Highly Scalable Inference Techniques for Mix-Membership Block Models

Eric Xing
epxing@cs.cmu.edu
Machine Learning Dept./Language Technology Inst./Computer Science Dept.
Carnegie Mellon University
Social networks
Estimating Evolving “Latent” Social Networks

Can I get his vote?

Corporativity, Antagonism, Cliques, ...

over time?
Asymptotic-consistent graph estimation algorithms [Kolar and Xing, 09,12]

**KELLER**

- Smooth Change
- Kernel Reweighting

**TESLA**

- Abrupt Change
- Structure Variation: $\Delta_t = |\beta_{t+1} - \beta_t|

**Smoothly evolving graphs**

**Abruptly evolving graphs**

\[
\mathbb{P} \left[ \hat{G}(\lambda_n) \neq G \right] = \mathcal{O} \left( \exp \left( -Cn^\epsilon \right) \right) \to 0
\]
Mixed Membership of Actors

- **Micro-inference vs. Meso- or Macro-inference**
- Multi-role of every node
- Context dependent role-instantiation
- Role dynamics
Mixed Membership Stochastic Blockmodel [Airoldi, Blei, Fienberg and Xing, JMLR 2008]

1. \( \{\theta_i\}_{i=1}^N \sim p(\theta|\alpha) \equiv \text{Dirichlet}(\theta; \alpha) \)
   sample mixed membership vectors.

2. For each actor \( v_j \) that actor \( v_i \) possibly interacts with:
   - \( z_{i\rightarrow j} \sim \text{Multinomial}(z|\theta_i) \)
     sample an indicator for \( v_i \);
   - \( z_{j\leftarrow i} \sim \text{Multinomial}(z|\theta_j) \)
     sample an indicator for \( v_j \);
   - \( e_{ij} \sim \text{Bernoulli}(e|z_{i\rightarrow j}^T B z_{i\leftarrow j}) \)
     sample a link.
In the mixed-membership simplex

[Airoldi, Blei, Fienberg and Xing, JMLR 2008]
Multi-Scale Community Blockmodel

[Ho, Parikh, and Xing, JASA 2012]
Challenge – Massive Data Scale

Does not fitting into memory, nor a single machine, a familiar problem!
Popular Statistical Network Models don’t scale well

- Mixed-Membership Stochastic Blockmodel
  - Models every element $A(i,j)$ of the adjacency matrix, which has size $\Theta(N^2)$
  - Hence $\Theta(N^2)$ latent variables
  - Hence $\Omega(N^2)$ time per iteration of approximate inference

- Latent Factor models
  - Only $\Theta(N)$ latent variables, but Markov Blanket of each variable is $\Theta(N)$ in size
  - Thus $\Omega(N^2)$ time per iteration of approximate inference

- Exponential Random Graph Models
  - Estimated via MCMC-MLE, which samples the adjacency matrix
  - So $\Omega(N^2)$ time for approximate inference

- Fundamental problem: the above models all represent the network by its adjacency matrix, i.e. the matrix of all relationships $A(i,j)$
  - Adjacency matrix has size $\Theta(N^2)$, so inference will take $\Theta(N^2)$ time as well!
  - The more compact adjacency list representation is NOT a solution, because the above models statistically depend on “missing edges” $A(i,j) = 0$ as well
ML with scalable, adaptive, online, parallelizable, and confident ...

**Representation:** simpler, but as informative

**Model:** Generic building blocks: loss functions, structures, constraints, priors ..

**Algorithm:** sequential, batch → parallel, stochastic

**System:** multi-core, distributed file system, shared memory, cloud

**Theory:** linear or sub-linear convergence rate, sample complexity, etc
Scalable Representation

[Ho and Xing, NIPS 2013]

### Node 0

#### Adj. Matrix Features

<table>
<thead>
<tr>
<th>Dest. Node</th>
<th>Edge Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>No</td>
</tr>
<tr>
<td>7</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Length O(N) vs. Length O(Degree²)**

- Four edges, five non-edges

#### 2/3-Triangle Features

<table>
<thead>
<tr>
<th>Dest. Nodes</th>
<th>Triangle Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2,3)</td>
<td></td>
</tr>
<tr>
<td>(2,4)</td>
<td></td>
</tr>
<tr>
<td>(2,9)</td>
<td></td>
</tr>
<tr>
<td>(3,4)</td>
<td></td>
</tr>
<tr>
<td>(3,9)</td>
<td></td>
</tr>
<tr>
<td>(4,9)</td>
<td></td>
</tr>
</tbody>
</table>

One 3-triangle, five 2-triangles

Triangle features more compact for low node degree!
Why 2/3-edge triangular motifs?

- Well studied in many fields:
  - Biology
  - Social science (transitivity)
  - Data mining (clustering coefficients)

- Basis for network clustering coefficient (CC)
  - Ratio of 2-edge motifs to 2-edge + 3-edge motifs
  - High CC implies stronger, more well-connected clusters

- 2/3-edge motifs contain almost all edges from the adjacency matrix
  - Exception: isolated components with exactly 1 edge
  - Thus, the triangular representation preserves almost all network information!
Triangular Model Intuition

- Adjacency matrix models (MMSB, Latent Factor) are concerned with **edge probabilities**
  - i.e. the distribution over events \{ A(i,j) = 0, A(i,j) = 1 \}

- Our triangular motif model is concerned with probabilities over 2/3-edge motifs
  - i.e. the probability of a triple \( (i,j,k) \) exhibiting one of the three possible 2-edge motifs, or the sole 3-edge motif
MMTM Generative Process

For each triple (i,j,k) being modeled:
1. Pick roles for i,j,k from their respective role vectors
For each triple \((i,j,k)\) being modeled:

1. Pick roles for \(i,j,k\) from their respective role vectors
2. Given the combination of roles (in this case 1,3,2), we look up a tensor of parameters \(B\) to get that role combination’s 2/3-edge motif distribution
3. Generate the motif from the distribution

Note: we permit adjacent node triples to generate “incompatible” 2/3-edge motifs. This is in line with the “bag-of-motifs” assumption!

Look up \(B(1,3,2)\) to get the 2/3-edge motif distribution

Motif:

Probability: 0.4 0.1 0.2 0.3
MMDM Graphical Model

\[ \theta_i \sim \text{Dirichlet}(\alpha) \]
\[ s_{i,jk} \sim \text{Multinomial}(\theta_i) \]
\[ B_{xyz} \sim \text{Dirichlet}(\lambda) \]
\[ E_{ijk} \sim \text{TriangleDistribution}(B,s_{i,jk},s_{j,ik},s_{k,ij}) \]

We use Rao-Blackwellized/Collapsed Gibbs Sampling for inference, with \( \theta \) and \( B \) integrated out.

Observed 2/3-edge triangular motifs
Tensor of motif distributions for each role combination
Additional modeling and scaling technologies

- Isomorphism

- $\delta$-subsampling
  - we pick a constant $\delta$ and subsample $\delta(\delta - 1)/2$ motifs from every node with degree $> \delta$
  - A possible theory of projection invariance

- $O(K)$ triangle probability parameters (instead of $O(K^3)$)

- Stochastic variation inference

- Parallel inference with parameter server architecture and bounded staleness
Simulations

- Statistics for N=4,000 simulation networks:

<table>
<thead>
<tr>
<th>Model</th>
<th>#0,1-edges</th>
<th>#1-edges</th>
<th>max(D₁)</th>
<th>#Δ₃, Δ₂</th>
<th>δ = 20</th>
<th>δ = 15</th>
<th>δ = 10</th>
<th>δ = 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMSB</td>
<td>7,998,000</td>
<td>55,696</td>
<td>51</td>
<td>1,541,085</td>
<td>749,018</td>
<td>418,764</td>
<td>179,841</td>
<td>39,996</td>
</tr>
<tr>
<td>Latent position</td>
<td>56,077</td>
<td></td>
<td>51</td>
<td>1,562,710</td>
<td>746,979</td>
<td>418,448</td>
<td>179,757</td>
<td>39,988</td>
</tr>
<tr>
<td>Biased scale-free</td>
<td>60,000</td>
<td></td>
<td>231</td>
<td>3,176,927</td>
<td>497,737</td>
<td>304,866</td>
<td>144,206</td>
<td>35,470</td>
</tr>
<tr>
<td>Pure membership</td>
<td>55,651</td>
<td></td>
<td>44</td>
<td>1,533,365</td>
<td>746,796</td>
<td>418,222</td>
<td>179,693</td>
<td>39,986</td>
</tr>
</tbody>
</table>

Table 2: Number of edges, maximum degree, and number of 3- and 2-edge triangles Δ₃, Δ₂ for each N = 4,000 synthetic network, as well as #triangles when subsampling at various degree thresholds δ. MMSB inference is linear in #0,1-edges, while our MMTM’s inference is linear in #Δ₃, Δ₂.
Simulations

- MMTM with $\delta$-subsampling is not only much faster, but also more accurate.
Improvement over state of the art

- As the number of nodes increases:
  - MMSB (edge-based representation) runtime increases quadratically
  - MMTM (triangle-based) runtime increases linearly when $\delta$ is held constant
  - Stochastic variational inference further improves speed

<table>
<thead>
<tr>
<th>Name</th>
<th>Nodes $N$</th>
<th>Edges</th>
<th>Roles $K$</th>
<th>Threads</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brightkite</td>
<td>58K</td>
<td>214K</td>
<td>64</td>
<td>4</td>
<td>35 min$^1$</td>
</tr>
<tr>
<td>Brightkite</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
<td>82K</td>
<td>504K</td>
<td>100</td>
<td>4</td>
<td>2.3 h</td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stanford Web</td>
<td>282K</td>
<td>2.0M</td>
<td>5</td>
<td>4</td>
<td>12 min$^2$</td>
</tr>
<tr>
<td>Stanford Web</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Berkeley- Stanford Web</td>
<td>685K</td>
<td>6.6M</td>
<td>100</td>
<td>8</td>
<td>20.7 h</td>
</tr>
<tr>
<td>Youtube</td>
<td>1.1M</td>
<td>3.0M</td>
<td>100</td>
<td>8</td>
<td>10.7 h</td>
</tr>
</tbody>
</table>

Competing methods

- 8 days (Blei, NIPS 2012)
- 18 hrs (Ho et al, NIPS 2012)
A Larger-scale Demonstration

- Stanford web graph, $N \approx 280,000$
  - Ran for 2,000 sampling iterations, convergence observed by 500 iterations
  - Total runtime: 74 hours on a single computational thread

Every circle represents a node in the network

Circle sizes are proportional to node degrees

Colors and positions represent inferred role MM vectors

Figure 5: $N = 281,903$ Stanford web graph, MMTM mixed-membership visualization.
MMTM Stochastic Variational

- Variational EM on randomly chosen triangles (data points)
  - Similar to stochastic variational for LDA and MMSB
  - Only need to touch every triangle 2-3 times to converge

- O(K) triangle probability parameters B
  - Old model:
    - B(a,b,c) for all $K^3$ choices of roles $a,b,c \rightarrow K^3$ parameters
  - New model:
    - 3-roles-same: $B(a,a,a)$ for each of the $K$ choices of $a \rightarrow K$ parameters
    - 2-roles-same: $B(a,a,\cdot)$ for each of the $K$ choices of $a$, where $\cdot \neq a \rightarrow K$ parameters
    - All-roles-different: $B(\cdot,\cdot,\cdot)$ where all three $\cdot$ are different $\rightarrow 1$ parameter
    - Total $2K + 1$ parameters

- Parallelization
  - Alternate inference between
    - Node topic vectors $\Theta$
    - Triangle role assignments $s$
    - Triangle probability parameters $B$
  - Each variable type can be parallelized given the other 2 types
Running time on real networks

<table>
<thead>
<tr>
<th>Name</th>
<th>Nodes $N$</th>
<th>Edges</th>
<th>$\delta$</th>
<th>2.3-Tris (for $\delta$)</th>
<th>Frac. of 3-Tris</th>
<th>Roles $K$</th>
<th>Threads</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brightkite</td>
<td>58K</td>
<td>214K</td>
<td>50</td>
<td>3.5M</td>
<td>0.11</td>
<td>64</td>
<td>4</td>
<td>35 min$^1$</td>
</tr>
<tr>
<td>Brightkite</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>4</td>
<td>2.7 h</td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
<td>82K</td>
<td>504K</td>
<td>50</td>
<td>9.0M</td>
<td>0.030</td>
<td>100</td>
<td>4</td>
<td>2.3 h</td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>4</td>
<td>6.8 h</td>
</tr>
<tr>
<td>Stanford Web</td>
<td>282K</td>
<td>2.0M</td>
<td>20</td>
<td>11.4M</td>
<td>0.57</td>
<td>5</td>
<td>4</td>
<td>12 min$^2$</td>
</tr>
<tr>
<td>Stanford Web</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100</td>
<td>4</td>
<td>6.3 h</td>
</tr>
<tr>
<td>Berkeley-Stanford Web</td>
<td>685K</td>
<td>6.6M</td>
<td>35</td>
<td>67.1M</td>
<td>0.55</td>
<td>100</td>
<td>8</td>
<td>20.7 h</td>
</tr>
<tr>
<td>Youtube</td>
<td>1.1M</td>
<td>3.0M</td>
<td>50</td>
<td>36.0M</td>
<td>0.053</td>
<td>100</td>
<td>8</td>
<td>10.7 h</td>
</tr>
</tbody>
</table>

Graphs/runtime are for 10 passes per data point (triangle)
Convergence occurs in 2-3 data passes
Youtube network (1.1M nodes, $K = 100$) in <2h using 8 threads
Further Improvement over state of the art

- As the number of nodes increases:
  - MMSB (edge-based representation) runtime increases **quadratically**
  - MMTM (triangle-based) runtime increases **linearly** when $\delta$ is held constant
  - Stochastic variational inference further improves speed

<table>
<thead>
<tr>
<th>Name</th>
<th>Nodes $N$</th>
<th>Edges</th>
<th>Roles $K$</th>
<th>Threads</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brightkite</td>
<td>58K</td>
<td>214K</td>
<td>64</td>
<td>4</td>
<td>35 min$^{-1}$</td>
</tr>
<tr>
<td>Brightkite</td>
<td>82K</td>
<td>504K</td>
<td>300</td>
<td>4</td>
<td>2.7 h</td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
<td>82K</td>
<td>504K</td>
<td>100</td>
<td>4</td>
<td>2.3 h</td>
</tr>
<tr>
<td>Stanford Web</td>
<td>282K</td>
<td>2.0M</td>
<td>5</td>
<td>4</td>
<td>6.8 h</td>
</tr>
<tr>
<td>Stanford Web</td>
<td>685K</td>
<td>6.6M</td>
<td>100</td>
<td>4</td>
<td>6.3 h</td>
</tr>
<tr>
<td>Berkeley-Stanford Web</td>
<td>1.1M</td>
<td>3.0M</td>
<td>100</td>
<td>8</td>
<td>20.7 h</td>
</tr>
<tr>
<td>Youtube</td>
<td>4.1M</td>
<td>11.4M</td>
<td>100</td>
<td>8</td>
<td>10.7 h</td>
</tr>
</tbody>
</table>

[Ho, Yin and Xing, NIPS 2012, UAI 2013]

Competing methods

- 8 days (Blei, NIPS 2012)
- 18 hrs (Ho et al, NIPS 2012)
And the improvement continues ...

<table>
<thead>
<tr>
<th>Real Networks — Statistics, Experimental Settings and Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
</tr>
<tr>
<td>------------------</td>
</tr>
<tr>
<td>Brightkite</td>
</tr>
<tr>
<td>Brightkite</td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
</tr>
<tr>
<td>Slashdot Feb 2009</td>
</tr>
<tr>
<td>Stanford Web</td>
</tr>
<tr>
<td>Stanford Web</td>
</tr>
<tr>
<td>Berkeley-Stanford Web</td>
</tr>
<tr>
<td>Youtube</td>
</tr>
</tbody>
</table>

Brightkite 58K nodes

Stochastic Variational MMSB (Gopalan et al, NIPS 2012) took 8 days using 4 threads

GGibbs MMTM (Ho et al, NIPS 2012) took 18.5 hours using 1 thread

New MMTM converges in 12 min

1000x speedup!

Stanford 282K nodes

New MMTM converges in 6 min

200x speedup!
Conclusion

- MMTM exploits a “bag-of-triangular-motifs” network representation
  - Specifically, MMTM models triangular motifs with 2 or 3 edges
  - Parsimonious alternative to edge-based adjacency matrix representation, which has size $N^2$

- MMTM scales to much larger networks than adjacency matrix models such as MMSB, ERGMs or latent position models
  - With $\delta$-subsampling, # 2/3-edge triangular motifs $\ll N^2$
  - 100K node networks are feasible with MMTM (single thread)
  - Whereas 10K node networks are already impractical for the a/m models

- MMTM inference yields better role MM vector recovery than MMSB inference on a variety of models
  - Even on the MMSB model itself!
  - This is partly because MMTM’s state space is much smaller (fewer latent variables), thus MMTM approximate inference converges much faster
A note on scalable ML

- Our New MMTM is built on 3 principles:
  - Compact **data representation** (triangles rather than edges)
  - Parsimonious **model** with linear $O(K)$ number of role parameters
  - Fast, scalable, distributable **inference algorithm** (stochastic variational EM)

- These principles are the building blocks for truly scalable Big ML

- Next step: distributed general ML inference engine for large clusters
Future Work

• Parallelization
  • One thread can perform inference on N=280K nodes in 3 days
  • We aim to parallelize to 1,000 threads, so as to perform inference on networks with N=100M nodes

• Subsampling strategies
  • What are the theoretical properties of $\delta$-subsampling?
  • Are there better subsampling strategies?