MOTIFNET: A MOTIF-BASED GRAPH CONVOLUTIONAL NETWORK FOR DIRECTED GRAPHS

Federico Monti[1], *Karl Otness*[2], *Michael M. Bronstein*[1,2,3]

[1]University of Lugano [2]Harvard University [3]Tel Aviv University

ABSTRACT

Deep learning on graphs and in particular, graph convolutional neural networks, have recently attracted significant attention in the machine learning community. Many of such techniques explore the analogy between the graph Laplacian eigenvectors and the classical Fourier basis, allowing to formulate the convolution as a multiplication in the spectral domain. One of the key drawback of spectral CNNs is their explicit assumption of an undirected graph, leading to a symmetric Laplacian matrix with orthogonal eigendecomposition. In this work we propose MotifNet, a graph CNN capable of dealing with directed graphs by exploiting local graph motifs. We present experimental evidence showing the advantage of our approach on real data.

***Index Terms*—** Geometric Deep Learning, Graph Convolutional Neural Networks, Directed Graphs, Graph Motifs

1. INTRODUCTION

Deep learning on graph-structured data has recently gained popularity in the machine learning community due to the increased interest in dealing with applications such as social network analysis and recommendation systems. One of the key challenges of generalizing successful deep neural network architectures such as convolutional neural networks (CNNs) to graphs is the lack of vector space structure and shift-invariance, resulting in the need to re-invent the basic building blocks of CNNs, including convolutional filters and pooling.

Broadly speaking, we can distinguish between two classes of graph CNN formulations [1]. Spatial approaches [2, 3, 4] generalize the notion of 'patch of pixels' by constructing a local system of weights on the graph. Spectral approaches [5, 6, 7, 8, 9, 10] use the analogy between the eigenfunctions of the graph Laplacian and the classical Fourier transform, and define a convolution-like operation in the spectral domain. So far, these methods were limited to undirected graphs, a restriction arising from the requirement to have a symmetric Laplacian matrix in order to obtain orthogonal eigendecomposition. At the same time, a wide variety of graph data, including citation networks, are directed, which limits the application of existing methods.

In this paper, we introduce MotifNet, a graph CNN for directed graphs. Our approach uses convolution-like anisotropic graph filters bases on local sub-graph structures (motifs) [11, 12]. We use an attention mechanism, allowing MotifNet to generalize some standard graph CNN models without significantly increasing the model complexity. Experimental validation on real data shows superior performance compared to previous approaches.

2. BACKGROUND

Let us be given a weighted undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$ with vertices $\mathcal{V} = \{1, \ldots, n\}$, edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ s.t. $(i, j) \in \mathcal{E}$ iff $(j, i) \in \mathcal{E}$, and edge weights $w_{ij} \geq 0$ for $(i, j) \in \mathcal{E}$ and zero otherwise. The graph structure is represented by the $n \times n$ symmetric adjacency matrix $\mathbf{W} = (w_{ij})$. We define the normalized graph Laplacian $\mathbf{\Delta} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$, where $\mathbf{D} = \mathrm{diag}(\sum_{j \neq 1} w_{1j}, \ldots, \sum_{j \neq n} w_{nj})$ denotes the degree matrix. In the above setting, the Laplacian is a symmetric matrix and admits an eigendecomposition $\mathbf{\Delta} = \mathbf{\Phi}\mathbf{\Lambda}\mathbf{\Phi}^\top$ with orthonormal eigenvectors $\mathbf{\Phi} = (\boldsymbol{\phi}_1^\top, \ldots, \boldsymbol{\phi}_n^\top)$ and non-negative eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \ldots \lambda_n$ arranged into a diagonal matrix $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$.

We are interested in manipulating functions $f : \mathcal{V} \rightarrow \mathbb{R}$ defined on the vertices of the graph, which can be represented as vectors $\mathbf{f} \in \mathbb{R}^n$ and form a Hilbert space with the standard inner product $\langle \mathbf{f}, \mathbf{g} \rangle = \mathbf{f}^\top \mathbf{g}$. The eigenvectors of the Laplacian form an orthonormal basis in the aforementioned space of functions, allowing a Fourier decomposition of the form $\mathbf{f} = \mathbf{\Phi}\mathbf{\Phi}^\top \mathbf{f}$, where $\hat{\mathbf{f}} = \mathbf{\Phi}^\top \mathbf{f}$ is the *graph Fourier transform* of $\mathbf{f}$. The Laplacian eigenvectors thus play the role of the standard Fourier atoms and the corresponding eigenvalues that of frequencies. Finally, a convolution operation can be defined in the spectral domain by analogy to the Euclidean case as $\mathbf{f} \star \mathbf{g} = \mathbf{\Phi}(\hat{\mathbf{f}} \cdot \hat{\mathbf{g}}) = \mathbf{\Phi}(\mathbf{\Phi}^\top \mathbf{f}) \cdot (\mathbf{\Phi}^\top \mathbf{g})$

Bruna et al. [5] exploited the above formulation for designing graph convolutional neural networks, in which a basic layer has the following form:

$$\tilde{\mathbf{f}}_l = \xi \left(\sum_{l'=1}^{q'} \mathbf{\Phi} \hat{\mathbf{G}}_{ll'} \mathbf{\Phi}^\top \mathbf{f}_{l'} \right), \quad l = 1, \ldots, q, \tag{1}$$

where q', q denote the number of input and output channels, respectively, $\hat{\mathbf{G}}_{ll'} = \mathrm{diag}(\hat{g}_{ll',1}, \dots, \hat{g}_{ll',n})$ is a diagonal matrix of spectral multipliers representing the filter, and ξ is a nonlinearity (e.g. ReLU). Among the notable drawbacks of this architecture putting it at a clear disadvantage compared to classical Euclidean CNNs is high computational complexity ($\mathcal{O}(n^2)$ due to the cost of computing the forward and inverse graph Fourier transform, incurring dense $n \times n$ matrix multiplication), $\mathcal{O}(n)$ parameters per layer, and no guarantee of spatial localization of the filters. In order to cope with the two latter problems, Henaff et al. [6] argued that filter localization is achieved by smoothness of its Fourier transform, and proposed parametrizing the filter as a smooth spectral transfer function. In particular, filters of the form $\hat{g}_k = \tau_{\boldsymbol{\theta}}(\lambda_k) = \sum_{j=1}^{p} \theta_j \beta_j(\lambda_k)$ were considered, where $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_p)$ are the learnable filter parameters and $\beta_1(\lambda), \dots, \beta_p(\lambda)$ are spline basis functions.

Defferrard et al. [7] considered the spectral CNN framework with polynomial filters represented in the Chebyshev basis (referred to as ChebNet), which can be efficiently computed by applying powers of the graph Laplacian

$$\tilde{\mathbf{f}} = \mathbf{\Phi} \sum_{j=0}^{p} \theta_j T_j(\tilde{\mathbf{\Lambda}}) \mathbf{\Phi}^\top \mathbf{f} = \sum_{j=0}^{p} \theta_j T_j(\tilde{\mathbf{\Delta}}) \mathbf{f}, \tag{2}$$

and thus avoiding its eigendecomposition altogether. The computational complexity thus drops from $\mathcal{O}(n^2)$ to $\mathcal{O}(|\mathcal{E}|)$, and if the graph is sparsely connected, to $\mathcal{O}(n)$ (here $\tilde{\lambda}$ is a frequency rescaled in $[-1, 1]$, $\tilde{\mathbf{\Delta}} = 2\lambda_n^{-1}\mathbf{\Delta} - \mathbf{I}$ is the rescaled Laplacian with eigenvalues $\tilde{\mathbf{\Lambda}} = 2\lambda_n^{-1}\mathbf{\Lambda} - \mathbf{I}$, and $T_j(\lambda) = 2\lambda T_{j-1}(\lambda) - T_{j-2}(\lambda)$ denotes the Chebyshev polynomial of degree j, with $T_1(\lambda) = \lambda$ and $T_0(\lambda) = 1$).

Kipf and Welling [8] proposed a simplification of ChebNet (referred to as Graph Convolutional Network or GCN) by limiting the order of the polynomial to $p = 1$ and using a re-normalization of the Laplacian to avoid numerical instability. Despite the efficiency of ChebNet [7] and GCN [8], both methods struggle when dealing with graphs containing clustered eigenvalues, a phenomenon typical in community graphs. Levie et al. [10] used rational filter functions based on the Cayley transform, allowing to achieve better spectral resolution of the filters.

3. DEALING WITH DIRECTED GRAPHS

One of the key drawbacks of the above spectral constructions is the explicit assumption of an undirected graph – indeed, the existence of an orthonormal eigendecomposition of the Laplacian matrix crucially depends on the adjacency matrix $\mathbf{W}$ being symmetric, a property that is violated when the graph is directed. A further drawback is that the Laplacian operator is *isotropic*, i.e., has no preferred direction on the graph; consequently, the resulting spectral filters are rotationally symmetric when the underlying graph is a grid (see Figure 1). While a construction of anisotropic Laplacians and thus oriented filters is possible on manifolds due to a locally-Euclidean structure [3], it is more challenging on general graphs.

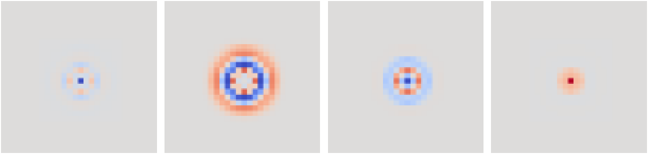

Fig. 1. Examples of Chebyshev filters of degree $p = 7$ on a regular grid. Note that the filters are isotropic due to rotational invariance of the Laplacian.

Benson et al. [12] proposed an elegant workaround these issues based on the analysis of small subgraphs called *motifs*. Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$ be a weighted directed graph (in which case $\mathbf{W}^\top \neq \mathbf{W}$), and let $\mathcal{M}_1, \dots, \mathcal{M}_K$ denote a collection of *graph motifs* (small directed graphs representing certain meaningful connectivity patterns; e.g., Figure 2 depicts thirteen 3-vertex motifs). For each edge $(i, j) \in \mathcal{E}$ of the directed graph $\mathcal{G}$ and each motif $\mathcal{M}_k$, let $u_{k,ij}$ denote the number of times the edge (i, j) participates in $\mathcal{M}_k$ (note that an edge can participate in multiple motifs). Benson et al. [12] define a new set of edge weights of the form $\tilde{w}_{k,ij} = u_{k,ij} w_{ij}$, which is now a *symmetric motif adjacency* matrix we denote by $\tilde{\mathbf{W}}_k$. The *motif Laplacian* $\tilde{\mathbf{\Delta}}_k = \mathbf{I} - \tilde{\mathbf{D}}_k^{-1/2} \tilde{\mathbf{W}}_k \tilde{\mathbf{D}}_k^{-1/2}$ associated with this adjacency acts *anisotropically* with a preferred direction along structures associated with the respective motif.

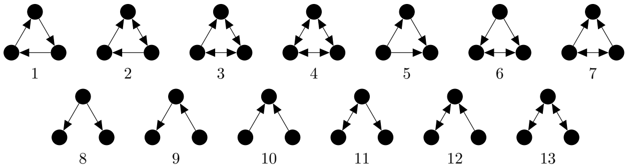

Fig. 2. Thirteen 3-vertex graph motifs used in this paper.

4. MOTIFNET

The key idea of this paper is using motif-induced adjacencies in the context of deep learning on graphs. We construct filters on the graph using *multivariate polynomial* filters of degree p applied to the motif Laplacian matrices. Note that since the matrix product is generally non-commutative (i.e., $\tilde{\mathbf{\Delta}}_i \tilde{\mathbf{\Delta}}_j \neq \tilde{\mathbf{\Delta}}_j \tilde{\mathbf{\Delta}}_i$), we have K^p products of the form $\tilde{\mathbf{\Delta}}_{k_1} \cdots \tilde{\mathbf{\Delta}}_{k_p}$, where $k_l \in \{1, \dots, K\}$. A general multivariate matrix polynomial has the form

$$\mathbf{P}_{\mathbf{\Theta}}(\tilde{\mathbf{\Delta}}_1, \dots, \tilde{\mathbf{\Delta}}_K) = \sum_{j=0}^{p} \sum_{k_1, \dots, k_j \in \{1, \dots, K\}} \theta_{k_1, \dots, k_j} \tilde{\mathbf{\Delta}}_{k_1} \cdots \tilde{\mathbf{\Delta}}_{k_j}, \tag{3}$$

where our convention is that for $j = 0$ we have only one zero-degree term $\theta_0 \mathbf{I}$, and $\mathbf{\Theta}$ denotes the set of all the coefficients. Overall, a polynomial of the form (3) has $\frac{1+K^{p+1}}{1-K}$

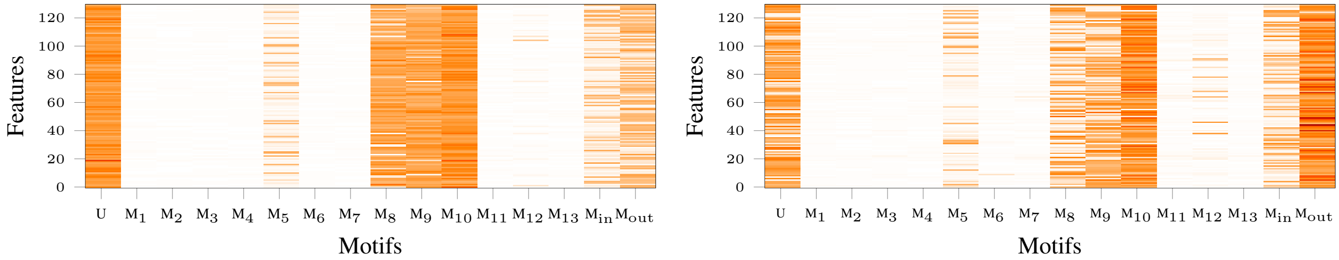

Fig. 3. Attention scores α obtained with a MotifNet of order $p = 1$ from the 1st and 2nd graph convolutional layers. Dark colors represent high probabilities, bright colors low ones. Only 7 of the 15 possible different motifs (considering the undirected adjacency matrix and the directed ones) appear as relevant for classifying the vertices of directed CORA.

coefficients, which is impractically large even for a modest number of motifs K or degree p.

We therefore study two possible simplifications of (3). First, we consider only $K = 2$ simple motifs corresponding to incoming and outgoing edges from a vertex. In this case, the polynomial becomes

$$\mathbf{P}_{\mathbf{\Theta}} = \theta_0\mathbf{I} + \theta_1\tilde{\mathbf{\Delta}}_1 + \theta_2\tilde{\mathbf{\Delta}}_2 + \theta_{11}\tilde{\mathbf{\Delta}}_1^2 + \ldots + \theta_{22}\tilde{\mathbf{\Delta}}_2^2 + \ldots \quad (4)$$

Second, we consider a simplified version of multivariate polynomials (3) defined recursively in the following manner,

$$\begin{aligned} \mathbf{P}_{\mathbf{\Theta}}(\tilde{\mathbf{\Delta}}_1, \ldots, \tilde{\mathbf{\Delta}}_K) &= \sum_{j=0}^{p} \theta_j \mathbf{P}_j; \qquad (5) \\ \mathbf{P}_j(\tilde{\mathbf{\Delta}}_1, \ldots, \tilde{\mathbf{\Delta}}_K) &= \sum_{k=1}^{K} \alpha_{k,j} \tilde{\mathbf{\Delta}}_k \mathbf{P}_{j-1}, \ \ j = 1, \ldots, p \\ \mathbf{P}_0 &= \mathbf{I}, \end{aligned}$$

where $0 \leq \alpha_{i,j} \leq 1$ and $\mathbf{\Theta} = (\theta_0, \ldots, \theta_p, \alpha_{1,1}, \ldots, \alpha_{K,p})$ denotes the set of coefficients, $Kp + 1$ in total.

MotifNet is a neural network architecture employing convolutional layers of the form

$$\tilde{\mathbf{f}}_l = \xi\left(\sum_{l'=1}^{q'} \mathbf{P}_{\mathbf{\Theta}_{ll'}}(\tilde{\mathbf{\Delta}}_1, \ldots, \tilde{\mathbf{\Delta}}_K)\mathbf{f}_{l'}\right), \quad l = 1, \ldots, q, \quad (6)$$

where q', q denote the number of input and output channels, respectively, and $\mathbf{P}_{\mathbf{\Theta}_{ll'}}$ is the simplified multivariate matrix polynomial (4) or (5). ChebNet is obtained as a particular instance of MotifNet with a single Laplacian of an undirected graph, in which case a univariate matrix polynomial is used.

5. EXPERIMENTAL EVALUATION

We tested our approach on the directed CORA citation network [13]. The vertices of the CORA graph represent 19,793 scientific papers, and directed edges of the form (i, j) represent citation of paper j in paper i. The content of each paper is represented by a vector of 8,710 numerical features (term frequency-inverse document frequency of various words that appear in the corpus), to which we applied PCA taking the first 130 components. The task is to classify the papers into one of the 70 different categories.

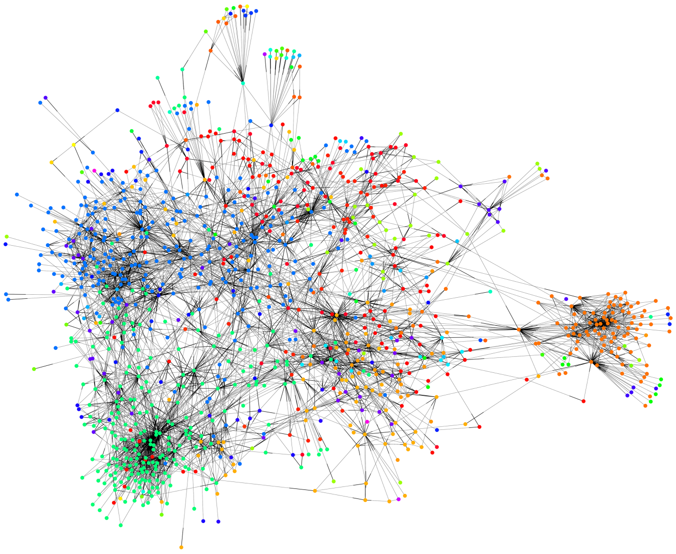

Fig. 4. Portion of the directed CORA dataset obtained extracting 1,000 vertices from the biggest weakly connected component. Different colors represent different classes.

We considered the semi-supervised learning setting [8] using 10% of the available vertices for training, 10% for validation and 10% for testing. We compared the following graph CNN architectures: ChebNet (applied to an undirected version of CORA and to the directed adjacency matrices $\mathbf{W}$ and $\mathbf{W}^\top$) and two versions of MotifNet using simplified multivariate matrix polynomial (4) or (5), to which we refer as MotifNet-m and MotifNet-d, respectively. All the graph CNNs contained two convolutional layers and a final fully connected layer followed by softmax. MotifNet-d contained only in/outgoing edges (denoted by $\mathcal{M}_{\text{in}}$ and $\mathcal{M}_{\text{out}}$). For MotifNet-m, we considered 13 motifs formed by triplets of vertices (shown in Figure 2 and denoted $\mathcal{M}_1, \ldots, \mathcal{M}_{13}$). To reduce the computational complexity of our model, we selected a subset of motifs in the following way. First, we trained MotifNet with order $p = 1$ and selected the model with minimum cross-entropy on the validation set. By analyzing the probabilities learned by our model, we discovered

Table 1. Number of parameters required by the considered GCNs. MotifNet requires just a handful of additional parameters for handling the considered adjacency matrices.

Order	ChebNet	MotifNet-m	MotifNet-d
1	94K	95K	128K
2	128K	131K	263K
3	162K	166K	534K
4	196K	202K	1,074K
5	229K	237K	2,156K
6	263K	272K	4,319K
7	297K	308K	8,646K
8	331K	343K	17,298K

that only several motifs (depicted in Figure 2) turned out to be relevant. Then, we used this subset of motifs $\mathcal{M}_5, \mathcal{M}_8, \mathcal{M}_9$ in addition to the undirected graph obtained by replacing each directed edge with an undirected one (denoted by $\mathcal{U}$) and $\mathcal{M}_{\text{in}}$ and $\mathcal{M}_{\text{out}}$[1].

All models were trained on NVIDIA Titan X GPU. Dropout with keep probability of 0.5 and weight decay with constant $\gamma = 10^{-3}$ were used as regularization. Adam [14] optimization method was used to train the models with learning rate equal to 10^{-3}.

The results we obtained are reported in Figures 5-6. MotifNet-m consistently outperforms the baseline (ChebNet) for a variety of different polynomial orders, at the expense of only a tiny increase in the number of parameters (Table 1).

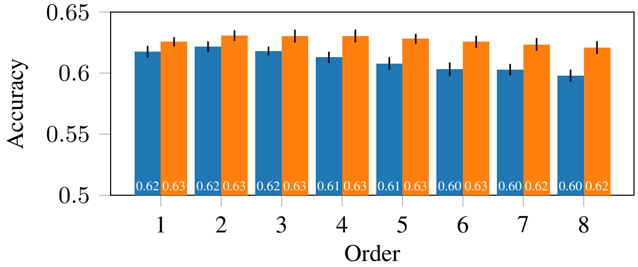

Fig. 5. Classification accuracy on CORA obtained with ChebNet on undirected graph (blue) and MotifNet-m (orange).

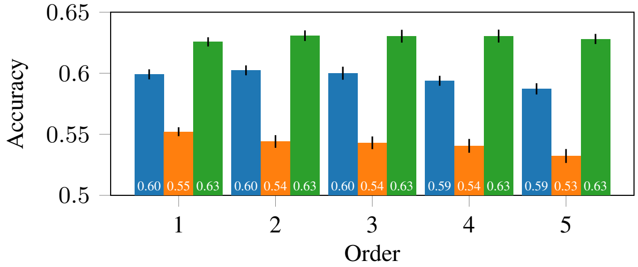

Fig. 6. Classification accuracy on CORA obtained with ChebNet applied with adjacency matrix $\mathbf{W}$ (blue) / $\mathbf{W}^\top$ (red), MotifNet-d (green) and MotifNet-m (orange).

[1] $\mathcal{M}_{10}$ has been discarded in our final architecture because of the dense motif adjacency matrix it was presenting.

6. REFERENCES

[1] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[2] J. Masci, D. Boscaini, M. Bronstein, and P. Vandergheynst, "Geodesic convolutional neural networks on riemannian manifolds," in *Proc. 3dRR*, 2015.

[3] D. Boscaini, J. Masci, E. Rodolà, and M. M. Bronstein, "Learning shape correspondence with anisotropic convolutional neural networks," in *Proc. NIPS*, 2016.

[4] F. Monti, D. Boscaini, J. Masci, E. Rodolà, J. Svoboda, and M. M. Bronstein, "Geometric deep learning on graphs and manifolds using mixture model cnns," in *Proc. CVPR*, 2017.

[5] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv:1312.6203*, 2013.

[6] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," *arXiv:1506.05163*, 2015.

[7] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. NIPS*, 2016.

[8] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2017.

[9] F. Monti, M. M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," in *Proc. NIPS*, 2017.

[10] R. Levie, F. Monti, X. Bresson, and M. M Bronstein, "Cayleynets: Graph convolutional neural networks with complex rational spectral filters," *arXiv:1705.07664*, 2017.

[11] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, 2002.

[12] A. R. Benson, D. F. Gleich, and J. Leskovec, "Higher-order organization of complex networks," *Science*, vol. 353, no. 6295, pp. 163–166, 2016.

[13] Aleksandar Bojchevski and Stephan Günnemann, "Deep gaussian embedding of attributed graphs: Unsupervised inductive learning via ranking," *arXiv:1707.03815*, 2017.

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.