Adaptive Initial Residual Connections for GNNs with Theoretical Guarantees

Mohammad Shirzadi, Ali Safarpoor Dehkordi, Ahad N. Zehmakan

School of Computing, Australian National University, Canberra, Australia
{mohammad.shirzadi, ali.safarpoordehkordi, ahadn.zehmakan}@anu.edu.au

Abstract

Message passing is the core operation in graph neural networks, where each node updates its embeddings by aggregating information from its neighbors. However, in deep architectures, this process often leads to diminished expressiveness. A popular solution is to use residual connections, where the input from the current (or initial) layer is added to aggregated neighbor information to preserve embeddings across layers. Following a recent line of research, we investigate an adaptive residual scheme in which different nodes have varying residual strengths. We prove that this approach prevents oversmoothing; particularly, we show that the Dirichlet energy of the embeddings remains bounded away from zero. This is the first theoretical guarantee not only for the adaptive setting, but also for static residual connections (where residual strengths are shared across nodes) with activation functions. Furthermore, extensive experiments show that this adaptive approach outperforms standard and state-of-the-art message passing mechanisms, especially on heterophilic graphs. To improve the time complexity of our approach, we introduce a variant in which residual strengths are not learned but instead set heuristically, a choice that performs as well as the learnable version.

Introduction

Neural message passing (Gilmer et al. 2017) serves as the foundation of modern graph representation learning, providing a core mechanism for aggregating neighborhood information in graph-structured data. This principle underlies Graph Convolutional Networks (GCNs) (Kipf and Welling 2017) and their variants, such as GraphSAGE (Hamilton et al. 2017) and Graph Attention Networks (GAT) (Veličković et al. 2018), which learn node embeddings through iterative feature aggregation and transformation.

The versatility of graph neural networks (GNNs) has led to their successful deployment across numerous domains, for example, accurate traffic prediction in transportation systems (Van Langendonck, Castell-Uroz, and Barlet-Ros 2024), pandemic analysis in public health (Panagopoulos, Nikolentzos, and Vazirgiannis 2021), economic forecasting (Ye et al. 2021), and social network analysis (Hevapathige, Wang, and Zehmakan 2025; Hevapathige, Wijesinghe, and Zehmakan 2025; Dehkordi and Zehmakan 2025).

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

While powerful, these architectures face inherent challenges, such as oversmoothing, a phenomenon in which network propagation results in indistinguishable node embeddings (Oono and Suzuki 2020). This phenomenon may even occur in shallow networks (Wu et al. 2022). A common strategy to mitigate oversmoothing in GNNs is the use of residual connections (Gasteiger, Bojchevski, and Günnemann 2019; Huang and Carley 2019; Chen et al. 2020b), a technique that adds the input from the previous layers to the aggregated neighbor information (e.g., through summation or concatenation) to preserve information across layers. This approach has been shown to improve training and performance in deep learning architectures such as ResNets (He et al. 2016).

Residual connections may combine the current embeddings with the initial node embeddings, forming initial residual connections (IRC). More generally, they may combine the current embeddings with the outputs from previous layers, which we refer to as residual connections (RC). In this work, we focus on IRC. Earlier works on IRC typically treat the residual strengths as a fixed hyperparameter shared across all nodes, referred to as *static* IRC for simplicity (see, e.g., (Chen et al. 2020b; Scholkemper et al. 2025)). To further unlock the power of IRC, some research has considered a more dynamic and adaptive approach, cf. (Zhang et al. 2023). Following this line of work, in this paper, we study the *adaptive IRC*, in which each node is permitted to have a personalized residual strength. Our primary goal is to theoretically and experimentally investigate the ability of adaptive IRC to preserve feature information across layers and achieve superior accuracy in downstream tasks.

To assess whether a message passing mechanism retains the ability to distinguish node embeddings as the network depth increases, a popular choice is the rank of the node embedding matrix (Daneshmand et al. 2020). A higher rank indicates that the message passing mechanism preserves distinct information across nodes, enabling richer representations. However, we note that rank alone is insufficient to diagnose oversmoothing, even in oversmoothing regimes, because numerical rank may remain full due to infinitesimal differences (though the effective rank collapses). For a more reliable diagnosis, effective rank or energy decay of the embedding matrix must be used to complement standard rank analysis. Hence, we also use the Dirichlet energy (see mean average distance (Chen et al. 2020a), spectral rank (Zhang

et al. 2025), and normalized node similarities (Chen et al. 2025) for alternative equivalent measures), which quantifies the smoothness of node embeddings. If it decays to zero with depth, the embeddings become indistinguishable, indicating a loss of expressiveness. Conversely, bounded energy implies that the mechanism maintains discriminative power. Unlike rank, energy is a continuous quantifier (Horn and Johnson 2012), making it robust for tracking the degree of smoothing.

Now that our message passing mechanism and our measures of oversmoothing are established, we can summarize our main contributions as follows:

- **Theoretical Guarantee**: We provide a theoretical analysis demonstrating that the adaptive IRC framework avoids oversmoothing, even in deep architectures. Specifically, we show that the Dirichlet energy of the node embeddings in the adaptive IRC remains bounded away from zero, ensuring that the embeddings retain their discriminative power. As a special case, we prove that static IRC with activation functions mitigates oversmoothing, extending the result of the prior work (Scholkemper et al. 2025), which considered only the linear case without activation functions. We also prove that, through the aggregation process, in the adaptive IRC mechanism, the rank of the embedding matrix is fully preserved.
- **Enhanced GNN Performance**: Through extensive experiments on benchmark datasets, we demonstrate that our framework consistently outperforms state-of-the-art GNNs. Our results highlight the framework's ability to learn robust and discriminative node embeddings while maintaining scalability, as the node-specific residual strength can be efficiently learned during training.
- **Adaptive IRC with Non-learnable Residual Strengths**: To reduce the time complexity of our approach, we propose a variant of adaptive IRC where residual strengths are not learned, but instead assigned heuristically based on node centrality. In particular, we use PageRank scores to rank the nodes, assigning a higher fixed residual strength to a small fraction of top-ranked nodes and a lower fixed value to the rest. This simple PageRank-based strategy performs as well as the fully learnable variant while significantly reducing computational overhead.

Preliminaries

Let $G = (V, E, \mathbf{A})$ be an undirected weighted graph with a node set V of cardinality $|V| = n$, an edge set E, and a weight matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ where $[\mathbf{A}]_{ij} = a_{ij}$ represents the weight of the edge between nodes i and j. If no edge exists between nodes i and j, $a_{ij} = 0$. The node set is equal to $V = \{1, 2, \cdots, n\}$, and the one-hop neighborhood of node i is denoted as $N_i = \{j \in V \mid (i, j) \in E\}$. Each node is associated with a feature vector of size $1 \times d$, and the feature matrix for all nodes is denoted by $\mathbf{X} \in \mathbb{R}^{n\times d}$, where d is the dimension of the feature.

The degree matrix, $\mathbf{D} \in \mathbb{R}^{n\times n}$, is a diagonal matrix where $[\mathbf{D}]_{ii} := d_i = \sum_{j\in V} a_{ij}$ represents the weighted degree of node i, i.e., the sum of weights of edges from node i to its neighbors. Considering the Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, the normalized Laplacian matrix $\boldsymbol{\mathcal{L}}$ is defined as

$$\boldsymbol{\mathcal{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2} = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1/2} = \mathbf{I} - \boldsymbol{\mathcal{A}},$$

where $\boldsymbol{\mathcal{A}} := \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ is the normalized adjacency matrix. Normalizing the adjacency matrix ensures that its largest eigenvalue is 1, leading to the following spectral properties. Let $\gamma_1 \le \cdots \le \gamma_n$ be the eigenvalues of $\boldsymbol{\mathcal{L}}$, then $0 = \gamma_1 \le \cdots \le \gamma_n \le 2$. Let $\alpha_1 \ge \cdots \ge \alpha_n$ be the eigenvalues of $\boldsymbol{\mathcal{A}}$, then $1 = \alpha_1 \ge \cdots \ge \alpha_n \ge -1$; see (Chung 1997) for more details.

Let $\mathbf{M} \in \mathbb{R}^{n\times m}$ be a matrix with $n, m \in \mathbb{N}^+$ and $\text{rank}(\mathbf{M}) = r$. Then, the singular value decomposition of $\mathbf{M}$ is given by $\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$, where the columns of $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{m\times m}$ are orthogonal (i.e., $\mathbf{U}^\top\mathbf{U} = \mathbf{I} \in \mathbb{R}^{n\times n}$, $\mathbf{V}^\top\mathbf{V} = \mathbf{I} \in \mathbb{R}^{m\times m}$) and $\boldsymbol{\Sigma} \in \mathbb{R}^{n\times m}$ is zero everywhere except for entries on the main diagonal where the (j, j) entry is σ_j for $j = 1, \cdots, \min\{m, n\}$ and

$$\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_r > \sigma_{r+1} = \cdots = \sigma_{\min\{m,n\}} = 0.$$

Denoting the columns of $\mathbf{U}$ and $\mathbf{V}$ as $\mathbf{u}_1, \cdots, \mathbf{u}_n$ and $\mathbf{v}_1, \cdots, \mathbf{v}_m$, we can write $\mathbf{M} = \sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^\top$. The decomposition satisfies $\mathbf{M}\mathbf{v}_i = \sigma_i \mathbf{u}_i$ and $\mathbf{M}^\top\mathbf{u}_i = \sigma_i \mathbf{v}_i$ for $i = 1, \ldots, r$. The Frobenius norm of $\mathbf{M}$ is $\|\mathbf{M}\|_F^2 = \sum_{i=1}^{r} \sigma_i^2$, and the spectral norm is $\|\mathbf{M}\|_2^2 = \sigma_1$ (see (Horn and Johnson 2012; Davydov and Safarpoor 2021) for more details).

The column space of a matrix $\mathbf{M} \in \mathbb{R}^{n\times m}$, denoted by $\mathcal{R}(\mathbf{M})$, represents all possible linear combinations of its columns and is formally defined as $\mathcal{R}(\mathbf{M}) = \{\mathbf{M}\mathbf{x} : \mathbf{x} \in \mathbb{R}^m\} \subseteq \mathbb{R}^n$. Similarly, *the row space* $\mathcal{R}(\mathbf{M}^\top)$ captures all linear combinations of the rows of $\mathbf{M}$ and can be expressed as $\mathcal{R}(\mathbf{M}^\top) = \{\mathbf{M}^\top\mathbf{y} : \mathbf{y} \in \mathbb{R}^n\} \subseteq \mathbb{R}^m$. There is a connection between these spaces and singular vectors. Specifically, if $\mathbf{M}$ has rank r, its column and row spaces are spanned by the singular vectors as

$$\mathcal{R}(\mathbf{M}) = \text{span}\{\mathbf{u}_1, \ldots, \mathbf{u}_r\},\ \mathcal{R}(\mathbf{M}^\top) = \text{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_r\},$$

where $\{\mathbf{u}_i\}$ and $\{\mathbf{v}_i\}$ are the left and right singular vectors, respectively. This result is known as the fundamental theorem of linear algebra, the singular value decomposition version (Trefethen and Bau 2022). Furthermore, *the null space* of matrix $\mathbf{M}$ is denoted by $\mathcal{N}(\mathbf{M}) = \{\mathbf{x} \in \mathbb{R}^m : \mathbf{M}\mathbf{x} = \mathbf{0}\}$.

Adaptive Initial Residual Connection

We consider the following message passing, called **adaptive IRC**:

$$\boldsymbol{H}^{(\ell+1)} = \sigma\Big(\boldsymbol{\Lambda}\boldsymbol{\mathcal{A}}\boldsymbol{H}^{(\boldsymbol{\ell})}\boldsymbol{W}^{(\ell)} + (\mathbf{I} - \boldsymbol{\Lambda})\boldsymbol{H}^{(0)}\boldsymbol{\Theta}^{(\ell)}\Big) \quad (1)$$

where $\boldsymbol{H}^{(\ell)} \in \mathbb{R}^{n\times d_l}$ represents the node embedding matrix at layer ℓ, with n being the number of nodes and d_l being the dimensionality of the node embeddings at this layer. Specifically, $\boldsymbol{H}^{(0)}$ corresponds to the initial node feature matrix, which is the input to the network. Also, $\sigma(\cdot)$[1] denotes a nonlinear activation function, such as ReLU or sigmoid, applied

[1]We use σ to denote the activation function and σ_i to denote singular values indexed by i.

element-wise to its input. $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)$ is the *residual strength diagonal matrix* where entries $\lambda_i \in (0, 1)$ determine the weight assigned to the neighborhood aggregation for each node vs its own initial embedding. Through the paper, we assume $\lambda_{\min}$ and $\lambda_{\max}$ are, respectively, the minimum and maximum values of λ_i, $i = 1, \cdots, n$. $\boldsymbol{W}^{(\ell)}, \mathbf{\Theta}^{(\ell)} \in \mathbb{R}^{d_l \times d_{\ell+1}}$ are learnable weight matrices at layer l, which transform the aggregated neighborhood embeddings and the initial node features, respectively and $\mathbf{I}$ is the identity matrix.

It is worth noting that setting $\mathbf{\Lambda} = \mathbf{I}$ will recover the vanilla GCN (Kipf and Welling 2017), where no residual term is present. Moreover, when $\mathbf{\Lambda} = \beta\mathbf{I}$ for some $\beta \in (0, 1)$, meaning all nodes share the same residual connection strength, the static IRC model (Gasteiger, Bojchevski, and Günnemann 2019; Scholkemper et al. 2025) is recovered. *It's worth emphasizing that the intuition behind design* (1) *is that, instead of relying solely on aggregating information from neighboring nodes, each node adaptively retains a learnable portion of its initial embedding, an idea inspired by the Friedkin-Johnsen opinion dynamics model (Friedkin and Johnsen 1990; Shirzadi, Cruciani, and Zehmakan 2025; Shirzadi and Zehmakan 2025).*

To ensure generalization to unseen nodes, we define $\mathbf{\Lambda}$ based on the initial node features rather than learning a separate value for each node. Specifically, we set $\mathbf{\Lambda} = \text{diag}\left(\sigma\left(\boldsymbol{H}^{(0)}\boldsymbol{W}_{\text{att}}\right)\right)$, where $\boldsymbol{W}_{\text{att}} \in \mathbb{R}^{d_0 \times 1}$. This applies a fully connected layer followed by a sigmoid function to each node's initial features, producing residual strengths in the range 0 to 1. As the weights are shared and input-dependent, this formulation enables flexible and generalizable $\mathbf{\Lambda}$ values for unseen nodes.

Related Works

Message Passing in GNNs. The earliest GNN architectures drew inspiration from spectral graph theory (Defferrard, Bresson, and Vandergheynst 2016), utilizing graph Fourier transforms to extract structural patterns in the spectral domain. This feature extraction can be performed either discretely, as seen in GCN (Kipf and Welling 2017), GraphSAGE (Hamilton et al. 2017), and GAT (Veličković et al. 2018), or continuously through diffusion PDEs, such as in graph neural diffusion (Chamberlain et al. 2021; Thorpe et al. 2022) and Allen-Cahn message passing (Wang et al. 2022). Continuous message passing is inspired by the framework of neural differential equations (Chen et al. 2018), which has led to many follow-up works in the GNN field (Avelar et al. 2019; Poli et al. 2019; Wu et al. 2023a; Rusch et al. 2022; Gallicchio and Micheli 2020).

Oversmoothing. A major challenge for GNNs is their limited depth. As layers increase, models like GCN (Oono and Suzuki 2020) and GAT (Wang et al. 2019; Wu et al. 2023b; Dong, Cordonnier, and Loukas 2021) often suffer performance degradation due to repeated neighborhood averaging, which makes node embeddings increasingly similar and eventually indistinguishable. The phenomenon, first noted by (Li, Han, and Wu 2018), arises from repeated Laplacian smoothing that drives embeddings in a connected graph toward uniform values. Later studies (Oono and Suzuki 2020; Cai and Wang 2020) further showed that the embedding energy decays to zero with depth.

Residual Connection. Motivated by the great success of residual neural networks in conventional deep learning (He et al. 2016), there has been a growing interest in incorporating RC in GNNs. An early example is provided by (Li et al. 2019), where the authors demonstrated that RC leads to significant improvements in experiments. (Liu et al. 2021) provided a message passing with an adaptive embedding aggregation and RC. (Yang et al. 2022; Chen et al. 2023) proposed difference RC, a method that helps GNNs focus only on the remaining useful information (the difference between input and output) at each layer. This prevents the loss of important details when stacking multiple layers.

Initial Residual Connection. GCNII (Chen et al. 2020b) demonstrated the effectiveness of IRC combined with identity mapping, enabling deeper architectures while preserving model performance. The study by (Scholkemper et al. 2025) demonstrated that IRC, in GNNs without activation functions, can mitigate oversmoothing when measured by mean average distance. The adaptive IRC with learnable residual strength through different layers has been studied by (Zhang et al. 2023), which is closely related to our work. However, unlike (Zhang et al. 2023), our adaptive message passing has significantly lower complexity. Firstly, in our framework, the personalized residual strengths are shared across layers, reducing the number of learnable parameters. Furthermore, we propose a PageRank-based heuristic variant of the framework, which further reduces complexity. More importantly, we provide a theoretical guarantee that adaptive IRC mitigates oversmoothing, in terms of Dirichlet energy, which was observed only experimentally by (Zhang et al. 2023) (for a more complex variant). As a byproduct of these results, we extend the theoretical results of (Scholkemper et al. 2025) on static IRC.

Similar methods that aggregate not just initial node features but also a combination of other layer embeddings show satisfactory performance, as seen in Jumping Knowledge Networks (JKNets) (Xu et al. 2018), Deep Adaptive Graph Neural Networks (DAGNNs) (Liu, Gao, and Ji 2020), R–SoftGraphAI (Li et al. 2024), and GODNF (Hevapathige, Wijesinghe, and Zehmakan 2025).

Theoretical Foundations

In this section, we present two key theoretical results on the expressive power of the message passing framework in Equation (1). First, Theorem 1 shows that, without activation functions or linear transformations, the embedding matrix preserves its initial rank, preventing dimensional collapse. Moreover, Theorem 2 proves that the energy function of message passing (1), even with nonlinearities and layer-specific weights, remains bounded away from zero as the network depth increases.

Rank Preservation in Simplified Adaptive IRC

We begin our theoretical analysis by considering a simplified setup, where we remove the intermediate activation functions

and linear transformations to isolate the core averaging behavior of message passing. This relaxation leads to the following propagation rule

$$\mathbf{H}^{(\ell+1)} = \mathbf{\Lambda}\boldsymbol{\mathcal{A}}\mathbf{H}^{(\ell)} + (\mathbf{I} - \mathbf{\Lambda})\mathbf{H}^{(0)}, \tag{2}$$

This simplification is consistent with prior work (Wu et al. 2019), which argues that most of the benefit in GCNs comes from local averaging rather than from nonlinear activation functions.

Theorem 1. *Considering the simplified version of the message passing* (1) *given by* (2)*, where we have no activation function, no linear transformation, the system stabilizes and it maintains full rank embeddings for all $\ell \in \mathbb{N}$. More precisely, the limiting behavior is governed by*

$$\lim_{\ell\to\infty} \mathbf{H}^{(\ell)} = (\mathbf{I} - \mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{-1}(\mathbf{I} - \mathbf{\Lambda})\mathbf{H}^{(0)},$$

and the embedding space never collapses, as

$$\operatorname{rank}\big(\mathbf{H}^{(\ell)}\big) = \operatorname{rank}\big(\mathbf{H}^{(0)}\big).$$

Proof Sketch. Unfolding the update rule (2), gives

$$\mathbf{H}^{(\ell+1)} = (\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{\ell+1}\mathbf{H}^{(0)} + \left(\sum_{i=0}^{\ell} (\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{i}\right)(\mathbf{I} - \mathbf{\Lambda})\mathbf{H}^{(0)} \tag{3}$$

We claim that the finite sum of matrix powers in the last expression admits an exact closed-form expression. First, as $\boldsymbol{\mathcal{A}}$ is symmetric, we have $\|\boldsymbol{\mathcal{A}}\|_2 = 1$. For any matrix $\mathbf{M}$, the spectral radius $\rho(\mathbf{M})$ satisfies $\rho(\mathbf{M}) \le \|\mathbf{M}\|_2$. Therefore, we have

$$\rho(\mathbf{\Lambda}\boldsymbol{\mathcal{A}}) \le \|\mathbf{\Lambda}\boldsymbol{\mathcal{A}}\|_2 \le \|\mathbf{\Lambda}\|_2\|\boldsymbol{\mathcal{A}}\|_2 = \|\mathbf{\Lambda}\|_2 \times 1 < 1.$$

This spectral radius condition guarantees the convergence of the Neumann series $\sum_{i=0}^{\infty}(\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^i = (\mathbf{I} - \mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{-1}$ (Horn and Johnson 2012). Hence, $\sum_{i=0}^{\ell}(\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^i = \Big(\mathbf{I} - (\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{\ell+1}\Big)(\mathbf{I} - \mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{-1}$.

Substituting this result into Equation (3) yields

$$\begin{aligned}\mathbf{H}^{(\ell+1)} &= (\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{\ell+1}\mathbf{H}^{(0)} \\ &\quad + \big(\mathbf{I} - (\mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{\ell+1}\big)(\mathbf{I} - \mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{-1}(\mathbf{I} - \mathbf{\Lambda})\mathbf{H}^{(0)}.\end{aligned}$$

Hence, $\lim_{\ell\to\infty}\mathbf{H}^{(\ell)} = (\mathbf{I} - \mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{-1}(\mathbf{I} - \mathbf{\Lambda})\mathbf{H}^{(0)}$. For the rank preservation property, note that as $\lambda_i < 1$, for $i = 1, \cdots, n$, so $\mathbf{I} - \mathbf{\Lambda}$ is full-rank, and $\rho(\mathbf{\Lambda}\boldsymbol{\mathcal{A}}) < 1$ ensures $(\mathbf{I} - \mathbf{\Lambda}\boldsymbol{\mathcal{A}})^{-1}$ is full-rank. Putting all these together, the final results are obtained as for any matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$ (invertible) and $\mathbf{B} \in \mathbb{R}^{n\times n}$, we have $\operatorname{rank}(\mathbf{AB}) = \operatorname{rank}(\mathbf{B})$. The detailed proof is provided in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025). □

Dirichlet Energy

Our analysis in this section proceeds as follows: first, we establish Lemma 1, Lemma 2, and Corollary 1, which then enable the proof of our main result in Theorem 2.

Definition 1 (Dirichlet Energy)**.** *The Dirichlet energy of a scalar vector $\boldsymbol{f} \in \mathbb{R}^{n\times 1}$ on the graph $G = (V, E, \mathbf{A})$ is given by*

$$\mathcal{E}(\boldsymbol{f}) = \boldsymbol{f}^{\top}\boldsymbol{\mathcal{L}}\boldsymbol{f} = \frac{1}{2}\sum_{(i,j)\in E} a_{ij}\left(\frac{f_i}{\sqrt{1+d_i}} - \frac{f_j}{\sqrt{1+d_j}}\right)^2.$$

For a feature matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ with rows $\mathbf{x}_i \in \mathbb{R}^{1\times d}$, the Dirichlet energy extends naturally as

$$\mathcal{E}(\mathbf{X}) = \operatorname{tr}(\mathbf{X}^{\top}\boldsymbol{\mathcal{L}}\mathbf{X}) = \frac{1}{2}\sum_{(i,j)\in E} a_{ij}\left\|\frac{\mathbf{x}_i}{\sqrt{1+d_i}} - \frac{\mathbf{x}_j}{\sqrt{1+d_j}}\right\|_2^2.$$

This formulation quantifies the smoothness of the features over the graph; higher energy values indicate greater differences in features between adjacent nodes (Rusch, Bronstein, and Mishra 2023).

Lemma 1. *Let $\boldsymbol{W} \in \mathbb{R}^{d_\ell\times d_{\ell+1}}$ be a weight matrix of rank r. For any vector $\boldsymbol{f} \in \mathbb{R}^{1\times d_l}$ in the column space of $\boldsymbol{W}$, we have*

$$\|\boldsymbol{f}\boldsymbol{W}\|_2 \ge \|\boldsymbol{f}\|_2\sigma_r(\boldsymbol{W}),$$

where $\sigma_r(\boldsymbol{W})$ is the smallest non-zero singular value of $\boldsymbol{W}$.

Proof Sketch. The proof applies the singular value decomposition $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\top}$ to express $\mathbf{fW}$ as $\mathbf{y}\mathbf{\Sigma}\mathbf{V}^{\top}$, where $\mathbf{y} = \mathbf{fU}$. Since $\mathbf{f}$ lies in $\mathbf{W}$'s column space, $\|\mathbf{fW}\|_2$ simplifies to $\|\mathbf{y}\mathbf{\Sigma}\|_2$, which is bounded below by the smallest non-zero singular value $\sigma_r(\mathbf{W})$ multiplied by $\|\mathbf{y}\|_2$. Orthogonality ensures $\|\mathbf{y}\|_2 = \|\mathbf{f}\|_2$, yielding $\|\mathbf{fW}\|_2 \ge \sigma_r(\mathbf{W})\|\mathbf{f}\|_2$. The detailed proof is provided in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025). □

Lemma 2. *Let $\mathbf{\Lambda}$ be a diagonal residual strength matrix, and let $\boldsymbol{\mathcal{A}}$ be a normalized adjacency matrix of rank r. Then, for any vector $\boldsymbol{f} \in \mathbb{R}^{n\times 1}$ in the row space of $\boldsymbol{\mathcal{A}}$, we have*

$$\big\|\mathbf{\Lambda}\boldsymbol{\mathcal{A}}\boldsymbol{f}\big\|_2 \ge \lambda_{\min}\,\sigma_r(\boldsymbol{\mathcal{A}})\,\|\boldsymbol{f}\|_2,$$

where $\lambda_{\min}$ denotes the smallest diagonal entry of $\mathbf{\Lambda}$, and $\sigma_r(\boldsymbol{\mathcal{A}})$ is the smallest non-zero singular value of $\boldsymbol{\mathcal{A}}$.

Proof. The proof follows similar steps to the proof of Lemma 1 and is provided in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025) □

Combining Lemmata 1 and 2 gives the following corollary.

Corollary 1. *Let $\mathbf{\Lambda}$ be the (diagonal) residual strength matrix with minimal entry $\lambda_{\min} > 0$, $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n\times n}$ the normalized adjacency matrix, and $\mathbf{W} \in \mathbb{R}^{d_\ell\times d_{\ell+1}}$ a weight matrix. For any $\mathbf{X} \in \mathbb{R}^{n\times d_\ell}$ whose rows lie in $\operatorname{col}(\mathbf{W})$ and whose columns lie in $\operatorname{row}(\boldsymbol{\mathcal{A}})$, the Dirichlet energy satisfies*

$$\mathcal{E}(\mathbf{\Lambda}\boldsymbol{\mathcal{A}}\mathbf{X}\mathbf{W}) \ge \lambda_{\min}^2\sigma_r^2(\boldsymbol{\mathcal{A}})\sigma_r^2(\mathbf{W})\mathcal{E}(\mathbf{X}),$$

where $\sigma_r(\cdot)$ denotes the smallest non-zero singular value.

Proof. The proof is obtained by following the definition of the Dirichlet energy function and applying Lemmas 1 and 2. The detailed proof is provided in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025). □

Now, putting all of these together in conjunction with the following properties, we will prove Theorem 2.

Property 1. *The activation function $\sigma : \mathbb{R} \to \mathbb{R}$ is such that for any vector $\boldsymbol{f} \in \mathbb{R}^{n\times 1}$, the Dirichlet energy satisfies $\mathcal{E}\big(\sigma(\boldsymbol{f})\big) \ge \alpha^2\mathcal{E}\big(\boldsymbol{f}\big)$ for some positive constant $\alpha > 0$.*

This property ensures the Dirichlet energy is not reduced by more than a constant factor, preserving signal variation after activation. For example, the leaky ReLU function with negative slope $0 < \alpha < 1$ is defined as

$$\text{Leaky ReLU}(x) = \begin{cases} x & \text{if } x \geq 0, \\ \alpha x & \text{if } x < 0, \end{cases}$$

satisfies Property 1, as shown in the following lemma.

Lemma 3. *Let σ be leaky ReLU with negative slope $0 < \alpha < 1$, then,*

$$\mathcal{E}\big(\sigma(\boldsymbol{f})\big) \geq \alpha^2 \mathcal{E}\big(\boldsymbol{f}\big).$$

for any vector $\boldsymbol{f} \in \mathbb{R}^{n\times 1}$.

Proof Sketch. The proof establishes the inequality $\mathcal{E}(\sigma(f)) \geq \alpha^2 \mathcal{E}(f)$ by analyzing the energy function's edge-wise contributions across four cases of input signs. The detailed proof is provided in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025). □

Property 2. *For any ℓ, the we assume* $\operatorname{tr}\left(\boldsymbol{X}^\top \mathcal{L} \boldsymbol{Y}\right) \geq 0$ *where $\boldsymbol{X} = \boldsymbol{\Lambda}\boldsymbol{\mathcal{A}}\boldsymbol{H}^{(l)}\boldsymbol{W}^{(\ell)}$ and $\boldsymbol{Y} = (\boldsymbol{I} - \boldsymbol{\Lambda})\boldsymbol{H}^{(0)}\boldsymbol{\Theta}^{(\ell)}$.*

This property ensures that the difference vectors $\mathbf{x}_i - \mathbf{x}_j$ and $\mathbf{y}_i - \mathbf{y}_j$ are, on average, positively aligned across graph edges (i, j). In other words, the embeddings $\mathbf{X}$ and $\mathbf{Y}$ vary in similar directions among neighboring nodes. A negative trace, by contrast, would suggest that the initial and propagated embeddings vary in opposite directions across edges, potentially contradicting the homophily assumption, which states that connected nodes should have similar embeddings.

Theorem 2. *Assume the energy function of the initial embedding is strictly positive, i.e., $\mathcal{E}(\boldsymbol{H}^{(0)}) > 0$, and the smallest non-zero singular values of the weight matrices are uniformly lower bounded, i.e., for some strictly positive ϵ,*

$$\inf_{\ell}\left\{\sigma_r^2(\boldsymbol{W}^{(\ell)})\right\} = \overline{\sigma}_r(\boldsymbol{W}), \;\; \inf_{\ell}\left\{\sigma_r^2(\boldsymbol{\Theta}^{(\ell)})\right\} = \overline{\sigma}_r(\boldsymbol{\Theta}) > \epsilon.$$

where $\sigma_r(\cdot)$ denotes the smallest non-zero singular value. Further, suppose $\lambda_{\min} \geq \delta$ and $\lambda_{\max} \leq 1 - \delta'$ for some strictly positive δ and δ' and the activation function $\sigma(\cdot)$ satisfies Property 1 with parameter $\alpha > 0$. Then, under the assumption of the Corollary 1 and Property 2, the energy of the final embedding of the message passing (1) *admits the following lower bound*

$$\mathcal{E}(\boldsymbol{H}^{(\ell+1)}) \geq \frac{\zeta\overline{\sigma}_r(\boldsymbol{\Theta})}{1 - \eta\overline{\sigma}_r}\mathcal{E}(\boldsymbol{H}^{(0)}) > 0,$$

where $\eta := \alpha^2\lambda_{\min}^2\sigma_r^2(\boldsymbol{\mathcal{A}})$ and $\zeta := \alpha^2(1 - \lambda_{\max})^2$.

Proof Sketch. Beginning with the message passing (1) and using the Property 1 and Property (2), we derive

$$\begin{aligned}\mathcal{E}(\boldsymbol{H}^{(\ell+1)}) \geq \alpha^2\mathcal{E}\left(\boldsymbol{\Lambda}\boldsymbol{\mathcal{A}}\boldsymbol{H}^{(\ell)}\boldsymbol{W}^{(\ell)}\right) \\ + \alpha^2\mathcal{E}\left((\boldsymbol{I} - \boldsymbol{\Lambda})\boldsymbol{H}^{(0)}\boldsymbol{\Theta}^{(\ell)}\right).\end{aligned}$$

Exploiting the spectral properties of the matrices involved in Corollary 1, this further simplifies to

$$\mathcal{E}(\boldsymbol{H}^{(\ell+1)}) \geq \eta\sigma_r^2(\boldsymbol{W}^{(\ell)})\mathcal{E}(\boldsymbol{H}^{(\ell)}) + \zeta\sigma_r^2(\boldsymbol{\Theta}^{(\ell)})\mathcal{E}(\boldsymbol{H}^{(0)}).$$

Iterating this inequality backward through the layers and using the fact that $\sigma_r^2(\boldsymbol{W}^{(i)}) \geq \overline{\sigma}_r(\mathbf{W})$ and $\sigma_r^2(\boldsymbol{\Theta}^{(i)}) \geq \overline{\sigma}_r(\boldsymbol{\Theta})$, we have

$$\begin{aligned}\mathcal{E}(\boldsymbol{H}^{(\ell+1)}) \geq \left((\eta\overline{\sigma}_r(\mathbf{W}))^{\ell+1}\right)\mathcal{E}(\boldsymbol{H}^{(0)}) \\ + \left(\zeta\overline{\sigma}_r(\boldsymbol{\Theta})\sum_{k=0}^{\ell}(\eta\overline{\sigma}_r(\mathbf{W}))^k\right)\mathcal{E}(\boldsymbol{H}^{(0)}). \quad (4)\end{aligned}$$

For $\eta\overline{\sigma}_r(\mathbf{W}) < 1$ (since otherwise the lower bound diverges as $\ell \to \infty$, making the result trivial), the geometric series converges, and the first term vanishes as $\ell \to \infty$, giving

$$\mathcal{E}(\boldsymbol{H}^{(\ell+1)}) \geq \frac{\zeta\overline{\sigma}_r(\boldsymbol{\Theta})}{1 - \eta\overline{\sigma}_r}\mathcal{E}(\boldsymbol{H}^{(0)}).$$

Note that ζ is strictly positive because $1 - \lambda_{\max} \geq \delta' > 0$. Also, $\overline{\sigma}_r(\boldsymbol{\Theta})$ is strictly positive by assumption. Additionally, since $\lambda_{\min} \geq \delta$, we have $0 < \eta\overline{\sigma}_r < 1$, which implies $\frac{1}{1-\eta\overline{\sigma}_r} > 0$, and $\mathcal{E}(\boldsymbol{H}^{(0)})$ is positive by assumption. The detailed proof is provided in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025). □

Before providing our experimental results, we analyze the complexity of adaptive IRC below.

Time Complexity

The time complexity of one layer in the adaptive IRC message passing framework is derived from three key computational components: (1) the sparse neighborhood aggregation $\boldsymbol{\mathcal{A}}\boldsymbol{H}^{(\ell)}$ requiring $O(|E|d_\ell)$ operations; (2) the subsequent dense embedding transformation $\boldsymbol{H}^{(\ell)}\boldsymbol{W}^{(\ell)}$ with $O(nd_\ell d_{\ell+1})$ complexity; and (3) the residual projection $\boldsymbol{H}^{(0)}\boldsymbol{\Theta}^{(\ell)}$ costing $O(nd_0d_{\ell+1})$. The diagonal scaling operations contribute $O(nd_\ell)$ term, which is asymptotically negligible. Combining these dominant terms yields an overall layer complexity of $O(|E|d_\ell + nd_\ell d_{\ell+1} + nd_0d_{\ell+1})$. When all embedding dimensions are unified ($d_\ell = d_{\ell+1} = d_0 = d$), this expression simplifies to the more compact form $O(|E|d + nd^2)$, demonstrating that the adaptive IRC maintains computational efficiency of the vanilla GCN while providing the benefits of residual connections.

Experiments

Our experiments, whose source code is available at https://github.com/aSafarpoor/AdaptiveIRC, aim to (i) evaluate whether adaptive IRC mitigates over-smoothing, and (ii) assess its performance on node classification compared to established GNNs.

Oversmoothing Mitigation

We first demonstrate that in adaptive IRC, the nodes' embeddings do not collapse, and the energy level of the nodes' embeddings remains non-zero. We use an undirected synthetic graph generated from a stochastic block model, similar

to the setup in (Wang et al. 2022; Wu et al. 2022). The graph consists of 200 nodes divided into two equal classes, with two-dimensional features sampled from a normal distribution. Both classes share the same standard deviation 2 but have different means ($\mu_1 = -0.5$, $\mu_2 = 0.5$). To model homophily, nodes within the same class are connected with a higher probability ($p = 0.2$), while nodes from different classes are connected with a lower probability ($q = 0.05$). As an early observation, Figure 1 shows that GCN embeddings collapse after a few layers due to rank collapsing, while adaptive IRC preserves class separation even after 16 layers. A similar collapse behavior is observed for GAT and GraphSAGE with mean and max pooling, as reported in the full version of the paper (Shirzadi, Safarpoor Dehkordi, and Zehmakan 2025).

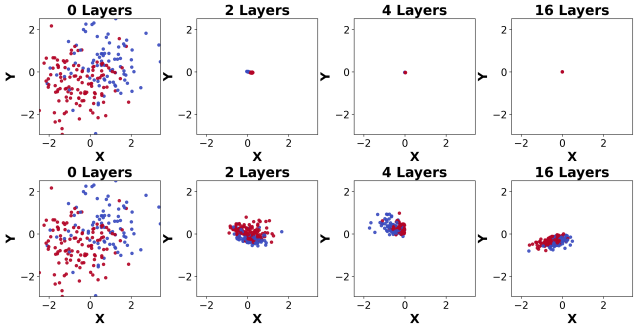

Figure 1: Embedding evolution in GCN vs adaptive IRC. GCN leads to embedding collapse, while adaptive IRC preserves distinct clusters.

In Figure 2, we plot the Dirichlet energy for the output of GCN, GAT, GraphSAGE, and adaptive IRC on a logarithmic scale with varying numbers of layers. The figure shows that the energy functions of all GCN, GAT, and GraphSAGE diminish as the number of hidden layers increases. In contrast, the energy level of the adaptive IRC remains notably positive, confirming that it effectively mitigates oversmoothing.

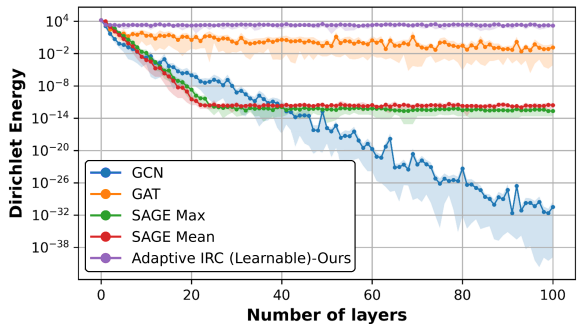

Figure 2: Dirichlet energy (log scale) for output of GCN, GAT, GraphSAGE (mean and max), and learnable adaptive IRC with varying numbers of layers.

Node Classification

We measure the performance of adaptive IRC on a range of common node classification benchmarks, using a collection of standard and state-of-the-art GNN models for comparison.

Other Methods. We compare the adaptive IRC against some classical, but popular, GNN architectures like GCN (Kipf and Welling 2017), GAT (Veličković et al. 2018), SGC (Wu et al. 2019), GraphSAGE (Hamilton et al. 2017), Mixhop (Abu-El-Haija et al. 2019), JKNet (Xu et al. 2018), and GNCII (Chen et al. 2020b). In addition, we compare our approach to some state-of-the-art GNN architectures, such as GraphGPS (Rampášek et al. 2022) and DIRGNNs (Rossi et al. 2024).

Our Methods. We investigate two variants of the adaptive IRC model. First, the **learnable adaptive IRC**, where the diagonal residual strength matrix is optimized during training. While effective, it requires optimizing residual strengths for each node to achieve optimal results. A natural question is then: can we avoid learning these parameters and instead assign them heuristically to reduce time complexity while still benefiting from adaptivity? We observed (on small synthetic graphs) a positive correlation between a node's centrality, such as PageRank, and its residual strength; that is, the higher the centrality, the higher the residual strength. Based on this, we propose the **PageRank-based adaptive IRC**, where we compute PageRank scores for all nodes and assign $\lambda_{\max}$ to the top $k\%$ (with $k \in \{5, 7, 10\}$), and $\lambda_{\min}$ to the rest. *This approach avoids learning residual strengths, reducing computational cost while continuing to perform comparably, and sometimes even better on average, than the learnable variant, as discussed below.*

Datasets. To evaluate model performance across diverse scenarios, for node classification task, we conduct experiments on homophilic graphs such as Cora (McCallum et al. 2000), Citeseer (Sen et al. 2008), and Pubmed (Namata et al. 2012), and on heterophilic graphs such as Texas, Wisconsin, and Cornell from WebK, as well as Chameleon, Squirrel, and Actor (Rozemberczki, Allen, and Sarkar 2021; Tang et al. 2009). These datasets exhibit different degrees of homophily, quantified by the homophily ratio H shown in Table 1. A higher ratio indicates greater homophily. For all datasets, we used 10 random weight initializations. We followed the default train/validation/test splits for datasets provided by the `torch-geometric` library.

Hyperparameters. For optimization, we employ the Adam optimizer with learning rates lr $\in \{10^{-2}, 10^{-3}\}$, weight decay set to 10^{-4}, hidden dimensions selected from $\{64, 128\}$, the number of hidden layers from $\{2, 4\}$, a dropout rate of 0.4, and residual strength assignments $\lambda_{\max} \in \{0.6, 0.7, 0.8\}$ and $\lambda_{\min} \in \{0.1, 0.2, 0.3\}$ for the PageRank-based version. For each model, the optimal hyperparameter configuration is determined via fine-tuning.

Performance. Table 1 summarizes the results of our experiments. Based on this table, except for the Actor dataset, the adaptive IRC consistently outperforms other methods. Between the two versions of our method, *PageRank-based* performs as well as, and sometimes even better than, the learnable version. This is a surprising result, as one might expect the learnable version to achieve higher accuracy. No-

Method	Cora (H: 0.83)	CiteSeer (H: 0.71)	Pubmed (H: 0.79)	Texas (H: 0.11)	Wisconsin (H: 0.21)	Cornell (H: 0.30)	Chameleon (H: 0.23)	Squirrel (H: 0.22)	Actor (H: 0.24)
GCN (Kipf and Welling 2017)	79.2±0.4	64.9±0.8	76.7±0.8	55.9±6.4	47.1±8.5	40.3±7.1	33.4±2.2	27.2±0.7	27.3±1.1
GraphSage-max (Hamilton et al. 2017)	75.6±1.0	62.3±1.1	75.8±0.8	72.7±6.6	74.3±5.8	69.7±3.1	50.6±2.2	36.8±1.3	32.9±1.4
GraphSage-mean (Hamilton et al. 2017)	79.8±0.5	66.9±1.0	75.8±0.6	74.6±6.0	76.7±5.3	69.5±3.8	50.2±1.7	36.7±2.3	34.1±1.0
GAT (Veličković et al. 2018)	75.6±0.4	66.2±1.1	75.8±1.7	57.3±5.9	50.4±7.8	44.6±5.7	39.6±1.8	30.7±1.6	28.1±1.2
JKNet (Xu et al. 2018)	79.5±0.4	66.7±0.7	76.7±0.4	55.1±7.1	51.6±4.6	46.8±5.4	35.5±1.6	28.0±1.0	28.4±0.9
Mixhop (Abu-El-Haija et al. 2019)	75.0±1.0	64.9±0.7	74.2±0.6	71.1±5.6	80.4±7.4	65.4±6.1	48.4±1.7	34.5±2.3	**36.2±1.0**
SGC (Wu et al. 2019)	79.6±0.9	65.8±0.8	**77.3±0.2**	57.0±3.5	48.8±6.6	41.4±6.0	35.2±2.6	27.7±1.0	27.3±1.0
GCNII (Chen et al. 2020b)	79.9±0.5	67.7±0.5	76.5±1.3	59.5±5.3	60.4±7.4	47.0±7.0	36.2±2.7	28.8±1.0	35.2±1.0
GraphGPS (Rampášek et al. 2022)	61.6±4.0	43.6±4.7	64.6±9.0	58.1±8.8	66.9±6.3	52.7±13	41.4±2.7	31.3±1.8	29.5±2.4
DirGNN (Rossi et al. 2024)	77.5±1.2	66.0±1.7	75.0±2.1	**84.6±6.1**	**82.2±2.3**	**71.6±3.9**	60.6±2.2	45.3±1.5	**36.6±0.8**
Adaptive IRC (Learnable)-Ours	**80.1±1.0**	**69.3±0.6**	76.6±0.6	73.0±5.8	**82.4±4.7**	67.8±4.8	**64.1±1.1**	**47.7±2.2**	34.2±1.3
Adaptive IRC (PageRank-based)-Ours	**80.7±0.4**	**70.2±0.4**	**77.4±0.6**	**77.0±6.8**	79.0±3.3	**72.4±5.7**	**65.0±2.0**	**49.0±2.2**	33.8±1.1

Table 1. Test accuracy and standard deviation over 10 experiments on each dataset, using different train/validation/test splits. Red is the best, Blue, the second best. The variable H stands for the homophily rate.

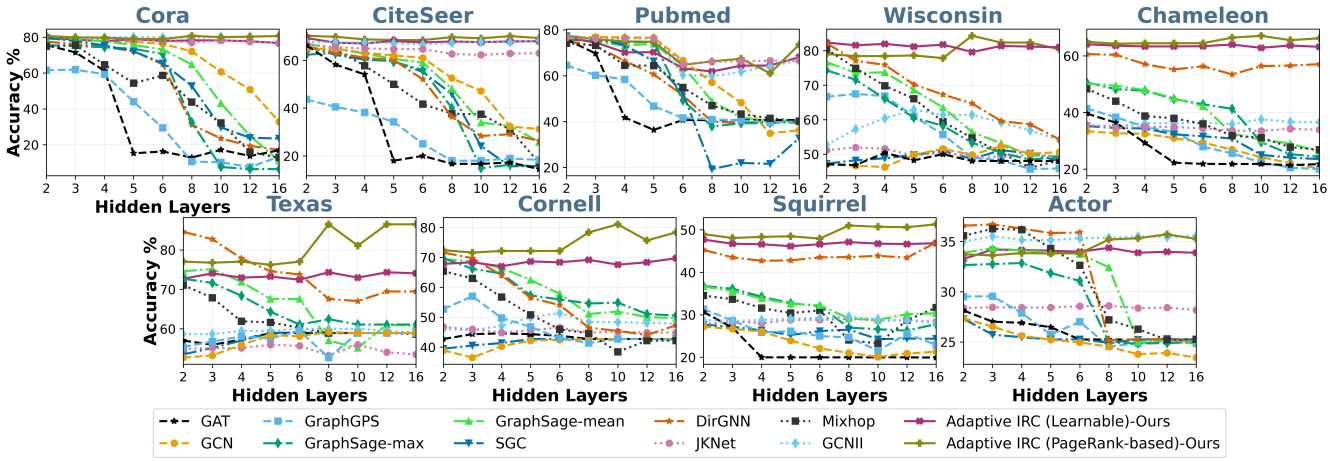

Figure 3: Performance across depths. Adaptive IRC (learnable and PageRank-based) remains accurate with increasing layers and outperforms other methods, except Actor, in shallow settings.

tably, there is a significant improvement in accuracy on heterophilic datasets compared to static residual connection models such as GCNII. To be specific, the PageRank-based IRC improves node classification accuracy (of the GCNII) by 17.5%, 18.6%, 25.4%, 28.8%, 20.2%, and -1.4% on the Texas, Wisconsin, Cornell, Chameleon, Squirrel, and Actor datasets, respectively, which shows the power of the adaptive IRC. This improvement stems from the adaptability of our method. Unlike GCNII, adaptive IRC adjusts the node-neighbor balance, preserving key node embeddings while leveraging neighbors when beneficial.

Depth. To demonstrate the versatility of the adaptive IRC, we plot the accuracy of all models across all datasets as the number of hidden layers increases. As shown in Figure 3, while other methods degrade beyond four layers, both the learnable and PageRank-based variants of the adaptive IRC maintain stable and high performance even with deeper architectures. On the *PubMed* dataset, accuracy drops from layer 5 to 6 across all models, likely due to the use of batch training in deeper networks. On the *Actor* dataset, while some baselines perform better with shallow architectures (up to 5 layers), our methods outperform them in deeper settings. The best hyperparameters were used for each model up to five layers. For models with more than five layers, we adopted the hyperparameters from the five-layer models.

Conclusion and Future Work

We study an adaptive residual scheme in which different nodes can have varying residual strengths. Our analysis demonstrates that this prevents oversmoothing (the Dirichlet energy remains non-zero) and maintains the embedding matrix's rank across layers. To improve the time complexity of our approach, we introduce a variant in which residual strengths are not learned but are set heuristically, a choice that performs as well as the learnable version. An interesting future direction is the *adaptive selection of residual strengths* without learning, as in our PageRank-based approach, which outperforms the learnable variant and other residual-based GNNs. A practical strategy is to assign shared residual strengths to groups of structurally similar nodes, leading to more effective and interpretable designs.

References

Abu-El-Haija, S.; Perozzi, B.; Kapoor, A.; Alipourfard, N.; Lerman, K.; Harutyunyan, H.; Ver Steeg, G.; and Galstyan, A. 2019. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *International Conference on Machine Learning*, 21–29. PMLR.

Avelar, P. H.; Tavares, A. R.; Gori, M.; and Lamb, L. C. 2019. Discrete and continuous deep residual learning over graphs. *arXiv preprint arXiv:1911.09554*.

Cai, C.; and Wang, Y. 2020. A note on over-smoothing for graph neural networks. *arXiv preprint arXiv:2006.13318*.

Chamberlain, B.; Rowbottom, J.; Gorinova, M. I.; Bronstein, M.; Webb, S.; and Rossi, E. 2021. Grand: Graph neural diffusion. In *International Conference on Machine Learning*, 1407–1418. PMLR.

Chen, D.; Lin, Y.; Li, W.; Li, P.; Zhou, J.; and Sun, X. 2020a. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 3438–3445.

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020b. Simple and deep graph convolutional networks. In *International Conference on Machine Learning*, 1725–1735. PMLR.

Chen, R. T.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018. Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, 31.

Chen, Y.; Luo, Y.; Tang, J.; Yang, L.; Qiu, S.; Wang, C.; and Cao, X. 2023. LSGNN: Towards General Graph Neural Network in Node Classification by Local Similarity. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 3550–3558.

Chen, Z.; Lin, Z.; Chen, S.; Polyanskiy, Y.; and Rigollet, P. 2025. Residual connections provably mitigate oversmoothing in graph neural networks. *arXiv e-prints*, arXiv–2501.

Chung, F. R. 1997. *Spectral graph theory*, volume 92. American Mathematical Soc.

Daneshmand, H.; Kohler, J.; Bach, F.; Hofmann, T.; and Lucchi, A. 2020. Batch normalization provably avoids ranks collapse for randomly initialised deep networks. *Advances in Neural Information Processing Systems*, 33: 18387–18398.

Davydov, O.; and Safarpoor, M. 2021. A meshless finite difference method for elliptic interface problems based on pivoted QR decomposition. *Applied Numerical Mathematics*, 161: 489–509.

Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems*, 29.

Dehkordi, A. S.; and Zehmakan, A. N. 2025. Graph-based Fake Account Detection: A Survey. *arXiv preprint arXiv:2507.06541*.

Dong, Y.; Cordonnier, J.-B.; and Loukas, A. 2021. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, 2793–2803. PMLR.

Friedkin, N. E.; and Johnsen, E. C. 1990. Social influence and opinions. *Journal of Mathematical Sociology*, 15(3-4): 193–206.

Gallicchio, C.; and Micheli, A. 2020. Fast and deep graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 3898–3905.

Gasteiger, J.; Bojchevski, A.; and Günnemann, S. 2019. Predict then Propagate: Graph Neural Networks meet Personalized PageRank. In *International Conference on Learning Representations*.

Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, 1263–1272. PMLR.

Hamilton et al. 2017. Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems*, 30.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Hevapathige, A.; Wang, Q.; and Zehmakan, A. N. 2025. DeepSN: A Sheaf Neural Framework for Influence Maximization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17177–17185.

Hevapathige, A.; Wijesinghe, A.; and Zehmakan, A. N. 2025. Graph Neural Diffusion via Generalized Opinion Dynamics. *arXiv preprint arXiv:2508.11249*.

Horn, R. A.; and Johnson, C. R. 2012. *Matrix analysis*. Cambridge University Press.

Huang, B.; and Carley, K. M. 2019. Residual or gate? towards deeper graph neural networks for inductive graph representation learning. *arXiv preprint arXiv:1904.08035*.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

Li, G.; Muller, M.; Thabet, A.; and Ghanem, B. 2019. DeepGCNs: Can GCNs go as deep as CNNs? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9267–9276.

Li, J.; Zhang, Q.; Xu, S.; Chen, X.; Guo, L.; and Fu, Y.-G. 2024. Curriculum-enhanced residual soft an-isotropic normalization for over-smoothness in deep GNNs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 13528–13536.

Li, Q.; Han, Z.; and Wu, X.-M. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 32.

Liu, M.; Gao, H.; and Ji, S. 2020. Towards deeper graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 338–348.

Liu, X.; Ding, J.; Jin, W.; Xu, H.; Ma, Y.; Liu, Z.; and Tang, J. 2021. Graph neural networks with adaptive residual. *Advances in Neural Information Processing Systems*, 34: 9720–9733.

McCallum, A. K.; Nigam, K.; Rennie, J.; and Seymore, K. 2000. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3: 127–163.

Namata, G.; London, B.; Getoor, L.; Huang, B.; and Edu, U. 2012. Query-driven active surveying for collective classification. In *10th International Workshop on Mining and Learning with Graphs*, volume 8, 1.

Oono, K.; and Suzuki, T. 2020. Graph Neural Networks Exponentially Lose Expressive Power for Node Classification. In *International Conference on Learning Representations*.

Panagopoulos, G.; Nikolentzos, G.; and Vazirgiannis, M. 2021. Transfer graph neural networks for pandemic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 4838–4845.

Poli, M.; Massaroli, S.; Park, J.; Yamashita, A.; Asama, H.; and Park, J. 2019. Graph neural ordinary differential equations. *arXiv preprint arXiv:1911.07532*.

Rampášek, L.; Galkin, M.; Dwivedi, V. P.; Luu, A. T.; Wolf, G.; and Beaini, D. 2022. Recipe for a general, powerful, scalable graph transformer. *Advances in Neural Information Processing Systems*, 35: 14501–14515.

Rossi, E.; Charpentier, B.; Di Giovanni, F.; Frasca, F.; Günnemann, S.; and Bronstein, M. M. 2024. Edge directionality improves learning on heterophilic graphs. In *Learning on Graphs Conference*, 25–1. PMLR.

Rozemberczki, B.; Allen, C.; and Sarkar, R. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2): cnab014.

Rusch, T. K.; Bronstein, M. M.; and Mishra, S. 2023. A Survey on Oversmoothing in Graph Neural Networks. *SAM Research Report*, 2023.

Rusch, T. K.; Chamberlain, B.; Rowbottom, J.; Mishra, S.; and Bronstein, M. 2022. Graph-coupled oscillator networks. In *International Conference on Machine Learning*, 18888–18909. PMLR.

Scholkemper, M.; Wu, X.; Jadbabaie, A.; and Schaub, M. T. 2025. Residual Connections and Normalization Can Provably Prevent Oversmoothing in GNNs. In *Proceedings of the Twelfth International Conference on Learning Representations*.

Sen, P.; Namata, G.; Bilgic, M.; Getoor, L.; Galligher, B.; and Eliassi-Rad, T. 2008. Collective classification in network data. *AI magazine*, 29(3): 93–93.

Shirzadi, M.; Cruciani, E.; and Zehmakan, A. N. 2025. Opinion Dynamics: A Comprehensive Overview. *arXiv preprint arXiv:2511.00401*.

Shirzadi, M.; Safarpoor Dehkordi, A.; and Zehmakan, A. N. 2025. Adaptive Initial Residual Connections for GNNs with Theoretical Guarantees. *arXiv preprint arXiv:2511.06598*.

Shirzadi, M.; and Zehmakan, A. N. 2025. Do stubborn users always cause more polarization and disagreement? a mathematical study. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, 309–317.

Tang, J.; Sun, J.; Wang, C.; and Yang, Z. 2009. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 807–816.

Thorpe, M.; Nguyen, T.; Xia, H.; Strohmer, T.; Bertozzi, A.; Osher, S.; and Wang, B. 2022. GRAND++: Graph neural diffusion with a source term. *ICLR*.

Trefethen, L. N.; and Bau, D. 2022. *Numerical linear algebra*. SIAM.

Van Langendonck, L.; Castell-Uroz, I.; and Barlet-Ros, P. 2024. Towards a graph-based foundation model for network traffic analysis. In *Proceedings of the 3rd GNNet Workshop on Graph Neural Networking Workshop*, 41–45.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

Wang, G.; Ying, R.; Huang, J.; and Leskovec, J. 2019. Improving graph attention networks with large margin-based constraints. *arXiv preprint arXiv:1910.11945*.

Wang, Y.; Yi, K.; Liu, X.; Wang, Y. G.; and Jin, S. 2022. ACMP: Allen-Cahn message passing for graph neural networks with particle phase transition. *arXiv preprint arXiv:2206.05437*.

Wu, F.; Souza, A.; Zhang, T.; Fifty, C.; Yu, T.; and Weinberger, K. 2019. Simplifying graph convolutional networks. In *International Conference on Machine Learning*, 6861–6871. PMLR.

Wu, Q.; Yang, C.; Zhao, W.; He, Y.; Wipf, D.; and Yan, J. 2023a. DIFFormer: Scalable (Graph) Transformers Induced by Energy Constrained Diffusion. In *The Eleventh International Conference on Learning Representations*.

Wu, X.; Ajorlou, A.; Wu, Z.; and Jadbabaie, A. 2023b. Demystifying oversmoothing in attention-based graph neural networks. *Advances in Neural Information Processing Systems*, 36: 35084–35106.

Wu, X.; Chen, Z.; Wang, W.; and Jadbabaie, A. 2022. A non-asymptotic analysis of oversmoothing in graph neural networks. *arXiv preprint arXiv:2212.10701*.

Xu, K.; Li, C.; Tian, Y.; Sonobe, T.; Kawarabayashi, K.-i.; and Jegelka, S. 2018. Representation learning on graphs with jumping knowledge networks. In *International Conference on Machine Learning*, 5453–5462. PMLR.

Yang, L.; Peng, W.; Zhou, W.; Niu, B.; Gu, J.; Wang, C.; Guo, Y.; He, D.; and Cao, X. 2022. Difference residual graph neural networks. In *Proceedings of the 30th ACM international Conference on Multimedia*, 3356–3364.

Ye, J.; Sun, L.; Du, B.; Fu, Y.; and Xiong, H. 2021. Coupled layer-wise graph convolution for transportation demand prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 4617–4625.

Zhang, K.; Deidda, P.; Higham, D.; and Tudisco, F. 2025. Rethinking Oversmoothing in Graph Neural Networks: A Rank-Based Perspective. *arXiv preprint arXiv:2502.04591*.

Zhang, L.; Yan, X.; He, J.; Li, R.; and Chu, W. 2023. Drgcn: Dynamic evolving initial residual for deep graph convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 11254–11261.