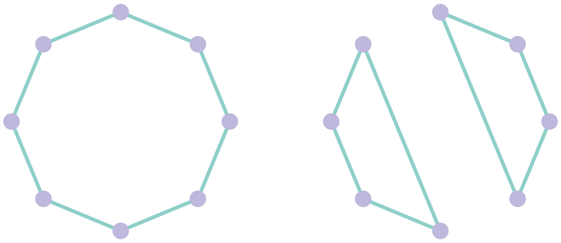

Figure 5. Two graphs that cannot be distinguished by SWL but can be differentiated by PWL.

□

Proof of Theorem B.9. Let b^t and d^t be the coloring at iteration t of PWL and the t-th layer of an PCNN respectively. Assume the PCNN has L layers and assume $d^t = d^L(t > L)$. We use induction to prove that $d^t \sqsubseteq b^t$. The base case holds by definition. Suppose the result holds for $t = k$, when $t = k + 1$, we prove that $d^{k+1} \sqsubseteq b^{k+1}$. For any two n-paths σ, τ of any two path complexes P_1, P_2 such that $d^{k+1}_\sigma = d^{k+1}_\tau$, we prove that $b^{k+1}_\sigma = b^{k+1}_\tau$.

The condition means all the update, aggregate and message functions are injective, so their composition is also injective. Hence $d^k_\sigma = d^k_\tau$, $d^k_{\mathcal{B}}(\sigma) = d^k_{\mathcal{B}}(\tau)$, $d^k_\uparrow(\sigma) = d^k_\uparrow(\tau)$.

$d^k_{\mathcal{B}}(\sigma) = d^k_{\mathcal{B}}(\tau)$ means

$$\{\{d^k_s | s \in \mathcal{B}(\sigma)\}\} = \{\{d^k_t | t \in \mathcal{B}(\tau)\}\}$$

$d^k_\uparrow(\sigma) = d^k_\uparrow(\tau)$ means

$$\{\{(d^k_s, d^k_{s\cup\sigma}) | s \in \mathcal{N}_\uparrow(\sigma)\}\} = \{\{(d^k_t, d^k_{t\cup\tau}) | t \in \mathcal{N}_\uparrow(\tau)\}\}$$

By the induction hypothesis, we have $b^k_\sigma = b^k_\tau$.

$$\{\{b^k_s | s \in \mathcal{B}(\sigma)\}\} = \{\{b^k_t | t \in \mathcal{B}(\tau)\}\}$$

$$\{\{(b^k_s, b^k_{s\cup\sigma}) | s \in \mathcal{N}_\uparrow(\sigma)\}\} = \{\{(b^k_t, b^k_{t\cup\tau}) | t \in \mathcal{N}_\uparrow(\tau)\}\}$$

So $b^k_\sigma = b^k_\tau$, $b^k_{\mathcal{B}}(\sigma) = b^k_{\mathcal{B}}(\tau)$, $b^k_\uparrow(\sigma) = b^k_\uparrow(\tau)$, these are the arguments of the hash function in PWL, so $b^{k+1}_\sigma = b^{k+1}_\tau$.

□