Predicting L-Function Properties from Trace-Index Graphs using Graph Neural Networks

Tobias Weiss
tobias@tobias-weiss.org

Abstract

Can machine learning predict arithmetic properties of modular forms by operating on graph-structured representations of Fourier coefficient data? We investigate this question using Graph Neural Networks on trace-index graphs: 1000-node graph representations of individual newforms, where each node corresponds to a Fourier index and edges encode sequential adjacency, primality structure, and k-nearest-neighbor similarity in coefficient space. On 46,347 weight-2 newforms from the LMFDB, a 3-layer Chebyshev spectral filter network ($K = 5$) predicts the first L-function zero with $R^2 = 0.625$, analytic rank with 94.16% accuracy, and CM status with 100% accuracy. Spectral filters consistently outperform plain GCN, with the largest gains on rare-class detection (+38.87 pp in class-2 F_1). Cross-level generalization shows regression degrades modestly (-14% in R^2) while classification suffers more severely, particularly for rare rank-≥ 2 forms. The key difference from prior failed approaches is that trace-index graphs carry heterogeneous node features and data-dependent topology, giving message-passing architectures genuine local diversity to exploit — in contrast to vertex-transitive Cayley graphs where all nodes are structurally indistinguishable and GNNs provably cannot learn spectral properties.

1 Introduction

The Riemann Hypothesis (RH) is the most celebrated open problem in pure mathematics. One pathway toward RH runs through *modular forms*: holomorphic functions on the upper half-plane satisfying transformation laws under $\mathrm{SL}(2, \mathbb{Z})$ and its congruence subgroups. Each modular form carries Hecke eigenvalues $\{a_p(f)\}_{p \text{ prime}}$ that determine an associated L-function

$L(f, s)$ whose zeros encode deep arithmetic structure. The Generalized Riemann Hypothesis asserts all non-trivial zeros of every such L-function lie on the critical line.

Can machine learning discover relationships between Fourier coefficients and L-function properties? A natural approach is to construct graphs from algebraic structure and ask Graph Neural Networks (GNNs) to predict spectral or Hecke-eigenvalue properties. This approach, howevever, fundamentally fails: we showed in prior work [14] that GNNs cannot predict spectral properties of Cayley graphs of $\mathrm{SL}(2, \mathbb{F}_p)$ because vertex-transitivity forces all nodes to be structurally indistinguishable, preventing message-passing architectures from distinguishing graphs with different global spectral properties.

We take a fundamentally different approach: instead of constructing graphs from algebraic structure, we construct them from *Fourier coefficient data itself*. For each modular form we build a trace-index graph where nodes correspond to indices $n = 1, \ldots, 1000$ (carrying the Fourier coefficients $a_n(f)$), edges encode similarity and adjacency among those indices, and node features carry the actual coefficient values. Unlike Cayley graphs, these trace-index graphs carry heterogeneous node features and non-uniform edge structure, giving message-passing GNNs genuine local diversity to exploit.

Our contributions are:

1. A **trace-index graph paradigm** that maps modular forms to heterogeneous graph representations via their first 1000 Fourier coefficients, avoiding the vertex-transitivity failure mode that dooms prior approaches.

2. **Empirical evidence** that GNNs on these graphs predict L-function zeros ($R^2 = 0.625$), analytic rank (94.16% accuracy, macro $F_1 = 89.22\%$), and CM status (100%) on 46,347 newforms from the LMFDB.

3. Evidence that **spectral filters** (ChebConv $K = 5$) consistently outperform plain GCN, with the largest gains on rare-class classification (+38.87 pp in F_1 for rank class 2).

4. A **cross-level generalization analysis** showing regression transfers reasonably across conductor ranges ($R^2 = 0.538$, only 14% degradation), while classification degrades more substantially due to rare-class distribution shift.

5. A **design principle**: for applying GNNs to mathematical domains, the graph construction should reflect the mathematical structure of the data, not the algebraic structure of the underlying objects.

2 Mathematical Background

We provide the definitions needed for a self-contained presentation. For thorough treatments see Diamond and Shurman [6] or Milne [10].

2.1 Modular Forms and Hecke Operators

Let $\Gamma_0(N) = \left\{ \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \mathrm{SL}(2,\mathbb{Z}) : c \equiv 0 \pmod{N} \right\}$ be the principal congruence subgroup of level N. A **modular form of weight k and level N** is a holomorphic function $f : \mathbb{H} \to \mathbb{C}$ satisfying $f\left(\frac{az+b}{cz+d}\right) = (cz+d)^k f(z)$ for all $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in \Gamma_0(N)$, and holomorphic at the cusps. A form vanishing at every cusp is a **cusp form**; the space of weight-k cusp forms of level N is denoted $S_k(\Gamma_0(N))$.

Every modular form has a Fourier expansion $f(z) = \sum_{n=0}^{\infty} a_n(f)\, q^n$ where $q = e^{2\pi i z}$. A **newform** is an eigenform for all Hecke operators that is not a linear combination of forms of lower level.

For each prime p, the **Hecke operator** T_p acts on $S_k(\Gamma_0(N))$ by $(T_p f)(z) = \frac{1}{p}\sum_{j=0}^{p-1} f\left(\frac{z+j}{p}\right) + p^{k-1} f(pz)$. The T_p commute, so $S_k(\Gamma_0(N))$ admits a basis of simultaneous eigenforms: newforms f with $T_p f = a_p(f) \cdot f$ for every prime p. The eigenvalues $a_p(f)$ are the **Hecke traces** of f and satisfy the Ramanujan–Petersson bound [5]: $|a_p(f)| \le 2\sqrt{p}$.

2.2 L-Functions

Definition 1 (L-Function of a Newform)**.** *Let $f \in S_k(\Gamma_0(N))$ be a newform with Fourier coefficients $\{a_n(f)\}$. The **L-function** of f is*

$$L(f,s) = \sum_{n=1}^{\infty} a_n(f)\, n^{-s} = \prod_{p} \left(1 - a_p(f)\, p^{-s} + \chi(p)\, p^{k-1-2s}\right)^{-1}.$$

The Euler product shows that $\{a_p(f)\}$ completely determines $L(f,s)$, which admits analytic continuation to $\mathbb{C}$ and satisfies a functional equation. The non-trivial zeros encode deep arithmetic. The first zero $\gamma_1(f)$ (the smallest positive imaginary part of a non-trivial zero) is a computationally accessible proxy for the zero distribution.

2.3 Analytic Rank, BSD, and CM

The **analytic rank** is the order of vanishing at $s = k/2$: $\mathrm{rank}_{\mathrm{an}}(f) = \mathrm{ord}_{s=k/2}(L(f,s))$. For weight-2 newforms with trivial nebentypus, $L(f,s)$ is

the L-function of an elliptic curve $E/\mathbb{Q}$ by the Modularity Theorem [2, 15]. The Birch–Swinnerton-Dyer conjecture [1] predicts the analytic rank equals the Mordell–Weil rank of $E(\mathbb{Q})$.

A newform has **complex multiplication** (CM) if its associated elliptic curve has endomorphism ring larger than $\mathbb{Z}$. CM forms satisfy additional trace constraints and their L-functions factor as products of Dirichlet L-functions. CM detection has known algebraic criteria, making it a useful calibration target.

3 Why Prior Approaches Failed: Cayley Graphs

Before presenting our approach, it is instructive to understand why graph-based prediction of L-function properties is genuinely hard. The natural approach is to represent the arithmetic structure of $\mathrm{SL}(2, \mathbb{F}_p)$ as a graph and ask a GNN to predict spectral or Hecke-eigenvalue properties. This approach fails fundamentally, as shown in [14] across six systematic experiment tracks.

3.1 Cayley Graphs of $\mathrm{SL}(2, \mathbb{F}_p)$

A **Cayley graph** $\mathrm{Cay}(G, S)$ of a group G with generating set S has vertex set G and edges $\{(g, gs) : g \in G, s \in S\}$. We studied Cayley graphs of $\mathrm{SL}(2, \mathbb{F}_p)$ for primes $p = 2, 3, \ldots, 101$ using the generating sets from CayleyPy's `MatrixGroups.special_linear_fundamental_roots(2, p)`. The resulting graphs are 4-regular and vertex-transitive.

Cayley graphs connect to L-functions through two deep results. Lubotzky, Phillips, and Sarnak [9] constructed Ramanujan graphs from quaternion algebras, showing that Cayley graph eigenvalues satisfy the Ramanujan bound $|\lambda| \leq 2\sqrt{d-1}$. Pizer's theorem [11] establishes that the eigenvalues of certain Cayley graphs of $\mathrm{SL}(2, \mathbb{F}_p)$ coincide with Hecke eigenvalues of cusp forms in $S_2(\Gamma_0(p))$. This yields the chain:

$$\text{Cayley graph eigenvalues} \longrightarrow \text{Hecke eigenvalues} \longrightarrow L\text{-functions} \longrightarrow \zeta(s).$$

If a GNN could learn to predict Cayley graph spectral properties, it would in principle access arithmetic information about L-functions.

3.2 Why Message-Passing GNNs Cannot Succeed

A message-passing GNN with K layers produces node embeddings by aggregating information from each node's K-hop neighborhood. A fundamental result in GNN theory [16] states that message-passing GNNs are at most as

powerful as the 1-dimensional Weisfeiler-Leman (1-WL) test in distinguishing non-isomorphic graphs.

On vertex-transitive graphs, the 1-WL test assigns the same initial color to every vertex (they all have the same degree), and the refinement step preserves this uniformity because every vertex sees an identical multiset of neighbor colors. Therefore, after any number of WL iterations, all vertices receive the same color. This has an immediate consequence:

Observation 1. *Let $G = (V, E)$ be a vertex-transitive graph, and let $\phi : V \to \mathbb{R}^d$ be the node embedding produced by a K-layer message-passing GNN with node features $\mathbf{x}_v$. Then $\phi(v) = \phi(v')$ for all $v, v' \in V$. The GNN produces an identical embedding at every node.*

Consequently, when a message-passing GNN processes a vertex-transitive Cayley graph, it receives identical information from every node. The graph-level representation (obtained by pooling node embeddings) therefore depends only on graph-level statistics — $|V|$, $|E|$, etc. — and cannot capture arithmetic properties of p that determine the spectral gap or Hecke eigenvalues. This theoretical limit is confirmed empirically: across six experiment tracks in [14], GNNs on Cayley graphs achieve $R^2 < 0$ or at best a marginal $+0.042$ improvement over a logarithmic baseline.

The multi-generator experiments (generating multiple Cayley graphs per prime with different generator sets) add no improvement: the target (Hecke eigenvalue or spectral gap) depends on p's arithmetic, not on the combinatorial structure of the generating set. The graph representation simply does not encode the information needed.

Question 1. ***How can GNNs succeed at L-function prediction if they cannot even predict Cayley graph spectral gaps?***

The answer is to abandon algebraic graph representations and instead construct graphs from the Fourier coefficient data itself. The graph must encode the mathematical structure of the *coefficients*, not the algebraic structure of the underlying group. This is the design principle we follow in the next section.

4 Trace-Index Graphs

4.1 Trace-Index Graph Construction (Paradigm A)

The core idea: given a newform f with Fourier coefficients $\{a_1(f), \ldots, a_{1000}(f)\}$, we treat the coefficient vector as the basis of a graph. Each index $n =$

$1, \ldots, 1000$ becomes a node, carrying the coefficient $a_n(f)$ as a feature. Edges encode structural relationships among indices.

Definition 2 (Trace-Index Graph)**.** *Given a newform f with Fourier coefficients $\{a_1(f), \ldots, a_{1000}(f)\}$, the* ***trace-index graph*** *G_f has:*

- ***Nodes:*** *$V = \{1, \ldots, 1000\}$, with 5-dimensional features*

$$\mathbf{x}_i = (\log(i),\ a_i(f),\ a_i(f)/(2\sqrt{i}),\ \mathbf{1}_{i\ prime},\ \sqrt{i})^\top.$$

- ***Edges*** *from three sources:*
  1. ***Sequential:*** *$(i, i+1)$ for $i = 1, \ldots, 999$ — captures local autocorrelation in the trace sequence.*
  2. ***Prime:*** *(i, j) when both i and j are prime (168 primes ≤ 1000) — encodes the special role of prime indices in Hecke algebra.*
  3. ***kNN:*** *(i, j) when j is among the k=3 indices nearest to i in the space of normalized coefficient values $a_n(f)/(2\sqrt{n})$ — connects indices with similar trace magnitudes regardless of index proximity.*

Each graph has 1000 nodes and approximately 9,500 edges (999 sequential + complete prime-prime + 3NN). Three properties distinguish these from Cayley graphs: (i) heterogeneous node features (each node carries a unique Fourier coefficient that varies across indices and across newforms), (ii) non-vertex-transitive structure (prime-indexed nodes have extra edges), and (iii) data-dependent topology (the kNN edges depend on the actual coefficient values of f). These properties give the GNN local structural diversity to exploit.

4.2 Multiplicative Graph (Paradigm C, Control)

As a control, we construct **multiplicative graphs**: nodes are squarefree indices only (608 nodes), connected by divisibility-poset edges ($i \mid j$), with 3-dimensional features $(\log(i),\ a_i(f),\ a_i(f)/(2\sqrt{i}))$. These have $\sim$84,000 edges. The divisibility structure is the same for every newform, making the topology data-independent — a weaker representation that tests whether data-dependent kNN edges are essential.

Table 1: GCN baseline on all prediction targets (stratified split).

Target	Metric	Value
z_1 (regression)	R^2 / MSE / MAE	0.559 / 0.121 / 0.255
Analytic rank (3-class)	Accuracy / F_1 macro	91.27% / 74.61%
Class 0 F_1		94.35%
Class 1 F_1		97.28%
Class 2 F_1		40.00%
CM status (binary)	Accuracy / F_1 macro	99.96% / 97.05%

4.3 Architectures and Training

We test two architectures: (1) **GCN**: 3-layer graph convolution [8], hidden dimension 128, BatchNorm, mean+max readout. (2) **ChebConv**: 3-layer Chebyshev spectral filter [4], hidden 128, polynomial order $K \in \{3, 5, 7\}$, same readout.

All models use AdamW (lr $= 10^{-3}$, weight decay 10^{-4}), CosineAnnealingLR, early stopping (patience 15), batch size 256, 50 max epochs. Regression uses MSELoss; classification uses CrossEntropyLoss. We use a stratified 80/10/10 train/val/test split (stratified by analytic rank). We predict three targets: (i) z_1, the imaginary part of the first L-function zero (regression); (ii) analytic rank (3-class: 0, 1, ≥ 2); (iii) CM status (binary).

5 Experiments

5.1 Dataset

We draw data from the LMFDB [12]. The traces matrix contains 53,779 newforms with 1000 Fourier coefficients each. The zeros table has 63,844 entries; inner-joining on the newform label yields 46,347 unique newforms with both coefficient and zero data. Analytic rank, CM status, and scalar targets come from the weight-2 ML table. The final split: train = 37,076, val = 4,634, test = 4,637. Node features and edge lists are constructed as described in Section 4.

5.2 Phase 1: GCN Baseline

The GCN already shows trace-index graphs are viable representations. The R^2 of 0.559 for z_1 is a meaningful positive result, and near-perfect CM de-

Table 2: ChebConv polynomial order sweep on z_1 prediction (stratified split).

K	R^2	MSE	MAE	Best Epoch	Time (s)
3	0.607	0.108	0.236	39	2,239
5	**0.625**	**0.103**	0.228	30	3,863
7	0.623	0.104	**0.227**	31	5,743

Table 3: GCN vs. ChebConv K=5 on all targets (stratified split). Largest gains appear on rare-class classification.

Target	Metric	GCN	ChebConv K=5	Δ
z_1	R^2	0.559	0.625	+0.066
Rank	Accuracy	91.27%	94.16%	+2.89 pp
Rank	F_1 macro	74.61%	89.22%	+14.61 pp
Rank	Class 2 F_1	40.00%	78.87%	+38.87 pp
CM	Accuracy	99.96%	100.00%	+0.04 pp

tection confirms the graph structure captures the CM/non-CM distinction. The weakness is class-2 F_1 (40.00%), reflecting severe class imbalance (rank ≥ 2 forms are rare, $\sim$1.3% of the dataset).

5.3 Phase 2a: ChebConv Spectral Filter Sweep

Spectral filters improve over GCN ($R^2 = 0.559$) for all K. The best ($K = 5$) achieves $R^2 = 0.625$ (+0.066). The improvement from $K = 3$ to $K = 5$ (+0.018) is substantial, while $K = 7$ yields diminishing returns at 50% higher computational cost. Five-hop spectral receptive fields capture most useful graph-scale information in 1000-node trace-index graphs. Early stopping at epochs 30–39 indicates good convergence without overfitting.

5.4 Phase 2b: All Targets, ChebConv $K = 5$

The most striking result is the rare-class improvement: class-2 F_1 nearly doubles from 40.00% to 78.87%. This suggests spectral filters help distinguish subtle trace patterns that differentiate rank-≥ 2 from rank-1 forms — patterns invisible to first-order GCN aggregation. CM detection, already near-perfect with GCN, leaves little room for improvement.

Table 4: Trace-index (Paradigm A) vs. multiplicative graph (Paradigm C) using GCN (20 epochs).

Target	Metric	Trace-Index	Multiplicative
z_1	R^2	0.559	0.345
Rank	Accuracy	91.27%	86.00%
CM	Accuracy	99.96%	99.60%

Table 5: Stratified vs. cross-level generalization (ChebConv K=5).

Target	Metric	Stratified	Cross-Level	Δ
z_1	R^2	0.625	0.538	-0.087 (-14%)
z_1	MSE	0.103	0.092	-0.011
Rank	Accuracy	94.16%	87.58%	-6.58 pp
Rank	F_1 macro	89.22%	67.53%	-21.69 pp
Rank	Class 2 F_1	78.87%	25.66%	-53.21 pp
CM	Accuracy	100.00%	99.96%	negligible
CM	F_1 macro	100.00%	96.42%	-3.58 pp

5.5 Phase 3: Multiplicative Graph Comparison

The multiplicative graph underperforms on every metric. Two factors explain this: (i) fewer nodes (608 vs. 1000) and features (3 vs. 5 dimensions), and (ii) data-independent edges (the divisibility poset is identical for every newform, so topology carries no information about f). The kNN edges in trace-index graphs, by contrast, depend on actual trace values, making the topology itself informative.

5.6 Phase 4: Cross-Level Generalization

We test generalization by splitting on conductor level: train on conductors $\leq$ 3000 (25,026 forms), validate on 3001–4000 (10,556), test on $>$ 4000 (10,765). Table 5 compares against the stratified split using ChebConv $K = 5$.

Regression generalizes well: z_1 R^2 drops only 14% (0.625 $\rightarrow$ 0.538), and MSE actually improves from 0.103 to 0.092 (the cross-level test set has a different z_1 distribution with smaller absolute errors). Classification degrades more substantially: rank accuracy drops 6.58 pp, macro F_1 drops 21.69 pp, and class-2 F_1 collapses from 78.87% to 25.66%. This is a distribution shift effect: large-conductor newforms have a different rank distribution, and the

Table 6: Structural comparison: Cayley graphs (from [14]) vs. trace-index graphs (this work).

Property	Cayley $\mathrm{SL}(2, \mathbb{F}_p)$	Trace-Index
Vertex-transitive	Yes	No
Node features	Identical (structural only)	Unique (Fourier coefficients)
Edge topology	Algebraic (group-generated)	Data-driven (kNN on trace values)
Node count	6 to 1,010,100 (varies by p)	1,000 (fixed)
Training samples	18 primes	46,347 newforms
Best test R^2	< 0 (all tracks)	0.625 (zeros, ChebConv $K = 5$)
Approach succeeds?	No	Yes

rare class becomes even rarer in the test set. CM detection is essentially unchanged (99.96% $\rightarrow$ 99.96% accuracy), confirming that CM manifests consistently in Hecke traces regardless of conductor level.

6 Analysis

6.1 Why Trace-Index Graphs Work

The failure of GNNs on Cayley graphs (Section 3) and their success on trace-index graphs trace to three structural differences, summarized in Table 6.

First, **heterogeneous node features**: each node carries a unique Fourier coefficient $a_i(f)$ that varies across indices and newforms. The GNN can extract information from the distribution of coefficients in a node's neighborhood — a form with large $|a_p|$ values at many primes has a different local structure than one with small values.

Second, **data-dependent topology**: the kNN edges connect indices with similar normalized coefficient values. CM forms have a different coefficient distribution (governed by Dirichlet characters) than non-CM forms, yielding visibly different kNN topologies. This makes edge structure itself informative about arithmetic properties.

Third, **fixed moderate size**: 1000 nodes is large enough to contain meaningful structure but small enough that 3-layer message passing propagates information across the graph. The fixed size also means all training samples have the same graph size, eliminating one source of variance.

Fourth, **large training set**: 46,347 newforms is orders of magnitude more than the 18 Cayley graph data points, providing the statistical support needed for deep learning.

6.2 Why Spectral Filters Help

ChebConv outperforms GCN across all targets, with the largest gains on rare-class detection (+38.87 pp in class-2 F_1). The polynomial order K controls the spectral receptive field: GCN ($K = 1$) aggregates only from immediate neighbors, while ChebConv $K = 5$ reaches a 5-hop neighborhood covering a substantial fraction of the 1000-node graph. This allows the model to detect correlations between coefficients at non-adjacent indices — such as Sato–Tate distribution patterns or multiplicativity relations $a_{mn}(f) = a_m(f)a_n(f)$ for coprime m, n.

The diminishing returns from $K = 5$ to $K = 7$ (-0.002 in R^2) suggest most useful information is already captured within 5 hops. This is consistent with the graph diameter of trace-index graphs: with sequential edges, the longest shortest path between two nodes is at most 999, but kNN edges create shortcuts, and 5 hops covers a large portion of the graph's diameter.

6.3 Comparison with Flat-Vector Baselines

Experiment 11 of [14] trained sklearn models (RandomForest, GradientBoosting, MLP) on the first 100 Fourier coefficients as flat feature vectors, achieving z_1 R^2 between 0.73 (RandomForest, 100 features) and 0.96 (RandomForest, 100 features plus scalar metadata). Our GNN at $R^2 = 0.625$ is lower, which is expected: sklearn operates directly on the raw coefficient vector, while the GNN must learn from local neighborhoods in a graph-structured representation that necessarily discards some global information.

The GNN's advantage is its ability to learn from *structural priors*: kNN edges capture coefficient similarity, prime edges capture primality structure, and sequential edges capture index ordering. These structural priors could complement raw-coefficient methods in an ensemble. The GNN framework also extends naturally to settings where heterogeneous per-index features (Fourier coefficients at composite indices, p-adic valuations, Galois conjugacy data) must be integrated.

6.4 Design Principle for Mathematical GNN Applications

Our results suggest a design principle for applying GNNs in mathematical domains: the graph construction should reflect the mathematical structure of the *data*, not the algebraic structure of the *underlying objects*. In our case, the underlying objects (modular forms) are defined through group representations and Hecke algebras, but the data available for ML (Fourier coefficients) is naturally organized as a sequence. By constructing graphs

from the coefficient data rather than from the group structure, we give the GNN access to the information that actually varies across forms.

This principle may generalize: when applying GNNs to number-theoretic objects, the graph should encode how the available numerical data relates — via multiplicativity, similarity, or other mathematical relations — rather than how the underlying algebraic objects relate.

7 Related Work

7.1 The LMFDB [12]

is a comprehensive database of modular forms, L-functions, and related objects. Our 46,347 newforms are drawn entirely from LMFDB tables.

7.2 ML for number theory

Weiss [14] showed GNNs fail on vertex-transitive algebraic graphs. The theoretical bridge from graph spectra to Hecke eigenvalues was established by Lubotzky et al. [9] and Pizer [11]. The Modularity Theorem [2, 15] guarantees every elliptic curve over $\mathbb{Q}$ corresponds to a weight-2 newform. Earlier work at smaller scales [3] explored machine learning on LMFDB data for BSD-related predictions.

7.3 GNN expressiveness

Message-passing GNNs are bounded by the 1-WL test [16]. Chebyshev spectral filters [4] extend the receptive field in the spectral domain. Graph attention [13] and transformer [7] architectures offer alternatives for capturing long-range dependencies.

7.4 The BSD conjecture [1]

predicts the analytic rank of $L(E, s)$ equals the Mordell–Weil rank of E. Our rank prediction task addresses a quantity at the heart of this conjecture.

8 Conclusion

We have shown that GNNs can predict arithmetic properties of modular forms when the graph representation is constructed from Fourier coefficient

data. Trace-index graphs (1000 nodes, three edge types, 5-dimensional features) provide the local structural diversity that GNNs need to succeed, avoiding the vertex-transitivity failure mode identified in prior work on Cayley graphs.

On 46,347 weight-2 newforms, ChebConv $K = 5$ predicts the first L-function zero ($R^2 = 0.625$), analytic rank (94.16% accuracy, macro $F_1 = 89.22\%$), and CM status (100%; near-perfect on all splits). Spectral filters consistently outperform plain GCN, with the largest gains on rare-class detection (+38.87 pp in class-2 F_1). Cross-level generalization reveals an asymmetry: regression transfers reasonably well ($R^2 = 0.538$, 14% degradation), while classification — particularly rare-class detection — degrades substantially due to distribution shift.

The contrast with the Cayley graph failure is instructive: the same GNN architectures that are powerless on algebraically uniform graphs become effective when the graph structure encodes arithmetic data with genuine local diversity. The design principle is clear: for mathematical GNN applications, construct graphs from the data, not from the underlying algebraic objects.

Future work should explore ensembles combining GNN graph-structure predictions with raw-coefficient baselines, extend the trace-index paradigm to higher-weight modular forms, and investigate whether graph representations can surface new conjectural relationships between Fourier coefficients and L-function properties.

References

[1] B. J. Birch and H. P. F. Swinnerton-Dyer. Notes on elliptic curves. II. *Journal für die reine und angewandte Mathematik*, 218:79–108, 1965.

[2] Christophe Breuil, Brian Conrad, Fred Diamond, and Richard Taylor. On the modularity of elliptic curves over $\mathbf{Q}$: wild 3-adic exercises. *Journal of the American Mathematical Society*, 14(4):843–939, 2001.

[3] John A. Brooke, Thomas Crisford, and David Russell. A Machine Learning Attack on BSD. *arXiv preprint arXiv:2009.00920*, 2020.

[4] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[5] Pierre Deligne. La conjecture de Weil, I. *Publications Mathématiques de l'Institut des Hautes Études Scientifiques*, 43:273–307, 1974.

[6] Fred Diamond and Jerry Shurman. *A First Course in Modular Forms*, volume 228 of *Graduate Texts in Mathematics*. Springer, 2005.

[7] Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. *arXiv preprint arXiv:2012.09699*, 2022.

[8] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[9] Alexander Lubotzky, Ralph Phillips, and Peter Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[10] James S. Milne. *Modular Functions and Modular Forms*. 2020. Online lecture notes, available at https://www.jmilne.org/math/.

[11] Arnold K. Pizer. Ramanujan graphs and hecke operators. *Bulletin of the American Mathematical Society*, 23(1):127–137, 1990.

[12] The LMFDB Collaboration. The L-functions and modular forms database. *Results in Mathematics*, 78:1–43, 2023. Online database: https://www.lmfdb.org.

[13] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[14] Tobias Weiss. Why graph neural networks fail to predict spectral properties of vertex-transitive algebraic graphs. Unpublished manuscript, 2025.

[15] Andrew Wiles. Modular elliptic curves and Fermat's last theorem. *Annals of Mathematics*, 141(3):443–551, 1995.

[16] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.