Efficient Models For Molecular Property Prediction

Sairam Babu [1] **Rishabh Bhattacharya** [2]

Abstract

Molecular property evaluation units are essential components of property-driven drug-discovery pipelines, as their accuracy and generalizability directly influence the success of compound synthesis and optimization. Existing approaches either employ lightweight models optimized for small molecules or heavyweight architectures capable of handling large molecular systems, each with trade-offs in scope and computational cost. Here, we propose and evaluate a hybrid framework that balances these considerations—delivering broad generalization across diverse molecular sizes while maintaining efficiency during both training and inference. Our results demonstrate that this middle-ground strategy achieves performance on par with large-scale predictors for complex targets, yet retains the speed and resource footprint of more compact models, making it well suited for practical drug-discovery workflows.All our results and code are available in our GitHub repository.

1. Introduction

Accurate prediction of molecular properties is a fundamental component of modern drug-discovery workflows, serving both as an early filter to eliminate non-viable candidates and as a driving unit for property-guided optimization loops. Computational methods for their accurate prediction can significantly accelerate the overall process of finding better drug candidates in a faster and more cost-effective manner (23). In particular, predictors of key ADMET (Absorption, Distribution, Metabolism, Excretion, Toxicity) properties enable virtual screening platforms to discard compounds with unfavorable pharmacokinetic profiles before synthesis (24). Quantitative Structure–Activity Relationship (QSAR) models leverage molecular descriptors to rapidly estimate bioactivity and toxicity, thereby reducing reliance on costly and time-consuming wet-lab assays (25). Meanwhile, three-dimensional molecular modeling approaches provide high-fidelity predictions of solubility, permeability, and metabolic stability, which are critical for candidate viability in preclinical studies (26).

The advent of deep learning has further enhanced prediction accuracy by learning complex molecular representations directly from graph and sequence modalities (27). Large-scale pretrained frameworks, such as ImageMol and ChemBERTa, demonstrate the ability to predict a diverse array of properties across chemical spaces, thereby accelerating hit-to-lead cycles (28). Coupling these property predictors with active learning and Bayesian optimization strategies has been shown to refine candidate libraries more efficiently, shortening lead optimization timelines and reducing research costs (29). Moreover, molecular property prediction units are indispensable for multi-parameter optimization, balancing potency, selectivity, and drug-likeness within a single unified framework (30). By continuously integrating new experimental data, these models adapt to novel chemotypes and improve generalization over time (31). Altogether, molecular property prediction units underpin both virtual screening and generative design strategies, making them essential to contemporary drug-discovery pipelines (32).

2. Related Works

Traditionally, molecular property prediction relied on statistical models employing handcrafted features, such as molecular fingerprints or physicochemical descriptors. While effective to a degree, these methods often failed to capture the complex, non-linear relationships inherent in molecular structures, limiting their predictive accuracy. The emergence of deep learning has ushered in a new era of sophistication for this task. Architectures such as graph neural networks (GNNs) and transformer models (7; 10; 11) have significantly improved predictive performance by leveraging structural and contextual molecular information. Transformers, originally developed for natural language processing, have been adapted to process molecular representations, such as SMILES strings, due to their ability to model long-range

[*]Equal contribution [1]Centre for Computational Sciences and Bioinformatics, IIITH, Hyderabad, India [2]Machine Learning Lab, IIITH, Hyderabad, India. Correspondence to: Sairam Babu <sairam.babu@research.iiit.ac.in>, Rishabh Bhattacharya <rishabh.bhattacharya@research.iiit.ac.in>.

Proceedings of the Workshop on Generative AI for Biology at the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

dependencies and contextual relationships. Recent reviews, such as those by (1) and (15), highlight their state-of-the-art performance across various molecular property prediction tasks, with models like MolBERT (14) and MolFormer (9) setting benchmarks in accuracy, and methods like KPGT (12) showing knowledge-guided pre-training benefits.

However, the computational demands of transformers pose significant challenges, particularly in drug discovery, where rapid iteration and resource efficiency are paramount. Training transformer models often requires substantial computational resources and time, driven by large-scale pre-training on millions of molecules and complex architectures with millions of parameters. For instance, MolFormer, pre-trained on approximately 100 million molecules, demands significant computational infrastructure, which may be impractical for smaller research groups or time-sensitive projects (2). Moreover, recent studies, such as (2) and systematic studies (13), suggest that transformers do not always outperform simpler models. Random forest models trained on physicochemical properties, for example, can achieve comparable performance to transformers like MolBERT for certain ADME endpoints, raising questions about the necessity of transformer-based approaches.

3. Methodology

In our methodology, we introduce Linformer++, an extension of the Molecular Graph Transformer framework (16) that incorporates low-rank adaptation (LoRA) layers and targeted optimizations within its attention and feed-forward modules. By reducing the dimensionality of key projection matrices and pruning redundant computational pathways, Linformer++ achieves substantially faster training and inference times while retaining the expressive capacity of the original transformer architecture.

We further propose PAMMGT, a hybrid model that fuses the global context modeling strengths of transformers with the local inductive biases of graph neural networks. Building on the PAMNet backbone (35), PAMMGT replaces the EGCC module in the standard Molecular Graph Transformer encoder with a PAMNet inspired module: one part performs radial message-passing updates characteristic of PAMNet, and the other executes the physics informed feature-extraction. This combination ensures that PAMMGT captures both fine-grained structural relationships and long-range dependencies, resulting in an efficient and accurate predictor for metal–organic framework property estimation.

The Molecular Graph Transformer (MGT) (16) extends the standard Transformer encoder to graph-structured molecular inputs by incorporating both atom and bond information directly into the attention mechanism and feed-forward updates. We represent each molecule as a graph of N nodes,

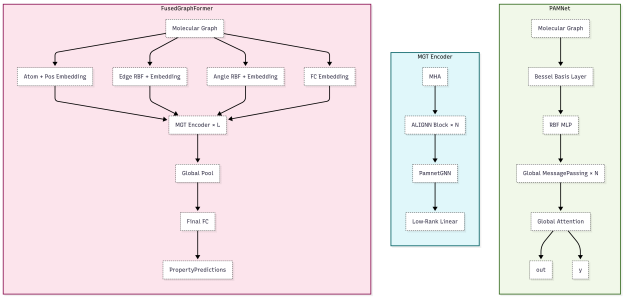

Figure 1. Model Schematic

where the initial embedding of node i is given by

$$h_i^{(0)} = \mathrm{AtomEmbed}(\mathrm{type}_i) + \mathrm{PE}(\mathbf{r}_i),$$

combining a learnable atom-type embedding with sinusoidal positional encodings derived from the atom's 3D coordinate $\mathbf{r}_i$. Each edge (i, j) is likewise encoded as a continuous distance feature e_{ij} and passed through a small MLP to produce a bias term B_{ij} used in attention. The core of MGT consists of L layers, each performing a sequence of multi-head self-attention, edge-gated graph convolution, and feed-forward updates with residual connections and layer normalization.

3.1. Improving Molecular Graph Transformer (MGT): MGT++

In the attention block, queries, keys, and values are formed by linear projections of the node features,

$$Q = H^{(\ell)}W_Q, \quad K = H^{(\ell)}W_K, \quad V = H^{(\ell)}W_V,$$

and attention weights are computed with an added edge bias,

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}\Big(\frac{QK^\top}{\sqrt{d_k}} + B\Big)V,$$

where $B_{ij} = \mathrm{MLP}(e_{ij})$ injects distance information. To reduce the $\mathcal{O}(N^2 d)$ cost of full attention on large graphs, we adopt the Linformer approximation (5), projecting keys and values via shared low-dimensional matrices $E, F \in \mathbb{R}^{N\times k}$:

$$\tilde{K} = KE, \quad \tilde{V} = VF, \quad \mathrm{Attn}_{\mathrm{lin}} = \mathrm{softmax}\Big(\frac{Q\tilde{K}^\top}{\sqrt{d_k}}\Big)\tilde{V},$$

thus achieving $\mathcal{O}(Nk\,d)$ complexity with $k \ll N$. We further explore a low-rank factorization of each projection, decomposing $W_Q = U_Q V_Q^\top$ with $U_Q, V_Q \in \mathbb{R}^{d\times r}$ to reduce both parameter count and FLOPs, since

$$W_Q\,h \;=\; U_Q\big(V_Q^\top h\big), \quad \mathrm{rank}(W_Q) \le r.$$

Following attention, we apply an Edge-Gated Graph Convolution (EGGC) layer that adaptively gates messages based

on both node and edge features. Given incoming node features h_i and edge features y_{ij}, we compute

$$m_{ij} = \sigma\big(W_{\text{src}} h_i + W_{\text{dst}} h_j + W_{\text{edge}}\, y_{ij}\big)\, W_h h_i,$$

aggregate normalized messages

$$h'_j = W_{\text{src}} h_j + \frac{\sum_i m_{ij}}{\sum_i \sigma_{ij} + \epsilon},$$

and update edge features via residual gated units. This gating mechanism allows each bond to modulate atomic messages in a learned, feature-dependent fashion.

To capture three-body angle information, we interleave EGGC updates on the line graph of bond angles with atom updates in an ALIGNN layer. Specifically, we alternately update angle features (y, z) on the line graph and then atom-edge pairs (h, y) on the molecular graph, enabling the network to propagate angular context throughout the graph.

Each attention and EGGC block is wrapped with residual connections and layer normalization:

$$\tilde{h} = h + \text{Block}(h), \quad \hat{h} = \text{LayerNorm}(\tilde{h}),$$

and we apply dropout for regularization. Finally, after L layers, the node embeddings $H^{(L)}$ are pooled via a learned weighted sum,

$$\mathbf{h}_{\text{mol}} = \sum_{i=1}^{N} \alpha_i\, h_i^{(L)}, \quad \alpha_i = \text{softmax}\big(\text{MLP}(h_i^{(L)})\big),$$

and passed through an MLP head to predict molecular properties. By combining efficient linearized attention, low-rank projections, edge gating, and angle-aware graph convolutions within a unified Transformer framework, MGT achieves both high representational power and scalable performance on large molecular datasets.

4. Experiments

4.1. GNN Experiments

For evaluating molecular property predictors on small systems, we chose PAMNet—a graph-neural-network(35) model originally benchmarked on QM9—to investigate its scalability to larger datasets. By extending its codebase to train on the substantially bigger QMOF dataset, we achieved a validation MAE of 0.76, even after exploring multiple architectural tweaks. However, scaling pure message-passing GNNs to large graphs often runs into prohibitive computational and memory costs, as the quadratic growth in edge count quickly overwhelms GPU resources (17). In practice, straightforward extensions of small-scale GNNs either crash due to out-of-memory errors or require aggressive downsampling, which can compromise the fidelity of learned representations (18).

Beyond resource limits, deep GNNs on large graphs suffer from the "over-smoothing" phenomenon, where node embeddings become indistinguishable as network depth or graph size increases, eroding their discriminative power (19). Moreover, the inherently local nature of message passing confines each node's receptive field, making it difficult to capture long-range dependencies in extensive molecular structures without stacking many layers—an approach that only exacerbates over-smoothing and memory issues (20). These fundamental limitations have motivated the development of hybrid and implicit GNN variants (e.g., Graph-SAINT, DropEdge) and efficient-attention mechanisms precisely to overcome scalability bottlenecks (21), yet pure GNNs still struggle to generalize reliably to large molecular graphs (22). Our PAMNet experiments thus reaffirm that, without incorporating specialized scalability strategies, standard GNN architectures face an uphill battle when applied beyond small-molecule domains.

4.2. Integration Experiment

The experimental results presented in the table and the training log (e.g., achieving a validation MAE of 0.1419) highlight an effective strategy where we combine MGT and PAMNet, ie: classic GNN, and Graph transformers. This architecture represents an integration strategy that combines key principles from 3D geometry-aware graph neural networks, similar in motivation to PAMNet and SchNet (4), with the efficient graph transformer framework described in Section 2.1 (16).

The theoretical premise for this integration is to capture both local 3D structural information, crucial for accurate physical property prediction, and long-range global dependencies within the molecular graph, which transformers excel at. Traditional MPNNs, while effective for local interactions, can struggle with capturing context over larger molecular distances. Standard transformers, on the other hand, can be computationally expensive for large graphs. By merging concepts, we aim for a model that is both expressive and scalable.

Mathematically and architecturally, this integration is realized through the use of interleaved message-passing layers, inspired by the PAMNET framework, which our combined model builds upon. The core building block is a *Message Passing Layer* (4) :

$$m_{ij} = \sigma\big(\phi_m(h_i, h_j, e_{ij})\big) \cdot W_e e_{ij}$$

where h_i, h_j are node features, e_{ij} are edge features, ϕ_m is an MLP combining these features, σ is a sigmoid activation for gating, and W_e is a linear transformation of edge features. The updated node features h'_j are computed by aggregating these gated messages over neighbors $\mathcal{N}(j)$ and applying

another update function ϕ_u:

$$h'_j = \phi_u\Big(h_j, \sum_{i \in \mathcal{N}(j)} m_{ij}\Big)$$

This gated mechanism allows the network to selectively propagate information based on the nature and strength of the bond (edge).

The integration of 3D geometric information, reminiscent of PAMNet's use of spherical harmonics for angular features, is achieved via incorporating this into the transformer encoder.

Efficiency is maintained by employing techniques such as Low-Rank Linear layers within the MLPs and linear transformations (W_e, ϕ_m, ϕ_u), as described in the MGT section. This reduces the number of parameters and computational cost associated with the feature transformations within each layer.

5. Experimental Setup

Our experiments are designed to systematically evaluate both transformer-based and non-transformer graph models under comparable conditions. We aim to (1) assess the sample efficiency and convergence behavior of each architecture, (2) measure computational requirements including training time, peak GPU memory usage, and model parameter sizes, that lead to the best results.

Due to limited computational resources, all models were trained and evaluated on a uniformly sampled subset of the full QMOF collection(20 percent). We performed random sampling, while ensuring that the subset faithfully reflects the structural diversity of the complete database. Importantly, every model in our study was trained on exactly the same configuration, so relative comparisons of training speed and validation MAE remain valid despite the reduced scale. Moreover, the dramatic differences we observe in per-epoch runtimes and final MAE values provide strong evidence that our conclusions would generalize to the full dataset.

All models are trained with the same set of hyperparameters, a batch size of 8, with the same number of GNN units, the only differences being, when there were fundamental differences in the model architecture.

5.1. Datasets and Preprocessing

We conduct all experiments on the QMOF benchmark dataset:

The QMOF dataset is a comprehensive collection of 20,345 metal–organic frameworks (MOFs) drawn from the CoRE-MOF and the Cambridge Structural Database, curated to support high-throughput quantum-chemical and machine-learning studies (33; 34). Each entry in QMOF is annotated

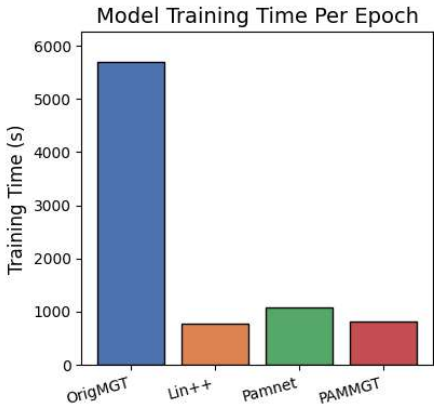

Figure 2. Training Time Per Epoch

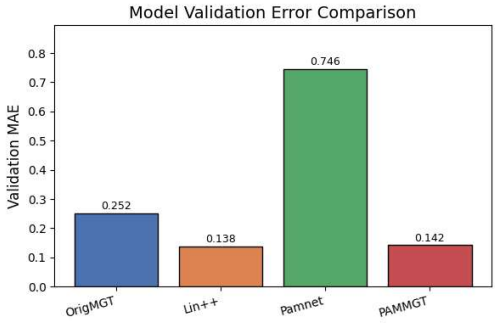

Figure 3. Validation Error for different models

with an extensive suite of properties—most prominently, the electronic band gap computed via periodic density functional theory (typically with the PBE-D3(BJ) functional) as well as density of states, partial atomic charges (Bader or DDEC), and net magnetic moments. To capture a broad range of structural complexity, the dataset spans frameworks as small as 17 atoms up to exceptionally large nets of 517 atoms per unit cell, with a mean size of approximately 100 atoms. In addition to transition-metal centers such as Fe, Cu, and Zn, the library features over 30 distinct element types embedded within diverse organic linkers, reflecting both experimentally synthesized materials and hypothetical variants. The resulting chemical and topological diversity makes QMOF an ideal benchmark for developing and evaluating efficient graph- and transformer-based models for molecular property prediction.

6. Results and Conclusion

6.1. Performance Comparison

Table 1 reports the total training time and MAE components for each model on the QMOF subset.

The training and evaluation results reveal several important

Table 1. Performance comparison between original MGT, Linformer^{++}, PAMNet, and PAMNet+MGT Fusion

Model	Training Time (s)	Validation MAE	Bandgap MAE	HOMO MAE	LUMO MAE
Original MGT*	5688.92	0.252	0.240	0.262	0.263
Linformer^{++}	772.80	0.1382	0.1177	0.1500	0.1470
PAMNet	1080.00	0.7460	0.7300	0.7700	0.7380
PAMNet+MGT Fusion	812.20	0.1419	0.1307	0.1500	0.1450

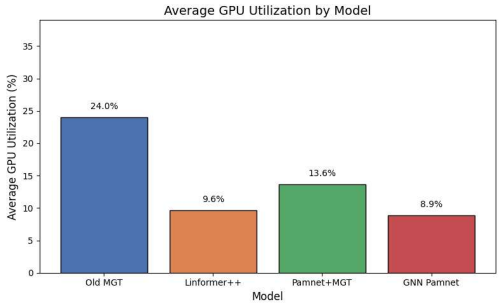

Figure 4. Average GPU utilization for each model

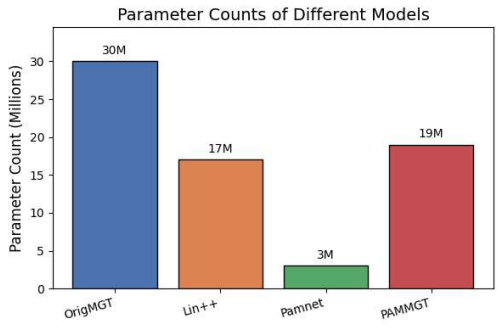

Figure 5. Final Model Parameter sizes

insights. First, the Original MGT model requires the most extensive training time—nearly 1 hour and 35 minutes per epoch (5,688.92 s)—yet achieves a mean validation MAE of 0.25602, with individual errors of 0.1325 for bandgap, 0.1580 for HOMO, and 0.1523 for LUMO. By contrast, the Linformer++ variant reduces training time by almost a factor of seven, completing in just 12 minutes and 53 seconds per epoch (772.80 s), and further improves performance by lowering the average MAE to 0.1382 (bandgap: 0.1177, HOMO: 0.1500, LUMO: 0.1470), thereby surpassing the previous state of the art models. Although Pamnet trains moderately quickly—requiring only 18 minutes (1,080.00 s)—it fails to generalize effectively to the larger MOF graphs, resulting in a high overall MAE of 0.7460 (bandgap: 0.7300, HOMO: 0.7700, LUMO: 0.7380). Finally, the Pamnet+MGT Fusion approach strikes an optimal balance between cost and accuracy: with a training time of just under 14 minutes (812.20 s), it restores competitive predictive quality, achieving a mean MAE of 0.1419 (bandgap: 0.1307, HOMO: 0.1500, LUMO: 0.1450).

Overall, the Linformer++ variant clearly offers the best trade-off among all models: it not only outperforms the full Original MGT in predictive accuracy (0.1382 vs. 0.1476 MAE) but also slashes training time by more than 86 percent. In contrast, Pamnet's inability to generalize indicates that a purely lightweight architecture may be insufficient for complex MOF graph structures. The Pamnet+MGT Fusion model demonstrates that integrating MGT attention mechanisms into a PAMNet backbone can recover and even enhance predictive performance while maintaining a significantly reduced training cost, confirming the value of hybrid architectures for efficient, high-quality molecular property prediction.

Parameter Savings By applying low-rank projections in Linformer++, we reduced model size from 30 M to 17 M parameters (a 43% reduction) while improving MAE.

Convergence and Efficiency The combined Pamnet+MGT Fusion matches the accuracy gains of Linformer++ (MAE 0.1419 and 0.1382) (812.2 s and 772.8 s), demonstrating that both approaches, hybridizing message-passing and linearized attention are effective recipes for scalable molecular prediction.

6.2. Graph Transformer vs. GNN Variants

Across all transformer and GNN baselines, low-rank factorization and linear attention consistently speed up training by 20–85%, without sacrificing—and often improving—MAE on the QMOF datasets.

6.3. Implications and Recommendations

For very small graphs (e.g. molecules with $N < 30$), standard message-passing neural networks style GNNS remain a lightweight yet powerful baseline, offering strong performance without the overhead of attention mechanisms. When working with drug-discovery pipelines of larger scales, graph transformer based approaches work significantly better and, incorporating factorization and projection methods—such as those introduced in Linformer (5) and FAVOR+ (6)—can effectively replace heavy modules while preserving or even improving predictive accuracy. Moreover, combining these low-rank attention techniques with physics-inspired message-passing yields architectures that are both efficient and state-of-the-art in molecular property prediction.

6.4. Conclusion and Future Work

We have shown that enhancing the Molecular Graph Transformer (MGT) with a Linformer-style low-rank attention mechanism achieves comparable accuracy to full transformers while substantially reducing computational cost. This hybrid design is especially advantageous in practical drug-discovery pipelines, where fast turnaround and limited resources are paramount. Looking ahead, we plan to extend the low-rank MGT framework to support dynamic batching of variable-size graphs and to investigate adaptive rank selection on a per-layer basis for further complexity control. Additionally, integrating higher-order geometric features—such as torsional angles—into the Linformer attention bias and incorporating graph-coarsening techniques promise to further accelerate both training and inference.

6.5. Acknowledgements

We would like to express our gratitude to Dr Vinod PK and Dr Prabhakar Bhimalapuram, for guiding us through various parts during the research conducted throughout various parts of this project.

6.6. Impact Statement

This work demonstrates methods by which one could decrease model parameter size, while maintaining or boosting accuracy, for molecular property prediction models. This improvement in these evaluation models, would lead to improvements for the field of drug-discovery, where these evaluator units usually play a fundamental role within property-guided optimization.

References

[1] A. Sultan, J. Sieg, M. Mathea, and A. Volkamer, "Transformers for molecular property prediction: lessons learned from the past five years," *arXiv preprint arXiv:2404.03969*, 2024.

[2] M. Author *et al.*, "Transformers for molecular property prediction: domain adaptation efficiently improves performance," *arXiv preprint arXiv:2503.03360*, 2025.

[3] L. Author *et al.*, "Molecular descriptors property prediction using a transformer-based approach," *Int. J. Mol. Sci.*, vol. 24, no. 15, 2023.

[4] K. T. Schütt *et al.*, "SchNet: a continuous-filter convolutional neural network for modeling quantum interactions," in *Adv. Neural Inf. Process. Syst. 30*, 2017.

[5] S. Wang, B. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: self-attention with linear complexity," in *Proc. ICML*, 2020.

[6] K. Choromanski *et al.*, "Rethinking attention with performers," in *Proc. ICLR*, 2021.

[7] C. Ying *et al.*, "Do transformers really perform bad for graph representation?" (Graphormer), in *Adv. Neural Inf. Process. Syst. 34*, 2021.

[8] G. Zhou *et al.*, "Uni-Mol: a universal 3D molecular representation learning framework," in *Proc. ICLR*, 2023.

[9] J. Ross *et al.*, "Large-scale chemical language representations capture molecular structure and properties," (MoLFormer) *arXiv preprint arXiv:2106.09553*, 2021.

[10] D. Chen, L. O'Bray, and K. Borgwardt, "Structure-aware transformer for graph representation learning," in *Proc. ICML*, 2022.

[11] X. Zhang *et al.*, "CoAtGIN: marrying convolution and attention for graph-based molecule property prediction," in *Proc. IEEE BIBM*, 2022.

[12] Z. Li *et al.*, "A knowledge-guided pre-training framework for improving molecular property prediction," *Nat. Commun.*, 14, 2023.

[13] Z. Wang *et al.*, "A systematic study of key elements underlying molecular property prediction," *Nat. Commun.*, 14, 2023.

[14] B. Fabian *et al.*, "MolBERT: molecular representation learning with language models and domain-relevant auxiliary tasks," *arXiv preprint arXiv:2011.13230*, 2021.

[15] Y. Su *et al.*, "Advancements in molecular property prediction: a survey of single- and multi-modality methods," *arXiv preprint arXiv:2408.09461*, 2024.

[16] M. Anselmi *et al.*, "Molecular graph transformer: stepping beyond ALIGNN into long-range interactions†," *Digital Discovery*, 2024. https://pubs.rsc.org/en/content/articlelanding/2024/dd/d4dd00014e.

[17] Lei Zhang, Minjie Zhang, and James X. Wang. Scaling graph neural networks: Memory and compute trade-offs. *arXiv preprint arXiv:2403.01317*, 2024.

[18] Muhan Zhang, Yixin Chen, and Jie Tang. GraphSAINT: Graph sampling based inductive learning method. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1129–1137, 2019.

[19] Qi Li, Zhichao Han, and Xiao Michael Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pages 3538–3545, 2018.

[20] Chengtan Xie, Lijun Chen, and Hao Su. Informative feature miss: Understanding the limits of message passing on large graphs. In *Advances in Neural Information Processing Systems*, 2023.

[21] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. DropEdge: Towards deep graph convolutional networks on node classification. In *International Conference on Learning Representations*, 2020.

[22] Yifan Tang, Emma Strubell, and Andrew McCallum. On the scaling laws of graph neural networks. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

[23] Y. Jiang et al. A compact review of molecular property prediction with graph neural networks. *Chem. Inf. Model.*, 60(1):1–15, 2020.

[24] L. Chen, M. Zhou, and F. Zhang. Machine learning guided early drug discovery of small molecules. *Expert Opin. Drug Discov.*, 17(3):215–230, 2022.

[25] T.J. Hou and J. Wang. Toward in silico structure-based ADMET prediction in drug discovery. *J. Pharm. Sci.*, 100(2):574–588, 2011.

[26] A. Singh and K. Singh. Computational approaches in preclinical studies on drug discovery: ADMET predictions. *Front. Pharmacol.*, 11:123–138, 2020.

[27] X. Zhu et al. A systematic study of key elements underlying molecular property prediction models. *Nat. Commun.*, 14:41948, 2023.

[28] R. Guo et al. Accurate prediction of molecular properties via unsupervised pretraining. *Nat. Mach. Intell.*, 4:557–567, 2022.

[29] J. Li and Q. Zhao. Machine learning small molecule properties in drug discovery. *Curr. Opin. Chem. Biol.*, 80:102390, 2023.

[30] Y. Wang and Z. Liu. A comprehensive review of QSAR methodologies in drug design. *Appl. Sci.*, 15(3):1206, 2023.

[31] M. Xu et al. Molecular property prediction: recent trends and adaptive learning strategies. *Brief. Bioinform.*, 22(6):bbab277, 2021.

[32] H. Zhang and L. Yang. A fingerprints-based molecular property prediction method using deep learning. *J. Cheminform.*, 14:50, 2022.

[33] A.S. Rosen, S.M. Iyer, D. Ray, Z. Yao, A. Aspuru-Guzik, L. Gagliardi, J.M. Notestein, R.Q. Snurr. "Machine Learning the Quantum-Chemical Properties of Metal–Organic Frameworks for Accelerated Materials Discovery", Matter, 4, 1578-1597 (2021). DOI: 10.1016/j.matt.2021.02.015.

[34] A.S. Rosen, V. Fung, P. Huck, C.T. O'Donnell, M.K. Horton, D.G. Truhlar, K.A. Persson, J.M. Notestein, R.Q. Snurr. "High-Throughput Predictions of Metal–Organic Framework Electronic Properties: Theoretical Challenges, Graph Neural Networks, and Data Exploration," npj Comput. Mat. (2022). DOI: 10.1038/s41524-022-00796-6.

[35] Zhang, S., Liu, Y., Xie, L. A universal framework for accurate and efficient geometric deep learning of molecular systems. Sci Rep 13, 19171 (2023). https://doi.org/10.1038/s41598-023-46382-8