Re-identification in Dynamic Networks: Challenges and Opportunities

Shawndra Hill
University of Pennsylvania

Kick-off Meeting, July 28, 2008
Motivating example:

**Repetitive Subscription Fraud**

- Large telecommunications company
- Telecom service
- Long experience with fraud detection
- Sophisticated models based on record linkage
Motivating Example: Repetitive Fraud
Lots of people can't pay their bill, but they want phone service anyway:

<table>
<thead>
<tr>
<th>Name</th>
<th>Ted Hanley</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>14 Pearl Dr St Peters, MN</td>
</tr>
<tr>
<td>Balance</td>
<td>$208.00</td>
</tr>
<tr>
<td>Disconnected</td>
<td>2/19/04 (nonpayment)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Debra Handley</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>14 Pearl Dr St Peters, MN</td>
</tr>
<tr>
<td>Balance</td>
<td>$142.00</td>
</tr>
<tr>
<td>Connected</td>
<td>2/22/04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Elizabeth Harmon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>APT 1045 4301 ST JOHN RD SCOTTSDALE, AZ</td>
</tr>
<tr>
<td>Balance</td>
<td>$149.00</td>
</tr>
<tr>
<td>Disconnected</td>
<td>2/19/04 (nonpayment)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Elizabeth Harmon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>180 N 40TH PL APT 40 PHOENIX, AZ</td>
</tr>
<tr>
<td>Balance</td>
<td>$72.00</td>
</tr>
<tr>
<td>Connected</td>
<td>1/31/04</td>
</tr>
</tbody>
</table>
Motivating Example: Repetitive Fraud

How can we identify that it is the same person behind both accounts?
Motivating Example: Challenges

- This is a problem of record linkage and graph matching, but because of obfuscation, we can only count on entity matching.
- But the number of potential matches is huge…
  - 10 K/day, 300K/month
  - 5 K/day, 150 K/month

  45 billion comparisons

- If we have an efficient representation of entities, we might be able to make a dent…

  Now, let's talk about our representation
Our Approach: Defining Dynamic Graphs

We adopt an *Exponentially Weighted Moving Average* (EWMA):

\[ G_t = \theta G_{t-1} \oplus (1 - \theta) g_t \]

i.e. today’s graph is defined recursively as a convex combination of yesterday’s graph and today’s data

- Advantages:
  - recent data has most influence
  - only one most recent graph need be stored

We also use two types of approximation of the graph, by pruning:

- **Global pruning of edges** – overall threshold \( \varepsilon \) below which edges are removed from the graph

- **Local pruning of edges** – designate a maximal in and out degree \( k \) for each entity, and assign an overflow bin
Our Approach: Representation

• Because we are interested in entities, and to facilitate efficient storage, we represent the entire graph as a union of entity graphs.

• These are our atomic units of analysis, a signature of the node’s behavior.

• Storing hundreds of millions of small graphs is much more efficient than storing one massive graph, especially in an indexed database.

• Pros: efficiency, recursion    Cons: redundancy

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2222222222</td>
<td>100.3</td>
<td>1111111111</td>
<td>90.1</td>
<td></td>
</tr>
<tr>
<td>3213232423</td>
<td>27.0</td>
<td>9098765453</td>
<td>11.3</td>
<td></td>
</tr>
<tr>
<td>8876457326</td>
<td>5.4</td>
<td>2122121212</td>
<td>3.0</td>
<td></td>
</tr>
<tr>
<td>9908989898</td>
<td>0.9</td>
<td>8887878787</td>
<td>0.1</td>
<td></td>
</tr>
</tbody>
</table>
Our Approach: Representation

Update the graph by updating all of the atomic units daily – so any time we access the data we have the most recent representation.

Yesterday's graph

\[
\begin{array}{c|c}
2222222222 & 100.3 \\
1111111111 & 90.1 \\
3213232423 & 27.0 \\
9098765453 & 11.3 \\
8876457326 & 5.4 \\
2122121212 & 3.0 \\
9908989898 & 0.9 \\
8887878787 & 0.1 \\
\end{array}
\]

Today's data

\[
\begin{array}{c|c}
1111111111 & 20.0 \\
2122121212 & 10.0 \\
9991119999 & 5.0 \\
\end{array}
\]

= Today's graph

\[
\begin{array}{c|c}
1111111111 & 92.1 \\
2222222222 & 90.3 \\
3213232423 & 24.3 \\
9098765453 & 10.1 \\
8876457326 & 4.9 \\
2122121212 & 3.7 \\
9991119999 & 0.5 \\
3990898989 & 0.8 \\
8887878787 & 0.09 \\
\end{array}
\]
Our Approach: Approximation

- Removes stale edges
- Reduces effect of supernodes
- Increases efficiency
- Preserves entity weight

\[
\begin{bmatrix}
1111111111 & 92.1 \\
2222222222 & 90.3 \\
3213232423 & 24.3 \\
9098765453 & 10.1 \\
8876457326 & 4.9 \\
2122121212 & 3.7 \\
9991119999 & 0.5 \\
3990898989 & 0.8 \\
8887878787 & 0.09
\end{bmatrix}
= \begin{bmatrix}
1111111111 & 92.1 \\
2222222222 & 90.3 \\
3213232423 & 24.3 \\
9098765453 & 10.1 \\
8876457326 & 4.9 \\
2122121212 & 3.7 \\
9991119999 & 0.5 \\
3990898989 & 0.8 \\
Other & 1.4
\end{bmatrix}
\]
Our Approach: Parameter Setting

- Let A and B be two entities.

- Weighted Dice:
  \[ WD(A, B) = \frac{I_{j \in A \cap B}}{1 + \sum_j p_A(j)} (p_A(j) + p_B(j)) \]

- Hellinger Distance:
  \[ HD(A, B) = \sum_{j \in (A \cap B)} \sqrt{p_A(j) p_B(j)} \]

- For each value
  - Set \( \varepsilon \) to be a low tolerance value
  - For a range of k, optimize \( \theta \)
  - Look at performance plots to select parameters
Applying our Method

• Results:
  – We identify 50-100 of these cases per day
  – 95% match rate
  – 85% block rate
  – Credited with saving telecom millions of dollars
  – By far the most reliable matching criteria is the entity based matching
  – Optimized parameter set outperforms both current process and current theta and optimized k

*We also demonstrate our method on email and clickstream data
Other Applications

• Subscription Fraud
  – People want phone service but don’t want to pay for it

• Target Marketing/Advertising
  – Firms want to perform direct one-to-one targeting

• Alias (via email) detection across social networking sites
  – Firms want to have a complete picture of social networks

• Author attribution
  – De-duplication of names in scientific texts
IT Works Well. BUT When?

– People don’t drastically change their behavior

– People are not highly clustered

– Sample: We can observe enough of the population to estimate the score distributions for matches (and non-matches)
Challenges

• Modeling (simulating) emerging networks
• Scale
• Sampling
• Incorporating additional node and edge attributes
• Making decisions with limited information
• Utilizing more of the network
• Observing real world complete dynamic network data
• Privacy implications
Opportunities

- Theoretical guarantees (provably-correct) for matching given a graph structure and model of behavioral change
- Matching is possible for a number of real world scenarios and applications
- Evaluation can incorporate individual level costs
- Problems where determining non-matches is of interest
References


Questions?

Thanks to former collaborators:

D. Agarwal, R. Bell, C. Volinsky

contact: shawndra@wharton.upenn.edu