Graph Summarization for Preserving Spectral Characteristics

Houquan Zhou*[1,2], Shenghua Liu†[1,2], Huawei Shen[1,2], and Xueqi Cheng[1,2]

[1]CAS Key Laboratory of AI Security, Institute of Computing Technology, Chinese Academy of Sciences

[2]University of Chinese Academy of Sciences

Abstract

How does the graph change if we summarize it by merging nodes? How can we summarize the graph while preserving its spectral characteristics? Graph summarization aims to present a graph in a compact summary graph form while keeping its important structural information. Existing methods primarily focus on preserving the adjacency matrix. In contrast, spectral graph theory provides a powerful tool to describe the characteristics of a graph. In this paper, we propose a novel graph summarization method that preserves the spectral characteristics, including spectral moments and heat traces. We analyze the change of the spectral characteristics after summarization and design a simple yet effective summarization method based on agglomerative clustering. Our approach is extensively evaluated on real-world datasets. The experimental results show that our method excels in preserving the spectral characteristics and obtains better performance on the subsequent graph classification task.

Keywords: Graph Summarization, Spectral Distribution, Spectral Graph Theory

1 Introduction

Graph analysis is a fundamental task in many real-world applications, such as social network analysis, drug discovery, and recommendation systems. Generally, analyzing large graphs is a challenging task due to the high computational complexity. Graph summarization addresses this problem by finding a compact representation of a graph, typically in the form of a summary graph, while preserving its structural information. The resulting summary graph can serve as a proxy for the original graph and be utilized for further analysis.

The main questions in the graph summarization are what property to preserve and how to compare the property of the original graph and the summary graph. For the first question, most existing works focus on preserving the adjacency matrix as it is the most fundamental representation of a graph. However, the adjacency matrix is easily affected by perturbation and lacks robustness to noise. Additionally, some high-order properties, such as degree distribution, global connectivity are not directly reflected in the adjacency matrix.

On the other hand, spectral graph theory provides an alternative perspective for understanding a graph [5]. For instance, the spectral gap of a graph, i.e., the second-smallest eigenvalue of the Laplacian matrix, is closely associated with the graph connectivity [10]. Compared to discrete graph representations (adjacency matrix), spectral features are more robust to noise and perturbation. Due to these advantages, spectral features have been widely used in graph analysis [14, 26, 25]. Therefore, it is highly desirable to preserve the spectral characteristics of a graph during summarization.

There are some existing works that study the problem of preserving the spectral properties of graphs. [19, 18] consider preserving the spectral properties of a restricted eigenspace. [15] aims to minimize the spectral distance regarding the perturbation of eigenvalues. In this work, we study the problem of graph summarization for preserving the spectral characteristics of graphs. We focus on two types of spectral characteristics closely related to graph structure, spectral moment and heat trace. We show that the spectral moment and the heat trace of the summary graph is equivalent to that of the degree-preserving reconstructed graph, and build a connection between the spectral properties of the summary graph and the original graph. Furthermore, we conduct in-depth analysis of the loss of spectral properties in the graph summarization process and derive the upper bound of it to the loss of the normalized adjacency matrix. Based on the analysis, we design a novel graph summarization method for preserving the spectral characteristics of graphs based on agglomerative clustering. We perform extensive experiments on several synthetic

*Email: zhouhouquan18z@ict.ac.cn

†Corresponding Author. Email: liushenghua@ict.ac.cn

and real-world datasets and show that our method can effectively preserve the spectral characteristics of graphs and outperform the state-of-the-art methods.

In summary, the contribution of this work is as follows:

- We establish a connection between the spectral characteristics of the summary graph and the original graph via the degree-preserving reconstructed graph.
- We analyze the loss of spectral characteristics in the graph summarization and derive the upper bound of it to the loss of the normalized adjacency matrix.
- We propose a simple yet effective graph summarization method for preserving the spectral characteristics of graphs.
- We perform extensive experiments on multiple datasets and show that our method can effectively preserve the graph spectral characteristics.

Reproducibility: The code of our method is available at `https://github.com/HQJo/SpectralCharacteristicSumm`.

2 Related Work

2.1 Graph Summarization Graph summarization is a family of methods that aims to reduce the size of a graph while preserving its crucial structural information. The reconstruction error of the adjacency matrix is the most commonly employed metric for assessing graph summarization. It is defined as some norm (e.g., Frobenius norm) of the difference between the original adjacency matrix and the reconstructed one. The reconstructed graph is derived from the summary graph via a specific reconstruction method. For instance, k-Gs [17] aims to find a summary graph with at most k supernodes, such that the L1 reconstruction error is minimized. Riondato et al. [24] establish a connection between the geometric clustering problem and the graph summarization problem, and propose a polynomial-time approximate graph summarization method based on geometric clustering algorithms. Beg et al. [1] develop a randomized algorithm SAA-Gs using weighted sampling and count-min sketch [6] techniques to find promising node pairs efficiently. SpecSumm [20] reformulates the error minimization problem as a trace optimization problem and propose using the k-largest eigenvectors of the adjacency matrix to obtain the summary graph.

Besides the adjacency matrix, some works also consider other graph properties such as spectral properties. For instance, [19, 18] propose to preserve the principal eigenvectors and eigenspaces of the original Laplacian matrix. GraphZoom [8] proposes merging nodes based on their spectral similarity. [15] proposes using the spectral distance (i.e., absolute deviation of eigenvalues) to measure the difference between the original graph and the summary graph, and designs two summarization methods, MGC and SGC, to minimize the spectral distance.

2.2 Graph Feature Extraction Extracting representative features from graphs is a fundamental problem in graph analysis. It aims to map each graph to a low-dimensional vector space that can be utilized for downstream tasks. Existing methods can be roughly categorized into two main approaches: explicit constructing methods and implicit learning methods.

Constructing-based methods constructs features in an explicit way. These features may consider **(1) the basic descriptive properties of graphs**, including degrees, clustering coefficients, and centrality measures, etc. For example, NetSimile [2] extracts 7 features from each node, including the number of neighbors, average degree, number of edges in ego-network, clustering coefficient, average clustering coefficient, number of outgoing edges from ego-network, and number of neighbors of ego-network. Graph signatures are then generated using 5 aggregators like median, mean, standard deviation, skewness and kurtosis. **(2) spectral properties**. [7] gives a simple method employing the k smallest positive eigenvalues of Laplacian matrix as features. [14] employs the moment of spectral density distribution and proposes spectral moment to characterize the properties of graphs. NetLSD [26] proposes heat trace derived from the graph heat kernel as features. VNGE [3, 22, 11, 21] transfers the Von Neumann entropy in quantum information theory to graphs and proposes Von Neumann graph entropy as features. FINGER [4] develops efficient algorithms to approximate VNGE. SLaQ [27] enhances the efficiency of NetLSD and VNGE leveraging the stochastic Lanczos quadrature method in numerical linear algebra. **(3) high-level structural information**, such as the shortest path, random walk, and graphlets. This kind of methods are closely related to graph kernels [16]. Since construction-based methods are constructed explicitly, it is easy to interpret and explain.

Learning-based methods, on the other hand, leverage the great power of machine learning methods to learn representative features implicitly. The state-of-the-art methods are based on graph neural networks (GNNs). Representative methods include GIN [30], DGCNN [32], and DiffPool [31]. Learning-based methods typically adopt an end-to-end training framework, which lack interpretability compared to constructing-based methods.

Table 1: Notations used in this paper.

Notation	Description
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	Original graph with nodeset $\mathcal{V}$ and edgeset $\mathcal{E}$
$\mathbf{A}$, $\mathbf{D}$	Adjacency matrix and degree matrix of $\mathcal{G}$
$\mathcal{G}_s, \mathbf{A}_s$	Summary graph and its adjacency matrix
$\mathcal{G}_r, \mathbf{A}_r$	Reconstructed graph and its adjacency matrix
n, n_s	Sizes of $\mathcal{G}$ and $\mathcal{G}_s$
d_i, D_k	Degree of node v_i and supernode $\mathcal{S}_k$
$\boldsymbol{\mathcal{L}}/\boldsymbol{\mathcal{L}}_s/\boldsymbol{\mathcal{L}}_r$	Normalized Laplacian matrix of $\mathcal{G}/\mathcal{G}_s/\mathcal{G}_r$
$\boldsymbol{\mathcal{A}}/\boldsymbol{\mathcal{A}}_s/\boldsymbol{\mathcal{A}}_r$	Normalized adjacency matrix of $\mathcal{G}/\mathcal{G}_s/\mathcal{G}_r$

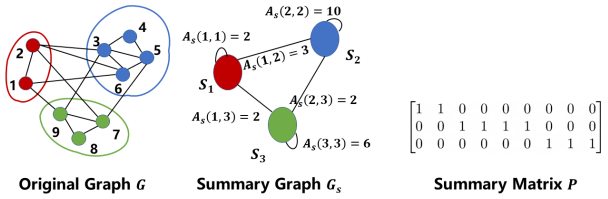


Figure 1: An example of graph summarization. The original graph with 9 nodes is summarized into a summary graph with 3 supernodes.

3 Backgrounds

3.1 Graph Summarization Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the original graph with nodeset $\mathcal{V}$, edgeset $\mathcal{E}$ and adjacency matrix $\mathbf{A}$. Graph summarization aims to find a summary graph $\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)$ such that $\mathcal{G}_s$ can well preserve the key properties of $\mathcal{G}$. The typical way is to aggregate nodes of $\mathcal{G}$ into supernodes of $\mathcal{G}_s$. This process can be formulated by a summarization matrix $\mathbf{P} \in \{0,1\}^{n_s \times n}$:

$$\mathbf{P}(k,i) = \begin{cases} 1, & \text{if node } v_i \text{ in supernode } \mathcal{S}_k \\ 0, & \text{otherwise} \end{cases} \tag{3.1}$$

The adjacency matrix of the summary graph $\mathbf{A}_s = \mathbf{P}\mathbf{A}\mathbf{P}^\top$ is the aggregation of edges in the original graph.

$$\mathbf{A}_s(k,l) = \sum_{v_i \in \mathcal{S}_k} \sum_{v_j \in \mathcal{S}_l} \mathbf{A}(i,j) \tag{3.2}$$

Fig. 1 shows an example of graph summarization.

To measure the quality of the summary graph, a reconstructed graph is obtained from $\mathcal{G}_s$ by some specific reconstruction method and is compared with the original graph $\mathcal{G}$. Here we introduce the degree-preserving reconstruction method [34]. It is based on the configuration model and the corresponding reconstructed graph has the same degree distribution as the original graph. The reconstruction process can be expressed using a reconstruction matrix $\mathbf{Q} \in \mathbb{R}^{n \times n_s}$:

$$\mathbf{Q}(k,i) = \begin{cases} \frac{d_i}{D_k}, & \text{if node } v_i \text{ in supernode } \mathcal{S}_k \\ 0, & \text{otherwise} \end{cases} \tag{3.3}$$

where $D_k = \sum_{v_i \in \mathcal{S}_k} d_i$ is the degree of supernode $\mathcal{S}_k$. The corresponding reconstructed adjacency matrix is $\mathbf{A}_r = \mathbf{Q}\mathbf{A}_s\mathbf{Q}^\top$.

$$\mathbf{A}_r(i,j) = \frac{d_i}{D_k}\mathbf{A}_s(k,l)\frac{d_j}{D_l} \quad (i \in \mathcal{S}_k, j \in \mathcal{S}_l) \tag{3.4}$$

3.2 Graph Spectral Characteristics Graph spectral characteristics describe the graph properties leveraging spectral graph theory. These characteristics are based on the eigenvalues and eigenvectors of the adjacency matrix or the Laplacian matrix. Recently, there has been a growing trend in modelling the graph through the global distribution of its eigenvalues, a.k.a., the spectral density or density of states (DOS) [9, 25, 12, 4]. The spectral distribution is defined in the form of generalized function [9]:

$$p(\lambda) = \frac{1}{n}\sum_{i=1}^{n} \delta(\lambda - \lambda_i) \tag{3.5}$$

where δ is the Dirac delta function.

By treating eigenvalues as a distribution, probabilistic tools can be leveraged to characterize various graph properties. Two representative examples are the spectral moment [14] and the heat trace [26].

[14] proposes the spectral moment $\{m_k\}_{k=1,2,\ldots}$, originated from the moment of the spectral density distribution, to characterize graph properties. It is equivalent to the average return probabilities of k-step random walk.

$$m_k(\mathcal{G}) = \mathbb{E}[\mu^k] = \frac{1}{n}\sum_{i=1}^{n} \mu_i^k = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{D}^{-1}\mathbf{A})^k(i,i)$$

Here μ_i is the i-th eigenvalue of the random walk matrix $\mathbf{D}^{-1}\mathbf{A}$ (and the normalized adjacency matrix $\boldsymbol{\mathcal{A}}$). It is shown that the spectral moments are closely related to graph structure and various graph properties including degree distribution and clustering coefficient [14, 13, 23].

[26] proposes heat trace, defined as the trace of the heat kernel matrix $\exp(-t\boldsymbol{\mathcal{L}})$ to construct graph signatures for graph classification. It is closely related to the moment generating function of the spectral distribution.

$$\begin{aligned} h_t(\mathcal{G}) &= \operatorname{tr}(\exp(-\boldsymbol{\mathcal{L}}t)) = \sum_{i=1}^{n} \exp(-\lambda_i t) \\ &= \exp(-t)\sum_{i=1}^{n} \exp(\mu_i t) \end{aligned} \tag{3.6}$$

Here λ_i is the i-th eigenvalue of the normalized Laplacian matrix $\boldsymbol{\mathcal{L}}$.

By expanding $\exp(-\lambda t)$ as series, we have

$$h_t(\mathcal{G}) = \exp(-t)\sum_{i=1}^{n}\sum_{k=0}^{\infty}\frac{t^k}{k!}\mu_i^k = n\cdot\sum_{k=0}^{\infty}\exp(-t)\frac{t^k}{k!}m_k(\mathcal{G}) \tag{3.7}$$

That is, **the heat trace can be regarded as a weighted sum of spectral moments**. The weight of the k-th spectral moment $\exp(-t)\frac{t^k}{k!}$ is the probability of a Poisson random variable with parameter t taking value k. Thus, the heat trace can be viewed as the average return probability of Poisson-length random walks.

Spectral moments and heat trace can be further unified in a general form using a test function $f(\cdot)$ [9].

$$\int f(\lambda)p(\lambda) = \mathrm{tr}(f(\boldsymbol{\mathcal{L}})) \tag{3.8}$$

For spectral moments, $f(\lambda) = \frac{1}{n}\cdot\lambda^k$. For heat trace, $f(\lambda) = \exp(-\lambda t)$.

4 Proposed Methods

In this section, we present our graph summarization method aimed at preserving graph spectral characteristics. We first investigate the spectral characteristics of summary graphs, and build a connection between the spectral characteristics of the summary graph and the original graph via the degree-preserving reconstructed graph. We then analyze the loss of spectral characteristics after merging two nodes and derive an upper bound of it. Based on the insights gained from the analysis, we propose a simple graph summarization method based on agglomerative clustering to preserve the spectral characteristics of graphs.

4.1 Spectral characteristics of summary graphs

To compare the spectral characteristics of the summary graph and the original graph, we first analyze the spectral characteristics of the summary graph. As we show next, **both spectral moments and heat trace of the summary graph is related to that of the degree-preserving reconstructed graph**.

THEOREM 4.1. *The trace of the k-th power of the normalized adjacency matrix of the summary graph is exactly that of the degree-preserving reconstructed graph(defined in Eq. (3.4)).*

$$\mathrm{tr}(\boldsymbol{\mathcal{A}}_s^k) = \mathrm{tr}(\boldsymbol{\mathcal{A}}_r^k) \qquad k\in\mathbb{N}^+ \tag{4.9}$$

Proof. For convenience, we first introduce matrix $\mathbf{R}\in\mathbb{R}^{n\times n_s}$ defined as:

$$\mathbf{R}(i,k) = \begin{cases}\sqrt{\frac{d_i}{D_k}}, & \text{if } v_i\in\mathcal{S}_k\\ 0, & \text{otherwise}\end{cases} \tag{4.10}$$

Note that $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$ and thus the columns of $\mathbf{R}$ are orthonormal. As shown in [33], the normalized adjacency matrix of the summary graph and the reconstructed graph are related by:

$$\boldsymbol{\mathcal{A}}_r^k = \mathbf{R}\boldsymbol{\mathcal{A}}_s^k\mathbf{R}^\top \quad (k\in\mathbb{N}^+) \tag{4.11}$$

Hence,

$$\mathrm{tr}(\boldsymbol{\mathcal{A}}_r^k) = \mathrm{tr}(\mathbf{R}\boldsymbol{\mathcal{A}}_s^k\mathbf{R}^\top) = \mathrm{tr}(\boldsymbol{\mathcal{A}}_s^k\mathbf{R}^\top\mathbf{R}) = \mathrm{tr}(\boldsymbol{\mathcal{A}}_s^k)$$

That is, *the trace of the k-th power of the normalized adjacency matrix of the summary graph is exactly that of the degree-preserving reconstructed graph.* ∎

As a result, the un-normalized k-th spectral moment of the summary graph is exactly that of the degree-preserving reconstructed graph.

$$n_s\cdot m_k(\mathcal{G}_s) = \mathrm{tr}((\mathbf{D}_s^{-1}\mathbf{A}_s)^k) = \mathrm{tr}(\boldsymbol{\mathcal{A}}_s^k) = \mathrm{tr}(\boldsymbol{\mathcal{A}}_r^k) = n\cdot m_k(\mathcal{G}_r)$$

Thus, we can use $m_k(\mathcal{G}_r)$ as a proxy of $m_k(\mathcal{G}_s)$ to compare the spectral moments of the summary graph and the original graph.

Similar conclusion can be easily drawn for the heat trace.

THEOREM 4.2. *The heat trace of the summary graph are different from that of the degree-preserving reconstructed graph by a constant arising from the size difference.*

Proof.

$$\begin{aligned} h_t(\mathcal{G}_s) &= n_s\cdot\sum_{k=0}^{\infty}\exp(-t)\frac{t^k}{k!}m_k(\mathcal{G}_s)\\ &= n_s\cdot\left(\exp(-t)+\sum_{k=1}^{\infty}\exp(-t)\frac{t^k}{k!}m_k(\mathcal{G}_s)\right)\\ &= n_s\cdot\exp(-t) + n\cdot\sum_{k=1}^{\infty}\exp(-t)\frac{t^k}{k!}m_k(\mathcal{G}_r)\\ &= n\cdot\sum_{k=0}^{\infty}\exp(-t)\frac{t^k}{k!}m_k(\mathcal{G}_r) - \exp(-t)(n-n_s)\\ &= h_t(\mathcal{G}_r) - \exp(-t)(n-n_s) \end{aligned}$$

Thus, *the heat trace of the summary graph and that of the reconstructed graph are only different by a constant arising from the size difference.* ∎

4.2 Loss of spectral characteristics As shown in Theorem 4.1 and Theorem 4.2, the spectral characteristics of summary graphs are closely related to that of reconstructed graph. Therefore, we aim to utilize the latter as a surrogate for the former and compare it with that of original graphs. Since the two spectral characteristics originate from the spectral distribution, our objective is to minimize the discrepancy of the spectral distribution to minimize the loss of spectral characteristics.

By [9], the Wasserstein distance between the spectral distribution of the original graph and that of the reconstructed graph is upper bounded by the Frobenius norm of the difference of the normalized adjacency matrix.

THEOREM 4.3. (C.F. THEOREM 4.2 IN [9])

$$W_1(\mathcal{G}, \mathcal{G}_r) \leq \|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_r\|_F$$

Thus, we can minimize the Frobenius norm of the difference of the normalized adjacency matrix to minimize the spectral difference. We first reformulate the Frobenius norm as:

LEMMA 4.1.

$$\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_r\|_F^2 = \|\boldsymbol{\mathcal{A}}\|_F^2 - \|\boldsymbol{\mathcal{A}}_r\|_F^2 \tag{4.12}$$

Proof. To prove this, we first show that $\boldsymbol{\mathcal{A}}_s = \mathbf{R}^\top \boldsymbol{\mathcal{A}} \mathbf{R}$ (where $\mathbf{R}$ is defined in Eq. (4.10)).

$$\begin{aligned}
\mathbf{R}^\top \boldsymbol{\mathcal{A}} \mathbf{R}(k,l) &= \sum_{i,j} \mathbf{R}(i,k) \boldsymbol{\mathcal{A}}(i,j) \mathbf{R}(j,l) \\
&= \sum_{v_i \in \mathcal{S}_k, v_j \in \mathcal{S}_l} \sqrt{\frac{d_i}{D_k}} \frac{\boldsymbol{\mathcal{A}}(i,j)}{\sqrt{d_i d_j}} \sqrt{\frac{d_j}{D_l}} \\
&= \sum_{v_i \in \mathcal{S}_k, v_j \in \mathcal{S}_l} \frac{1}{\sqrt{D_k}} \boldsymbol{\mathcal{A}}(i,j) \frac{1}{\sqrt{D_l}} \\
&= \frac{1}{\sqrt{D_k}} \boldsymbol{\mathcal{A}}_s(k,l) \frac{1}{\sqrt{D_l}} = \boldsymbol{\mathcal{A}}_s(k,l)
\end{aligned} \tag{4.13}$$

And,

$$\begin{aligned}
\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_r\|_F^2 &= \operatorname{tr}((\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_r)^\top (\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_r)) \\
&= \operatorname{tr}(\boldsymbol{\mathcal{A}}^2) + \operatorname{tr}(\boldsymbol{\mathcal{A}}_r^2) - 2\operatorname{tr}(\boldsymbol{\mathcal{A}} \boldsymbol{\mathcal{A}}_r) \\
&= \operatorname{tr}(\boldsymbol{\mathcal{A}}^2) + \operatorname{tr}(\boldsymbol{\mathcal{A}}_r^2) - 2\operatorname{tr}(\boldsymbol{\mathcal{A}} \mathbf{R} \boldsymbol{\mathcal{A}}_s \mathbf{R}^\top) \\
&= \operatorname{tr}(\boldsymbol{\mathcal{A}}^2) + \operatorname{tr}(\boldsymbol{\mathcal{A}}_r^2) - 2\operatorname{tr}(\mathbf{R}^\top \boldsymbol{\mathcal{A}} \mathbf{R} \boldsymbol{\mathcal{A}}_s) \\
&= \operatorname{tr}(\boldsymbol{\mathcal{A}}^2) + \operatorname{tr}(\boldsymbol{\mathcal{A}}_s^2) - 2\operatorname{tr}(\boldsymbol{\mathcal{A}}_s^2) \\
&= \operatorname{tr}(\boldsymbol{\mathcal{A}}^2) - \operatorname{tr}(\boldsymbol{\mathcal{A}}_s^2) \\
&= \|\boldsymbol{\mathcal{A}}\|_F^2 - \|\boldsymbol{\mathcal{A}}_s\|_F^2
\end{aligned} \tag{4.14}$$

□

Algorithm 1 Graph Summarization for Preserving Spectral Characteristic.

Input: Input graph $\mathcal{G}$, summary size k
Output: $\mathcal{G}_r$

```
G_r ← G
M ← D^{-1} A D^{-1/2}
G_r ← AgglomerativeClustering(M, k)
return G_r
```

Given Lemma 4.1, we only need to analyze the difference of $\|\boldsymbol{\mathcal{A}}\|_F^2$ and $\|\boldsymbol{\mathcal{A}}_r\|_F^2$, which is much more concise and easier to analyze. We will analyze the effect of merging two nodes. In short, by merging two node a and b, the difference of $\|\boldsymbol{\mathcal{A}}_r\|_F^2$ and $\|\boldsymbol{\mathcal{A}}\|_F^2$ is upper bounded by:

THEOREM 4.4.

$$\|\boldsymbol{\mathcal{A}}\|_F^2 - \|\boldsymbol{\mathcal{A}}_r\|_F^2 \leq H(d_a, d_b) \left\| \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-1} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \right\|_2^2 \tag{4.15}$$

where $H(d_a, d_b)$ is the harmonic mean of d_a and d_b.

Proof. See appendix. □

The right-hand side of Eq. (4.15) consists of two terms. The first term $H(a,b)$ is the harmonic mean of d_a and d_b. The second term is the Euclidean distance between the a-row and b-row of the matrix $\mathbf{D}^{-1}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$. This gives us two implications:

- Nodes with small degrees have higher priority to be merged together since $H(a,b) \geq \min\{d_a, d_b\}$. To make sure $H(a,b)$ is small, we need to make sure d_a and d_b are not too large.
- The a-row and b-row of $\mathbf{D}^{-1}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$ matrix should be close to minimize the second term. This aligns with the intuition that nodes with similar neighbors should be merged together.

4.3 Algorithm Based on the aforementioned analysis, we propose a simple yet effective graph summarization method. Our method adopts an agglomerative clustering strategy. Initially, each node is treated as a supernode containing only itself. At each step, the method merges two supernodes with the highest priority. According to Eq. (4.15), we define the priority of supernode pair (a,b) as the ℓ_2 norm of the difference between the a-row and b-row of $\mathbf{D}^{-1}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$ matrix. The process is repeated until the number of supernodes is reduced to the desired summary size. We implement the process using agglomerative clustering with ward linkage [29] and use heap data structure to maintain the priority of supernode pairs and improve the efficiency. The detailed algorithm is shown in Alg. 1.

Table 2: Dataset statistics.

Dataset	#Graphs	#Classes	Avg. Size
MUTAG	188	2	17.93
PTC	344	2	25.53
ENZYMES	600	6	32.63
PROTEINS	1,113	2	39.06
IMDB-B	1,000	2	19.77
IMDB-M	1,500	3	13.00
REDDIT-B	2,000	2	429.6
REDDIT-M	5,000	5	508.5

5 Experiments

In this section, we perform experiments to answer the following questions:

- Q1: How well does the proposed method preserve the spectral characteristics compared to baselines?
- Q2: How does the proposed method affect the performance of graph classification compared to baselines?
- Q3: How efficient is the proposed method compared to baselines?

Datasets We use 8 real-world datasets from bioinformatics and social networks. MUTAG, PTC, ENZYMES and PROTEINS are chemical compounds and proteins represented as graphs. IMDB-B, IMDB-M, REDDIT-B and REDDIT-M contains graphs sampled from social networks. The detailed statistics of the datasets are shown in Table 2.

Baselines We compare our method with the following baselines:

- Spectral Clustering [28]. Spectral clustering is a classical graph clustering method based on the eigenvectors of the Laplacian matrix.
- MGC [15]. MGC adopts the iterative merging framework and performs multi-level summarization according to the connectivity of nodes.
- SGC [15]. SGC utilizes eigenvectors with the eigenvalues corresponding to the head and tail eigenvalues and runs k-means clustering on the eigenvectors to summarize the graph.
- GraphZoom [8]. GraphZoom is a spectral coarsening method merging nodes with high similarity in the spectral representations.
- Local Variation [18]. Local variation method considers spectral similarity on a restricted subspace and summarize the graph by contracting sets of nodes.

Implementation We implement our method in Python and perform experiments on a machine with Intel Xeon E5-2640 v4 CPU and 128 GB RAM. For baselines, we use the source codes provided by the authors.

5.1 Q1. Spectral Characteristics Preservation

In this experiment, we evaluate the effectiveness of our method in preserving the spectral characteristics of graphs.

Settings We calculate eigenvalues, spectral moments and heat trace of the original graph and the summary graph, and compare their difference as the metric of spectral characteristics' preservation. Smaller difference indicates better preservation of spectral characteristics. For eigenvalues, we report the mean absolute difference as "eigenvalue loss". For spectral moments, we calculate the first 4 order moments (following [14]) of the original graph and the summary graph, and report the mean absolute difference as "spectral moment loss". For heat trace, the time parameter t is from 0.025 to 2.5 and contains 100 points distributed uniformly in the logarithmic scale. We report the mean absolute difference as "heat trace loss".

Results The results are shown in Table 3. From the results, we have several observations. On small datasets like MUTAG and PTC, MGC achieves the smallest loss in all metrics while our methods achieve the second-best or the third-best performance. For larger datasets, our method achieves the smallest loss in spectral characteristics in most cases. On two REDDIT datasets, our method obtains much smaller loss in all metrics than other methods. For example, on REDDIT-M dataset, our method gets *11.68 times smaller* eigenvalue loss than the second-best method. We also give the average rank of each method at the bottom of Table 3. Our method achieves the best rank in eigenvalue loss and heat trace loss and the second-best rank in spectral moment loss. These results demonstrate the effectiveness of our method in preserving the spectral characteristics of graphs.

5.2 Q2. Performance of Graph Classification

In this experiment, we aim to assess the preservation of graph spectral properties by evaluating the performance of downstream graph classification tasks. A better classification performance indicates better preservation of spectral properties.

Settings For each dataset, we apply multiple summarization methods to summarize each graph in the

Table 3: Spectral characteristic loss of different methods on multiple datasets. Three types of metrics, i.e., eigenvalue, spectral moment, and heat trace, are used to measure the spectral characteristic loss. The best and the second-best scores are highlighted in bold and underline, respectively. '-' denotes running out of time (> 6 hours).

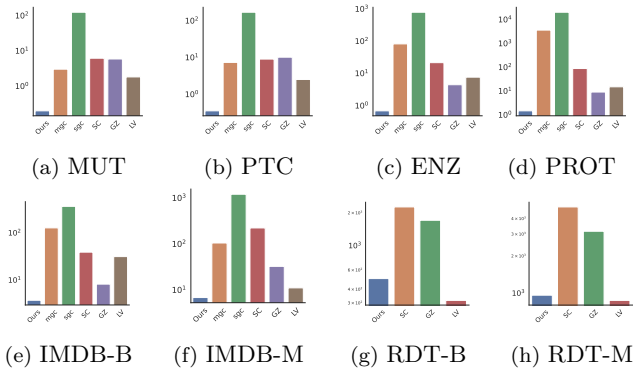
Dataset	Loss	MGC	SGC	Spectral Clustering	GraphZoom	Local Variation	Ours
MUTAG	eigenvalue loss	**0.206**	0.218	0.325	0.309	0.280	0.220
	spectral moment	**2.456**	2.805	4.320	3.985	4.049	2.609
	heat trace	**0.193**	0.229	0.351	0.254	0.339	0.228
PTC	eigenvalue	**0.211**	0.225	0.319	0.299	0.287	0.259
	spectral moment	**2.153**	2.714	3.933	3.449	3.947	2.368
	heat trace	**0.184**	0.238	0.358	0.232	0.352	0.222
ENZYMES	eigenvalue	0.146	0.150	0.216	0.197	0.172	**0.145**
	spectral moment	**4.102**	4.300	5.265	4.769	4.876	4.129
	heat trace	0.235	0.280	0.449	0.328	0.482	**0.225**
PROTEINS	eigenvalue	0.148	0.152	0.216	0.199	0.173	**0.147**
	spectral moment	**5.225**	5.499	6.773	6.183	6.660	5.288
	heat trace	0.317	0.358	0.575	0.418	0.629	**0.316**
IMDB-B	eigenvalue	0.681	0.721	0.997	1.358	0.793	**0.679**
	spectral moment	2.243	2.281	2.443	2.382	2.245	**2.240**
	heat trace	0.233	0.194	0.210	**0.115**	0.223	0.227
IMDB-M	eigenvalue	0.608	0.629	0.776	1.203	0.747	**0.607**
	spectral moment	1.355	1.370	1.440	1.366	1.601	**1.354**
	heat trace	0.137	0.122	0.126	**0.049**	0.152	0.145
REDDIT-B	eigenvalue	-	-	0.216	0.233	0.204	**0.042**
	spectral moment	-	-	72.860	64.699	63.862	**45.412**
	heat trace	-	-	3.741	3.576	3.296	**0.631**
REDDIT-M	eigenvalue	-	-	0.219	0.245	0.222	**0.019**
	spectral moment	-	-	72.860	64.699	63.862	**45.412**
	heat trace	-	-	3.741	3.576	3.296	**0.631**
Avg Rank	eigenvalue	1.667	2.667	4.875	5.000	3.625	**1.500**
	spectral moment	**1.333**	3.333	5.250	3.750	4.250	1.500
	heat trace	2.833	2.667	4.375	3.000	4.500	**2.250**

Table 4: Graph classification accuracy on summary graphs generated by different methods. The best and the second-best scores are highlighted in bold and underline, respectively. Orig is the classification accuracy on the original graphs. '-' denotes running out of time (> 6 hours).

	MUTAG	PTC	ENZYMES	PROTEINS	IMDB-B	IMDB-M	REDDIT-B	REDDIT-M
Orig	87.78	60.19	38.51	74.46	72.00	48.00	85.32	45.73
MGC	86.72	60.27	29.66	74.59	70.86	46.84	-	-
SGC	87.20	59.86	29.77	73.79	68.67	47.36	-	-
Spectral Clustering	85.44	57.88	**32.03**	73.95	68.02	**48.37**	81.26	41.99
GraphZoom	82.50	57.33	28.92	74.61	65.97	41.37	81.45	42.60
Local Variation	83.72	58.59	29.77	73.48	69.65	47.53	82.67	42.53
Ours	**88.25**	**61.19**	30.52	**74.82**	**71.96**	46.71	**84.75**	**44.76**

dataset. The summarization ratio is set to 0.5 for all datasets, meaning that the size of the summary graph is 50% of the original graph size. Additionally, to ensure that the summary graph is not too small and remains meaningful, we set the minimum size of the summary graph to 5. We evaluate the classification performance on the original graphs and the summary graphs generated by different methods. By comparing the performance, we can evaluate the quality of the summarized graphs.

To extract graph features for classification, we utilize the heat-trace-based method NetLSD [26] with default parameters. We employ SVM as the classifier and perform grid search to find the best hyperparameters of SVM for each method. For the linear kernel, the range of parameter C is $[10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2]$; for the RBF kernel, the range of parameter C is the same as the linear kernel, and the range of parameter γ is $[10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3]$. To evaluate the classification performance, we use 10-fold cross validation



(a) MUT (b) PTC (c) ENZ (d) PROT

(e) IMDB-B (f) IMDB-M (g) RDT-B (h) RDT-M

Figure 2: Running time of different methods on multiple datasets. 'GZ', 'SC', and 'LV' are short for GraphZoom, Spectral Clustering, and Local Variation, respectively.

and report the average accuracy over 10 runs.

Results The classification results are presented in Table 4. Our method achieves the highest classification accuracy on all dataset except ENZYMES and IMDB-MULTI. Spectral clustering (SC) performs the best on ENZYMES dataset where our method attains the second-best performance. Notably, on some dataset (MUTAG, PTC and PROTEINS), our accuracy scores are even higher than the classification accuracy scores on original graphs. These results demonstrate that our method can preserve the spectral properties of graphs well and thus improve the performance of graph classification task.

5.3 Q3. Efficiency

In this experiment, we evaluate the efficiency of our method compared to baselines.

Settings Similar to the previous experiment, we apply multiple summarization methods to summarize each graph in the dataset. The summarization ratio is set to 0.5 for all datasets, meaning that the size of the summary graph is 50% of the original graph size. We report the average runtime of each method over 10 runs.

Results The results are shown in Fig. 2. Our method achieves the best efficiency on all datasets except two REDDIT datasets. Compared to MGC, which also adopts the iterative merging framework, our method is more efficient. That is because our method utilizes heap to maintain the similarity between nodes and thus achieves better efficiency. Note that our method only use the fundamental agglomerative clustering algorithm, and further optimization is possible.

6 Conclusion and Discussion

In this work, we study the problem of preserving graph spectral characteristics (i.e., spectral moment and heat trace) in graph summarization. We perform analysis on the loss of spectral characteristics after summarization and relate it to the loss of the normalized adjacency matrix. Based on the analysis, we propose a simple yet effective graph summarization method based on agglomerative clustering. We validate the analysis on 8 real-world datasets and the effectiveness of our method on preserving the spectral characteristics of graphs.

There are some interesting directions for future work. **(1) More spectral characteristics**. In this work, we only investigate two kinds of spectral characteristics, spectral moment and heat trace, which can be seen as the moment and the moment generating function of the spectral distribution. Based on the spectral distribution, more spectral characteristics can be defined. We hope more spectral characteristics can be unified into this framework and more general conclusion can be drawn for general spectral characteristics. **(2) Combine with the attribute information**. The spectral characteristics investigated in this work are merely based on the structural information of the graph. However, real-world graphs may contain rich attribute information. It would be useful to study combining the attribute information with the structural information together in graph summarization. **(3) More comprehensive analysis**. The summary graph can serve as a proxy for the original graph in many graph analysis tasks. It would be interesting to study how to utilize the summary graph to perform more comprehensive analysis with theoretical guarantees.

Acknowledgments

This paper is partially supported by the National Science Foundation of China under Grant No.U1911401, U21B2046, and 6237075198.

References

[1] M. A. Beg, M. Ahmad, A. Zaman, and I. Khan, *Scalable approximation algorithm for graph summarization*, in PAKDD, 2018, pp. 502–514.

[2] M. Berlingerio, D. Koutra, T. Eliassi-Rad, and C. Faloutsos, *Network similarity via multiple social theories*, in ASONAM, 2013, pp. 1439–1440.

[3] S. L. Braunstein, S. Ghosh, and S. Severini, *The laplacian of a graph as a density matrix: a basic combinatorial approach to separability of mixed states*, Annals of Combinatorics, 10 (2006), pp. 291–317.

[4] P.-Y. Chen, L. Wu, S. Liu, and I. Rajapakse, *Fast incremental von neumann graph entropy computation: Theory, algorithm, and applicaggtions*, in ICML, PMLR, 2019, pp. 1091–1101.

[5] F. R. Chung, *Spectral graph theory*, vol. 92, American Mathematical Soc., 1997.

[6] G. Cormode and S. Muthukrishnan, *An improved data stream summary: the count-min sketch and its applications*, Journal of Algorithms, 55 (2005), pp. 58–75.

[7] N. de Lara and E. Pineau, *A simple baseline algorithm for graph classification*, Relational Representation Learning Workshop, NIPS 2018, (2018).

[8] C. Deng, Z. Zhao, Y. Wang, Z. Zhang, and Z. Feng, *Graphzoom: A multi-level spectral approach for accurate and scalable graph embedding*, in ICLR, 2019.

[9] K. Dong, A. R. Benson, and D. Bindel, *Network density of states*, in KDD, 2019, pp. 1152–1161.

[10] M. Fiedler, *Algebraic connectivity of graphs*, Czechoslovak mathematical journal, 23 (1973), pp. 298–305.

[11] L. Han, F. Escolano, E. R. Hancock, and R. C. Wilson, *Graph characterizations from von neumann entropy*, Pattern Recognition Letters, 33 (2012), pp. 1958–1967.

[12] L. Huang, A. J. Graven, and D. Bindel, *Density of states graph kernels*, in SDM, SIAM, 2021, pp. 289–297.

[13] S. Jin, H. Tian, J. Li, and R. Zafarani, *A spectral representation of networks: The path of subgraphs*, in KDD, 2022, pp. 698–708.

[14] S. Jin and R. Zafarani, *The spectral zoo of networks: Embedding and visualizing networks with spectral moments*, in KDD, 2020, pp. 1426–1434.

[15] Y. Jin, A. Loukas, and J. JaJa, *Graph coarsening with preserved spectral properties*, in International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 4452–4462.

[16] N. M. Kriege, F. D. Johansson, and C. Morris, *A survey on graph kernels*, Applied Network Science, 5 (2020), pp. 1–42.

[17] K. LeFevre and E. Terzi, *Grass: Graph structure summarization*, in SDM, SIAM, 2010, pp. 454–465.

[18] A. Loukas, *Graph reduction with spectral and cut guarantees.*, J. Mach. Learn. Res., 20 (2019), pp. 1–42.

[19] A. Loukas and P. Vandergheynst, *Spectrally approximating large graphs with smaller graphs*, in ICML, PMLR, 2018, pp. 3237–3246.

[20] A. Merchant, M. Mathioudakis, and Y. Wang, *Graph summarization via node grouping: A spectral algorithm*, in WSDM, 2022.

[21] G. Minello, L. Rossi, and A. Torsello, *On the von neumann entropy of graphs*, Journal of Complex Networks, 7 (2019), pp. 491–514.

[22] F. Passerini and S. Severini, *Quantifying complexity in networks: the von neumann entropy*, IJATS, 1 (2009), pp. 58–67.

[23] V. M. Preciado and A. Jadbabaie, *Moment-based spectral analysis of large-scale networks using local structural information*, IEEE/ACM Transactions on Networking, 21 (2012), pp. 373–382.

[24] M. Riondato, D. García-Soriano, and F. Bonchi, *Graph summarization with quality guarantees*, Data mining and knowledge discovery, 31 (2017), pp. 314–349.

[25] S. Sawlani, L. Zhao, and L. Akoglu, *Fast attributed graph embedding via density of states*, in ICDM, IEEE, 2021, pp. 559–568.

[26] A. Tsitsulin, D. Mottin, P. Karras, A. Bronstein, and E. Müller, *Netlsd: hearing the shape of a graph*, in KDD, 2018, pp. 2347–2356.

[27] A. Tsitsulin, M. Munkhoeva, and B. Perozzi, *Just slaq when you approximate: Accurate spectral distances for web-scale graphs*, in WWW, 2020, pp. 2697–2703.

[28] U. Von Luxburg, *A tutorial on spectral clustering*, Statistics and computing, 17 (2007), pp. 395–416.

[29] J. H. Ward Jr, *Hierarchical grouping to optimize an objective function*, Journal of the American statistical association, 58 (1963), pp. 236–244.

[30] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, *How powerful are graph neural networks?*, in ICLR, 2019.

[31] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, and J. Leskovec, *Hierarchical graph representation learning with differentiable pooling*, NeurIPS, 31 (2018).

[32] M. Zhang, Z. Cui, M. Neumann, and Y. Chen, *An end-to-end deep learning architecture for graph classification*, in AAAI, vol. 32, 2018.

[33] H. Zhou, S. Liu, D. Koutra, H. Shen, and X. Cheng, *A provable framework of learning graph embeddings via summarization*, in AAAI, vol. 37, 2023, pp. 4946–4953.

[34] H. Zhou, S. Liu, K. Lee, K. Shin, H. Shen, and X. Cheng, *Dpgs: Degree-preserving graph summarization*, in SDM, SIAM, 2021, pp. 280–288.