Understanding Virtual Nodes in Graph Neural Networks: Oversmoothing, Oversquashing and Node Heterogeneity

Johannes Lutzeyer

Data Science and Mining Team, Laboratoire d'Informatique (LIX),
École Polytechnique, Institut Polytechnique de Paris

March 31, 2025

Overview of Today's Talk

1) Brief Introduction to Graph Neural Networks (GNNs), virtual nodes in GNNs, and Graph Transformers;

2) Understanding Virtual Nodes;

3) Reaction to Recent Position Paper: GNNs for Digital Pathology.

Graph Neural Networks

Graph Neural Networks (GNNs) are neural networks that take graph-structured data as input.

We consider graph-structured data to be the combination of

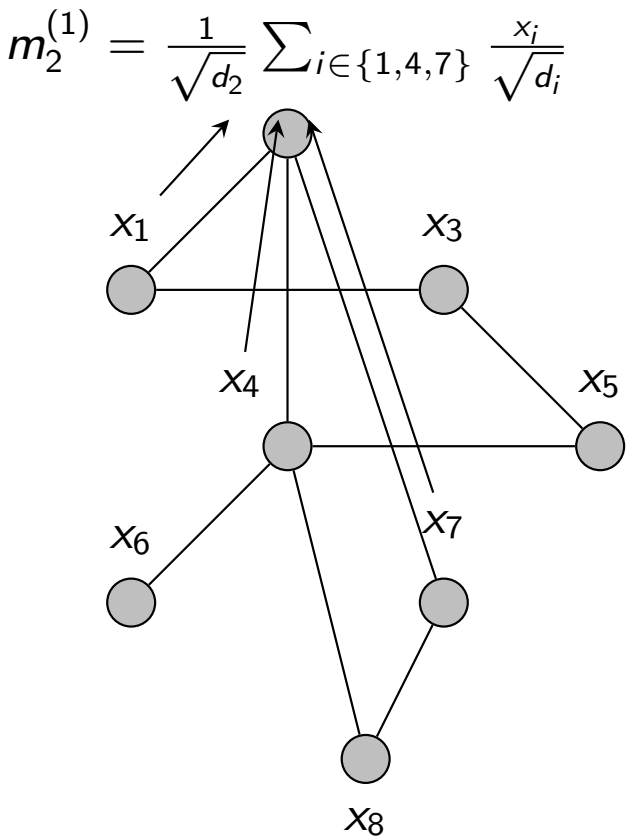
- a graph $G = (V, E)$;
- node-features $X = [x_1, \ldots, x_n]^T$.

In this talk we will only see a specific type of GNN, the Message Passing Neural Networks.

$$m_v^{(k)} = M^{(k)}\left(\left\{h_w^{(k-1)} : w \in \mathcal{N}(v)\right\}\right),$$
$$h_v^{(k)} = U^{(k)}\left(h_v^{(k-1)}, m_v^{(k)}\right).$$

E.g., the Graph Convolutional Network (GCN, Kipf and Welling, 2017)

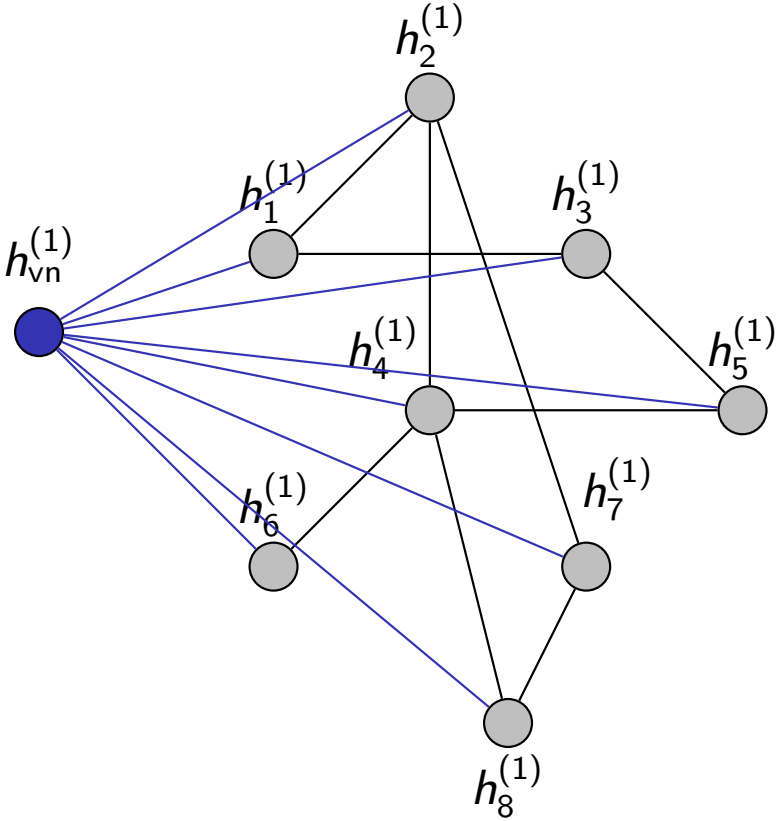
$$H^{(1)} = \mathrm{ReLU}\left(\tilde{A}XW^{(1)}\right).$$



Graph Neural Networks with Virtual Nodes

We consider graph-structured data to be the combination of

- a graph $G = (V, E)$;
- node-features $X = [x_1, \ldots, x_n]^T$;
- and add a virtual node connected to all nodes in V (Pham, et al., 2017; Gilmer et al., 2017).



The model equations for MPNN+VN,

$$h_{\text{vn}}^{(\ell+1)} = \mathrm{U}_{\text{vn}}^{(\ell)}(h_{\text{vn}}^{(\ell)}, \ \mathrm{M}_{\text{vn}}^{(\ell)}(\{h_j^{(\ell)} : j \in V\})),$$
$$h_i^{(\ell+1)} = \mathrm{U}^{(\ell)}(h_i^{(\ell)}, \ \mathrm{M}^{(\ell)}(\{h_j^{(\ell)} : j \in \mathcal{N}(i)\}), h_{\text{vn}}^{(\ell)}).$$

E.g., one layer of the GCN + VN,

$$h_{\text{vn}}^{(\ell+1)} = \sigma\left(\Omega_{\text{vn}}^{(\ell)} h_{\text{vn}}^{(\ell)} + \frac{1}{n}\sum_{j=1}^{n} W_{\text{vn}}^{(\ell)} h_j^{(\ell)}\right), \quad h_i^{(\ell+1)} = \sigma\left(\Omega^{(\ell)} h_i^{(\ell)} + \sum_{j \in N_i} \frac{1}{\sqrt{d_i d_j}} W^{(\ell)} h_j^{(\ell)} + h_{\text{vn}}^{(\ell)}\right),$$

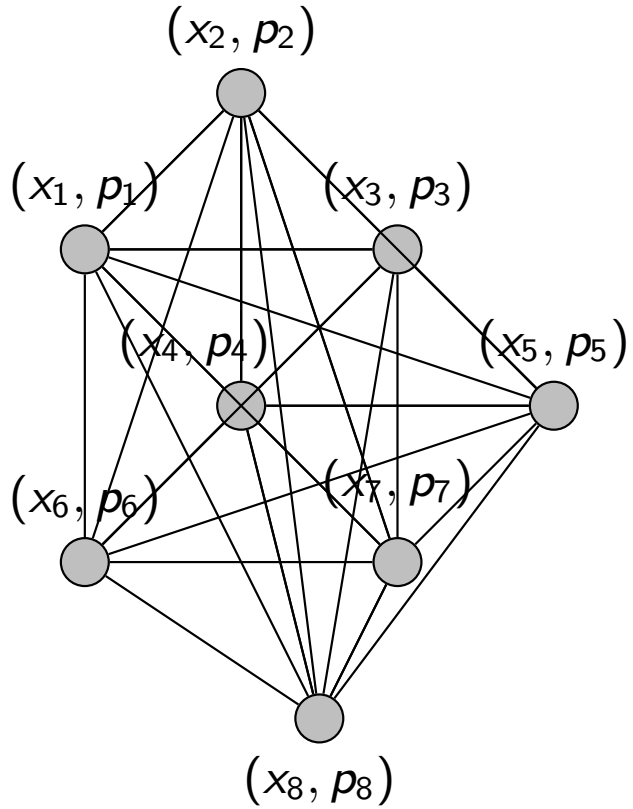
where d_i denotes the node degree of node i and $\Omega_{\text{vn}}^{(\ell)}, W_{\text{vn}}^{(\ell)}, \Omega^{(\ell)}, W^{(\ell)}$ denote trainable weight matrices.

Graph Transformers (GTs)

We consider graph-structured data to be the combination of

- a graph $G = (V, E)$;
- node-features $X = [x_1, \ldots, x_n]^T$;
- we construct positional encoding p_i for node $i \in V$ (random walk or spectral) and concatenate them with x_i.

We use the dense 'query, key, value self-attention scheme' to message pass the concatenation of node features and positional encodings (Kreuzer et al., 2021; Ying et al., 2021).



The GraphGPS architecture (Rampášek et al., 2022) runs both a GT and MPNN layer in parallel and then sums node representations.

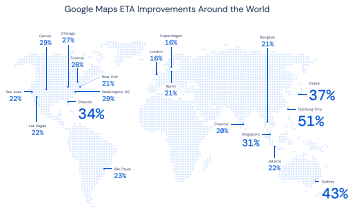
Complexity Comparison

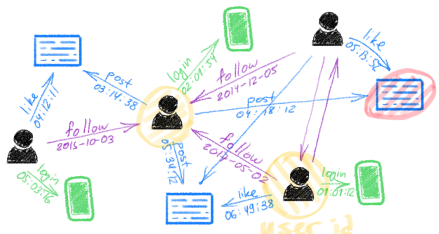
MPNN: $\mathcal{O}(|E|)$, MPNN+VN: $\mathcal{O}(|E| + |V|)$, GT: $\mathcal{O}(|V|^2)$.

Academic and Industrial Success of GNNs

Empirical and Theoretical **Research**:

- expressivity analysis of GNNs (Xu et al., 2019; Geerts and Reutter, 2022);
- bottlenecks, e.g., oversmoothing and over-squashing (Alon and Yahav, 2020);
- robustness of GNNs (Günnemann, 2022; Abbahaddou et al., 2024);
- generalisation of GNNs (Vasileiou et al., 2025).





Successful **Applications** of GNNs:

- Google Maps (Lange and Perez, 2020);
- Twitter (Bronstein, 2020);
- Amazon, Alibaba, Pinterest & Uber Eats (Virinchi et al., 2022; Wang et al., 2018; Ying et al., 2018; Jain et al., 2019);
- Discovery of two *new antibiotics* (Stokes et al., 2020; Liu et al., 2023);
- LinkedIn (Borisyuk et al., 2024);
- Snapchat (Zhao et al., 2025).

Understanding Virtual Nodes: Oversmoothing, Oversquashing, and Node Heterogeneity

Southern, Di Giovanni, Bronstein & Lutzeyer (2025, ICLR)



Oversmoothing I

Oversmoothing: Refers to the phenomenon of node representations becoming indistinguishable in deep GNNs.

Setting: We assume a linear setting without activation functions, where

$$\text{MPNN:} \qquad H^{(\ell+1)} = H^{(\ell)} + AH^{(\ell)}W^{(\ell)},$$

$$\text{MPNN +VN:} \qquad H^{(\ell+1)} = H^{(\ell)} + AH^{(\ell)}W^{(\ell)} + \frac{1}{n}\mathbb{1}\mathbb{1}^\top H^{(\ell-1)}Q^{(\ell)}.$$

The PairNorm model (related to *MPNN - VN*, i.e., $Q^{(\ell)} = -I$ for all $\ell \in \mathbb{N}$) is designed to prevent oversmoothing. Previous work credited the strong empirical performance of MPNN+VN to its ability to emulate PairNorm.

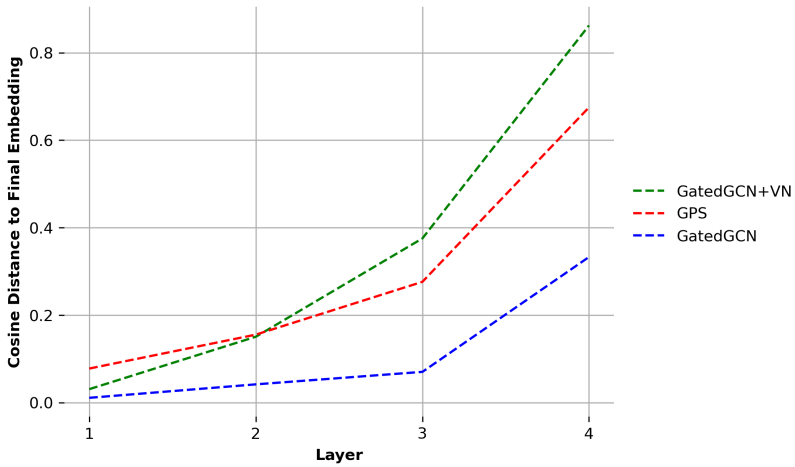
Theorem (Expressivity in Terms of Polynomial Filters)

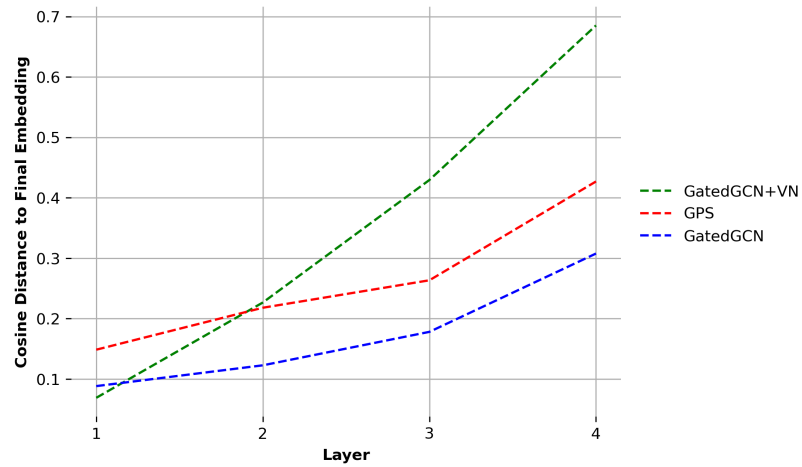
There are polynomial filters $\text{Mean}\big(\sum_{k=0}^{m} A^k H\Theta_k\big)^\top$ that can be learned by *MPNN* but not by *MPNN - VN*.
Conversely, any polynomial filter learned by *MPNN* can also be learned by *MPNN + VN*.
Furthermore, there exist polynomial filters that can be learned by *MPNN + VN* but not by *MPNN*.

$\Rightarrow$ The empirical performance gain of MPNN+VN is unlikely to result from its ability to prevent oversmoothing.

Oversmoothing II

In fact, we observe MPNN+VN to smooth node features more drastically than the baselines on the Peptides-func and Peptides-struct datasets:





Oversmoothing: Refers to the phenomenon of node representations becoming indistinguishable in deep GNNs. We further specify,

- for node-level tasks indistinguishable node representations are clearly **harmful**.
- However, for graph-level tasks oversmoothing may be **beneficial** if the common node representation aligns with the label distribution.

Oversquashing I

Oversquashing prevents the underlying model from exchanging information between nodes at large **commute time** τ, where $\tau(i,j)$ measures the expected number of steps for a random walk to commute between i and j.

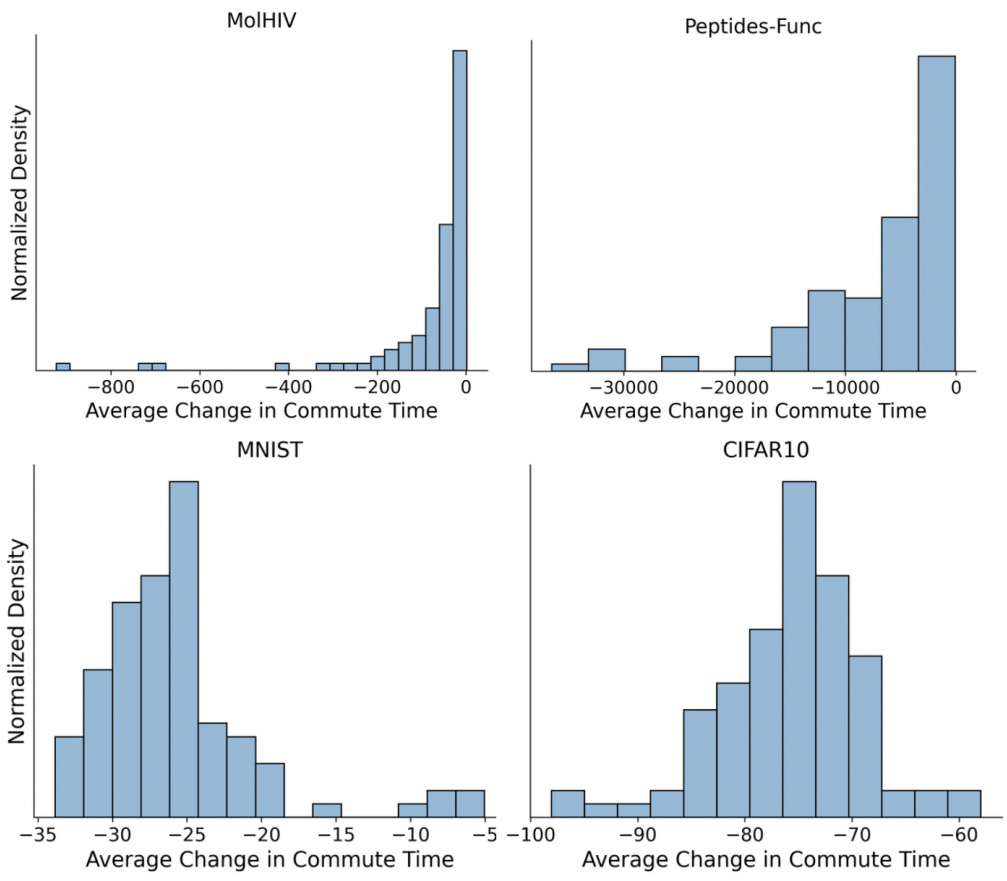
Theorem

The average commute time after adding a VN changes as

$$\frac{1}{|V|^2}\sum_{i,j=1}^{n}(\tau_{\text{vn}}(i,j)-\tau(i,j)) = \frac{4|E|}{|V|}\sum_{\ell=1}^{n-1}\frac{1}{\lambda_\ell(\lambda_\ell+1)}\Big(\frac{|V|}{|E|}\lambda_\ell - 1\Big), \qquad (1)$$

where $0=\lambda_0<\lambda_1\leq\ldots\leq\lambda_{n-1}$ denote the eigenvalues of the unnormalised graph Laplacian $L=D-A$.

Oversquashing II



Observation: On our datasets the addition of a VN results in a reduction of the average commute time and therefore alleviates potential oversquashing issues.

Node Heterogeneity

Node Heterogeneity: Do nodes affect a central node's representation heterogeneously, all equally or not at all?

We consider nodes i and k that are separated by more than 2 hops and study the Jacobian $\partial h_i^{(\ell+1)} / \partial h_k^{(\ell-1)}$, which allows us to analyse the dependence of hidden states on each other:

- Trivially, for *MPNN*, the Jacobian equals 0.
- For *MPNN* + *VN*, the Jacobian is independent of k.
- For *MPNN* + VN_G, the Jacobian is dependent on k, where MPNN + VN_G is formulated as follows,

$$
\begin{aligned}
h_{i,\mathrm{loc}}^{(\ell+1)} &= \mathrm{up}^{(\ell)}(h_i^{(\ell)}, \mathrm{agg}^{(\ell)}(\{h_j^{(\ell)} : j \in \mathcal{N}(i)\})), \\
h_{\mathrm{vn}}^{(\ell+1)} &= \mathrm{up}_{\mathrm{vn}}^{(\ell)}(h_{\mathrm{vn}}^{(\ell)}, \mathrm{agg}_{\mathrm{vn}}^{(\ell)}(\{h_{j,\mathrm{loc}}^{(\ell+1)} : j \in V\})), \\
h_i^{(\ell+1)} &= \widetilde{\mathrm{up}}^{(\ell)}(h_{i,\mathrm{loc}}^{(\ell+1)}, h_{\mathrm{vn}}^{(\ell+1)}).
\end{aligned}
$$

- For *GTs*, the Jacobian depends on both i and k.

Empirical Results I

Observation: First layer attention maps of the self-attention matrix in the GraphGPS framework for different datasets.

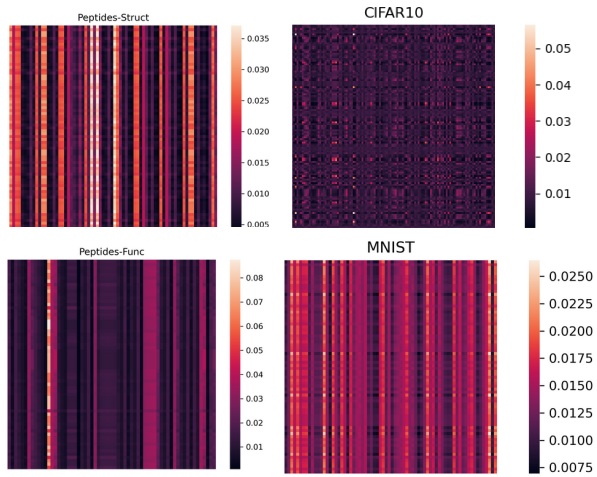


Table: Effects of projecting the non-local part of GraphGPS onto the mean and its comparison to using a VN. Arrows indicate if the performance improves with higher (↑) or lower (↓) scores. We also report the standard deviation of the column sums in the first attention layer.

Method	Pept-Func (↑)	Pept-Struct (↓)	MNIST (↑)	CIFAR10 (↑)
GPS	0.6534 ±.0091	0.2509 ±.0014	98.051 ±.126	**72.298** ±.356
GPS + projection	0.6498 ±.0054	**0.2487** ±.0011	**98.176** ±.120	71.455 ±.513
GatedGCN+PE+VN	**0.6712** ±.0066	**0.2481** ±.0015	**98.122** ±.102	70.280 ±.380
std attention layer	0.0011	0.0007	0.0006	0.0038

Empirical Results II

Table: Test performance on two LRGB datasets (Dwivedi et al., 2023) and three other benchmarks from (Dwivedi et al., 2022). For Peptides-Func and Peptides-Struct, ± std is shown over 4 runs whilst the remaining datasets are over 10 runs (missing values from literature are indicated by '-'). The first, second and third best results for each task are color-coded.

Method	Pept-Func (↑)	Pept-Struct (↓)	MNIST (↑)	CIFAR10 (↑)	MalNet-Tiny (↑)
GCN	0.5930 ±0.0023	0.3496 ±0.0013	90.705 ±0.218	55.710 ±0.381	81.0
GINE	0.5498 ±0.0079	0.3547 ±0.0045	96.485 ±0.252	55.255 ±1.527	88.98 ±0.56
GatedGCN	0.5864 ±0.0077	0.3420 ±0.0013	97.340 ±0.143	67.312 ±0.311	92.23 ±0.65
GatedGCN+PE	0.6765 ±0.0047	0.2477 ±0.0009	-	69.948 ±0.499	-
GatedGCN+PE-ViT	0.6942 ±0.0075	0.2465 ±0.0015	98.460 ±0.090	71.580 ±0.090	-
GatedGCN+PE-Mixer	0.6932 ±0.0017	0.2508 ±0.0007	98.320 ±0.040	70.600 ±0.220	-
CRaWl	0.7074 ±0.0032	0.2506 ±0.0022	97.940 ±0.050	69.010 ±0.259	-
DRew	**0.7150** ±0.0044	0.2536 ±0.0015	-	-	-
SAN+RWSE	0.6439 ±0.0075	0.2545 ±0.0012	-	-	-
EGT	-	-	98.173 ±0.087	68.702 ±0.409	-
GRIT	0.6988 ±0.0082	**0.2460** ±0.0012	98.108 ±0.111	**76.468** ±0.881	-
GPS	0.6534 ±0.0091	0.2509 ±0.0014	98.051 ±0.126	72.298 ±0.356	93.50 ±0.41
Exphormer	0.6527 ±0.0043	0.2481 ±0.0007	98.414 ±0.035	74.690 ±0.125	**94.02** ±0.21
GatedGCN+PE+VN	0.6712 ±0.0066	0.2481 ±0.0015	98.122 ±0.102	70.280 ±0.380	92.62 ±0.57
GatedGCN+PE+VN$_G$	0.6822 ±0.0052	**0.2458** ±0.0006	**98.626** ±0.100	76.080 ±0.330	93.67 ±0.37

$\Rightarrow$ Our MPNN+VN$_G$ consistently outperforms its MPNN+VN counterpart.

Reaction to Recent Position Paper: GNNs for Digital Pathology

Position: Graph Learning Will Lose Relevance Due To Poor Benchmarks (Bechler-Speicher, et al., 2025)

Issues:

- "Missing transformative real-world applications and supporting benchmarks"
- "Graphs are not necessarily constructed in a meaningful way"
- "Bad benchmarking culture"

"This position paper calls for a paradigm shift toward more meaningful benchmarks, rigorous evaluation protocols, and stronger collaboration with domain experts to drive impactful and reliable advances in graph learning research, unlocking the potential of graph learning."

$\Rightarrow$ These are good points. Our recent work on GNNs on digital pathology is a step in this direction (Kormann et. al, 2025).

HIEGNet: A Heterogenous Graph Neural Network Including the Immune Environment in Glomeruli Classification

Kormann, Ramuz, Nisar, Schaadt, Annuth, Doerr, Feuerhake, Lampert & Lutzeyer
(2025, MIDL)





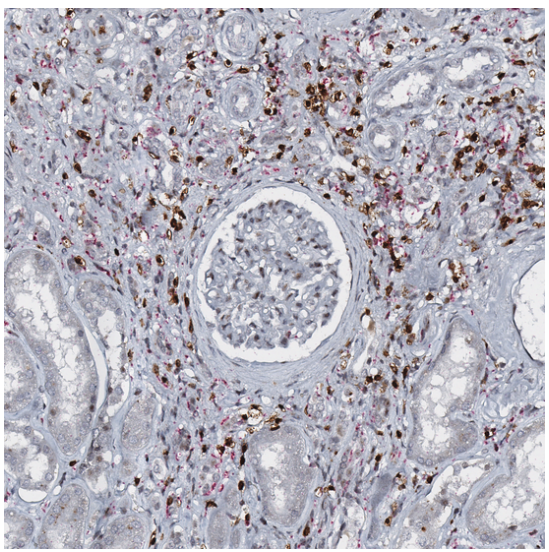



Glomeruli Classification

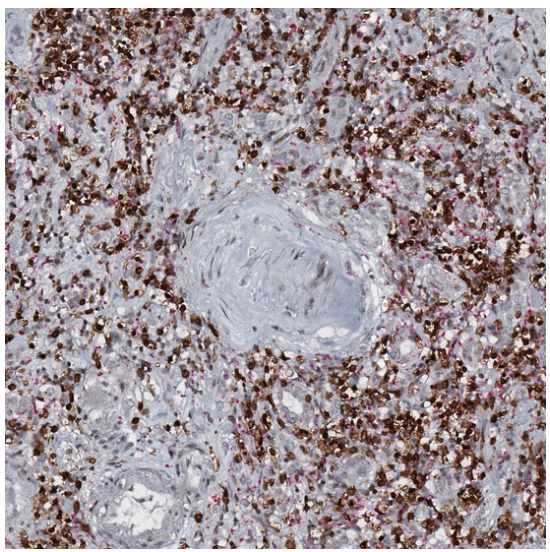
Context: Glomeruli are functional units in the kidney that are relevant in chronic kidney disease diagnosis.

Machine Learning Task: Glomeruli Health State Classification.

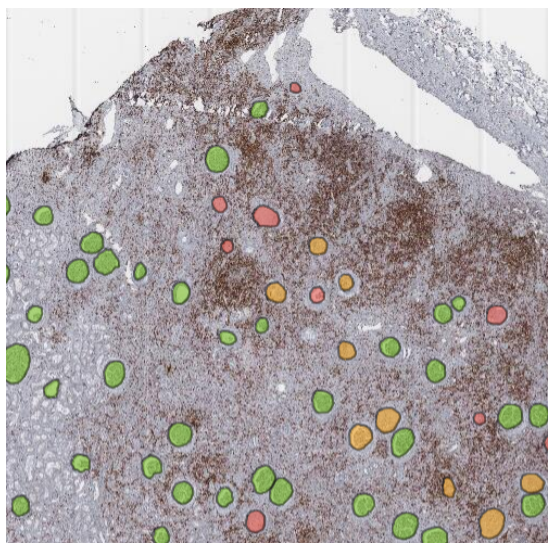
Relevant Data: Whole Slide Images.



Healthy Glomerulus



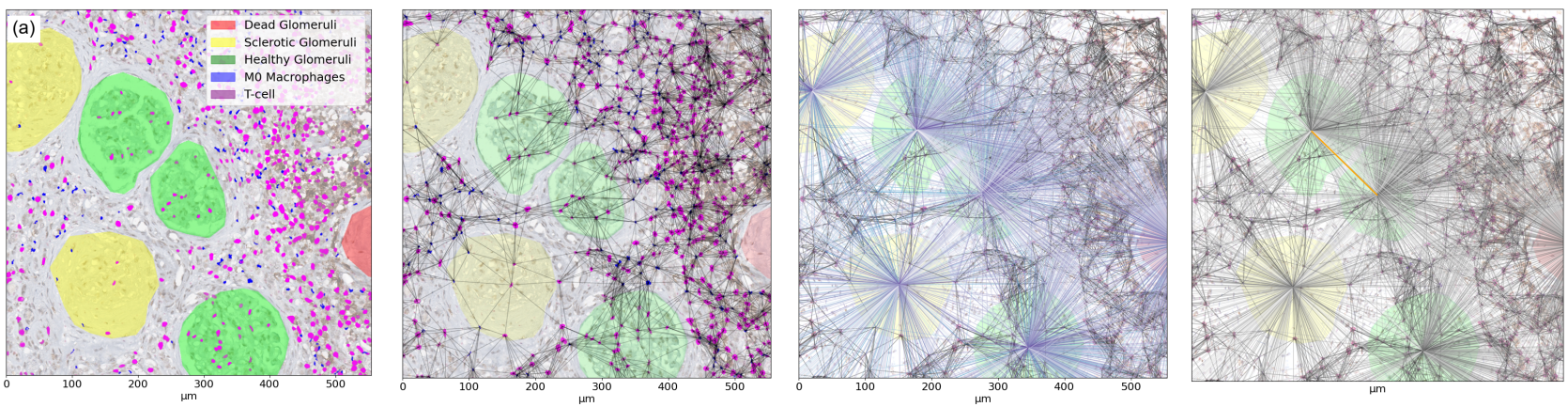
Dead Glomerulus



Tissue Slice

Heterogeneous Graph Construction

1) We start with segmented whole slide images, giving rise to different node types.
2) We draw edges between immune cells using a 5-nearest neighbour graph construction combined with a maximal edge length of $\epsilon = 100$ µm.
3) We draw edges between immune cells and glomeruli within a radius $\epsilon = 277$ µm around glomeruli.
4) We draw edges between glomeruli using a ϵ-neighbourhood construction with $\epsilon = 138.6$ µm.
5) We furthermore attribute nodes with shape-based features and local binary patterns.

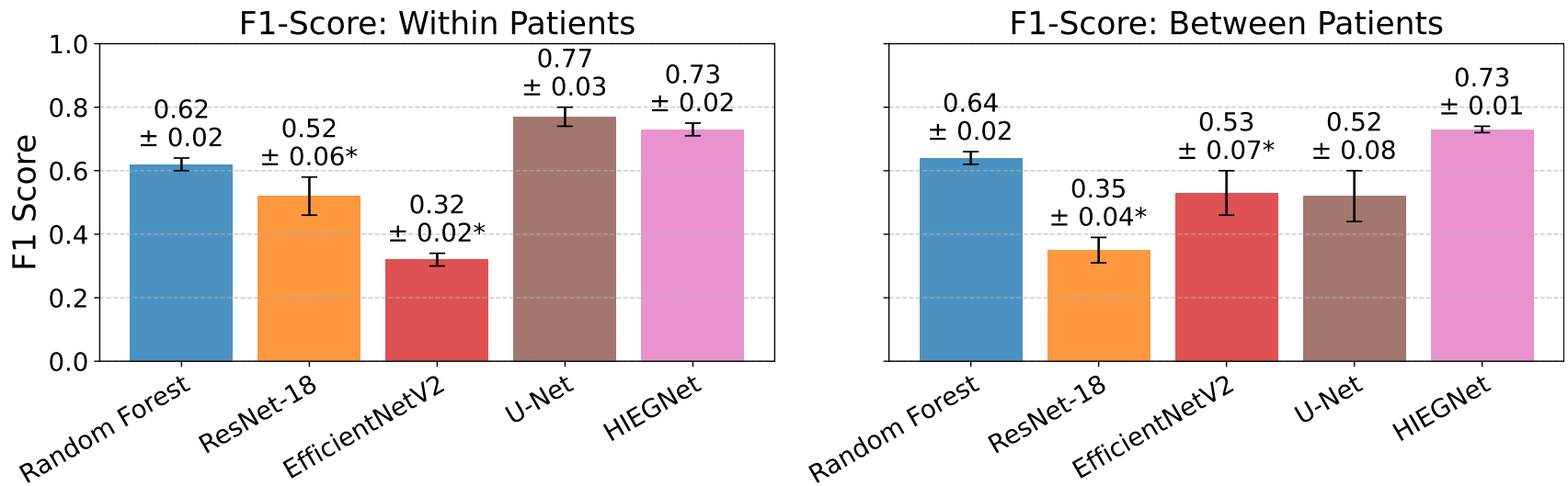


Graph Neural Network & Results

We then define a heterogeneous GNN operating on the heterogeneous graph we constructed. Here we fit separate update $U_r^{(\ell)}$ and message passing functions $M_r^{(\ell)}$ per edge type.

We hence define a message passing layer of HIEGNet as

$$h_v^{(\ell+1)} = \sum_{r \in \mathcal{R}} U_r^{(\ell)} \left(h_v^{(\ell)}, M_r^{(\ell)} \left(h_v^{(\ell)}, \{ h_u^{(\ell)} : (v, r, u) \in E \} \right) \right).$$



$\Rightarrow$ In a dataset of 6 patients, we generalise to three unseen patient better than the baseline models.

Conclusions

- Virtual Nodes (VNs) enable global information propagation in GNNs in a computationally efficient way.
- MPNN + VN falls in between MPNN and Graph Transformers.
- VNs are particularly effective for graph-level tasks especially, and the following is conjecture, if the chosen VN-aggregation scheme projects into a label-informative subspace.
- We should find more meaningful benchmarks for graph learning and our work on digital pathology is a step in this direction.

Thank you for your attention!

johanneslutzeyer.com

References

Y. Abbahaddou, S. Ennadir, J. F. Lutzeyer, M. Vazirgiannis & H. Boström, "Bounding the Expected Robustness of Graph Neural Networks Subject to Node Feature Attacks," *International Conference on Learning Representations(ICLR),* 2024.

Y. Abbahaddou, F. D. Malliaros, M. Vazirgiannis & J. F. Lutzeyer, "Centrality Graph Shift Operators for Graph Neural Networks," *arXiv 2411.04655,* 2024.

Y. Abbahaddou, F. D. Malliaros, J. F. Lutzeyer, A. M. Aboussalah & M. Vazirgiannis, "Gaussian Mixture Models Based Augmentation Enhances GNN Generalization," *arXiv 2411.08638,* 2024.

U. Alon & E. Yahav, "On the Bottleneck of Graph Neural Networks and its Practical Implications," In: *International Conference on Learning Representations (ICLR)*, 2020.

M. Bechler-Speicher, B. Finkelshtein, F. Frasca, L. Müller, J. Tönshoff, A. Siraudin, V. Zaverkin, M. M. Bronstein, M. Niepert, B. Perozzi, M. Galkin & C. Morris, "Position: Graph Learning Will Lose Relevance Due To Poor Benchmarks," *arXiv:2502.14546*, 2025.

F. Borisyuk, S. He, Y. Ouyang, M. Ramezani, P. Du, X. Hou, C. Jiang, N. Pasumarthy, P. Bannur, B. Tiwana, P. Liu, "LiGNN: Graph Neural Networks at LinkedIn," *arXiv:2402.11139*, 2024.

S. Brody, U. Alon & E. Yahav, "How Attentive Are Graph Attention Networks?," In: *International Conference on Learning Representations (ICLR)*, 2022.

M. Bronstein, "Graph ML at Twitter," *Twitter Engineering Blog Post*, https://blog.twitter.com/engineering/en_us/topics/insights/2020/graph-ml-at-twitter, 2020.

M. Chatzianastasis, J. F. Lutzeyer, G. Dasoulas & M. Vazirgiannis, "Graph Ordering Attention Networks," *Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI),* 2023.

G. Dasoulas, J. F. Lutzeyer & M. Vazirgiannis, "Learning Parametrised Graph Shift Operators," In: *International Conference on Learning Representations (ICLR)*, 2021.

A. Deac, M. Lackenby & P. Veličković, "Expander Graph Propagation," *arXiv:2210.02997*, 2022.

V. P. Dwivedi, L. Rampášek, M. Galkin, A. Parviz, G. Wolf, A. T. Luu & D. Beaini, "Long range graph benchmark," *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

V. P. Dwivedi, C. K. Joshi, A. T. Luu, T. Laurent, Y. Bengio & X. Bresson, "Benchmarking graph neural networks," *Journal of Machine Learning Research (JMLR)*, 2023.

S. Ennadir, Y. Abbahaddou, J. F. Lutzeyer, M. Vazirgiannis & H. Boström, "A Simple and Yet Fairly Effective Defense for Graph Neural Networks," *Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI),* 2024.

F. Geerts & J. L. Reutter, "Expressiveness and Approximation Properties of Graph Neural Networks," *International Conference on Learning Representations (ICLR)*, 2022.

J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals & G. E. Dahl, "Neural message passing for Quantum chemistry," *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

S. Günnemann, "Graph Neural Networks: Adversarial Robustness," *Graph Neural Networks: Foundations, Frontiers, and Applications*, pp. 149–176, 2022.

A. Jain, I. Liu, A. Sarda & P. Molino, "Food Discovery with Uber Eats: Using Graph Learning to Power Recommendations," *Uber Engineering Blog Post*, https://eng.uber.com/uber-eats-graph-learning/, 2019.

Thomas N. Kipf & M. Welling, "Semi-supervised classification with graph convolutional networks," *International Conference on Learning Representations (ICLR)*, 2017.

N. Kormann, M. Ramuz, Z. Nisar, N. S. Schaadt, H. Annuth, B. Doerr, F. Feuerhake, T. Lampert & J. F. Lutzeyer, "HIEGNet: A Heterogenous Graph Neural Network Including the Immune Environment in Glomeruli Classification," *Medical Imaging with Deep Learning Conference,* 2025.

N. Kormann, M. Ramuz, Z. Nisar, N. S. Schaadt, H. Annuth, B. Doerr, F. Feuerhake, T. Lampert & J. F. Lutzeyer, "HIEGNet: A Heterogenous Graph Neural Network Including the Immune Environment in Glomeruli Classification," *Medical Imaging with Deep Learning Conference,* 2025.

D. Kreuzer, D. Beaini, W. Hamilton, V. Létourneau & P. Tossou, "Rethinking graph transformers with spectral attention," *Advances in Neural Information Processing Systems (NeurIPS*, 2021.

O. Lange & L. Perez, "Traffic prediction with advanced Graph Neural Networks," *DeepMind Research Blog Post*, `https://deepmind.com/blog/article/traffic-prediction-with-advanced-graph-neural-networks`, 2020.

G. Liu, D. B. Catacutan, K. Rathod, K. Swanson, W. Jin, J. C. Mohammed, A. Chiappino-Pepe, S. A. Syed, M. Fragis, K. Rachwalski, J. Magolan, M. G. Surette, B. K. Coombes, T. Jaakkola, R. Barzilay, J. J. Collins, J. M. Stokes, "Deep learning-guided discovery of an antibiotic targeting Acinetobacter baumannii," *Nature Chemical Biology*, pp. 1–9, 2023.

G. Michel, G. Nikolentzos, J. Lutzeyer & M. Vazirgiannis, "Path Neural Networks: Expressive and Accurate Graph Neural Networks," *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023.

T. Pham, T. Tran, H. Dam & Svetha Venkatesh, "Graph classification via deep learning with virtual nodes," *arXiv:1708.04357*, 2017.

L. Rampášek, M. Galkin, V. P. Dwivedi, A. T. Luu, G. Wolf & D. Beaini, "Recipe for a general, powerful, scalable graph transformer," *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

G. Salha-Galvan, J. F. Lutzeyer, G. Dasoulas, R. Hennequin & M. Vazirgiannis, "Modularity-Aware Graph Autoencoders for Joint Community Detection and Link Prediction," *arxiv:2202.00961*, 2022.

M. E. A. Seddik, C. Wu, J. F. Lutzeyer & M. Vazirgiannis, "Node Feature Kernels Increase Graph Convolutional Network Robustness," *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

J. Southern, F. Di Giovanni, M. Bronstein & J. F. Lutzeyer, "Understanding Virtual Nodes: Oversmoothing, Oversquashing, and Node Heterogeneity," *International Conference on Learning Representations (ICLR)*, 025.

J. M. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. M. Donghia, C. R. MacNair, S. French, L. A. Carfrae, Z. Bloom-Ackermann, V. M. Tran, A. Chiappino-Pepe, A. H. Badran, I. W. Andrews, E. J. Chory, G. M. Church, E. D. Brown, T. S. Jaakkola, R. Barzilay & J. J. Collins, "A Deep Learning Approach to Antibiotic Discovery," *Cell*, pp. 688–702, 2020.

A. Vasileiou, S. Jegelka, R. Levie & C. Morris, "Survey on Generalization Theory for Graph Neural Networks," *arXiv:2503.15650,* 2025.

S. Virinchi, A. Saladi & A. Mondal, "Recommending Related Products Using Graph Neural Networks in Directed Graphs," In: *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, 2022.

J. Wang, P. Huang, H. Zhao, Z. Zhang, B. Zhao & Dik Lun Lee, "Billion-scale Commodity Embedding for E-Commerce Recommendation in Alibaba," In Proceedings of the *24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pp. 839–848, 2018.

K. Xu, W. Hu, J. Leskovec & S. Jegelka."How powerful are graph neural networks?", *International Conference on Learning Representations (ICLR)*, 2019.

R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton & J. Leskovec, "Graph Convolutional Neural Networks for Web-Scale Recommender Systems," In Proceedings of the *24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pp. 974–983, 2018.

C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen & T.-Y. Liu, "Do transformers really perform badly for graph representation?," *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

T. Zhao, Y. Liu, M. Kolodner, K. Montemayor, E. Ghazizadeh, A. Batra, Z. Fan, X. Gao, X. Guo, J. Ren, S. Park, P. Yu, J. Yu, S. Vij,& N. Shah, "GiGL: Large-Scale Graph Neural Networks at Snapchat," *arXiv:2502.15054* , 2025.

Y. Zhou, H. Zheng & X. Huang, "Graph Neural Networks: Taxonomy, Advances and Trends," *arXiv:2012.08752*, 2020.