Exploratory Paper Analysis: Why Does Your Graph Neural Network Fail on Some Graphs?

Mateusz Korytkowski (mkk51)

MPhil in Advanced Computer Science
University of Cambridge

Abstract

Graph Neural Networks (GNNs) often succeed on benchmark graphs yet fail on others in ways that are not fully explained by architectural narratives such as over-smoothing or over-squashing. The target paper reframes this problem through exact generalisation error, showing that alignment between graph structure, node features, and labels is the key driver of when a model generalises [1]. This analysis first traces the backward-thinking context that motivated this reframing and then proposes a forward-thinking direction: predictive model auditing as a pre-training diagnostic. Work is complemented with the experiments on a synthetic stress suite and real datasets, including a heterophily benchmark, to illustrate how alignment signals track performance and to test the audit idea.

1 Backward-thinking.

This analysis examines the paper "Why Does Your Graph Neural Network Fail on Some Graphs? Insights from Exact Generalisation Error" [1]. The paper asks why GNN performance varies across graphs and proposes a principled explanation that links generalisation to alignment between graph structure and signal. This analysis uses the paper to ground the backward looking discussion, to identify the problem that led to producing this paper, to clarify the limits of prior explanations, and to motivate a forward looking diagnostic proposal.

Problem setting. GNNs are a standard tool for graph-structured learning, yet performance varies drastically across graph families. This variability is surprising because many GNNs share similar expressivity bounds and are trained in comparable ways, but they still behave inconsistently across datasets. The inconsistency appears even when evaluation protocols are similar, which suggests that properties of the graph and labels can dominate architectural differences. The field has historically explained failures with architectural narratives: over-smoothing in deep message passing, over-squashing under long-range dependencies, or limits of the 1-WL expressivity class. These explanations are useful, but they do not predict why two closely related GNNs can diverge sharply in practice [1]. That gap motivates a closer look at what existing theory can and cannot explain.

Limits of prior theory. The literature contains generalisation error bounds for GNNs, but they are often loose, tied to a single architecture, or too abstract to diagnose real failures. As a result, researchers have strong but fragmented stories, as architectural limitations explain some trends, while empirical benchmarks suggest other trends. These bounds rarely translate into actionable diagnostics for practitioners deciding which model to use for a given dataset. The missing link is a principled, model-agnostic quantity that explains when the graph structure helps and when it hurts [1]. This missing link sets up the paper's reframing.

The paper's reframing. The paper addresses this gap by deriving an *exact generalisation error* for a broad class of GNNs under a transductive, fixed-design setting. The key move is a signal-processing perspective where graph convolutions are interpreted as graph filters, and the alignment between the spectral structure of the graph and the signal in node features/labels becomes the decisive factor. The analysis applies to convolutional, PageRank-based, and attention models, which makes the reframing broadly applicable [1, 2]. This shifts the question from *which*

architecture is best to *when does graph structure provide aligned signal that a GNN can exploit?* This reframing also connects spatial and spectral views where many architectures can be described by their frequency response, so failures can be traced to misaligned signal frequencies rather than to a specific model family [1]. This reframing leads naturally to a spectral filter lens.

Graph filter lens. To make the reframing concrete, I adopt a spectral filter lens. Under this lens, GNN layers behave like filters in the graph spectral domain. The data signal is decomposed into graph frequencies, and a model's frequency response determines which parts of the signal are amplified or suppressed. This makes two concepts central: (i) the spectral distribution of the graph itself, and (ii) whether the target signal is concentrated in frequencies that a given model can preserve. The paper argues that failures often occur not because a model is weak in an abstract sense, but because the graph signal is misaligned with what the model can represent [1]. This view sets up the specific failure mechanisms the paper isolates.

Failure Mechanisms. Here are the core failure mechanisms that the paper highlights:

1) Misalignment. Even shallow GNNs can fail if the adjacency structure, features, and targets are misaligned. The paper formalises this with a misalignment measure. When the target signal does not lie in the subspace captured by the GNN representation, generalisation error increases. The definition uses a projection onto the representation space, which makes the loss of signal geometrically explicit. This explains why a simple concatenation baseline can outperform convolution in some regimes [1].

2) Heterophily as high-frequency signal. The paper connects homophily/heterophily to spectral properties: homophilic labels are low-frequency signals, while heterophilic labels are high-frequency. A homophily parameter is used to formalise this range, which allows the generalisation error to be traced as homophily changes. Convolutional GNNs are biased toward low-pass behaviour, which makes them struggle when the label signal is high-frequency. This ties failures to the graph's spectrum rather than only to the model class [1].

3) Attention limitations under repeated eigenvalues. For attention-based models, the paper shows a spectral limitation. When eigenvalues repeat, certain attention mechanisms cannot fully distinguish graph structure, revealing a structural blind spot even for flexible models. The analysis predicts a gap between GAT and Specformer in this setting, and the paper validates it with a synthetic construction that increases eigenvalue multiplicity. This again suggests that graph spectral properties are as important as model design [1]. These mechanisms imply concrete empirical signatures, which the paper then evaluates on data.

Empirical evidence in the paper. The paper uses real datasets to illustrate these mechanisms. In the misalignment analysis, convolution can lose to simple concatenation when the graph and features are poorly aligned, highlighting that graph structure is not universally beneficial. The study spans datasets such as Cora, CiteSeer, Wikipedia, Squirrel, and Chameleon, which helps to separate alignment effects from heterophily effects. For heterophily, the paper links poor performance to high-frequency label signals that are suppressed by common GNN filters and uses controlled perturbations of Cora to examine how accuracy changes as heterophilic edges are added. The empirical claims reinforce the conclusion that the failures are predictable once alignment and spectral structure are quantified [1]. This empirical picture leads directly to **why benchmarks can mislead** because standard benchmarks such as Cora and CiteSeer are biased toward alignment-friendly regimes, which lets GNNs appear strong in evaluation while still being fragile in settings with weaker alignment or heterophilic structure. The misalignment score is not necessarily high for heterophilic graphs, which implies that relying on a single benchmark property can hide failure modes. The broader implication is that failure is not an exception, and that rather it is the predictable outcome of misalignment between data and graph structure [1]. These observations motivate the backward looking conclusion that follows.

Backward-looking conclusion. The historical focus on architectural fixes explains only part of the failure landscape. The paper's key contribution is to show that *alignment and spectral structure are the controlling variables*. This reframes GNN failures from *model design flaws* to *data-model mismatch*, and it motivates a shift toward diagnostics that can be applied before training. This sets the stage for a forward-looking audit approach that makes the theory in the paper practical and actionable.

2 Forward-thinking: Predictive Model Auditing

A natural research direction is to operationalise the paper's insights as a *pre-training diagnostic*: Predictive Model Auditing. The idea is to compute a failure risk score from graph-structural signals, e.g., homophily, spectral summaries, and feature-label alignment proxies, before any training occurs. This moves the field from post-hoc explanations toward pre-flight assessments of when a GNN is expected to generalise. This proposal raises the question of what the audit should predict in practice.

What the audit predicts The audit is not a single number that replaces training. Instead, it is a set of warnings and expectations. If a graph exhibits low homophily and a feature-label signal dominated by high-frequency components, the audit would predict high risk for standard low-pass GNNs and suggest either alternative filters or preprocessing. Conversely, in strongly aligned regimes, the audit predicts that standard convolutional GNNs should perform reliably, which helps avoid unnecessary architectural complexity. This leads to a concrete pipeline for how the audit is run.

1. **Audit stage:** Compute structural signals from the graph and label distribution to estimate a failure risk score (e.g., inverse homophily as a proxy for high-frequency label signals).
2. **Decision stage:** If risk is high, select architectures or preprocessing steps better aligned with the data (e.g., alternative filters, rewiring, or feature augmentation).
3. **Validation stage:** Stress-test predictions on controlled graph families to verify that the risk score correlates with actual failure rates.

Experimental design choices The model set includes GCN, GraphSAGE, GAT, GPRGNN, and MLP to cover the main architectural families used in practice. GCN provides a low-pass convolutional baseline, GraphSAGE offers a different aggregation rule, GAT represents attention-based architectures, and GPRGNN introduces a learned spectral response that can adapt to broader frequency profiles [6, 5, 9, 2]. MLP is included as a feature-only baseline so that the contribution of graph structure can be isolated. This selection allows the diagnostic to be tested across graph-aware and graph-agnostic models under comparable conditions. The stress suite uses synthetic families that expose controlled axes of variation. SBM with balanced and imbalanced communities provides direct control of homophily and class balance, LFR style graphs introduce more realistic community structure with tunable mixing [7], and degree-skew graphs test robustness under heavy-tailed degree distributions. Together these families cover homophily control, community realism, and degree heterogeneity in a way that a single generator cannot.

Squirrel is included as a real dataset because it is a widely used heterophily benchmark and it is discussed in the target paper [1] and introduced as a heterophily dataset in Geom-GCN [8]. It provides a realistic setting where low-pass biases can fail, which makes it a direct test of the diagnostic beyond homophilic benchmarks such as Cora and CiteSeer. The audit score combines homophily, a feature-label alignment proxy, and a spectral summary such as lambda max. Homophily captures label smoothness, alignment measures how much label signal is represented in features, and the spectral term reflects how graph structure affects filtering behaviour. Combining them turns the diagnostic into a multi-signal predictor rather than a restatement of homophily alone. These choices set up the empirical probes that follow.

Audit validation experiments. I ran a stress suite of experiments to test whether structural signals predict performance. The goal is not to maximise accuracy, but to test whether the audit signals hold across families rather than only within a single generator. I start with the synthetic stress suite and then move to alignment and real dataset checks.

Synthetic stress suite. Across SBM balanced and imbalanced, LFR, and degree skew graphs, accuracy increases with homophily for graph-aware models while the MLP baseline remains comparatively flat. Results for this experiment are presented in Figure 1. The multi-signal risk score is strongly negatively correlated with test accuracy, for example about $-\mathbf{0.94}$ for GAT and $-\mathbf{0.66}$ for GCN across families, while MLP remains weakly correlated at about $-\mathbf{0.25}$. This supports the audit idea while also showing that better weighting of signals could further improve predictive strength. I then isolate feature alignment in a controlled sweep.

Alignment (feature-signal) sweep. Holding graph structure fixed and varying how much label signal appears in node features, GAT and GCN remain high and stable, GraphSAGE and GPRGNN show mid-range performance, and the

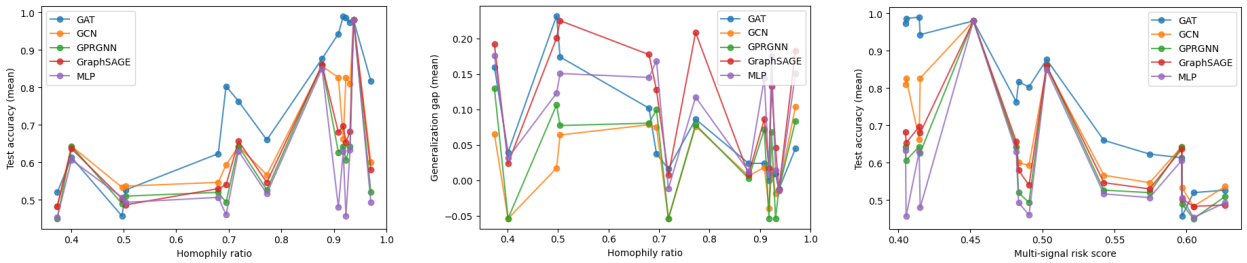

Figure 1: Stress suite results. Left to right: test accuracy versus homophily, generalisation gap versus homophily, and test accuracy versus the multi-signal risk score. Lines connect points within each model.

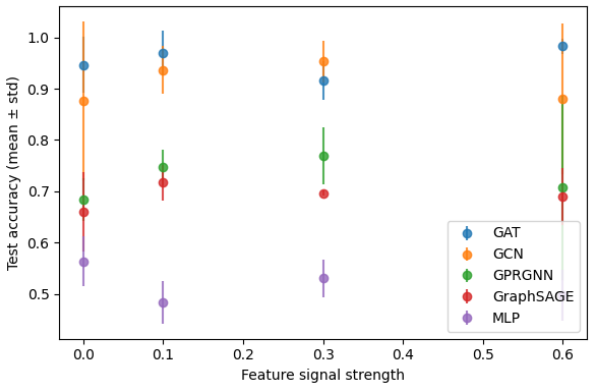

Figure 2: Alignment sweep. Test accuracy versus feature-signal strength with graph structure held fixed.

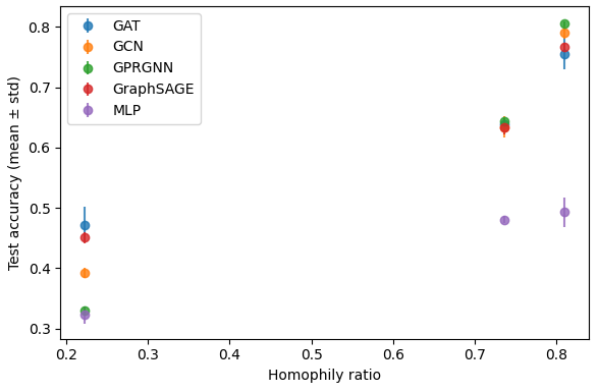

Figure 3: Real datasets. Homophily versus accuracy for Cora, CiteSeer, and Squirrel.

MLP baseline stays near chance. Results for this experiment are presented in Figure 2. For example, GAT stays above about **0.92** across feature strengths, while MLP remains around **0.48** to **0.56**. The trend is not perfectly monotonic across feature strengths, but the overall pattern supports the misalignment hypothesis. I then examine real datasets to see if the same signals appear in practice.

Real datasets. On Cora and CiteSeer [3, 4], which are more homophilic, GNNs consistently outperform MLP and GPRGNN performs best among the tested models. On Cora, GPRGNN reaches about **0.81** test accuracy while MLP is around **0.49**. On Squirrel, a heterophily benchmark with much lower homophily [8], overall accuracy drops and the ranking shifts, with GAT and GraphSAGE outperforming GCN. On Squirrel, GAT reaches about **0.47** while MLP is around **0.32**. Results for this experiment are presented in Figure 3 and Table 1 from the Appendix. This aligns with the paper's claim that benchmark success is linked to alignment-friendly regimes and highlights failure modes under heterophily [1]. The gap between graph-aware models and MLP is largest on Cora, smaller on CiteSeer, and smallest on Squirrel, which is consistent with the homophily ordering. The variance across seeds is modest, indicating the trend is stable rather than accidental. The ordering shift on Squirrel provides a concrete stress test for the audit, showing that model choice matters most when alignment is weak. This leads into the closing assessment and limitations.

Closing and limitations. These experiments are not exhaustive, but they demonstrate the core mechanism that performance tracks alignment-related signals across multiple families and real datasets, including a heterophily benchmark. Across these tests the audit pipeline is consistent, suggesting a promising pre-training diagnostic. In the stress suite, the risk score correlates with accuracy at about -0.94 for GAT and -0.66 for GCN, and on Squirrel GAT reaches about 0.47 versus MLP at 0.32. This supports the feasibility of a pre-training audit that predicts when GNNs are likely to fail, while also indicating that a next step is to expand the stress suite and calibrate a multi-signal risk score across broader graph properties. The audit should be explicit about what it captures, such as alignment and spectrum, and what it does not, such as label noise and task-specific complexity.

3 Appendix

Table 1: Real dataset test accuracy mean and standard deviation across five seeds.

Dataset	Model	Homophily	Test Acc (mean)	Std
Cora	GPRGNN	0.810	0.8050	0.0041
Cora	GCN	0.810	0.7898	0.0102
Cora	GraphSAGE	0.810	0.7664	0.0069
Cora	GAT	0.810	0.7556	0.0258
Cora	MLP	0.810	0.4932	0.0247
CiteSeer	GPRGNN	0.736	0.6434	0.0055
CiteSeer	GAT	0.736	0.6380	0.0111
CiteSeer	GCN	0.736	0.6340	0.0174
CiteSeer	GraphSAGE	0.736	0.6334	0.0125
CiteSeer	MLP	0.736	0.4806	0.0061
Squirrel	GAT	0.222	0.4720	0.0293
Squirrel	GraphSAGE	0.222	0.4507	0.0095
Squirrel	GCN	0.222	0.3925	0.0082
Squirrel	GPRGNN	0.222	0.3305	0.0065
Squirrel	MLP	0.222	0.3228	0.0145

References

[1] Nil Ayday, Mahalakshmi Sabanayagam, and Debarghya Ghoshdastidar. Why does your graph neural network fail on some graphs? insights from exact generalisation error, 2025.

[2] Edward Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[3] Mark Craven et al. Cora dataset. Cora dataset, 1998.

[4] C. Lee Giles et al. Citeseer dataset. CiteSeer dataset, 1998.

[5] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034, 2017.

[6] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[7] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4):046110, 2008.

[8] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, Bo Yang, Xuanjing Huang, and Jure Leskovec. Geom-gcn: Geometric graph convolutional networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[9] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. *CoRR*, abs/1710.10903, 2017.