A MAGNETIC FRAMELET-BASED CONVOLUTIONAL NEURAL NETWORK FOR DIRECTED GRAPHS

Lequan Lin and Junbin Gao

Discipline of Business Analytics, The University of Sydney Business School
The University of Sydney, Camperdown, NSW 2006, Australia
llin0615@uni.sydney.edu.au, junbin.gao@sydney.edu.au

ABSTRACT

Spectral Graph Convolutional Networks (spectral GCNNs), a powerful tool for analyzing and processing graph data, typically apply frequency filtering via Fourier transform to obtain representations with selective information. Although research shows that spectral GCNNs can be enhanced by framelet-based filtering, the massive majority of such research only considers undirected graphs. In this paper, we introduce Framelet-MagNet, a magnetic framelet-based spectral GCNN for directed graphs (digraphs). The model applies the framelet transform to digraph signals to form a more sophisticated representation for filtering. Digraph framelets are constructed with the complex-valued magnetic Laplacian, simultaneously leading to signal processing in both real and complex domains. We empirically validate the predictive power of Framelet-MagNet over a range of state-of-the-art models in node classification, link prediction, and denoising.

Index Terms— Directed Graph, Graph Convolutional Neural Network, Magnetic Laplacian, Graph Framelets, Graph Framelet Transform

1. INTRODUCTION

Recent years have witnessed the surging popularity of research on graph convolutional neural networks (GCNNs) [1]. Through the integration of graph signals and topological structures in graph convolution, GCNNs usually produce more valuable insights than models that analyze data in isolation. Especially, spectral GCNNs define their graph convolution in the frequency domain, enabling the filtering of different frequency components in graph signals. However, the majority of studies on signal processing in spectral GCNNs only focus on undirected graphs [2, 3, 4]. In this paper, we aim to extend framelet-based signal processing to spectral GCNNs for directed graphs (digraphs).

Many directional relationships are naturally modelled as digraphs, such as citation relationships [5], website hyperlinks [6], and road directions [7]. Using graph edges to represent directional information backbones the exploration of more aspects of the underlying data, which usually provides more useful findings. Nonetheless, while spectral GCNNs assume the eigendecomposition of symmetric graph Laplacian to provide real-valued eigenvalues and orthonormal eigenvectors, the digraph Laplacian is asymmetric. Converting digraphs to undirected graphs facilitates the extension of spectral methods to digraphs, but destroys the digraph structure. Therefore, many recent studies engage in designing symmetric digraph Laplacian that can preserve the directional information [8, 9, 10, 11]. Magnetic Laplacian [11, 12] is one of the most successful instances. It is a complex-valued Hermitian matrix, whose real part shows edge existence, and imaginary part indicates edge directions. Magnetic Laplacian-based digraph networks exploit magnetic Laplacian in classic spectral GCNN architectures and have demonstrated their power in various graph tasks [11].

Classic spectral GCNNs adopt Fourier transform in their convolutional layer. Converting graphs signals to the Fourier frequency domain allows the processing of signal frequencies, but only from a global perspective. More specifically, although we can detect signal frequencies, we cannot identify their position in the graph. To investigate both global and local information, we can ensemble spectral GCNNs with framelet transform instead. The framelet frequency domain is composed of graph framelets, which are constructed through dilation and translation of a set of localized scaling functions. Nevertheless, most existing wavelet/framelet-based networks are solely applicable to undirected graphs [2, 3, 4]. Although SVD-GCN [13] accomplishes digraph framelet transform via singular value decomposition (SVD) of the asymmetric digraph Laplacian, its theoretical rationale is very vague, for example, how the Laplacian frequency can be linked to the signal frequency in the SVD domain.

In this paper, we propose Framelet-MagNet, a magnetic Laplacian-assisted framelet-based spectral GCNN for digraphs. Multiresolution Analysis enables us to construct digraph framelets with the magnetic Laplacian and a filter bank [14, 15]. In addition, we also construct quasi-framelet directly in the frequency domain to impose double regulation on digraph signals [4]. We exploit Chebyshev polynomial approximation for fast framelet transform and reconstruction.

The contributions of our work are threefold. (1) To our best knowledge, this is the first attempt to construct a framelet-based digraph GCNN without discarding the role of Laplacian eigendecomposition. (2) We realize framelet-based convolution on digraph data in both real and complex domains, which enriches the basis for signal processing. (3) Vast experiment results validate the superiority of our model over the state-of-the-art approaches in various digraph tasks.

2. THE METHOD

2.1. Magnetic Laplacian

Magnetic Laplacian, a complex-valued Hermitian matrix, is a digraph representation whose real part indicates edge existence and imaginary part shows edge directions. Let $\mathcal{G}_d\{\mathcal{V}, \mathcal{E}, \mathbf{A}\}$ be a digraph, where $\mathcal{V}$ is a set of N vertices, $\mathcal{E}$ is a set of edges, and $\mathbf{A}$ is the adjacency matrix. The first step to construct the magnetic Laplacian is to decompose the adjacency matrix into a symmetric part $\mathbf{A}_{sym}$ and a skew-symmetric part $\mathbf{A}_{skew}$ as following:

$$\mathbf{A}_{sym}(i,j) = \frac{1}{2}(\mathbf{A}(i,j) + \mathbf{A}(j,i)), \quad 1 \le i,j \le N,$$

$$\mathbf{A}_{skew}(i,j) = \mathbf{A}(i,j) - \mathbf{A}(j,i), \quad 1 \le i,j \le N.$$

The degree matrix corresponding to the symmetric part is denoted as $\mathbf{D}_{sym}$. Then, the magnetic Laplacian (normalized) can be defined as

$$\mathcal{L}^{(q)} = \mathbf{I} - \mathbf{\Psi}^{(q)} \odot \left(\mathbf{D}_{sym}^{-\frac{1}{2}} \mathbf{A}_{sym} \mathbf{D}_{sym}^{-\frac{1}{2}} \right), \quad (1)$$

where

$$\mathbf{\Psi}^{(q)}(i,j) = \exp\left(2\pi i q \mathbf{A}_{skew}(i,j)\right), \quad 1 \le i,j \le N.$$

Here q in $\mathbf{\Psi}^{(q)}$ is a non-negative electric charge parameter [11, 12]. q is designed as a hyperparameter whose range is between 0 and 0.25, where higher q allows the Laplacian to encode more directional information. When we set $q = 0$, no directional information will be stored. Since the complex component $\mathbf{\Psi}^{(q)} \odot \mathbf{A}_{sym}$ is Hermitian, and the degree matrix is diagonal, the magnetic Laplacian is Hermitian as well. According to [11], the magnetic Laplacian is positive semi-definite, hence it supports the eigendecomposition required by spectral GCNNs.

2.2. Magnetic Graph Framelet Transform

The most important component in framelet-based spectral GCNNs is the framelet convolution. We firstly transform graph signals to the framelet frequency domain for filtering, then convert the processed data back to the spatial domain with the framelet reconstruction function. Intuitively, we desire no information loss during the whole process, which

Fig. 1. Framelet-MagNet for a node classification task. In the framelet convolutional layer, graph signal is decomposed into low-pass and high-pass components stored in a long vector, then processed by filter g_ω before we convert it back to the spatial domain and apply activation. Then, we unwind the new representation and apply a fully connected linear layer for classification.

means we expect the framelet transform to be "tight". Accordingly, we design the Magnetic Graph Framelet Transform (MGFT), which is a tight framelet transform defined on digraphs. The fundamental principle of MGFT is to incorporate magnetic Laplacian in the traditional undecimated tight framelet transform on undirected graphs. For a review of the traditional approaches and their theoretical background, we refer readers to [14, 15, 16].

As the magnetic Laplacian is positive semi-definite, we can write it as $\mathcal{L}^{(q)} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^*$, where $*$ denotes conjugate transpose. We let $\{(u_k, \lambda_k)\}_{k=0}^{N-1}$ be the eigenvector and eigenvalue pairs of $\mathcal{L}^{(q)}$. With the transition position $n \in \mathcal{V}$ and the dilation level $s = 1, ..., S$, we define the low-pass and high-pass magnetic graph framelets $\rho_{n,s}^{(q)}$ and $\varrho_{n,s,r}^{(q)}$ as

$$\rho_{n,s}^{(q)}(m) = \sum_{k=0}^{N-1} u_k(m) \hat{\zeta}_0 \left(\frac{\lambda_k}{2^s} \right) u_k^*(n),$$

$$\varrho_{n,s,r}^{(q)}(m) = \sum_{k=0}^{N-1} u_k(m) \hat{\zeta}_r \left(\frac{\lambda_k}{2^s} \right) u_k^*(n), \ 1 \le r \le R, \quad (2)$$

where $Z = \{\zeta_0, ..., \zeta_R\}$ is a set of real-valued scaling functions. The low-pass and high-pass framelets will decompose graph signals into low and high frequency components during transform. According to [14], we can find the appropriate set of scaling functions via Multiresolution Analysis. Basically, we derive scaling functions from a filter bank $a = \{a_0, ..., a_R\}$ defined in the spatial domain with the following relationship

$$\hat{\zeta}_r(2\delta) = \hat{a}_r(\delta)\hat{\zeta}_0(\delta), \quad (3)$$

for $r = 1, ..., R$ and any $\delta \in \mathbb{R}$. Then, "quasi-framelet" proposed by Yang et al. [4] relaxes the requirement of Multiresolution Analysis by straightforwardly constructing a quasi filter bank $b = \{b_0, ..., b_R\}$ in the Fourier domain. By definition, b should satisfy the identity condition

$$\sum_{r=0}^{R} b_r(\delta)^2 \equiv 1, \quad \forall \delta \in [0, \pi], \tag{4}$$

such that the value of b_0 decreases from 1 to 0 while the value of b_R increases from 0 to 1 over the Fourier domain $[0, \pi]$. This will allow framelet convolution to impose "double regulation" on the graph signals. More specifically, graph signals are regulated by not only the learnable filter in traditional convolutions, but also the modulation functions $b_0, ..., b_R$, where b_0 and b_R attenuate high and low frequency components, and the rest regulates frequency in between. In the following discussions, we denote $\hat{a} = \{\hat{a}_0, ..., \hat{a}_R\}$ and $b = \{b_0, ..., b_R\}$ collectively as $z = \{z_0, ..., z_R\}$ for simplicity. With a single signal $x \in \mathbb{R}^N$, we can define MGFT as

$$\begin{aligned}
\left\{ \langle x, \rho_{n,S}^{(q)} \rangle \right\}_{n \in \mathcal{V}} &= \mathcal{F}_{0,S}^{(q)} x, \\
\left\{ \langle x, \varrho_{n,s,r}^{(q)} \rangle \right\}_{n \in \mathcal{V}} &= \mathcal{F}_{r,s}^{(q)} x, \quad 1 \leq r \leq R, 1 \leq s \leq S,
\end{aligned} \tag{5}$$

where for $s = 1$, $\mathcal{F}_{r,1}^{(q)} = \mathbf{U} z_r \left(\frac{\mathbf{\Lambda}}{2^M} \right) \mathbf{U}^*$ and for $s = 2, ..., S$,

$$\mathcal{F}_{r,s}^{(q)} = \mathbf{U} z_r \left(\frac{\mathbf{\Lambda}}{2^{M+s-1}} \right) z_0 \left(\frac{\mathbf{\Lambda}}{2^{M+s-2}} \right) \cdots z_0 \left(\frac{\mathbf{\Lambda}}{2^M} \right) \mathbf{U}^*.$$

In these equations, M is the smallest number such that $\lambda_{\max} \leq 2^M \pi$, and $z_r \left(\frac{\mathbf{\Lambda}}{2^M} \right) = \text{diag} \left(z_r \left(\frac{\lambda_\mathbf{k}}{2^M} \right) \right)$ for $r = 0, ..., R$ and $k = 0, ..., N - 1$. We can use a vertically stacked transform matrix $\mathcal{F}^{(q)} = \left[\mathcal{F}_{0,S}^{(q)}; \mathcal{F}_{1,1}^{(q)}; ...; \mathcal{F}_{1,S}^{(q)}; ...; \mathcal{F}_{R,S}^{(q)} \right]$ to express magnetic framelete transform more concisely as

$$F^{(q)} = \mathcal{F}^{(q)} x, \tag{6}$$

where $F^{(q)}$ can be considered as framelet coefficients. Then, magnetic framelet reconstruction is given by

$$\tilde{x} = \mathcal{F}^{(q)*} F^{(q)}. \tag{7}$$

Recall that we expect MGFT to be tight, that is, $x = \tilde{x}$. We can achieve this by selecting appropriate filter banks. Existing examples include `Haar` [14], `Linear` [14], `Quadratic` [14], `Sigmoid` [4], and `Entropy` [4]. We will exploit them in our experiment.

Employing the fast computation proposed in [16], we approximate the filter bank z with with Chebyshev polynomials $\tilde{T}_r(\cdot)$, $r = 0, ..., R$. We propose Fast Magnetic Framelet Transform (FMFT) that defines the transform operator as $\tilde{\mathcal{F}}^{(q)} = \left[\tilde{\mathcal{F}}_{0,S}^{(q)}; \tilde{\mathcal{F}}_{1,1}^{(q)}; ...; \tilde{\mathcal{F}}_{1,S}^{(q)}; ...; \tilde{\mathcal{F}}_{R,S}^{(q)} \right]$, where for s = 1, $\tilde{\mathcal{F}}_{r,1}^{(q)} = \mathcal{T}_r(2^{-M} \mathcal{L}^{(q)})$, and for $s = 2, ..., S$, $\mathcal{F}_{r,s}^{(q)} = \mathcal{T}_r(2^{-M-s+1} \mathcal{L}^{(q)}) \mathcal{T}_0(2^{-M-s+2} \mathcal{L}^{(q)}) \cdots \mathcal{T}_0(2^{-M} \mathcal{L}^{(q)})$. Accordingly, the framelet transform and reconstruction are approximated by $F^{(q)} \approx \tilde{\mathcal{F}}^{(q)} x$ and $x \approx \tilde{\mathcal{F}}^{(q)*} F^{(q)}$.

2.3. Framelet-Magnet

With the FMFT, we define the ith magnetic framelet convolutional layer as

$$\sigma(g_{\omega_i} * X_{i-1}) = \sigma \left(\tilde{\mathcal{F}}^{(q)*} \left(\text{diag}(\omega_i) \left(\tilde{\mathcal{F}}^{(q)} (\mathbf{X}_{i-1} \mathbf{W}_i) \right) \right) \right),$$

where σ is a non-linear activation function, $g_{\omega_i} = \text{diag}(\omega_i)$ is a learnable filter, $\mathbf{X}_{i-1}$ is an $N \times D_{i-1}$ feature matrix with $\mathbf{X}_0$ being the original graph feature matrix, and $\mathbf{W}_i$ is a $D_{i-1} \times D_i$ matrix, where D_{i-1} and D_i are dimensions of the input and output channels.

Framelet-Magnet is composed of one or multiple magnetic framelet convolutional layers, an unwind operator, and a fully connected linear layer before the output layer. Suppose we have C convolutional layers, then we will obtain an $N \times D_C$ complex-valued graph representation. The purpose of the unwind operator is to unwind this representation to an $N \times 2D_C$ real-valued representation for further processing. In node classification tasks, we use this real representation directly for prediction as in **Fig. 1**. In link prediction tasks, on the other hand, we will concatenate the rows corresponding to the node pair connected by each edge before feeding it to the following layers.

3. EXPERIMENTS

We compare our model, Framelet-MagNet, with 10 state-of-the-art models in node classification, link prediction, and denoising. Link prediction consists of two different tasks, link existence prediction and link direction prediction.

3.1. Datasets and Implementation Details

Node classification experiment is a semi-supervised task based on two citation datasets, CORA_ML and CITESEER [24], and a WebKB dataset, CORNELL [25]. For citation datasets, following the experiments in [18], we use 20 labels from each class for training, 500 labels for validation, and the rest for testing. For CORNELL, we use a 60%/20%/20% train/validation/test split. Link prediction tasks are conducted on CORA_ML, CORNELL, and a WikipediaNetwork dataset, CHAMELEON [26]. We remove 5% edges for validation, 15% edges for testing, and we keep the rest of the edges for training, such that the number of nodes remains constant after the split. The experiments are conducted on 10 random subsets from each dataset, and we will evaluate the models with the average results. We use node attributes as the input features for node classification. For link prediction, we identify edges through ordered node pairs and use in-degrees and out-degrees as input features to learn directly from the graph structure. Let $\mathcal{G}_d\{\mathcal{V}, \mathcal{E}, \mathbf{A}\}$ be a digraph. For an ordered node pair $v_i, v_j \in \mathcal{V}$, we define its label as following (1) existence prediction: 0 if $(v_i, v_j) \in \mathcal{E}$ and 1 otherwise; (2) direction prediction: 0 if $(v_i, v_j) \in \mathcal{E}$ and 1 if $(v_j, v_i) \in \mathcal{E}$. Note that

Table 1. Experiment Results: Node Classification and Link Prediction Accuracy (%)

Models	Node Classification			Link Existence			Link Direction		
	CORA_ML	CITESEER	CORNELL	CORA_ML	CORNELL	CHAMELEON	CORA_ML	CORNELL	CHAMELEON
ChebNet [17]	60.8 ± 3.3	53.3 ± 2.6	74.1 ± 2.8	50.1 ± 0.1	49.7 ± 1.4	50.1 ± 0.0	50.1 ± 0.2	49.6 ± 10.3	50.0 ± 0.0
GCN [18]	69.7 ± 2.0	60.1 ± 2.6	42.4 ± 5.7	73.1 ± 5.3	51.1 ± 3.7	$\mathbf{89.8 \pm 0.5}$	79.1 ± 1.5	52.0 ± 2.8	96.8 ± 0.6
APPNP [19]	79.4 ± 2.7	66.7 ± 2.0	42.7 ± 5.5	69.5 ± 3.9	61.4 ± 8.0	87.1 ± 4.9	81.9 ± 0.9	70.3 ± 10.9	97.4 ± 0.2
GraphSAGE [20]	78.7 ± 1.1	66.4 ± 1.3	69.2 ± 3.5	67.2 ± 3.7	63.5 ± 9.4	86.0 ± 0.5	69.1 ± 0.5	69.0 ± 7.4	94.2 ± 0.3
GIN [21]	78.7 ± 1.8	63.9 ± 2.2	48.1 ± 5.0	75.0 ± 3.4	65.5 ± 8.5	83.9 ± 7.1	84.2 ± 0.9	77.0 ± 7.1	97.6 ± 0.2
GAT [22]	81.2 ± 2.0	66.2 ± 1.7	45.4 ± 10.4	50.0 ± 0.2	51.4 ± 3.5	50.4 ± 1.0	50.0 ± 0.6	50.7 ± 3.1	51.6 ± 2.2
DGCN [23]	79.8 ± 1.5	65.9 ± 1.4	65.1 ± 6.1	60.6 ± 7.6	60.8 ± 10.1	86.3 ± 1.4	70.9 ± 1.4	58.7 ± 5.1	93.7 ± 6.4
Digraph [10]	76.7 ± 1.9	62.9 ± 1.8	54.6 ± 6.8	76.7 ± 1.9	54.6 ± 6.8	83.9 ± 11.4	73.2 ± 11.7	49.1 ± 2.7	92.2 ± 14.1
DiGCN [10]	76.5 ± 1.6	61.6 ± 1.9	54.3 ± 7.5	72.8 ± 7.7	65.6 ± 12.1	88.9 ± 0.6	83.4 ± 1.5	73.3 ± 15.3	97.4 ± 0.2
MagNet [11]	78.7 ± 2.2	64.6 ± 2.2	74.6 ± 4.4	77.1 ± 1.4	68.2 ± 7.0	$\mathbf{89.8 \pm 0.5}$	87.0 ± 0.6	78.6 ± 10.6	97.7 ± 0.2
Framelet-MagNet	$\mathbf{83.8 \pm 1.4}$	$\mathbf{67.8 \pm 1.5}$	$\mathbf{77.0 \pm 3.5}$	$\mathbf{78.1 \pm 1.2}$	$\mathbf{73.8 \pm 6.0}$	89.7 ± 0.4	$\mathbf{88.5 \pm 1.0}$	$\mathbf{86.7 \pm 5.7}$	$\mathbf{97.8 \pm 0.2}$
q	0.00	0.05	0.25	0.15	0.25	0.25	0.15	0.25	0.25
Framelet type	Sigmoid	Sigmoid	Quadratic	Haar	Haar	Linear	Haar	Linear	Sigmoid

Fig. 2. Denoising results on CORA_ML. Framelet-MagNet (red) achieves higher classification accuracy at every noise level over other models.

we choose only linked node pairs for the link direction task. Moreover, we use either asymmetric or symmetrized adjacency matrix to train spatial models. ChebNet is trained with only symmetrized Laplacian. For the rest of the models, we adopt the asymmetric adjacency matrix and construct their special Laplacians accordingly. Hyperparameters including the number of filters, the learning rate, and magnetic parameter q are tuned following grid search as common practice.

3.2. Experiment Results

The experiment results of node classification and link prediction tasks are shown in **Table 1**. In node classification, Framelet-MagNet presents a good performance across all three datasets with the highest classification accuracy. It improves the state-of-the-art accuracy by 2.6% on CORA_ML and 2.4% on CORNELL. Framelet-MagNet selects small q for citation datasets. We suggest that this is because classification of articles does not depend on directional relationship. No matter one paper cites or is cited by another paper, they are likely to be in the same category. In link existence prediction, our model achieves the best accuracy on CORA_ML and CORNELL, enhancing the state-of-the-art performance by 1% and 5.6%, respectively. In terms of CHAMELEON, GCN and MagNet have the highest accuracy of 89.8% while our model achieves the second best with an accuracy of

89.7%. In link direction prediction, our model again beats other approaches over all datasets. Especially, on CORNELL, Framelet-MagNet produces an accuracy that is 8.1% higher than the second best model. Compared with the node classification experiment, the optimal q value is larger, implying that directional information is very useful in link-level tasks.

3.3. Test of Denoising Capability

In real-life applications, it is inevitable that some information such as adversarial examples is harmful to model prediction [27]. Such information is considered as "noises". Intuitively, removing noises from graph signals will enhance predictive performance. This procedure is known as "denoising". We test the denoising capability of Framelet-MagNet on the disturbed CORA_ML dataset with node classification experiment settings. To perturb CORA_ML whose features are normalized numbers, we manually impose Gaussian distributed noises and regulate noise level by altering the distribution standard deviation. **Fig. 2** presents the results, based on which we conclude that Framelet-MagNet shows clear superiority over MagNet and GCN in the denoising task.

4. CONCLUSION

We propose Framelet-MagNet, a magnetic framelet-based spectral GCNN for digraphs, and demonstrate its power over state-of-the-art methods via empirical results. We realize framelet convolution and process digraph signals in a complex frequency domain to achieve effective filtering. However, due to the limitation of magnetic Laplacian, our method is not applicable to weighted mixed graphs. Besides, our link-level experiment has no clear definition of undirected edges, so we treat undirected information as noises. Future works may investigate the solutions to current limitations.

5. REFERENCES

[1] Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski, "Graph convolutional networks: a comprehensive review," *Computational Social Networks*, vol. 6, no. 1, pp. 1–23, 2019.

[2] Bingbing Xu, Huawei Shen, Qi Cao, Yunqi Qiu, and Xueqi Cheng, "Graph wavelet neural network," *arXiv:1904.07785*, 2019.

[3] Xuebin Zheng, Bingxin Zhou, Junbin Gao, Yuguang Wang, Pietro Lió, Ming Li, and Guido Montúfar, "How framelets enhance graph neural networks," in *ICML*, 2021.

[4] Mengxi Yang, Xuebin Zheng, Jie Yin, and Junbin Gao, "Quasi-framelets: Another improvement to graph neural networks," *arXiv:2201.04728*, 2022.

[5] Yuan An, Jeannette Janssen, and Evangelos E Milios, "Characterizing and mining the citation graph of the computer science literature," *Knowledge and Information Systems*, vol. 6, no. 6, pp. 664–678, 2004.

[6] Babak Abedin and Babak Sohrabi, "Graph theory application and web page ranking for website link structure improvement," *Behaviour & Information Technology*, vol. 28, no. 1, pp. 63–72, 2009.

[7] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *ICLR*, 2018.

[8] Federico Monti, Karl Otness, and Michael M Bronstein, "Motifnet: a motif-based graph convolutional network for directed graphs," in *IEEE DSW*, 2018, pp. 225–228.

[9] Yi Ma, Jianye Hao, Yaodong Yang, Han Li, Junqi Jin, and Guangyong Chen, "Spectral-based graph convolutional network for directed graphs," *arXiv:1907.08990*, 2019.

[10] Zekun Tong, Yuxuan Liang, Changsheng Sun, Xinke Li, David S. Rosenblum, and Andrew Lim, "Digraph inception convolutional networks," in *NeurIPS*, 2020.

[11] Xitong Zhang, Yixuan He, Nathan Brugnone, Michael Perlmutter, and Matthew J. Hirn, "MagNet: A neural network for directed graphs," in *NeurIPS*, 2021.

[12] Michaël Fanuel, Carlos M Alaiz, and Johan AK Suykens, "Magnetic eigenmaps for community detection in directed networks," *Physical Review E*, vol. 95, no. 2, pp. 022302, 2017.

[13] Chunya Zou, Andi Han, Lequan Lin, and Junbin Gao, "A simple yet effective SVD-GCN for directed graphs," *arXiv:2205.09335*, 2022.

[14] Bin Dong, "Sparse representation on graphs by tight wavelet frames and applications," *Applied and Computational Harmonic Analysis*, vol. 42, pp. 452–479, 2017.

[15] Xuebin Zheng, Bingxin Zhou, Yu Guang Wang, and Xiaosheng Zhuang, "Decimated framelet system on graphs and fast g-framelet transforms.," *Journal of Machine Learning Research*, vol. 23, pp. 18–1, 2022.

[16] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[17] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *NeurIPS*, 2016.

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

[19] Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann, "Predict then propagate: Graph neural networks meet personalized pagerank," in *ICLR*, 2019.

[20] Will Hamilton, Zhitao Ying, and Jure Leskovec, "Inductive representation learning on large graphs," in *NeurIPS*, 2019.

[21] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka, "How powerful are graph neural networks?," in *ICLR*, 2019.

[22] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio, "Graph attention networks," in *ICLR*, 2018.

[23] Zekun Tong, Yuxuan Liang, Changsheng Sun, David S Rosenblum, and Andrew Lim, "Directed graph convolutional network," *arXiv:2004.13970*, 2020.

[24] Aleksandar Bojchevski and Stephan Günnemann, "Deep Gaussian embedding of graphs: Unsupervised inductive learning via ranking," in *ICLR*, 2018.

[25] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang, "Geom-GCN: Geometric graph convolutional networks," in *ICLR*, 2020.

[26] Benedek Rozemberczki, Carl Allen, and Rik Sarkar, "Multi-scale attributed node embedding," *Journal of Complex Networks*, vol. 9, no. 2, pp. cnab014, 2021.

[27] Daniel Zügner, Amir Akbarnejad, and Stephan Günnemann, "Adversarial attacks on neural networks for graph data," in *ACM SIGKDD*, 2018, pp. 2847–2856.