New Perspectives and Methods in Link Prediction

Ryan N. Lichtenwalter, Jake T. Lussier, and Nitesh V. Chawla

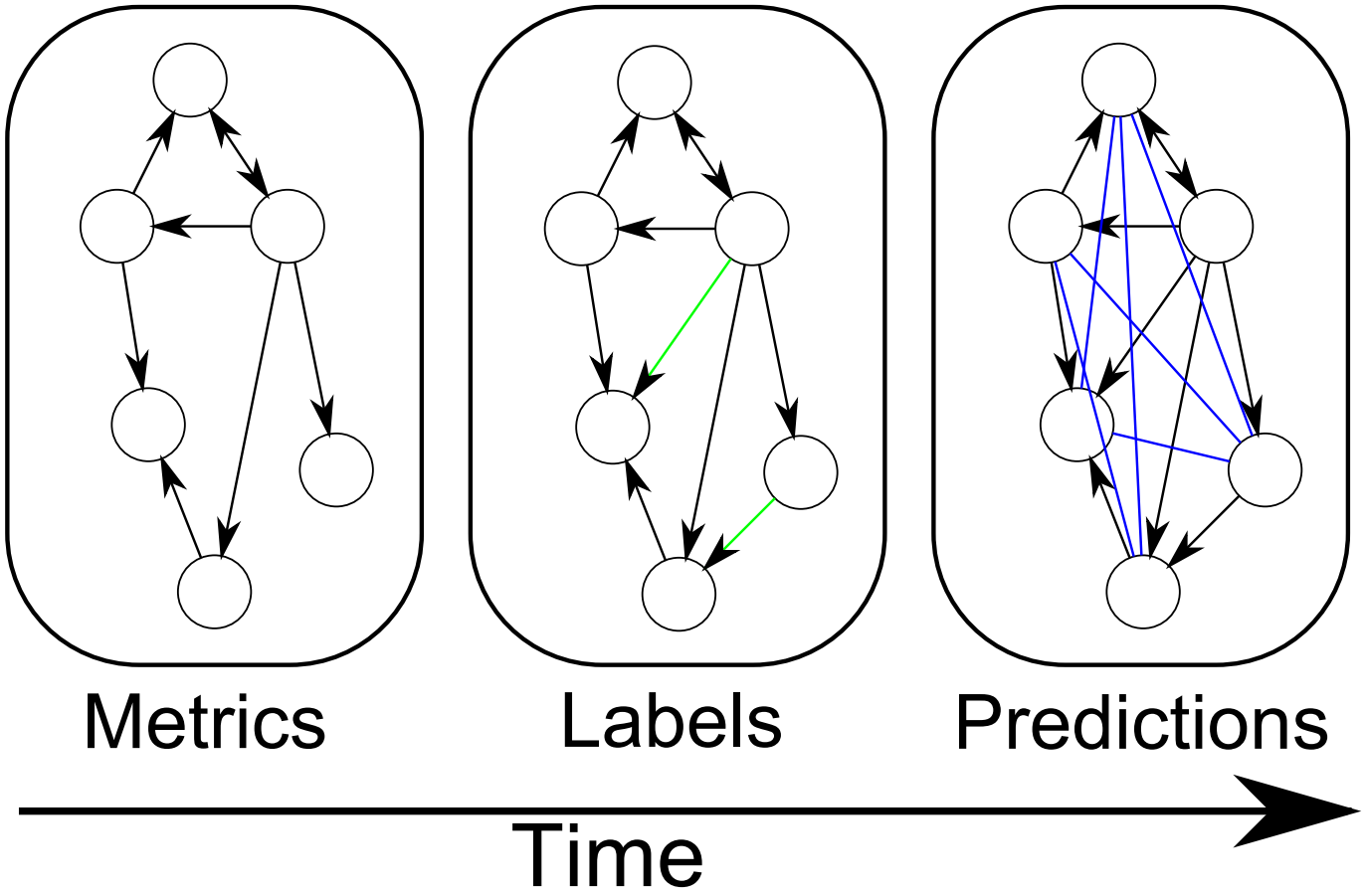
KDD 2010, Washington, D.C.

25 July 2010

UNIVERSITY OF NOTRE DAME

iCeNSA
Interdisciplinary Center for Network Science & Applications

Introduction



Data

`phone` (weighted, directed)- a stream of event records from a major non-American cellular phone service provider

- Weeks 1-5
  - 5.5M nodes
  - 19.7M links
- Week 6
  - 4.4M nodes
  - 8.5M links

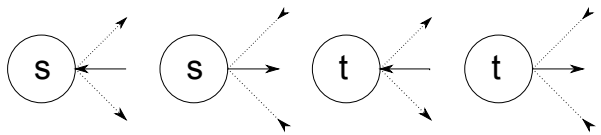
`condmat` (weighted, undirected) - a physics collaboration data set from Mark Newman

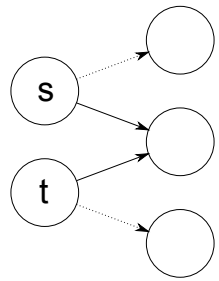
- Years 1995-1999
  - 13.9K nodes
  - 80.6K links
- Year 2000
  - 8.5K nodes
  - 41.0K links

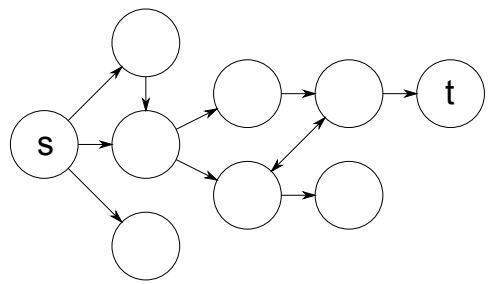
	`phone`	`condmat`
Assortativity Coef.	0.293	0.177
Average Clustering Coef.	0.187	0.642
Mean Degree	3.88	6.42
Median Degree	3	4
Number of SCCs	1,023,044	652
Largest SCC	4,293,751	15,081
Largest SCC Diameter	25	19

Single-Metric Methods

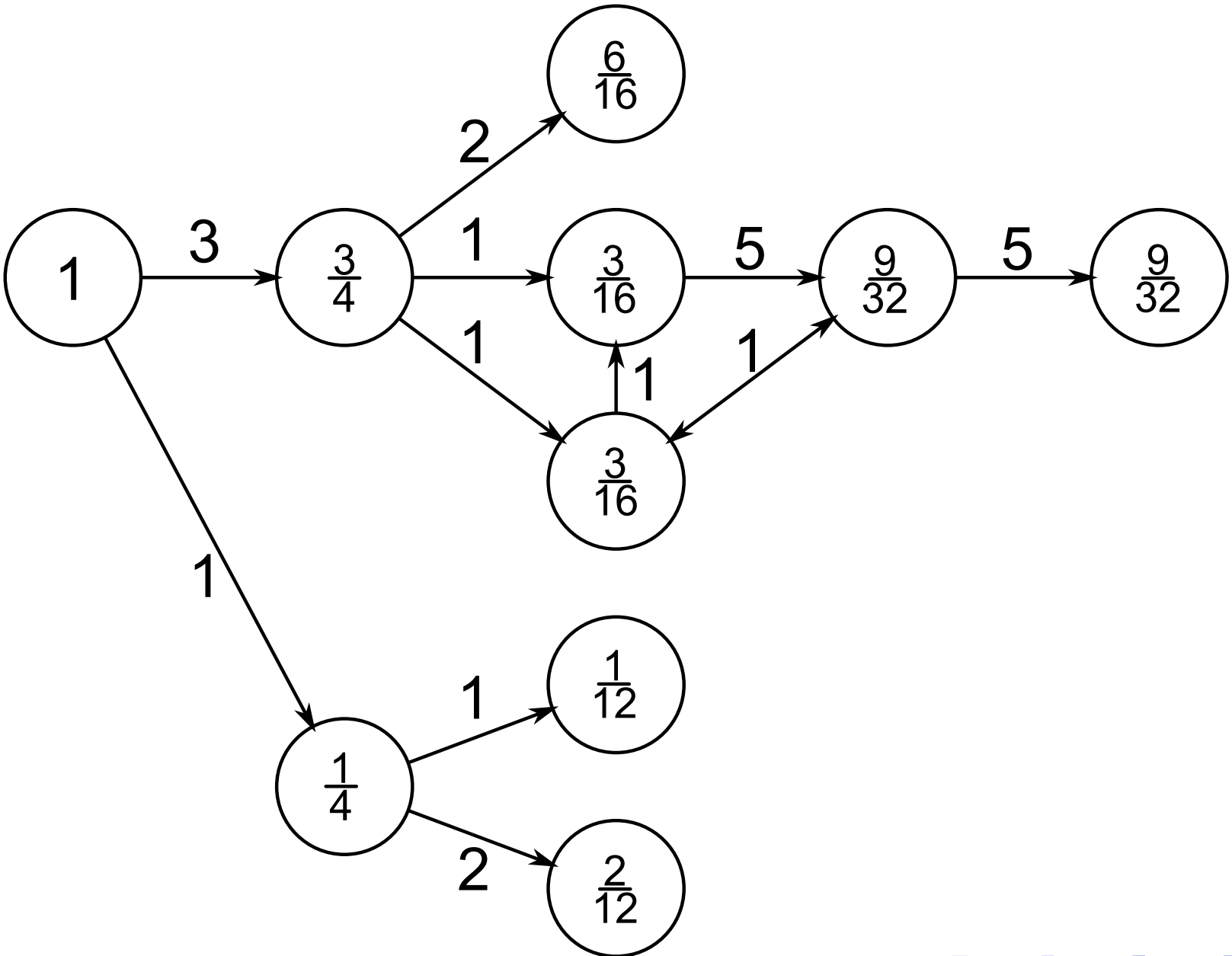
- Node-based
  - Degree
  - Weight
  - Centrality
- Neighbor-based
  - Common Neighbors
  - Jaccard's Coefficient
  - Adamic/Adar
- Path-based
  - Katz
  - Rooted PageRank
  - Hitting Time
  - Commute Time



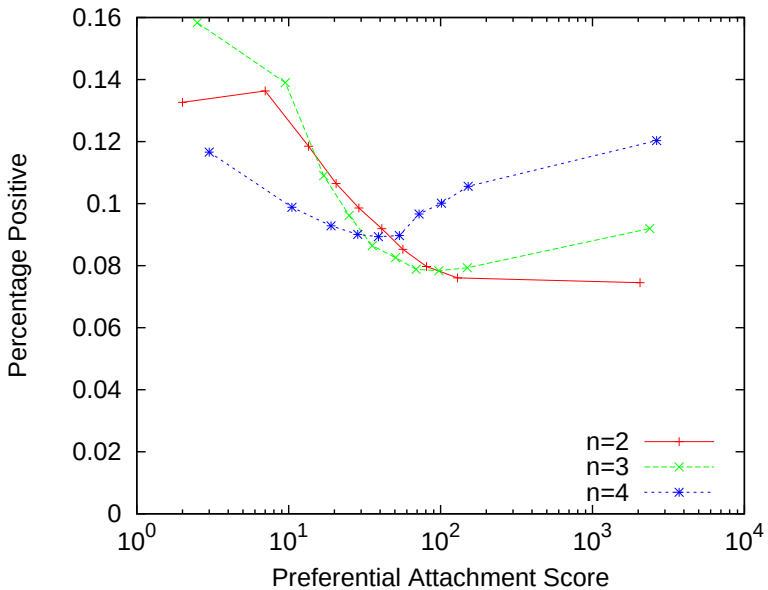




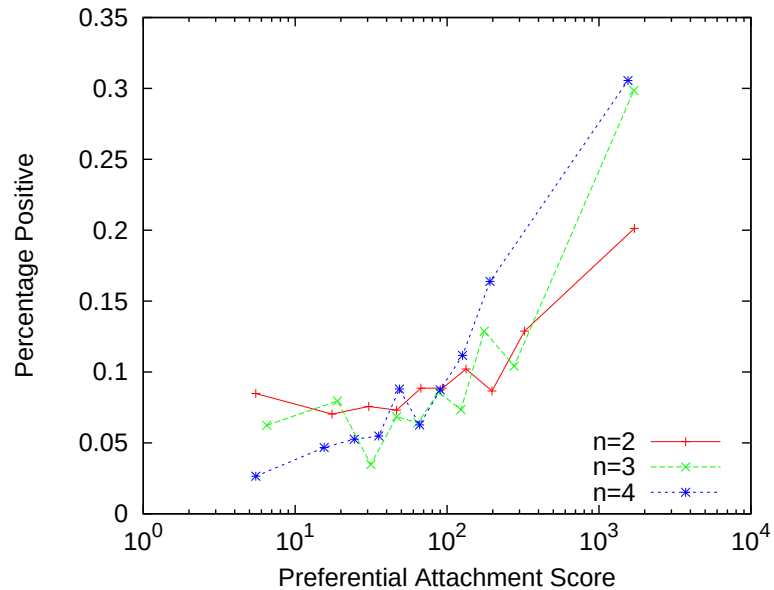
PropFlow



Local and Global Influences



phone



condmat

Feature Listing

Name	Parameters
In-Degree(i)	-
In-Volume(i)	-
In-Degree(j)	-
In-Volume(j)	-
Out-Degree(i)	-
Out-Volume(i)	-
Out-Degree(j)	-
Out-Volume(j)	-
Common Nbrs(i,j)	-
Max. Flow(i,j)	$\ell = 5$
Shortest Paths(i,j)	$\ell = 5$
PropFlow(i,j)	$\ell = 5$
Adamic/Adar(i,j)	-
Jaccard's Coef(i,j)	-
Katz(i,j)	$\ell = 5$, $\beta = 0.005$
Pref Attach(i,j)	-

The Case for Supervised Learning

The Case for Supervised Learning

- Many networks are built from data streams. We are inundated with new data, which eventually becomes truth data.

The Case for Supervised Learning

- Many networks are built from data streams. We are inundated with new data, which eventually becomes truth data.
- The classification step takes less time than gathering topological measures.

The Case for Supervised Learning

- Many networks are built from data streams. We are inundated with new data, which eventually becomes truth data.
- The classification step takes less time than gathering topological measures.
- Networks have different underlying mechanisms driving the formation of links.

The Case for Supervised Learning

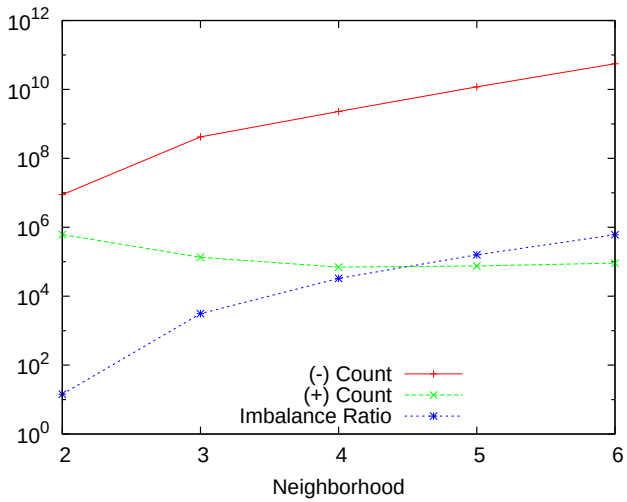
- Many networks are built from data streams. We are inundated with new data, which eventually becomes truth data.
- The classification step takes less time than gathering topological measures.
- Networks have different underlying mechanisms driving the formation of links.
- We can capture and describe interrelated mechanisms of link formation.

The Case for Supervised Learning

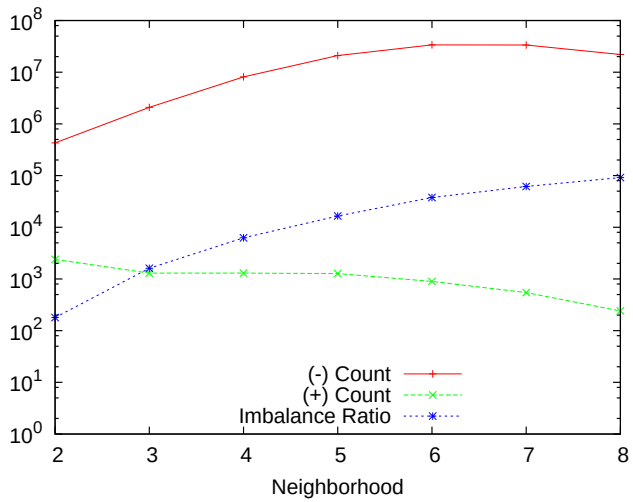
- Many networks are built from data streams. We are inundated with new data, which eventually becomes truth data.
- The classification step takes less time than gathering topological measures.
- Networks have different underlying mechanisms driving the formation of links.
- We can capture and describe interrelated mechanisms of link formation.
- We gain the opportunity to focus on differentiating the class boundaries.

Imbalance and Geodesic Distance

The class imbalance ratio for link prediction in a sparse network is lower-bounded by the number of vertices in the network.
Solution: Geodesic decomposition and undersampling!

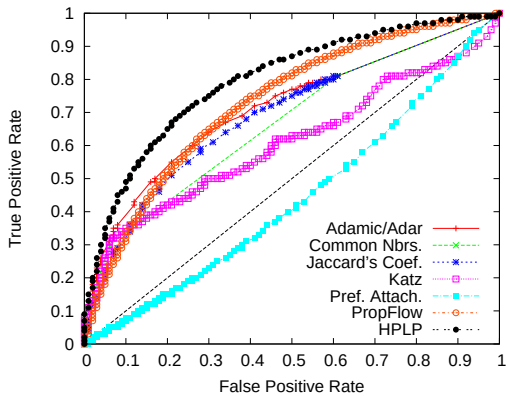


`phone`

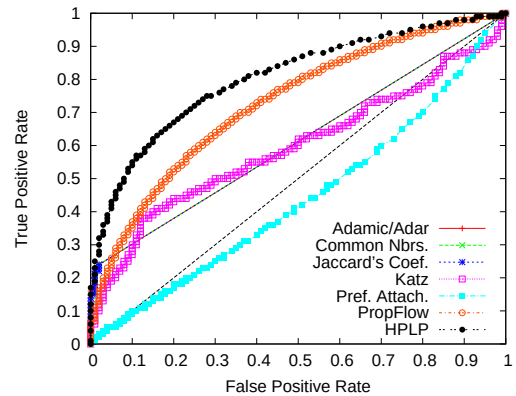


`condmat`

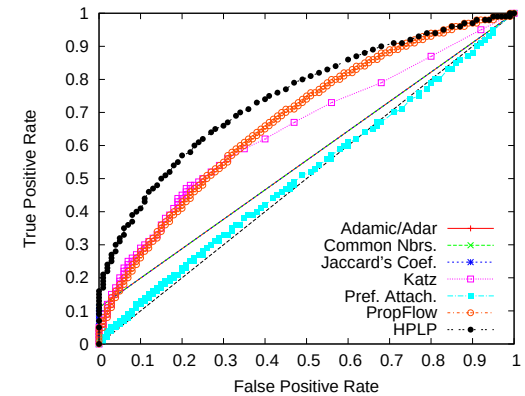
ROC Results



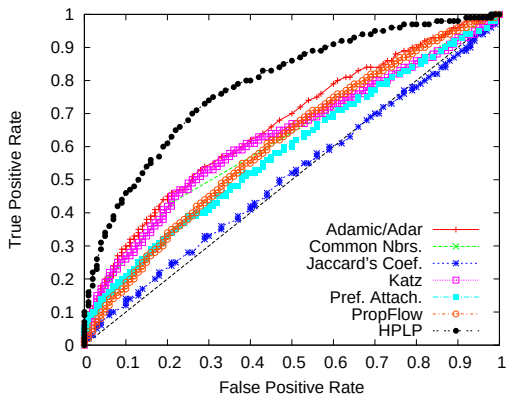
phone $n = 2$



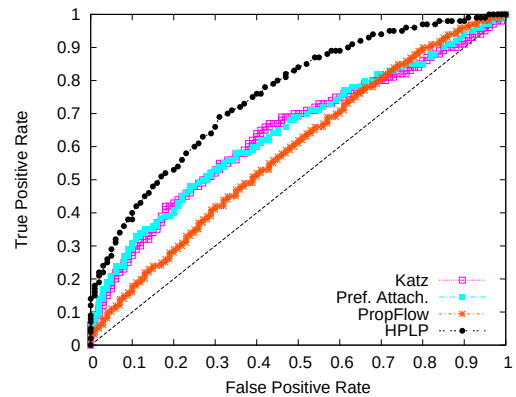
phone $n = 3$



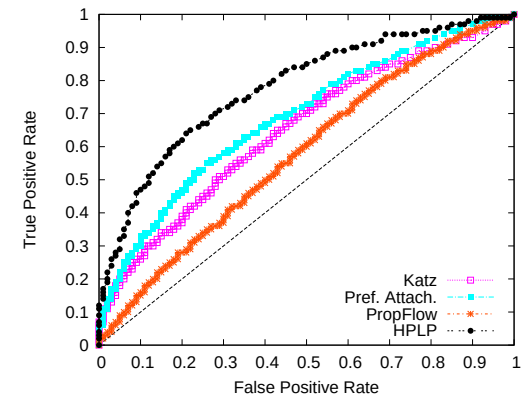
phone $n = 4$



condmat $n = 2$



condmat $n = 3$



condmat $n = 4$

Conclusions

- Links form according to a competition between local and global influences, and the dominant factor depends upon the network.
- We recommend PropFlow on communication networks such as `phone` when supervised predictors are somehow infeasible.
- HPLP outperforms single methods by 30% AUROC using a combination of undersampling and ensembles, and it requires less time than feature construction.
- Resulting data sets present a whole new world of imbalanced data challenges. Typical techniques work poorly or not at all.

Acknowledgments

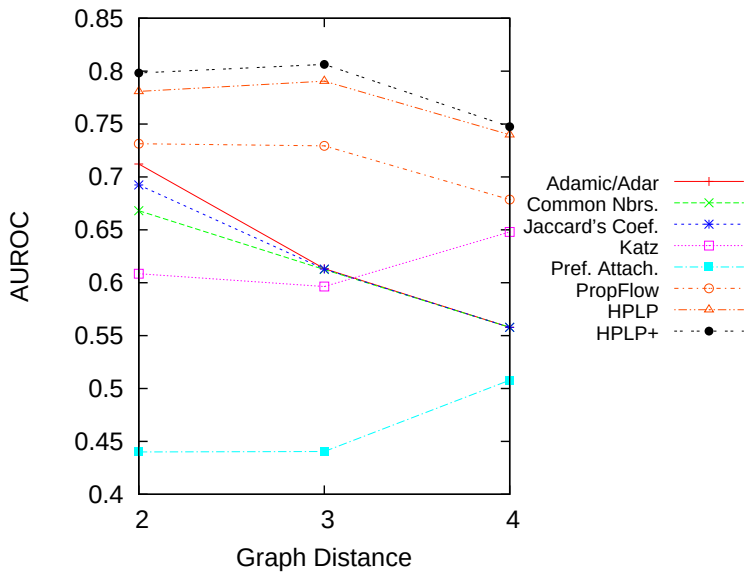
Research sponsored by:

- Army Research Laboratory (ARL) Cooperative Agreement Number W911NF-09-2-0053
- National Science Foundation (NSF) Grant BCS-0826958

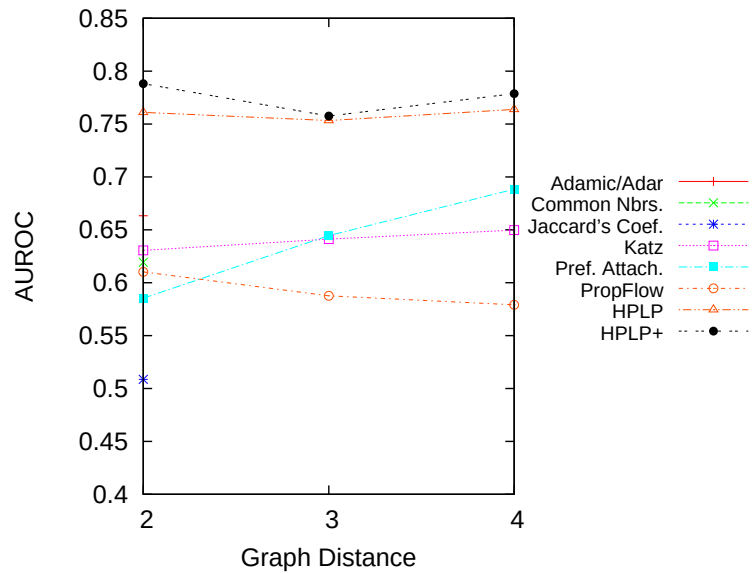
Thanks to Dawei (David) Yin for pointing out an error in the transcription of Algorithm 1 in the proceedings.
The transcription error has been fixed. For a corrected version, please see:
`http://nd.edu/~dial/papers/KDD10.pdf`

Questions, comments, or input?

AUROC Results



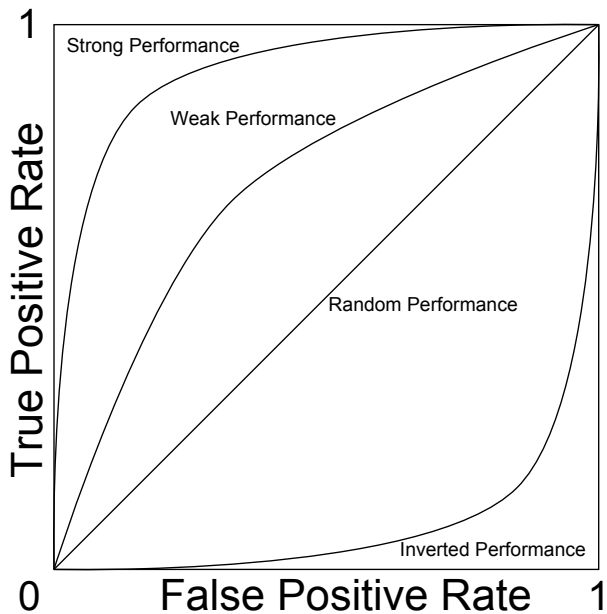
phone



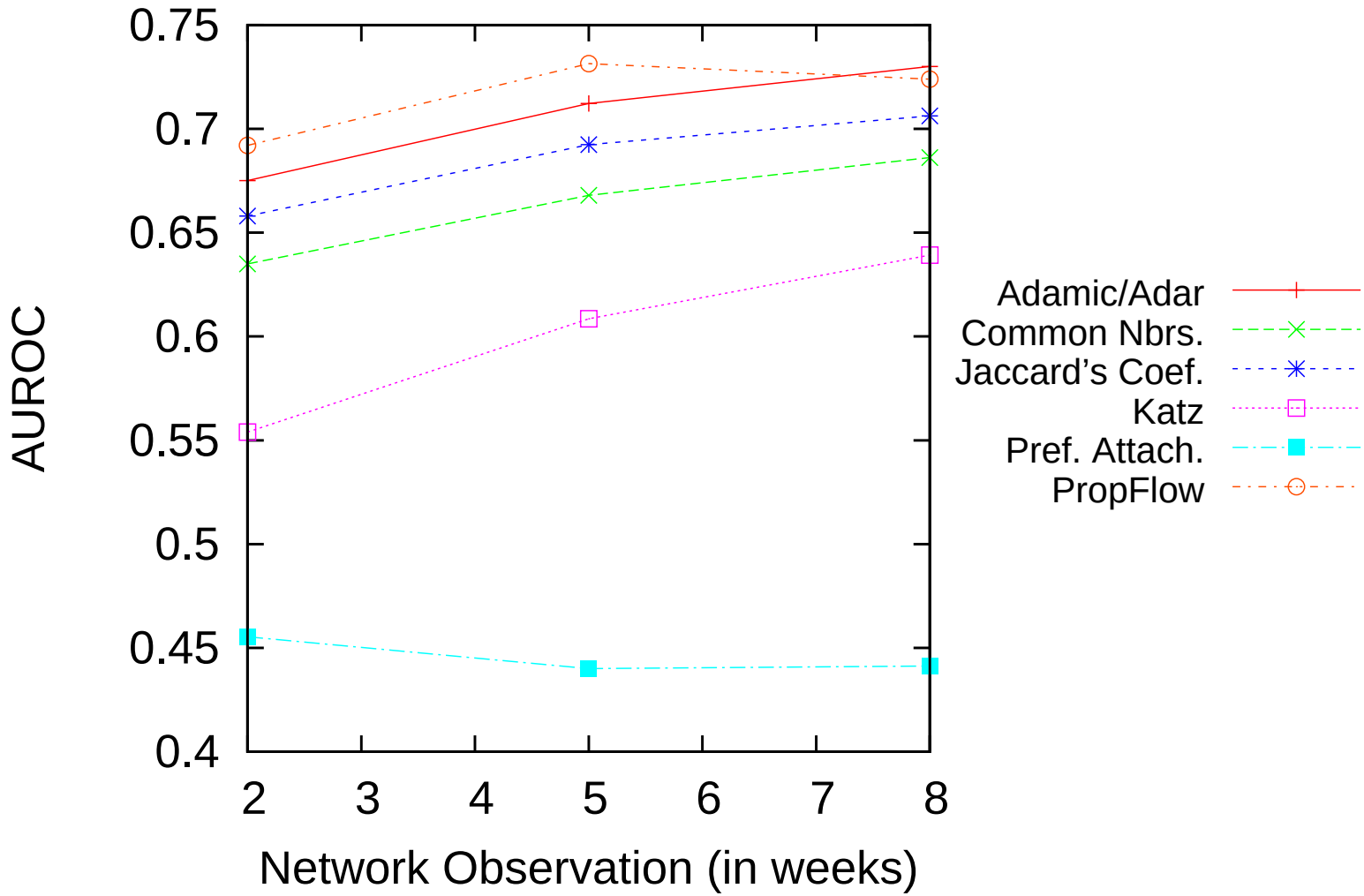
condmat

Evaluation

- Link prediction papers often report results in terms of precision and recall.
- This requires selecting arbitrary thresholds (e.g. 50% probability).
- We use threshold-agnostic measures because any threshold is domain-specific or entirely unknown.



Data and Evaluation: Network Saturation



Feature Listing

Name	Parameters	HPLP	HPLP+
In-Degree(i)	-	✓	✓
In-Volume(i)	-	✓	✓
In-Degree(j)	-	✓	✓
In-Volume(j)	-	✓	✓
Out-Degree(i)	-	✓	✓
Out-Volume(i)	-	✓	✓
Out-Degree(j)	-	✓	✓
Out-Volume(j)	-	✓	✓
Common Nbrs(i,j)	-	✓	✓
Max. Flow(i,j)	$l = 5$	✓	✓
Shortest Paths(i,j)	$l = 5$	✓	✓
PropFlow(i,j)	$l = 5$	✓	✓
Adamic/Adar(i,j)	-		✓
Jaccard's Coef(i,j)	-		✓
Katz(i,j)	$l = 5$, $\beta = 0.005$		✓
Pref Attach(i,j)	-		✓

Imbalance

Definition

Let a network $G = (V, E)$ be described as *sparse* if it maintains the property $|E| = k|V|$ for some constant $k \ll |V|$.

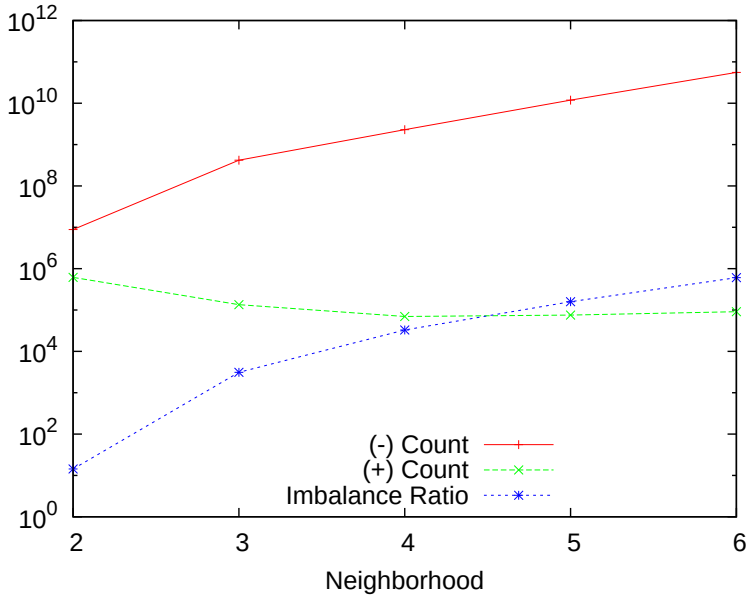
Theorem

The class imbalance ratio for link prediction in a sparse network G is $\Omega\left(\frac{|V|}{1}\right)$ when at most $|V|$ nodes may join the network.

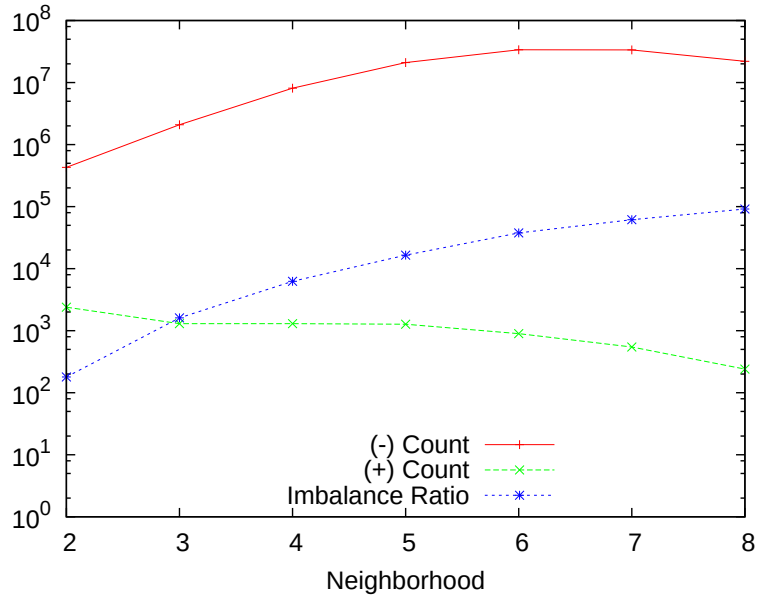
Proof.

The number of possible links in G is $|V|^2$. Then the number of missing links, $|E^C|$, is $|V|^2 - k|V| \in \Theta(|V|^2)$. Let $|V'|$ nodes and $|E'|$ links join the network. Since $|V| + |V'| \leq 2|V| \in \Theta(|V|)$, $|E| + |E'| \in \Theta(|V|)$, which requires that $|E'| \in O(|V|)$. The number of positives is $|E'|$, and there are $\left|(E \cup E')^C\right| \in \Theta(|V|^2)$ negatives. This gives us $\frac{\Theta(|V|^2)}{O(|V|)}$, equivalent to $\Omega\left(\frac{|V|}{1}\right)$, as the class ratio.

Graph Distance and Imbalance

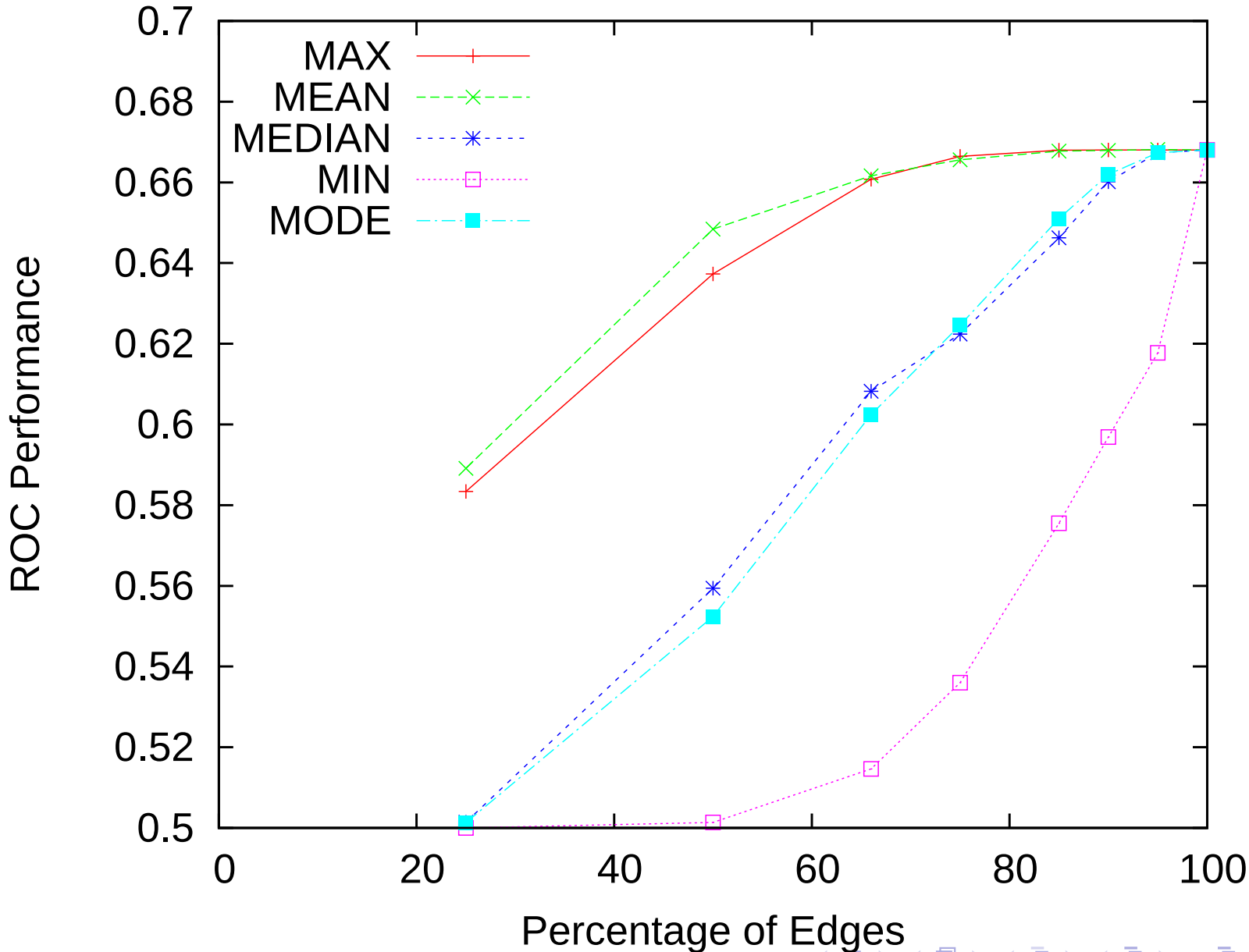


`phone`



`condmat`

Variance Reduction and Ensembles



PropFlow - Erratum

Require: network $G = (V, E)$, node v_s, max length l
Ensure: score S_{sd} for all $n \leq l$-degree neighbors v_d of v_s

1: insert v_s into *Found*
2: push v_s onto *NewSearch*
3: insert $(v_s, 1)$ into S
4: **for** $CurrentDegree \leftarrow 0$ to l **do**
5: $OldSearch \leftarrow NewSearch$
6: empty *NewSearch*
7: **while** *OldSearch* is not empty **do**
8: pop v_i from *OldSearch*
9: find *NodeInput* using v_i in S
10: $SumOutput \leftarrow 0$
11: **for** each v_j in neighbors of v_i **do**
12: add weight of e_{ij} to *SumOutput*
13: **end for**
14: $Flow \leftarrow 0$
15: **for** each v_j in neighbors of v_i **do**
16: $w_{ij} \leftarrow$ weight of e_{ij}
17: $Flow \leftarrow NodeInput \times \frac{w_{ij}}{SumOutput}$
18: insert or sum $(v_i, Flow)$ into S
19: **end for**
20: **if** v_i is not in *Found* **then**
21: insert v_i into *Found*
22: push v_i onto *NewSearch*
23: **end if**
24: **end while**
25: **end for**

Require: network $G = (V, E)$, node v_s, max length l
Ensure: score S_{sd} for all $n \leq l$-degree neighbors v_d of v_s

1: insert v_s into *Found*
2: push v_s onto *NewSearch*
3: insert $(v_s, 1)$ into S
4: **for** $CurrentDegree \leftarrow 0$ to l **do**
5: $OldSearch \leftarrow NewSearch$
6: empty *NewSearch*
7: **while** *OldSearch* is not empty **do**
8: pop v_i from *OldSearch*
9: find *NodeInput* using v_i in S
10: $SumOutput \leftarrow 0$
11: **for** each v_j in neighbors of v_i **do**
12: add weight of e_{ij} to *SumOutput*
13: **end for**
14: $Flow \leftarrow 0$
15: **for** each v_j in neighbors of v_i **do**
16: $w_{ij} \leftarrow$ weight of e_{ij}
17: $Flow \leftarrow NodeInput \times \frac{w_{ij}}{SumOutput}$
18: insert or sum $(v_j, Flow)$ into S
19: **if** v_j is not in *Found* **then**
20: insert v_j into *Found*
21: push v_j onto *NewSearch*
22: **end if**
23: **end for**
24: **end while**
25: **end for**

Wang, Satuluri, Parthasarathy Evaluation

- "The testing dataset is formed in a similar fashion: the links that are formed in the 10th year (T10 in Figure 5) are treated as testing instances that need to be predicted as positive, and we include a sample of the links that are not formed in the whole of the dataset as testing instances whose ground truth labeling is negative. The features that are used by the classifier trained previously are formed from the first 9 years of data."

Hasan and Zaki Evaluation

- "Pairs of authors that represent positive class or negative class were chosen randomly from the list of pairs that qualify. Then we constructed the feature vector for each pair of authors."
- "So, a baseline classifier would have an accuracy around 50% by classifying all the testing data points to be equal to 1 or 0, whereas all the models that we tried reached an accuracy above 80%."
- "In our experiments, we used standard cross validation approach to report the performance, so training and testing datasets are drawn from the same distribution."