Entropy Guided Semi-Supervised Graph Coarsening

Anonymous authors
Paper under double-blind review

Abstract

Graphs are foundational abstractions in data-intensive domains, yet the scale of modern datasets strains computation and memory for downstream learning. From recommender systems to biological networks, graphs have emerged as a fundamental substrate for learning. As graph sizes grow, the cost of training and inference becomes prohibitive, thereby necessitating compact surrogates that retain spectral properties and feature semantics. We propose an entropy-regularized, semi-supervised framework for attributed graph coarsening that jointly leverages the original graph's Laplacian, node features, and partially observed labels. Central to our approach is an information-theoretic regularizer that minimizes the per-supernode Shannon entropy of the node-profile matrix $\phi = C^T Y$, encouraging label-coherent aggregations. We formulate a principled objective that balances structural fidelity and feature alignment, and we solve it using an efficient block MM/BSUM algorithm. We establish learning and show structural guarantees that control Dirichlet-energy distortion, preserve low-order spectral moments, and bound deviations in cut costs and effective resistances. Experiments on standard benchmarks (e.g., Cora, Citeseer, PubMed, coauthor-CS) demonstrate that our method produces a node-profile matrix with low row-wise entropy, in which nodes with the same label are grouped into the same supernode, and achieves competitive or superior node-classification accuracy and link-prediction performance across multiple GNN backbones (GCN, GAT, APPNP), while remaining computationally efficient.

1 Introduction

Graphs are fundamental mathematical structures that effectively model relationships and interactions across a wide range of domains, such as social networks, biology, chemistry, and computer vision (Battaglia et al., 2018; Wu et al., 2020; Zhou et al., 2020; Bruna et al., 2013; Chen et al., 2020; Defferrard et al., 2016; Kumar et al., 2020). However, as real-world graph datasets continue to grow in size and complexity, they present significant challenges in terms of storage requirements, computational efficiency, and execution of downstream tasks such as node classification, graph classification, link prediction, etc. To address these challenges, graph dimensionality reduction techniques, such as graph coarsening (Loukas & Vandergheynst, 2018; Loukas, 2019; Kumar et al., 2023a;b;a; Dorfler & Bullo, 2011; Ron et al., 2010; Hendrickson et al., 1995), condensation (Jin et al., 2021), or graph summarization (Riondato et al., 2017) have emerged. The primary goal of graph coarsening (Hashemi et al., 2024) is to construct a smaller, tractable representation of the original graph while preserving its essential properties. This enables downstream tasks to be performed efficiently without compromising the quality of results, as well as requiring less memory to store the data.
Various methods have been proposed for graph coarsening. For instance, (Loukas & Vandergheynst, 2018; Loukas, 2019) are a heuristic-based approach that considers the Laplacian matrix of the original graph to learn a coarsened graph. Optimization-based techniques, such as (Kumar et al., 2024; 2023a;b), incorporate both the Laplacian and feature matrix to learn a coarsened representation. Additionally, (Jin et al., 2021) introduces a deep learning-based method that leverages graph neural networks for coarsening. On the other hand, graph condensation is a related technique that aims to learn a synthetic graph tailored for specific downstream tasks, often utilizing gradient-matching strategies to optimize the graph representation.

Most graph coarsening methods rely on the graph Laplacian, with or without node features, and infer labels on the coarsened graph via majority voting within supernodes. While label-informed coarsening approaches

exist, they are largely deep learning–based and require expensive training on the original large graph through gradient matching or bi-level optimization, thereby failing to eliminate the underlying computational bottleneck and offering limited flexibility. Moreover, these approaches are typically tightly coupled to specific architectures or tasks, limiting their flexibility and general applicability.

Optimization-based coarsening frameworks are more scalable and interpretable but typically ignore label structure, focusing instead on sparsity in the node profile matrix. Such sparsity alone does not guarantee label purity within supernodes. In this work, we incorporate label information directly into the coarsening process by regularizing the entropy of the node profile matrix (Section 2.3, (Ghoroghchian et al., 2021)). Unlike Frobenius-norm regularization (Kumar et al., 2024), entropy minimization explicitly enforces row-wise concentration. This objective encourages nodes with the same label to be assigned to the same supernode, thereby yielding coarsenings that are both computationally efficient and better aligned with downstream classification.

Moreover, prior label-aware coarsening methods (Kumar et al., 2024) did not provide explicit guarantees on the extent to which the coarse graph, when lifted back to the original node space, preserves key structural quantities. Unlike Frobenius penalties that encourage a small Frobenius norm $\|\phi\|_F^2$, our entropy term acts on the normalized row-wise label distributions ϕ_i/N_i, thereby directly penalizing label mixing within each supernode. Concretely, minimizing $\sum_i H\left(\frac{\phi_i}{\|\phi_i\|_1}\right)$ promotes the merging of nodes with the same label into the same supernode and provides a transparent *regularization parameter* to balance supervision against coarsening strength; by contrast, quadratic penalties may still yield supernodes containing nodes from different labels and with small ℓ_2 error due to majority-class dominance. Finally, we provide such guarantees by deriving three bounds tied to the mapping from original nodes to supernodes: (i) an upper bound on the difference in Dirichlet energy between the original and the lifted coarse graphs, certifying that the lifted coarse Laplacian remains close to the original; (ii) an upper bound on cut preservation, formalizing the view of coarsening as clustering so that cut costs are not overly distorted; and (iii) an upper bound on the deviation of the Laplacian and its pseudoinverse after lifting, which controls how far the lifted operator can drift from the original.
Entropy drives the formation of each supernode, which consists of nodes that share the same label, thereby reducing the projection error and improving the stability/conditioning of the mapping. We show that the error term becomes smaller under an entropy regularizer than under a Frobenius norm regularizer, and the resulting relative cut distortion is correspondingly lower. By contrast, a Frobenius regularizer tends to spread each supernode across many labels, mixing classes and increasing the projection error, thereby yielding looser guarantees. Furthermore, the main contribution of this paper are: **(i) Entropy–Based Regularizer**: We develop a framework that uses an entropy regularizer to improve downstream performance. **(ii) Theoretical guarantees**: We theoretically establish that entropy regularization produces sparser rows of the node profile matrix in contrast to Frobenius norm-based regularization. Beyond this, we derive an explicit upper bound on the Dirichlet energy difference between the original Laplacian and the lifted Laplacian. This also leads to bounds on cut preservation and effective resistance. Finally, we show how entropy controls misclassification. We next establish a direct connection between entropy regularization and label homogeneity within supernodes. **(iii) Evaluating Label mixing:** introduce three metrics, SII (Supernode Impurity Index) to count how many different labeled nodes are clustered into a supernode, AMM(Average Majority Miss) is the average error if we label each supernode by its majority class, and EIEC(Entropy-Induced Error Ceiling) as an entropy-based upper bound to check consistency with theory.

1.1 Outline

The paper is organized as follows: in Section 2, we present foundational background information covering graphs, graph learning from data, and graph coarsening techniques. Additionally, we introduce the proposed ENGC formulation in this section. Section 3 is dedicated to the development of our algorithm. Finally, in Section 6, we present the outcomes of our experiments conducted on real-world datasets. Additional experiments, proofs of the theoretical guarantees, and the notation are provided in the supplementary material.

2 BACKGROUND AND PROBLEM FORMULATION

In this section, we cover the fundamentals of graphs, graph coarsening, node profile matrix, and entropy. Furthermore, we provide details on how entropy is used for grouping and aggregating nodes.

2.1 Graphs and Graph Learning from Data

We consider an undirected graph $\mathcal{G} = (V, E, A, X, Y)$, where $V = \{v^1, \ldots, v^p\}$ is the set of nodes, $E \subseteq V \times V$ is the set of edges, and $A \in \mathbb{R}^{p\times p}$ is the adjacency matrix, with $A_{ij} \neq 0$ indicating an edge between nodes i and j. Each node v_i is associated with a feature vector $X_i \in \mathbb{R}^f$, and the feature matrix is denoted by $X = [X_1, \ldots, X_p]^T \in \mathbb{R}^{p\times f}$. In the semi-supervised setting, label information is given by $Y \in \{0, 1\}^{p\times l}$, where a labeled node has a one-hot label vector and an unlabeled node has a zero vector. Graphs are commonly represented using either the adjacency matrix or the combinatorial Laplacian matrix $\Theta \in \mathbb{R}^{p\times p}$, defined as $\Theta_{ij} = \Theta_{ji} \leq 0$ for $i \neq j$ and $\Theta_{ii} = -\sum_{j\neq i} \Theta_{ij}$ (Kumar et al., 2020), with the relation $A_{ij} = -\Theta_{ij}$ for $i \neq j$. The Laplacian matrix is symmetric, positive semidefinite, and has zero row sums, which makes it convenient for graph learning and coarsening. Given node features X, we learn a connected graph Laplacian by solving the optimization problem in (Kalofolias, 2016):

$$\underset{\Theta\in\mathcal{S}_\Theta}{\text{minimize}} -\gamma \log(\det(\Theta + J)) + \operatorname{tr}(X^T\Theta X) + \beta h(\Theta) \tag{1}$$

which balances feature smoothness, structural regularization, and graph connectivity through a log-determinant term. For graph coarsening, we define the coarse Laplacian as $\Theta_c = C^\top \Theta C$ and the lifted Laplacian as $\widetilde{\Theta} = C\Theta_c C^\top$, which will be used in the following sections.

2.2 Graph Coarsening

Given a graph $\mathcal{G}(\Theta, X, Y)$ with p nodes, graph coarsening constructs a reduced graph $\mathcal{G}_c(\Theta_c, \tilde{X}, \tilde{Y})$ with $k \ll p$ nodes while preserving structural and spectral properties. The coarse Laplacian $\Theta_c \in \mathbb{R}^{k\times k}$, feature matrix $\tilde{X} \in \mathbb{R}^{k\times f}$, and label matrix $Y_c \in \mathbb{R}^{k\times l}$ are related to their counterparts by:$\Theta_c = C^\top \Theta C, \quad X = C\tilde{X}, \quad Y_c = \arg\max(C^T Y)$, where $C \in \mathbb{R}_+^{p\times k}$ is a node-to-supernode mapping matrix. To ensure valid aggregation, C is constrained to

$$\mathcal{C} = \Big\{ C \geq 0 \;\Big|\; \langle C_i, C_j\rangle = 0 \;\forall i \neq j,\; \langle C_i, C_i\rangle = d_i, \|C_i\|_0 \geq 1,\; \|[C^\top]_i\|_0 = 1 \Big\}, \tag{2}$$

which enforces non-overlapping assignments and prevents a node from belonging to two different supernodes.

We define the reconstruction error matrix as

$$\Delta := \Theta - C\Theta_c C^\top, \tag{3}$$

so that $x^\top \Delta x$ quantifies the error for any $x \in \mathbb{R}^p$.

2.3 Node Profile Matrix

We assess the quality of the coarsened graph in terms of semantic fidelity using the node profile matrix, which has been widely adopted in the prior work (Kumar et al., 2024; Ghoroghchian et al., 2021). Given a mapping matrix $C \in \mathbb{R}^{p\times k}$, where $C_{vk} = 1$ if node v is assigned to supernode k, the node profile matrix is defined as

$$\phi = C^T Y \tag{4}$$

Each entry ϕ_{ij} represents the number of original nodes with label j that are mapped to the i-th supernode in G_c. Each row of ϕ captures the empirical label distribution induced by assigning original nodes to a supernode. While several existing coarsening methods implicitly aim to control the structure of ϕ, many rely on objectives such as $\|C^T Y\|_F^2$ (Kumar et al., 2024) , which primarily penalize the magnitude of entries rather than their distribution. Since these objectives do not directly penalize label mixing, which can result in mixed-label supernodes, even though the Frobenius norm remains small.

We introduce the probabilistic node-profile matrix ϕ', where each entry in ϕ' is defined as:

$$[\phi']_{ij} = \frac{\phi_{ij}}{\|[\phi^T]_i\|_1} \tag{5}$$

$[\phi']_{ij}$ denotes the probability that a node assigned to the i-th supernode has label j. $[\phi']_{ij}$ can be interpreted as the conditional probability $[\phi']_{ij} = \mathbb{P}(Y = j \mid \text{node assigned to supernode } i)$

In the context of the original graph shown in Figure 1, there are two coarsened graphs, associated with mapping matrices C_1 and C_2, and the corresponding probabilistic node-profile matrices ϕ'_1 and ϕ'_2 are:

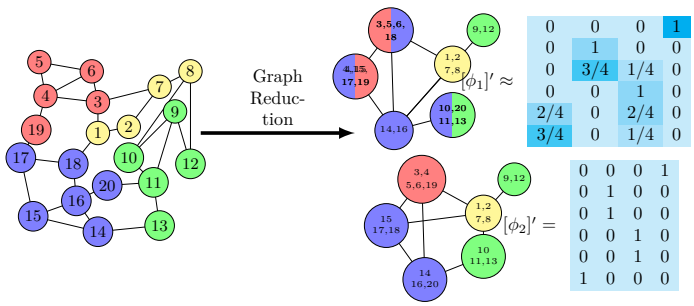

Figure 1: This figure illustrates that, for a given original large graph $\mathcal{G}$ on the left side, there exist numerous possibilities for the coarsened graph. Notably, the coarsened graph $\mathcal{G}_{c2}$ is more informative because nodes with similar labels are mapped to the same supernode, making it particularly well-suited for downstream tasks.

The probabilistic node profile matrix ϕ' therefore provides a principled way to quantify how nodes with different labels are distributed within each supernode. Since semantic fidelity requires these distributions to be as concentrated as possible, we next employ entropy as an information-theoretic measure to quantify and penalize uncertainty in ϕ'. This motivates the entropy-based formulation introduced in the following section.

The entropy of a discrete random variable $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ with a probability distribution $P = \{p_1, p_2, \ldots, p_n\}$, where p_i represents the probability of each sample x_i, is given by the Shannon entropy $H(X)$. It is defined as:

$$H(X) = -\sum_{i=1}^{n} p_i \log p_i \tag{6}$$

Here, $H(X)$ quantifies the uncertainty or information content associated with the random variable $\mathcal{X}$. The entropy of the i-th row of the matrix ϕ', denoted as $H([(\phi'_i)])$, quantifies the uncertainty associated with the distribution of nodes from the original graph, each labeled $1, 2, \ldots, l$, that are mapped to the i-th supernode. This row, $[(\phi'_i)] = [\phi'_{i1}, \phi'_{i2}, \ldots, \phi'_{il}]$, represents the probabilities ϕ'_{ij} of nodes with the j-th label being mapped to the i-th supernode. The entropy is calculated as $H([(\phi'_i)]) = -\sum_{j=1}^{l} \phi'_{ij} \log \phi'_{ij}$, where l is the number of labels. Lower entropy indicates that nodes with similar labels are more consistently grouped into the same supernode.

As illustrated in Figure1, the probabilistic node-profile matrix corresponding to the two coarsenings $\phi_1^{'}$ and $\phi_2^{'}$ And the corresponding entropy vectors $H(\phi'_1)$ and $H(\phi'_2)$ are:

$$\phi'_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & \frac{3}{4} & \frac{1}{4} & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{3}{4} & 0 & \frac{1}{4} & 0 \end{bmatrix} \quad \phi'_2 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad H(\phi'_1) = \begin{bmatrix} 0 \\ 0 \\ -\left(\frac{3}{4}\log\frac{3}{4} + \frac{1}{4}\log\frac{1}{4}\right) \\ 0 \\ \log 2 \\ -\left(\frac{3}{4}\log\frac{3}{4} + \frac{1}{4}\log\frac{1}{4}\right) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.56 \\ 0 \\ 0.69 \\ 0.56 \end{bmatrix} \quad H(\phi'_2) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$H(\phi'_1)$ contains multiple nonzero entries, indicating that nodes with different labels have been merged into some supernodes, whereas $H(\phi'_2) = 0$ for all rows, implying supernodes in the second graph are composed of

nodes with the same label. Lower row-wise entropy therefore corresponds to merging of nodes sharing the same label and a more informative coarsened graph.

We propose entropy as a structural regularizer to control label mixing within supernodes. By penalizing uncertainty in the aggregated label distribution, entropy regularization promotes label-homogeneous clusters and discourages mixed-label supernodes. In Section 4, we prove how the entropy term affects the coarsening objective and theoretical bounds. We now proceed to the algorithmic development.

3 Proposed Entropy Guided Graph Coarsening Algorithm

In this section we delve into the algorithmic development with the objective of reducing the row-wise entropy of the node profile matrix while preserving semantic and structural properties of the graph. To this end, we adopt an entropy-based regularizer:

$$r_E(C, Y) = -\sum_{i=1}^{k}\sum_{j=1}^{\ell} \phi'_{ij}\ \log_2 \phi'_{ij} \tag{7}$$

which encourages a low-entropy node-profile matrix by reducing the entropy of each row. Now, we formally propose our problem formulation below.

3.1 Proposed Formulation

Given a graph $\mathcal{G}(\Theta \in \mathbb{R}^{p\times p}, X \in \mathbb{R}^{p\times n}, Y)$, where $Y \in \{0,1\}^{p\times l}$, the label matrix Y follows a binary encoding, with $\mathbf{y}_i$: representing the corresponding one-hot indicator vector if node v^i is labeled; otherwise, $\mathbf{y}_i := 0$ for unlabelled nodes in a semi-supervised fashion. The proposed formulation for learning a coarsened graph is as follows:

$$\begin{aligned} &\underset{\Theta_c,\tilde{X},C}{\text{minimize}}\, r_E(C,Y) - \gamma \log\det(\Theta_c + J) + \text{tr}(\tilde{X}^T\Theta_c\tilde{X}) + \beta h(\Theta_c) + \frac{\lambda}{2} g(C) \\ &\text{s.t.}\quad C \geq 0,\ \Theta_c = C^T\Theta C,\ X = C\tilde{X},\ \Theta_c \in \mathcal{S}_\Theta, C \in \mathcal{C} \end{aligned} \tag{8}$$

Where the term $-\log\det(\Theta_c + J)$ ensures the connectedness of the coarsened graph where, $J = \frac{1}{k}\mathbf{1}_{k\times k}$ is a rank-one matrix with each element equals to $\frac{1}{k}$. On the other hand, the original hard constraint $X = C\tilde{X}$ poses challenges in optimization. To address this, we relax $X = C\tilde{X}$ to $\|C\tilde{X} - X\|_F^2$. The term $h(\Theta_c) = \|\Theta_c\|_F^2$ promotes sparsity in the resulting coarsened graph. The term $g(C) = \|C^T\|_{1,2}^2$ensures that each node is assigned to exactly one supernode and that every supernode contains at least one node. Substituting $\Theta_c = C^\top\Theta C$ into Eq.8 reduces the original three-variable optimization problem to an equivalent problem involving only two variables, given by:

$$\underset{\tilde{X},C\in\mathcal{C}}{\text{minimize}} -\sum_{i=1}^{k}\sum_{j=1}^{l}\left(\frac{\phi_{ij}}{\|[\phi^T]_i\|_1}\log\frac{\phi_{ij}}{\|[\phi^T]_i\|_1}\right) + \frac{\lambda}{2}||C^\top||_{1,2}^2 - \gamma\log\det(C^\top\Theta C + J) + \text{tr}(\tilde{X}^\top C^\top\Theta C\tilde{X}) + \frac{\beta}{2}||\Theta_c||_F^2 \tag{9}$$

where, the first term is incorporated to reduce the entropy of each row of ϕ' in the learned coarsened graph. This condition ensures that nodes sharing the same label are consistently mapped to the same supernode, thereby enhancing the coherence and consistency of the mapping process.

The proposed formulation in Eq.9 is a non-convex optimization problem when all variables are considered jointly. We develop an efficient block majorization–minimization (MM)–based method to solve the multi-block non-convex problem in Eq.9. Solving the mazorized function at each itteration using a block successive upper-bound minimization (BSUM) which updates one variable at a time keeping the other fixed.

3.2 Update of *C*

When considering C as a variable and holding $\tilde{X}$ constant, the resulting sub-problem for C is expressed as:

$$\underset{C\in\mathcal{C}}{\text{minimize}}\, f(C) = r_E(C,Y) - \gamma\log\det(C^\top\Theta C + J) + \frac{\lambda}{2}\|C^\top\|_{1,2}^2 + \frac{\alpha}{2}\|C\tilde{X} - X\|_F^2 + \text{tr}(\tilde{X}^\top C^\top\Theta C\tilde{X}) + \frac{\beta}{2}\|C^\top\Theta C\|_F^2 \tag{10}$$

Where, $r_E(C,Y) = -\sum_{i=1}^{k}\sum_{j=1}^{l}\left(\frac{\phi_{ij}}{\|[\phi^T]_i\|_1}\log\frac{\phi_{ij}}{\|[\phi^T]_i\|_1}\right)$.

By using the Taylor series approximation, a majorised function for $f(C)$ at $C^{(t)}$ can be obtained as (Beck & Pan, 2018; Razaviyayn et al., 2012; Sun et al., 2017):

$$g(C|C^{(t)}) = f(C^{(t)}) + (C - C^{(t)})\nabla f(C^{(t)}) + \frac{L}{2}||C - C^{(t)}||^2 \tag{11}$$

where $f(C)$ has an $L-$Lipschitz continuous gradient $L = \max(L_1, L_2, L_3, L_4, L_5, L_6)$ with $L_1, L_2, L_3, L_4, L_5, L_6$ the Lipschitz constants of $-\gamma \log\det(C^T\Theta C + J)$, $\mathrm{tr}(\tilde{X}^T C^T \Theta C \tilde{X})$, $\|C\tilde{X} - X\|_F^2$, $\|C^T\|_{1,2}^2$, $\frac{\beta}{2}\|C^T\Theta C\|_F^2$, $r_E(C, Y)$ respectively. We have given the proof of $r_E(\cdot, Y)$ to be Lipschitz continuous in Appendix A.15. After ignoring the constant term, the majorised problem of Eq.10 is

$$\underset{C \in \mathcal{C}}{\text{minimize}} \quad \frac{1}{2}C^T C - C^T A \tag{12}$$

where $A = C^{(t)} - \frac{1}{L}\nabla f(C^{(t)})$ and

$$\nabla f(C^{(t)}) = -2\gamma\Theta C^{(t)}({C^{(t)}}^{\top\Theta} C^{(t)} + J)^{-1} + \alpha\left(C^{(t)}\tilde{X} - X\right)\tilde{X}^\top + 2\Theta C^{(t)}\tilde{X}\tilde{X}^\top + \lambda C^{(t)}\mathbf{1} + \beta\Theta C C^\top \Theta C + \nabla r_E(C, Y)$$

Where, $\mathbf{1}$ represents an all-ones matrix of dimension $k \times k$ and $\nabla r_E(C, Y) = -Y\left[\frac{1}{\|[\phi^\top]_i\|_1}\left(\log_2\left(\frac{\phi}{\|[\phi^\top]_i\|_1}\right) - \frac{1}{\|[\phi^\top]_i\|_1}\sum_{t=1}^{\ell}\phi_{it}\log_2\left(\frac{\phi_{it}}{\left\|[\phi^\top]_i\right\|_1}\right)\right)\right]^\top$

Lemma 1. *Using the KKT optimality condition we obtain the optimal solution of Eq.12 as*

$$C^{(t+1)} = \left(C^{(t)} - \frac{1}{L}\nabla f\left(C^{(t)}\right)\right)^+ \tag{13}$$

where $(X_{ij})^+ = \max(\frac{X_{ij}}{\|[X^T]_i\|_2}, 0)$ *and* $[X^T]_i$ *is the i-th row of the matrix* X*.*

Proof. The proof is deferred to the appendix A.1 □

3.3 Update of $\tilde{X}$

Fixing C, we obtain the following problem for $\tilde{X}$:

$$\underset{\tilde{X}}{\text{minimize}}\, f(\tilde{X}) = \mathrm{tr}(\tilde{X}^T C^T \Theta C \tilde{X}) + \frac{\alpha}{2}\|C\tilde{X} - X\|_F^2 \tag{14}$$

The optimization problem inEq.14 is strongly convex as $C^T\Theta C$ and $C^T C$ are the positive semi-definite and definite matrices, respectively. The closed form solution of problem Eq.14 can be obtained by setting the gradient to zero, i.e., $2C^T\Theta C\tilde{X} + \alpha C^T(C\tilde{X} - X) = 0$, we get

$$\tilde{X}^{t+1} = \left(\frac{2}{\alpha}C^T\Theta C + C^T C\right)^{-1} C^T X \tag{15}$$

Theorem 1. *The sequence* $\{C^{(t)}, \tilde{X}^{(t)}\}$ *generated by Algorithm refer converges to the set of Karush–Kuhn–Tucker (KKT) points of Problem .*

Proof. The detailed proof is deferred to A.2. □

Algorithm 1 summarizes the implementation of our entropy-guided graph coarsening (ENGC) method. In ENGC, the worst-case per-iteration cost for learning the coarsening matrix $C \in \mathbb{R}^{p\times k}$ is $O(p^2k)$, due to matrix multiplications in the gradient computation. We provide the detailed time-complexity derivation and empirical runtimes in Appendix A.19

4 Structural Spectral and Learning Guarantees

In this section, we establish theoretical guarantees on the structural, spectral, and label-consistency preservation of the proposed entropy-guided coarsening framework.

Algorithm 1: ENGC Algorithm

Input: $\mathcal{G}(X, Y, \Theta), \alpha, \gamma, \lambda, \beta, \delta$

$t \leftarrow 0$;
while *stopping criteria not met* **do**
  Update C^{t+1} and $\tilde{X}^{t+1}$ as in Eq.13 and Eq.15 respectively.
  $t \leftarrow t + 1$;
end

Output: C, Θ_c, and $\tilde{X}$

4.1 Entropy as a Structural Regularizer for Label Homogeneity

Consider a supernode S_i with empirical label distribution $p_i(j) = \frac{n_{ij}}{|S_i|}$, where n_{ij} is the number of nodes in S_i with label j. The probability that two randomly sampled nodes from S_i have different labels is

$$M(p_i) := 1 - \sum_j p_i(j)^2, \tag{16}$$

known as the *Gini impurity*. This quantity measures label mixing within the supernode: $M(p_i) = 0$ corresponds to perfectly homogeneous clusters, while larger values indicate increasing label heterogeneity. Since Shannon entropy ($H(p)$) dominates the Rényi entropy of order 2, we have $H(p) \geq -\log\left(\sum_j p_j^2\right)$ (Renyi, 1961; Cover & Thomas, 2006). Exponentiating both sides yields $\sum_j p_j^2 \geq e^{-H(p)}$ which implies

$$M(p) = 1 - \sum_j p_j^2 \leq 1 - e^{-H(p)} \tag{17}$$

that reducing entropy necessarily bounds and reduces label impurity. In the context of graph coarsening, this indicates that entropy regularization naturally promotes label-homogeneous supernodes by discouraging mixed label distributions within each cluster. The next theorem characterizes the structural behavior induced by the two regularizers.

Theorem 2. *For each i, define $N_i = \sum_j \phi_{ij}$ and the feasible set $S_i = \{\phi_i \in \mathbb{R}_+^\ell : \sum_{j=1}^\ell \phi_{ij} = N_i\}$.*

Consider,

$$E_{\text{ent}}(\phi) = \sum_{i=1}^{k} H(\phi_i/N_i), \qquad E_{\text{fro}}(\phi) = \|\phi\|_F^2.$$

(i) E_{ent} is minimized when each row ϕ_i is one-hot, $\phi_i = N_i e_{j^\star}$, corresponding to perfectly label-homogeneous supernodes.

(ii) E_{fro} is minimized when each row is uniform, $\phi_{ij} = \frac{N_i}{\ell}$ with minimum value $\sum_{i=1}^k N_i^2/\ell$.

Proof. The proof is provided in Appendix A.3. □

Theorem 2 highlights the structural contrast between the two regularizers. Entropy minimization drives each row of ϕ toward a single nonzero entry, grouping nodes with the same label into the same supernode. For unlabeled nodes (rows of Y equal to zero), the entropy term is neutral; their assignments are then determined by the Laplacian and feature terms, typically attaching them to nearby label-consistent supernodes.

In contrast, the Frobenius objective used in LAGC (Kumar et al., 2024) minimizes $\|\phi\|_F^2$, which favors uniform rows in ϕ. This promotes balanced label proportions within supernodes and can therefore lead to mixed-label clusters.

4.2 Controlling Projection Distortion via Label Impurity

In this subsection We quantify how well range(C) represents node labels and show that this representation quality controls Dirichlet-energy, cut, and effective-resistance distortions. We use the Dirichlet energy induced by the Laplacian Θ: $\|\mathbf{u}\|_\Theta^2 := \mathbf{u}^\top \Theta \mathbf{u}$, and its matrix version, summing the Dirichlet energies of all columns $\|U\|_{\Theta,F}^2 := \mathrm{Tr}(U^\top \Theta U)$. Given C, any supernode-level label matrix $Y_c \in \mathbb{R}^{k\times\ell}$ lifts the original graph as $CY_c \in \mathbb{R}^{p\times\ell}$. We then project the label matrix Y onto the coarse space range(C) in this norm:

$$Y_c^\star := \arg\min_{Y_c\in\mathbb{R}^{k\times\ell}} \|Y - CY_c\|_{\Theta,F}. \tag{18}$$

This finds the best supernode-level matrix Y_c such that the lifted node-level label signal CY_c matches the true label matrix Y as closely as possible. The optimality condition yields $\Theta_c Y_c = C^\top \Theta Y$, and we select the minimum-ℓ_2 norm solution provided in A.14 as $Y_c^\star = \Theta_c^\dagger C^\top \Theta Y$. We define the normalized projection error as:

$$\delta^\star := \max_{\mathcal{F}\in[\ell]} \frac{\| Y - CY_c^\star \|_{\Theta,\mathcal{F}}}{\| Y \|_{\Theta,\mathcal{F}}}$$

So, $\delta^\star$ tells us how much label information is lost when we represent node labels using only k supernodes. To understand why entropy regularization improves coarsening quality, we analyze the structure of the projection distortion $\delta^\star$.

Lemma 2. *From Eq.18, the Dirichlet residual of the projected labels satisfies*

$$\|Y - CY_c^\star\|_{\Theta,F}^2 \lesssim \sum_{i=1}^{k} |S_i|^2 \left(1 - e^{-H(p_i)}\right), \tag{19}$$

Consequently, the projection distortion obeys

$$\delta^\star \lesssim \frac{\sqrt{\sum_{i=1}^{k} |S_i|^2\, h_2^{-1}(H(p_i))}}{\|Y\|_{\Theta,F}}, \tag{20}$$

where h_2^{-1} denotes the inverse binary entropy function.

Consequently, minimizing the entropy term directly bounds the projection distortion through its control of label impurity within supernodes. Therefore, δ^* mainly depends on how mixed the labels are inside each supernode. Entropy regularization reduces this mixing and gives a direct bound, while Frobenius regularization can allow more mixing. This explains why entropy-guided coarsening can yield better performance.

Proof. The proof is deferred to Appendix A.6 □

4.3 Dirichlet-Energy Distortion

Theorem 3. *Let $\widetilde{\Theta} = C\Theta_c C^\top$ with $\Theta_c = C^\top \Theta C$, and define $\Delta := \Theta - \widetilde{\Theta}$. Then, for the label matrix Y,*

$$\mathrm{Tr}(Y^\top \Delta Y) \le \|\Theta\|_2 \|C\|_2^4 g(\delta^*)\, \|Y\|_F^2, \qquad g(\delta^*) = 2\delta^* + (\delta^*)^2.$$

In particular, the Dirichlet energy is preserved up to an error that decreases as $\delta^\star$ decreases.

Proof. The proof is deferred to the Appendix A.4 □

Corollary 4 (Cut Preservation)**.** *Suppose Theorem .2 holds. Then for any cut indicator $\mathbf{z} \in \{0,1\}^p$ and the associated set $S = \{v : z_v = 1\}$, the cut energy distortion satisfies the following equation,*

$$\left|\mathbf{z}^\top \Theta \mathbf{z} - \mathbf{z}^\top \widetilde{\Theta} \mathbf{z}\right| = \left|\mathbf{z}^\top \Delta \mathbf{z}\right| \le \varepsilon_{\mathrm{dir}} \|\mathbf{z}\|_2^2 = \varepsilon_{\mathrm{dir}} |S|.$$

This equivalently encodes that the cut cost is preserved up to the cut size $|S|$

Proof. The proof is deferred to the Appendix A.12 □

4.4 Bounding Effective-Resistance Distortion

For nodes $u, v \in \{1, \ldots, p\}$, let $e_u, e_v \in \mathbb{R}^p$ be the standard basis vector of $\mathbb{R}^p$. The effective resistance on the original graph is

$$R_{uv} = (e_u - e_v)^\top \Theta^\dagger (e_u - e_v) \quad \text{and} \tag{21}$$

Effective resistance on the coarse graph is:

$$R_{uv}^{(c)} = (e_u - e_v)^\top \widetilde{\Theta}^\dagger (e_u - e_v), \tag{22}$$

where † is the Moore–Penrose pseudoinverse; $\Theta^\dagger$ equals Θ^{-1} (Klein & Randić, 1993; Doyle & Snell, 1984; Chung, 1997; Ben-Israel & Greville, 2003). Thus, to control $|R_{uv} - R_{uv}^{(c)}|$ for all pairs (u, v), it suffices to control the difference between $\|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2$.

Lemma 3. *Let Θ be the Laplacian of a connected graph with spectral gap $\lambda_{\text{gap}} := \lambda_2(\Theta) > 0$. Define the residual $\Delta_{err} := \|\Theta - \widetilde{\Theta}\|_2$. Then*

$$\|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2 \;\le\; \frac{\Delta_{err}}{\lambda_{\text{gap}}\,(\lambda_{\text{gap}} - \Delta_{err})}, \tag{23}$$

provided $\Delta_{err} < \lambda_{\text{gap}}$. In particular, if $\Delta_{err} \le \frac{1}{2}\lambda_{\text{gap}}$, then

$$\|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2 \;\le\; \frac{2\,\Delta_{err}}{\lambda_{\text{gap}}^2}. \tag{24}$$

Since λ_{gap} is a fixed property of the original graph, lowering Δ_{err} directly tightens the resistance-preservation guarantee. We provide formal guarantees on effective resistance preservation under graph coarsening; the detailed analysis and proofs are given in Appendix A.11.

We adopt an ϵ-similarity notion that measures how well the coarsened graph preserves feature smoothness under the graph Laplacian. The precise definition is given in Appendix A.13.

5 Entropy as an Information-Theoretic Objective

We formalize the probabilistic framework induced by coarsening and relate node-profile entropy to information-theoretic quantities and classification performance.

5.1 Entropy controls misclassifications of super-node labels

Theorem 5. *Let $p_i = (p_i(1), \ldots, p_i(\ell))$ be the class-distribution within supernode i, let $m_i := \max_j p_i(j)$ be the majority mass and $e_i := 1 - m_i$ the majority-vote error fraction in that supernode. Let the entropy be $H(p_i) := -\sum_{j=1}^{\ell} p_i(j) \log p_i(j)$ and define the binary entropy $h_2(x) := -x \log x - (1-x)\log(1-x)$. Then, for every i, $e_i \;\le\; h_2^{-1}\big(H(p_i)\big)$, where h_2^{-1} denotes the inverse of h_2 on $[0, \frac{1}{2}]$. Consequently, with N_i the size of supernode i and $q := \sum_{i=1}^{k} N_i e_i$ the size-weighted majority error, we have*

$$q \;\le\; \sum_{i=1}^{k} N_i\, h_2^{-1}\big(H(p_i)\big). \tag{25}$$

Proof. The proof is deferred to the Appendix A.7 □

5.2 Evaluating Label Mixing:

In the coarsened graph, each supernode should contain mostly nodes from the same class. To evaluate the label mixing in supernodes we use three label-aware metrics. **SII** measures how mixed the labels are. **AMM** is the average error if we label each supernode by its majority class. **EIEC** turns entropy into a data-driven upper bound on that error and should be at least as large as AMM. Together, these metrics reveal mixing (SII), quantify its cost (AMM), and check consistency with theory (EIEC).

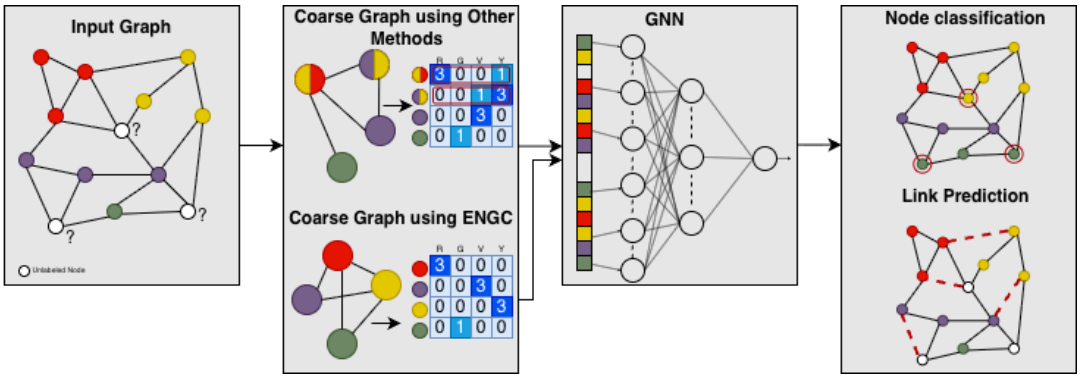

Figure 2: Schematic of the proposed ENGC framework and its use in downstream tasks. Starting from the original partially labeled graph $\mathcal{G}(\Theta, X, Y)$, ENGC learns a node-to-supernode mapping C and constructs a coarsened graph $\mathcal{G}_c(\Theta_c, \tilde{X}, \tilde{Y})$ where, $\tilde{Y}$ is obtained by aggregating the available labels through the learned mapping. A GNN is then trained on the coarsened graph and used for downstream prediction, with performance evaluated on the original graph for tasks such as node classification and link prediction.

Definition 5.1 (SII — Supernode Impurity Index)**.** *Let $p_i \in \Delta^{\ell-1}$ be the label distribution of supernode i, Over the k' supernodes with $N_i > 0$, the* Supernode Impurity Index (SII) *is* $\frac{1}{k'}\sum_{i=1}^{k'} H(p_i)$.

Definition 5.2 (AMM — Average Majority Miss)**.** *For supernode i, define the majority-vote error $e_i := 1-\max_j p_i(j)$ Over the k' supernodes with $N_i > 0$, the* Average Majority Miss (AMM) *is* AMM $= \frac{1}{k'}\sum_{i=1}^{k'} e_i$.

Definition 5.3 (EIEC — Entropy-Induced Error Ceiling)**.** *Let $H_2(x) = -x\log x - (1-x)\log(1-x)$ be the binary entropy with inverse H_2^{-1} on $[0, \frac{1}{2}]$. For supernode i, $b_i := H_2^{-1}\big(H(p_i)\big) \in [0, \frac{1}{2}]$, $\quad \bar{b} := \frac{1}{k'}\sum_{i=1}^{k'} b_i$.*

Theorem 5 implies AMM $\leq$ EIEC.

6 Experiments

This section presents experiments validating our proposed ENGC algorithm, beginning with experimental settings, followed by a comparative analysis against key baselines, and concluding with a concise demonstration of ENGC's advantages.

6.1 Experimental Setup

Dataset: We performed experiments on the datasets shown in 4.

Baseline Techniques: We validate our algorithm through extensive experiments on real datasets, benchmarking against state-of-the-art methods: GCOND (Jin et al., 2021), SCAL (Huang et al., 2021), and FGC (Kumar et al., 2023b). These selections are based on their recent advancements and superior performance, establishing them as leading coarsening approaches. Next, we have evaluated the performance of our algorithm through node classification accuracy and time taken(τ) for coarsening and classification. Experiment on real dataset using our model outperforms all other state-of-the-art methods in node classification and time complexity.

6.2 Node Classification

We follow the usual setup on citation datasets and use the public Planetoid train/validation/test splits. While running ENGC (i.e., during coarsening), we give the method the graph structure and node features, but only the training labels. Concretely, we keep labels as one–hot for training nodes and put zeros for validation/test nodes, and then run ENGC on $G(\Theta, X, Y_{\text{train}})$.

Once we learn the mapping C and the coarsened graph $G_c = (\Theta_c, \tilde{X})$, we assign a label to each supernode using only the training labels (basically a majority vote inside each supernode, e.g., using $C^\top Y_{\text{train}}$). After

that, we train the downstream GNN on G_c using only those supernodes that have at least one training node. We tune all hyperparameters on the validation split. Moreover, we have also compared the node classification task using the coarsened graph with the task using the original graph. The runtime complexity of this algorithm is discussed in Appendix A.19.

		Baseline		**Proposed**			
Dataset	**r=k/p**	**GCOND**	**SCAL**	**FGC**	**LAGC**	**ENGC**	**Whole dataset**
CORA	0.3	81.56 ± 0.62	79.42 ± 1.71	84.03 ± 0.08	87.62 ± 0.01	87.05 ± 0.05	89.5 ± 1.23
	0.1	81.37 ± 0.40	71.38 ± 3.62	79.96 ± 0.18	86.10 ± 0.03	86.00 ± 1.05	
	0.05	78.93 ± 0.45	55.32 ± 7.03	77.31 ± 0.65	82.85 ± 0.02	86.74 ± 0.12	
CITESEER	0.3	72.43 ± 0.49	68.87 ± 1.37	72.85 ± 0.10	78.51 ± 1.25	79.69 ± 0.37	78.09 ± 1.96
	0.1	70.46 ± 0.49	71.38 ± 3.62	69.46 ± 0.22	76.00 ± 0.50	78.65 ± 1.95	
	0.05	64.03 ± 2.41	55.32 ± 7.03	69.02 ± 0.24	75.70 ± 0.31	77.30 ± 0.01	
CO-PHY	0.05	93.05 ± 0.26	73.09 ± 7.41	93.31 ± 0.11	94.46 ± 0.58	93.67 ± 0.06	96.22 ± 0.74
	0.03	92.81 ± 0.31	63.65 ± 9.65	92.00 ± 1.78	94.28 ± 0.21	94.04 ± 0.23	
	0.01	90.46 ± 0.58	61.07 ± 5.68	91.08 ± 0.78	93.26 ± 0.89	93.80 ± 0.05	
PUBMED	0.05	78.16 ± 0.30	72.82 ± 2.62	78.14 ± 0.29	82.85 ± 0.32	83.74 ± 0.02	88.89 ± 0.59
	0.03	78.04 ± 0.47	70.24 ± 2.63	77.60 ± 0.16	82.10 ± 0.21	82.73 ± 0.36	
	0.01	77.20 ± 0.02	50.49 ± 10.5	76.10 ± 1.91	81.27 ± 0.91	80.8 ± 0.02	
CO-CS	0.05	86.29 ± 0.63	34.45 ± 10.07	89.12 ± 0.08	91.36 ± 0.48	90.93 ± 0.07	93.32 ± 0.62
	0.03	86.32 ± 0.45	26.06 ± 9.29	86.32 ± 0.43	90.32 ± 0.97	89.39 ± 0.49	
	0.01	84.01 ± 0.02	14.42 ± 8.51	85.41 ± 0.24	88.27 ± 0.34	89.21 ± 0.04	
DBLP	0.05	79.15 ± 0.20	76.52 ± 2.88	80.08 ± 0.01	81.64 ± 0.42	81.34 ± 0.28	85.35 ± 0.80
	0.03	78.42 ± 1.26	75.49 ± 2.84	79.92 ± 0.48	80.93 ± 0.12	80.98 ± 0.66	
	0.01	74.29 ± 0.57	72.01 ± 1.83	77.47 ± 0.33	79.49 ± 0.53	80.18 ± 0.70	

Table 1: The table presents the node classification accuracy on real benchmark datasets for the proposed ENGC algorithms. Compairing the results with GCOND, SCAL, FGC and LAGC it is visible that proposed ENGC algorithm outperforms state-of-the art methods in most of the cases.

Algorithm 2: Node Classification using proposed ENGC

Input: $\mathcal{G}(\Theta, X, Y)$
Output: Trained weight matrix W^*
1 Apply ENGC on $\mathcal{G}$ to learn $P^\dagger$; $P^\dagger = C^T$;
2 Compute feature matrix of the coarsened graph: $X' = P^\dagger X$;
3 Compute labels of the coarsened graph: $Y' = \arg\max(P^\dagger Y)$;
4 Learn W^* matrix to minimize $\ell(GCN_{G_c}(W^*), Y')$;

6.3 Link prediction

We further test the usefulness of ENGC for the *link prediction* task and compare against strong coarsening baselines, including LAGC (Kumar et al., 2024) and FGC (Kumar et al., 2023b). Experiments are conducted on three standard citation networks: Cora, Citeseer, and PubMed. Following the SEAL framework (Zhang & Chen, 2018), the task is to predict whether an edge exists between a pair of nodes.

For each dataset, we randomly split the observed edges into disjoint train/validation/test sets with an 80/10/10 ratio and average results over multiple random splits. We ensure the training graph remains connected; if a split breaks connectivity, we move only the minimum required edges back into the training set. For every split, we generate negative samples by uniformly sampling an equal number of non-edges for train, validation and test. ENGC is run only on the training graph to learn the coarsened representation. We then train the SEAL link predictor on the coarsened graph using the training positives and corresponding negatives, tune hyperparameters on the validation split, and finally report test performance using AUC-(ROC) . We also report a full-graph baseline (i.e., training SEAL on the original graph without coarsening) for reference. Note that GCOND (Jin et al., 2021) is designed for node classification and is therefore not included in the link prediction comparisons. The detailed algorithm is given in Section 3

Data set	r=k/p	LAGC	ENGC	FGC	Whole Data
Cora	0.3	0.78	0.82	0.77	0.84
	0.1	0.77	0.81	0.75	
	0.05	0.75	0.76	0.72	
CITESEER	0.3	0.75	0.75	0.73	0.78
	0.1	0.74	0.73	0.70	
	0.05	0.72	0.70	0.68	
PubMed	0.05	0.77	0.79	0.67	0.83
	0.03	0.72	0.76	0.70	
	0.01	0.68	0.69	0.66	

Table 2: This table presents the Area Under the ROC Curve (AUC) metric to evaluate the link prediction task using ENGC algorithm, we have compared our result with state-of-the-art FGC(Kumar et al., 2023a) and LAGC(Kumar et al., 2024). Whole Data column presents the result of link prediction while using the entire dataset.

Variant	All	$-T_1$	$-T_2$	$-T_3$	$-T_4$	$-T_5$	$-T_6$
CORA 0.1	86.00 ± 1.05	77.41 ± 0.16	79.61 ± 0.41	84.84 ± 0.66	82.42 ± 0.01	82.53 ± 0.41	78.80 ± 0.15

Table 3: ENGC ablation on CORA(r = 0.1). Removing T_i denotes dropping that term from the objective.

6.4 Generalizability of the proposed ENGC Algorithm

To demonstrate the generalizability of learning a coarsened graph from our proposed algorithms, we employed various architectures to train the Graph Neural Network (GNN). Specifically, we utilized GNN architectures such as GCN (Kipf & Welling, 2016), APPNP (Gasteiger et al., 2018), and GAT(Veličković et al., 2017) for training and executing the node classification task. Table5 in the appendix illustrates that our methods for learning the coarsened graph are compatible with different widely used GNN architectures, yielding nearly identical node classification accuracy obtained on different GNN structures.

6.5 Ablation Study

We quantify the contribution of each component appearing in the C–update of ENGC by ablating one term at a time from the gradient $\nabla f(C)$ of the subproblem in Eq.10, keeping all other settings fixed. Specifically, starting from the full objective in Eq. 9 and the majorized update in Eq.12, we remove a single term T_i from $\nabla f(C)$ and retrain. We report node-classification accuracy (mean $\pm$ s.d.). The detailed discussion of the result is given in A.18. The gradient used in the C–update decomposes as:

$$\nabla f(C) = \underbrace{-2\gamma\,\Theta C\,(C^\top \Theta C + J)^{-1}}_{T_1} + \underbrace{\alpha\,(C\tilde{X} - X)\tilde{X}^\top}_{T_2} + \underbrace{2\,\Theta C\,\tilde{X}\tilde{X}^\top}_{T_3} + \underbrace{\lambda\,C\,\mathbf{1}}_{T_4} + \underbrace{\beta\,\Theta C C^\top \Theta C}_{T_5} + \underbrace{\nabla r_E(C, Y)}_{T_6}.$$

7 Conclusion

In this work, we introduced Entropy-Guided Semi-Supervised Graph Coarsening (ENGC), a principled framework for attributed graph coarsening that integrates graph topology, node features, and partial supervision through an entropy-based regularization of the node-profile matrix. By explicitly reducing label mixing within supernodes, ENGC yields coarse graphs that are both structurally meaningful and semantically more coherent. We developed an efficient BSUM-based solver, established theoretical guarantees relating entropy minimization to preservation and learning properties, and validated the approach on benchmark datasets for node classification and link prediction. Overall, the proposed framework shows that entropy regularization is a promising and theoretically grounded mechanism for improving semi-supervised graph coarsening.

References

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

Amir Beck and Dror Pan. *Convergence of an Inexact Majorization-Minimization Method for Solving a Class of Composite Optimization Problems*, pp. 375–410. 01 2018. ISBN 978-3-319-97477-4. doi: 10.1007/978-3-319-97478-1_13.

A. Ben-Israel and T. N. E. Greville. *Generalized Inverses: Theory and Applications*. Springer, 2 edition, 2003.

Derrick Blakely, Jack Lanchantin, and Yanjun Qi. Time and space complexity of graph convolutional networks. *Accessed on: Dec*, 31, 2021.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International Conference on Machine Learning*, pp. 1725–1735. PMLR, 2020.

Fan R. K. Chung. *Spectral Graph Theory*, volume 92 of *CBMS Regional Conference Series in Mathematics*. American Mathematical Society, 1997.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006. ISBN 9780471241959.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

Florian Dorfler and Francesco Bullo. Kron reduction of graphs with applications to electrical networks, 2011. URL `https://arxiv.org/abs/1102.2950`.

Peter G. Doyle and J. Laurie Snell. *Random Walks and Electric Networks*. Mathematical Association of America, 1984. Freely available online.

Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*, 2018.

Nafiseh Ghoroghchian, Gautam Dasarathy, and Stark Draper. Graph community detection from coarse measurements: Recovery conditions for the coarsened weighted stochastic block model. In *International Conference on Artificial Intelligence and Statistics*, pp. 3619–3627. PMLR, 2021.

Mohammad Hashemi, Shengbo Gong, Juntong Ni, Wenqi Fan, B. Aditya Prakash, and Wei Jin. A comprehensive survey on graph reduction: Sparsification, coarsening, and condensation. In Kate Larson (ed.), *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 8058–8066. International Joint Conferences on Artificial Intelligence Organization, 8 2024. doi: 10.24963/ijcai.2024/891. URL `https://doi.org/10.24963/ijcai.2024/891`. Survey Track.

Bruce Hendrickson, Robert W Leland, et al. A multi-level algorithm for partitioning graphs. *SC*, 95(28): 1–14, 1995.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2013.

Zengfeng Huang, Shengzhong Zhang, Chong Xi, Tang Liu, and Min Zhou. Scaling up graph neural networks via graph coarsening. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 675–684, 2021.

Wei Jin, Lingxiao Zhao, Shichang Zhang, Yozen Liu, Jiliang Tang, and Neil Shah. Graph condensation for graph neural networks. *arXiv preprint arXiv:2110.07580*, 2021.

Vassilis Kalofolias. How to learn a graph from smooth signals. In Arthur Gretton and Christian C. Robert (eds.), *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 920–929, Cadiz, Spain, 09–11 May 2016. Proceedings of Machine Learning Research.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Douglas J. Klein and Milan Randić. Resistance distance. *Journal of Mathematical Chemistry*, 12(1):81–95, 1993. doi: 10.1007/BF01164627.

Manoj Kumar, Anurag Sharma, and Sandeep Kumar. A unified framework for optimization-based graph coarsening. *Journal of Machine Learning Research*, 24(118):1–50, 2023a.

Manoj Kumar, Anurag Sharma, Shashwat Saxena, and Sandeep Kumar. Featured graph coarsening with similarity guarantees. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17953–17975. PMLR, 23–29 Jul 2023b.

Manoj Kumar, Subhanu Halder, Archit Kane, Ruchir Gupta, and Sandeep Kumar. Optimization framework for semi-supervised attributed graph coarsening. In *The 40th Conference on Uncertainty in Artificial Intelligence*, 2024. URL `https://openreview.net/forum?id=ehcPPdsFx4`.

Sandeep Kumar, Jiaxi Ying, José Vinícius de Miranda Cardoso, and Daniel P Palomar. A unified framework for structured graph learning via spectral constraints. *Journal of Machine Learning Research*, 21(22):1–60, 2020.

Andreas Loukas. Graph reduction with spectral and cut guarantees. *Journal of Machine Learning Research*, 20(116):1–42, 2019.

Andreas Loukas and Pierre Vandergheynst. Spectrally approximating large graphs with smaller graphs. In *International Conference on Machine Learning*, pp. 3237–3246. Proceedings of Machine Learning Research, 2018.

Meisam Razaviyayn, Mingyi Hong, and Zhi-Quan Luo. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization*, 23, 09 2012. doi: 10.1137/120891009.

Alfred Renyi. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, 1961.

Matteo Riondato, David García-Soriano, and Francesco Bonchi. Graph summarization with quality guarantees. *Data mining and knowledge discovery*, 31(2):314–349, 2017.

Dorit Ron, Ilya Safro, and Achi Brandt. Relaxation-based coarsening and multiscale graph organization, 2010. URL `https://arxiv.org/abs/1004.1220`.

Ying Sun, Prabhu Babu, and Daniel P. Palomar. Majorization-minimization algorithms in signal processing, communications, and machine learning. *IEEE Transactions on Signal Processing*, 65(3):794–816, 2017. doi: 10.1109/TSP.2016.2601299.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 5171–5181, Red Hook, NY, USA, 2018. Curran Associates Inc.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020.

A Appendix

A.1 Proof of Lemma 1

Proof. The Lagrangian function of Eq.13 is

$$L(C, \tilde{X}, \boldsymbol{\mu}_1) = \frac{1}{2} C^\top C - C^\top A - \boldsymbol{\mu}_1^\top C \tag{26}$$

where $\boldsymbol{\mu}_1$ is the dual variable. The KKT conditions of Eq.13 are

$$C - A - \boldsymbol{\mu}_1 = 0, \tag{27}$$

$$\boldsymbol{\mu}^\top C = 0, \tag{28}$$

$$C \geq 0, \tag{29}$$

$$\boldsymbol{\mu}_1 \geq 0 \tag{30}$$

Therefore, the solution C that satisfying the KKT conditions in (27-30) is $C^{t+1} = \left(C^{(t)} - \frac{1}{L}\nabla f\left(C^{(t)}\right)\right)^+$. This concludes the proof. □

A.2 Proof of Theorem 1

Proof of Theorem 1. We show that each limit point $(C^t, \tilde{X}^t)$ satisfies KKT condition for Eq.**??**. Let $(C^\infty, \tilde{X}^\infty)$ be a limit point of the generated sequence.
The Lagrangian function of Eq.**??** is

$$\begin{aligned} L(C, \tilde{X}, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2) = &-\gamma \log\det(C^T \Theta C + J) + \mathrm{tr}(\tilde{X}^T C^T \Theta C \tilde{X}) + \frac{\alpha}{2}||X - C\tilde{X}||_F^2 + r(C, Y) \\ &+ \frac{\beta}{2}\|C^\top \Theta C\|_F^2 + \frac{\lambda}{2}\sum_{i=1}^{p}\|[C^T]_i\|_1^2 - \boldsymbol{\mu}_1^\top C + \boldsymbol{\mu}_2^T \Big[\|C_1^T\|_2^2 - 1 \quad \|C_2^T\|_2^2 - 1 \ldots \|C_p^T\|_2^2 - 1\Big]^T \end{aligned} \tag{31}$$

where $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are the dual variables.
(1) The KKT condition with respect to C is

$$\begin{aligned} &-2\gamma\Theta C Z^{-1} + \alpha\left(C\tilde{X} - X\right)\tilde{X}^\intercal + 2\Theta C\tilde{X}\tilde{X}^\top + \lambda C \mathbf{1}_{k\times k} - \boldsymbol{\mu}_1 \\ &\quad + 2\Big[\mu_{21}C_1^\top, \ldots, \mu_{2p}C_p^\top\Big]^\top + \beta\Theta C C^\top \Theta C + \nabla r_E(C, Y) = 0, \end{aligned} \tag{32}$$

$$\boldsymbol{\mu}_2^\top \begin{bmatrix} \|C_1^\top\|_2^2 - 1 \\ \|C_2^\top\|_2^2 - 1 \\ \vdots \\ \|C_p^\top\|_2^2 - 1 \end{bmatrix} = 0 \tag{33}$$

$$\boldsymbol{\mu}_1^\top C = 0 \tag{34}$$

$$\boldsymbol{\mu}_1 \geq 0, \qquad C \geq 0, \qquad \boldsymbol{\mu}_2 \geq 0, \qquad \|[C^\top]_i\|_2^2 \leq 1 \tag{35}$$

where $\mathbf{1}_{k\times k}$ is a $k \times k$ matrix whose all entry is one and $Z = C^T\Theta C + J$. C is derived by using KKT condition from Eq.13:

$$\begin{aligned} C^\infty - C^\infty + \frac{1}{L}\Big(&- 2\gamma\Theta C^\infty\big((C^\infty)^\top\Theta C^\infty + J\big)^{-1} + \alpha\big(C^\infty\tilde{X}^\infty - X\big)(\tilde{X}^\infty)^\top \\ &+ 2\Theta C^\infty\tilde{X}^\infty(\tilde{X}^\infty)^\top + \lambda C^\infty\mathbf{1}_{k\times k} + \beta\Theta C^\infty(C^\infty)^\top\Theta C^\infty + \nabla r_E(C^\infty, Y)\Big) = 0. \end{aligned} \tag{36}$$

For $\boldsymbol{\mu}_1 = 0$ and $\mu_{2i}[C^T]_i^\infty = 0 \;\forall\; i = 1, 2, \ldots p$, we observe that C^∞ satisfies the KKT condition.
(2) The KKT condition with respect to $\tilde{X}$ is

$$2C^T\Theta C\tilde{X} + \alpha C^T(C\tilde{X} - X) = 0$$

This concludes the proof. □

A.3 Proof of Theorem 2

Proof of Theorem 2. We fix a row i and write $p := \phi_i/N_i$. Then $p_j \geq 0$ and $\sum_j p_j = 1$. For each j, $-p_j \log p_j \geq 0$, for all $p_j \in [0, 1]$. Hence $H(p) = \sum_j -p_j \log p_j \geq 0$, and $H(p) = 0$ iff p is one–hot. Equivalently, the minimizers are $\phi_i = N_i e_{j^\star}$. This proves (1) after summing over i. For (2), we fix row i. Then by Cauchy–Schwarz,

$$\Big(\sum_{j=1}^{\ell}\phi_{ij}\Big)^2 \leq \ell\sum_{j=1}^{\ell}\phi_{ij}^2 \quad\Longrightarrow\quad \sum_{j=1}^{\ell}\phi_{ij}^2 \;\geq\; \frac{N_i^2}{\ell},$$

with equality iff all ϕ_{ij} are equal, i.e., $\phi_{ij} = N_i/\ell$ for all j. Summing over i yields the stated global minimum for E_{fro}. □

A.4 Proof of Theorem 3

Proof of Theorem 3. Consider the matrix projection problem

$$Y_c^\star \;:=\; \arg\min_{Y_c\in\mathbb{R}^{k\times\ell}} \|Y - CY_c\|_{\Theta,F}, \qquad \|U\|_{\Theta,F}^2 := \mathrm{Tr}(U^\top\Theta U), \qquad \|\mathbf{z}\|_\Theta^2 := \mathbf{z}^\top\Theta\mathbf{z}.$$

Let the residual matrix be

$$R \;:=\; Y - CY_c^\star \in \mathbb{R}^{p\times\ell}.$$

Optimality for the squared objective

$$F(Y_c) \;:=\; \|Y - CY_c\|_{\Theta,F}^2 = \mathrm{Tr}\big((Y - CY_c)^\top\Theta(Y - CY_c)\big)$$

yields the first-order condition

$$C^\top\Theta(Y - CY_c^\star) \;=\; C^\top\Theta R \;=\; 0, \tag{A.1}$$

Writing $Y = CY_c^\star + R$ and using Eq.A.1 gives the decomposition

$$\mathrm{Tr}(Y^\top\Theta Y) = \mathrm{Tr}\big((CY_c^\star + R)^\top\Theta(CY_c^\star + R)\big) = \mathrm{Tr}\big((Y_c^\star)^\top(C^\top\Theta C)Y_c^\star\big) + \mathrm{Tr}(R^\top\Theta R) = \mathrm{Tr}\big((Y_c^\star)^\top\Theta_c Y_c^\star\big) + \|R\|_{\Theta,F}^2, \tag{A.2}$$

where $\Theta_c := C^\top\Theta C$.
Now let $\widetilde{\Theta} = C\Theta_c C^\top$ and $\Delta := \Theta - \widetilde{\Theta}$. We have

$$\mathrm{Tr}(Y^\top\widetilde{\Theta}Y) = \mathrm{Tr}\big(Y^\top C\Theta_c C^\top Y\big) = \mathrm{Tr}\big((C^\top Y)^\top\Theta_c(C^\top Y)\big) = \|\Theta_c^{1/2}C^\top Y\|_F^2. \tag{A.3}$$

Substituting $Y = CY_c^\star + R$ gives $C^\top Y = C^\top C\,Y_c^\star + C^\top R$. Hence, by the triangle inequality,

$$\|\Theta_c^{1/2}C^\top Y\|_F \geq \|\Theta_c^{1/2}C^\top C\,Y_c^\star\|_F - \|\Theta_c^{1/2}C^\top R\|_F. \tag{A.4}$$

Squaring and using $(a-b)^2 \geq a^2 - 2ab - b^2$ yields

$$\|\Theta_c^{1/2} C^\top Y\|_F^2 \geq \|\Theta_c^{1/2} C^\top C\, Y_c^\star\|_F^2 - 2\|\Theta_c^{1/2} C^\top C\, Y_c^\star\|_F \, \|\Theta_c^{1/2} C^\top R\|_F - \|\Theta_c^{1/2} C^\top R\|_F^2. \tag{A.5}$$

Therefore,

$$\mathrm{Tr}(Y^\top \Delta Y) = \mathrm{Tr}(Y^\top \Theta Y) - \mathrm{Tr}(Y^\top \widetilde{\Theta} Y) \leq 2\|\Theta_c^{1/2} C^\top C\, Y_c^\star\|_F \, \|\Theta_c^{1/2} C^\top R\|_F + \|\Theta_c^{1/2} C^\top R\|_F^2, \tag{A.6}$$

where we dropped the nonnegative term $\|\Theta_c^{1/2} C^\top C\, Y_c^\star\|_F^2$.

We now bound the two factors using spectral norms. First, since

$$\|\Theta_c\|_2 = \|C^\top \Theta C\|_2 \leq \|C\|_2^2 \|\Theta\|_2, \quad \text{we have} \quad \|\Theta_c^{1/2}\|_2 \leq \|C\|_2 \sqrt{\|\Theta\|_2}. \tag{A.7}$$

Hence

$$\|\Theta_c^{1/2} C^\top R\|_F \leq \|\Theta_c^{1/2}\|_2 \, \|C^\top\|_2 \, \|R\|_F \leq \|C\|_2^2 \sqrt{\|\Theta\|_2} \, \|R\|_F \leq \|C\|_2^2 \, \|R\|_{\Theta,F}, \tag{A.8}$$

where the last inequality uses $\|R\|_{\Theta,F}^2 = \mathrm{Tr}(R^\top \Theta R) \leq \|\Theta\|_2 \|R\|_F^2$.

Second, by Eq.A.2,

$$\|\Theta_c^{1/2} Y_c^\star\|_F^2 = \mathrm{Tr}\big((Y_c^\star)^\top \Theta_c Y_c^\star\big) \leq \mathrm{Tr}(Y^\top \Theta Y) = \|Y\|_{\Theta,F}^2,$$

so $\|\Theta_c^{1/2} Y_c^\star\|_F \leq \|Y\|_{\Theta,F}$. Using this and $\|C^\top C\|_2 \leq \|C\|_2^2$ gives

$$\|\Theta_c^{1/2} C^\top C\, Y_c^\star\|_F \leq \|C^\top C\|_2 \, \|\Theta_c^{1/2} Y_c^\star\|_F \leq \|C\|_2^2 \, \|Y\|_{\Theta,F}. \tag{A.9}$$

Substituting Eq.A.8–Eq.A.9 into Eq.A.6 yields

$$\mathrm{Tr}(Y^\top \Delta Y) \leq \|C\|_2^4 \Big(2\|Y\|_{\Theta,F} \|R\|_{\Theta,F} + \|R\|_{\Theta,F}^2\Big). \tag{A.10}$$

Define the normalized projection error

$$\delta(Y) \;:=\; \frac{\|R\|_{\Theta,F}}{\|Y\|_{\Theta,F}} \;=\; \frac{\|Y - CY_c^\star\|_{\Theta,F}}{\|Y\|_{\Theta,F}}.$$

Then $\|R\|_{\Theta,F} = \delta(Y)\|Y\|_{\Theta,F}$ and

$$\mathrm{Tr}(Y^\top \Delta Y) \leq \|C\|_2^4 \big(2\delta(Y) + \delta(Y)^2\big) \, \|Y\|_{\Theta,F}^2. \tag{A.11}$$

Finally, since $\|Y\|_{\Theta,F}^2 = \mathrm{Tr}(Y^\top \Theta Y) \leq \|\Theta\|_2 \|Y\|_F^2$, we obtain

$$\mathrm{Tr}(Y^\top \Delta Y) \leq \|\Theta\|_2 \, \|C\|_2^4 \, g(\delta(Y)) \, \|Y\|_F^2, \qquad g(t) = 2t + t^2.$$

This proves the stated bound with $\Delta \leq \|\Theta\|_2 \|C\|_2^4 g(\delta(Y))$.

□

A.5 Proof of Lemma 3

Proof of Lemma 3. Because the graph is connected, $\Theta\mathbf{1} = 0$ and $\lambda_2(\Theta) = \lambda_{\mathrm{gap}} > 0$. It is standard that the Moore–Penrose pseudoinverse $\Theta^\dagger$ acts as the true inverse of Θ on all zero-sum vectors x with $\mathbf{1}^\top x = 0$, and similarly $\widetilde{\Theta}^\dagger$ acts as the true inverse of $\widetilde{\Theta}$ on that same subspace. On this subspace both Θ and $\widetilde{\Theta}$ are symmetric positive definite.

Let $A := \Theta$, $B := \widetilde{\Theta}$, and $E := A - B = \Theta - \widetilde{\Theta}$, all viewed as operators on the zero-sum subspace. For invertible A, B we have

$$B^{-1} - A^{-1} \;=\; B^{-1}(A - B)A^{-1} \;=\; B^{-1} E A^{-1}.$$

Taking operator norms and using submultiplicativity,

$$\|A^{-1} - B^{-1}\|_2 \;\leq\; \|B^{-1}\|_2 \, \|E\|_2 \, \|A^{-1}\|_2. \tag{37}$$

By definition, $\|E\|_2 \le \|\Theta - \widetilde{\Theta}\|_2 = \Delta_{\text{res}}$. Since the smallest nonzero eigenvalue of Θ is λ_{gap}, the inverse of Θ on this subspace has norm $\|A^{-1}\|_2 = 1/\lambda_{\text{gap}}$ (Horn & Johnson, 2013). Moreover, by Weyl's eigenvalue inequality for Hermitian matrices, the smallest nonzero eigenvalue of $\widetilde{\Theta}$ satisfies

$$\lambda_{\min}(\widetilde{\Theta}) \;\ge\; \lambda_{\text{gap}} - \|\Theta - \widetilde{\Theta}\|_2 \;=\; \lambda_{\text{gap}} - \Delta,$$

so

$$\|B^{-1}\|_2 \;=\; \frac{1}{\lambda_{\min}(\widetilde{\Theta})} \;\le\; \frac{1}{\lambda_{\text{gap}} - \Delta}.$$

Substituting these bounds into Eq.37 gives

$$\|A^{-1} - B^{-1}\|_2 \;\le\; \frac{1}{\lambda_{\text{gap}} - \Delta} \cdot \Delta \cdot \frac{1}{\lambda_{\text{gap}}} \;=\; \frac{\Delta}{\lambda_{\text{gap}}\,(\lambda_{\text{gap}} - \Delta)}.$$

Finally, $\Theta^\dagger$ and $\widetilde{\Theta}^\dagger$ agree with A^{-1} and B^{-1} on all zero-sum vectors and vanish on $\mathbf{1}$, so the same bound holds for $\|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2$, proving Eq.23.

If $\Delta \le \frac{1}{2}\lambda_{\text{gap}}$, then $\lambda_{\text{gap}} - \Delta \ge \frac{1}{2}\lambda_{\text{gap}}$, which simplifies Eq.23 to Eq.24.

As an immediate consequence of Lemma 3, we obtain for any pair of distinct nodes u, v,

$$\begin{aligned} \left|R_{uv} - R_{uv}^{(L)}\right| \;&=\; \left|(\mathbf{e}_u - \mathbf{e}_v)^\top (\Theta^\dagger - \widetilde{\Theta}^\dagger)(\mathbf{e}_u - \mathbf{e}_v)\right| \\ &\le\; \|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2 \, \|\mathbf{e}_u - \mathbf{e}_v\|_2^2 \;\le\; 4\,\|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2. \end{aligned}$$

Therefore, if $\Delta \ll \lambda_{\text{gap}}$, all effective resistances are uniformly preserved.

□

A.6 Proof of Lemma 2

Proof of Lemma2. Recall that the Dirichlet residual is

$$\|Y - CY_c^\star\|_{\Theta,F}^2 = \operatorname{Tr}\big((Y - CY_c^\star)^\top \Theta (Y - CY_c^\star)\big). \tag{38}$$

Using the Laplacian quadratic form identity,

$$\operatorname{Tr}(A^\top \Theta A) = \sum_{(u,v)\in E} w_{uv}\, \|a_u - a_v\|_2^2, \tag{39}$$

and noting that $CY_c^\star$ is constant on each supernode S_i, we obtain for intra-supernode edges

$$\|y_u - y_v\|_2^2 = \begin{cases} 0, & y_u = y_v, \\ 2, & y_u \ne y_v. \end{cases} \tag{40}$$

Hence the Dirichlet residual contains the exact label-mixing term

$$\|Y - CY_c^\star\|_{\Theta,F}^2 = 2\sum_{i=1}^{k} \sum_{\substack{(u,v)\in E \\ u,v\in S_i}} w_{uv}\, \mathbf{1}[y_u \ne y_v] + (\text{nonnegative cross-supernode terms}). \tag{41}$$

Thus, projection distortion is governed by weighted intra-supernode label disagreement. The expected disagreement weight scales as

$$\mathbb{E}\left[\sum_{\substack{(u,v)\in E \\ u,v\in S_i}} w_{uv} \mathbf{1}[y_u \ne y_v]\right] \propto |S_i|^2 M(p_i). \tag{42}$$

Consequently,

$$\mathbb{E}\big[\|Y - CY_c^\star\|_{\Theta,F}^2\big] \propto \sum_i |S_i|^2 M(p_i), \tag{43}$$

Using Eq.17 above, we get

$$\mathbb{E}\big[\|Y - CY_c^\star\|_{\Theta,F}^2\big] \lesssim \sum_{i=1}^{k} |S_i|^2 \left(1 - e^{-H(p_i)}\right), \tag{44}$$

is controlled by the impurity of label distributions within supernodes.

Moreover, using the entropy–majority from 5,

$$M(p_i) \leq 2\, h_2^{-1}\big(H(p_i)\big), \tag{45}$$

we obtain the explicit control as below:

$$\delta^\star \;\lesssim\; \frac{\sqrt{\sum_i |S_i|^2\, h_2^{-1}(H(p_i))}}{\|Y\|_{\Theta,F}}. \tag{46}$$

□

A.7 Proof of Theorem 5

Proof of Theorem5. Fix a supernode i and write $p := p_i$, $m := m_i$, $e := e_i = 1 - m$. Without loss of generality, relabel coordinates so that $p_1 = m = \max_j p_j$. Then $p_1 = 1 - e$, $\sum_{j=2}^{\ell} p_j = e$, and $p_j \geq 0$ for $j \geq 2$.

Consider the function $\varphi(x) := -x \log x$, which is concave on $[0, 1]$. For fixed $p_1 = 1 - e$, the entropy can be written as

$$H(p) \;=\; \varphi(1-e) \;+\; \sum_{j=2}^{\ell} \varphi(p_j), \text{subject to} \;\; \sum_{j=2}^{\ell} p_j = e, \;\; p_j \geq 0. \tag{47}$$

Since $\sum_{j=2}^{\ell} \varphi(p_j)$ is a concave function over $\{(p_2, \ldots, p_\ell) \in \mathbb{R}_+^{\ell-1} : \sum_{j=2}^{\ell} p_j = e\}$, its minimum over this convex set is attained at an extreme point, i.e. when there is single element which is non zero. Hence

$$\min_{\substack{p_j \geq 0 \\ \sum_{j\geq 2} p_j = e}} H(p) \;=\; \varphi(1-e) + \varphi(e) + \sum_{j=3}^{\ell} \varphi(0) \;=\; h_2(e) \tag{48}$$

Therefore, for every feasible p with majority error e,

$$H(p) \;\geq\; h_2(e). \tag{49}$$

Next, observe that h_2 is strictly increasing on $[0, \frac{1}{2}]$: indeed,

$$h_2'(x) \;=\; \log\frac{1-x}{x} \;>\; 0 \quad \text{for } x \in (0, \tfrac{1}{2}). \tag{50}$$

Thus h_2 admits an inverse h_2^{-1} on $[0, \frac{1}{2}]$, and from $H(p) \geq h_2(e)$ we obtain

$$e \;\leq\; h_2^{-1}\big(H(p)\big). \tag{51}$$

Restoring the index i yields the stated pointwise bound $e_i \leq h_2^{-1}(H(p_i))$.

So we get,

$$q = \sum_{i=1}^{k} N_i e_i \;\leq\; \sum_{i=1}^{k} N_i\, h_2^{-1}\big(H(p_i)\big) \tag{52}$$

This completes the proof. □

A.8 Notations

In terms of notation, lower case (bold) letters denote scalars (vectors) and upper case letters denote matrices. The dimension of a matrix is omitted whenever it is clear from the context. The (i, j)-th entry of a matrix X is denoted by X_{ij}. $X^\dagger$ and $X^\top$ denote the pseudoinverse and transpose of matrix X, respectively. X_i and $[X^T]_j$ denote the i-th column and j-th row of matrix X. The all-zero and all-one vectors or matrices of appropriate sizes are denoted by $\underline{0}$ and $\underline{1}$, respectively. The $\|X\|_1$, $\|X\|_F$, $\|X\|_{1,2}$ denote the ℓ_1-norm, Frobenius norm and $\ell_{1,2}$-norm of X, respectively. The Euclidean norm of the vector X is denoted as $\|X\|_2$. $\det(X)$ is defined as the generalized determinant of a positive definite matrix X, i.e., the product of its non-zero eigenvalues. The inner product of two matrices is defined as $\langle X, Y\rangle = \mathrm{tr}(X^\top Y)$, where $\mathrm{tr}(\cdot)$ is the trace operator. $\mathbb{R}_+$ represents positive real numbers. The inner product of two vectors is defined as $\langle X_i, X_j\rangle = X_i^T X_j$ where X_i and X_j are the i-th and j-th column of matrix X.

A.9 Error Bounds: Entropy vs. Frobenius

Figure 3 reports three summary metrics. On Cora at $r = 0.1$ ($k = 270$), **entropy regularization** yields purer supernodes and fewer majority-vote mistakes: SII drops from 1.2818 to 0.3193 (a $\approx 75\%$ reduction), and AMM falls from 0.4002 to 0.1242 (also $\approx 75\%$ lower). Its EIEC is 0.4111, comfortably above its AMM (0.1242) as theory predicts, indicating a valid, non-tight ceiling. In contrast, the Frobenius baseline exhibits much higher impurity and error (SII 1.2818, AMM 0.4002); its reported EIEC (0.4407) sits slightly below AMM. Overall, entropy regularization directly encourages label-consistent, compact supernodes and delivers markedly lower error than Frobenius, making it the preferred choice for label-aware graph coarsening with theoretically consistent error control.

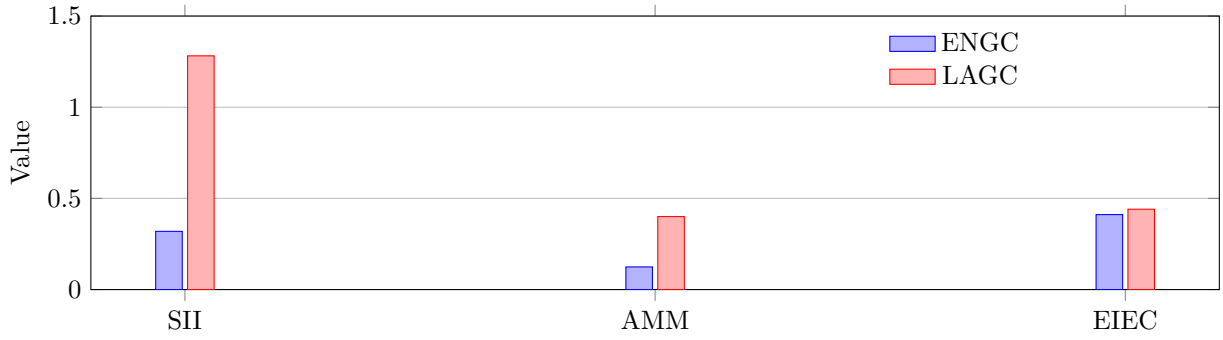

Figure 3: Cora ($r = 0.1, k = 270$): SII, AMM, EIEC for ENGC vs. LAGC.

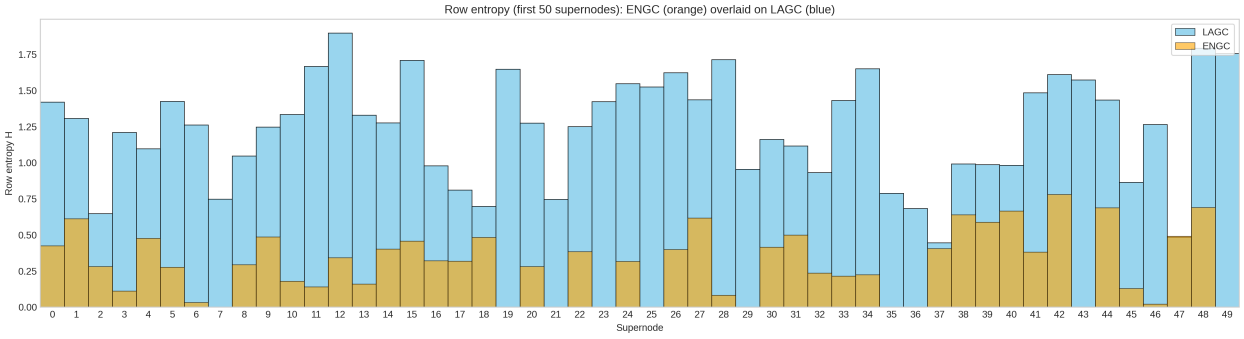

Figure 4: **Row-entropy per supernode (first 50).** Dataset: Cora; label rate $r = 0.1$; $k = 270$ supernodes. Bars are *overlaid* : LAGC/Frobenius in sky blue, ENGC/Entropy in orange. Height equals $H(p_i)$ Lower bars indicate purer label allocation within a supernode; ENGC generally yields lower or comparable $H(p_i)$ than LAGC on the supernodes shown.

A.10 Dataset Description

Dataset We have performed the experiments on the datasets as shown in the Table 4.

Dataset	Nodes	Edges	Features	Classes
CORA	2,708	5,429	1,433	7
CITESEER	3,327	9,104	3,703	6
DBLP	17,716	52,867	1,639	4
CO-CS	18,333	163,788	6,805	15
PUBMED	19,717	44,338	500	3
CO-PHYSICS	34,493	247,962	8,415	5

Table 4: Overview of the datasets employed for node classification

A.11 Resistance preservation.

Let $G(\Theta, X, Y)$ be a connected graph with Laplacian $\Theta \in \mathbb{R}^{p\times p}$. Denote the all-ones vector by $\mathbf{1}$

For connected G, with smallest eigenvalue $\lambda_{\text{gap}} := \lambda_2(\Theta) > 0$ Fiedler (1973); Chung (1997). Let $C \in \mathbb{R}_+^{p\times k}$ be the mapping matrix, $\Theta_c := C^\top \Theta C \in \mathbb{R}^{k\times k}$ the coarse Laplacian, and $\widetilde{\Theta} := C\Theta_c C^\top \in \mathbb{R}^{p\times p}$ its lifted. Define the energy distortion

$$\Delta_{err} \;:=\; \big\|\, \Theta - \tilde{\Theta}\|_2. \tag{53}$$

Using $\|e_u - e_v\|_2^2 = 2$ and $|x^\top M x| \le \|M\|_2\, \|x\|_2^2$,

$$|R_{uv} - R_{uv}^{(c)}| \;=\; |(e_u - e_v)^\top (\Theta^\dagger - \widetilde{\Theta}^\dagger)(e_u - e_v)| \;\le\; 2\,\|\Theta^\dagger - \widetilde{\Theta}^\dagger\|_2.$$

Combining with Eq.23 gives

$$|R_{uv} - R_{uv}^{(c)}| \;\le\; \frac{2\,\Delta_{err}}{\lambda_{\text{gap}}\,(\lambda_{\text{gap}} - \Delta_{err})}, \tag{54}$$

and under $\Delta \le \frac{1}{2}\lambda_{\text{gap}}$ the simplified bound

$$|R_{uv} - R_{uv}^{(c)}| \;\le\; \frac{4\,\Delta_{err}}{\lambda_{\text{gap}}^2}. \tag{55}$$

A.12 Cut-Preservation

Proof of Corollary 4. Let $\mathbf{z} \in \{0,1\}^p$ be a cut indicator and set $S := \{v :\, z_v = 1\}$. By definition $\Delta := \Theta - \widetilde{\Theta}$, so

$$\mathbf{z}^\top \Theta \mathbf{z} - \mathbf{z}^\top \widetilde{\Theta}\mathbf{z} = \mathbf{z}^\top(\Theta - \widetilde{\Theta})\mathbf{z} = \mathbf{z}^\top \Delta f z.$$

Applying Theorem 2 with $\mathbf{u} = \mathbf{z}$ gives

$$\left|\mathbf{z}^\top \Theta \mathbf{z} - \mathbf{z}^\top \widetilde{\Theta}\mathbf{z}\right| = \left|\mathbf{z}^\top \Delta \mathbf{z}\right| \le \Delta \|\mathbf{z}\|_2^2.$$

Since $\mathbf{z} \in \{0,1\}^p$, we have $\|\mathbf{z}\|_2^2 = \sum_{v=1}^p z_v^2 = \sum_{v=1}^p z_v = |S|$, and therefore

$$\left|\mathbf{z}^\top \Theta \mathbf{z} - \mathbf{z}^\top \widetilde{\Theta}\mathbf{z}\right| \le \Delta |S|.$$

□

A.13 ϵ-similarity

The coarsened graph data $\mathcal{G}_c(\Theta_c, \tilde{X})$ is ϵ similar to the original graph data $\mathcal{G}(\Theta, X)$, i.e., there exist an $0 \le \epsilon \le 1$ such that

$$(1-\epsilon)\|X\|_\Theta \le \|\tilde{X}\|_{\Theta_c} \le (1+\epsilon)\|X\|_\Theta \tag{56}$$

where $\|X\|_\Theta^2 = \text{tr}(X^T \Theta X)$ and $\|\tilde{X}\|_{\Theta_c}^2 = \text{tr}(\tilde{X}^T \Theta_c \tilde{X})$.

A.14 Derivation of the optimality condition and the minimum-ℓ_2 solution

Proof. Fix a class $c \in [\ell]$ and consider the c-th column of the matrix projection problem

$$Y_c^\star := \arg\min_{Y_c \in \mathbb{R}^{k\times \ell}} \|Y - CY_c\|_{\Theta,F}, \qquad \|U\|_{\Theta,F}^2 := \mathrm{Tr}(U^\top \Theta U) = \sum_{t=1}^{\ell} \|(Y - CY_c)_{:,t}\|_\Theta^2,$$

where $\|\mathbf{u}\|_\Theta^2 := \mathbf{u}^\top \Theta \mathbf{u}$. Since the objective is a sum over columns, minimizing over Y_c decouples across columns. In particular, the optimal c-th column $(Y_c^\star)_{:,c} \in \mathbb{R}^k$ solves

$$(Y_c^\star)_{:,c} \;=\; \arg\min_{\mathbf{y}\in\mathbb{R}^k} \; \|Y_{:,c} - C\mathbf{y}\|_\Theta.$$

Define $\boldsymbol{\ell} := Y_{:,c} \in \mathbb{R}^p$ and write $\mathbf{y} := (Y_c)_{:,c} \in \mathbb{R}^k$. Minimizing $\|\boldsymbol{\ell} - C\mathbf{y}\|_\Theta$ is equivalent to minimizing its square

$$f(\mathbf{y}) \;:=\; \|\boldsymbol{\ell} - C\mathbf{y}\|_\Theta^2 = (\boldsymbol{\ell} - C\mathbf{y})^\top \Theta (\boldsymbol{\ell} - C\mathbf{y}) = \boldsymbol{\ell}^\top \Theta \boldsymbol{\ell} - 2\mathbf{y}^\top C^\top \Theta \boldsymbol{\ell} + \mathbf{y}^\top (C^\top \Theta C)\mathbf{y}.$$

Define the coarse Laplacian $\Theta_c := C^\top \Theta C$, which is symmetric positive semidefinite. Differentiating and using $\nabla_{\mathbf{y}}(\mathbf{y}^\top \Theta_c \mathbf{y}) = 2\Theta_c \mathbf{y}$ and $\nabla_{\mathbf{y}}(\mathbf{y}^\top b) = b$ yields

$$\nabla f(\mathbf{y}) \;=\; 2\Theta_c \mathbf{y} - 2C^\top \Theta \boldsymbol{\ell}.$$

Setting $\nabla f(\mathbf{y}) = \mathbf{0}$ gives the optimality condition:

$$\Theta_c \mathbf{y} \;=\; C^\top \Theta \boldsymbol{\ell} \;=\; C^\top \Theta Y_{:,c}.$$

Collecting these optimality conditions for all $c \in [\ell]$ yields the matrix condition

$$\Theta_c Y_c \;=\; C^\top \Theta Y.$$

Since Θ is a Laplacian, $\Theta \mathbf{1}_p = \mathbf{0}$, and thus $\Theta_c \mathbf{1}_k = C^\top \Theta C \mathbf{1}_k = C^\top \Theta \mathbf{1}_p = \mathbf{0}$, implying that Θ_c is singular and the equation may have infinitely many solutions. Among all solutions, the Moore–Penrose pseudoinverse selects the unique minimum-ℓ_2-norm solution, hence

$$Y_c^\star \;=\; \Theta_c^\dagger C^\top \Theta Y,$$

which completes the derivation.

□

A.15 Proof of Lipschitz Continuity

Theorem 6. *Assume there exist constants $\eta > 0$ and $\varepsilon > 0$ such that for every admissible C,*

$$N_i(C) \geq \eta \qquad \forall i = 1, \ldots, k,$$

and

$$[p_i(C)]_j \geq \varepsilon \qquad \forall i = 1, \ldots, k, \;\; \forall j = 1, \ldots, \ell.$$

Then $r_E(\cdot, Y)$ is Lipschitz continuous on this restricted set. In particular, for any two matrices $C_1, C_2 \in \mathbb{R}^{p\times k}$,

$$|r_E(C_1, Y) - r_E(C_2, Y)| \leq \frac{2(1 + |\ln \varepsilon|)}{\eta \ln 2} \|C_1 - C_2\|_{1,1}.$$

Consequently, since $\|M\|_{1,1} \leq \sqrt{pk}\, \|M\|_F$,

$$|r_E(C_1, Y) - r_E(C_2, Y)| \leq \frac{2(1 + |\ln \varepsilon|)}{\eta \ln 2} \sqrt{pk}\, \|C_1 - C_2\|_F.$$

Proof. Let $c_i^{(1)}$ and $c_i^{(2)}$ denote the i-th columns of C_1 and C_2, respectively. Then the i-th rows of $C_1^\top Y$ and $C_2^\top Y$ are

$$\Phi_i(C_1) = \left(c_i^{(1)}\right)^\top Y, \qquad \Phi_i(C_2) = \left(c_i^{(2)}\right)^\top Y.$$

Also define

$$N_i^{(1)} := N_i(C_1) = \|\Phi_i(C_1)\|_1, \qquad N_i^{(2)} := N_i(C_2) = \|\Phi_i(C_2)\|_1,$$

and

$$p_i(C_1) = \frac{\Phi_i(C_1)}{N_i^{(1)}}, \qquad p_i(C_2) = \frac{\Phi_i(C_2)}{N_i^{(2)}}.$$

We first bound the difference of the i-th rows:

$$\Phi_i(C_1) - \Phi_i(C_2) = \left(c_i^{(1)} - c_i^{(2)}\right)^\top Y.$$

Hence,

$$\|\Phi_i(C_1) - \Phi_i(C_2)\|_1 = \left\|\left(c_i^{(1)} - c_i^{(2)}\right)^\top Y\right\|_1.$$

Write $x := c_i^{(1)} - c_i^{(2)} \in \mathbb{R}^p$. Then

$$x^\top Y = \sum_{u=1}^{p} x_u \, y_u^\top,$$

so by the triangle inequality,

$$\|x^\top Y\|_1 \le \sum_{u=1}^{p} |x_u| \, \|y_u\|_1 \le \sum_{u=1}^{p} |x_u| = \|x\|_1.$$

Therefore,

$$\|\Phi_i(C_1) - \Phi_i(C_2)\|_1 \le \|c_i^{(1)} - c_i^{(2)}\|_1.$$

Next, by the reverse triangle inequality,

$$|N_i^{(1)} - N_i^{(2)}| = |\|\Phi_i(C_1)\|_1 - \|\Phi_i(C_2)\|_1| \le \|\Phi_i(C_1) - \Phi_i(C_2)\|_1 \le \|c_i^{(1)} - c_i^{(2)}\|_1.$$

Now,

$$p_i(C_1) - p_i(C_2) = \frac{\Phi_i(C_1)}{N_i^{(1)}} - \frac{\Phi_i(C_2)}{N_i^{(2)}}.$$

Add and subtract $\Phi_i(C_2)/N_i^{(1)}$:

$$p_i(C_1) - p_i(C_2) = \frac{\Phi_i(C_1) - \Phi_i(C_2)}{N_i^{(1)}} + \Phi_i(C_2)\left(\frac{1}{N_i^{(1)}} - \frac{1}{N_i^{(2)}}\right).$$

Taking ℓ_1-norms gives

$$\|p_i(C_1) - p_i(C_2)\|_1 \le \frac{\|\Phi_i(C_1) - \Phi_i(C_2)\|_1}{N_i^{(1)}} + \|\Phi_i(C_2)\|_1 \left|\frac{1}{N_i^{(1)}} - \frac{1}{N_i^{(2)}}\right|.$$

Since $\|\Phi_i(C_2)\|_1 = N_i^{(2)}$, this becomes

$$\|p_i(C_1) - p_i(C_2)\|_1 \le \frac{\|\Phi_i(C_1) - \Phi_i(C_2)\|_1}{N_i^{(1)}} + \frac{|N_i^{(1)} - N_i^{(2)}|}{N_i^{(1)}}.$$

Using $N_i^{(1)} \ge \eta$, we obtain

$$\|p_i(C_1) - p_i(C_2)\|_1 \le \frac{2}{\eta} \, \|c_i^{(1)} - c_i^{(2)}\|_1.$$

Next, consider the entropy function

$$H(p) = -\sum_{j=1}^{\ell} p_j \log_2 p_j.$$

Its partial derivatives are

$$\frac{\partial H}{\partial p_j} = -\frac{\ln p_j + 1}{\ln 2}.$$

Since $[p_i(C)]_j \geq \varepsilon$, we have on the restricted set

$$\|\nabla H(p)\|_\infty \leq \frac{1 + |\ln \varepsilon|}{\ln 2}.$$

Therefore, by the mean value theorem,

$$|H(p) - H(q)| \leq \frac{1 + |\ln \varepsilon|}{\ln 2} \|p - q\|_1$$

for all p, q in this truncated simplex.

Applying this with $p = p_i(C_1)$ and $q = p_i(C_2)$, we get

$$|H(p_i(C_1)) - H(p_i(C_2))| \leq \frac{1 + |\ln \varepsilon|}{\ln 2} \|p_i(C_1) - p_i(C_2)\|_1.$$

Using the bound above,

$$|H(p_i(C_1)) - H(p_i(C_2))| \leq \frac{1 + |\ln \varepsilon|}{\ln 2} \cdot \frac{2}{\eta} \|c_i^{(1)} - c_i^{(2)}\|_1.$$

Finally, summing over $i = 1, \ldots, k$,

$$|r_E(C_1, Y) - r_E(C_2, Y)| = \left| \sum_{i=1}^{k} H(p_i(C_1)) - \sum_{i=1}^{k} H(p_i(C_2)) \right|$$

$$\leq \sum_{i=1}^{k} |H(p_i(C_1)) - H(p_i(C_2))| \leq \frac{2(1 + |\ln \varepsilon|)}{\eta \ln 2} \sum_{i=1}^{k} \|c_i^{(1)} - c_i^{(2)}\|_1.$$

Hence,

$$|r_E(C_1, Y) - r_E(C_2, Y)| \leq \frac{2(1 + |\ln \varepsilon|)}{\eta \ln 2} \|C_1 - C_2\|_{1,1}.$$

Using $\|C_1 - C_2\|_{1,1} \leq \sqrt{pk}\, \|C_1 - C_2\|_F$, we further get

$$|r_E(C_1, Y) - r_E(C_2, Y)| \leq \frac{2(1 + |\ln \varepsilon|)}{\eta \ln 2} \sqrt{pk}\, \|C_1 - C_2\|_F.$$

This proves the claim. □

A.16 Performance Across Different GNN Architectures

Dataset	r	Architecture	LAGC	ENGC
CORA	**0.1**	**GCN**	84.45 ± 0.18	86.00 ±1.05
		GAT	80.23 ± 0.25	81.37 ± 0.00
		APPNP	86.05 ± 0.41	87.85 ± 0.71
CITESEER	**0.1**	**GCN**	75.61 ± 0.62	78.65 ± 1.95
		GAT	72.72 ± 0.97	77.54 ± 0.01
		APPNP	76.40 ± 0.21	78.46 ± 0.01
PUBMED	**0.01**	**GCN**	80.91 ± 0.08	82.73 ± 0.36
		GAT	73.92 ± 0.20	74.68 ± 0.00
		APPNP	79.62 ± 0.61	79.55 ± 0.02

Table 5: Resulting Accuracy, where r is the Coarsening Ratio

A.17 Link Prediction Algorithm

Algorithm 3: Link Prediction using proposed ENGC

Input: $\mathcal{G}(\Theta, X, Y)$
Output: GNN model $\mathcal{G}_\theta$
1 Randomly initialize model parameter W^*;
2 Treat the existing edges as Positive Examples $\mathbf{P}$;
3 Randomly sample a set of edges to serve as negative examples $\mathbf{N}$;
4 Divide $\mathbf{P}$ and $\mathbf{N}$ into training and test sets;
5 Apply ENGC on train set to learn P; $P = C^\dagger$;
6 Update W^* by minimizing binary cross-entropy loss $\ell(GNN_{G_C}(W^*), Y'_{u\sim v})$;

A.18 Ablation result discussion

Reffering to table 3 Removing T_1 (log-det/spectral connectivity) yields the largest degradation (-8.59 points), followed by ablating T_6 (entropy regularization, -7.20) and T_2 (feature reconstruction, -6.39). This indicates that (i) maintaining a well-conditioned coarse connectivity through the log-det term, (ii) enforcing label coherence via $r_E(C, Y)$, and (iii) aligning coarse and fine features are most critical for downstream accuracy on Cora. In contrast, dropping T_4 and T_5 produces moderate declines (-3.58 and -3.47), consistent with their role as stabilizing regularizers. The effect of T_3 is comparatively small (-1.16). Overall, the ablation confirms that the principal gains stem from the spectral connectivity, entropy guided coarsening, and feature similarity components introduced in the ENGC objective.

A.19 Run-time Complexity:

Given an input graph with p nodes, E_1 edges, and a feature vector of size n for each node, the time complexity for node classification using a Graph Convolutional Network (GCN) with l layers is $\mathcal{O}(lp^2n + lpE_1n)$ Blakely et al. (2021).

The worst-case per iteration computational complexity of our proposed ENGC algorithm is $\mathcal{O}(p^2k)$ for learning a coarsened graph. However, when both coarsening and node classification are performed, the overall time complexity of our algorithm is $\mathcal{O}(p^2k + lk^2n + lkE_2n)$, where k represents the number of nodes in the coarsened graph, and E_2 is the number of edges in the coarsened graph. Given that ($p >> k$) and $E_1 >> E_2$, and choosing k such that $k < n$, the time complexity for coarsening and node classification becomes significantly lower compared to performing node classification solely on the original graph. The effectiveness of this approach is evident in Table A.19, demonstrating that the proposed ENGC algorithm is notably faster than baseline methods and exhibits similar time complexity compared to the FGC algorithm Kumar et al. (2023a;b).

Dataset(τ)	r = k/p	**GCOND**	**SCAL**	FGC	LAGC	ENGC	Whole dataset
CORA	0.05	329.86	27.76	1.71	1.55	1.72	2.86
CITESEER	0.05	331.33	56.21	2.15	2.03	1.90	5.24
PUBMED	0.05	202.04	54.09	19.81	20.35	17.98	58.85
CO-CS	0.05	1600.32	180.16	34.45	49.87	37.04	72.31

Table 6: This table summarizes the time (τ) in sec. required to perform coarsening and node classification for a coarsening ratio of 0.05. It is evident that the proposed ENGC is faster than state-of-the-art algorithms. Moreover, the time required to perform coarsening and node classification using the proposed methods is less than that required to perform node classification using the given original graph.