The Study of Positional Encodings in GNNs Using Classical Graph Methods

GAL SALMAN, ROY DAHAN, NOY NETANEL, and ORI TADELA

TL;DR. We propose to incorporate traversal-based positional encodings (e.g., BFS rank) into GNNs, and evaluate whether sinusoidal transformations and classical features improve expressivity and alignment performance.

1 BACKGROUND & MOTIVATION

Graph Neural Networks (GNNs) are powerful tools for learning over structured data. However, most GNNs are inherently permutation-invariant and lack awareness of node positions beyond local neighborhoods. This limits their ability to reason about structural roles, symmetries, and alignments between graphs.

- **Problem statement** — We aim to explore whether traversal-based positional encodings (e.g., BFS order) can serve as useful inductive biases for GNNs. Specifically, we study their potential to enhance performance in tasks where relative node position matters, such as graph alignment.
- **Why it matters** — Tasks like node matching and graph isomorphism detection require structural awareness beyond local neighborhoods. Encoding traversal order may provide lightweight, interpretable global signals that help distinguish node roles and improve alignment robustness, all while utilizing BFS which has the time complexity: Moreover, traversal-based methods such as BFS operate with a time complexity of $O(|V| + |E|) \leq O(V^2)$, making them substantially more efficient than spectral decomposition techniques, which typically incur a higher computational cost of $O(|V|^3)$
- **Prior work limits** — Spectral methods like Laplacian eigenvectors capture global information but are unstable under perturbations (Dwivedi et al., 2020). Rewiring techniques with positional encodings (Dwivedi et al., 2022) improve structure-awareness, but do not exploit traversal order explicitly. Expressiveness limits of message-passing GNNs were analyzed in Xu et al., 2019, motivating the search for richer node representations.

2 PROPOSED APPROACH

- **Core idea** — We propose using traversal-derived ranks (e.g., BFS order) as positional encodings in GATs (Seongjun Yun et al.). These encodings are either inserted directly as scalar features or passed through sinusoidal transformation (like in transformers). We also investigate combining them with classical graph features (e.g., centrality measures).
- **Technical outline** — We will evaluate five model configurations:

(1) **Baseline GAT** — a standard GAT with no positional encoding.
(2) **BFS GAT** — GAT with raw BFS traversal index as node feature.
(3) **BFS + Sinusoidal** — BFS rank passed through sine and cosine encoding.
(4) **BFS + Classical** — BFS rank combined with node centrality metrics (e.g., eigenvector, closeness).
(5) **Full Combination** — All above encodings combined as a multi-channel input.

- **Novelty** — While positional encoding in graphs has been explored using Laplacian eigenvectors and random walksDwivedi et al., our method leverages a deterministic, traversal-based ordering that may offer new expressive

Authors' address: Gal Salman; Roy Dahan; Noy Netanel; Ori Tadela.

power. Using sinusoidal activations on BFS order is, to our knowledge, novel in GNN literature and offers a lightweight method to encode positions of nodes.

3 EXPERIMENTAL PLAN

- **Datasets** — We plan to evaluate on graph alignment benchmarks such as the dataset introduced in Graph Alignment for Benchmarking GNNs (Xu et al., 2024). This dataset includes noisy isomorphic pairs, where distinguishing roles and matching structures is essential.
- **Baselines / comparisons** — We will compare to:
  - A standard GAT without positional encodings
  - GAT with Laplacian/spectral encodings
- **Metrics** — We will measure:
  - Node alignment accuracy
  - Structural matching F1 score
  - Embedding similarity metrics across aligned nodes
- **Additional variations to explore** —
  - Multiple BFS traversals from high-centrality nodes (e.g., top-k by betweenness)
  - Injecting positional encodings after each GNN layer (layerwise PE)
  - Rewiring edges based on BFS tree structure to evaluate structural transformations

REFERENCES

(1) Graph Alignment for Benchmarking Graph Neural Networks
(2) Rewiring with Positional Encodings for Graph Neural Networks
(3) Graph Transformer Networks
(4) Graph neural networks with learnable structural and positional representations