Recurrent Subgraph Prediction
PReSub

Saurabh Nagrecha and Nitesh V. Chawla
University of Notre Dame

Horst Bunke
University of Bern

ASONAM 2015
Interactions in networks aren’t always dyadic.

Fig 1a: Simple group messages (hub-and-spoke).

Fig 1b: More complex hierarchies in businesses.
Influence in social networks
Inferring attacks in anonymized social networks
Functional Discovery in biological networks
Recurrent Subgraphs

Fig 2: Weekly snapshots of the Enron email corpus.

<table>
<thead>
<tr>
<th>Source Node</th>
<th>Destination Node</th>
<th>$t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>432</td>
<td>23432</td>
<td>54</td>
</tr>
<tr>
<td>4254</td>
<td>437854</td>
<td>54</td>
</tr>
<tr>
<td>473743</td>
<td>32</td>
<td>55</td>
</tr>
<tr>
<td>93535</td>
<td>35443</td>
<td>55</td>
</tr>
</tbody>
</table>
Fig 3: Distribution of recurrent edges in the Enron network (across all timestamps).
Say we have network snapshots as below:

- $G_1 = \{l_1, l_2, l_3\}$
- $G_2 = \{l_2, l_3, l_4\}$
- $G_3 = \{l_1, l_2, l_4\}$
- $G_4 = \{l_1, l_2, l_3\}$

We aren’t interested in the links that are static signals; instead we want to register the “blips”:

- $G'_1 = \{\}$
- $G'_2 = \{\}$
- $G'_3 = \{l_1\}$
- $G'_4 = \{l_3\}$

Network instance

- = bag of links
- = set of transactions

Most frequently recurring links

- $GetFrequentTransactions(allTransactions, minSupport)$
- $GetFrequentTransactions([G'_1, ..., G'_n], minSupport)$
There are predictable patterns in networks. Can we identify:

- *What* these patterns are
- *When* they occur
- Effective “early warning” methods to predict them
There are predictable patterns in networks.

Can we identify:

- \textit{What} these patterns are
- \textit{When} they occur
- Effective “early warning” methods to predict them

\textbf{Solution:} Frequent Subgraph Mining
Problem Statement

There are predictable patterns in networks. Can we identify:

- *What* these patterns are
- *When* they occur
- Effective “early warning” methods to predict them

**Solution:** Subgraph Prediction
There are predictable patterns in networks. Can we identify:

- *What* these patterns are
- *When* they occur
- Effective “early warning” methods to predict them

**Solution:** Early Warning Subgraphs
Predict individual links of the subgraph.

If $l \in$ subgraph, predict for occurrence of $l$. If $l \notin$ subgraph, predict for non-occurrence of $l$.

State-of-the-art link prediction methods employed using LPMade suite [Lichtenwalter, R. & Chawla, N. JMLR (2011)].
We use GEDs to contextualize the subgraphs occurrence with respect to the global scenario in the network.

Advantages of using GEDs:

- Flexible definition allows for weighted/unweighted, directed/undirected graphs.
- Inexact matching allows for “unknown” links in graph instances.

Fig 5: A pictorial summary of vector space embedding in GEDs.
Networks often exhibit a telltale “build up” to the desired structure.

Use these early warning subgraphs as features to predict target subgraph.

Learn how these breadcrumbs lead to target subgraph for given data.
Conclusion: We need to think of subgraphs as emergent structures in their own right and not just a composition of links.

Our method achieves high AUROC performance in predicting subgraphs, and outperforms link prediction on:

- Commercial cellular phone calls
- Wikipedia Co-authorship
- Enron Email Corpus
- Facebook Wall Posts

Fig 7: AUROC Performance of our method v/s baseline link prediction.
Research was supported in part by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053, National Science Foundation (NSF) Grant OCI-1029584, and by the grant FA9550-12-1-0405 from the U.S. Air Force Office of Scientific Research (AFOSR) and the Defense Advanced Research Projects Agency (DARPA).
Bibliography


Thanks!

Q & A
### Parameters

<table>
<thead>
<tr>
<th>Prototype Selection Method</th>
<th>AUROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.869</td>
</tr>
<tr>
<td>Border</td>
<td>0.843</td>
</tr>
<tr>
<td>TargetSphere</td>
<td>0.887</td>
</tr>
</tbody>
</table>

Fig 8a (above): How prototypes are selected.

Fig 8b (right): How many prototypes are enough?
Fig 9: If a very broad window is chosen, the fine-grained aspects of recurrence may be lost, and if a very narrow window is chosen, there may be very little difference between two consecutive snapshots.
Table 1: A summary of the data sources used.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Number of Nodes</th>
<th>Number of Edges</th>
<th>Time Span</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile</td>
<td>8,321,119</td>
<td>712 million</td>
<td>65 days</td>
</tr>
<tr>
<td>Wiki</td>
<td>25,323,882</td>
<td>266 million</td>
<td>~4 years</td>
</tr>
<tr>
<td>Enron</td>
<td>87,098</td>
<td>1,147,028</td>
<td>~4 years</td>
</tr>
<tr>
<td>Facebook</td>
<td>46,715</td>
<td>803,744</td>
<td>~2 years</td>
</tr>
</tbody>
</table>