Node label and Link prediction in complex networks

Miguel Romero
October 11, 2019

Pontificia Universidad Javeriana de Cali
Seminario Permanente de la Facultad de Ingeniería y Ciencias
Overview

1. Node label prediction:
   In-Silico Gene Annotation Prediction using the Co-expression Network Structure

2. Link prediction:
   Spectral Evolution of Twitter Mention Networks
Node label prediction:
In-Silico Gene Annotation
Prediction using the Co-expression Network Structure
Objective

In-silico prediction of functional gene annotations.

How?

- gene co-expression network
- existing knowledge body of gene annotations of a given genome
- supervised machine learning model
Co-expression networks are generally, represented as undirected, weighted graphs built from empirical data (expression profiles). Vertices denote genes and edges indicate a weighted relationship about their co-expression.

**Definition 1**

Let $V$ a set of genes, $E$ a set of edges that connect pairs of genes and $w$ a weight function. A (weighted) gene co-expression network is a weighted graph $G = (V, E, w : E \rightarrow \mathbb{R}_{\geq 0})$. 
Gene annotation

The goal of gene annotation is to determine the structural organization of a genome and discover sets of gene functions, i.e., the locations of genes and coding regions in a genome that determine what genes do.
Gene annotation

Gene annotations are classified in

- molecular function: molecular activities of individual gene products,
- cellular components: location of the active gene products,
- **biological processes**: pathways to which a gene contributes.
Definition 2
Let $A$ be a set of biological functions. An *annotated gene co-expression network* is a gene co-expression network $G = (V, E, w)$ complemented with an annotation function $\phi : V \rightarrow 2^A$. 
Topological Properties

Given $G = (V, E, w)$, properties of its network structure are computed for each gene $u \in V$:

- degree,
- eccentricity,
- clustering coefficient,
- closeness centrality,
- betweenness centrality,
- neighborhood connectivity,
- topological coefficient.
Network-based approach

Given an annotated co-expression network $G = (V, E, w)$ with annotation function $\phi$, the goal is to use the information represented by $\phi$ together with topological properties of $G$ to obtain a function $\psi : S \mapsto 2^A$.

Function $\psi$ predicts associations between annotations and genes based on a supervised machine learning technique.
The gene co-expression network $G = (V, E, w)$ comprises 19,665 vertices (genes) and 553,125 edges.

The dataset summarizes data for the 19,665 genes, 615 annotations, and 7 topological measures. It comprises 19,665 rows and 222 columns.
The dataset is heavily imbalanced!
Supervised training

Synthetic Minority Over-sampling TEchnique (SMOTE) is used to over-sample the minority class to potentially improve the performance of a classifier without loss of data.

A supervised machine learning technique for the annotation prediction is used. In particular, the XGBoost implementation of gradient boosted decision trees is used.
Two models are trained for predicting gene annotations, one per biological function (141 annotations). Namely, one in which the topological measures of G are used and another one in which they are not.
Annotation prediction

![Graphs showing accuracy, F1-score, and AUC ROC over annotations. The graphs compare 'Only Annotations' and 'Topological measures'.]
<table>
<thead>
<tr>
<th>ID</th>
<th>Biological process</th>
<th># Genes</th>
<th>Max FP</th>
<th># FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>0006807</td>
<td>nitrogen compound metabolic process</td>
<td>15</td>
<td>41</td>
<td>1</td>
</tr>
<tr>
<td>0006289</td>
<td>nucleotide-excision repair</td>
<td>20</td>
<td>46</td>
<td>1</td>
</tr>
<tr>
<td>0006397</td>
<td>mRNA processing</td>
<td>17</td>
<td>48</td>
<td>1</td>
</tr>
<tr>
<td>0007017</td>
<td>microtubule-based process</td>
<td>18</td>
<td>49</td>
<td>1</td>
</tr>
<tr>
<td>0070588</td>
<td>calcium ion transmembrane transport</td>
<td>10</td>
<td>36</td>
<td>1</td>
</tr>
<tr>
<td>0006184</td>
<td>GTP catabolic process</td>
<td>49</td>
<td>47</td>
<td>1</td>
</tr>
<tr>
<td>0044267</td>
<td>cellular protein metabolic process</td>
<td>25</td>
<td>49</td>
<td>1</td>
</tr>
<tr>
<td>0007186</td>
<td>G-protein coupled receptor protein signaling ...</td>
<td>11</td>
<td>50</td>
<td>1</td>
</tr>
<tr>
<td>0006281</td>
<td>DNA repair</td>
<td>62</td>
<td>50</td>
<td>2</td>
</tr>
<tr>
<td>0006754</td>
<td>ATP biosynthetic process</td>
<td>24</td>
<td>49</td>
<td>3</td>
</tr>
<tr>
<td>0006904</td>
<td>vesicle docking involved in exocytosis</td>
<td>11</td>
<td>50</td>
<td>4</td>
</tr>
<tr>
<td>0055114</td>
<td>oxidation-reduction process</td>
<td>870</td>
<td>47</td>
<td>5</td>
</tr>
</tbody>
</table>
Link prediction: Spectral Evolution of Twitter Mention Networks
Evaluate various link prediction methods that underlie the spectral evolution model.

Applies the spectral evolution model to networks of mentions between individuals who used trending political hashtags in Twitter between August 2017 and August 2018.
The dataset consists of 31 mention networks between Twitter users who defined their profile location as Colombia. These networks capture conversations around a set of hashtags related to popular political topics between August 2017 and August 2018.
Mention networks

Users are represented by the set of vertices $V$ and the set of edges is denoted by $E$. There exists an edge $\{i, j\} \in V \times V$ between users $i$ and $j$, if user $i$ uses a political hashtag (e.g., #eleccionesseguras) and mentions user $j$ (via @username).

A mention network $G = (V, E)$ is represented as a weighted multi-graph without self-loops. Our analysis is based on the largest connected component of the multi-graph, denoted by $G_c = (V_c, E_c)$.
## Mention networks

| Hashtag                               | $|V_c|$ | $|E_c|$ |
|---------------------------------------|-----|-----|
| 0 abortionlegalya                     | 1282| 1538|
| 1 alianzasporlaseguridad              | 150 | 351 |
| 2 asiconstruimospaz                   | 2405| 6950|
| 3 colombialibredefracking             | 1476| 3127|
| 4 colombialibredeminas                | 655 | 1421|
| 5 dialogosmetropolitanos              | 932 | 4134|
| 6 edutransforma                       | 161 | 404 |
| 7 eleccionesseguras                   | 2634| 7969|
| 8 elquedigauribe                      | 2052| 5272|
| 9 frutosdelapaz                       | 1479| 3468|
Spectral evolution model

Let $A$ denote the adjacency matrix of $G_c$. Furthermore, let $A = U \Lambda U^T$ denote the eigen decomposition of $A$, where $\Lambda$ represents the spectrum of $G_c$.

The spectral evolution model characterizes the dynamics of $G_c$ (i.e., how new edges are created over time) in terms of the evolution of the spectrum of the network, assuming that its eigenvectors in $U$ remain unchanged.
If this condition is satisfied, estimating the **formation of new edges** can be expressed as **transformations of the spectrum** through the application of real functions (using graph kernels) or extrapolation methods (using learning algorithms that estimate the spectrum trajectories).
To apply the spectral evolution model, we need to verify the assumption on the evolution of the spectrum and eigenvectors. Every network $G_c$ has a timestamp associated to each edge, representing the time at which the edge is created.

- spectral evolution (eigenvalues),
- eigenvector evolution,
- eigenvector stability, and
- spectral diagonality test
Spectral evolution model verification

(a) Spectral evolution (eigenvalues)

(b) Eigenvector evolution
Spectral evolution model verification

(c) Eigenvector stability (eigenvalues)  
(d) Spectral diagonality test
Let $K(A)$ be a kernel of an adjacency matrix $A$, whose eigen decomposition is $A = U\Lambda U^T$. **Graph kernels** assume that there exists a real function $f(\lambda)$ that describes the growth of the spectrum.

In particular, $K(A)$ can be written as $K(A) = UF(\Lambda)U^T$, for some functions $F(\Lambda)$ that applies a real function $f(\lambda)$ to the eigenvalues of $A$. 
In particular, we use graph kernels of the triangle closing, exponential and Neumann growth.

<table>
<thead>
<tr>
<th>Kernel</th>
<th>$K(A)$</th>
<th>$f(\lambda)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Triangle closing</td>
<td>$A^2$</td>
<td>$f(\lambda) = \lambda^2$</td>
</tr>
<tr>
<td>Exponential</td>
<td>$exp(\alpha A)$</td>
<td>$f(\lambda) = e^{\alpha \lambda}$</td>
</tr>
<tr>
<td>Neumann</td>
<td>$(I - \alpha A)^{-1}$</td>
<td>$f(\lambda) = \frac{1}{1 - \alpha \lambda}$</td>
</tr>
</tbody>
</table>
When the evolution of the spectrum is irregular it is not possible to find a simple function that describe network growth. This model extrapolates each eigenvalue of the network, assuming that the network to be analyzed follows the spectral evolution model.
Structural similarity (SSIM) is a method for measuring similarity between two images based on the idea that the pixels have strong inter-dependencies especially when they are spatially close.

Adjacency matrices $A$ and $\hat{A}_c$ are assumed to be images, and the edges $A_{ij}$ and $\hat{A}_{c,ij}$ represent pixels.
The extrapolation method tends to outperform the other methods based on the performance metrics. Specifically, for 28 out of 31 networks (91% of the total), the extrapolation method provides distinct, if slight, improvement.

The outperformance of the spectral extrapolation method seems to be explained by the method being able to consider the irregular evolution of the eigenvalues.
Questions?
Thanks!