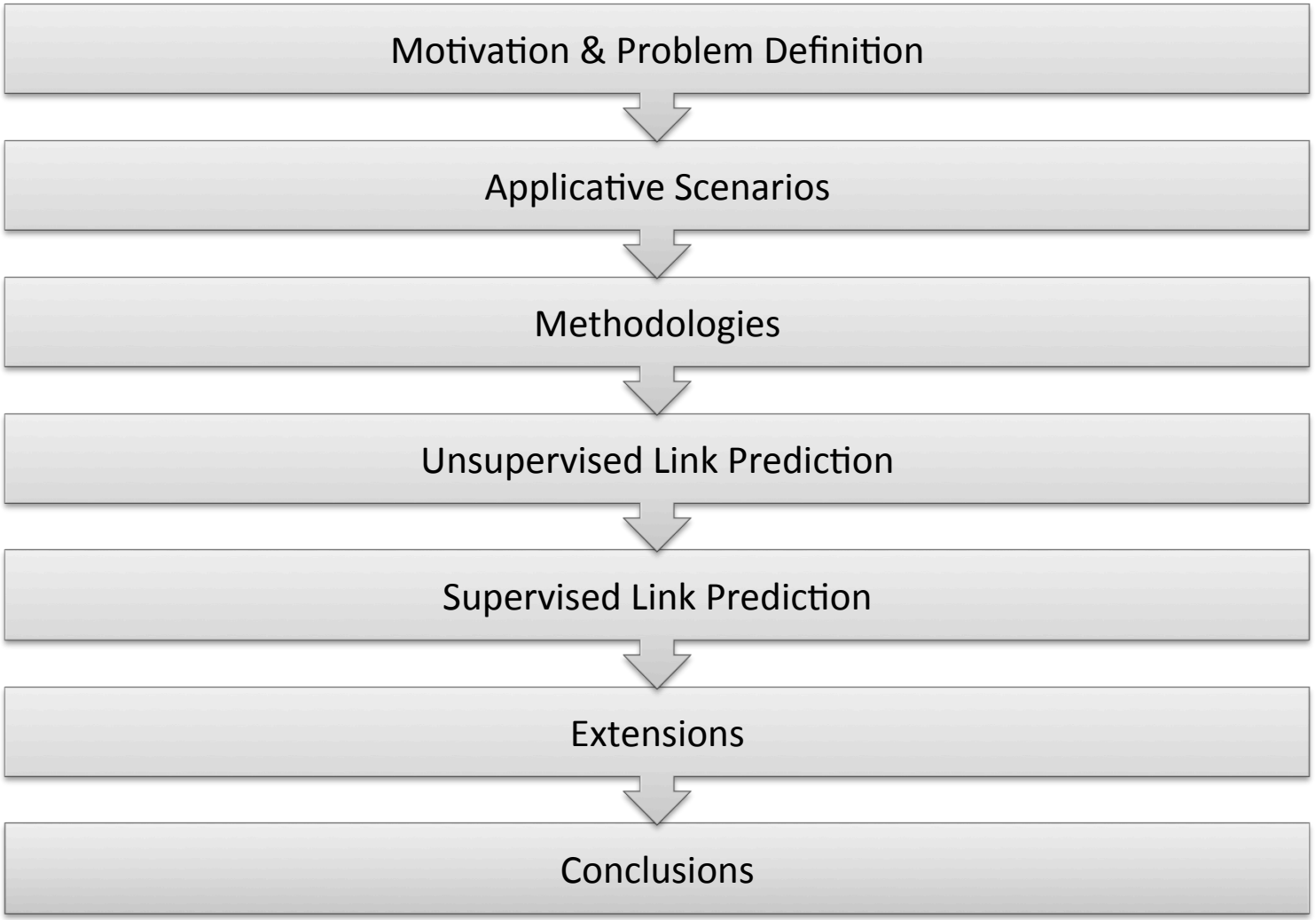
The Link Prediction Problem for Social Network

Giulio Rossetti

22/04/2013

Outline

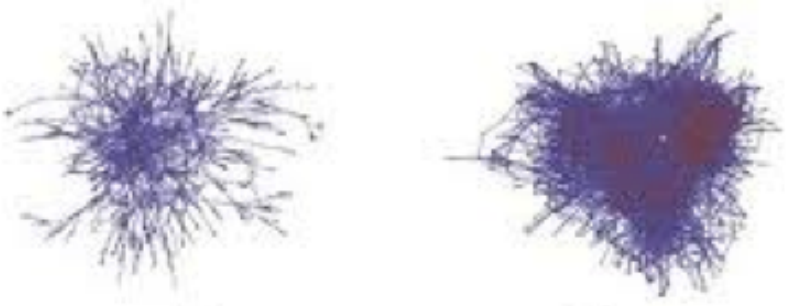

Motivation & Problem Definition
Applicative Scenarios
Methodologies
Unsupervised Link Prediction
Supervised Link Prediction
Extensions
Conclusions

Motivation & Problem Definition

- **Motivation**:
  Understanding how networks evolve

- **Problem definition**

  Given a snapshot of a network at time t, we want to accurately predict the edges that will appear in the network during the interval (t, t+1)



Link Prediction

Link Prediction is not an easy task because:

1. Given a graph G = (V,E) the set of possible edges to be predicted is O(|V|^2);
2. Real networks appear to be sparse.

False Positive prediction issue!!

Applicative scenarios

- Suggest interactions or collaborations that haven't yet been exploited within an organization;
- Monitor terrorist networks – deducing possible interaction between terrorists (without direct evidence);
- Friendship prediction (i.e. as in Facebook and LinkedIn)

Link Prediction hint…

Co-authorship network:

- Scientists who are "close" in the network (i.e. have common colleagues) → will likely collaborate in the future

Goal:

make this intuitive notion precise and understand which measures of "proximity" leads to accurate predictions

Methods for Link Prediction

1. Consider as input a graph G at time t
2. Consider all the possible couples of nodes (u,v)
3. Compute a link formation probability score:

score(u,v)

4. Build a list of all possible edges ordered by scores (from highest tolowest)
5. Verify the prediction on the same graph at time t+1

score is a measure of *proximity*

Evaluate the results

Given a predictor p is there a way to decide if it is a ”good” one?

$$performance(p) = \frac{TP}{TP+FP}$$

Idea: verify if p outperform the random predictor.

Random Predictor: each edge has the same probability to appear in the network

$$ratio = \frac{performance(p)}{performance(p_{random})} = \frac{performance(p)}{\frac{|E_{new}|}{\frac{|V|*(|V|-1)}{2} - |E_{old}|}}$$

if ratio > 1 the predictor p is meaningful.

Comparing performances of different predictors

Which Link Predictor is the best?

We need to analyze either the performances ratio, ROC and/or Precision Recall curve.

	p'	n'
p	TP	FN
n	FP	TN

Confusion Matrix

ROC and PR curve

Precision Vs. Recall :

- Precision: $PPV = Performance = \frac{TP}{TP+FP}$
- Recall: $TPR = \frac{TP}{TP+FN}$

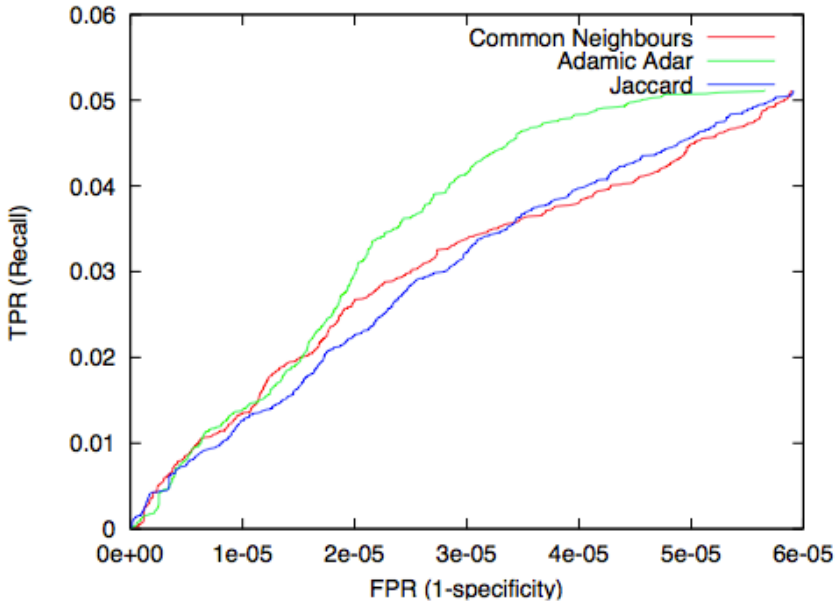
ROC (Receiver operating characteristic):

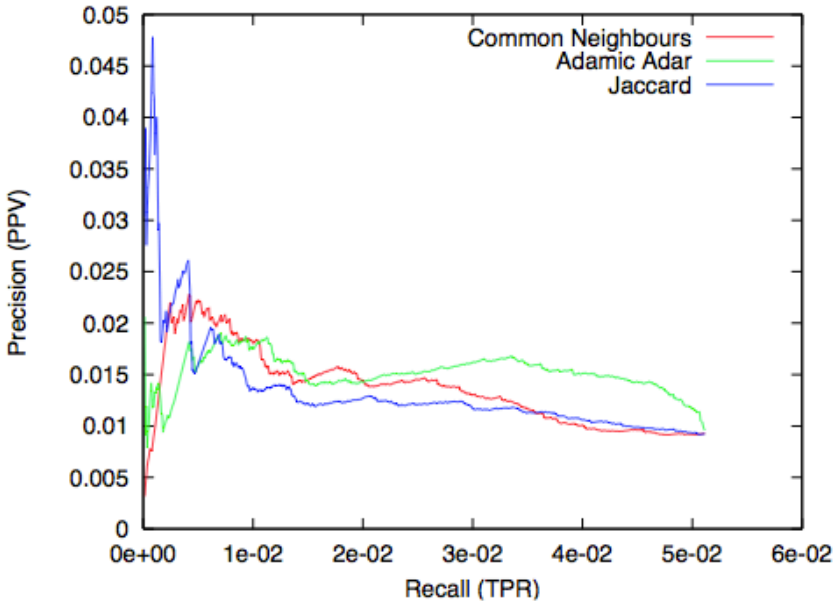
- 1-Specificity: $FPR = \frac{FP}{FP+TN}$
- Recall: $TPR = \frac{TP}{TP+FN}$

NB:

- ROC and PR spaces are isomorphic.
- Another measure often used is AUC (area under curve)

ROC and PR curve





Link Prediction Approaches

Link Prediction problem can be tackled following two different ways:

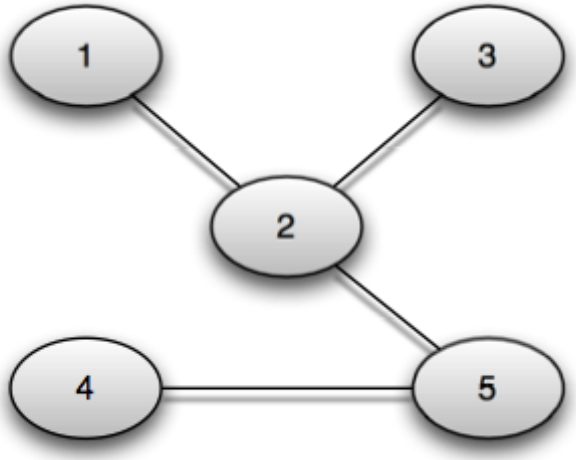
1. **Unsupervised**:
   - defining a set of standard proximity measures unrelated to the particular network

2. **Supervised**:
   - extracting knowledge from the network in order to improve prediction accuracy

Unsupervised Link Prediction

Unsupervised measurements rely on different structural properties of networks:

- **Neigborhood** measures
  - Common Neighbors, Adamic Adar, Jaccard, Preferential Attachment
- **Path-based** measures
  - Graph distance, Katz
- **Ranking**
  - Sim Rank, Hitting time, Page Rank



Neighborhood measures

"How many friends we have to share in order to become friends?"

Common Neighbors: the more friends we share, the more likely that we will become friends

$$score(u, v) = |\Gamma(u) \cap \Gamma(v)|$$

Jaccard: the more similar our friends circles are, the more likely that we will become friends

$$score(u, v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|}$$

Neighborhood measures

"How many friends we have to share in order to become friends?"

Adamic Adar: the more *selective* our mutual friends are, the more likely that we will become friends

$$score(u, v) = \sum_{z \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\log(|\Gamma(z)|)}$$

Preferential Attachment: more friends we have, the more likely that we will become friends

$$score(u, v) = |\Gamma(u)| * |\Gamma(v)|$$

Path-based Measures

"How distant we are?"

Graph Distance: (negated) length of shortest path between u & v

Katz$_\beta$: weighted sum over all the paths between u & v

$$score(u, v) = \sum_{l=1}^{\infty} \beta^l \left| paths_{u,v}^{\langle l \rangle} \right|$$

where: $paths_{u,v}^{\langle l \rangle}$ ={paths of length exactly l from u to v}

SimRank

"Two nodes are similar to the extent that they are joined by similar neighbors"

$$similarity(u,v) = \gamma * \frac{\sum_{a \in \Gamma(u)} \sum_{n \in \Gamma(v)} similarity(a,b)}{|\Gamma(u)| * |\Gamma(v)|}$$

$$score(u,v) = similarity(u,v)$$

Undupervised LP: Results & Limits

Results

- No single clear winner
- Almost all predictors outperform the random predictor

  ⇒ there is useful information in network topology

Limits

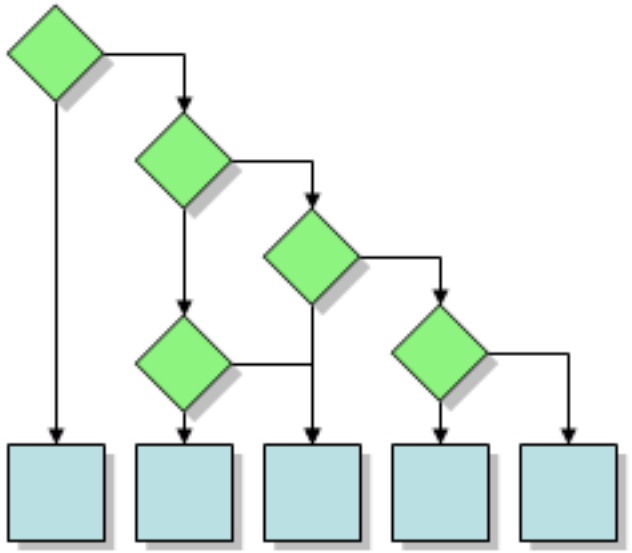
- Different kinds of networks are described by general closed formulae
- An average overall performance between 10% and 16%.

Supervised LP: Classification

The process is now organized in 2 steps:

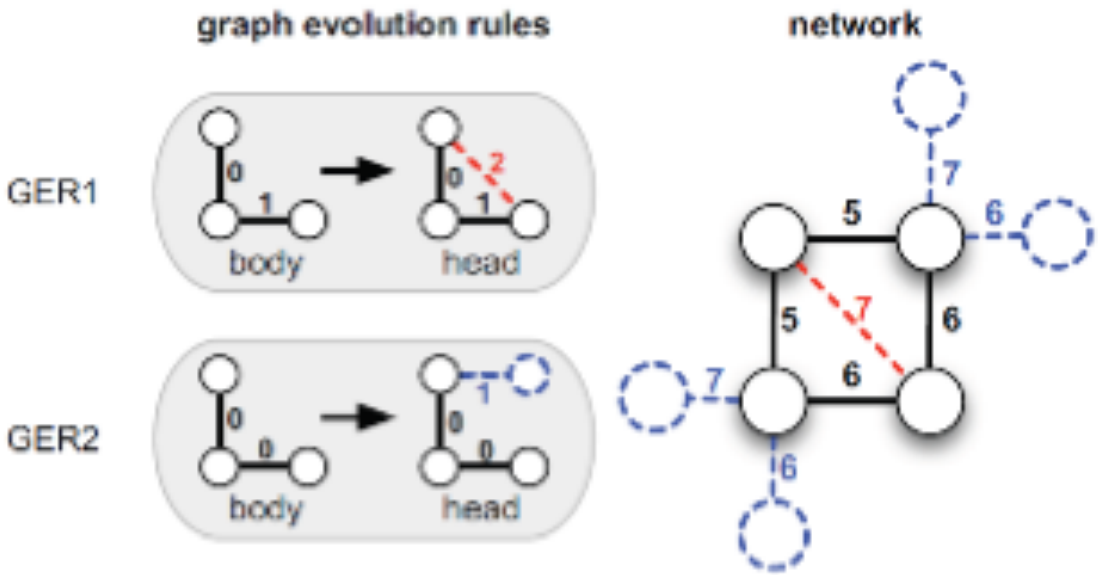
1. Learning a model
2. Use the model for the prediction

A natural way to do it:
build a Classifier over a set of attributes.



Supervised LP: Patterns

GERM: Evolutionary rules can be extracted from the network in order to predict recurrent patterns.



Supervised LP: Results & Limits

Results

- Higer performances w.r.t. unsupervised approaches

Limits

- The two-step predictive process is slower than unsupervised ones.

Extensions

Accuracy could be improved extending simple models with more complex informations:

- Temporal & evolutionary analysis
- Link strength
- Multidimensionality
- Geographical information
- …

Conclusions

Predict new link that will arise in a network is not an easy task because:

1. Networks are, usually, sparse
2. Weak links are difficult to predict
3. Huge False Positive prediction
4. Simple approaches are “too simple”
5. Complex approaches are costly