Leveraging Weighted Sums for Integrating Message-Passing and Global Attention in GPS Graph Transformer

TAL BEN TOV - 208634766, OMER TALMI - 318900883, and LIA SOFFER - 318848983

The General, Powerful, and Scalable (GPS) framework aims to integrate local message-passing with global attention for graph learning. However, its fixed-layer design may not optimally adapt to different graph structures. In this work, we propose **Weighted GPS (WGPS)**, an extension of the GPS framework, which employs a dynamic gating mechanism to adaptively balance local message-passing and global attention layers. Each node in the graph receives customized scaling coefficients computed via a dynamic gating network, enabling more flexible and context-aware representation learning. Our model leverages node-level features to compute weighted sums of local and global components, allowing for greater adaptability across datasets. Experimental results on several graph benchmarks demonstrate that while WGPS can tailor the contribution of each layer to dataset-specific needs, achieving optimal performance remains challenging due to the inherent stochasticity in training. Nonetheless, our analysis reveals valuable insights into graph structure, highlighting how different datasets benefit from varying emphases on local or global information. This adaptive architecture offers a promising direction for future research in modular and scalable graph neural networks.

1 INTRODUCTION

Graph Neural Networks (GNNs) have become a key paradigm for modeling relational data across numerous applications, including social network analysis, chemical graph prediction, and recommendation systems [Gilmer et al. 2017]. Among the most widely studied GNN variants are **Message Passing Neural Networks (MPNNs)**, which aggregate information from node neighbors through iterative message-passing steps. MPNNs have been shown to excel at capturing local graph structures, making them well-suited for tasks such as molecular property prediction and community detection in social networks [Gilmer et al. 2017]. However, MPNNs struggle to capture long-range dependencies due to over-squashing [Alon and Yahav 2021] and often suffer from high locality bias due to over-smoothing [Oono and Suzuki 2021], limiting their performance on tasks requiring global information flow [Dwivedi et al. 2022a].

On the other hand, **Graph Transformers (GTs)** were introduced to address these limitations by incorporating global attention mechanisms. Transformers in the graph domain allow nodes to attend to all other nodes in the graph, making them highly effective at alleviating over-smoothing and over-squashing [Alon and Yahav 2021; Topping et al. 2022]. The attention mechanism enables nodes to focus on more distant regions of the graph, alleviating issues of locality and improving the expressiveness of the model [Morris et al. 2021; Xu et al. 2019]. However, a major drawback of GTs lies in their quadratic computational complexity $O(N^2)$ for a graph with N nodes and E edges [Vaswani et al. 2023], which can be restrictive for large-scale graphs commonly encountered in real-world tasks like traffic systems or genomic analysis [Dwivedi et al. 2022a]. Another disadvantage is their sensitivity to hyperparameters and lack of robustness across different datasets. The absence of a one-size-fits-all solution makes tuning critical and challenging, especially for diverse real-world graphs [Rampášek et al. 2023].

To strike a balance between local processing in GNNs and global attention in GTs, the **General, Powerful, and Scalable (GPS) framework** was proposed [Rampášek et al. 2023]. GPS combines the best of both worlds by applying a message-passing layer to capture local information and a global attention layer to capture long-range dependencies in each layer of the model. This alternating structure allows GPS to efficiently process graphs while maintaining the benefits of both MPNNs and GTs.

However, while GPS successfully integrates local and global mechanisms, the fixed alternation between message-passing and global attention layers may not always be optimal for every graph structure, as some graphs may require more attention on local interactions, while others might benefit from global connectivity patterns [Bronstein et al. 2017].

[1] Code is publicly available at https://github.com/omertalmi5/WeightedGraphGPS.

Authors' address: Tal Ben Tov - 208634766; Omer Talmi - 318900883; Lia Soffer - 318848983.

In this work, we propose a novel approach to dynamically scale between the local message-passing and global attention layers within the GPS architecture - **Weighted GPS (WGPS)**. Instead of a fixed alternation between layers, we introduce a scaling mechanism that computes a gating scalar vector to scale the contribution of each layer based on node-level features. This adaptive weighting aimed to:

- **Improve the performance of the GPS framework** by tailoring the balance between local and global information to the specific characteristics of the graph.
- **Understand the conditions under which global attention or message-passing is more effective** by analyzing the output scalar vector across diverse types of graphs.

By investigating the scaling weights learned for different graphs, we aim to uncover properties of graphs, that influence the relative importance of local versus global mechanisms. This approach offers insights into how MPNN and Transformer components can be optimally combined in a single model to achieve better performance across diverse graph datasets.

2 RELATED WORK

Recent studies have further explored hybrid models that integrate GNN-based message-passing and Transformer-based global attention. For instance, in molecular property prediction tasks, hybrid models like **Graphormer** [Shi et al. 2023; Ying et al. 2021] have shown that combining MPNN and Transformer components leads to superior performance by capturing both local molecular interactions and global structural motifs. Graphormer improves Transformer performance on molecular graphs by introducing a spatial encoding that preserves the inherent structure of graphs while leveraging attention [Gilmer et al. 2017; Vaswani et al. 2023]. Similarly, **SAN (Graph Structure-Aware Transformer)** [Kreuzer et al. 2021] enhances standard Transformers by incorporating structural information into the attention mechanism, enabling the model to account for graph connectivity while processing global attention. These models demonstrate that combining local and global mechanisms can significantly improve expressiveness and performance in diverse domains, particularly for tasks where long-range dependencies are critical, such as molecular dynamics or large-scale social network analysis.

In the GPS framework [Rampášek et al. 2023], the MPNN and global attention layers are combined through a summing-up approach, where the output of the message-passing layer is added to the output of the global attention layer at each iteration. This additive approach offers several benefits. First, it allows the model to capture both local and global information simultaneously, leveraging the strengths of MPNNs in encoding neighborhood structures and the ability of attention mechanisms to capture long-range dependencies. By summing these outputs, GPS ensures that both types of information contribute equally to the final node embeddings without one mechanism overpowering the other. This balance helps mitigate issues like over-smoothing, which can occur if MPNN layers dominate [Oono and Suzuki 2021], or computational complexity, which arises from relying solely on attention layers [Vaswani et al. 2023]. Moreover, the additive fusion avoids excessive model complexity, keeping the architecture simple yet powerful, which is particularly useful in large-scale graphs where computational efficiency is critical.

However, challenges remain in such hybrid systems, including the GPS framework. The use of both MPNN and Transformer layers often introduces **model complexity**, requiring careful tuning of hyperparameters to balance local and global components [Rampášek et al. 2023]. Moreover, certain tasks may disproportionately rely on local or global information, making it difficult to generalize the relative importance of each component across different datasets.

3 METHOD

In this paper, we propose a novel architecture where each layer consists of two parallel components: a message-passing layer and a global attention layer. Instead of summing them sequentially as in GPS, we compute a weighted sum of their outputs using

learned weights throughout the layers. This is done by incorporating a dynamic gating mechanism based on node-level features. This allows the model to decide at each layer the relative importance of message passing versus global attention per node.

Dynamic Gating Network Design. The core innovation of this work lies in the introduction of a dynamic gating network within the GPS layer, as shown in Figure 1, transforming it into the WGPS layer. The gating network consists of two GCNConv layers [Kipf and Welling 2016], where the first layer reduces the dimensionality of node features, and the second layer outputs two gating scalars for each node. These scalars are passed through a softmax function.
Thus, the network learns the optimal scaling coefficients for each node based on the graph structure, allowing the GCNConv to balance the contributions of local message passing and global attention dynamically.
The softmax has multiple roles. First, it ensures that the combined weights for the local and global representations sum to 1, thereby providing a normalized weighting scheme. More importantly, the softmax mechanism allows the network to dynamically choose between the two modalities (local and global) by assigning a higher weight to the more relevant representation in each context. Unlike a simple averaging or summation of the representations, the softmax emphasizes the better one for the given input by selectively amplifying the scalar corresponding to the dominant modality. This dynamic selection ensures that the WGPS layer is context-aware and capable of prioritizing either message passing (local) or global attention (global), depending on the specific requirements of the node's environment.
In mathematical terms, the scalars are computed as:

$$\mathbf{a}_{\mathrm{M}}^{\ell+1}, \mathbf{a}_{\mathrm{T}}^{\ell+1} = \mathrm{softmax}\left(\mathrm{GCNConv}_2\left(\mathrm{ReLU}\left(\mathrm{GCNConv}_1(\mathbf{X}^{\ell})\right)\right)\right) \tag{1}$$

Here, $\mathbf{a}_{\mathrm{M}}^{\ell+1}$ and $\mathbf{a}_{\mathrm{T}}^{\ell+1}$ are the gating scalar vectors corresponding to the message passing and global attention mechanisms, respectively. X is initially (in layer 1) the node features, and then is updated by the rule:

$$\mathbf{X}^{\ell+1} = MLP\left(\mathbf{a}_{\mathrm{M}}^{\ell+1} \cdot \mathbf{X}_{\mathrm{M}}^{\ell+1} + \mathbf{a}_{\mathrm{T}}^{\ell+1} \cdot \mathbf{X}_{\mathrm{T}}^{\ell+1}\right) \tag{2}$$

Here, $\mathbf{X}_{\mathrm{M}}^{\ell+1}$ represents the node features obtained from local GNN-based message-passing, and $\mathbf{X}_{\mathrm{T}}^{\ell+1}$ represents the node features obtained from the global attention mechanism. The scalar outputs dynamically adjust the importance of the local and global representations at the node level.
The softmax ensures these scalars are non-negative and sum to 1, promoting competition between them to emphasize the stronger signal. This formulation allows the WGPS layer to leverage the most appropriate mechanism—whether it is graph-based message passing or transformer-style attention—depending on the specific input characteristics and graph topology.
The method's advantage lies in its flexibility—by dynamically adjusting the contributions of local and global information for each node, the model can adapt to a wide variety of graph structures and tasks. The experimental setup follows the GPS framework, leveraging both small and large-scale graph benchmarks to demonstrate the scalability and effectiveness of the proposed method.

4 EXPERIMENTS AND RESULTS

4.1 Datasets

We utilize datasets from both the Benchmarking GNNs [Dwivedi et al. 2022a] and the Open Graph Benchmark (OGB) [Hu et al. 2021]. From Benchmarking GNNs, we employ the ZINC dataset, and from OGB we employ three graph-level datasets: ogbg-molhiv, ogbg-molpcba and ogbg-code2.

ZINC Dataset [Dwivedi et al. 2022a]. ZINC (MIT License) is a molecular dataset consisting of 12,000 molecular graphs sourced from the ZINC database, which is widely used in chemical compound research. These molecular graphs range in size from 9 to 37 nodes,

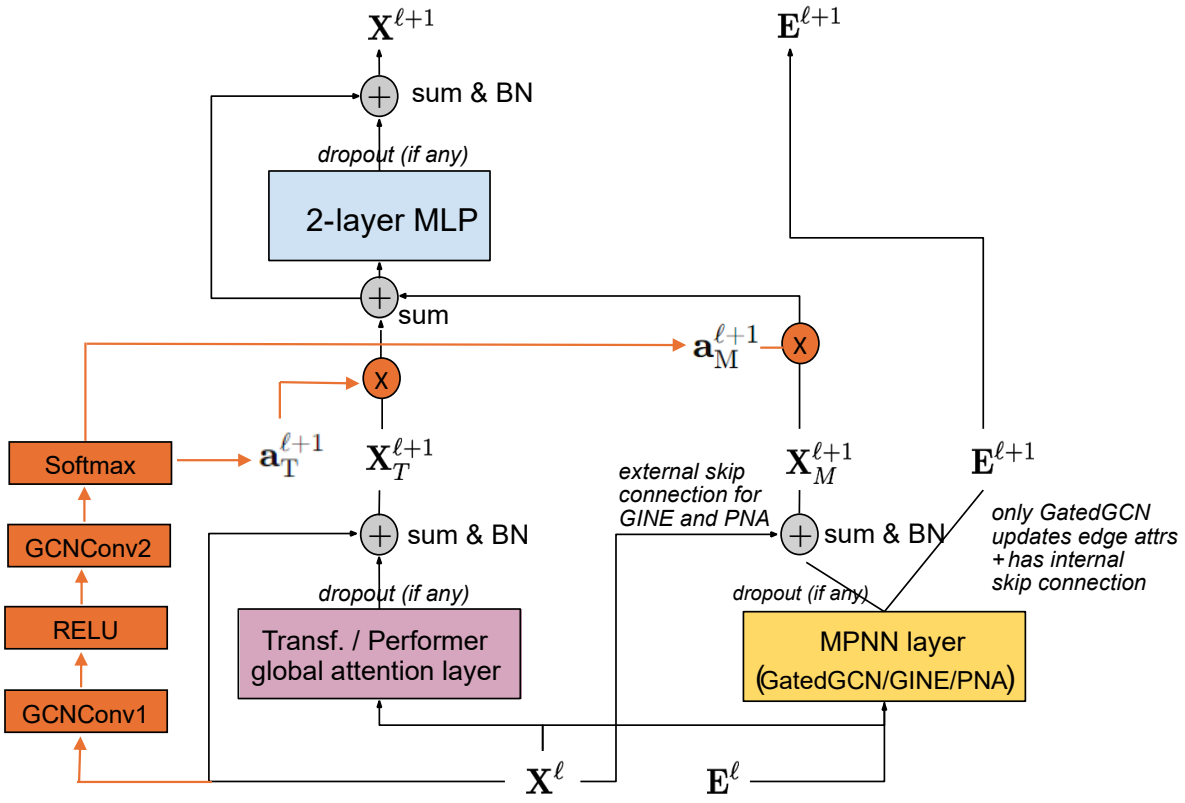


Fig. 1. The WGPS layer, with modifications from the original GPS layer shown in orange: the dynamic gating network outputs two scalars, $a_T^{\ell+1}$ and $a_M^{\ell+1}$. These scalars are used to compute a weighted sum of $X_T^{\ell+1}$ and $X_M^{\ell+1}$, where $X_T^{\ell+1}$ represents the node features after passing through the global attention layer, and $X_M^{\ell+1}$ represents the node features following the MPNN layer. The resulting weighted sum is given by: $a_M^{\ell+1} \cdot X_M^{\ell+1} + a_T^{\ell+1} \cdot X_T^{\ell+1}$.

Table 1. Summary of the graph learning datasets utilized in this paper.

Dataset	# Graphs	Avg. # nodes	Avg. # edges	Directed	Prediction level	Prediction task
ZINC	12,000	23.2	24.9	No	graph	regression
ogbg-molhiv	41,127	25.5	27.5	No	graph	binary classification
ogbg-molpcba	437,929	26.0	28.1	No	graph	128-task classification
ogbg-code2	452,741	125.2	124.2	Yes	graph	5-token sequence classification

with each node representing a heavy atom (from 28 possible types) and each edge denoting a bond (3 bond types). The task for this dataset is to predict a molecule's constrained solubility, defined as logP SA cycle (where logP is the octanol-water partition coefficient, penalized by the synthetic accessibility score, SA, and the number of long cycles). The dataset includes predefined splits for training (10K), validation (1K), and testing (1K), with node features representing atom types and edge features capturing bond types. The average runtime on this dataset is approximately *9.77* hours.

ogbg-molhiv and ogbg-molpcba Datasets [Hu et al. 2021]. The ogbg-molhiv and ogbg-molpcba datasets (MIT License), adopted from MoleculeNet, are used for molecular property prediction. These datasets feature a common node (atom) and edge (bond) representation that captures chemophysical properties. The ogbg-molhiv dataset's task is binary classification, predicting whether a molecule inhibits HIV replication, with an average runtime of *2.75* hours. In contrast, ogbg-molpcba is a multi-task binary classification dataset derived from PubChem BioAssay, where the objective is to predict the outcomes of 128 bioassays, with an average runtime of *10.24* hours.

ogbg-code2 Dataset [Hu et al. 2021]. The ogbg-code2 dataset (MIT License) is composed of abstract syntax trees (ASTs) derived from Python function code. The task is to predict the first five subtokens of a function's name. Due to the presence of exceptionally large ASTs, ASTs with over 1,000 nodes were truncated to the first 1,000 nodes based on their depth in the tree. This truncation affected only 0.5% (2,521 graphs) of the dataset. The average runtime on this dataset is approximately *26.04* hours.

Dataset Splits and Random Seeds: For all evaluated benchmarks, we adhere to the predefined train, validation, and test splits provided by the datasets. We report the mean performance along with the standard deviation across 10 independent runs, each with a different random seed. This methodology ensures a robust assessment of model performance and, importantly, aligns with the approach used in the original GPS framework, allowing for a fair comparison of results. Additionally, on the ZINC dataset, we used fewer training epochs compared to other benchmarks to reduce computational cost while maintaining reliable performance evaluations.

Github: We forked the official GPS repository and modified the `GPSLayer` class to implement our proposed changes. The complete code for our experiments, along with instructions to reproduce the results, is publicly available at the following GitHub repository: GitHub Repository. The specific modifications to the `GPSLayer` class can be found in this modified file.

4.2 Results

Our suggested model does not succeed in outperforming the existing models, as seen in Table 2. While some individual runs produce outstanding results, the mean performance does not surpass that of the baseline models.
One possible explanation can be the non-deterministic behavior of PyTorch that introduces additional variance across runs. As PyTorch is not entirely deterministic, even under seemingly identical conditions, slight deviations can occur between experiments. This makes it challenging to achieve consistent top-performing results across all test splits.

Scaling Values. The model successfully learned the preferences for each dataset, recognizing that in some cases, local message-passing is more dominant, whereas in others, global attention is more significant. Moreover, one notable observation concerns the **scaling vector output** of the gating network, specifically the vector a_{mag}, which is the set of learned coefficients for each node in the dataset, referred to as a_M in the formula. The values within each scaling vector tend to be quite similar across all items. For instance, typical scaling vectors across runs exhibit values like $(0.10, 0.11, 0.12, 0.10, \dots)$.
To provide further statistical insight, we measured the **mean and standard deviation** of each scaling vector across the different datasets on the test splits. The standard deviation values were consistently small, indicating minimal variability across vector components. For the different datasets the mean of the standard deviation are 0.2541 for ogb-molhiv, 0.2602 for ogb-molpcba, 0.0857 for ZINC and 0.2206 for ogb-code2. This suggests that, while the model dynamically determines scaling values for each node individually, it exhibits a tendency to apply similar scaling behavior across all nodes in practice. This preference for a consistent scaling pattern across the dataset implies that the model implicitly favors a unified scaling strategy, despite its capability to make node-specific decisions.

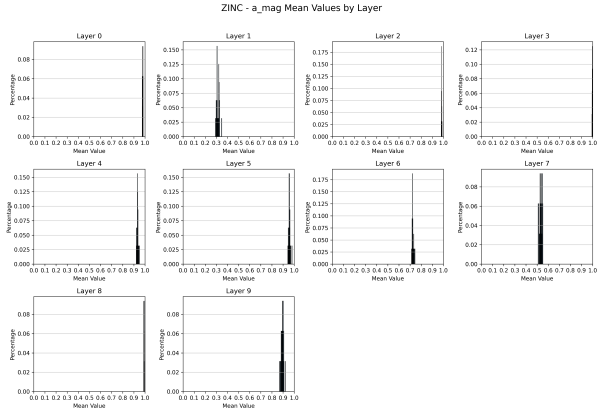
Table 2. Test performance on the selected benchmarks. Shown is the mean ± s.d. of 10 runs with different random seeds. Highlighted are the top **first**, **second**, and **third** results.

Model	ZINC	ogbg-molhiv	ogbg-molpcba	ogbg-code2
	MAE ↓	AUROC ↑	Avg. Precision ↑	F1 score ↑
GCN	0.367 ± 0.011	–	–	–
GCN+virtual node	–	0.7599 ± 0.0119	0.2424 ± 0.0034	0.1595 ± 0.0018
GIN [Xu et al. 2019]	0.526 ± 0.051	–	–	–
GIN+virtual node	–	0.7707 ± 0.0149	0.2703 ± 0.0023	0.1581 ± 0.0026
GAT [Veličković et al. 2018]	0.384 ± 0.007	–	–	–
GatedGCN [Bresson and Laurent 2018; Dwivedi et al. 2022a]	0.282 ± 0.015	–	–	–
GatedGCN-LSPE [Dwivedi et al. 2022b]	0.090 ± 0.001	–	0.267 ± 0.002	–
PNA [Corso et al. 2020]	0.188 ± 0.004	0.7905 ± 0.0132	0.2838 ± 0.0035	0.1570 ± 0.0032
DGN [Beaini et al. 2021]	0.168 ± 0.003	**0.7970 ± 0.0097**	–	–
GSN [Bouritsas et al. 2023]	0.101 ± 0.010	**0.8039 ± 0.0090**	–	–
CIN [Bodnar et al. 2022]	**0.079 ± 0.006**	**0.8094 ± 0.0057**	–	–
CRaWl [Tönshoff et al. 2023]	0.085 ± 0.004	–	**0.2986 ± 0.0025**	–
GIN-AK+ [Zhao et al. 2022]	**0.080 ± 0.001**	0.7961 ± 0.0119	**0.2930 ± 0.0044**	–
SAN [Kreuzer et al. 2021]	0.139 ± 0.006	0.7785 ± 0.2470	0.2765 ± 0.0042	–
Graphormer [Ying et al. 2021]	0.122 ± 0.006	–	–	–
K-Subgraph SAT [Chen et al. 2022]	0.094 ± 0.008	–	–	**0.1937 ± 0.0028**
EGT [Hussain et al. 2022]	0.108 ± 0.009	–	–	–
DeeperGCN [Li et al. 2020]	–	0.7858 ± 0.0117	0.2781 ± 0.0038	–
ExpC [Yang et al. 2022]	–	0.7799 ± 0.0082	0.2342 ± 0.0029	–
GraphTrans (GCN-Virtual)	–	–	–	**0.1830 ± 0.0024**
GPS	**0.070 ± 0.004**	0.7880 ± 0.0101	**0.2907 ± 0.0028**	**0.1894 ± 0.0024**
WGPS (ours)	0.124 ± 0.0101	0.7823 ± 0.0063	0.2839 ± 0.0037	0.1783 ± 0.0006

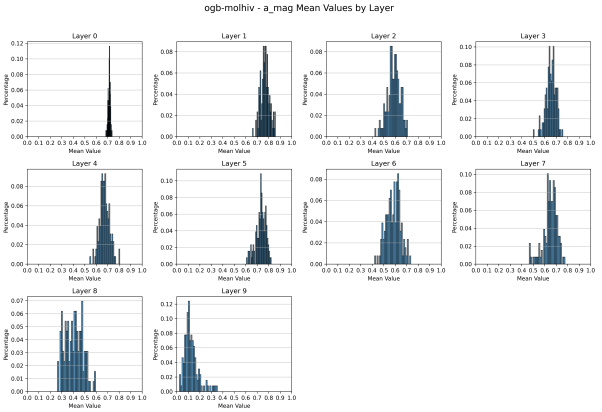
Analysis of Layer Contributions. In our analysis, we observed distinct behaviors across the layers of different datasets, as shown in Figures 2a, 2b, 2c, and 2d. These figures illustrate the mean scaling values across all layers for the ZINC, ogbg-molhiv, ogbg-molpcba, and ogbg-code2 datasets, respectively. In the molecular datasets—ZINC, ogbg-molhiv, and ogbg-molpcba—most layers predominantly perform message-passing operations, with the exception of the second layer in ZINC and the last layer in ogbg-molhiv, where global attention plays a more significant role. However, in ogbg-code2, the distribution centers around 0.4, suggesting that message-passing might not be as dominant. We assume the pattern aligns with the varying structural characteristics and learning requirements of the datasets.

In ZINC (Figure 2a), the distribution suggests a preference for message-passing in most layers, with a sharp peak near 1.0. However, the second layer shows greater engagement with global attention, likely capturing long-range dependencies important for molecular interactions. In contrast, the ogbg-molhiv dataset (Figure 2b) displays a more balanced approach across its layers, peaking between 0.6 and 0.7, while relying on global attention in the final layer to aggregate information from across the entire graph, ensuring a comprehensive representation for the classification decision. The ogbg-molpcba dataset (Figure 2c) shows a strong reliance on message-passing, with most mean values concentrated around 0.7 to 0.8, highlighting the importance of localized information in the multi-task classification setting. Finally, the ogbg-code2 dataset (Figure 2d) exhibits a distinct distribution, with most values centered around 0.4, indicating a lesser reliance on message-passing compared to the other datasets.

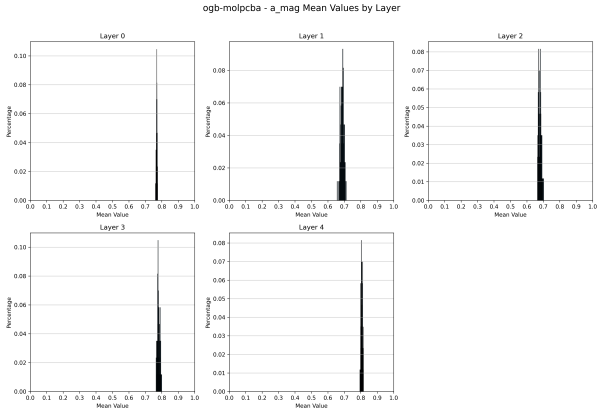
The molecular datasets ZINC, ogbg-molhiv, and ogbg-molpcba appear to favor message-passing mechanisms due to their relatively small node counts—an average of 25.5 nodes for ogbg-molhiv—which align well with the local interactions typically needed in chemical property prediction. Both these molecular graphs and the abstract syntax trees (ASTs) in the ogbg-code2 dataset share characteristics such as being tree-like with small average node degrees, small average clustering coefficients, and large average graph diameters [Hu et al. 2021]. However, they differ significantly in structure and scale. Molecular graphs have smaller node



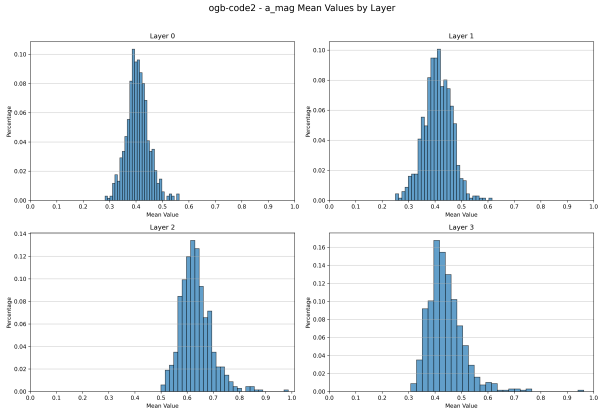
(a) Histogram of mean scaling values across all layers in the ZINC dataset.



(b) Histogram of mean scaling values across all layers in the ogbg-molhiv dataset.



(c) Histogram of mean scaling values across all layers in the ogbg-molpcba dataset.



(d) Histogram of mean scaling values across all layers in the ogbg-code2 dataset.

Fig. 2. Histograms of mean scaling values across all layers for different datasets.

counts, whereas ASTs are exactly trees with a hierarchical and acyclic structure, featuring a much larger average number of nodes (approximately 125.2) and well-defined root nodes. This reflects the nested and non-cyclic nature of code, potentially reducing the effectiveness of message-passing methods that rely on local interactions. Instead, the hierarchical nature and larger size of ASTs suggest that code graphs may benefit more from global attention mechanisms to effectively capture long-range dependencies. The observed mean value around 0.4 for the ogbg-code2 dataset indicates that message-passing might be less effective compared to models that leverage global context.

These differences in where global attention is emphasized between ZINC and ogbg-molhiv can be attributed to their specific dataset properties and learning objectives. In ZINC, the regression task of predicting molecular properties like constrained solubility benefits from early global attention in the second layer to capture long-range dependencies essential for accurate predictions. This early integration of global information allows the model to consider subtle interactions within the molecular structure, such as the

influence of atom types and bond types, which affect properties like solubility, synthetic accessibility, and the presence of long cycles [Dwivedi et al. 2022a]. Conversely, ogbg-molhiv involves a binary classification task aiming to determine whether a molecule inhibits HIV replication, which requires integrating localized information from throughout the molecule [Hu et al. 2021]. Increased global attention in the final layer helps aggregate this information, ensuring the classification decision is based on a comprehensive representation of the molecule. The influence of the task—regression in ZINC and classification in ogbg-molhiv—thus plays a significant role in where global attention is emphasized within the network.

5 FUTURE WORK

To enhance the flexibility and adaptability of our model, future work can focus on two main directions:

(1) **Modular Gating Network:** Transform the gating mechanism into a modular design, allowing users to select the most suitable model (e.g., GCN, MLP, or attention-based layers) based on the specific characteristics of their dataset. This will enable broader applicability across diverse datasets, beyond those tested in this work.
(2) **Framework for Architecture Selection:** Use the proposed model as a framework to guide architecture choices. After training, analyzing the scaling values can inform the replacement of the gating network with precomputed scalars optimized for specific datasets.

These improvements aim to make the model more generalizable and efficient for various tasks and datasets.

6 CONCLUSION

In this project, we aimed to enhance the performance and flexibility of the General, Powerful, and Scalable (GPS) framework by introducing a dynamic gating mechanism that adaptively balances the contributions of local message-passing and global attention within each layer. Our proposed Weighted GPS (WGPS) model computes weighted sums of these components using learned scalar vectors, allowing the architecture to tailor its focus to the structural properties of the input graph dynamically. Although the WGPS model demonstrated the ability to capture different graph characteristics and adapt its behavior across datasets, the results revealed challenges in consistently outperforming baseline models. Despite individual runs showing promising results, the variability introduced by non-deterministic behavior in PyTorch limited its overall performance. Nonetheless, the insights gained from the scaling behavior indicate that the model effectively distinguishes when to prioritize local versus global mechanisms, making it a valuable step toward more adaptive graph learning architectures. Further refinement and exploration of modular gating and architecture selection could unlock the full potential of this approach in future work.

REFERENCES

Uri Alon and Eran Yahav. 2021. On the Bottleneck of Graph Neural Networks and its Practical Implications. arXiv:2006.05205 [cs.LG] https://arxiv.org/abs/2006.05205

Dominique Beaini, Saro Passaro, Vincent Létourneau, William L. Hamilton, Gabriele Corso, and Pietro Liò. 2021. Directional Graph Networks. arXiv:2010.02863 [cs.LG] https://arxiv.org/abs/2010.02863

Cristian Bodnar, Fabrizio Frasca, Nina Otter, Yu Guang Wang, Pietro Liò, Guido Montúfar, and Michael Bronstein. 2022. Weisfeiler and Lehman Go Cellular: CW Networks. arXiv:2106.12575 [cs.LG] https://arxiv.org/abs/2106.12575

Giorgos Bouritsas, Fabrizio Frasca, Stefanos Zafeiriou, and Michael M. Bronstein. 2023. Improving Graph Neural Network Expressivity via Subgraph Isomorphism Counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 1 (Jan. 2023), 657–668. https://doi.org/10.1109/tpami.2022.3154319

Xavier Bresson and Thomas Laurent. 2018. Residual Gated Graph ConvNets. arXiv:1711.07553 [cs.LG] https://arxiv.org/abs/1711.07553

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. 2017. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine* 34, 4 (July 2017), 18–42. https://doi.org/10.1109/msp.2017.2693418

Dexiong Chen, Leslie O'Bray, and Karsten Borgwardt. 2022. Structure-Aware Transformer for Graph Representation Learning. arXiv:2202.03036 [stat.ML] https://arxiv.org/abs/2202.03036

Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. 2020. Principal Neighbourhood Aggregation for Graph Nets. arXiv:2004.05718 [cs.LG] https://arxiv.org/abs/2004.05718

Vijay Prakash Dwivedi, Chaitanya K. Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. 2022a. Benchmarking Graph Neural Networks. arXiv:2003.00982 [cs.LG] https://arxiv.org/abs/2003.00982

Vijay Prakash Dwivedi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. 2022b. Graph Neural Networks with Learnable Structural and Positional Representations. arXiv:2110.07875 [cs.LG] https://arxiv.org/abs/2110.07875

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. 2017. Neural Message Passing for Quantum Chemistry. arXiv:1704.01212 [cs.LG] https://arxiv.org/abs/1704.01212

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2021. Open Graph Benchmark: Datasets for Machine Learning on Graphs. arXiv:2005.00687 [cs.LG] https://arxiv.org/abs/2005.00687

Md Shamim Hussain, Mohammed J. Zaki, and Dharmashankar Subramanian. 2022. Global Self-Attention as a Replacement for Graph Convolution. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22, Vol. 14)*. ACM, 655–665. https://doi.org/10.1145/3534678.3539296

Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907* (2016).

Devin Kreuzer, Dominique Beaini, William L. Hamilton, Vincent Létourneau, and Prudencio Tossou. 2021. Rethinking Graph Transformers with Spectral Attention. arXiv:2106.03893 [cs.LG] https://arxiv.org/abs/2106.03893

Guohao Li, Chenxin Xiong, Ali Thabet, and Bernard Ghanem. 2020. DeeperGCN: All You Need to Train Deeper GCNs. arXiv:2006.07739 [cs.LG] https://arxiv.org/abs/2006.07739

Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. 2021. Weisfeiler and Leman Go Neural: Higher-order Graph Neural Networks. arXiv:1810.02244 [cs.LG] https://arxiv.org/abs/1810.02244

Kenta Oono and Taiji Suzuki. 2021. Graph Neural Networks Exponentially Lose Expressive Power for Node Classification. arXiv:1905.10947 [cs.LG] https://arxiv.org/abs/1905.10947

Ladislav Rampášek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. 2023. Recipe for a General, Powerful, Scalable Graph Transformer. arXiv:2205.12454 [cs.LG] https://arxiv.org/abs/2205.12454

Yu Shi, Shuxin Zheng, Guolin Ke, Yifei Shen, Jiacheng You, Jiyan He, Shengjie Luo, Chang Liu, Di He, and Tie-Yan Liu. 2023. Benchmarking Graphormer on Large-Scale Molecular Modeling Datasets. arXiv:2203.04810 [cs.LG] https://arxiv.org/abs/2203.04810

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. 2022. Understanding over-squashing and bottlenecks on graphs via curvature. arXiv:2111.14522 [stat.ML] https://arxiv.org/abs/2111.14522

Jan Tönshoff, Martin Ritzert, Hinrikus Wolf, and Martin Grohe. 2023. Walking Out of the Weisfeiler Leman Hierarchy: Graph Learning Beyond Message Passing. arXiv:2102.08786 [cs.LG] https://arxiv.org/abs/2102.08786

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. arXiv:1706.03762 [cs.CL] https://arxiv.org/abs/1706.03762

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. arXiv:1710.10903 [stat.ML] https://arxiv.org/abs/1710.10903

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How Powerful are Graph Neural Networks? arXiv:1810.00826 [cs.LG] https://arxiv.org/abs/1810.00826

Mingqi Yang, Renjian Wang, Yanming Shen, Heng Qi, and Baocai Yin. 2022. Breaking the Expression Bottleneck of Graph Neural Networks. https://ieeexplore.ieee.org/document/9759979

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. 2021. Do Transformers Really Perform Bad for Graph Representation? arXiv:2106.05234 [cs.LG] https://arxiv.org/abs/2106.05234

Lingxiao Zhao, Wei Jin, Leman Akoglu, and Neil Shah. 2022. From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness. arXiv:2110.03753 [cs.LG] https://arxiv.org/abs/2110.03753