1. Nonlinear Laplacian for Digraphs and its Application for Network Analysis
2. Rare Category Detection on Time-Evolving Graphs
NONLINEAR LAPLACIAN FOR DIGRAPHS . . .

OUTLINE

1. Introduction
2. Preliminaries
3. Related Works
4. Spectral Theory for Digraphs
5. Experiments
INTRODUCTION

Spectral Graph Theory:
Relations between graph theoretic measures and eigenvalues and eigenvectors of Laplacian

Laplacian \[ L = D - A, \]

Normalized Laplacian \[ \mathcal{L} = D^{-1/2}LD^{-1/2} = \sqrt{I - D^{-1/2}AD^{-1/2}} \]

Where D is Diagonal degree matrix and A is adjacency matrix
PRELIMINARIES FOR UNDIRECTED GRAPHS

Volume of a Node Set:  \( \text{vol}(S) = \sum_{v \in S} d_v \).

Cut of a Node Set:  \( \text{cut}(S) = |\partial(S)| \), where  \( \partial(S) = \{\{u, v\} \in E \mid u \in S, v \in V \setminus S\} \).

Conductance of a Node set:  \( \phi(S) = \frac{\text{cut}(S)}{\min\{\text{vol}(S), \text{vol}(V \setminus S)\}} \).

Conductance of Graph:  \( \phi_G = \min_{\emptyset \subsetneq S \subsetneq V} \phi(S) \).
PRELIMINARIES FOR DIGRAPHS

Out degree: $d^+_v$ and In degree: $d^-_v$

Degree: $d_v = d^+_v + d^-_v$

Cut+: $\text{cut}^+(S) = |\partial^+(S)|$, where $\partial^+(S) = \{(u, v) \in E \mid u \in S, v \in V \setminus S\}$

Out-Conductance: $\phi^+(S) = \frac{\text{cut}^+(S)}{\min\{\text{vol}(S), \text{vol}(V \setminus S)\}}$

Conductance: $\phi(S) = \frac{\min\{\text{cut}^+(S), \text{cut}^+(V \setminus S)\}}{\min\{\text{vol}(S), \text{vol}(V \setminus S)\}}$

Conductance of Graph: $\phi_G = \min_{\emptyset \neq S \subseteq V} \phi(S)$.
RELATED WORK

Chung’s Normalized Laplacian:  \[ \mathcal{L}_\pi = \Pi^{-1/2} L \Pi^{-1/2} \]

Where \( \Pi \) is diagonal matrix with \( \Pi_{u,u} = \pi_u \), \( \pi \) is stationary distribution.

Following inequality holds for Chung’s Normalized inequality:

\[ \frac{\lambda}{2} \leq \min_{\emptyset \subseteq S \subseteq V} \phi_\pi(S) \leq \sqrt{2\lambda}. \]

\( \phi_\pi(S) \) is the conductance with respect to random walk process.

Where,

\[ \pi(S) = \sum_{u \in S} \pi_u \]
\[ \text{cut}_\pi(S) = \sum_{(u,v) \in E} \frac{\pi_u}{d_u} \]
Second eigenvector of Chung’s Normalized Laplacian turns out be minimizer of

\[ \sum_{(u,v) \in E} (x_u - x_v)^2 \frac{\pi_u}{d_u^+} \]

Where \( \sum x_u^2 \pi_u = 1 \) and \( x \) is a variable vector.

Arc \((u,v)\) brings nodes \(u\) and \(v\) closer in spectral ordering.

The effect is larger when \(\pi_u\) is larger.
SPECTRAL THEORY FOR DIGRAPHS

Definition 4.1 Let $G = (V, E)$ be a digraph. Given a vector $x \in \mathbb{R}^V$, the nonlinear Laplacian $L_G : \mathbb{R}^V \rightarrow \mathbb{R}^V$ computes a vector $L_G(x)$ as follows:

1. We construct an undirected graph $G_\infty$ on the vertex set $V$ as follows: for each $(u, v) \in E$, we add an edge $\{u, v\}$ if $x_u \geq x_v$, and we add self-loops $\{u, u\}$ and $\{v, v\}$ otherwise.
2. Let $L_{G_\infty}$ be the Laplacian of $G_\infty$.
3. Then, $L_G(x) = L_{G_\infty}x$. 

(a) $G$ and $x$

(b) $G_\infty$ and $L_G(x)$
Definition 4.2 Let $G = (V, E)$ be a digraph. Given a vector $x \in \mathbb{R}^V$, the normalized nonlinear Laplacian $\mathcal{L}_G : \mathbb{R}^V \to \mathbb{R}^V$ computes a vector $\mathcal{L}_G(x)$ as follows:

1. We construct an undirected graph $G_\infty$ on the vertex set $V$ as follows: for each $(u, v) \in E$, we add an edge \{u, v\} if $\frac{x_u}{\sqrt{d_u}} \geq \frac{x_v}{\sqrt{d_v}}$, and we add self-loops \{u, u\} and \{v, v\} otherwise.

2. Let $\mathcal{L}_{G_\infty}$ be the normalized Laplacian of $G_\infty$.

3. Then, $\mathcal{L}_G(x) = \mathcal{L}_{G_\infty}x$. 
EIGENVALUES AND EIGENVECTORS

Normalized Laplacian $\mathcal{L}_G$ has eigenvalue 0 and associated eigenvector $\mu_G$.

What about other?

1. Since $\mathcal{L}_G$ is nonlinear Markov operator the number of eigenvalues and eigenvectors are not known.

2. Calculating eigenvalues of nonlinear Markov operator is NP-hard in general.
EIGENVALUES AND EIGENVECTORS

Theorem 4.4 Given a digraph $G$, for every subspace $U \leq \mathbb{R}^V$ of positive dimension, the operator $\Pi_U \mathcal{L}_G$ has an eigenvector, i.e., there exists a non-zero vector $x \in U$ and $\gamma \in \mathbb{R}_+$ such that

$$\Pi_U \mathcal{L}_G(x) = \gamma x \quad \text{and} \quad \gamma = \mathcal{R}_U(x),$$

where $\mathcal{R}_U$ is the Rayleigh quotient of the Markov operator $\Pi_U \mathcal{L}_G$.

They define second eigenvalue $\lambda_G$ of $\mathcal{L}_G$ as the smallest eigenvalue of $\Pi_{\mu_G} \mathcal{L}_G$. 
They show the following:

**Theorem 4.11** Let \( G = (V, E) \) be a digraph. Then, we have

\[
\frac{\lambda_G}{2} \leq \phi_G \leq 2\sqrt{\lambda_G}.
\]

This is more natural extension of cheeger’s inequality for undirected graphs than Chung’s method.
Algorithm 1 Partitioning using the heat equation

Input: A digraph \( G = (V, E) \), \( \Delta t \in \mathbb{R}_+ \), \( T \in \mathbb{R}_+ \)

Output: A set \( S \).

1. \( x_0 \leftarrow \) an arbitrary vector with \( x_0 \in \mu^\perp \).
2. while \( t \leq T \) do
   3. \( x = x - \Pi_{\mu_G^\perp} \mathcal{L}_G(x) \Delta t \).
   4. \( t = t + \Delta t \).
5. \( S \leftarrow \) the set obtained by applying Lemma 4.14 to \( x \).
6. return \( S \).
SPECTRAL ORDERING

The second eigenvector of normalized laplacian is minimizer of

\[
\sum_{(u,v) \in E} \left( [x_u - x_v]^+ \right)^2.
\]

Where \( \sum x_u^2 d_u = 1 \), and \([a]^+ = \max\{a, 0\}\).
Running Time for Algorithm 1:

<table>
<thead>
<tr>
<th>Name</th>
<th>n</th>
<th>m</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Narragan [30]</td>
<td>30</td>
<td>139</td>
<td>0.16 s</td>
</tr>
<tr>
<td>s208_st [29]</td>
<td>122</td>
<td>189</td>
<td>0.04 s</td>
</tr>
<tr>
<td>C. elegans [35]</td>
<td>239</td>
<td>1918</td>
<td>1.04 s</td>
</tr>
<tr>
<td>Political Blogs [1]</td>
<td>793</td>
<td>15839</td>
<td>2.83 s</td>
</tr>
<tr>
<td>US Airports [23]</td>
<td>1402</td>
<td>28032</td>
<td>7.40 s</td>
</tr>
<tr>
<td>Enron [23]</td>
<td>9164</td>
<td>761911</td>
<td>65.1 s</td>
</tr>
<tr>
<td>Epinions1 [23]</td>
<td>32223</td>
<td>443506</td>
<td>71.4 s</td>
</tr>
</tbody>
</table>
EXPERIMENTS

Figure 3: Convergence of $R_G$
SPECTRAL EMBEDDING

Figure 5: Embedding based on the eigenvectors
SPECTRAL ORDERING

(a) C. elegans (Nonlinear)  
(d) C. elegans (Chung)
CONDUCTANCE

(a) C. elegans
RARE CATEGORY DETECTION ON TIME EVOLVING GRAPHS

Rare category detection: Find minority classes (rare category) in big data by requesting minimum number of labels from the oracle.

For static graph: RACH, MUVIR, GRADE and so on

This paper is extension of GRADE.
1. Compute pair-wise similarity matrix (Adjacency matrix for graph data)
2. Calculate normalized matrix $W$, \[ W = D^{-1/2}W'D^{-1/2} \]
3. Calculate global similarity matrix $A$ by applying random walk with restart \[ A = (I_{n \times n} - \omega W)^{-1} \]
4. Identify rare classes by querying oracle for nodes (data points) near the boundaries

Intuition is that changes in $A$ becomes sharp at the boundary of minority classes.
DYNAMIC RARE CATEGORY DETECTION

Instead of performing GRADE at each step, make incremental changes to A and neighborhoods of nodes

Assumptions

1) Number of examples is fixed
2) Dataset in imbalanced
3) Minority classes are not separable from Majority classes
SINGLE UPDATE

If only one edge (self-loop) is added at time step $t$:

$$A^{(t)} = A^{(t-1)} + \alpha \frac{A^{(t-1)} u v^T A^{(t-1)}}{I + v^T A^{(t-1)} u}$$

Where $u = D(:,a)^{-1/2}$ and $v = \Delta S^{(t)}(a,b) D(:,b)^{-1/2}$.

**Theorem 2.** Suppose there is only one self loop edge $(a,a)$ being updated at time step $t$. If it satisfies the condition that

$$\frac{\alpha}{I + v^T A^{(t-1)} u} \leq \frac{\delta^{(t-1)}_i}{A^{(t-1)}_{i,a} \phi_a},$$

the first $K^{(t)}$ elements in $NN^{(t)}(i,:)$ are the same as $NN^{(t-1)}(i,:)$.
Algorithm 1 SIRD Algorithm

Input: \( M^{(1)}, A^{(1)}, \Delta S^{(2)}, \ldots, \Delta S^{(T)}, p_c^{(1)}, \alpha \).
Output: The set \( I \) of labeled examples

1: Construct the \( n \times n \) diagonal matrix \( D \), where \( D_{ii} = \sum_{j=1}^{n} S^{(1)}(i, j) \), \( i = 1, \ldots, n \).
2: Sort row \( i \) of \( A^{(1)} \) and saved into \( NN^{(1)}(i, :) \), where \( i = 1, \ldots, n \).
3: for \( t = 2 : T \) do
4: \hspace{1em} Let \( K^{(t)} = \max_{C=2}^{C=n} n \times p_c^{(t)} \).
5: \hspace{1em} Let column vector \( u = D(:, a)^{-1/2} \) and column vector \( v = \Delta S^{(t)}(a, a) D(:, a)^{-1/2} \), where \( \Delta S^{(t)}(a, a) \) is the non-zeros element in \( \Delta S^{(t)} \).
6: \hspace{1em} Update the global similarity matrix as follows:
\[
A^{(t)} = A^{(t-1)} + \alpha \frac{A^{(t-1)}uv^T A^{(t-1)}}{I + v^T A^{(t-1)} u}
\]
7: \hspace{1em} for \( i = 1 : n \) do
8: \hspace{2em} Based on Theorem 2, identify whether the first \( K^{(t)} \) elements of \( NN^{(t)}(i, :) \) is changed. If true, update the first \( K^{(t)} \) element in \( NN^{(t)}(i, :) \); otherwise, let \( NN^{(t)}(i, :) = NN^{(t-1)}(i, :) \).
7: \hspace{1em} end for
9: end for
10: for \( c = 2 : C \) do
11: \hspace{1em} Let \( k_c = n \times p_c^{(T)} \).
12: \hspace{1em} Find the first \( k_c \) element in each row of \( NN^{(T)} \). Set \( a^c \) to be the largest value of them.
13: \hspace{1em} Let \( KNN^c(x_i, a^c) = \{ x | NN^{(T)}(i, j) > a^c \} \), and \( n_i^c = |KNN^c| \), where \( i = 1, \ldots, n \) and \( j = 1, \ldots, n \).
14: \hspace{1em} for index = 1 : n do
15: \hspace{2em} For each nodes \( x_i \) has been labeled \( y_i \), if \( A^{(T)} > a^{y_i} \), \( score_j = -\infty \); else, let \( score_j = \frac{\max_{x \in KNN^c} (n_i^c - n_j^c)}{\max_{x \in KNN^c} n_i^c} \).
16: \hspace{1em} Select the examples \( x \) with largest score to oracle.
17: \hspace{2em} If the label of \( x \) is exact class \( c \), break; else, mark the class that \( x \) belongs to as discovered.
The first $K^{(t)}$ elements in $NN^{(t)}(i,:)$ are the same as $NN^{(t-1)}(i,:)$, if it satisfies the condition that

$$\frac{\alpha}{I + V^T A^{(t-1)} U} \leq \min_{i=1,\ldots,m} \{T_i\}$$

where $T_i = \min\{\frac{\delta^{(t-1)}_{i,a_i}}{A_{i,a_i}^{(t-1)} \phi_{b_i}^{a_i}}, \frac{\delta^{(t-1)}_{i}}{A_{i,b_i}^{(t-1)} \phi_{a_i}}\}$. 
QUERY DYNAMICS

$$T_{opt} = \max_{t=1,\ldots,T} \frac{I^{(1)} - I^{(t)}}{I^{(1)} - I^{(T)}} \cdot (RS^{(1)} - RS^{(T)}) - C \cdot t$$

1) allocate all budgets at the first time step
2) allocate all budgets at the last time step
3) Allocate all budget at time $T_{opt}$
4) Allocate query budget evenly
5) Allocate query budget following exponential distribution
# EXPERIMENTS

<table>
<thead>
<tr>
<th>Name</th>
<th>n</th>
<th>d</th>
<th>m</th>
<th>Largest-Class</th>
<th>Smallest-Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abalone</td>
<td>4177</td>
<td>8</td>
<td>5</td>
<td>56.93%</td>
<td>0.41%</td>
</tr>
<tr>
<td>Adult</td>
<td>48842</td>
<td>14</td>
<td>2</td>
<td>1.30%</td>
<td>98.70%</td>
</tr>
<tr>
<td>Statlog</td>
<td>58000</td>
<td>9</td>
<td>6</td>
<td>79.16%</td>
<td>0.04%</td>
</tr>
</tbody>
</table>

**TABLE I: Real Datasets**
EXPERIMENTS

(a) Synthetic Data  
(b) Abalone  
(c) Adult  
(d) Statlog
EXPERIMENTS

Fig. 3: Efficiency
EXPERIMENTS

Fig. 4: Query Locating

(a) Abalone
(b) Adult
(c) Statlog

(a) Synthetic Dataset
(b) Real Dataset (Adult)