How Wide and How Deep? Mitigating Over-squashing of GNNs via Channel Capacity Constrained Estimation

Zinuo You[1], **Jin Zheng**[2], **John Cartlidge**[2]

[1]School of Computer Science, University of Bristol
[2]School of Engineering Mathematics and Technology, University of Bristol
{zinuo.you, jin.zheng, john.cartlidge}@bristol.ac.uk

Abstract

Existing graph neural networks typically rely on heuristic choices for hidden dimensions and propagation depths, which often lead to severe information loss during propagation, known as over-squashing. To address this issue, we propose Channel Capacity Constrained Estimation (C^3E), a novel framework that formulates the selection of hidden dimensions and depth as a nonlinear programming problem grounded in information theory. Through modeling spectral graph neural networks as communication channels, our approach directly connects channel capacity to hidden dimensions, propagation depth, propagation mechanism, and graph structure. Extensive experiments on nine public datasets demonstrate that hidden dimensions and depths estimated by C^3E can mitigate over-squashing and consistently improve representation learning. Experimental results show that over-squashing occurs due to the cumulative compression of information in representation matrices. Furthermore, our findings show that increasing hidden dimensions indeed mitigates information compression, while the role of propagation depth is more nuanced, uncovering a fundamental balance between information compression and representation complexity.

Code — https://github.com/pixelhero98/C3E

Introduction

Graph Neural Networks (GNNs) have emerged as powerful tools in various graph-related learning tasks (Bruna et al. 2014; Defferrard, Bresson, and Vandergheynst 2016), which stem from the goal of learning meaningful representations over graphs. Broadly, GNNs can be categorized into spectral and spatial methods. The spatial GNNs propagate information in the spatial domain, such as node-wise or edge-wise operations (Hamilton, Ying, and Leskovec 2017; Xu et al. 2018a; Lee, Lee, and Kang 2019) and feature-dependent operations (Xu et al. 2018b; Veličković et al. 2018; Rampášek et al. 2022). Conversely, spectral GNNs (Defferrard, Bresson, and Vandergheynst 2016; Kipf and Welling 2017; He et al. 2021; Wang and Zhang 2022) are grounded in spectral graph theory, where propagation methods are grounded on spectral filters. Moreover, spectral GNNs offer transparent, traceable propagation over graphs that can be expressed in closed form via matrix operations, unlike most spatial GNNs. Despite these merits, GNNs suffer a fundamental drawback from increasing propagation depth, which leads to severe performance degradation. This phenomenon is commonly attributed to over-smoothing (Nt and Maehara 2019; Cai and Wang 2020) or over-squashing (Alon and Yahav 2020; Topping et al. 2021; Di Giovanni et al. 2023; Huang et al. 2024). The former refers to node representation becoming overly similar during propagation, and the latter refers to information in node representation being severely squashed into limited-size vectors during propagation, leading to catastrophic information loss in the learned representation (Alon and Yahav 2020; Topping et al. 2021; Di Giovanni et al. 2023).

While over-smoothing has been extensively studied and can be mitigated with various normalization and residual techniques (Nt and Maehara 2019; Cai and Wang 2020; Chen et al. 2020; Bodnar et al. 2022; Maskey et al. 2024), over-squashing remains less understood. Existing works have largely focused on modifying graph structures or attribute features to mitigate over-squashing. For instance, graph rewiring techniques (e.g., adding or dropping edges (Topping et al. 2021)) and spectral methods (e.g., spectral graph rewiring (Karhadkar, Banerjee, and Montúfar 2022) or modifying edges via spectral gap optimization (Gravina et al. 2025)). However, a growing body of theoretical works, Loukas (2020a); Di Giovanni et al. (2023), demonstrate that realizing effective information propagation in GNNs is not solely a function of propagation methods and graph structures; it is critically dependent on network architectures, i.e., propagation depths, and hidden dimensions. For example, Loukas (2020b) points out that existing GNNs fail to effectively propagate information unless the product of their propagation depths and hidden dimensions exceeds a certain polynomial related to the graph size. Similarly, Di Giovanni et al. (2023) further show that larger hidden dimensions can mitigate over-squashing, and specific propagation depths can be helpful for representation learning before leading to vanishing gradients. These align with prior empirical analysis (Yang et al. 2020; Cong, Ramezani, and Mahdavi 2021; Zhou et al. 2021) that such degradation arises mainly from over-simplifying learnable matrices in analysis. While prior works show that specific hidden dimensions and depth can mitigate over-squashing, they offer no method to obtain these parameters. In mod-

ern representation learning, graph features propagated via message passing are subsequently transformed by learnable weight matrices, producing fixed-size representation embeddings (Dwivedi et al. 2023). From an information-theoretic perspective, over-squashing essentially corresponds to information loss in these learned representation matrices. Since entropy measures a variable's uncertainty or information content (Jaynes 1957; Kullback 1997), it serves as a natural choice for quantifying the information retained in representation matrices. This provides an explicit way to track the information flow propagated through the network (Saxe et al. 2019; Wu et al. 2020; Shen et al. 2023), rather than inferring it indirectly from graph structures alone, like former remedies.

In this paper, we propose the **Channel Capacity Constrained Estimation** (C^3E), a novel theoretical framework for estimating optimal hidden dimensions and propagation depth for spectral GNNs before training. Although knowing the exact state of the network before training is impossible, we can leverage the principle of maximum entropy (Jaynes 1957) to estimate an upper bound on the information that a spectral GNN can propagate. Moreover, by invoking Shannon's Theorem (Shannon 1948), we note that a communication channel can achieve near error-free information transmission with a certain encoding scheme, if its capacity exceeds the information load. As former studies (Bruna et al. 2014; Henaff, Bruna, and LeCun 2015; Defferrard, Bresson, and Vandergheynst 2016) suggest, learnable matrices can be interpreted as encoders of graph signals, affecting the encoded representation of the network. Similarly, we model a spectral GNN as a communication channel whose capacity depends on hidden dimensions and propagation depth. Under this perspective, estimating optimal width and depth reduces to maximizing channel capacity under Shannon's Theorem, yielding a nonlinear programming formulation. The contributions of this work are threefold. First, we provide an information-theoretic view to model information flow in spectral GNNs, which links hidden dimensions, depth, propagation methods, and graph structures to the encoded representation. Second, we formulate a nonlinear programming problem to estimate optimal hidden dimensions and propagation depth, providing a principled approach to choosing network architectures. Third, we demonstrate that optimal hidden dimensions and propagation depth derived from C^3E effectively mitigate over-squashing and consistently improve representation learning, without altering propagation methods or graphs.

Related Work

Information Theory in Representation Learning

Information theory (Shannon 1948; Jaynes 1957; Kullback 1997) has long been a powerful tool for analyzing neural networks. For instance, Saxe et al. (2019) have explored the entropy distribution and the information flow of learned representation in deep neural networks. It reveals that the information compression in representation matrices occurs along with the increase in network depth. The information bottleneck principle is broadly applied in neural networks to learn minimal effective representations, which maximize the mutual information between the learned representation and the target to alleviate potential information loss (Tishby, Pereira, and Bialek 2000; Wu et al. 2020). Furthermore, Sun et al. (2021) have managed to generate highly competitive deep convolutional neural networks (CNNs) based on the principle of maximum entropy (Jaynes 1957, 2003). Recently, some studies (Chan et al. 2022; Roberts, Yaida, and Hanin 2022) endeavor to establish relationships between entropy and representation matrices of neural networks.

Spectral GNNs

Spectral-based GNNs define propagation using spectral filters or kernel functions. Much of the progress in the field, from early models like GCN (Kipf and Welling 2017), APPNP (Gasteiger, Bojchevski, and Günnemann 2018), and SGC (Wu et al. 2019), to advanced methods like GDC (two variants: GDC_{HK} and GDC_{PPR}) (Gasteiger, Weißenberger, and Günnemann 2019), GPRGNN (Chien et al. 2020), S^2GC (Zhu and Koniusz 2021), ChebyNetII (He, Wei, and Wen 2022), and JacobiConv (Wang and Zhang 2022). has focused on designing sophisticated filters to overcome over-smoothing. While successful, this focus on the propagation mechanism has revealed a more fundamental and architectural bottleneck: over-squashing. This form of information loss stems not from the filter but from the capacity of learnable weight matrices that process propagated signals. As recent studies (Zhou et al. 2021; Cong, Ramezani, and Mahdavi 2021; Di Giovanni et al. 2023) confirm, the dimensions and depth of learnable transformations are critical factors, yet their choice has often been overlooked.

Preliminary

Entropy of Matrix

We define the entropy of a real-valued matrix $\mathbf{Z}$ by treating its entries as samples from a random variable $Z \sim p$. This entry-wise definition serves as a tractable proxy for the matrix's total information content by abstracting away higher-order correlations between entries (detailed **justification** and **discussion** are provided in Appendix A.1). If the latent distribution p is continuous, then its entropy is given by,

$$H(Z) = -\int_{-\infty}^{\infty} p(z) \ln\left(p(z)\right) dz. \tag{1}$$

For the given mean μ_Z and variance σ_Z^2, it is maximized by a Gaussian distribution $\mathcal{N}(\mu_Z, \sigma_Z^2)$, which yields,

$$H(Z) \leq \frac{1}{2} \ln\left(2\pi \mathrm{e} \sigma_Z^2\right). \tag{2}$$

Yet, we do not have access to the true underlying distribution p a priori. Instead, we have the finite-dimensional matrix $\mathbf{Z} \in \mathbb{R}^{\alpha \times \beta}$, which constitutes a finite set $\alpha\beta$ samples. These samples form an empirical distribution whose information content is captured by the discrete form,

$$H(Z) = -\sum_{i=1}^{|\mathrm{Supp}(Z)|} P(Z = z_i) \ln\left(P(Z = z_i)\right), \tag{3}$$

where $\text{Supp}(\cdot)$ denotes the support set, i.e., the set of all possible values $\{z_i\}$ that entries of $\mathbf{Z}$ can take. Then, the maximum entropy is bounded by,

$$H(Z) \leq \ln\left(|\text{Supp}(Z)|\right) \leq \ln\left(\alpha\beta\right). \tag{4}$$

The **proofs** of Eq. (2) and Eq. (4) are provided in Appendix A.2. These show that the maximum information a matrix can convey is capped by its dimensionality. A lower-dimensional matrix inherently has a smaller maximum support size, imposing a coarser discretization on the representation of any underlying distribution.

Entropy of Graph

Graph entropy quantifies the uncertainty or information content of a graph $\mathcal{G}$. While definitions vary, they generally measure the information content of the graph based on some property extraction function $g(\cdot)$ (e.g., eigenvector centrality or homophily/heterophily metrics). A generalized form of graph entropy (Dehmer and Mowshowitz 2011) is,

$$H(\mathcal{G}) = -\sum_{i=1}^{n} \frac{g(v_i)}{\sum_{j=1}^{n} g(v_j)} \ln \frac{g(v_i)}{\sum_{j=1}^{n} g(v_j)}. \tag{5}$$

Here, v_i denotes the i-th vertex, and n denotes the number of nodes.

Channel Capacity

In terms of information theory (Shannon 1948; Jaynes 1957; Gallager 1968), channel capacity is defined as the theoretical maximum of which information can be reliably transmitted over a communication channel. The channel capacity of a communication channel is expressed as (Shannon 1948),

$$\phi = \max\ I(f(\mathcal{M}); \mathcal{M}'). \tag{6}$$

Here, $I(f(\mathcal{M}); \mathcal{M}') = H(f(\mathcal{M})) - H(f(\mathcal{M})|\mathcal{M}')$ denotes the mutual information between encoded input $f(\mathcal{M})$ and the output $\mathcal{M}'$, and $f(\cdot)$ denotes the encoder.

Methodology

Theoretical Channel Capacity of Spectral GNNs

Drawing on prior research (Saxe et al. 2019; Sun et al. 2021; Chan et al. 2022; Shen et al. 2023; Yang et al. 2023) about information flows propagated in neural networks, we extend the classical definition of channel capacity to GNNs, which we model as communication channels. To establish a formal theoretical framework grounded in this perspective, a class of models with analytical tractability is required. Spectral GNNs provide the ideal characteristic, as their propagation mechanisms can be collapsed into a single matrix operator.

Consider a spectral GNN with L propagation layers learning representations on a graph $\mathcal{G}$. We collapse the propagation operation on the adjacency matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ into a propagation matrix $\mathbf{S}_l \in \mathbb{R}^{n\times n}$. Then, its layer-wise representation learning operation becomes,

$$\mathbf{H}_l = \Delta(\mathbf{S}_l \mathbf{H}_{l-1} \mathbf{W}_l). \tag{7}$$

Here, $\mathbf{H}_l \in \mathbb{R}^{n\times w_l}$ denotes the latent representation matrix, $\mathbf{H}_0 \in \mathbb{R}^{n\times m}$ denotes the initial feature matrix ($m = w_0$), $\mathbf{W}_l \in \mathbb{R}^{w_{l-1}\times w_l}$ denotes the learnable weight matrix, and $\Delta(\cdot)$ denotes the nonlinear activation function. Therefore, the encoded representation for the downstream task is $\mathbf{H}_L \in \mathbb{R}^{n\times w_L}$. In addition, the above layer-wise propagation framework directly extends to models like SGC and APPNP, whose underlying spectral formalisms provide a collapsible operator that is computed via a spatial message-passing scheme.

If the GNN encoder $f(\cdot)$ learns the exact mapping $f : \mathcal{G} \to \mathcal{G}'$, where $\mathcal{G}'$ is the graph $\mathcal{G}$ with node representations that are sufficient to resolve uncertainty about downstream tasks (e.g., semi-supervised node classification or property prediction), then the conditional entropy $H(\mathcal{G}'|f(\mathcal{G}))$ vanishes:

$$H(\mathcal{G}'|f(\mathcal{G})) = 0. \tag{8}$$

This means the encoded representation $f(\mathcal{G}) = \mathbf{H}_L$ perfectly matches $\mathcal{G}'$ with zero uncertainty. Given our lack of prior knowledge about network states in advance, then, by the principle of maximum entropy, we should select the probability distribution that best represents the system's current state, which is the one with the largest entropy, subject to known constraints (Jaynes 1957, 2003). Consequently, combining this crucial premise with Eq. (2), Eq. (6), Eq. (7), and Eq. (8), we arrive at the following Theorem.

Theorem 1 *The channel capacity of a spectral GNN is defined by maximizing the entropy of the encoded representation $\mathbf{H}_L$, which is expressed as,*

$$\begin{aligned}
\phi &= \max\ I(f(\mathcal{G}); \mathcal{G}') \\
&= \max\ H(f(\mathcal{G})) - H(f(\mathcal{G})|\mathcal{G}') \\
&= \max\ H(\mathbf{H}_L) \\
&= \max\ \left[\frac{1}{2}\ln\left(2\pi\mathrm{e}\right) + \frac{1}{2}\sum_{l=1}^{L} \ln\left(n w_{l-1} \sigma^2_{\mathbf{S}_l}\right)\right].
\end{aligned} \tag{9}$$

The **proof** is provided in Appendix B. Here, a larger channel capacity corresponds to greater information content that can be carried by the network, indicating that the network can represent a more complex and informative distribution. The formula shows that hidden dimensions and propagation depth are crucial to the channel capacity. The variance term $n\sigma^2_{\mathbf{S}_l}$ depicts the role of graph structures and propagation mechanisms, where larger hidden dimensions amplify their effects and smaller ones diminish them. Moreover, if hidden dimensions are too small, it results in reductions in channel capacity and information loss in the encoded representation as the propagation depth increases. This aligns with theoretical analysis from previous studies (Loukas 2020a; Di Giovanni et al. 2023) that the product of hidden dimensions and propagation operator should be sufficiently large to avoid information loss.

Information Compression and Over-squashing

Recently, some studies (Topping et al. 2021; Di Giovanni et al. 2023; Huang et al. 2024) have attempted to measure the information compression in GNNs via the graph bottleneck

or the Jacobian of latent representation. For instance, Topping et al. (2021) illustrate that the graph bottleneck leads to severe information compression and hence over-squashing, originating from high negative curvature edges. According to Saxe et al. (2019), information compression occurs along with the representation learning process. Furthermore, other studies (Di Giovanni et al. 2023; Huang et al. 2024) show that over-squashing is closely related to choices of hidden dimensions and propagation depth. However, these measures are hard to obtain due to complex graph structures and variations in training. Thus, we introduce a metric termed representation compression ratio to measure the severity of information compression,

$$\theta = \frac{\phi}{\overline{w}},\ \overline{w} = (\prod_{l=1}^{L} w_l)^{\frac{1}{L}}. \tag{10}$$

Here, $\overline{w}$ denotes the geometric mean of hidden dimensions, serving as an equivalent representation dimension per layer. The metric θ provides an explicit measure of the information compression intensity in the representation matrix. If $\theta \to \infty$, i.e., $\phi \gg \overline{w}$, then this implies that per equivalent representation dimension compresses massive information content severely. If $\theta \to 0$, i.e., $\overline{w} \gg \phi$, then this means that the representation dimension is over-provisioned relative to the information content.

To better understand the specific effects of hidden dimensions and propagation depth on information compression, we analyze based on the representation compression ratio θ. Substituting Eq. (9) into Eq. (10) and deriving the partial derivatives yields the following Corollary.

Corollary 2 *The representation compression ratio θ of a spectral GNN exhibits a dual dependency on $\overline{w}$ and L. Let $\overline{K} = \mathbb{E}_l[\ln(n\sigma^2_{\mathbf{S}_l})]$ be the average log propagation variance of graph structures and propagation operations.*

- *On $\overline{w}$: Given fixed L, θ is maximized at a threshold $\overline{w}^*$, where θ monotonically increases for $0 < \overline{w} \leq \overline{w}^*$ and monotonically decreases for $\overline{w} > \overline{w}^*$. The threshold $\overline{w}^* = \mathrm{e}^{\left[1 - \frac{\ln(2\pi\mathrm{e}) + \ln(m) - \ln(w_L) + \sum_{l=1}^{L} \ln(n\sigma^2_{\mathbf{S}_l})}{L}\right]}$ eventually converges to $\lim_{L\to\infty} \overline{w}^* = \mathrm{e}^{1-\overline{K}}$.*
- *On L: Given fixed $\overline{w}$, the effect of increasing L depends on $\overline{w}$ relative to properties of the graph and propagation method. If $\ln(\overline{w}) > -\overline{K}$, increasing L increases θ, exacerbating information compression. Conversely, when $\ln(\overline{w}) \leq -\overline{K}$ increasing L directly decreases ϕ and declines θ, symptomatic of information loss.*

The **proof** is provided in Appendix C. First, expanding hidden dimensions is the primary choice for mitigating high information compression (over-squashing). Second, better propagation methods or graph rewiring indeed mitigate such information compression by decreasing $\overline{w}^*$ and increasing $\overline{K}$. Nevertheless, the role of propagation depth is conditional and reveals two failure modes: deep and wide GNNs might suffer from over-squashing due to cumulative information compression (θ increases when $\overline{w} \leq \overline{w}^*$), while deep and narrow GNNs suffer from over-squashing (since the propagated information ϕ vanishes when $\ln(\overline{w}) < -\overline{K}$). These results in Corollary 2 align with former empirical findings (Loukas 2020b; Topping et al. 2021; Di Giovanni et al. 2023).

From Theoretical Limit to Effective Channel Capacity

Previously, Theorem 1 establishes the theoretical channel capacity of GNNs, which offers a critical upper bound on the information the network can encode. Nevertheless, this global perspective treats the network as a whole system, which does not explicitly consider the architectural constraints imposed by the layer-by-layer information propagation (Achille and Soatto 2018; Saxe et al. 2019). In practice, a very narrow layer following a very wide one will structurally cap the information passed to subsequent layers.

To fill this gap, we introduce the effective channel capacity ϕ_0, which accounts for architectural constraints between adjacent layers. By modeling each learnable weight matrix transformation as a communication channel, whose structure is analogous to a complete bipartite graph between its input and output neurons (Pellizzoni et al. 2024), the effective channel capacity ϕ_0 is defined by the following expression,

$$\phi_0 = \sum_{l=1}^{L} \frac{\ln\left(\frac{w_{l-1}w_l}{w_{l-1}+w_l}\right)}{\frac{\ln(2\pi\mathrm{e})}{\ln(nw_{l-1}\sigma^2_{\mathbf{S}_l})} + \sum_{o=1}^{l} \frac{\ln\left(nw_{o-1}\sigma^2_{\mathbf{S}_o}\right)}{\ln(nw_{l-1}\sigma^2_{\mathbf{S}_l})}}. \tag{11}$$

The **justification** is provided in Appendix D. This expression provides a practical and architecture-aware measure of channel capacity by capturing two fundamental dynamics. First, the numerator models architectural bottlenecks; it shows that large disparities between the widths of adjacent layers structurally reduce the information that can be retained. Second, the denominator reflects a cumulative attenuation effect, where the information from preceding layers (including initial features) diminishes the relative contribution of the current layer. Together, these terms formalize that to preserve sufficient information capacity, besides propagation mechanisms and graph structures, GNNs should avoid sharp changes in hidden dimensions and that deeper layers provide diminishing returns on the capacity. These findings provide a principled framework for empirical results observed in prior work (Loukas 2020a,b; Cong, Ramezani, and Mahdavi 2021; Di Giovanni et al. 2023).

Channel Capacity Constrained Estimation

The preceding analysis demonstrates the need to balance channel capacity against the risk of information compression. Accordingly, our estimation should effectively manage this trade-off. First, Shannon's Theorem (Shannon 1948) states that for near error-free information propagation, the channel capacity must meet or exceed the information being transmitted: $\phi \geq H(\cdot)$. Since $\mathcal{G}'$ is unknown a priori, we utilize the maximum possible graph entropy, $H_{\max}(\mathcal{G}) = \ln(n)$, as a safe lower bound for the required channel capacity. This gives the first condition: $\phi_0 \geq \ln(n)$. Second, as

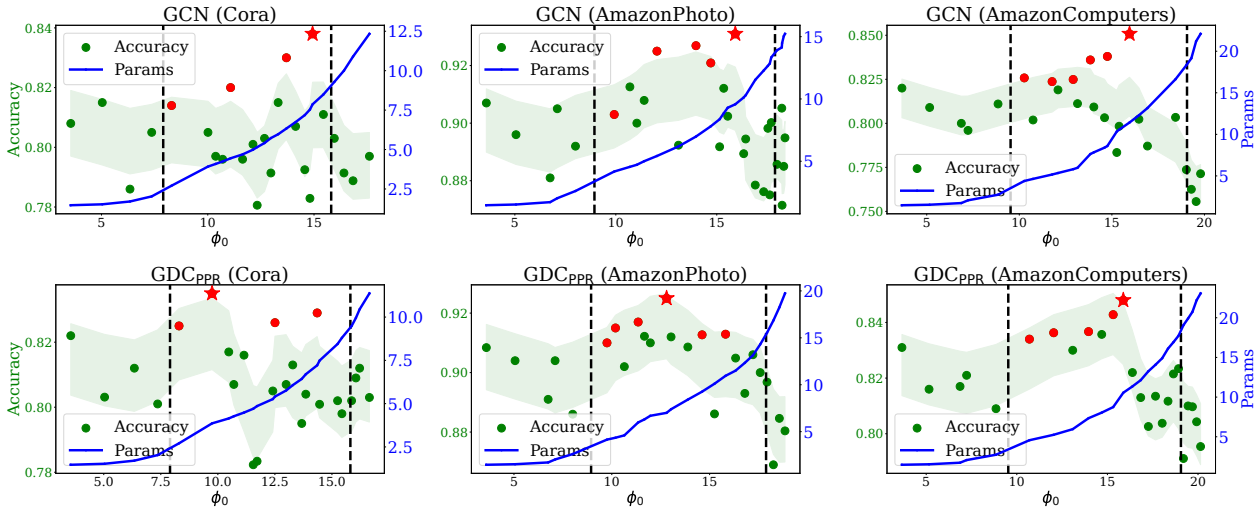

Figure 1: The green-axis (**left**) denotes performance, and blue-axis (**right**) denotes parameter counts (in millions). C^{3}E solutions (red points, **starred** for optimal) consistently land in high-performance regions within **dashed** intervals defined in Eq. (12), outperforming baselines with heuristic dimensions (green points, e.g., 8, 16, 32, 64, 128, 256, 512, 1024, and 2048).

Statistics	Cora	Citeseer	Pubmed	AmazonPhoto	AmazonComputers	Chameleon	Squirrel	ogbn-arxiv	ogbn-papers100M
# Node	2,708	3,327	19,717	7,650	13,752	2,277	5,201	169,343	111,059,956
# Feature	1,433	3,703	500	745	767	2,325	2,089	128	128
# Edges	5,429	4,732	44,338	119,043	245,778	36,101	217,073	1,166,243	1,615,685,872
# Classes	7	6	3	8	10	5	5	40	172
Avg Time	3.7	4.9	5.9	5.3	5.7	3.6	5.0	239.2	879.6

Table 1: Statistics of the experimental datasets, including average time (in seconds) for C^{3}E to generate solutions for baselines.

established in Corollary 2, simply maximizing channel capacity is not ideal, as it can lead to over-squashing by excessive information compression. To prevent this, we introduce a hyperparameter $\eta \in (0, 1]$ to regularize the effective channel capacity, imposing an upper bound on ϕ_0. This gives the second condition: $\phi_0 \leq \frac{1}{\eta}\ln(n)$. Then, the two conditions form the effective trade-off,

$$\ln(n) \leq \phi_0 \leq \frac{1}{\eta}\ln(n). \tag{12}$$

This constraint ensures the GNN has sufficient channel capacity and prevents runaway information compression. We treat η as a tunable regularizer. Our sensitivity analysis in Appendix F demonstrates, the framework remains robust across a wide range of η values; tuning up η speeds solutions via fewer candidate solutions, and vice versa.

Putting Theorem 1, Corollary 2, Eq. (12) together, the C^{3}E is expressed by the following form,

$$\begin{aligned} \max_{\mathbf{w}^{(L)},L} \quad & \left[\frac{1}{2}\ln(2\pi\mathrm{e}) + \frac{1}{2}\sum_{l=1}^{L}\ln(nw_{l-1}\sigma^2_{\mathbf{S}_l})\right] \\ s.t. \quad & \overline{w} > \overline{w}^*,\ \ln(\overline{w}) > -\overline{K}, \\ & \ln(n) \leq \phi_0 \leq \frac{1}{\eta}\ln(n). \end{aligned} \tag{13}$$

Here, $\mathbf{w}^{(L)} = \{w_1, ..., w_L\}$. The primary objective seeks to maximize the theoretical channel capacity. The first constraint prevents two failure modes: either excessive information compression or information loss as identified in Corollary 2. The second constraint enforces the GNN to possess sufficient channel capacity while avoiding the pitfalls of naively increasing L or $\overline{w}$. These constraints steer trivial solutions of infinitely large width or depth away and instead force a trade-off between propagation depth and hidden dimensions. Feasible solutions can be obtained using off-the-shelf solvers for constrained nonlinear programming, such as SLSQP (Kraft 1988). For implementation, hidden dimensions are treated as continuous values and rounded post hoc, and $\sigma^2_{\mathbf{S}_l}$ is precalculated sparsely by population variance. Example C^{3}E solutions are provided in Appendix G.

Experiments

In this work, we conduct semi-supervised node classification with eight GNNs on seven public graphs: Cora (Sen et al. 2008), Citeseer (Sen et al. 2008), Pubmed (Sen et al. 2008), AmazonPhoto (Shchur et al. 2018), AmazonComputer (Shchur et al. 2018), Chameleon (Rozemberczki, Allen, and Sarkar 2021), and Squirrel (Rozemberczki, Allen, and Sarkar 2021). In addition, we perform node property

Model	Cora	Citeseer	Pubmed	AmzPhoto	AmzComp	Chameleon	Squirrel	ogb-arxiv	ogb-papers100M
GCN	$0.808_{\pm 0.03}$	$0.707_{\pm 0.04}$	$0.785_{\pm 0.01}$	$0.907_{\pm 0.03}$	$0.821_{\pm 0.01}$	$0.381_{\pm 0.04}$	$0.311_{\pm 0.01}$	$0.714_{\pm 0.00}$	$0.733_{\pm 0.00}$
GCN*	$\underline{\mathbf{0.837}}_{\pm 0.07}$	$\mathbf{0.723}_{\pm 0.06}$	$\underline{\mathbf{0.801}}_{\pm 0.03}$	$\underline{\mathbf{0.929}}_{\pm 0.04}$	$\underline{\mathbf{0.846}}_{\pm 0.03}$	$\mathbf{0.432}_{\pm 0.09}$	$\underline{\mathbf{0.346}}_{\pm 0.08}$	$\underline{\mathbf{0.729}}_{\pm 0.00}$	$\underline{\mathbf{0.760}}_{\pm 0.00}$
APPNP	$0.824_{\pm 0.03}$	$0.715_{\pm 0.06}$	$0.791_{\pm 0.02}$	$0.914_{\pm 0.02}$	$0.817_{\pm 0.02}$	$0.317_{\pm 0.02}$	$0.240_{\pm 0.01}$	$0.680_{\pm 0.00}$	$0.637_{\pm 0.00}$
APPNP*	$\mathbf{0.833}_{\pm 0.05}$	$\mathbf{0.724}_{\pm 0.05}$	$\mathbf{0.800}_{\pm 0.02}$	$\underline{\mathbf{0.926}}_{\pm 0.03}$	$\underline{\mathbf{0.832}}_{\pm 0.03}$	$\underline{\mathbf{0.366}}_{\pm 0.08}$	$\underline{\mathbf{0.282}}_{\pm 0.05}$	$\underline{\mathbf{0.704}}_{\pm 0.00}$	$\underline{\mathbf{0.681}}_{\pm 0.00}$
GDC$_{\text{HK}}$	$0.826_{\pm 0.02}$	$0.718_{\pm 0.03}$	$0.792_{\pm 0.02}$	$0.920_{\pm 0.02}$	$0.832_{\pm 0.03}$	$0.335_{\pm 0.01}$	$0.262_{\pm 0.02}$	$0.679_{\pm 0.00}$	$0.667_{\pm 0.00}$
GDC$_{\text{HK}}^\star$	$\mathbf{0.835}_{\pm 0.04}$	$\mathbf{0.726}_{\pm 0.04}$	$\mathbf{0.797}_{\pm 0.04}$	$\mathbf{0.928}_{\pm 0.05}$	$\underline{\mathbf{0.850}}_{\pm 0.05}$	$\underline{\mathbf{0.372}}_{\pm 0.08}$	$\underline{\mathbf{0.302}}_{\pm 0.06}$	$\underline{\mathbf{0.692}}_{\pm 0.00}$	$\underline{\mathbf{0.671}}_{\pm 0.00}$
GDC$_{\text{PPR}}$	$0.824_{\pm 0.02}$	$0.720_{\pm 0.02}$	$0.789_{\pm 0.01}$	$0.910_{\pm 0.02}$	$0.830_{\pm 0.03}$	$0.330_{\pm 0.02}$	$0.264_{\pm 0.01}$	$0.677_{\pm 0.00}$	$0.650_{\pm 0.00}$
GDC$_{\text{PPR}}^\star$	$\underline{\mathbf{0.837}}_{\pm 0.04}$	$\mathbf{0.725}_{\pm 0.03}$	$\underline{\mathbf{0.798}}_{\pm 0.03}$	$\underline{\mathbf{0.925}}_{\pm 0.05}$	$\mathbf{0.843}_{\pm 0.06}$	$\underline{\mathbf{0.377}}_{\pm 0.05}$	$\underline{\mathbf{0.309}}_{\pm 0.06}$	$\underline{\mathbf{0.695}}_{\pm 0.00}$	$\underline{\mathbf{0.688}}_{\pm 0.00}$
SGC	$0.783_{\pm 0.01}$	$0.700_{\pm 0.02}$	$0.753_{\pm 0.01}$	$0.869_{\pm 0.02}$	$0.808_{\pm 0.01}$	$0.287_{\pm 0.01}$	$0.231_{\pm 0.02}$	$0.696_{\pm 0.00}$	$0.660_{\pm 0.00}$
SGC*	$\underline{\mathbf{0.827}}_{\pm 0.02}$	$\underline{\mathbf{0.719}}_{\pm 0.02}$	$\underline{\mathbf{0.790}}_{\pm 0.04}$	$\underline{\mathbf{0.915}}_{\pm 0.02}$	$\underline{\mathbf{0.827}}_{\pm 0.03}$	$\underline{\mathbf{0.334}}_{\pm 0.04}$	$\underline{\mathbf{0.269}}_{\pm 0.05}$	$\underline{\mathbf{0.712}}_{\pm 0.00}$	$\underline{\mathbf{0.679}}_{\pm 0.00}$
S^2GC	$0.829_{\pm 0.03}$	$0.718_{\pm 0.03}$	$0.795_{\pm 0.01}$	$0.919_{\pm 0.04}$	$0.829_{\pm 0.03}$	$0.398_{\pm 0.02}$	$0.312_{\pm 0.01}$	$0.707_{\pm 0.00}$	$0.715_{\pm 0.00}$
S^2GC*	$\underline{\mathbf{0.841}}_{\pm 0.04}$	$\mathbf{0.724}_{\pm 0.05}$	$\underline{\mathbf{0.803}}_{\pm 0.03}$	$\mathbf{0.928}_{\pm 0.05}$	$\underline{\mathbf{0.847}}_{\pm 0.04}$	$\underline{\mathbf{0.435}}_{\pm 0.06}$	$\underline{\mathbf{0.352}}_{\pm 0.06}$	$\underline{\mathbf{0.726}}_{\pm 0.00}$	$\underline{\mathbf{0.753}}_{\pm 0.00}$
JacobiConv	$0.827_{\pm 0.01}$	$0.722_{\pm 0.01}$	$0.799_{\pm 0.01}$	$0.924_{\pm 0.03}$	$0.838_{\pm 0.02}$	$0.423_{\pm 0.02}$	$0.328_{\pm 0.02}$	$0.718_{\pm 0.00}$	$0.722_{\pm 0.00}$
JacobiConv*	$\underline{\mathbf{0.841}}_{\pm 0.09}$	$\mathbf{0.729}_{\pm 0.05}$	$\underline{\mathbf{0.807}}_{\pm 0.02}$	$\mathbf{0.928}_{\pm 0.06}$	$\mathbf{0.849}_{\pm 0.04}$	$\underline{\mathbf{0.469}}_{\pm 0.06}$	$\mathbf{0.351}_{\pm 0.07}$	$\underline{\mathbf{0.730}}_{\pm 0.00}$	$\underline{\mathbf{0.759}}_{\pm 0.00}$
GPRGNN	$0.821_{\pm 0.01}$	$0.692_{\pm 0.01}$	$0.792_{\pm 0.02}$	$0.917_{\pm 0.02}$	$0.824_{\pm 0.01}$	$0.348_{\pm 0.02}$	$0.243_{\pm 0.02}$	$0.711_{\pm 0.00}$	$0.654_{\pm 0.00}$
GPRGNN*	$\underline{\mathbf{0.843}}_{\pm 0.08}$	$\underline{\mathbf{0.728}}_{\pm 0.07}$	$\underline{\mathbf{0.809}}_{\pm 0.04}$	$\underline{\mathbf{0.931}}_{\pm 0.08}$	$\underline{\mathbf{0.847}}_{\pm 0.05}$	$\mathbf{0.379}_{\pm 0.06}$	$\underline{\mathbf{0.288}}_{\pm 0.07}$	$\underline{\mathbf{0.720}}_{\pm 0.00}$	$\underline{\mathbf{0.668}}_{\pm 0.00}$
ChebNetII	$0.822_{\pm 0.01}$	$0.696_{\pm 0.01}$	$0.791_{\pm 0.01}$	$0.908_{\pm 0.03}$	$0.815_{\pm 0.03}$	$0.430_{\pm 0.04}$	$0.336_{\pm 0.01}$	$0.720_{\pm 0.00}$	$0.670_{\pm 0.00}$
ChebNetII*	$\underline{\mathbf{0.842}}_{\pm 0.09}$	$\underline{\mathbf{0.727}}_{\pm 0.04}$	$\underline{\mathbf{0.807}}_{\pm 0.03}$	$\underline{\mathbf{0.923}}_{\pm 0.06}$	$\underline{\mathbf{0.848}}_{\pm 0.05}$	$\mathbf{0.466}_{\pm 0.07}$	$\mathbf{0.357}_{\pm 0.09}$	$\underline{\mathbf{0.733}}_{\pm 0.00}$	$\underline{\mathbf{0.691}}_{\pm 0.00}$

Table 2: The semi-supervised node classification results (random splits and averaged over 10 runs) and node property prediction results (public splits and averaged over 10 runs). Starred models (*) are C^{3}E-estimated baselines; **bold** fonts mark better average performance and underlines mark statistical significance (t-test, $p < 0.05$).

prediction on two large-scale graphs: ogb-arxiv (Hu et al. 2020) and ogb-papers100M (Hu et al. 2020). The hyperparameter configurations are provided in Appendix E.

Performance Evaluation

Figure 1 and Table 2 illustrate that C^{3}E-estimated optimal models consistently outperform baselines using heuristic configurations across all scenarios. Figure 1 visually confirms this: we see that many heuristic solutions (green dots) achieve scattered and suboptimal performance, whereas C^{3}E-estimated solutions reach optimal performance regions. This empirically validates that optimal performance is achieved when ϕ_0 falls within bounds defined in Eq. (12), and that simply scaling up parameters does not guarantee better results. Meanwhile, these results verify that while C^{3}E is formulated based on the principle of maximum entropy for pre-training estimation, C^{3}E-estimated models learn stable representations, in contrast to ill-behaved representations in naively configured models (see Appendix I for post-training analysis). Furthermore, unlike trial-and-error, which can take between 2.45 hours to 120 hours (Cai et al. 2021), our method generates solutions within seconds (3.7 seconds to 879.6 seconds) as shown in Table 1.

Representation Compression Ratio

As shown in Table 3, the representation compression ratio θ monotonically rises as the propagation depth L increases. Increasing propagation depth consistently compresses information in the encoded representation, which aligns with previous empirical findings (Saxe et al. 2019; Loukas 2020a; Di Giovanni et al. 2023) and Corollary 2. The C^{3}E-estimated baseline achieves optimal performance when $\theta = 0.010$ and $L = 4$. Conversely, plain baselines consistently degrade as θ increases beyond their minima ($\theta = 0.558$ when $L = 1$). These changes in model performance and representation compression ratio imply that over-squashing arises from cumulative information compression (increasing θ). First, larger hidden dimensions reduce θ, alleviating the information compression; however, overly small θ ($\overline{w}$ relatively larger) results in the encoded representation matrix becoming overly informative, and overestimates the latent distribution's complexity. In such cases, increasing propagation depth L is not detrimental, facilitating compression of high-dimensional representations into appropriately parameterized lower-dimensional ones. In general, these results empirically verify Corollary 2 with further supportive visualizations provided in Appendix H.

L	$\overline{w}$	θ	GCN	$\overline{w}$	θ	GCN*
1	16	0.558	**0.707**	32765	0.000	-
2	16	0.856	0.663	3960.89	0.004	0.716
3	16	1.155	0.624	3453.01	0.007	0.717
4	16	1.450	0.612	3203.64	0.010	**0.723**
5	16	1.750	0.605	2975.99	0.014	0.715
6	16	2.050	0.572	2808.91	0.017	0.713
7	16	2.347	0.550	2399.07	0.021	0.714

Table 3: Comparison results between the baseline and C^{3}E-estimated baseline on Citeseer ($n = 3312$, $m = 3703$). Here, the empty cell denotes no valid solution.

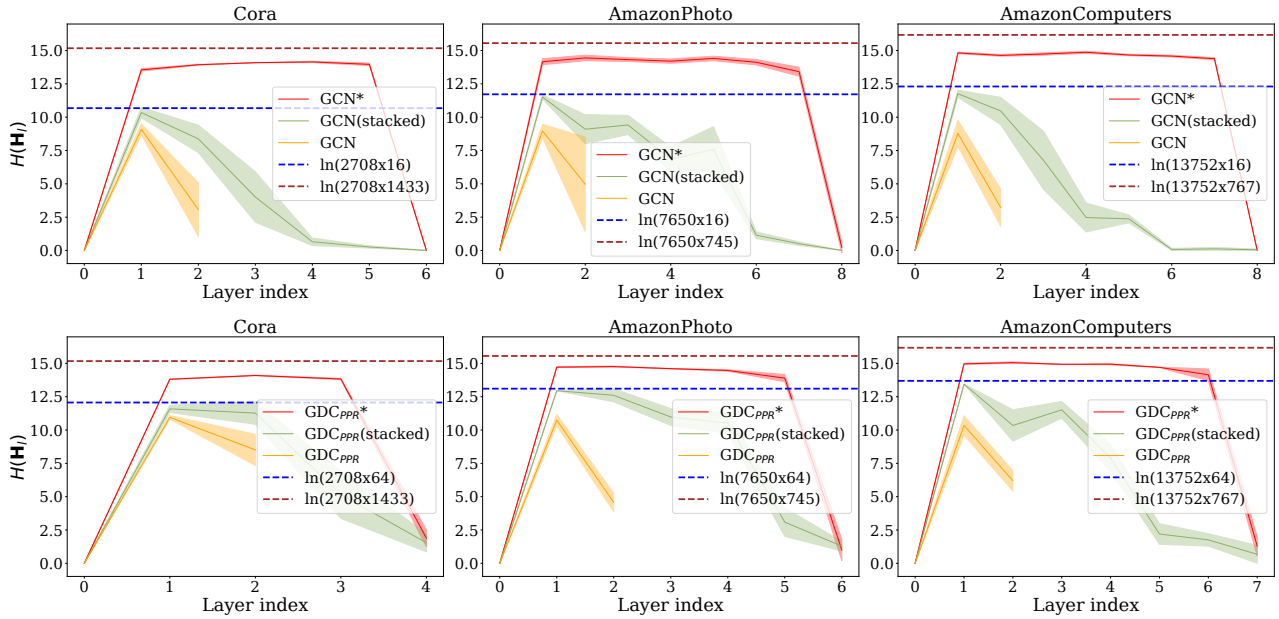

Figure 2: C^{3}E-estimated models (red lines) avoid information loss by maintaining high $H(\mathbf{H}_l)$ across layers. In contrast, naively stacked baselines (green lines) suffer from over-squashing, with entropy collapsing to near-zero long before the final layer. Dotted lines indicate the maximum entropy for initial feature dimensions (brown) and fixed hidden dimensions (blue).

Entropy Transitions in Representation Matrix

This section illustrates the layer-wise representation entropy transitions of original baselines, C^{3}E-estimated baselines, and naively stacked baselines. Based on previous analysis, over-squashing originates from cumulative growth in the information compression. Consequently, this should result in entropy reductions in encoded representation matrices. Figure 2 further visualizes the consequence of cumulative information compression on the network's information content. The entropy of naively stacked baselines (green lines) quickly collapses during propagation, demonstrating catastrophic information loss. This visual trend is the direct result of the runaway representation compression ratio (same in Table 3), where rising compression chokes off information flow. In other words, effective representation learning terminates early, causing encoded representation matrices to fail to preserve information about the latent distribution. In contrast, C^{3}E-estimated models maintain consistently high entropy representation learning, successfully preventing information loss and enabling effective representation learning during the information propagation. In addition, we can observe that the representation entropy of baselines using 16 or 64 as hidden dimensions is strictly bounded by their dimensionalities, as they do not exceed the blue dotted lines. This observation reiterates that the information conveyed in representation matrices is capped by their dimensions.

Conclusion

This paper presents Channel Capacity Constrained Estimation (C^{3}E), an information-theoretic framework for estimating the hidden dimensions and depth for GNNs to mitigate over-squashing. Our analysis and results lead to three primary conclusions. First, through the lens of information theory, we show that over-squashing is a direct consequence of cumulative information compression within representation matrices. Second, ensuring hidden dimensions are sufficiently large can effectively mitigate such information compression. Third, the role of network depth is nuanced: for networks with a low representation compression ratio, deeper propagation is beneficial, helping concentrate high-dimensional signals to lower dimensions; for those with a high ratio, it aggravates information compression and leads to information loss.

Despite the promising results, we acknowledge the limitations of this work. First, the framework is grounded in the principle of maximum entropy, which may overestimate what practical networks can achieve but provides a useful theoretical ceiling before training. By Jaynes, the max-entropy prior is independent Gaussian, yielding a safe pre-training bound; we also discuss its pros/cons and higher-order correlations (see Appendix A). Second, though theoretical derivations are established with analytically tractable spectral GNNs, the results uncovered provide a critical basis for future extensions to more complex GNN architectures. Besides these, we note that scaling effectively to larger architectures hinges on better optimization techniques (see Appendix B.3) as well. The primary focus for future research is to generalize C^{3}E to spatial GNNs that depend on learned features for propagation, such as Graph Transformers. Moreover, we plan to tighten the entropy upper bounds with large-scale empirical results to reflect the practical trained GNNs more closely.

Acknowledgments

This work was supported by UK Research and Innovation (UKRI) Engineering and Physical Sciences Research Council (EPSRC) Grant Number EP/Y028392/1: AI for Collective Intelligence (AI4CI).

References

Achille, A.; and Soatto, S. 2018. Emergence of invariance and disentanglement in deep representations. *Journal of Machine Learning Research*, 19(50): 1–34.

Alon, U.; and Yahav, E. 2020. On the bottleneck of graph neural networks and its practical implications. *arXiv preprint arXiv:2006.05205*.

Bodnar, C.; Di Giovanni, F.; Chamberlain, B.; Lio, P.; and Bronstein, M. 2022. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in gnns. *Advances in Neural Information Processing Systems*, 35: 18527–18541.

Bruna, J.; Zaremba, W.; Szlam, A.; and LeCun, Y. 2014. Spectral networks and deep locally connected networks on graphs. In *2nd International Conference on Learning Representations, ICLR 2014*.

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

Cai, S.; Li, L.; Deng, J.; Zhang, B.; Zha, Z.-J.; Su, L.; and Huang, Q. 2021. Rethinking graph neural architecture search from message-passing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6657–6666.

Chan, K. H. R.; Yu, Y.; You, C.; Qi, H.; Wright, J.; and Ma, Y. 2022. Redunet: A white-box deep network from the principle of maximizing rate reduction. *Journal of machine learning research*, 23(114): 1–103.

Chen, D.; Lin, Y.; Li, W.; Li, P.; Zhou, J.; and Sun, X. 2020. 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.

Chien, E.; Peng, J.; Li, P.; and Milenkovic, O. 2020. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*.

Cong, W.; Ramezani, M.; and Mahdavi, M. 2021. On provable benefits of depth in training graph convolutional networks. *Advances in Neural Information Processing Systems*, 34: 9936–9949.

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.

Dehmer, M.; and Mowshowitz, A. 2011. A history of graph entropy measures. *Information Sciences*, 181(1): 57–78.

Di Giovanni, F.; Giusti, L.; Barbero, F.; Luise, G.; Lio, P.; and Bronstein, M. M. 2023. On over-squashing in message passing neural networks: The impact of width, depth, and topology. In *International Conference on Machine Learning*, 7865–7885. PMLR.

Dwivedi, V. P.; Joshi, C. K.; Luu, A. T.; Laurent, T.; Bengio, Y.; and Bresson, X. 2023. Benchmarking graph neural networks. *Journal of Machine Learning Research*, 24(43): 1–48.

Gallager, R. G. 1968. *Information theory and reliable communication*, volume 588. Springer.

Gasteiger, J.; Bojchevski, A.; and Günnemann, S. 2018. Predict then Propagate: Graph Neural Networks meet Personalized PageRank. In *International Conference on Learning Representations*.

Gasteiger, J.; Weißenberger, S.; and Günnemann, S. 2019. Diffusion improves graph learning. *Advances in neural information processing systems*, 32.

Gravina, A.; Eliasof, M.; Gallicchio, C.; Bacciu, D.; and Schönlieb, C.-B. 2025. On oversquashing in graph neural networks through the lens of dynamical systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39(16), 16906–16914.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

He, M.; Wei, Z.; and Wen, J.-R. 2022. Convolutional neural networks on graphs with chebyshev approximation, revisited. *Advances in neural information processing systems*, 35: 7264–7276.

He, M.; Wei, Z.; Xu, H.; et al. 2021. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in Neural Information Processing Systems*, 34: 14239–14251.

Henaff, M.; Bruna, J.; and LeCun, Y. 2015. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*.

Hu, W.; Fey, M.; Zitnik, M.; Dong, Y.; Ren, H.; Liu, B.; Catasta, M.; and Leskovec, J. 2020. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33: 22118–22133.

Huang, K.; Wang, Y. G.; Li, M.; et al. 2024. How Universal Polynomial Bases Enhance Spectral Graph Neural Networks: Heterophily, Over-smoothing, and Over-squashing. *arXiv preprint arXiv:2405.12474*.

Jaynes, E. 2003. *Probability Theory: The Logic of Science*, volume 727. Cambridge University Press.

Jaynes, E. T. 1957. Information theory and statistical mechanics. *Physical review*, 106(4): 620.

Karhadkar, K.; Banerjee, P. K.; and Montúfar, G. 2022. FoSR: First-order spectral rewiring for addressing oversquashing in GNNs. *arXiv preprint arXiv:2210.11790*.

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

Kraft, D. 1988. A software package for sequential quadratic programming. *Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt*.

Kullback, S. 1997. *Information theory and statistics*. Courier Corporation.

Lee, J.; Lee, I.; and Kang, J. 2019. Self-attention graph pooling. In *International conference on machine learning*, 3734–3743. pmlr.

Loukas, A. 2020a. How hard is to distinguish graphs with graph neural networks? *Advances in neural information processing systems*, 33: 3465–3476.

Loukas, A. 2020b. What graph neural networks cannot learn: depth vs width. In *International Conference on Learning Representations*.

Maskey, S.; Paolino, R.; Bacho, A.; and Kutyniok, G. 2024. A fractional graph laplacian approach to oversmoothing. *Advances in Neural Information Processing Systems*, 36.

Nt, H.; and Maehara, T. 2019. Revisiting graph neural networks: All we have is low-pass filters. *arXiv preprint arXiv:1905.09550*.

Pellizzoni, P.; Schulz, T. H.; Chen, D.; and Borgwardt, K. 2024. On the expressivity and sample complexity of node-individualized graph neural networks. *Advances in Neural Information Processing Systems*, 37: 120221–120251.

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.

Roberts, D. A.; Yaida, S.; and Hanin, B. 2022. *The principles of deep learning theory*, volume 46. Cambridge University Press Cambridge, MA, USA.

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

Saxe, A. M.; Bansal, Y.; Dapello, J.; Advani, M.; Kolchinsky, A.; Tracey, B. D.; and Cox, D. D. 2019. On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12): 124020.

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.

Shannon, C. E. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3): 379–423.

Shchur, O.; Mumme, M.; Bojchevski, A.; and Günnemann, S. 2018. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*.

Shen, X.; Wang, Y.; Lin, M.; Huang, Y.; Tang, H.; Sun, X.; and Wang, Y. 2023. Deepmad: Mathematical architecture design for deep convolutional neural network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6163–6173.

Sun, Z.; Lin, M.; Sun, X.; Tan, Z.; Li, H.; and Jin, R. 2021. Mae-det: Revisiting maximum entropy principle in zero-shot nas for efficient object detection. *arXiv preprint arXiv:2111.13336*.

Tishby, N.; Pereira, F. C.; and Bialek, W. 2000. The information bottleneck method. *arXiv preprint physics/0004057*.

Topping, J.; Di Giovanni, F.; Chamberlain, B. P.; Dong, X.; and Bronstein, M. M. 2021. Understanding over-squashing and bottlenecks on graphs via curvature. *arXiv preprint arXiv:2111.14522*.

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

Wang, X.; and Zhang, M. 2022. How powerful are spectral graph neural networks. In *International conference on machine learning*, 23341–23362. PMLR.

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, T.; Ren, H.; Li, P.; and Leskovec, J. 2020. Graph information bottleneck. *Advances in Neural Information Processing Systems*, 33: 20437–20448.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018a. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*.

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

Yang, C.; Wang, R.; Yao, S.; Liu, S.; and Abdelzaher, T. 2020. Revisiting over-smoothing in deep GCNs. *arXiv preprint arXiv:2003.13663*.

Yang, Z.; Zhang, G.; Wu, J.; Yang, J.; Sheng, Q. Z.; Peng, H.; Li, A.; Xue, S.; and Su, J. 2023. Minimum entropy principle guided graph neural networks. In *Proceedings of the sixteenth ACM international conference on web search and data mining*, 114–122.

Zhou, K.; Dong, Y.; Wang, K.; Lee, W. S.; Hooi, B.; Xu, H.; and Feng, J. 2021. Understanding and resolving performance degradation in deep graph convolutional networks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2728–2737.

Zhu, H.; and Koniusz, P. 2021. Simple spectral graph convolution. In *International conference on learning representations*.