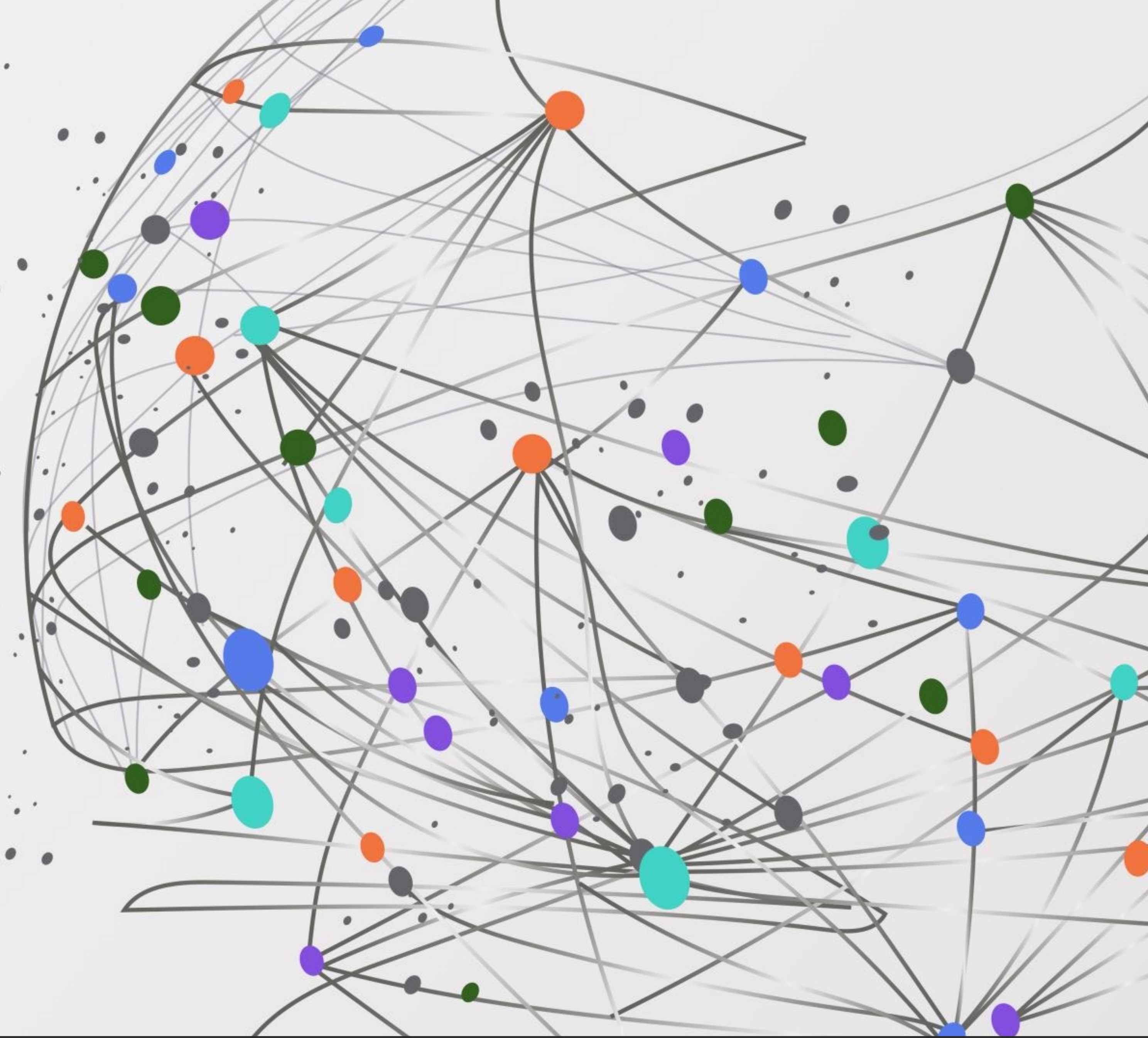
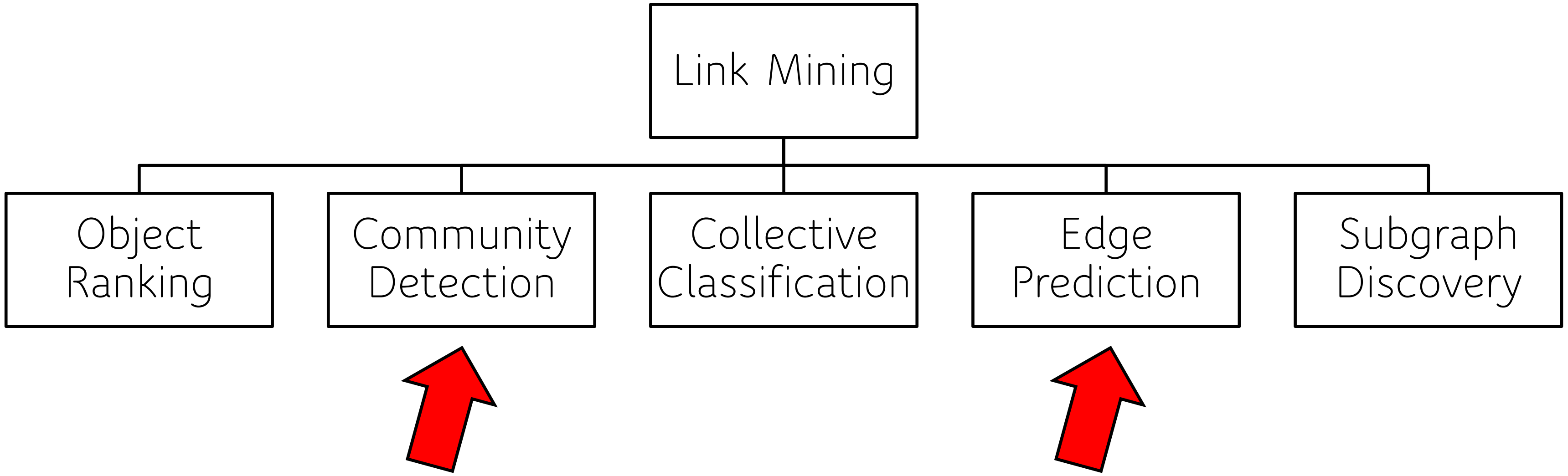
Link Mining

PRESENTED BY: MUHAMMAD SAAD ATIQUE
RIN: 662003489

10-14-2021





Main Goal: Discovering relations between vertices in networks

Detecting Community Structure in Networks using Edge Prediction methods

Authors: Bowen Yan and Steve Gregory

- What is a Community?
  - Group of vertices in a network
  - Dense edges between vertices of same community, sparse between different communities
- What is Edge Prediction?
  - Predicting edge that should exist in a network or may exist in future

What's different here?

Previous work:

Edge prediction consider vertex similarity:

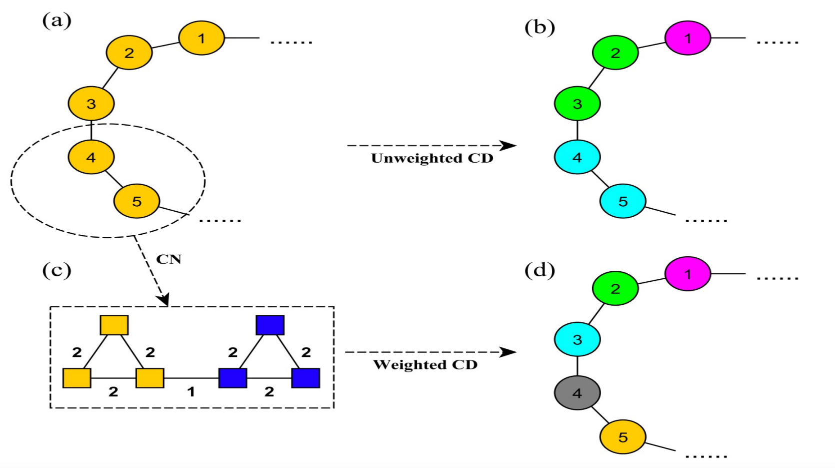
- Vertex neighbors
- Local and Global structure of network
- Assign score to each pair of vertex

New Approach:

- Instead of predicting missing edges predict weights of existing edges
- Use vertex similarity measures to compute weights

Methodology

1. Phase 1 - Get the weights
   - Vertex similarity measure based on local structure
   - Calculate scores of edges using Common Neighbors (CN)
     - CN(u,v): common neighbors between u and v +1
2. Apply existing community detection algorithms to the weighted network



Community Detection Algorithms

1. RFT
   - This is a fast fine-tuning algorithm, which is similar to a Kernighan{Lin optimization.
2. CNM
   - fast greedy modularity optimization
   - begins with a trivial partition, with very low modularity, in which each vertex is a separate community
   - repeatedly merges the pair of communities that results in the greatest increase in modularity
3. Infomap
   - uses random walks for information flows and detects community structure by compressing them.
4. COPRA
   - initializes every vertex with a unique label, and the labels then propagate between neighbors. After several iterations, the vertices in each community have the same label.
5. Louvain method
   - Optimize modularity

Data Set and Evaluation

1. Artificial Networks:
   - LFR benchmark networks
   - Normalized Mutual Information (NMI) for evaluation
   - Amount of info obtained about one var by observing another
   - NMI(X|Y) $= 1 - \frac{(H(X|Y)+H(Y|X))}{2}$

=> H(X|Y): normalized conditional entropy of X w.r.t. Y -> amount of in

required to describe X given value of Y is known

Parameters:

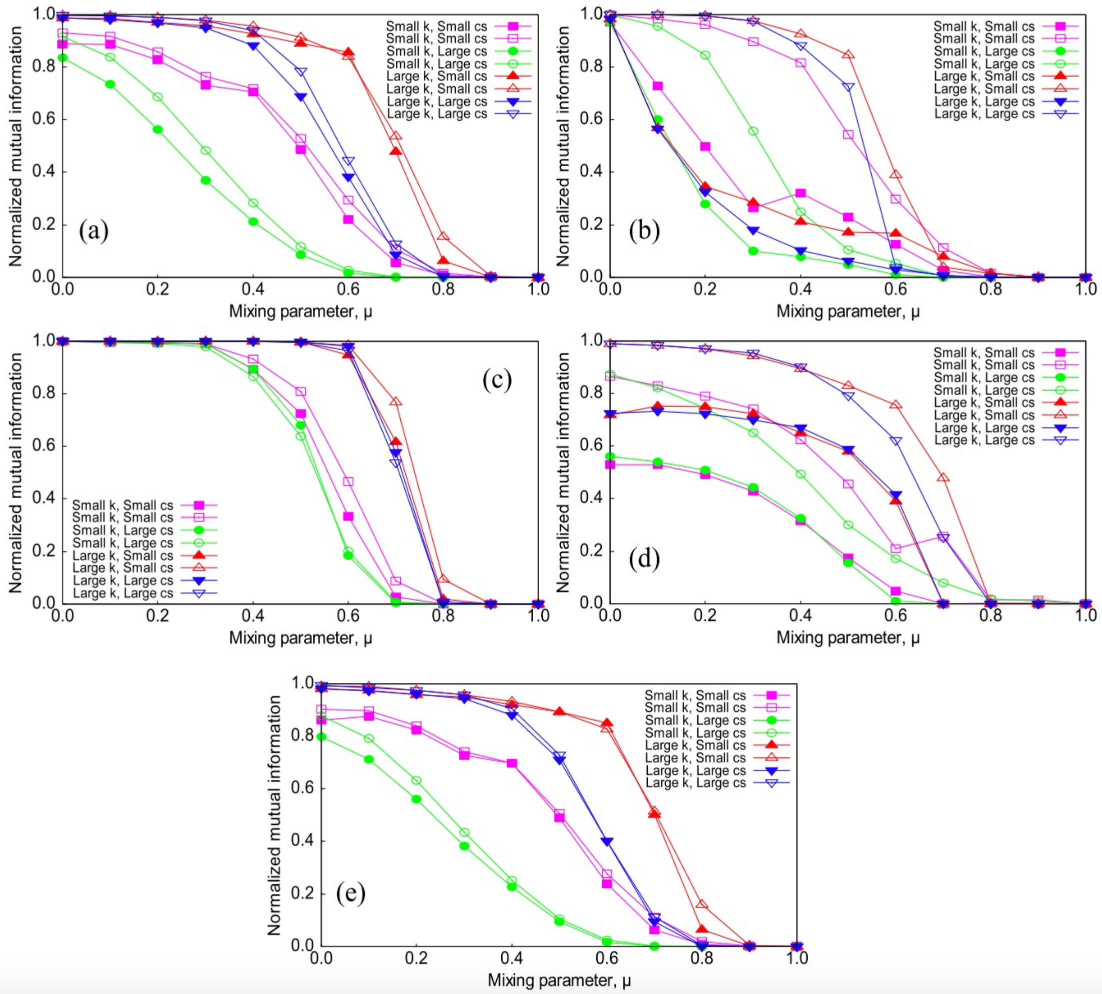
I. <k> =6, k_{max} = 25, c_{min} =5, c_{max} = 10

II. <k> =6, k_{max} = 25, c_{min} =10, c_{max} = 20

III. <k> =10, k_{max} = 25, c_{min} =5, c_{max} = 10

IV. <k> =10, k_{max} = 25, c_{min} =10, c_{max} = 20

- u is the mixing parameter: each vertex shares fraction u of its edges with vertices in other communities



Data Set and Evaluation

2. Real-World Networks:

- Don't know real number of communities
- Modularity for evaluation
- Measures strength of division of network into modules – better partition on the network

Q = $\sum_i (e_{ii} - a_i^2)$ => e_{ij} fraction of edges between communities i and j

=> e_{ii} fraction of edges edges internal to communities

=> $a_i^2 = \sum_j e_{ij}$ fraction of edges that have at least one end vertex in community i

Table 1. Real-world networks.

Networks	Reference	Type	Vertices	Edges
Netscience	[22]	Collaboration	379	914
Email	[23]	Social	1 133	5 451
Scientometrics	[24]	Citation	2 678	10 368
Blogs	[25]	Social	3 982	6 803
Erdös1997	[26]	Collaboration	5 482	8 972
Erdös1998	[26]	Collaboration	5 816	9 505
Erdös1999	[26]	Collaboration	6 094	9 939
Erdös2002	[24]	Collaboration	6 927	11 850
PGP	[27]	Social	10 680	24 316
Cond-mat-2003	[28]	Collaboration	27 519	116 181

Table 2. Real-world networks: modularity.

Network	RFT	RFT +CN	CNM	CNM +CN	Infomap	Infomap +CN	COPRA	COPRA +CN	Louvain	Louvain +CN
Netscience	0.720	0.759	0.837	0.844	0.817	0.822	0.752	0.771	0.721	0.755
Email	0.490	0.517	0.512	0.546	0.534	0.522	0.237	0.489	0.490	0.500
Scientometrics	0.514	0.528	0.540	0.597	0.549	0.540	0.376	0.535	0.597	0.597
Blogs	0.605	0.732	0.850	0.856	0.798	0.803	0.447	0.573	0.594	0.716
Erdös1997	0.569	0.696	0.699	0.726	0.650	0.703	0.063	0.388	0.570	0.682
Erdös1998	0.566	0.701	0.706	0.730	0.653	0.704	0.050	0.387	0.575	0.686
Erdös1999	0.568	0.705	0.699	0.732	0.655	0.706	0.050	0.393	0.571	0.677
Erdös2002	0.544	0.676	0.673	0.691	0.627	0.679	0.081	0.142	0.566	0.673
PGP	0.730	0.781	0.855	0.870	0.814	0.814	0.618	0.681	0.856	0.865
Cond-mat-2003	0.612	0.653	0.676	0.738	0.673	0.677	0.618	0.638	0.723	0.736

Evaluation

- Proposed a simple method for community detection
- Lack of network information, added information by adding weights
- Improved community detection
- What about weighted and bipartite graphs?
- What about overlapping communities

Follow up Papers

1. Identifying robust communities and multi-community nodes by combining top-down and bottom-up approaches to clustering

Chris Gaiteri, Mingming Chen3, Boleslaw Szymanski, Konstantin Kuzmin, Jierui Xie, Changkyu Lee, Timothy Blanche, Elias Chaibub Neto, Su-Chun Huang, Thomas Grabowski, Tara Madhyastha & Vitalina Komashko

2. Adaptive modularity maximization via edge weighting scheme

Xiaoyan Lua, Konstantin Kuzmina, Mingming Chenb, Boleslaw K. Szymanskia

3. Robust link prediction in criminal networks: A case study of the Sicilian Mafia

Francesco Calderoni, Salvatore Catanese, Pasquale De Meo, Annamaria Ficara, Giacomo Fiumara

Identifying Robust Communities and Multi-Community Nodes by Combining Top-down and Bottom-up Approaches to Clustering

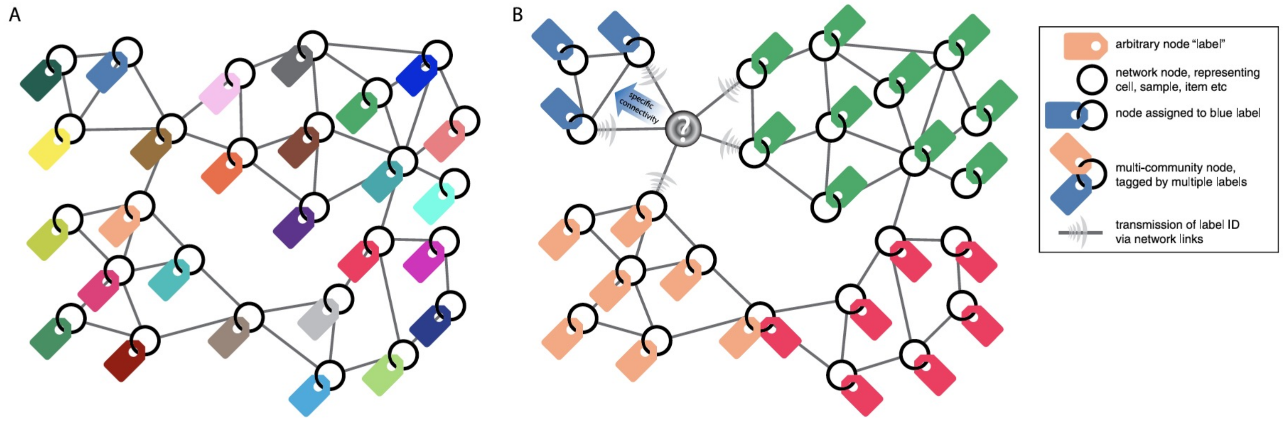
- Communities: interacting molecules, cell or tissues – overlapping communities
- Detected communities are unstable and difficult to replicate
- Previous approaches are sensitive to noise
- SpeakEasy uses both top-down and bottom-up approaches
- Nodes join communities based on local connections and global network structure

Features of SpeakEasy

1. Does not require manual tuning of clustering parameters for good results
2. Can cluster networks with any type of links (weighted/unweighted, directed/undirected, positive/negative-valued edges)
3. Scalable – can cluster networks with thousands of nodes
4. Stochastic clustering – efficient, repeated clustering can identify multi-community nodes

Labeling

- Labels are unique bits of information not some theoretical community title
- Assigned randomly
- Label update: Neighbors' labels - expected frequency i.e label that is most specific to neighbor.



Testing

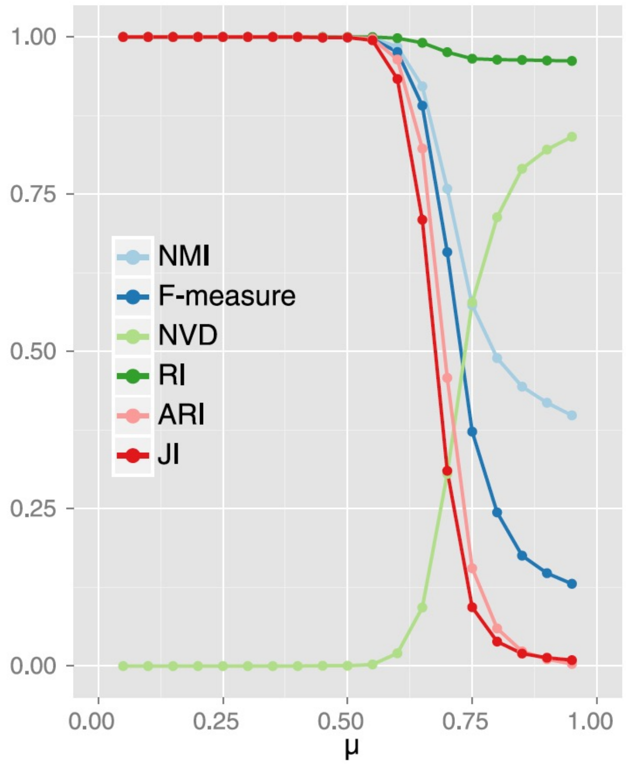
Tested on:

1. Synthetic Networks – LFR benchmarks
2. Real World Networks
3. Biological Networks: Protein–Protein interaction, cell-type clustering, coexpressed gene set, neuronal spike sorting, resting-state fMRI

Synthetic Networks

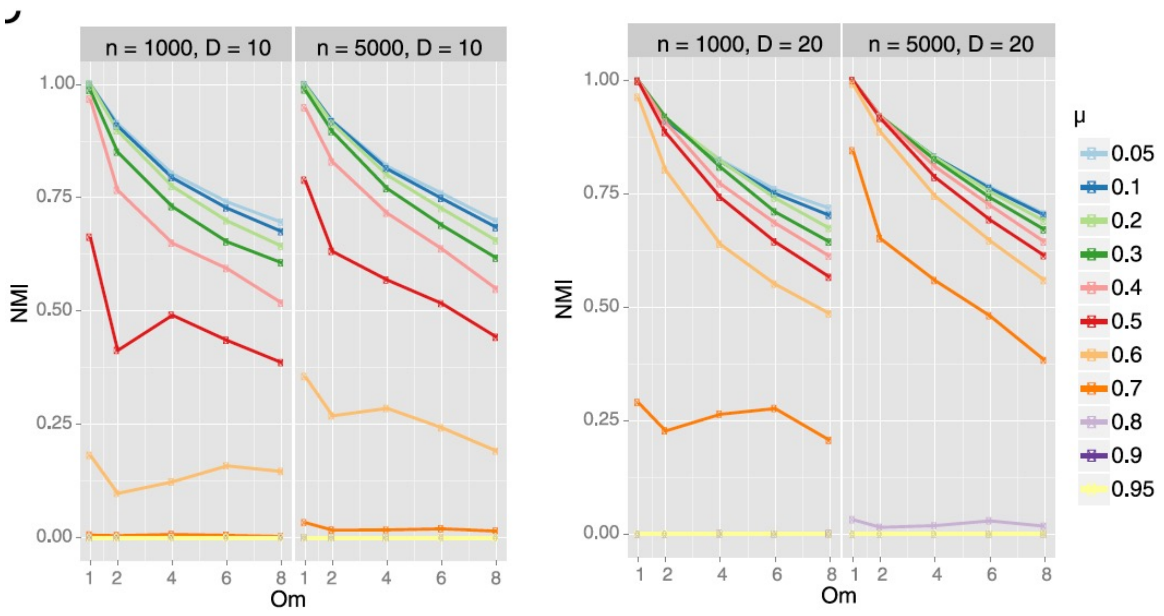
- LFR benchmarks- both well separated and highly cross-linked clusters
- Average of 10 replicate runs vs increasing cross-linking (u)
- Tested using different matric
- Normalized mutual information (NMI) for evaluation
- Lower NMI with increase in u
- Rand Index have varying inputs and sensitivity

A



Synthetic Networks

- Better community detection accuracy for networks with higher average connectivity (D)
- Higher D greater cluster density
- Community detection difficult with higher Om – number of communities that are tied to multi community nodes
- Generally, community detection scores decreases with larger networks
- SpeakEasy performs better – incorporation of global network information (label popularity) into local clustering



Real World Networks

- 15 real-world networks
- Evaluation: modularity (Q) and modularity density (Q_{ds}), how well network is segmented into dense clusters
- SpeakEasy vs GANXis
- Improved performance on 6/15 networks using Q, mean percent difference of over 2%
- Better performance on 14/15 networks using Q_{ds}, over 28% mean percent difference
- Higher Q_{ds} vs Q – SpeakEasy detects more small and highly dense clusters

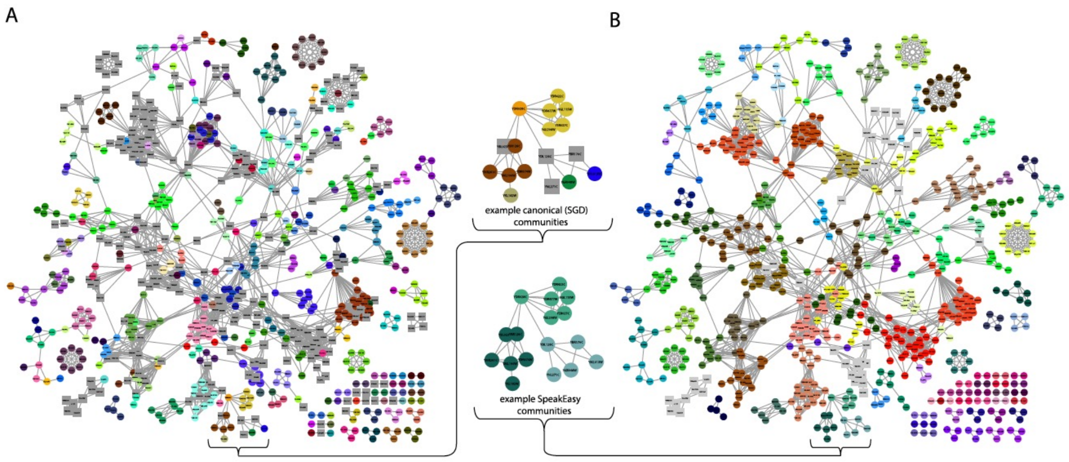
Real World Networks

Network	n	m	GANXiS (Q)	SpeakEasy (Q)	Percentage difference (Q)	GANXiS (Q_{ds})	SpeakEasy (Q_{ds})	Percentage difference (Q_{ds})
karate	34	78	0.3924	**0.4198**	6.75	0.2116	**0.2302**	8.42
dolphins	62	159	0.4408	**0.5017**	12.92	0.1664	**0.2378**	35.33
Les. Mis.	77	254	0.5224	**0.5480**	4.78	0.2808	**0.3438**	20.17
pol. books	105	441	0.4831	**0.4973**	2.90	0.1634	**0.2396**	37.82
football	115	613	**0.5878**	0.5811	−1.15	0.3792	**0.4856**	24.61
Santa Fe	118	200	**0.7166**	0.4792	−39.69	0.2099	**0.2963**	34.13
jazz	198	2742	0.2816	**0.4443**	44.83	0.1917	**0.2134**	10.71
railway	297	1213	**0.6989**	0.6098	−13.61	0.2632	**0.3756**	35.20
c. elegans	453	2525	0.1706	**0.3883**	77.90	0.05151	**0.1079**	70.75
email	1133	5254	**0.5035**	0.4916	−2.39	0.05366	**0.1025**	62.55
pol. blogs	1224	19022	**0.4177**	0.3533	−16.71	0.0230	**0.0426**	59.78
net science	1461	2742	**0.9039**	0.7657	−16.55	**0.5797**	0.3600	−46.76
PGP	10680	24316	**0.8039**	0.7315	−9.43	0.1595	**0.1906**	17.77
DBLP	260998	950059	**0.6622**	0.6066	−8.76	0.2018	**0.2628**	26.29
Amazon	319948	880215	**0.7659**	0.7094	−7.66	0.2007	**0.2556**	24.04

Protein-Protein Interaction

- Single protein may be a part of more than one protein complex - overlapping community
- High throughput protein interaction networks: Gavin et al and Collins et al
- Derived from affinity purification and mass spectrometry (AP-MS)
- Compare with three gold-standards for protein complexes
- NMI score for SpeakEasy better
- Examine precision and recall for detection of

multi-community nodes

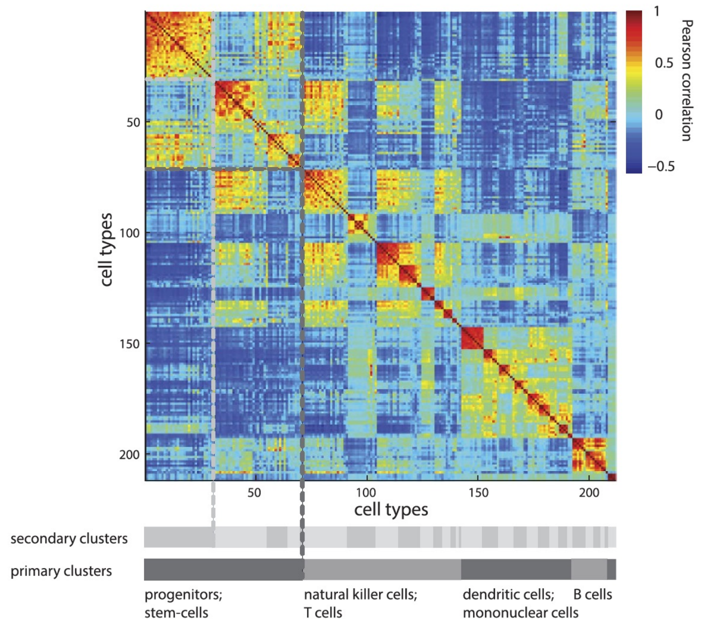


Protein-Protein Interaction

Ground truth definition source	Network dataset	SpeakEasy output type	NMI	Omega	Precision	Recall	F-score	Precision (overlapping)	Recall (overlapping)	F-score (overlapping)
CYC2008	Collins *et al.*	Disjoint	0.7237	0.6382	0.9844	0.8259	0.8982	NA	NA	NA
CYC2008	Collins *et al.*	Overlapping	0.7120	0.5961	0.9845	0.8055	0.8860	0.2151	0.1170	0.1515
CYC2008	Gavin *et al.*	Disjoint	0.4502	0.4530	0.8915	0.5395	0.6722	NA	NA	NA
CYC2008	Gavin *et al.*	Overlapping	0.4498	0.4265	0.8837	0.5292	0.6620	0.2105	0.0958	0.1317
MIPS	Collins *et al.*	Disjoint	0.6669	0.3740	0.9208	0.8701	0.8947	NA	NA	NA
MIPS	Collins *et al.*	Overlapping	0.6665	0.3880	0.9118	0.8588	0.8845	0.7821	0.1227	0.2122
MIPS	Gavin *et al.*	Disjoint	0.5155	0.2001	0.8889	0.7238	0.7979	NA	NA	NA
MIPS	Gavin *et al.*	Overlapping	0.4929	0.2259	0.9053	0.7127	0.7975	0.7143	0.2092	0.3236
SGD	Collins *et al.*	Disjoint	0.7147	0.5652	0.9597	0.7510	0.8426	NA	NA	NA
SGD	Collins *et al.*	Overlapping	0.7058	0.5106	0.9733	0.7470	0.8453	0.4766	0.2048	0.2865
SGD	Gavin *et al.*	Disjoint	0.5474	0.5215	0.9907	0.6255	0.7668	NA	NA	NA
SGD	Gavin *et al.*	Overlapping	0.5460	0.5130	0.9722	0.6175	0.7553	0.3659	0.0652	0.1107

Cell-type Clustering

- Collection of cell population from Immunologic Genomic Project (Immgen) – 212 cell types
- Cells can be distinguished by specific combinations of cell surface markers
- Strong correlation between predicted and tissue organs of cells
- Successive application of SpeakEasy identifies sub-communities better

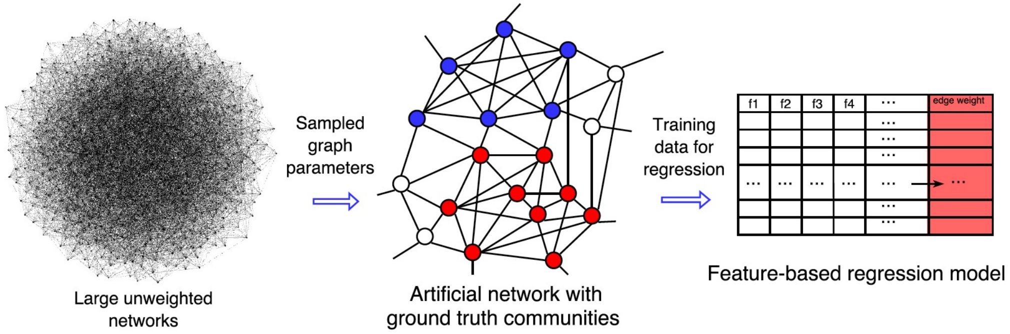


Adaptive Modularity Maximization via Edge Weighting Scheme

- Modularity: difference between observed edges within a community and edges in random graph with same degree and number of nodes
- High modularity indicates quality of a community structure in network
- Modularity Maximization: discovering partition of network which maximizes modularity
- Modularity maximizations prefers large communities over small ones
- Previously edges indicate propensity of nodes for sharing the same community
- Now this refers to edges with +ve weights with -ve weight edges indicate aversion for putting their end nodes in one community

Methodology

1. Artificial Network Construction
   - Estimate network parameters of input graph, construct similar artificial network – ground truth known
   - Goal: these ground truth communities can be successfully separated
2. Edge Feature Extraction
   - Executed on edges of artificial network
   - Edge features used as input to the regression model
3. Regression on edge weights
   - Regression model to compute edge weights
   - Modularity Maximization can separate adjacent truth communities in artificial network



Artificial Network Construction

- Same degree distribution and clustering coefficient
- Multiple Stochastic Block Model (SBM) networks
- Edges of SBM randomly removed until average node degree becomes close to input graph
- SBM with closest average clustering coefficient to input graph is chosen
- Ground truth communities (red and blue) used to train regression model

Extracting Edge Features

- Communities = local structures hence local topological features of every edge

F-1: Square Root of number of common neighbors $\sqrt{N(u) \cap N(v)}$

F-2: Difference in clustering coefficients of endpoints $|c(u) - c(v)|$

F-3: Jaccard-coefficient: $\frac{N(u) \cap N(v)}{N(u) \cup N(v)}$

F-4: Resource Allocation Index: $\frac{1}{|N(u) \cap N(v)|}$

F-5: Adamic Adar Index: $\frac{1}{\log |N(u) \cap N(v)|}$

F-6: Relative degree ration: $\frac{\min(|N(u),| \ |N(v)|)}{max(|N(u),| \ |N(v)|)}$

Regression Model

- Regression model to convert unweighted graph to weighted
- Model takes local topological features of edges as input and outputs weights of each edge
- Decrease modularity to handle resolution limit problem
- Regularization on average edge weight results in small weights that are inconvenient in community detection
- Regularization on variance of edge weights limits total number of negative edges

Testing

Model is tested on:

1. LFR Benchmarks
2. American College Football Network
3. Large Networks

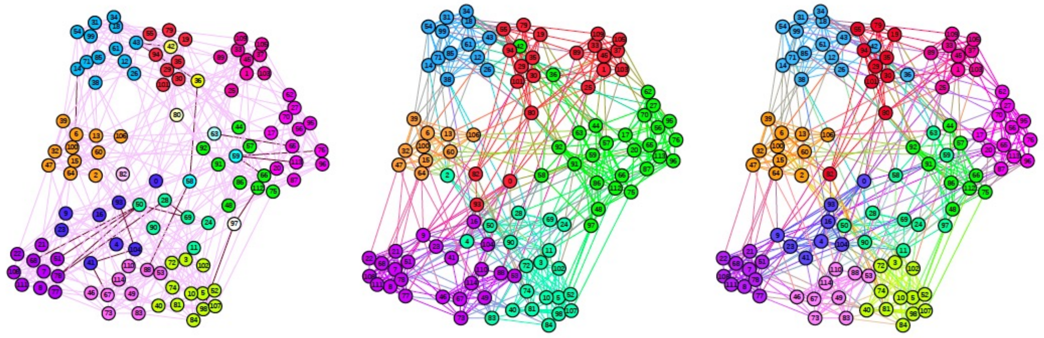
No.	Network	#Nodes	#Edges	Type
1	American college football	115	613	Real
2	LFR benchmark	5000	≈35000	Synthetic
3	Amazon product co-purchasing network	334863	925872	Real
4	DBLP collaboration network	317080	1049866	Real

LFR Benchmarks

μ	Method	VI	NMI	F-measure	ARI	Q	Q_{ds}
0.45	FG	3.2135	0.5953	0.3379	0.2355	0.4214	0.0366
	Fine-Tuned Q_{ds}	1.1523	0.8925	0.8806	0.7337	0.4536	0.1632
	FG-w	0.0137	0.9987	0.9990	0.9972	0.5152	0.1668
0.5	FG	3.5187	0.5481	0.2937	0.1993	0.3739	0.0274
	Fine-Tuned Q_{ds}	1.9677	0.8036	0.7489	0.4984	0.3563	0.1196
	FG-w	0.0678	0.9934	0.9950	0.9864	0.4625	0.1381

American College Football Network

- 115 nodes – team, 11 conferences, link if two teams played with each other in 2000 season
- Teams in each 11 conferences – community, 8 independent teams – single community (a)
- 19 ground truth communities, Fast Greedy on unweighted detects 6 communities(b), on weighted detects 11 communities (c)

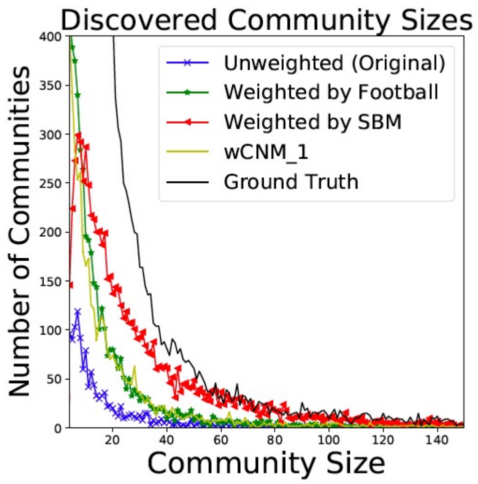


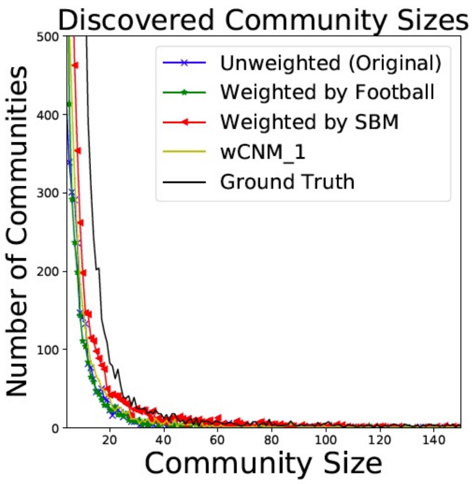
(a) (b) (c)

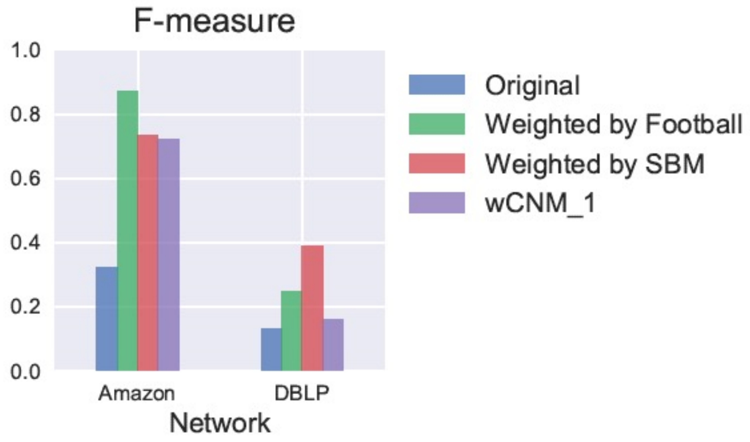
Metric	Graph	FG	LE	LP	RW	ML
NMI	Original	0.58528	0.58140	0.76962	0.83833	0.83391
	Weighted	0.91117	0.85903	0.92635	0.91117	0.87272
ARI	Original	0.49333	0.49441	0.71749	0.86938	0.85815
	Weighted	0.94723	0.88982	0.91539	0.94723	0.90085
Q	Original	0.56860	0.49326	0.57668	0.60337	0.60503
	Weighted	0.60140	0.59338	0.57315	0.60140	0.60356
Q_{ds}	Original	0.15877	0.13661	0.21106	0.23650	0.23626
	Weighted	0.25696	0.23893	0.24025	0.25696	0.24889

Large Networks

1. Amazon co-purchasing – 334,863 products, link between 2 frequently co-purchased items, each collection of products from same category forms a ground truth community
2. DBLP collaboration network – co-authorship network. Node: researcher, link between two researchers who published a paper together. Publication venue – ground truth community







Robust link prediction in criminal networks: A case study of the Sicilian Mafia

- Link prediction is challenging with noisy and incomplete networks such as criminal networks
- Link prediction: predict edges which are more likely to appear in future, discover missing edges
- Tackle the problem of estimating robustness of link prediction algo in criminal networks
- Accuracy should be integrated with stability (robustness) – extent with which the insertion of unobserved edges modifies the network
- Simulations to investigate how these algorithms perform on noisy and incomplete networks
- Likelihood to decide if an edge exists in actual network

Data

- Dataset manually extracted from judicial documents of operation "Montagna" – Sicilian Mafia
- Extracted 2 graphs:

1. Meeting Graph G_M – meeting between person under investigation
2. Phone Call Graph G_P - monitoring phone calls

- Node: an individual that belongs to mafia group
- Edge: interaction between two individuals

Statistics of G_M and G_P graphs.

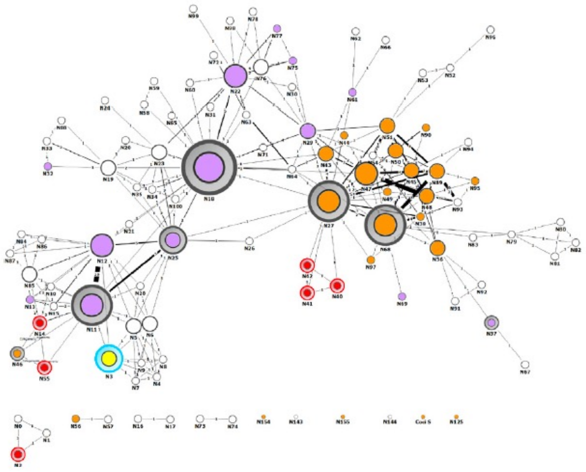
Parameter	G_M	G_P
Number of Nodes $\|N\|$	101	100
Number of Edges $\|E\|$	256	124
Average Degree $\overline{d}$	5.07	2.48
Nensity δ	0.051	0.025
Average Clustering Coefficient ac	0.656	0.105

Methodology

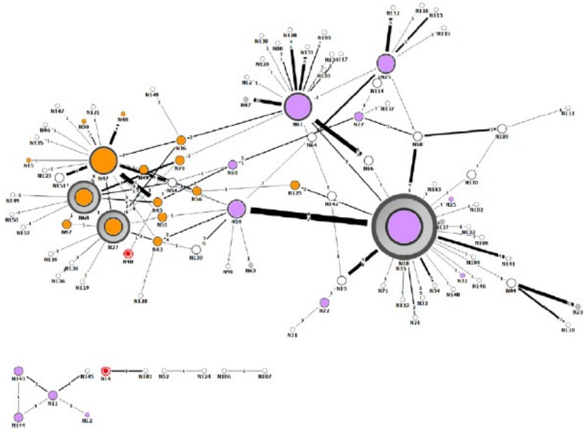
- GM (sample of true graph GM’) and GP (sample of true graph GM’) are not completely known
- Scoring function (likelihood) decide whether edge non observed in GM actually exists in GM’
- Simulation to create GM’ and GP’

Link Prediction

- G’ is subset of G, contains all nodes in G, subset $E' \sqsubseteq E$
- Goal: getting list of non-edges in G’ which are edges in G
- Link prediction algorithms needs degree of similarity between nodes i and j
- All pair of non edges <i,j> in G are ranked in decreasing order of similarity, largest similarity edge is most likely to exist



GM



GP

Node Similarity – Local Methods

- Require knowledge of neighbors of two nodes i and j
- Jaccard Coefficient, Common Neighbors, Preferential Attachment, Adamic-Adar Coefficient
- But GM and GP are highly sparse – edge prediction hard using local methods
- High clustering coefficient – if two nodes I and j share at least one neighbors, then there is a high chance that i and j will be linked by an edge too. Quantifies to which extent the neighbors of the node i tend to form a tie
- Need measure which uses full knowledge of graph topology

Node Similarity – Global Methods

- Katz score k(i,j)
  - Considers whole ensemble of walks connecting I and j
  - Each walk provides contribution to determine degree of similarity between I and j
  - Long walks penalized over shorter ones
  - i.e nodes are highly similar if connected by many short walks in graph G
- If similarity score greater than certain threshold keep the edge

Likelihood

- Investigate the robustness of link prediction algorithm on uncertain networks
- Starting point: observed networks
- Add missing edges and find difference in Katz score

Questions?