Scalable tensor methods for multi-relational learning across graphs

Zhuliu Li

Department of Computer Science and Engineering
Outline

• Task description: multi-relational learning among heterogeneous graphs
  • Examples
  • Formal definition
• Review: label propagation and tensor product graph
  • Label propagation algorithm
  • Tensor product graph
• Scalable tensor methods for multi-relational learning
  • Low-rank label propagation algorithm on tensor product graph
  • Sparse label propagation algorithm on tensor product graph
  • Tensor decomposition with graph constraint
• Experiments: Aligning multiple PPI networks, CT scans and DBLP data
Task description: examples

**Enzyme data**
- Blue edges: within graph interactions
- Red edges: cross graph interactions

“Chemical compound (drug) A is targeting on protein B.” (Compound: A, Protein: B)

“John publish a reinforcement learning paper at ICML.” (Author: John, Paper: RL, Venue: ICML)

**DBLP data**

Liu, H., & Yang, Y. (ICML, 2016)
Task description: examples

• Aligning protein-protein interaction networks across species.
• Nodes are proteins
• Edges connect interacting proteins.
• The relations are evolutionary relations.
Task description: examples

- Align the same organ (color) across the CT scans of human body
Task description: definition

• Given:
  • Individual graph $W^{(i)} \in \mathbb{R}^{I_i \times I_i}, i = 1, \ldots n$.
  • A tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_n}$ storing (parts of) the initial scores of the multi-relations (tuples) between the nodes from different graphs.

• Task:
  • Predict the scores of the unscored tuples.
  • Correct the scores of the scored tuples based on the graph structures.
Task description: definition
Background: label propagation

\[ y^{t+1} = \alpha S y^t + (1 - \alpha) y^0 \]

\[ S = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \]

Zhou, Denny, et al. (NIPS, 2004)
Background: label propagation

- Objective function

\[ J(y) = \sum_{i,j} w_{ij} \left( \frac{y_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}} \right)^2 + \mu \| y - y^0 \|^2 \]

\[ = y^T Ly + \mu \| y - y^0 \|^2, \]

where \( \mu = \frac{1-\alpha}{\alpha} \), \( L = I - S \).

- Closed-form solution

\[ y^* = (1 - \alpha)(I - \alpha S)^{-1} y^0 \]
PageRank (random walk with restart)

- Define a Markov Chain to represent the web surfing jumping probability
- Start with a random page and take random walks.
- The stationary distribution of the Markov Chain gives the probability of stopping at a particular page (a good rank!)
- Label propagation is a very similar algorithm.

\[
R = \begin{bmatrix}
\frac{(1-d)}{N} & \ell(p_1, p_1) & \ell(p_1, p_2) & \cdots & \ell(p_1, p_N) \\
\frac{(1-d)}{N} & \ell(p_2, p_1) & \cdots & \ell(p_2, p_N) \\
\vdots & \vdots & \ddots & \vdots & \ell(p_j, p_i) \\
\frac{(1-d)}{N} & \ell(p_N, p_1) & \cdots & \ell(p_N, p_N)
\end{bmatrix} + d \begin{bmatrix}
\ell(p_1, p_1) & \ell(p_1, p_2) & \cdots & \ell(p_1, p_N) \\
\ell(p_2, p_1) & \cdots & \ell(p_2, p_N) \\
\vdots & \ddots & \ell(p_j, p_i) & \ell(p_j, p_N) \\
\ell(p_N, p_1) & \cdots & \ell(p_N, p_N)
\end{bmatrix}
\]
Background: tensor product graph (TPG)

\[ W = W^{(1)} \otimes W^{(2)} \otimes W^{(3)} \in \mathbb{R}^{I_1 I_2 I_3 \times I_1 I_2 I_3} \]

\[ W_{(1,I,A),(2,II,B)} = W^{(1)}_{1,2} W^{(2)}_{I,II} W^{(3)}_{A,B} \]
Revisit the learning task

• Given:
  • Individual graph $W^{(i)} \in \mathbb{R}^{I_i \times I_i}$, $i = 1, \ldots, n$.
  • A tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_n}$ storing (parts of) the initial scores of the multi-relations (tuple) between the nodes from different graphs.

• Task:
  • Predict the scores of the unscored tuples.
  • Correct the scores of the scored tuples based on the graph structures.
Revisit the learning task

\[ \mathcal{P}(W^{(1)}, W^{(2)}, W^{(3)}) = \begin{array}{c}
\mathcal{Y} \\
\in \mathbb{R}^{I_1I_2I_3 \times 1}
\end{array} \]

- Task:
  - Predict the scores of the unscored tuples =
  - Predict the labels of the unlabeled nodes of \( W \).

- Correct the scores of the scored tuples based on the graph structures =
- Correct the labels of the labeled nodes of \( W \) based on the graph structures
Revisit the learning task

\[
\tilde{Y}^{t+1} = \alpha (S^{(1)} \otimes S^{(2)}) \tilde{Y}^t + (1 - \alpha) \tilde{Y}^0
\]

\[
Y^{t+1} = \alpha S^{(2)} Y^t S^{(1)} + (1 - \alpha) Y^0
\]
A simple simulation
A simple simulation
Scalability issue

- Iterative approach:

\[ \mathbf{Y}^{t+1} = \alpha (\otimes_{i=1}^{n} S^{(i)}) \mathbf{Y}^{t} + (1 - \alpha) \mathbf{Y}^{0} \]

\[ \mathbf{Y}^{t+1} = \alpha \mathbf{Y}^{t} \times_{1} S^{(n)} \times_{2} S^{(n-1)} \cdots \times_{n} S^{(1)} + (1 - \alpha) \mathbf{Y}^{0} \]

- Time complexity of one iteration \( O((\prod_{i=1}^{n} I_{i})(\sum_{i=1}^{n} I_{i})) \)

- Space complexity \( O(\prod_{i=1}^{n} I_{i}) \)

- Closed form solution:

\[ \mathbf{Y}^{*} = (1 - \alpha) (I - \alpha S)^{-1} \mathbf{Y}^{0} \]

\[ = (1 - \alpha) (\otimes_{i=1}^{n} Q^{(i)})(I - \alpha (\otimes_{i=1}^{n} \Lambda^{(i)}))^{-1} (\otimes_{i=1}^{n} Q^{(i)T}) \mathbf{Y}^{0} \]
Proposition 1: Low rank approximation of TPG

- Idea:

\[
\begin{align*}
\text{minimize } & \quad \| (I - \alpha S)^{-1} - (I - \alpha S_k)^{-1} \|_{2,F} \\
\text{subject to } & \quad \text{rank}(S_k) = k
\end{align*}
\]

**Lemma.** Let \( \lambda_1, \ldots, \lambda_n \) be eigenvalues of \( A \) with corresponding eigenvectors \( x_1, \ldots, x_n \), and let \( \mu_1, \ldots, \mu_m \) be eigenvalues of \( B \) with corresponding eigenvectors \( y_1, \ldots, y_m \). Then the eigenvalues and eigenvectors of \( A \otimes B \) are \( \lambda_i \mu_j \) and \( x_i \otimes y_j \), \( i = 1, \ldots, n \), \( j = 1, \ldots, m \).
Proposition 1: Low rank approximation of TPG

- Time complexity: $O(mnk)$
- Space complexity: $O(\max(k \sum_{i=1}^{n} I_i, m))$

\[
\overrightarrow{y}^* = (1 - \alpha)(Q_{select}^{(1)} \circ \cdots \circ Q_{select}^{(n)})M(Q_{select}^{(1)} \circ \cdots \circ Q_{select}^{(n)})^T\overrightarrow{y}^0 + (1 - \alpha)\overrightarrow{y}^0
\]
Experiment 1 – CT Scan Image Alignment

• 134 CT Scan images of size 512 x 512.
• Each image was segmented into regions (features).
• Sampled spots are aligned across the images.
Proposition 2: keep the tensor sparse

- Tensor becomes dense after every propagation step
- Adding L1-norm regularizer to keep the sparsity
  \[ J_{L1}(\vec{Y}) = J(\vec{Y}) + \beta \sum_{i=1}^{N} |\vec{Y}_i|. \]
- Apply FISTA (fast iterative shrinkage-thresholding algorithm)
  \[ \vec{Y}^{t+1} = \frac{1}{L_J} (\vec{Y}^t \times_1 S^{(n)} \times_2 S^{(n-1)} \cdots \times_n S^{(1)}) + (1 - \frac{1 + \mu}{L_J}) \vec{Y}^t + \frac{\mu}{L_J} \vec{Y}^0. \]
  where the step size \( L_J = 1 + \mu - \min(\otimes_{i=1}^{n} [\lambda_{\min}(S^{(i)}), \lambda_{\max}(S^{(i)})]) \)
- Use METTM (Memory-Efficient Tensor Times Matrix) for matrix-tensor multiplication
Experiment 2 – PPI Network Alignment

- PPI subnetworks of four species Human (HSA), Mouse (MMU), Fly (DME) and Yeast (SCE).
- Four pathways (subnetworks) are tested.

<table>
<thead>
<tr>
<th>Pathway</th>
<th>DME</th>
<th>HSA</th>
<th>MMU</th>
<th>SCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>00020</td>
<td>43/332</td>
<td>30/257</td>
<td>32/312</td>
<td>32/265</td>
</tr>
<tr>
<td>00190</td>
<td>144/208</td>
<td>133/292</td>
<td>134/315</td>
<td>72/284</td>
</tr>
<tr>
<td>00564</td>
<td>63/151</td>
<td>95/318</td>
<td>94/187</td>
<td>18/192</td>
</tr>
<tr>
<td>04320</td>
<td>28/226</td>
<td>28/300</td>
<td>26/228</td>
<td>-</td>
</tr>
</tbody>
</table>

- **Validation:**
  Checking the agreement of the alignment scores and the protein functional similarities returned by Gene Ontology
  
  *(Consortium et al., 2015)*
Experiment 2 – PPI Network Alignment
Experiment 2 – PPI Network Alignment
Future work: tensor decomposition

• Revisit objective function

\[ J(y) = \sum_{i,j} w_{ij} \left( \frac{y_i}{\sqrt{d_{ii}}} - \frac{y_j}{\sqrt{d_{jj}}} \right)^2 + \mu \|y - y^0\|^2 \]

\[ = y^T Ly + \mu \|y - y^0\|^2 , \]

where \( \mu = \frac{1-\alpha}{\alpha} \), \( L = I - S \).
Future work: tensor decomposition

• CPD form assumption

\[ J(A, B, C) = \text{vec}([A, B, C])^T L \text{vec}([A, B, C]) + \mu \|[A, B, C] - Y^0\|^2 \]
\[ = 1^T (C \odot B \odot A)^T L (C \odot B \odot A) 1 + \mu \|Y^0 - [A, B, C]\|^2 \]

\[ \frac{\partial J}{\partial A} = 2(AM_1 - \mu S^{(1)} AM_2 - (1 - \mu)M_3) \]

\[ M_1 = C^T C \ast B^T B, \quad M_2 = C^T S^{(3)} C \ast B^T S^{(2)} B \text{ and } M_3 = Y^{(0)}_{(1)} (C \odot B). \]
Experiment 3 – DBLP dataset

- 13823 Authors, 11372 papers and 10167 venues.
- 12066 tuples.
- Experiment setting: randomly sampling 0.1%, 1%, 5%, 10%, 50% and 90% of tuples to be the training data. The rest are testing data.

Liu, H., & Yang, Y. (ICML, 2016)


