# A Sparse-Motif Ensemble Graph Convolutional Network against Over-smoothing

**Xuan Jiang**[1] , **Zhiyong Yang**[1] , **Peisong Wen**[1,2] , **Li Su**[1,3*] and **Qingming Huang**[1,2,3*]

[1]School of Computer Science and Tech., University of Chinese Academy of Sciences, Beijing,China
[2]Key Lab of Intell. Info. Process., Institute of Computing Technology, CAS, Beijing, China
[3]Peng Cheng Laboratory, Shenzhen, China

{jiangxuan20, wenpeisong20}@mails.ucas.ac.cn, {yangzhiyong21, suli, qmhuang}@ucas.ac.cn

## Abstract

The over-smoothing issue is a well-known challenge for Graph Convolutional Networks (GCN). Specifically, it is often observed that increasing the depth of GCN ends up in a trivial embedding subspace where the difference among node embeddings belonging to the same cluster tends to vanish. This paper believes that the main cause lies in the limited diversity along the message passing pipeline. Inspired by this, we propose a **S**parse-**M**otif **E**nsemble **G**raph **C**onvolutional **N**etwork (SMEGCN). We argue that merely employing the original graph Laplacian as the spectrum of the graph cannot capture the diversified local structure of complex graphs. Hence, to improve the diversity of the graph spectrum, we introduce local topological structures of complex graphs into GCN by employing the so-called graph motifs or the small network subgraphs. Moreover, we find that the motif connections are much denser than the edge connections, which might converge to an all-one matrix within a few times of message-passing. To fix this, we first propose the notion of sparse motif to avoid spurious motif connections. Subsequently, we propose a hierarchical motif aggregation mechanism to integrate the graph spectral information from a series of different sparse-motif message passing paths. Finally, we conduct a series of theoretical and experimental analyses to demonstrate the superiority of the proposed method.

## 1 Introduction

Nowadays, graph structure has emerged as one of the major means to express real-world data such as knowledge base, molecules, social networks, paper citing, where a large set of nodes are organized in an irregular and complicated manner. Consequently, how to learn effectively on graphs becomes an urgent problem. A natural solution is to resort to the recent success of deep learning. This idea brings about a class of neural networks known as Graph Neural Networks (GNN). This paper mainly focuses on a major branch of GNN called Graph Convolutional Network (GCN). The main idea of GCN is to extend the convolution operator on the euclidean image data to non-euclidean graph data. Hitherto, GCN has grown up as a popular topic in the machine learning community and has been applied to a wide range of applications including protein structure prediction [Shen and others, 2021; Tsubaki *et al.*, 2019], molecular fingerprint learning [Shui and Karypis, 2020; Rahaman and Gagliardi, 2020], visual reasoning [Gao *et al.*, 2020; Narasimhan *et al.*, 2018], traffic prediction [Jiang and Luo, 2021; Xie *et al.*, 2020], and social network analysis [Liu *et al.*, 2021; Guo and Wang, 2020].

A key challenge for GCN is that it fails to perform well when its depth grows increasingly. Such an observation significantly contradicts the well-known empirical results of Convolutional Neural Networks (CNN), where increasing the depth of a network often leads to a sharp performance gain. This is because the expressive power of GCN is often hindered by the notorious over-smoothing issue, which is because that graph convolution defined in GCN is essentially a Laplacian smoothing operator. As shown in [Huang *et al.*, 2020], after repeatedly applying the Laplacian smoothing in GCN many times, the features of all nodes in a (connected) graph would converge to similar values.

Recently, a series of studies have arisen to tackle the over-smoothing problem. However, the vast majority of the literature captures the spectral information only from the Laplacian of the original graph [Kipf and Welling, 2017; Wu *et al.*, 2019; Xu *et al.*, 2018]. Since over-smoothing is a natural property of Laplacian smoothing, we argue that over-smoothing might be mitigated by introducing a series of diversified representations of the graph spectrum. Moreover, it is noteworthy that the inputs of GCNs are often complex networks. It is well-known that [Benson *et al.*, 2016] the local topological structure of such complex networks could be captured effectively by the so-called motifs, or the small network subgraphs. On the other hand, such information might be ignored if only the original graph Laplacian is employed. Inspired by this, our goal in this paper is to mitigate the over-smoothing issue through the diversity introduced by graph motifs. Specifically, we propose a Sparse-Motif Ensemble Graph Convolutional Network (SMEGCN)[1]. In a nutshell, our contribution is three-fold:

*Corresponding Author

[1]Code is available at: https://github.com/BoloJX/SMEGCN

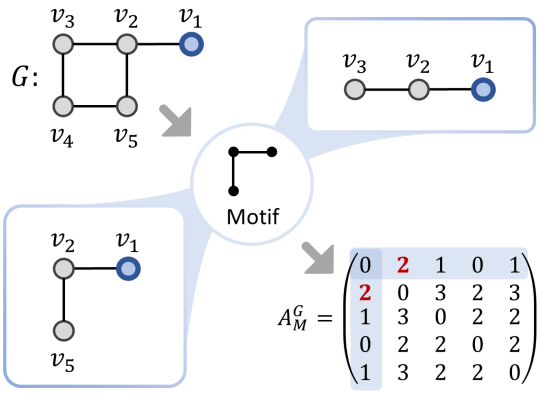



Figure 1: **Motif and motif-adjacency matrix.** In this case, Motif is a chain-like subgraph. For the node $v_1$ and $v_2$, there exist two subgraphs satisfy this pattern: $v_1$ - $v_2$ - $v_3$ and $v_1$- $v_2$ - $v_5$. Thus, we have $A_{1,2} = A_{2,1} = 2$. To avoid information imbalance issue, we instead define $\boldsymbol{A}$ as an indicator function, that is, $A_{1,2} = A_{2,1} = 1$.

(A) Since the connections in the original definition of the motif adjacency matrix are dense, blindly employing the motif-based propagation will lead to even faster convergence to the over-smoothed representation. Seeing this, we first propose the notion of the sparse motif to remove the spurious connections and then explore the properties of its message-passing. In this way, we can extract the local topology captured by the sparse motif without worsening the over-smoothing issue.

(B) On top of this, we propose a hierarchical motif aggregation mechanism to further mitigate the over-smoothing issue, where the features from motif-level message-passing and graph-level message-passing are given in a learnable manner.

(C) Theoretically, we show that the output embedding of our proposed method will not converge to a trivial subspace under mild assumptions. Empirically, we perform a series of experiments on 7 real-world datasets, demonstrating our proposed method's superiority.

## 2 Prior Art

In recent years, GCNs have attracted significant attention as an effective tool to handle graph data. Early methods [Kipf and Welling, 2017; Veličković *et al.*, 2018] only use shallow networks since deeper networks lead to performance degradation, which is known as over-smoothing [Huang *et al.*, 2020]. To solve this limitation, various methods are proposed [Xu *et al.*, 2018; Liu and others, 2020; Liu and others, 2020; Chen *et al.*, 2020]. See Appendix A for more details. In this work, we propose a motif-based GCN to overcome over-smoothing. Unlike previous work, our main idea is to improve the diversity of the graph spectrum by multiple message-passing paths of motifs.

## 3 Preliminaries

### 3.1 Learning from the Graph Structure

In this paper, we focus on the unweighted undirected graph $G = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{v_1, ..., v_n\}$ is the node set and $\mathcal{E}$ is the edge set defined on $\mathcal{V}$. Let $\boldsymbol{A} \in \{0,1\}^{n\times n}$ denote the adjacency matrix with $\boldsymbol{A}_{i,j} = \boldsymbol{A}_{j,i}$. Then, the degrees of the nodes are defined by $\boldsymbol{d} = \{d_1, ..., d_n\}$, where $d_i = \sum_{j\neq i} \boldsymbol{A}_{i,j}$. The degree matrix is further defined as the diagonal matrix $\boldsymbol{D}$ with $\boldsymbol{D}_{i,i} = d_i, i = 1, \cdots, n$. Meanwhile, let $\boldsymbol{X} \in \mathbb{R}^{n\times h}$ denote the feature matrix of the nodes, where $h$ is the dimension of feature.

Our task is to capture the informative patterns hidden in the nodes and the graph structure. As a representative method, GCNs propagate the node features by the adjacency matrix of the original graph, leading to the following formulation [Kipf and Welling, 2017]:

$$\boldsymbol{H}^{(l+1)} = \sigma(\hat{\boldsymbol{A}}\boldsymbol{H}^{(l)}\boldsymbol{W}^{(l)}), \tag{1}$$

where $\hat{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{I}$; $\sigma(\cdot)$ is the activation function; $\boldsymbol{H}^{(l)}$ and $\boldsymbol{W}^{(l)}$ are the embeddings and the weight matrix of the layer $l$, respectively. However, as presented in the introduction, focusing on the Laplacian of the original graph will lead to the notorious over-smoothing. To eliminate this issue, we propose to exploit the local topological structure of the graph, with the help of a classic tool named Motif.

### 3.2 Motif as a Local Topology Indicator

According to [Benson *et al.*, 2016], the motif is a kind of graph pattern, defined as any induced subgraph, which is small and non-isomorphic. One of the motifs with three nodes is shown in Figure 1. Compared with the global information in the original graph, motifs can capture local graph topologies more effectively. Consequently, we expect to introduce motifs into the GCN framework. To perform the calculations in GCN, we need to define the adjacency matrix for a given motif. To do this, [Windels and others, 2019] defines the adjacency matrix $\boldsymbol{A}'$ as:

$\boldsymbol{A}'_{i,j}$ = the number of times $v_i$ and $v_j$ occur in the same motif.

We observe that the elements in $\boldsymbol{A}'$ exhibit a highly skewed distribution, where only a small number of edges in $\boldsymbol{A}'$ have a large frequency while the others remain small. These edges will cover the information of other important edges. Therefore, in this paper, we define $\boldsymbol{A}'$ as an indicator function:

$$\boldsymbol{A}'_{i,j} = \begin{cases} 1, & v_i \text{ and } v_j \text{ co-occur in at least one motif,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

## 4 Methodology

In this section, we formally present the proposed SMEGCN. In Section 4.1, we propose the **Sparse-motif-based Message Passing** mechanism. On top of this, the **Hierarchical Motif Aggregation** is established in Section 4.2.

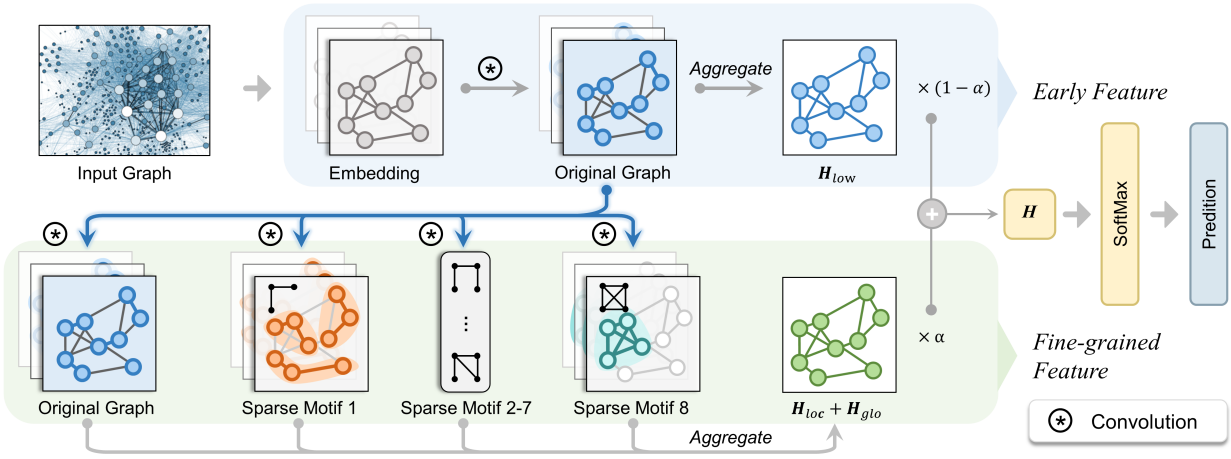



Figure 2: An overview of the SMEGCN. On top of the early features, fine-grained features are generated by our sparse-motif-based message passing, including local features from motifs and global features from the adajency matrix. These features are then integrate with our hierarchical motif aggregation method.

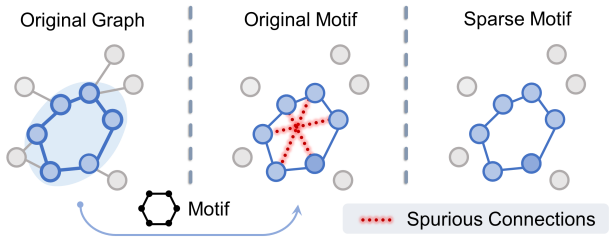



Figure 3: An example of spurious connections. In the original graph, we expect to capture the local topological information of these blue nodes and cut the links to gray nodes. The original motif adjacency matrix will introduce extra edges, while the sparse motif will not.

### 4.1 Sparse-motif-based Message Passing

**Sparse Motif.** To begin with, we present the message passing with a single motif. The adjacency matrix corresponding to the $k$-th motif is denoted as $\boldsymbol{A}_k$. Directly passing messages with $\boldsymbol{A}_k$ will suffer from **spurious connections**: since two disconnected nodes $v_i, v_j$ might belong to one motif, $\boldsymbol{A}_{k,i,j}$ could be positive even if $v_i$ and $v_j$ are not adjacent. Fig.3 illustrates a typical example. With Spurious connections, $\boldsymbol{A}_k^T$ will rapidly evolve into an all-one matrix when applying $T$ times message passing, which contradicts the target of capturing local information. Therefore, we propose a preprocessing method to transfer the motif adjacency matrix into a sparse motif matrix $\tilde{\boldsymbol{A}}_k$:

$$\tilde{\boldsymbol{A}}_k = \boldsymbol{A} \odot \boldsymbol{A}_k,$$

where $\odot$ refers to element-wise multiplication. In other words, we can explain each entry in $\tilde{\boldsymbol{A}}_k$ as: $\tilde{\boldsymbol{A}}_{k,i,j} = 1$ only if $v_i$ and $v_j$ are connected and co-occurs in at least one motif.

**Sparse-Motif-based Message Passing.** Given the output feature $\boldsymbol{H}_k^{(l)}$ of the $l$-th layer, the propagation process is to pass the message from the $l$-th layer to the $l$+1-th layer. Existing message-passing techniques like GCNII [Chen *et al.*, 2020] and APPNP [Klicpera *et al.*, 2018] could be applied here in a plug-and-play style. Here we apply the following formulation:

$$\boldsymbol{H}_k^{(l+1)} = \tilde{\boldsymbol{A}}_k \boldsymbol{H}_k^{(l)}, \tag{3}$$

where $\sigma$ is the activation function. In this way, the node features are passed to relevant nodes through the local topology of the $k$-th motif.

**Spectrum of the Sparse-Motif-based Message Passing.** By introducing the sparse motif, we can largely enrich the spectral information obtained from message passing. First, the bottom $m$ eigenvalues of $k$-th motif's graph Laplacian $\boldsymbol{L}_k = \boldsymbol{I} - D^{-1/2}\tilde{\boldsymbol{A}}_k D^{-1/2}$ becomes:

$$\min_{\boldsymbol{F}\in\mathbb{R}^{N\times m}, \boldsymbol{F}^\top \boldsymbol{F} = \boldsymbol{I}_m} tr\left(\boldsymbol{F}^\top \Delta_k \boldsymbol{F}\right)$$

Here $tr\left(\boldsymbol{F}^\top \Delta_k \boldsymbol{F}\right)$ is a dirichlet energy term:

$$\sum_{i=1}^{m} \sum_{v_i \overset{k}{\sim} v_j} (F_{i,m} - F_{j,m})^2$$

where $v_i \overset{k}{\sim} v_j$ if and only if $\tilde{A}_{k,i,j} = 1$. Another crucial spectral information is $\mathcal{M}$ (see Definition 1): the null space of $\boldsymbol{L}_k$, which is the linear space spanned by the eigenvectors of $\boldsymbol{L}_k$ associated with the eigenvalue 0 [Elsner and Tsonis, 1996]. According to the classical results in spectral graph theory, if the graph induced by $\tilde{A}_k$ has connected components $\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_M$, then the basis of $\mathcal{M}$ are:

$$I_{\mathcal{C}_1}, I_{\mathcal{C}_2}, \cdots I_{\mathcal{C}_M},$$

where $M$ is the number of eigenvalues; $I_{\mathcal{C}_i} \in \mathbb{R}^{N\times 1}$ and $I_{\mathcal{C}_{ij}} = 1$ if node $j$ belongs to $\mathcal{C}_i$, otherwise $I_{\mathcal{C}_{ij}} = 0$. Above all, we see that both eigenvalues and the null space depend on $\overset{k}{\sim}$, which is a function of the local topology conveyed in the $k$-th sparse motif. Hence to improve the diversity of the graph spectrum, we propose to integrate spectral information from multiple sparse motifs, which is shown in the next subsection.

### 4.2 Hierarchical Motif Aggregation

**Outline.** An overview of our method is available in Figure 2. Given the feature matrix $\boldsymbol{X}$ of a graph, we pass it through two linear layers followed by ReLU activation units and obtain a feature matrix $\boldsymbol{H}^{(0)}$. Afterward, we perform propagation with the adjacency matrix for a few steps to obtain early features:

$$\boldsymbol{H}^{(l+1)} = \boldsymbol{A}\boldsymbol{H}^{(l)}, \tag{4}$$

where $1 \leq l < L_0$ and $L_0$ is a hyperparameter. Then, we aggregate motif-based information by sparse motif-based message passing. Finally, local motif features and global features are integrated by our proposed hierarchical Motif Aggregation. Fig.2 presents a brief summary of the proposed network.

As we analyzed above, integrating information of multiple motifs is significant to the diversity of the graph spectrum. Therefore, we propose a hierarchical motif aggregation as follows.

After extracting the initial feature $\boldsymbol{H}^{(L_0)}$ we extract multiscale and multi-topological features based on $\boldsymbol{H}^{(L_0)}$. Specifically, for the $k$-th motif, we initialize $\boldsymbol{H}_k^{(0)}$ with $\boldsymbol{H}^{(L_0)}$, and perform motif-based message passing as in Eq. 3 for $L_1$ times. In this way, we obtain two-level features: for a specific motif $k$, $\{\boldsymbol{H}_k^{(l)}\}_{l=1}^{L_1}$ contains multiscale information; for a specific layer $l$, $\{\boldsymbol{H}_k^{(l)}\}_{k=1}^{K}$ contain multi-topological information. Therefore, we could aggregate the motif-based features with rich local information in an adaptive manner:

$$\boldsymbol{H}_{loc} = \sum_{l=1}^{L_1}\sum_{k=1}^{K} w_k^{(l)} \boldsymbol{H}_k^{(l)}, \tag{5}$$

where $w_k^{(l)} > 0$ are learnable parameters.

Notice that $\tilde{\boldsymbol{A}}_k$ only contains partial edges of the original graph, thus it might fail to capture global information. Therefore, only using motif-based features for prediction is far from enough. To this end, we propose to further aggregate global information with the adjacency matrix $\boldsymbol{A}$. Specifically, we perform similar aggregation from $\boldsymbol{H}^{(L_0)}$ with the adjacency matrix $\boldsymbol{A}$, and obtain $\{\boldsymbol{H}_{glo}^{(l)}\}_{l=1}^{L_1}$. These features are also aggregated with learnable parameters $\{w_{glo}^{(l)}\}_{l=1}^{L_1} > 0$:

$$\boldsymbol{H}_{glo} = \sum_{l=1}^{L_1} w_{glo}^{(l)} \boldsymbol{H}_{glo}^{(l)}. \tag{6}$$

$w_k^{(l)}$ and $w_{glo}^{(l)}$ are normalized with the Softmax operation, such that $\sum_{l=1}^{L_1}\sum_{k=1}^{K} w_k^{(l)} + \sum_{l=1}^{L_1} w_{glo}^{(l)} = 1$.

Moreover, to further take low level information into consideration, the early features $\{H^{(l)}\}_{l=1}^{L_0}$ mentioned in Eq. 4 are aggregated in the following manner:

$$\boldsymbol{H}_{low} = \sum_{l=1}^{L_0} w^{(l)} \boldsymbol{H}^{(l)}, \tag{7}$$

where $\sum_{l=1}^{L_0} w^{(l)} = 1$.

Finally, the merged feature is a linear combination of these features:

$$\boldsymbol{H} = \alpha\left(\boldsymbol{H}_{loc} + \boldsymbol{H}_{glo}\right) + (1-\alpha)\boldsymbol{H}_{low}, \tag{8}$$

where $\alpha \in (0, 1]$ is a tunable hyperparameter.

## 5 Theoretical Analysis

In our proposed method, we claim that the over-smoothing issue could be mitigated by integrating multiple motif-based message-passing branches. In this section, we will present a theoretical analysis of the claim. According to [Huang *et al.*, 2020], the over-smoothing issue could be mathematically expressed as the distance from the final output to the null space of the graph Laplacian matrix tends to 0 as the depth increases. Hence, we will prove that such distance would not vanish with the SMEGCN method.

For simplicity, we only consider motif features from one layer and omit the superscripts if there is no ambiguity. With the representation of the output of the motif path $i$ and the corresponding weight $w_i$, the aggregated result $\boldsymbol{H}_{loc}$ is obtained as follows:

$$\boldsymbol{H}_{loc} = \sum_{i=1}^{K} w_i \boldsymbol{H}_i \tag{9}$$

where $K$ is the number of motifs. Without loss of generality, we assume $\sum_{i=1}^{K} w_i = 1$.

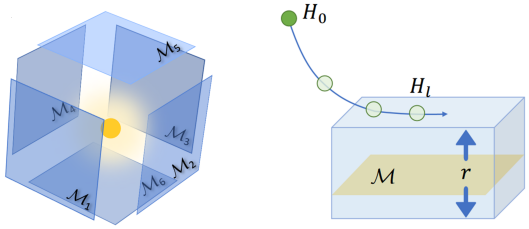



Figure 4: Result of convergence: (a) The node embeddings in **SMEGCN** will not converge to any trivial subspace when $div$ is large; (b) The node embeddings in **General GCN** will converge to a single subspace with a radius of $r$ and cause over-smoothing.

In such a manner, $\boldsymbol{H}_{loc}$ will integrate the spectral information from different motifs. To survive over-smoothing, we have to additionally require the spectrum of the motifs to exhibit a certain degree of diversity. First, we have to clarify the basic definitions of the null space of the graph Laplacian, which we call trivial subspace in this paper since it is only related to the connected components and nodes degrees [Huang *et al.*, 2020]:

**Definition 1** (Trivial Subspace)**.** *Given the normalized adjacency matrix $\hat{\boldsymbol{A}}$, and the number of classes $C$, we define $\mathcal{M} := \{\boldsymbol{H} \in \mathbb{R}^{N\times C} \big| \boldsymbol{H} = \hat{\boldsymbol{E}}\boldsymbol{C}, \boldsymbol{C} \in \mathbb{R}^{M\times C}\}$ as an $M$-dimensinal subspace in $\mathbb{R}^{N\times C}$. $\hat{\boldsymbol{E}} = \{\hat{\boldsymbol{e}}_1, \ldots, \hat{\boldsymbol{e}}_M\} \in \mathbb{R}^{N\times M}$ denotes the collection of bases of the largest eigenvalue of $\hat{\boldsymbol{A}}$. Then the subspace $\mathcal{M}$ is called the **trivial subspace** associated with $\hat{\boldsymbol{A}}$. Moreover, the distance from matrix $\boldsymbol{H} \in \mathbb{R}^{N\times C}$ to $\mathcal{M}$ is denoted as $d_{\mathcal{M}}(\boldsymbol{H}) := \inf_{\boldsymbol{Y}\in\mathcal{M}} \|\boldsymbol{H} - \boldsymbol{Y}\|_F$ with the $\|\cdot\|_F$ being the Frobenius norm. Denote $\mathcal{M}^k$ as the subspace of the $k$-th motif.*

Then we can define the diversity based on the distance.

**Definition 2** (Diversity)**.** *For outputs $\boldsymbol{H}_i \in \mathbb{R}^{N\times C}(i \in [K])$ from different motif paths, we define diversity as follows:*

$$div = 1/\max_{i,j,k} \left|\cos\Big(\mathsf{Proj}_k(vec(\boldsymbol{H}_i)), \mathsf{Proj}_k(vec(\boldsymbol{H}_j))\Big)\right|,$$

| Datasets | Classes | Nodes | Edges | Features | Density | Hom. Ratio | d-max |
|---|---|---|---|---|---|---|---|
| CoauthorCS | 15 | 18,333 | 81,894 | 6,805 | 0.0005 | 0.80 | 24 |
| CoauthorPhysics | 5 | 34,493 | 247,962 | 8,415 | 0.0004 | 0.93 | 17 |
| AmazonComputers | 10 | 13,381 | 245,778 | 767 | 0.0027 | 0.78 | 10 |
| AmazonPhoto | 8 | 7,487 | 119,043 | 745 | 0.0042 | 0.83 | 11 |
| TEXAS | 5 | 183 | 309 | 1,703 | 0.0185 | 0.11 | 8 |
| WISCONSIN | 5 | 251 | 499 | 1,703 | 0.0158 | 0.21 | 8 |
| CORNELL | 5 | 183 | 295 | 1,703 | 0.0176 | 0.30 | 8 |

Table 1: Statistics of datasets. The *density* is computed by $\frac{2m}{n^2}$ and the *Homophily Ratio* is computed following [Zhu and others, 2020]. The *d-max* denotes the largest diameter among all connected components in the network. Note that we only consider the largest connected component in co-purchase graphs as [Shchur *et al.*, 2018].

*where* $\mathsf{Proj_k}$ *denotes the projection on* $\mathcal{M}^k$ *and* $vec(\cdot)$ *denotes the vectorization of a matrix.*

Intuitively, $div$ is large when the outputs from two different motif paths are almost orthogonal. In the next proposition, we continue to show that the diversity will be large at a high probability under certain assumptions.

**Proposition 1.** *Assuming that for any* $i \in [K]$, $\boldsymbol{H}_i$ *is uniformly randomly sampled with a* $d$*-dimensional uniform distribution, if* $d$ *is sufficiently large, we have:*

$$\mathbb{P}\left[div \geq \left[1 - \left(\frac{\delta}{\pi M g(d)}\right)^{\frac{2}{d-2}}\right]^{-\frac{1}{2}}\right] \geq 1 - \delta \qquad (10)$$

*where* $M = k(k-1)$.

Following the above proposition, we see that $div$ will be large with a high probability when $N \times C$ is large.

**Assumption 1.** *We assume that* $div$ *has a large lower bound with a small* $\varepsilon$*:*

$$div \geq \frac{1}{\varepsilon} \qquad (11)$$

Now, we are ready to derive the distance from $\tilde{\boldsymbol{H}}$ to subspace $\mathcal{M}^k$.

**Proposition 2.** *For* $\forall k \in [\boldsymbol{K}]$*, for the case of simplicity, we assume that* $||\mathsf{Proj_k}(vec(\boldsymbol{H}_i))|| = 1,\ \forall i, k$*, then:*

$$d^2_{\mathcal{M}^k}(\tilde{\boldsymbol{H}}) \geq \frac{1}{K} - \varepsilon \sum_{i,j=1, i\neq j}^{K} \alpha_i \alpha_j \qquad (12)$$

From Proposition 2, we can draw the final conclusion: *By ensembling the motif spectrum, the final out will not converge to any trivial subspace and thus the over-smoothing problem is largely mitigated.* See Appendix B for more detailed proof.

# 6 Experiments

In this section, we conduct comprehensive experiments on semi-supervised node classification to evaluate the effectiveness of SMEGCN.

## 6.1 Datasets

To validate the effectiveness of our proposed method across different application scenarios, we conduct experiments on seven datasets with various sizes, densities, and homophily. The statistic of datasets is provided in Table 1. For a fair comparison, we follow the official data split. These datasets could be categorized into three types:

- **Coauthorship graphs** includes *Coauthor CS* and *Coauthor Physics* [Shchur *et al.*, 2018]. In each graph, the node set consists of authors, while the edges are determined by whether two authors have co-authored. The features are the keywords of the authors' publications.
- **Co-purchase graphs** includes *Amazon Computers* and *Amazon Photo* [Shchur *et al.*, 2018]. The node set consists of different goods, and two goods are connected by an edge if they are frequently bought together.
- **Web networks graphs** [Pei *et al.*, 2020] includes *Cornell*, *Texas* and *Wisconsin*. The nodes in these graphs are webpages, and the edge set consists of hyperlinks connecting these webpages. The feature of each node is a bag-of-words representation of the webpage.

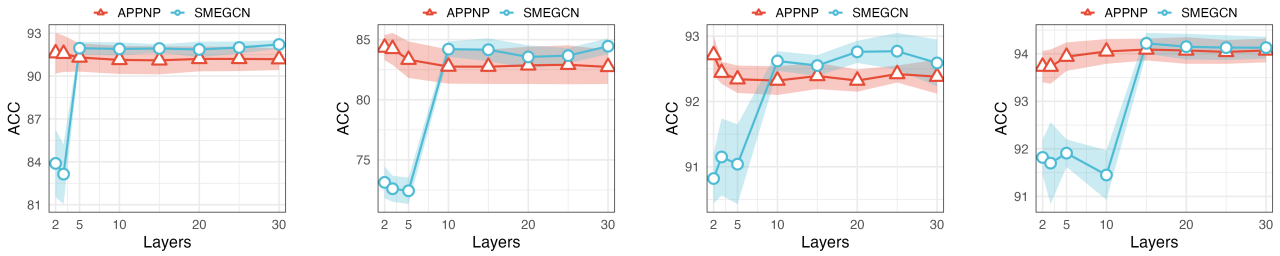
## 6.2 Competitors

To show the advantages of the proposed SMEGCN against the state-of-the-art methods, we compare our method with three categories of baselines:

- **Naive methods** including GCN [Kipf and Welling, 2017], the first work to use GCN for node classification, and GAT [Veličković *et al.*, 2018], an attention-based spatial method.
- **Initial residual methods** including APPNP [Klicpera *et al.*, 2018] and GCNII [Chen *et al.*, 2020]. The main idea of these methods is to keep local information by adding the input to deeper features.
- **Skip connection methdos** including JKNet [Xu *et al.*, 2018], GPRGNN [Chien *et al.*, 2020] and DAGNN [Liu and others, 2020]. The key design of these methods is adding skip connection to aggregate multiscale information.

## 6.3 Implementation Details

**Calculation of Motif.** According to our preliminary experiments, neighbors within four hops are more useful. Therefore, eight kinds of motifs with no more than four nodes are chosen. See Appendix C for more details. The algorithm in [Ahmed *et al.*, 2015] is used for counting node motifs.




(a) AmazonPhoto (b) AmazonComputers (c) CoauthorCS (d) CoauthorPhysics

Figure 5: Results of SMEGCN and APPNP with different depths.

| | CoauthorCS | CoauthorPhy. | AmazonCom. | AmazonPho. | TEXAS | WISC. | CORNELL |
|---|---|---|---|---|---|---|---|
| GCN | 91.51±0.56 | 92.64±0.49 | 82.13±1.11 | 91.20±0.53 | 64.86±1.71 | 56.47±0.78 | 48.65±0.00 |
| GAT | 88.78±0.86 | 91.14±1.38 | 82.32±1.01 | 90.12±0.85 | 62.16±2.96 | 55.69±1.57 | 50.27±2.16 |
| SGC | 91.55±0.58 | OOM | 65.48±6.07 | 91.49±0.23 | 64.86±0.00 | 56.86±0.00 | 54.05±0.00 |
| JKNet | 90.63±0.67 | 91.80±0.56 | 69.47±7.75 | 90.07±0.65 | 70.81±2.02 | 54.90±4.11 | 56.22±2.02 |
| APPNP | 92.32±0.21 | 94.06±0.23 | 83.57±1.44 | 91.25±0.82 | 70.27±0.00 | 57.65±1.57 | 69.73±1.08 |
| GCNII | 91.05±0.56 | 93.66±0.29 | 83.21±1.07 | 90.89±0.58 | 64.86±1.71 | 74.51±1.24 | 65.95±1.32 |
| DAGNN | 92.81±0.42 | 94.12±0.43 | 84.10±0.89 | 91.38±1.04 | 65.41±1.08 | 67.84±2.00 | 60.00±1.08 |
| GPRGNN | 91.77±0.35 | 93.97±0.32 | 82.76±1.58 | 92.12±0.54 | 62.16±1.71 | 66.27±1.47 | 58.92±2.02 |
| **OURS** | **93.03±0.00** | **94.21±0.43** | **84.45±0.66** | **92.21±0.26** | **71.89±1.32** | **80.39±0.78** | **72.43±1.08** |

Table 2: Performance Comparison Results, where OOM represents that the results are unavailable due to the out-of-memory exception.

**Training strategy.** Our model and the competitors are trained by minimizing the cross-entropy loss with the Adam optimizer. Concretely, the initial learning rate is set to 0.01, and the weight decay factors for the first two linear layers are respectively taken from $\{0.00001, 0.00005, 0.005\}$ and $\{0, 0.0005\}$. Other hyperparameters are tuned according to the performances on the validation sets. For our model, we add dropout layers with the probability searched from $[0, 0.9]$ with an interval of $0.1$. Hyperparameters of competitors are also tuned similarly. The models are trained for at most 1000 epochs. The checkpoint with the best performance on the validation set is preserved to evaluate the corresponding performance on the test set.

## 6.4 Quantitative Results

The main results are shown in Table 2. Each experiment is repeated 5 times, and the mean classification accuracy, and the standard deviation are reported as well. From Table 2, we could make the following observations: **a)** Our proposed method achieves state-of-the-art performance in all seven datasets, which validates the effectiveness of our method. **b)** In the webpage datasets TEXAS, WISCONSIN, CORNELL, the improvement is more significant. The reasons are two-fold: first, webpage datasets have higher heterophily, thus finer local information from motifs is required to ensure discriminability; second, these datasets are denser and more likely to suffer from over-smoothing, while the proposed method could overcome the over-smoothing. **c)** Although our improvements against the best baseline are not significant in all datasets, our results are consistent in these datasets, which shows that the aggregation with multiple motifs can adapt to graphs with various topologies.

## 6.5 Effect of the Model Depths

To validate the effect of our method against the over-smoothing issue, we continue to show the empirical result when the model depths keep increasing from 2 to 30. We find that the top-2 model, *i.e.* SMEGCN and APPNP significantly outperform the other competitors. Hence, we here only visualize the top-2 results in Fig.5. Taking Fig.5-(c) for example, the results show that our model performs poorly when the model is shallow. However, when the depth exceeds 10, we can observe a sharp performance improvement of our method so that SMEGCN outperforms APPNP when the depth is 10. Meanwhile, the performance of APPNP is rather stable in terms of depth and starts to reduce when the depth exceeds 10. Above all, we can clearly see the advantage of SMEGCN in terms of the over-smoothing issue.

# 7 Conclusion

In this paper, we consider the over-smoothing problem by exploring the diversity of the graph spectrum. Specifically, we introduce the local topology into the graph Laplacian by introducing the notion of the motif. We find that the motif-level graph Laplacian is much denser than the original graph, which will lead to a trivial all-one matrix after even a few iterations of message-passing. Consequently, we propose the sparse motif to remove the irrelevant connections. Furthermore, we propose a hierarchical motif aggregation mechanism to integrate early global features, motif-level local features, and graph-level global features. Theoretically, we show that our proposed method will not lead to an over-smoothed representation if the diversity across motifs is large. Experimental results on 7 real-world datasets demonstrate the superiority of our method.

## Ethical Statement

There are no ethical issues.

## Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0102000, in part by National Natural Science Foundation of China: U21B2038, 61931008, 61836002, 6212200758 and 61976202, in part by the Fundamental Research Funds for the Central Universities, in part by Youth Innovation Promotion Association CAS, in part by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDB28000000, in part by the National Postdoctoral Program for Innovative Talents under Grant BX2021298, and in part by mindspore, which is a new AI computing framework [2].

## References

[Ahmed *et al.*, 2015] Nesreen K Ahmed, Jennifer Neville, Ryan A Rossi, and Nick Duffield. Efficient graphlet counting for large networks. In *ICDM*, 2015.

[Benson *et al.*, 2016] Austin R Benson, David F Gleich, and Jure Leskovec. Higher-order organization of complex networks. *Science*, 2016.

[Chen *et al.*, 2020] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *ICML*, 2020.

[Chien *et al.*, 2020] Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. In *ICLR*, 2020.

[Elsner and Tsonis, 1996] James B Elsner and Anastasios A Tsonis. *Singular spectrum analysis: a new tool in time series analysis*. Springer Science & Business Media, 1996.

[Gao *et al.*, 2020] Difei Gao, Ke Li, Ruiping Wang, Shiguang Shan, and Xilin Chen. Multi-modal graph neural network for joint reasoning on vision and scene text. In *CVPR*, 2020.

[Guo and Wang, 2020] Zhiwei Guo and Heng Wang. A deep graph neural network-based mechanism for social recommendations. *TII*, 2020.

[Huang *et al.*, 2020] Wenbing Huang, Yu Rong, Tingyang Xu, Fuchun Sun, and Junzhou Huang. Tackling over-smoothing for general graph convolutional networks. *arXiv preprint arXiv:2008.09864*, 2020.

[Jiang and Luo, 2021] Weiwei Jiang and Jiayun Luo. Graph neural network for traffic forecasting: A survey. *arXiv preprint arXiv:2101.11174*, 2021.

[Kipf and Welling, 2017] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[Klicpera *et al.*, 2018] Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. In *ICLR*, 2018.

[Liu and others, 2020] Meng Liu et al. Towards deeper graph neural networks. In *SIGKDD*, 2020.

[Liu *et al.*, 2021] Yujia Liu, Kang Zeng, Haiyang Wang, Xin Song, and Bin Zhou. Content matters: a gnn-based model combined with text semantics for social network cascade prediction. In *KDD*, 2021.

[Narasimhan *et al.*, 2018] Medhini Narasimhan, Svetlana Lazebnik, and Alexander G Schwing. Out of the box: Reasoning with graph convolution nets for factual visual question answering. In *NeurIPS*, 2018.

[Pei *et al.*, 2020] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. In *ICLR*, 2020.

[Rahaman and Gagliardi, 2020] Obaidur Rahaman and Alessio Gagliardi. Deep learning total energies and orbital energies of large organic molecules using hybridization of molecular fingerprints. *JCIM*, 2020.

[Shchur *et al.*, 2018] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

[Shen and others, 2021] Zi-Ang Shen et al. Npi-gnn: Predicting ncrna–protein interactions with deep graph neural networks. *BIB*, 2021.

[Shui and Karypis, 2020] Zeren Shui and George Karypis. Heterogeneous molecular graph neural networks for predicting molecule properties. In *ICDM*, 2020.

[Tsubaki *et al.*, 2019] Masashi Tsubaki, Kentaro Tomii, and Jun Sese. Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics*, 2019.

[Veličković *et al.*, 2018] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018.

[Windels and others, 2019] Sam FL Windels et al. Graphlet laplacians for topology-function and topology-disease relationships. *Bioinformatics*, 2019.

[Wu *et al.*, 2019] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *ICML*, 2019.

[Xie *et al.*, 2020] Yi Xie, Yun Xiong, and Yangyong Zhu. Sast-gnn: A self-attention based spatio-temporal graph neural network for traffic prediction. In *DASFAA*, 2020.

[Xu *et al.*, 2018] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *ICML*, 2018.

[Zhu and others, 2020] Jiong Zhu et al. Beyond homophily in graph neural networks: Current limitations and effective designs. In *NeurIPS*, 2020.

[2] https://www.mindspore.cn/