# Efficient and Expressive Graph Neural Networks

**Monika Varshney**[1] **Tanima Dutta**[1]
[1]Department of Computer Science and Engineering,
Indian Institute of Technology (Banaras Hindu University)
{monikavarshney.rs.cse21, tanima.cse}@iitbhu.ac.in

## Abstract

Graph neural networks (GNNs) have achieved remarkable success but remain limited in distinguishing non-isomorphic graphs with similar local structures, due to their reliance on neighborhood aggregation. Higher-order or subgraph-counting GNNs offer greater expressivity but at prohibitive computational cost. We introduce *Polynomial-time Cycle basis-GNNs* (PCB-GNNs), a topology-aware architecture that augments message passing with polynomial-time cycle-basis features, capturing essential global invariants overlooked by standard MPNNs. PCB-GNNs distinguishes challenging graph families where 1-WL and typical GNNs fail, while incurring only polynomial overhead. Experiments across synthetic, molecular, and protein benchmarks show that PCB-GNNs consistently outperforms state-of-the-art models on both expressiveness and large-scale tasks. PCB-GNNs achieve classification accuracy on MUTAG (98.53%), PTC (86.48%), PROTEINS (82.21%), and NCI1 (88.37%) while scaling effectively to larger datasets such as IMDB-B. On the ZINC-Subset molecular regression task, our model attains MAE of 0.054.

## 1 Introduction

Graph neural networks (GNNs) have become a cornerstone of machine learning on structured data, with wide-ranging applications in social network analysis, recommender systems, and molecular biology Kipf and Welling [2016], Xu et al. [2019], Sharma et al. [2024], Wu et al. [2022a,b]. Among the many variants, message passing neural networks (MPNNs) Scarselli et al. [2009] are the most widely adopted. By iteratively aggregating information from local neighborhoods, they learn expressive graph representations that perform well on a variety of benchmarks.

However, MPNNs face a fundamental limitation: they are at most as powerful as the 1-dimensional Weisfeiler–Lehman (1-WL) test for graph isomorphism Xu et al. [2019]. The WL test refines node labels by hashing multisets of neighbors, which allows it to separate many non-isomorphic graphs, but it fails on entire families such as regular graphs, biconnected graphs, and strongly regular graphs(see Figure 1) Cai et al. [1992], Morris et al. [2019]. Standard MPNNs inherit this weakness because their aggregation is purely local.

A natural solution is to move to higher-order architectures. Indeed, $k$-GNNs Maron et al. [2019a] and subgraph-based methods Bevilacqua et al. [2022], Zhao et al. [2022a] achieve the expressivity of higher-order WL tests. While powerful, these methods suffer from exponential growth in computation and memory as $k$ increases, making them impractical on large graphs Li et al. [2023], Huang et al. [2023].

**Our perspective differs from this typical WL hierarchy race.** Instead of competing with WL hierarchy, we propose a complementary axis of expressivity: *topology-aware features* that capture global invariants invisible to purely local aggregation. In particular, we focus on the *cycle basis* of a graph—a minimal set of independent cycles that spans the entire cycle space and can be computed

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: New Perspectives in Graph Machine Learning.

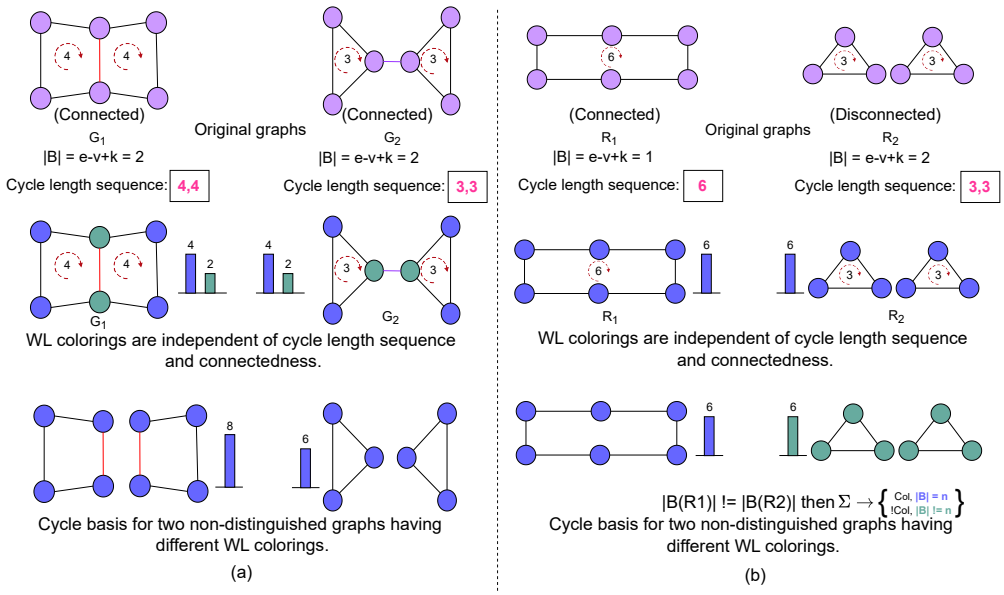


Figure 1: Illustration showing the importance of cycle length in case of non-isomorphic graphs (a) with biconnectivity and (b) regular graphs. It is shown here that due to identical degree sequences WL test fails to distinguish these graphs but in both cases, the graphs possess cycles of different length irrespective of degree sequences and can be utilised to distinguish the graphs. In biconected graph, the edge corresponding to bridge is shown red, which is responsible for different cycle lengths.

in polynomial time Harary [1969]. Cycles are fundamental structures across domains: rings in molecules Duvenaud et al. [2015], Gilmer et al. [2017], motifs in social networks Jiang et al. [2022], and communities in communication graphs. Building on these observations, we propose *Polynomial-time Cycle Basis GNNs* (PCB-GNNs), a topology-aware framework that augments message passing with efficiently computable cycle-basis features, thereby bridging the gap between lightweight MPNNs and costly higher-order models.

**Contributions.**

- We introduce *PCB-GNNs*, a novel architecture that integrates cycle-basis features into message passing through a competitive gating mechanism, capturing global topological signals at $O(n^3)$ cost.
- We formalize Polynomial-time *CycleBasis-WL*, a cycle-aware variant of the Weisfeiler–Lehman test, and show it can distinguish families of graphs (e.g., strongly regular graphs) that 1-WL cannot.
- We provide an efficient polynomial-time algorithm for extracting cycle bases, ensuring scalability while avoiding the exponential complexity of higher-order WL and subgraph-based GNNs.
- We validate PCB-GNNs on both synthetic expressiveness benchmarks and real-world datasets, demonstrating consistent improvements over standard MPNNs while maintaining computational efficiency.

## 2 Preliminaries

We consider undirected, simple graphs $G = (V, E)$ with $n = |V|$ vertices and $m = |E|$ edges. The neighborhood of $v \in V$ is $\mathcal{N}(v) = \{u : \{u, v\} \in E\}$ and $\deg(v) = |\mathcal{N}(v)|$. A *path* of length $k$ is a sequence $(v_1, \ldots, v_{k+1})$ of distinct vertices with $\{v_i, v_{i+1}\} \in E$. A *cycle* is a closed path $(v_1, \ldots, v_k, v_1)$ with $k \geq 3$ and $v_i \neq v_j$ for $1 \leq i < j \leq k$ except $v_{k+1} = v_1$.

**Definition 1** (Cycle Space and Cycle Basis)**.** *The* cycle space $\mathcal{C}(G) \subseteq 2^E$ *is the set of edge-sets expressible (mod 2) as sums of simple cycles. A* cycle basis $\mathcal{B} = \{C_1, \ldots, C_b\}$ *is a minimal set of simple cycles whose symmetric-difference span equals* $\mathcal{C}(G)$*. Every cycle can be written as a unique sum of basis cycles over* $\mathbb{F}_2$*.*

**Lemma 2.** *If* $G$ *has* $n$ *vertices,* $m$ *edges, and* $k$ *connected components, then any cycle basis has size* $b = m - n + k$*. A cycle basis can be computed in* $\mathcal{O}(n^3)$ *time via Gaussian elimination or*

*spanning tree methods Harary [1969], and a minimum (lexicographically canonical) cycle basis can be obtained using Horton's polynomial-time algorithm Horton [1987].*

**Message Passing Neural Networks.** An MPNN iteratively updates node features $h_v^{(t)}$ by aggregating messages $m_v^{(t)} = \sum_{u \in \mathcal{N}(v)} \psi(h_v^{(t)}, h_u^{(t)}, e_{uv})$ and updating via $h_v^{(t+1)} = \phi(h_v^{(t)}, m_v^{(t)})$, where $\psi$ and $\phi$ are learnable, permutation-invariant functions.

**Weisfeiler-Lehman (1-WL) Test.** The 1-WL test assigns node colors $c_v^{(0)}$ (often $\deg(v)$), and iteratively updates:

$$c_v^{(t+1)} = \mathrm{Hash}\Big(c_v^{(t)}, \{\{c_u^{(t)} : u \in \mathcal{N}(v)\}\}\Big).$$

If two graphs yield identical multisets after convergence, 1-WL cannot distinguish them.

## 3 Related Work

*Expressiveness and WL Hierarchy.* The expressive power of standard MPNNs is bounded by the 1-WL test Xu et al. [2019], which fails on numerous non-isomorphic graph pairs Cai et al. [1992]. To bridge this gap, higher-order GNNs and $k$-WL tests have been proposed Maron et al. [2019b], matching $k$-WL expressiveness at the cost of $O(n^k)$ time and memory.
*Substructure and Cycle-Counting GNNs.* Augmenting GNNs with explicit substructure counts improves discrimination. Chen et al. [2020] showed GNNs can count small motifs via neighborhood sampling, while Morris et al. [2020a] use subgraph counts to emulate higher-order WL. Cycle-count GNNs Xu et al. [2019], Zhang et al. [2023] enumerate all simple cycles, but exact cycle enumeration is #P-complete and impractical for large graphs. Kanatsoulis and Ribeiro [2024] have explored the expressive capacity of GNNs to count substructures using constructive message-passing designs. Ji et al. [2025] combine persistent homology with spectral information—such as SpectRe to enhance expressivity via topological descriptors. Theoretical work on subgraph GNNs shows that $k$-hop aggregation suffices to approximate any function on graphs with bounded cycles Chen et al. [2025]. Cycle bases have a long history in graph theory Harary [1969], yet their use in scalable GNN architectures is largely unexplored. Prior motif- and cycle-aware GNNs Jiang et al. [2022], Zhang et al. [2023] focus on raw counts or subgraph enumeration, while our approach systematically integrates the minimal cycle basis, computable in $\mathcal{O}(n^3)$ time as a compact and informative global topological descriptor.

## 4 Polynomial-time Cycle Basis-GNNs

Local aggregation in MPNNs, being equivalent to 1-WL, cannot capture global cyclic invariants. Strongly regular graphs (SRGs) illustrate this limitation: for example, the $4 \times 4$ Rook's graph $R(4,4)$ and the Shrikhande graph $S(16,6,2,2)$ both have 16 vertices, degree 6, and identical neighborhood statistics, making them indistinguishable by 1-WL and standard MPNNs Harary [1969]. Yet their cycle bases differ: $R(4,4)$ contains only 4- and 6-cycles, while $S(16,6,2,2)$ also has 3-cycles. This example highlights how cycle-basis features provide a compact and efficient signal for distinguishing graphs that defeat purely local methods. To exploit this insight, we introduce ***PCB-GNNs***, which augments message passing with cycle-basis features. Rather than attempting to enumerate all cycles which is computationally intractable, cycle bases can be computed in $\mathcal{O}(|V|^3)$ (Lemma 2).

### 4.1 PCB-GNNs Architecture

We first determine the cycle basis $\mathcal{B}$ of each graph $G = (V, E)$. To remove ambiguity due to non-unique cycle bases, we compute a canonical minimum cycle basis (MCB) using lexicographically ordered edges with deterministic tie–breaking Harary [1969], Horton [1987]. Let $\mathcal{B} = \{C_1, \ldots, C_b\}$ denote the resulting canonical MCB of $G$. For each node $v \in V$, we define a *cycle-basis feature vector*

$$z_v = \big[f_\ell(v)\big]_{\ell=3}^{L_{\max}} \in \mathbb{R}^{L_{\max}-2},$$

where $f_\ell(v)$ counts the number of basis cycles of length $\ell$ incident to node $v$, and $L_{\max}$ is the maximum basis-cycle length in $G$. These statistics depend only on cycle-length distributions and node–cycle incidences, which remain stable across all valid bases that span the same cycle space. Hence, the features are effectively basis-invariant and reproducible across runs. At each layer $t$,

PCB-GNNs updates node states via two channels: (i) standard neighborhood aggregation, and (ii) cycle-basis transformation, processed by a learnable MLP $g$:

$$m_v^{(t)} = \sum_{u \in \mathcal{N}(v)} \psi\big(h_v^{(t)}, h_u^{(t)}, e_{uv}\big), \tag{1}$$

$$c_v^{(t)} = g(z_v), \tag{2}$$

$$h_v^{(t+1)} = \phi\big(h_v^{(t)}, m_v^{(t)}, \sigma(W_c c_v^{(t)})\big), \tag{3}$$

where $\psi$ and $\phi$ are learnable functions (e.g., MLPs), $W_c$ is a trainable matrix, and $\sigma$ is a nonlinearity. When no edge features are present, $e_{uv}$ is omitted. To balance local and global information, we employ a competitive gating mechanism:

$$h_v^{(t+1)} = \alpha_v^{(t)} \cdot \phi\big(h_v^{(t)}, m_v^{(t)}\big) + (1 - \alpha_v^{(t)}) \cdot \sigma(W_c c_v^{(t)}),$$

where the gate $\alpha_v^{(t)} = \sigma(w^\top [h_v^{(t)} \| c_v^{(t)}])$ is dynamically learned from the concatenation of local and cycle-basis features. This convex combination enables the network to adaptively balance local and global topological signals. After $L$ layers, a permutation-invariant readout $R$ (e.g., sum/mean) produces the final graph embedding:

$$h_G = R\big(\{h_v^{(L)} : v \in V\}\big).$$

### 4.2 Polynomial-time CycleBasis-WL

To analyze the expressive power of our model, we introduce a cycle-aware variant of the Weisfeiler–Lehman test.

**Definition 3** (CycleBasis-WL)**.** *Let $G = (V, E)$ be a graph with cycle basis $\mathcal{B}$. In* CycleBasis-WL*, each node $v \in V$ is initialized with*

$$c_v^{(0)} = \big(\deg(v),\ \{\,|C| : C \in \mathcal{B},\ v \in C\,\}\big),$$

*i.e., its degree and the multiset of lengths of incident basis cycles. Colors are then refined as in 1-WL:*

$$c_v^{(t+1)} = \mathrm{Hash}\big(c_v^{(t)}, \{\{c_u^{(t)} : u \in \mathcal{N}(v)\}\}\big).$$

This construction retains the computational efficiency of WL while incorporating global topological signals via the cycle basis. For example, CycleBasis-WL can separate the $4 \times 4$ Rook's graph and the Shrikhande graph, which 1-WL (and hence standard MPNNs) cannot, due to their differing basis-cycle length distributions.

## 5 Experiments

We conduct our experiments on a diverse set of ten publicly available real-world graph datasets of different scales. The small-scale datasets are sourced from the TUDataset Morris et al. [2020b]. This collection encompasses noteworthy datasets such as MUTAG, PTC, PROTEINS, NCI1, and IMDB-B, originating from the realms of biology, chemistry, and social networks. The second category of datasets comprises of large scale molecular benchmarks that are ZINC (12K graphs) Dwivedi et al. [2023] and ZINC-FULL (250k molecular graphs) taken from the ZINC database Sterling and Irwin [2015] and OGBG-PCBA (437k graphs) taken from Open Graph Benchmark Hu et al. [2020]. Further details of the datasets are provided in Table 1. To ensure the reliability of results, We repeated the experiments 15 times for the expressiveness datasets and three times for the large-scale datasets. This repetition helps us calculate the average values and the standard deviation. For the TUDataset, we use 10-fold cross-validation to calculate the average and standard deviation of validation accuracy. We conducted a hyperparameter study to determine the optimal configuration for PCB-GNNs. We varied several key parameters, including the number of layers, embedding length, learning rate, and epochs. We experimented with a range of layers from 2 to 20. Based on experiments, the best set of hyperparameters are a number of layers as 6, embedding length as 64, learning rate as 0.00025, and epochs as 200.

**Baselines.** We evaluate PCB-GNNs against a wide spectrum of state-of-the-art GNNs. On small-scale TU datasets, we include strong expressive baselines: DGCNN Zhang et al. [2018], IGN Maron et al.

Table 1: Dataset statistics. $\bar{v}$, $\bar{e}$ = average nodes and edges.

| Dataset | #Graphs | $\bar{v}$ | $\bar{e}$ | Task |
|---|---|---|---|---|
| MUTAG | 188 | 17.9 | 19.8 | Graph classification |
| PTC | 351 | 14.6 | 15.0 | Graph classification |
| PROTEINS | 1,113 | 39.1 | 72.8 | Graph classification |
| NCI1 | 4,110 | 29.9 | 32.3 | Graph classification |
| IMDB-B | 1,000 | 19.8 | 96.5 | Graph classification |
| ENZYMES | 600 | 32.6 | 62.1 | Graph classification |
| ZINC-Subset | 12,000 | 23.2 | 24.9 | Graph regression |
| ZINC-Full | 249,456 | 23.1 | 24.9 | Graph regression |
| QM9 | 133,885 | 18.0 | 19.4 | Graph regression |
| ogbg-molpcba | 437,929 | 26.0 | 28.1 | Graph classification |
| ogbn-proteins | 1 | 132,534 | 39M | Node classification |

[2019b], GIN Xu et al. [2019], GSN Bouritsas et al. [2022], CIN Bodnar et al. [2021], GIN-AK+ Zhao et al. [2022b], ESAN-GIN Bevilacqua et al. [2022], and SAGIN Zeng et al. [2023]. On expressiveness benchmarks (EXP, SR25) and large-scale molecular datasets (ZINC, ZINC-FULL, MolPCBA), we compare to both classical and advanced baselines: GIN, PNA Zhang et al. [2018], PNA-AK+ Maron et al. [2019b], GSN, CIN, GIN-AK+, ESAN-GIN, SAGIN, GD-WL Zhang et al. [2023], and $I^2$-GNN Huang et al. [2023]. We have conducted the experiments on the Google collaboratory platform with an Intel Xeon CPU @2.20 GHz (GeForce RTX 2080Ti) with 13GB RAM. The configurations of our environments and packages include CUDA 11.1, PYTHON 3.9, PyTorch 1.9.1, and PyTorch Geometric 2.0.1.

## 6 Results and Discussions

In this section, we validate the effectiveness of the PCB-GNNs method by assessing its performance across diverse datasets encompassing small, large, and expressiveness datasets.

**Performance of PCB-GNNs on small scale datasets.** To assess the effectiveness of our proposed PCB-GNNs on small-scale datasets, we conducted a series of experiments comparing its performance against state-of-the-art graph representation methods. Table 2 outlines a comparative study between the PCB-GNNs approach and several baseline methods on TUdataset. In the evaluation across multiple datasets, our method demonstrated superior performance compared to baseline methods in MUTAG, PTC, PROTEINS and NCI1 with 98.53%, 86.48%, 82.21%, and 88.37% accuracies. This superior performance suggests that PCB-GNNs are more effective in capturing relevant patterns within these datasets, resulting in better predictive accuracy.

**Performance of PCB-GNNs on large and expressive datasets.** To evaluate the scalability and expressiveness of our proposed PCB-GNNs, we conducted extensive experiments on large and expressive datasets commonly used in graph representation learning. Results indicate that PCB-GNNs maintain competitive performance on EXP, SR25, and PTC datasets with 100% accuracy. In the case of the ZINC-FULL, PCB-GNNs achieves MAE of 0.011 and 31.39% AP on MolPCBA dataset. These results emphasize the potential of PCB-GNNs in addressing the challenges posed by large and expressive datasets.

**Efficiency Evaluation.** We evaluate our approach by measuring efficiency in terms of parameter

Table 2: Test accuracies for TU Datasets.

| Method | MUTAG | PTC | PROTEINS | NCI1 | IMDB-B |
|---|---|---|---|---|---|
| DGCNN Zhang et al. [2018] | $85.64 \pm 1.45$ | $58.72 \pm 2.57$ | $75.82 \pm 0.56$ | $74.82 \pm 0.83$ | $70.46 \pm 0.85$ |
| IGN Maron et al. [2019b] | $83.26 \pm 13.57$ | $58.37 \pm 6.77$ | $76.34 \pm 5.63$ | $74.32 \pm 2.77$ | $72.44 \pm 5.53$ |
| GIN Xu et al. [2019] | $88.67 \pm 0.56$ | $63.87 \pm 5.44$ | $75.87 \pm 2.76$ | $82.57 \pm 1.53$ | $75.37 \pm 5.41$ |
| GSN Bouritsas et al. [2022] | $92.43 \pm 7.55$ | $65.25 \pm 7.82$ | $76.36 \pm 5.20$ | $83.77 \pm 2.33$ | $77.87 \pm 3.31$ |
| CIN Bodnar et al. [2021] | $92.25 \pm 6.61$ | $68.87 \pm 7.43$ | $77.82 \pm 4.33$ | $83.66 \pm 1.84$ | $76.46 \pm 3.87$ |
| GIN-AK+ Zhao et al. [2022b] | $90.53 \pm 5.50$ | $67.87 \pm 8.34$ | $77.84 \pm 5.03$ | $83.97 \pm 2.74$ | $77.51 \pm 4.46$ |
| ESAN-GIN Bevilacqua et al. [2022] | $90.33 \pm 6.67$ | $69.38 \pm 6.85$ | $77.24 \pm 5.33$ | $83.72 \pm 2.92$ | $\mathbf{77.56 \pm 3.70}$ |
| SAGIN'Zeng et al. [2023] | $94.48 \pm 3.95$ | $72.52 \pm 7.46$ | $79.84 \pm 3.48$ | $85.92 \pm 1.58$ | $75.46 \pm 6.28$ |
| $I^2$-GNNHuang et al. [2023] | $93.66 \pm 2.87$ | $71.37 \pm 3.65$ | $77.42 \pm 1.94$ | $83.97 \pm 2.66$ | $72.64 \pm 4.29$ |
| PCB-GNNs | $\mathbf{98.53 \pm 1.65}$ | $\mathbf{86.48 \pm 1.64}$ | $\mathbf{82.21 \pm 5.63}$ | $\mathbf{88.37 \pm 1.28}$ | $75.44 \pm 2.65$ |

usage and time cost(per epoch) for ZINC-subset dataset according to Dwivedi et al. [2023].

Table 3: Test results for expressiveness datasets and large scale datasets.

| Method | EXP (ACC) | SR25 (ACC) | ZINC-Subset (MAE) | ZINC-FULL (MAE) | MolPCBA (AP) |
|---|---|---|---|---|---|
| GIN Xu et al. [2019] | 50% | 6.67% | $0.164 \pm 0.003$ | $0.087 \pm 0.002$ | $26.83 \pm 0.04$ |
| GSN Bouritsas et al. [2022] | N/A | N/A | $0.107 \pm 0.0180$ | N/A | $27.98 \pm 0.33$ |
| CIN Bodnar et al. [2021] | N/A | N/A | $0.079 \pm 0.002$ | $0.023 \pm 0.004$ | N/A |
| GIN-AK+ Zhao et al. [2022b] | **100%** | 6.77% | $0.080 \pm 0.002$ | N/A | $29.22 \pm 0.46$ |
| ESAN-GIN Bevilacqua et al. [2022] | **100%** | N/A | $0.101 \pm 0.002$ | N/A | N/A |
| PNA Zhang et al. [2018] | 50% | 6.67% | $0.141 \pm 0.006$ | N/A | $27.44 \pm 0.06$ |
| PNA-AK+ Maron et al. [2019b] | **100%** | 6.67% | $0.085 \pm 0.006$ | N/A | $28.54 \pm 0.03$ |
| SAGIN'Zeng et al. [2023] | **100%** | **100%** | $0.072 \pm 0.001$ | $0.016 \pm 0.06$ | $28.55 \pm 0.07$ |
| GD-WLZhang et al. [2023] | **100%** | **100%** | $0.081 \pm 0.009$ | $0.025 \pm 0.04$ | $26.28 \pm 0.01$ |
| $I^2$-GNNHuang et al. [2023] | **100%** | **100%** | $0.083 \pm 0.001$ | $0.023 \pm 0.06$ | $26.01 \pm 0.01$ |
| PCB-GNNs | **100%** | **100%** | $\mathbf{0.054 \pm 0.001}$ | $\mathbf{0.011 \pm 0.02}$ | $\mathbf{31.39 \pm 0.03}$ |

PCB-GNNs strikes a balance between complexity and efficiency. With 493,672 parameters and a training time of 15.48 seconds per epoch as shown in Table 4, it performs comparably to modern transformer-based models while offering improved theoretical properties and possibly better generalization.

### 6.1 Expressivity and Complexity

**Expressivity.** CycleBasis-WL extends 1-WL by initializing nodes with incident cycle-basis features. This allows it to separate families of graphs indistinguishable by 1-WL, such as the $4 \times 4$ Rook's graph and the Shrikhande graph, which share degree distributions but differ in cycle-basis structure. Thus, PCB-GNNs inherits strictly greater power than 1-WL on such families. Importantly, this approach is *complementary* to higher-order WL tests: short motifs may be better captured by 3-WL, while global cycle statistics are efficiently captured by basis features.

**Expressivity conditions.** PCB-GNNs distinguish any pair of graphs that are 1-WL–indistinguishable yet differ in the multiset of incident basis-cycle lengths at one or more nodes. Specifically, if there exists a node $v$ such that $\{|C| : C \in \mathcal{B}(G_1), v \in C\} \neq \{|C| : C \in \mathcal{B}(G_2), v \in C\}$, then CycleBasis-WL assigns distinct node colors and separates $G_1$ and $G_2$. Conversely, when two non-isomorphic graphs have identical degree distributions and identical node-level cycle-basis statistics, their CycleBasis-WL color refinements coincide. These limiting cases include (i) *acyclic* graphs, where $\mathcal{B} = \varnothing$ and PCB-GNN reduces to its MPNN backbone, (ii) *cycle-regular* graphs in which all nodes share identical basis-cycle incidences, and (iii) *cospectral* graph pairs with identical cycle-basis statistics. In these settings, the expressivity of PCB-GNN collapses to that of 1-WL.

**Complexity.** A cycle basis of size $|\mathcal{B}| = e - v + k$ can be computed in $\mathcal{O}(n^3)$ time using standard polynomial-time algorithms Harary [1969]. Once precomputed, forming per-node cycle-basis features is linear in $|\mathcal{B}|$, and each PCB-GNNs layer costs $\mathcal{O}(ed)$ (message passing) $+ \mathcal{O}(vd)$ (gating), matching standard MPNNs up to a negligible additive term. Hence, the only overhead is a one-time preprocessing step.

Table 4: Efficiency of GNN models.

| Model | #Params | Time (s) |
|---|---|---|
| GCN | 516,153 | 5.92 |
| GraphSAGE | 503,396 | 6.25 |
| GIN | 510,448 | 8.57 |
| GAT | 542,639 | 8.83 |
| SAGPool | 529,544 | 26.54 |
| 3WLGNN | 507,543 | 179.77 |
| Graphormer | 489,638 | 14.66 |
| Graphormer-GD | 502,884 | 15.31 |
| $I^2$-GNNs | 511,684 | 20.73 |
| PCB-GNNs | 493,672 | 15.48 |

**Limitations.** While PCB-GNN achieves higher expressivity than 1-WL, its gains are most pronounced on graphs where global cyclic structure is informative. In extremely dense graphs, the cycle-basis preprocessing step may become memory-intensive, though it remains polynomial-time and performed only once.

## 7 Conclusion

We introduced Polynomial-time Cycle Basis-GNNs, a topology-aware architecture that augments message passing with efficiently computable cycle-basis features. Our analysis shows that CycleBasis-WL strictly extends 1-WL on families of graphs with distinct basis-cycle statistics, while remaining complementary to higher-order refinements. Empirically, PCB-GNNs improves over strong MPNN baselines on synthetic, molecular, and biological datasets, with negligible overhead and increased robustness to perturbations.

## References


Thomas N. Kipf and Max Welling. Variational Graph Auto-Encoders. *arXiv1611.07308*, 2016.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *Proceedings of the ICLR*, 2019.

Kartik Sharma, Yeon-Chang Lee, Sivagami Nambi, Aditya Salian, Shlok Shah, Sang-Wook Kim, and Srijan Kumar. A survey of graph neural networks for social recommender systems. *ACM Comput. Surv.*, 2024.

Lingfei Wu, Peng Cui, Jian Pei, Liang Zhao, and Xiaojie Guo. Graph neural networks: Foundation, frontiers and applications. In *Proceedings of the ACM SIGKDD*, page 4840–4841, 2022a.

Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: A survey. *ACM Comput. Surv.*, 55(5), 2022b.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

Jin Yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identification. *Springer Combinatorica*, 12(4):389–410, 1992.

Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI*, volume 33, pages 4602–4609, 2019.

Haggai Maron, Heli Ben-Hamu, Hadar Serviansky, and Yaron Lipman. Provably powerful graph networks. In *Proceedings of the NeurIPS*, volume 32, pages 2156–2167, 2019a.

Beatrice Bevilacqua, Fabrizio Frasca, Derek Lim, Balasubramaniam Srinivasan, Chen Cai, Gopinath Balamurugan, Michael M. Bronstein, and Haggai Maron. Equivariant Subgraph Aggregation Networks. In *Proceedings of the ICLR*, 2022.

Lingxiao Zhao, Wei Jin, Leman Akoglu, and Neil Shah. From stars to subgraphs: Uplifting any gnn with local structure awareness. In *Proceedings of the ICLR*, 2022a.

J. Li, H. Peng, Y. Cao, Y. Dou, H. Zhang, P. S. Yu, and L. He. Higher-Order Attribute-Enhancing Heterogeneous Graph Neural Networks. *IEEE Transactions on Knowledge &; Data Engineering*, 35(01):560–574, 2023.

Yinan Huang, Xingang Peng, Jianzhu Ma, and Muhan Zhang. Boosting the cycle counting power of graph neural networks with $I^2$-GNNs. In *Proceeding of the ICLR*, 2023.

Frank Harary. *Graph Theory*. CRC Press, Reading, Massachusetts, 1st edition, 1969. doi: 10.1201/9780429493768. On Demand Printing Of 02787.

David K. Duvenaud, David Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gomez-Bombarelli, Timothy Hirzel, Alan Aspuru-Guzik, and Ryan P. Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in Neural Information Processing Systems*, volume 28, pages 2224–2232, 2015.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the ICML*, pages 1263–1272, 2017.

Y. Jiang, Y. Cheng, H. Zhao, W. Zhang, X. Miao, Y. He, L. Wang, Z. Yang, and B. Cui. Zoomer: Boosting Retrieval on Web-scale Graphs by Regions of Interest. In *Proceedings of the ICDE*, pages 2224–2236, 2022.

Joseph Douglas Horton. A polynomial-time algorithm to find the shortest cycle basis of a graph. *SIAM Journal on Computing*, 16(2):358–366, 1987.

Haggai Maron, Heli Ben-Hamu, Hadar Serviansky, and Yaron Lipman. Provably powerful graph networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Proceedings of the NeurIPS*, volume 32, 2019b.

Zhengdao Chen, Lei Chen, Soledad Villar, and Joan Bruna. Can graph neural networks count substructures? In *Proceedings of the NeurIPS*, pages 10383–10395, 2020.

Christopher Morris, Gaurav Rattan, and Petra Mutzel. Weisfeiler and leman go sparse: towards scalable higher-order graph embeddings. In *Proceedings of the NeurIPS*, pages 21824–21840, 2020a.

Bohang Zhang, Shengjie Luo, Liwei Wang, and Di He. Rethinking the expressive power of GNNs via graph biconnectivity. In *Proceeding of the ICLR*, 2023.

Charilaos Kanatsoulis and Alejandro Ribeiro. Counting graph substructures with graph neural networks. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=qaJxPhkYtD.

Mattie Ji, Amauri H. Souza, and Vikas Garg. Graph persistence goes spectral, 2025. URL https://arxiv.org/abs/2506.06571.

Ziang Chen, Qiao Zhang, and Runzhong Wang. On the expressive power of subgraph graph neural networks for graphs with bounded cycles, 2025. URL https://arxiv.org/abs/2502.03703.

Christopher Morris, Nils M. Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. TUDataset: A collection of benchmark datasets for learning with graphs. *arXiv2007.08663*, 2020b.

Vijay Prakash Dwivedi, Chaitanya K. Joshi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24(43):1–48, 2023.

Teague Sterling and John J. Irwin. Zinc 15 – ligand discovery for everyone. *Journal of Chemical Information and Modeling*, 55(11):2324–2337, 2015.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Proceeding of the NeurIPS*, volume 33, pages 22118–22133, 2020.

Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. An end-to-end deep learning architecture for graph classification. In *Proceedings of the AAAI*, 2018.

Giorgos Bouritsas, Fabrizio Frasca, Stefanos Zafeiriou, and Michael M Bronstein. Improving graph neural network expressivity via subgraph isomorphism counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):657–668, 2022.

Cristian Bodnar, Fabrizio Frasca, Yu Guang Wang, Nina Otter, Guido Montúfar, Pietro Liò, and Michael Bronstein. Weisfeiler and Lehman Go Topological: Message Passing Simplicial Networks. *arXiv2103.03212*, 2021.

Lingxiao Zhao, Wei Jin, Leman Akoglu, and Neil Shah. From stars to subgraphs: Uplifting any GNN with local structure awareness. In *Proceedings of the ICLR*, 2022b.

DingYi Zeng, Wanlong Liu, Wenyu Chen, Li Zhou, Malu Zhang, and Hong Qu. Substructure Aware Graph Neural Networks. *Proceedings of the AAAI*, 37(9):11129–11137, Jun. 2023.

## NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

**The checklist answers are an integral part of your paper submission.** They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: **[TODO]**

   Justification: **[TODO]**

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: **[TODO]**

    Justification: **[TODO]**

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.