EasyChair Preprint

№ 8046

Frequency Inception Based Graph Convolutional Neural Networks

Feifei Wei and Kuizhi Mei

May 22, 2022

Frequency Inception Based Graph Convolutional Neural Networks

Anonymous submission

ABSTRACT

Graph convolutional neural networks (GCNs) have demonstrated powerful representing ability of irregular data, e.g., skeletal data and graph-structured data, providing the effective mechanism to fuse the neighbor nodes. The representative are the spectral-based methods, which are designed to obtain the beneficial discriminative information from input signals for learning. However, many works have been shown that the essence of the GCN are low-pass filters, which propagate information and distill the beneficial signals, thus performing the information denoising. Although there are some efforts not only concentrate on the fixed low-pass filters, but also the adaptive frequency filters, which harness the dynamic frequency, they do not go deeply into the intrinsic part of the useful signals of all nodes. To explore the core of signals propagating, we design a novel framework FiGCN that leverages the each of channel signal, which comprises of all the neighbor. Specifically, every channel of a node and its neighborhoods contribute dynamically to the final channel signal, which can capture the inherent difference of different channel and neighbor nodes and even determine whether a node is neighborhood or not. Meanwhile, it can enhance the representation ability of nodes and ameliorate the over-smoothing problem. On the other hand, our model can dynamically adjust the importance of neighborhoods to the central vertex. We empirically validate the effectiveness of the proposed framework FiGCN on various benchmark datasets. Experimental results show that our method achieves substantial improvements and outperforms the state-of-the-art performance significantly.

1 INTRODUCTION

Graph convolutional neural networks (GCNs) have received broad attention for its excellent performance in various of graph tasks, e.g., node classification [1, 32, 35], link prediction [31, 39], and recommendation [4, 24, 29]. In essence, the GCNs send and receive the nodes feature representation to and from its neighbor nodes and stack multiple layers to learn the global node features [10]. Inspired by the convolutional neural networks, the GCNs are developed to the domain of irregular data. Generally, the GCNs are categorized as spectral-based and spatial-based methods. The spectral-based methods are found on the theory of Graph Signal Processing (GSP) [27] and Spectral Graph Theory [7], which perform convolution operation in the spectral domain. As for the spatial-based methods, they are exhibited in straightforward and understandable forms, which perform convolutions on spatial domain and the information is conveyed along the edge [2, 12]. Until the work [19] arises, which leverages the localized first-order approximation of spectral graph convolution, the two categories methods are connected. From another perspective, the GCN is also spatial-based, which possesses of the obvious sense of the node localization, thus giving rise to the prosperity of this methods [2, 11, 12, 17, 20, 23, 28, 35]. Although the spatial-based methods have achieved remarkable success in various applications, they are not investigated from perspective of the essence of the signal-passing, just stacking multiple layers to learn the high-level node feature representations. Recently, some researches find that the essence of the signal-passing is a fixed low-pass filter [25], which means that the beneficial signals of the node feature representation are low frequency with the high frequency degraded [33]. Whereas, the real-world is sophisticated and the data exhibits hierarchies. To tackle this problem, there are researchers that dive into deeply to explore the intrinsic useful information of the node feature representation [3, 6, 9], which shows that the high frequency may also contain the beneficial information for contributing the performance of the tasks. Especially, the work [9] harnesses the frequency filter adaptively to capture the beneficial information of every layer, which is based on the spectral method and achieves remarkable results. However, the work [9] neglects the neighbor nodes frequency dependency of the same channel, which contributes greatly to the final performance.

To tackle the aforementioned problems, we dive into the essence of signal-passing, which alleviates the over-smooth and computation efficiency of spectral-based methods. Specifically, we propose a framework FiGCN that explores the neighbor nodes frequency dependency of the same channel and meanwhile adapts the importance of different channel. For each channel, it consists of a node and its neighbor nodes, which dynamically adjust the frequency components together, resulting in learning the discriminative and informative frequency information to the channel's final aggregated output feature representation. To the best of our knowledge, this is the first attempt that takes the neighbor nodes frequency components of one channel into consideration and dynamically adjust its weight to fit the node feature representation, which contributes to boosting the network performance aggressively. Our contributions are summarized as follows:

- We propose a novel framework named as FiGCN that investigates the neighbor nodes frequency dependency of the same channel and meanwhile adapts the importance of different channel. The idea may pave the way for revealing the intrinsic signal-passing mechanism of graphs.
- Compared to the conventional GCNs, our framework effectively tackle the unequal importance of information from different vertices.
- Our framework FiGCN implementation is simple and efficient, thus giving rise to high hardware computation efficiency. That makes it highly feasible to the real applications.
- We conduct extensive experiments on multiple datasets and the results show that our method achieves promising results and outperform state-of-the-art methods significantly.

2 RELATED WORK

GCNs are derived from classical convolutional neural networks, which are naturally suitable for the graph data, e.g., social networks

and chemical molecules. By transforming graph signals into the spectral domain via the graph Fourier transform, the spectral-based GCNs are built [5, 8, 16, 19]. Compared to the spectral-based GCNs, the spatial-based GCNs are intuitively straightforward, which aggregates the message from its neighbor nodes, resembling to the convolution kernel of CNN. The seminal work [19] propose a localized first-order approximation of spectral graph convolution [8], which bridges the spectral-based and spatial-based methods of GCNs, leading to the boom of spatial-based GCNs. However, the spatial-based GCNs may not investigate the intrinsic mechanism of signal-passing of the graph. For this sake, spectral-based GCNs are thoroughly explored [5, 8, 9, 16, 19, 25]. To explore the essence of the propagation of GCN, the work [25] discovers that the frequency response of the convolution operation is equal to the fixed low-pass filter at each layer. However, owing the complex real-world graph data, it may be not enough to demonstrate that all the useful signals are low-passed. Recent researches show that while high-frequency components may also have useful information that contributes to the performance of some tasks [3, 6, 9]. In this paper, based on the work [9], , we propose a framework FiGCN to dive deeply into the varying frequency components of a node and its neighborhoods, which play the key role in graph tasks. Concurrently with this work, the framework of FiGCN have applied to the knowledge graph of our study and achieved astonishing results, which indicates that the dynamic frequency may prone to adapting to various scenarios and obtaining beneficial information of the latent hierarchy.

3 NOTATIONS AND PRELIMINARIES

3.1 Notations

Let $\mathcal{G} = (V, E)$ be a graph, where $V = \{v_1, v_2 \cdots v_N\}$ and E denote a set of nodes and edges, respectively. Each node $v \in V$ has feature representation of $x_v \in \mathbb{R}^F$. Let $X = \{x_1, x_2 \cdots x_N\} \in \mathbb{R}^{N \times F}$. Here, N and F denote the total number of nodes and feature representation dimension of each node, respectively. Let $A \in \mathbb{R}^{N \times N}$ be adjacent matrix of the graph, where $A_{i,j} = 1$ if $v_j \in \mathcal{N}(v_i)$, otherwise $A_{i,j} = 0$. $\mathcal{N}(\cdot)$ denotes the neighbor nodes. Let $L = D - A$ be Laplacian matrix of the graph, where $D = \text{diag}(d_1, d_2 \cdots d_N)$ is the diagonal degree matrix of A. The symmetric normalized Laplacian matrix is defined as $L_{sys} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$. Spectral-based GNN methods is derived from graph filter based on Graph Signal Processing [27]. According to the theory, L_{sys} can also be rewritten as $L_{sys} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = U \Lambda U^T$. Here, $U = [u_1, u_2 \cdots u_N]$, where u_i is the i-th eigenvector of L_{sys} and $\Lambda = \text{diag}(\lambda_1, \lambda_2 \cdots \lambda_N)$ is the corresponding eigenvalue matrix.

3.2 Frequency Response Filter

In image signal processing, the Laplacian kernel is usually harnessed to obtain the high-frequency edge information for sharpening and blurring [13, 15, 34, 37] of images. As such, the graph signal process (GSP) has similar operations. To obtain the high-frequency signals, graph Laplacian matrix L can be multiplied by the input signal x, i.e., $h = Lx$, which characterizes the sharp high-frequency along the edge. Recently, the pioneering works [25, 33, 36] find out that the essence of signal passing along the edge of graph is the low-pass filter that filters the high-frequency and keeps the low-frequency for making a node and its neighbor nodes similar. Subsequently, the work [9] develops the adaptive frequency response filtering that adaptively adjusts the frequency of the channels. Specifically, they characterize the low-frequency by $z = x - Lx$, which leverages the original signal x to subtract the high-frequency signal Lx. If symmetric normalized Laplacian matrix L_{sym} is used, z conforms with the GCN [19]:

$$\begin{aligned} z &= x - \tilde{L}_{sym} x \\ &= \left(I - \tilde{D}^{-\frac{1}{2}} (\tilde{D} - \tilde{A}) \tilde{D}^{-\frac{1}{2}}\right) x \\ &= \left(I - \tilde{D}^{-\frac{1}{2}} \tilde{D} \tilde{D}^{\frac{1}{2}} + \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}\right) x \\ &= \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X, \end{aligned} \tag{1}$$

where $\tilde{A} = A + I$ is self-loop augmented adjacency matrix and $\tilde{D}$ is degree matrix derived from $\tilde{A}$. After Eq. (1), the low-frequency signals are obtained, which corresponds to the fixed low-pass filter in the spectral domain. In practical, however, high-frequency signals may play the complementary roles for learning representation [3, 6]. Besides that, with the layers deeper, the high-frequency signals are usually weakened, which is called well-known over-smoothing problem [22]. To alleviate the phenomenon, the work [9] proposes an adaptive frequency response filter method, which not only can capture the low-frequency, but also adjust the useful high-frequency signals. Specifically, for the j-th channel of the nodes, a parameter ϕ is assigned to learn the single channel information: $z_j = x_j - \phi \tilde{L} x_j \ (1 \leqslant j \leqslant F)$. If applied to all channels of the all nodes, it can be formulated as:

$$E = X - \tilde{L} X \Phi, \tag{2}$$

where $\Phi = \text{diag}(\phi_1, \phi_2 \cdots \phi_F)$, ϕ_j represents the learnable parameter of the j-th channel and E denotes the output signals of the nodes. At this point, the channel-wise paradigm has been built for adaptively capturing the beneficial low-frequency and high-frequency signals. In fact, this theory is special case of our FiGCN, which will be presented in the next section.

4 PROPOSED FRAMEWORK

In this section, we detail the frequency inception and the framework of FiGCN. First, we present the overall architecture of FiGCN. Then, we make a thorough analysis of the spatial and spectral to show the superiority of our model.

4.1 Overall Architecture of FiGCN

From the Eq. (2), we can observe that the channel-wise paradigm has obvious drawbacks. Firstly, for the specific channel, each component shares the identical parameter, which results in degrading of diversity of each component as shown in Fig. 1(a). Then, the number of parameters is too small, leading to the model little-explored. On the contrary, our model, as shown in Fig. 1(b), dynamically dives deeply into each frequency component to investigate the diversity of information. Formally, the adjacent matrix A can be rewritten as sparse form in order to fuse the parameter of α , which is used to dynamically adjust the frequency components in the form of inception. From the perspective of the spatial, the inception parameters

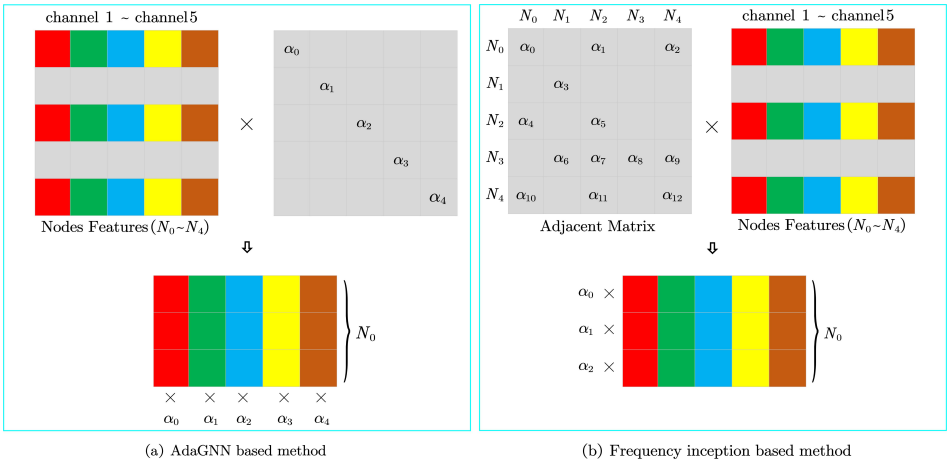

Figure 1: (a) AdaGNN based method [9], leveraging the diagonal matrix to channel-wise adaptively learning the beneficial information. (b) Frequency inception based method, dynamically diving deep into the each frequency component.

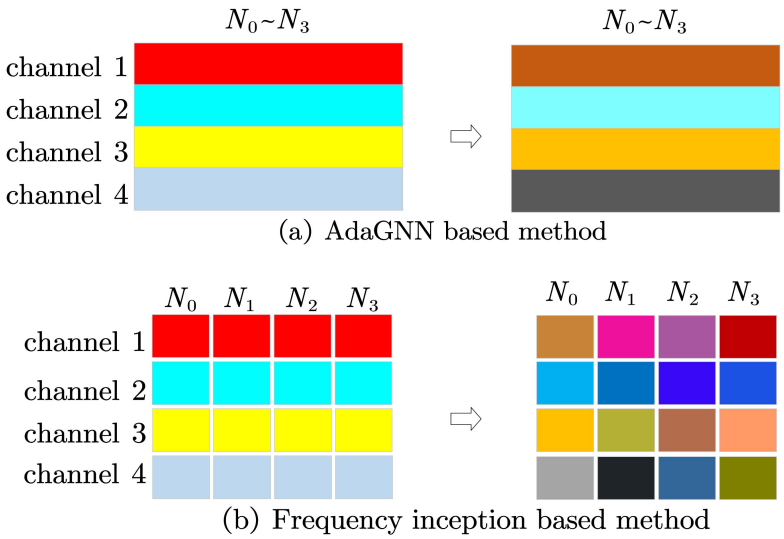

Figure 2: The effect of AdaGNN (a) and FiGCN (b). It is obvious that (b) achieves the diversity of each channel in terms of discrimination and information, compared to (a).

are equipped with the ability of dynamically deciding whether a node is its neighborhood or not.

Let n_i be the number of adjacent nodes of i-th node and n is the total number of adjacent nodes. The output filtered by the inception is:

$$E = X - \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}\alpha X, \tag{3}$$

where $\alpha = [\alpha_0, \alpha_1 \cdots \alpha_n]$ is the learnable inception parameter. With the end-to-end learning, α is able to learn and obtain the discriminative and informative signals, which is beneficial to the final accuracy. Specifically, taking the Fig. 1 as an example, if $\alpha_0, \alpha_1, \alpha_2$ are identical, the output of channel 1 is same with the Eq. (2). In this regard, our model can thoroughly explore the intrinsic frequency varying, which obtains the vital frequency components of each channel and ameliorates over-smoothing in deeper layers. Intuitively, the effect of our model can be depicted as in Fig. 2, which achieves the discriminative and informative of each channel. Though comparison of the two effect, the inception-based method is able to dive deeply into the channel components and investigate the most beneficial signals.

4.2 Spectral and Spatial Analysis of FiGCN

In this section, we analysis FiGCN from the perspectives of the spectral and spatial, which gives a thorough explanation of the advantages of FiGCN.

4.2.1 ***Spectral Analysis.*** The normalized adjacent matrix is defined as $\hat{\tilde{A}} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$. For convince, the learnable inception

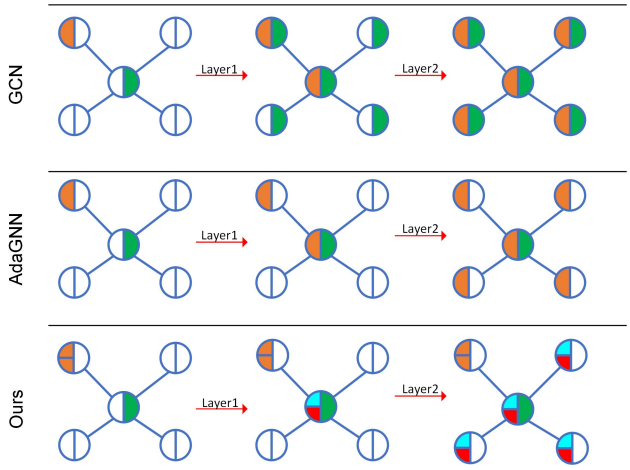

Figure 3: A comparison of three scheme. The upper is GCN that learn similar representation; the middle is AdaGNN [9] that becomes discriminative compared to the central node; the lower is our model, which introduces the inception parameters, learning the representation with much discriminative information.

parameters have been fused into $\tilde{A}$. The FiGCN consists of K layers, which are stacked to provide the inception frequency information of each layer.

Theorem 1. For the K layers of FiGCN, the frequency response function of the j-th input feature channel can be formulated as $f_K\left(\tilde{\lambda}_i, \alpha\right) = \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_i, \alpha\right) = \prod_{k=1}^{K}\left(1-\tilde{\lambda}_{i,\alpha}\right)$, where $\tilde{\lambda}_{i,\alpha}$ denotes the eigenvalue, which contains the learnable inception parameter α.

Proof. Considering the j-th feature channel, $x_j \in \mathbb{R}^N$ of the input signal at the k-th layer, we obtain:

$$\begin{aligned} x_j - \tilde{L}_{sym}x_j &= x_j - \tilde{U}\tilde{\Lambda}\tilde{U}^T x_j \\ &= \left(\tilde{U}\tilde{U}^T - \tilde{U}\tilde{\Lambda}\tilde{U}^T\right)x_j \\ &= \tilde{U}\left(I-\tilde{\Lambda}\right)\tilde{U}^T x_j, \end{aligned} \tag{4}$$

where $\Lambda = \mathrm{diag}\left(\lambda_1, \lambda_2 \cdots \lambda_N\right)$ and $\alpha = [\alpha_0, \alpha_1 \cdots \alpha_n]$ is contained in λ for dynamically adjusting the inception frequency. It can be derived that the frequency response function of FiGCN is $g_k\left(\tilde{\lambda}_i, \alpha\right) = 1 - \tilde{\lambda}^k_{i,\alpha^k}$. For the K layers of FiGCN, the frequency response filter function is:

$$f_K\left(\tilde{\lambda}_i, \alpha\right) = \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_i^k, \alpha^k\right) = \prod_{k=1}^{K}\left(1-\tilde{\lambda}^k_{i,\alpha^k}\right). \tag{5}$$

It can be noted that the frequency response function of K-layered GCN is $f_K\left(\tilde{\lambda}_i\right) = \left(1-\tilde{\lambda}_i\right)^K$ for any feature channel with obvious drawbacks. The FiGCN has four advantages: (1) the inception parameter α^k can dynamically adapt every frequency component. (2) for the K-layered FiGCN, α^k is leveraged to adjust different importance of different layers. (3) different nodes have its own inception parameters to learn the high-frequency and low-frequency components importance. (4) the inception parameters of different layers are decoupled, leading to learning discriminative and informative signals and alleviating the over-smooth problem.

4.2.2 ***Spatial Analysis.*** Intuitively, the FiGCN dynamically adjusts each feature channel in the form of inception and enables the framework to learn discriminative and informative signals. Compared with the AdaGNN [9], each node and its neighborhoods of FiGCN are equipped with independent inception parameters, leading to aggregating varying and beneficial information when multiple layers stacked. To better illustrate how it aggregates varying information, we provide a comparison of different frameworks. As shown in Fig. 3, for the middle subfigure [9], we suppose that the learnable parameters for these two channels across two layers are $\Phi_Y = \left\{\phi_{Y,1}, \phi_{Y,2}\right\}, \Phi_G = \left\{\phi_{G,1}, \phi_{G,2}\right\}$. If $\Phi_G = 0$, it can learn the information differently, compared with the upper subfigure in Fig. 3. However, for the FiGCN, the parameter is $\alpha = [\alpha_0, \alpha_1]$, which can adjust the two components and obtain much information when aggregating signals along edge. By adaptively learning the inception parameters, we can easily control the smoothness of each feature channel and its components, which naturally ameliorates the over-smoothing problem (the lower subfigures in Fig. 3).

Comparison of computational efficiency: Compared with the baseline methods, our proposed model achieves remarkable performance with even much fewer parameters, which makes it possible to apply the model to many practical applications with less computational power, such as IoT and edge devices. Conventional GCN [19] has complex and even probably redundant parameters when propagating with linear transformation and non-linearity activation repeatedly through K layers:

$$H = \sigma\left(\hat{\tilde{A}}\left(\cdots\sigma\left(\hat{\tilde{A}}XW^{(0)}\right)\cdots\right)W^{(K-1)}\right) \tag{6}$$

The number of parameters of K -layers model are $KF^{\ell}_{in}F^{\ell}_{out}$, where F^{ℓ}_{in} and F^{ℓ}_{out} are input and out node feature dimension at ℓ-th layer, respectively. However, our model has only Kn (the total number of adjacent nodes) parameters. Taking the dataset of Flickr as example, the total amount of parameters is more than 2,449,400 while our model is 484,292. Thus, it is usually true that $F^{\ell}_{in}F^{\ell}_{out} \gg n$, which dramatically ameliorates the computational pressure and is also hardware-friendly to deploy it on various hardware platforms.

4.3 Over-smoothing Analysis

We firstly analysis the inevitable over-smooth problem of GCN and then explain why our FiGCN can alleviate it.

For $\tilde{L}_{sym}x = \tilde{U}\hat{\Lambda}\tilde{U}^T x$, it can be regarded a projection of x onto N eigenvectors, re-weighting each component vector and accumulating them together. Through K layers, x can be expressed:

$$\begin{aligned} x = \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_1\right)\frac{u_1\cdot x}{|u_1|}u_1 + \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_2\right)\frac{u_2\cdot x}{|u_2|}u_2 + \\ \cdots + \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_N\right)\frac{u_N\cdot x}{|u_N|}u_N. \end{aligned} \tag{7}$$

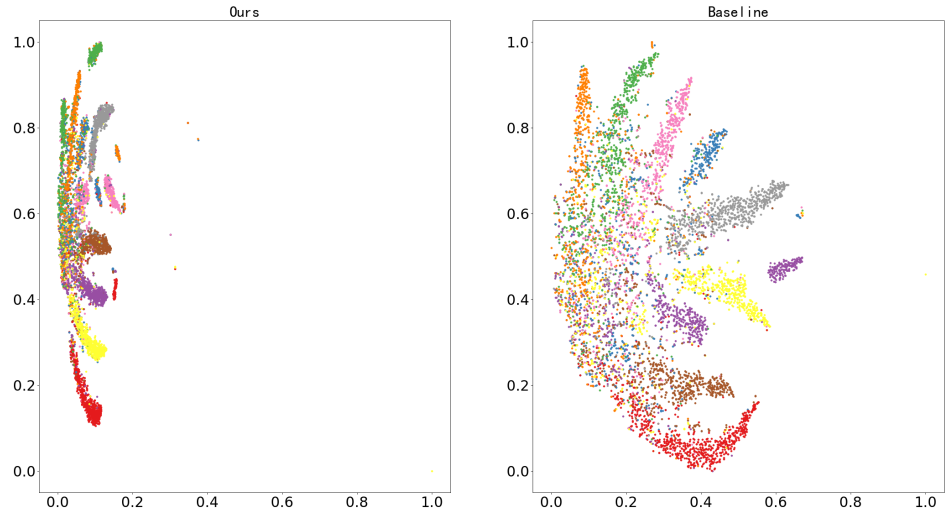

Figure 4: Spatial clustering visualization of Flickr in 2D from our model and baseline [9]. In the two figure, Different colors denote the different labels. From the perspective of the spatial, our model can learn much discriminative information, leading to the better spatial clustering than right.

It will be inevitable that x is collinear with u_i if and only if the Eq. (8) is satisfied.

$$\lim_{K\to\infty}\prod_{k=1}^{K} g_k\left(\tilde{\lambda}_i\right) = 0 \tag{8}$$

It means that feature values of different nodes are prone to be the same. However, most of the conventional GCN and its variants are mainly based on Eq. (6). To naturally alleviate the over-smooth problem, our model integrates the inception parameters into the frequency response function as expressed Eq. (5), which dynamically adjusts the frequency component in the form of inception. Thus, Eq. (6) can be rewritten as:

$$\begin{aligned} x = \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_1, \alpha\right)\frac{u_1\cdot x}{|u_1|}u_1 + \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_2, \alpha\right)\frac{u_2\cdot x}{|u_2|}u_2 + \\ \cdots + \prod_{k=1}^{K} g_k\left(\tilde{\lambda}_N, \alpha\right)\frac{u_N\cdot x}{|u_N|}u_N. \end{aligned} \tag{9}$$

From the Eq. (9), it can be noticed that every eigenvector of u_i is dynamically weighted, thus overcoming the over-smoothing problem.

5 EXPERIMENT AND RESULTS

5.1 Datasets

In this section, we will conduct our method on dataset of BlogCatalog [21], ACM [30] and Flickr [21].

BlogCatalog: BlogCatalog is social blog directory, in which bloggers and their blogs are managed. Each blogger has a tag with specific feature information. Users could thumb up to others, and then construct the link of social information. Ground truth for validation are based on the categories of bloggers' registration of their blogs.

ACM: ACM database is a collection of publications that are exhibited as nodes and any two publications are connected as edge when they share the same authors.

Flickr: Flickr is an image hosting and sharing website, in which user can specify a list of tags that they are interested in. Users also can interact with others like BlogCatalog. The photos are prespecified in different kinds, which are leveraged for the ground truth. The detailed information of these datasets is listed in Table 1.

5.2 Implementation details

The results of experiment will be compared with the state-of-the-art GNN: (1) GCN [19]; (2) GraphSAGE [14]; (3) SGC [33]; (4) DropEdge [26]; (5) PairNorm [38] including PairNorm-SI and PairNormSCS and (6) AdaGNN [9]. The dimensions of intermediate layers are set to 128, which is equal to the baseline model for fairness. We randomly sample 10% nodes for training, 20% for validation, and the rest 70% for test. The parameter of learning rate is set to 0.01, α 1e-6 and β 9e-4. Our model is conducted on NVIDIA GeForce RTX 2080 Ti and implemented by PyTorch. The optimizer is adaptive moment (Adam) algorithm [18]. The comparisons results are shown in Table 2.

5.3 Discussion

Though the Table 2, it can be noted that the performances of our model on datasets of BlogCatalog and Flickr are remarkable compared to other models. The most reasonable explanation may lie in the average node degree of two social networks BlogCatalog and

Table 1: Detailed information of datasets.

dataset	Nodes	Edges	Features	Average Degree	Classes
BlogCatalog	5196	173468	8189	66.8	6
ACM	16484	71980	8337	8.7	9
Flickr	7575	242146	12047	63.9	9

Table 2: Average accuracy with standard deviation on BlogCatalog, ACM and Flickr (the best results are marked with bold).

Datasets	BlogCatalog		ACM		Flickr	
Methods	8 Layer	16 Layer	8 Layer	16 Layer	8 Layer	16 Layer
GCN [19]	37.61±2.2%	20.61±1.9%	53.74±1.9%	35.97 ± 0.7%	12.20 ± 1.0%	13.29 ± 0.1%
GraphSAGE [14]	39.15±1.6%	18.34±3.9%	52.21 ± 1.5%	—	11.30 ± 0.1%	11.20 ± 0.2%
SGC [33]	47.94±0.9%	29.02±1.7%	73.80 ± 0.3%	64.01 ± 0.4%	16.64 ± 2.2%	14.40 ± 1.3%
DropEdge-GCN [26]	60.51±2.4%	51.88±.8%	71.43 ± 2.2%	67.37 ± 1.9%	36.16 ± 0.3%	27.30 ± 1.6%
Pairnorm-GCN-SI [38]	65.04±0.6%	67.51±0.4%	73.63 ± 0.2%	68.35 ± 2.0%	39.12 ± 0.8%	38.24 ± 0.2%
Pairnorm-GCN-SCS [38]	69.03±0.7%	69.75±1.2%	73.33 ± 0.1%	70.84 ± 1.4%	34.79 ± 0.3%	38.17 ± 0.2%
AdaGNN-S [9]	88.81 ± 0.1%	88.19 ± 0.2%	**75.64 ± 0.0%**	**74.95 ± 0.1%**	72.93 ± 0.1%	73.03 ± 0.4%
Ours	**91.2± 0.1%**	**91.5± 0.1%**	74.31 ± 0.1%	74.26 ± 0.25%	**79.10± 0.15%**	**79.50± 0.12%**

Flickr, which are much higher than ACM. Given the low average node degree of dataset of ACM, it achieves the comparable results compared to the baseline model. Thus, the inception channel components make a great difference especially in the datasets with high average node degree.

As shown in Fig. 4, the visualization of feature representations of our model and baseline model [9] in 2-D space is presented. Different classes of node label are indicated by different colors. Intuitively, through the distribution of the nodes, our model exhibits better spatial clustering than the baseline [9] from the perspective of compactness, which leads to high classification performance.

6 CONCLUSION AND FUTURE WORK

In this paper, we systematically dive into essence of signal passing along the edge of nodes and delicately design the framework of FiGCN in the form of inception. With the inception parameters, FiGCN is able to dynamically delve deeper into the frequency components of each channel and capture the varying low-frequency and high-frequency of different importance, leading to more beneficial signals obtained. From the perspective of spectral and spatial analysis, we investigate the superiorities over the conventional GNNs, which reflects on inception channel components and learnt discriminative information. As a result, we achieve significant advantages over the state-of-the-art GNNs, especially in the datasets with high average node degree. Due to the pioneering thinking, the inception framework can also inspire much sophisticated networks, which may integrate complementary information, e.g., knowledge graph, and contribute to investigate deeper useful information.

REFERENCES

[1] Sami Abu-El-Haija, Amol Kapoor, Bryan Perozzi, and Joonseok Lee. 2020. N-gcn: Multi-scale graph convolution for semi-supervised node classification. In *uncertainty in artificial intelligence*. PMLR, 841–851.

[2] James Atwood and Don Towsley. 2016. Diffusion-convolutional neural networks. *Advances in neural information processing systems* 29 (2016).

[3] Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. 2021. Beyond low-frequency information in graph convolutional networks. *arXiv preprint arXiv:2101.00797* (2021).

[4] Liu Bo, Zhiyong Che, Haowen Zhong, and Yanshan Xiao. 2021. A ranking based multi-view learning method for positive and unlabeled graph classification. *IEEE Transactions on Knowledge and Data Engineering* (2021).

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

[6] Yunpeng Chen, Haoqi Fan, Bing Xu, Zhicheng Yan, Yannis Kalantidis, Marcus Rohrbach, Shuicheng Yan, and Jiashi Feng. 2019. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3435–3444.

[7] Fan RK Chung and Fan Chung Graham. 1997. *Spectral graph theory*. Number 92. American Mathematical Soc.

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

[9] Yushun Dong, Kaize Ding, Brian Jalaian, Shuiwang Ji, and Jundong Li. 2021. Graph neural networks with adaptive frequency response filter. *arXiv preprint arXiv:2104.12840* (2021).

[10] Yushun Dong, Jian Kang, Hanghang Tong, and Jundong Li. 2021. Individual fairness for graph neural networks: A ranking based approach. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 300–310.

[11] Yushun Dong, Ninghao Liu, Brian Jalaian, and Jundong Li. 2021. EDITS: Modeling and Mitigating Data Bias for Graph Neural Networks. *arXiv preprint arXiv:2108.05233* (2021).

[12] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR, 1263–1272.

[13] Rafael C Gonzales and Richard E Woods. 2002. Digital Image Processing. New Jersey: PrenticeHall.

[14] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[15] Kaiming He, Jian Sun, and Xiaoou Tang. 2010. Guided image filtering. In *European conference on computer vision*. Springer, 1–14.

[16] Ruiqi Hu, Shirui Pan, Guodong Long, Qinghua Lu, Liming Zhu, and Jing Jiang. 2020. Going deep: Graph convolutional ladder-shape networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 2838–2845.

[17] Jeonghee Jo, Bumju Kwak, Byunghan Lee, and Sungroh Yoon. 2021. Flexible dual-branched message passing neural network for quantum mechanical property prediction with molecular conformation. *arXiv preprint arXiv:2106.07273* (2021).

[18] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[19] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[20] Chee-Kong Lee, Chengqiang Lu, Yue Yu, Qiming Sun, Chang-Yu Hsieh, Shengyu Zhang, Qi Liu, and Liang Shi. 2021. Transfer learning with graph neural networks

for optoelectronic properties of conjugated oligomers. *The Journal of Chemical Physics* 154, 2 (2021), 024906.

[21] Jundong Li, Xia Hu, Jiliang Tang, and Huan Liu. 2015. Unsupervised streaming feature selection in social media. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. 1041–1050.

[22] Qimai Li, Zhichao Han, and Xiao-Ming Wu. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. In *Thirty-Second AAAI conference on artificial intelligence*.

[23] Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang. 2018. Adaptive graph convolutional neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[24] Tinghuai Ma, Hongmei Wang, Lejun Zhang, Yuan Tian, and Najla Al-Nabhan. 2021. Graph classification based on structural features of significant nodes and spatial convolutional neural networks. *Neurocomputing* 423 (2021), 639–650.

[25] Hoang Nt and Takanori Maehara. 2019. Revisiting graph neural networks: All we have is low-pass filters. *arXiv preprint arXiv:1905.09550* (2019).

[26] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2019. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903* (2019).

[27] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. 2013. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine* 30, 3 (2013), 83–98.

[28] Peter C St. John, Caleb Phillips, Travis W Kemper, A Nolan Wilson, Yanfei Guan, Michael F Crowley, Mark R Nimlos, and Ross E Larsen. 2019. Message-passing neural networks for high-throughput polymer screening. *The Journal of chemical physics* 150, 23 (2019), 234111.

[29] Xiangguo Sun, Hongzhi Yin, Bo Liu, Hongxu Chen, Jiuxin Cao, Yingxia Shao, and Nguyen Quoc Viet Hung. 2021. Heterogeneous hypergraph embedding for graph classification. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 725–733.

[30] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 990–998.

[31] Lei Wang, Jing Ren, Bo Xu, Jianxin Li, Wei Luo, and Feng Xia. 2020. Model: Motif-based deep feature learning for link prediction. *IEEE Transactions on Computational Social Systems* 7, 2 (2020), 503–516.

[32] Yiwei Wang, Wei Wang, Yuxuan Liang, Yujun Cai, Juncheng Liu, and Bryan Hooi. 2020. Nodeaug: Semi-supervised node classification with data augmentation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 207–217.

[33] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*. PMLR, 6861–6871.

[34] Jiafei Wu, Chong Wang, and Yongze Xu. 2018. An Improved Guided Filtering Algorithm for Image Enhancement. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.

[35] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826* (2018).

[36] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. 2016. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*. PMLR, 40–48.

[37] Hui Yin, Yuanhao Gong, and Guoping Qiu. 2022. Guided Filter Bank. In *Intelligent Computing*. Springer, 783–792.

[38] Lingxiao Zhao and Leman Akoglu. 2019. Pairnorm: Tackling oversmoothing in gnns. *arXiv preprint arXiv:1909.12223* (2019).

[39] Dongmian Zou and Gilad Lerman. 2019. Encoding robust representation for graph generation. In *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–9.