Notes on Graph Spectral Theory

ACM Reference Format:
. 2023. Notes on Graph Spectral Theory. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 THE MAIN CHALLENGES OF SIGNAL PROCESSING ON GRAPHS

Graphs are widely used thanks to their generic data representation forms which are useful for describing the geometric structures of data domains in very different application fields, including social, energy, transportation, and brain modelisation. However, the classical tools of the signal processing techniques can not be used seamlessly/candidly on graphs unlike audio signal and images [3].

- The graphs have no inherent ordering of the vertices, unlike the images where pixel is uniquely identified by its position within the image. We therefore need algorithms that are node-order equivariant: they should not depend on the ordering of the nodes.
- Many graphs are irregular structures that lack a shift-invariant notion of translation, which is a key property of the Fourier transform [1].
- The analogous spectrum in the graph setting is discrete and irregularly spaced, and it is therefore non-trivial to define an operator that correspond to translation in the graph spectral domain.
- We may need to downsample a graph, and therefore we need a method to generate a coarser version of the graph that somehow captures the structural properties embedded in the original graph.
- We need localized transforms that compute information about the data at each vertex by using data from a small neighbourhood of vertices close to it in the graph

Graph can also be very large but, in our study, we consider sparse population graphs, i.e. individuals or nodes are connected to a limited number of nodes/number of edges is linear in the number of nodes. Spectral graph theory has enabled constructing, analyzing, and manipulating graphs. In signal processing on graphs, it is leveraged as a tool to define frequency spectra and expansion bases for graph Fourier transforms.

In this section, we present some basic definitions from spectral graph theory that will be needed to apply neural networks on graphs. As stated previously, we consider an undirected, connected, weighted graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 THE NON-NORMALIZED AND NORMALIZED GRAPH LAPLACIAN

The non-normalized graph Laplacian, also called the combinatorial graph Laplacian, is defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where the degree matrix $\mathbf{D}$ is a diagonal matrix whose ith diagonal element, d_i, is equal to the sum of the weights of all edges incident to vertex i.

The graph Laplacian $\mathbf{L}$ is a real symmetric matrix, it has therefore real, non-negative eigenvalues $\{\lambda_l\}_{l=0,\ldots,N-1}$. We denote their associated orthonormal eigenvectors by $\{u_l\}_{l=0,\ldots,N-1}$, As we considered connected graphs, the eigenvalue $\lambda_0 = 0$ has multiplicity 1 [3].

A popular practice is to normalized each weight $W_{i,j}$ by a factor of $\frac{1}{\sqrt{d_i d_j}}$. Doing so leads to the *normalized graph Laplacian*, which is defined as $\tilde{\mathbf{L}} = D^{-1/2}\mathbf{L}D^{-1/2} = I_N - D^{-1/2}\mathbf{W}D^{-1/2}$

3 A GRAPH FOURIER TRANSFORM AND NOTION OF FREQUENCY

The Fourier transform for analogous function f

$$\hat{f}(\xi) = \left\langle f \middle| e^{2\pi i \xi t} \right\rangle = \int_{\mathbb{R}} f(t) e^{-2\pi i \xi t} dt$$

is the expansion of a function f in terms of the complex exponentials, which are the eigenfunctions of the one-dimensional Laplace operator Δ :

$$-\Delta(e^{2\pi i \xi t}) = -\frac{\partial^2}{\partial t^2} e^{2\pi i \xi t} = (2\pi i \xi)^2 e^{2\pi i \xi t}$$

Similarly, we can define the *Graph Fourier Transform* $\hat{f}$ of any function on the vertices of $\mathcal{G}$ as the expansion of f in terms of the eigenvectors of the graph Laplacian :

$$\hat{f}(\lambda_l) = \langle f | u_l \rangle = \sum_{i=1}^{N} f(i) u_l^*(i) \tag{1}$$

where $u_l^*(i)$ is the conjugate of $u_l(i)$. The *inverse graph Fourier transform* is then given by

$$f(i) = \sum_{i=1}^{N} \hat{f}(\lambda_i) u_l(i) \tag{2}$$

Note that, in our case, the signal is $f : \mathcal{V} \rightarrow \mathbb{R}^N$ that associate a feature vector to each node of the graph.

4 SPECTRAL GRAPH CONVOLUTIONS

We consider spectral convol [2] polynomials of Laplacian of degree d : the node v is convolved with nodes that are at most at a distance d. Thus, these polynomials filters are localized.

5 A SIMPLE APPLICATION

REFERENCES

[1] Ameya Daigavane, Balaraman Ravindran, and Gaurav Aggarwal. [n. d.]. Understanding Convolutions on Graphs. 6, 9 ([n. d.]), e32. https://doi.org/10.23915/distill.00032

[2] Thomas N. Kipf and Max Welling. [n. d.]. Semi-Supervised Classification with Graph Convolutional Networks. arXiv:1609.02907 [cs, stat] http://arxiv.org/abs/1609.02907

[3] David I. Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. [n. d.]. The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains. 30, 3 ([n. d.]), 83–98. https://doi.org/10.1109/MSP.2012.2235192 arXiv:1211.0053 [cs]