

Node Similarity and Structural Roles

How the Class Fits Together

Properties

Small diameter,
Edge clustering

Scale-free

Strength of weak ties,
Core-periphery

Densification power law,
Shrinking diameters

Complex Graph Structure

Information virality,
Memetracking

Models

Small-world model,
Erdős-Renyi model

Preferential attachment,
Copying model

Kronecker Graphs

Microscopic model of
evolving networks

Graph Neural Networks

Independent cascade model,
Game theoretic model

Algorithms

Decentralized search

PageRank, Hubs and
authorities

Community detection:
Girvan-Newman, Modularity

Link prediction,
Supervised random walks

Node Classification
Graph Representation Learning

Influence maximization,
Outbreak detection, LIM

Node Equivalence, Structural Roles and Assortative Mixing

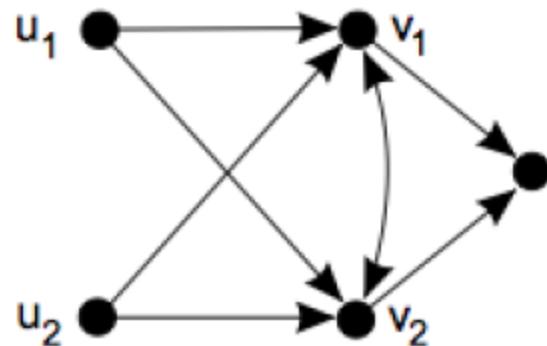
- ***Node equivalence***
 - Structural equivalence
 - Regular equivalence
- ***Node similarity***
 - Jaccard Similarity
 - Cosine Similarity
 - Pearson correlation
- ***Structural Roles***
 - Structural similarity
 - Role generalization and transfer learning
 - Making sense of roles
- ***Assortative mixing***
 - Mixing by value
 - Degree correlation

Patterns of relations

- ***Global, statistical properties of the networks***
 - average node degree (degree distribution)
 - average clustering
 - average (shortest) path length
- ***Local, per vertex properties***
 - node centrality
 - page rank
- ***Pairwise properties***
 - node equivalence
 - node similarity
 - correlation between pairs of vertices (node values)

Structural Equivalence

- ***Structural equivalence:*** two vertices are structurally equivalent if their respective sets of in-neighbors and out-neighbors are the same

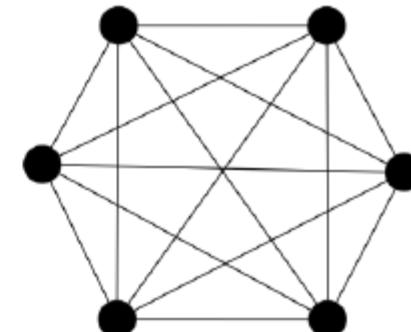
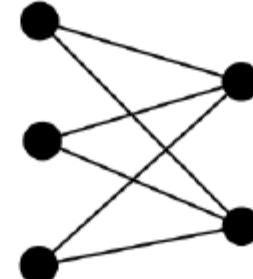
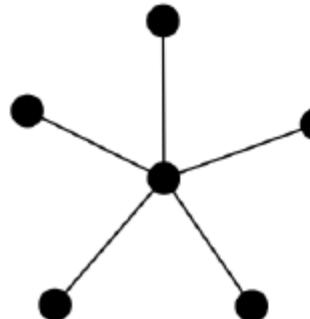


	u1	u2	v1	v2	w
u1	0	0	1	1	0
u2	0	0	1	1	0
v1	0	0	0	1	1
v2	0	0	1	0	1
w	0	0	0	0	0

rows and columns of adjacency matrix of structurally equivalent nodes are identical, “connect to the same neighbors”

Structural Equivalence

- In order for adjacent vertices to be structurally equivalent, they might have self loops
- Sometimes called “***strong structural equivalence***”
 - Sometimes ***relax requirements*** for self loops for adjacent nodes



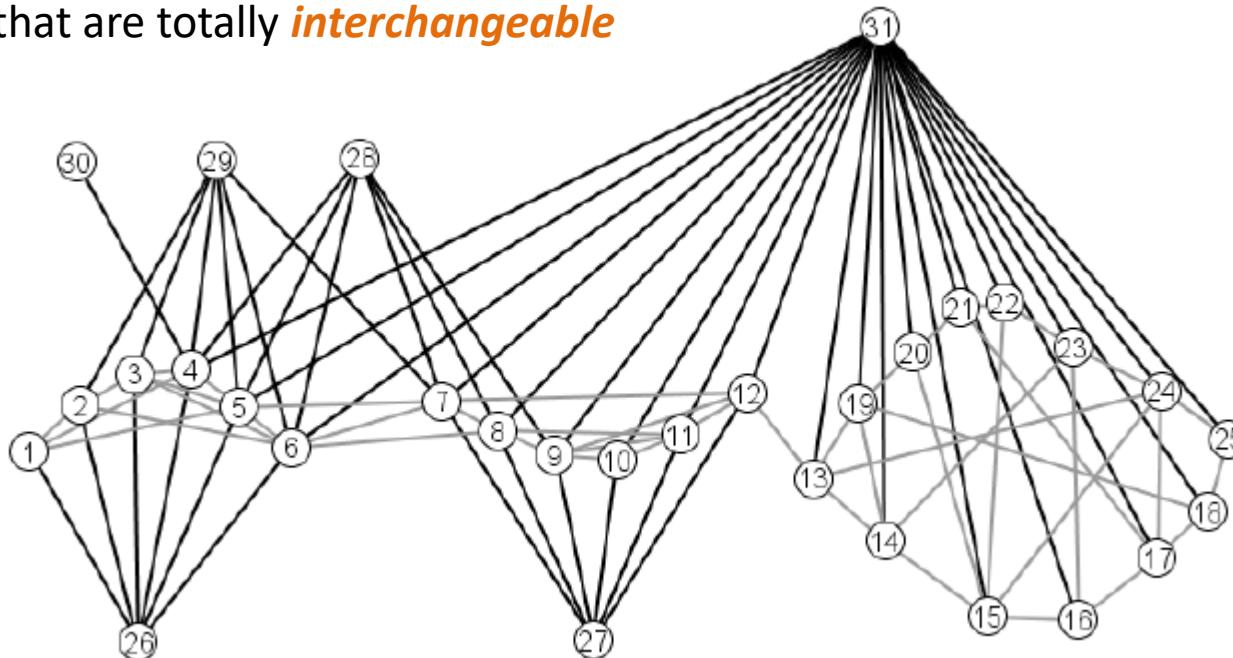
Similarity Measures

- *Jaccard similarity*

$$J(v_i, v_j) = \frac{|\mathcal{N}(v_i) \cap \mathcal{N}(v_j)|}{|\mathcal{N}(v_i) \cup \mathcal{N}(v_j)|}$$

SE comes from sociology

structurally equivalent nodes are the people that have positions that are totally **interchangeable**



Similarity Measures

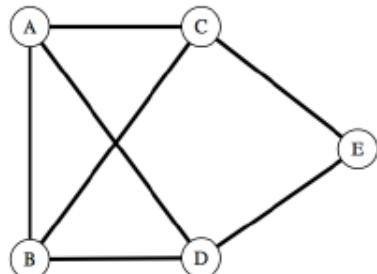
- **Cosine similarity** (vectors in n -dim space)

$$\sigma(v_i, v_j) = \cos(\theta_{ij}) = \frac{\mathbf{v}_i^T \mathbf{v}_j}{\|\mathbf{v}_i\| \|\mathbf{v}_j\|} = \frac{\sum_k A_{ik} A_{kj}}{\sqrt{\sum A_{ik}^2} \sqrt{\sum A_{jk}^2}}$$

- **Pearson correlation coefficient**

think of each of the 2 nodes
as 2 random variables
(adjacency vectors)

$$r_{ij} = \frac{\sum_k (A_{ik} - \langle A_i \rangle)(A_{jk} - \langle A_j \rangle)}{\sqrt{\sum_k (A_{ik} - \langle A_i \rangle)^2} \sqrt{\sum_k (A_{jk} - \langle A_j \rangle)^2}}$$



0	1	0	1	1
1	0	1	0	1
0	1	0	1	0
1	0	1	0	1
1	1	0	1	0

Adjacency Matrix \mathbf{A}_{ij}

Similarity Measures

- **Unweighted undirected graph** $A_{ik} = A_{ki}$, binary matrix, only 0 and 1 (can simplify the formula)

$$k_i = \sum_k A_{ik} = \sum_k A_{ik}^2 - \text{node degree}$$

$$n_{ij} = \sum_k A_{ik} A_{kj} = (A^2)_{ij} - \text{number of shared neighbors}$$

$$\langle A_i \rangle = \frac{1}{n} \sum_k A_{ik}$$

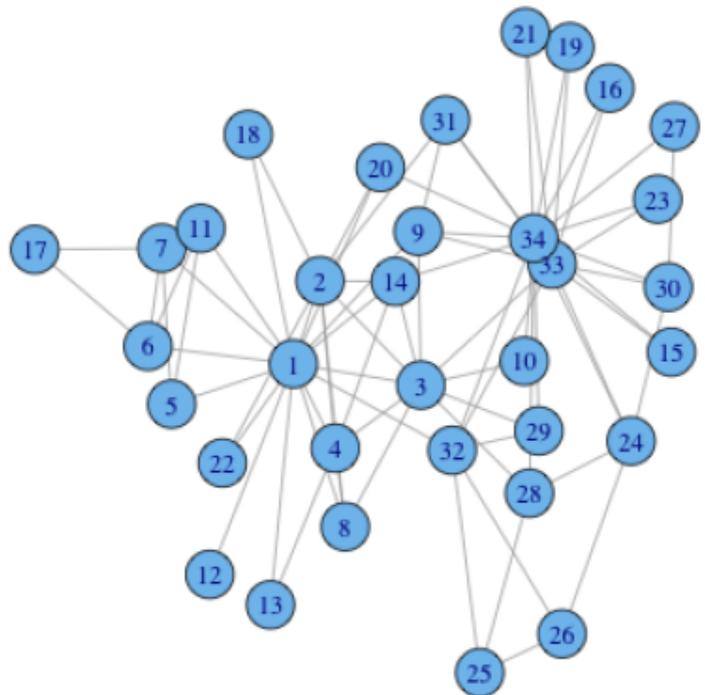
- **Cosine similarity** (vectors in n -dim space)

$$\sigma(v_i, v_j) = \cos(\theta_{ij}) = \frac{n_{ij}}{\sqrt{k_i k_j}}$$

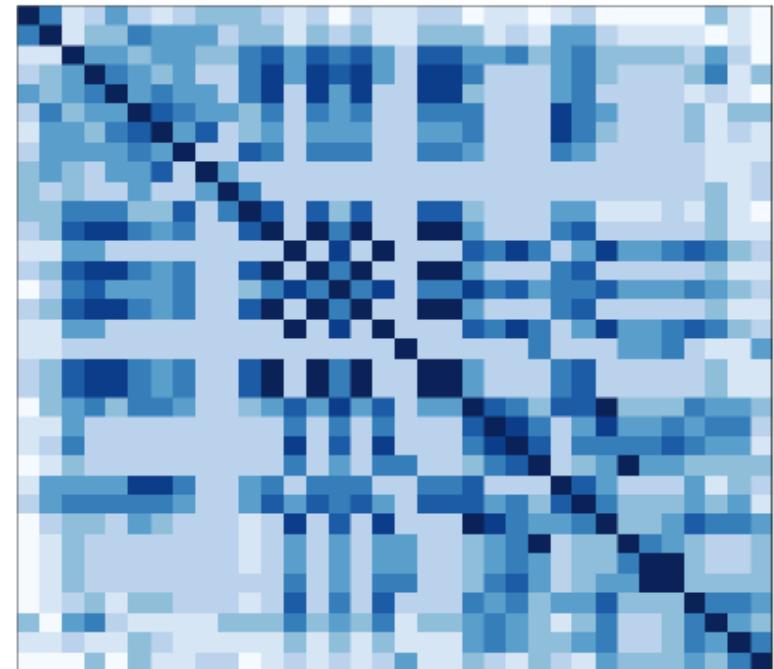
- **Pearson correlation coefficient:**

$$r_{ij} = \frac{n_{ij} - \frac{k_i k_j}{n}}{\sqrt{k_i - \frac{k_i^2}{n}} \sqrt{k_j - \frac{k_j^2}{n}}}$$

Similarity Matrix



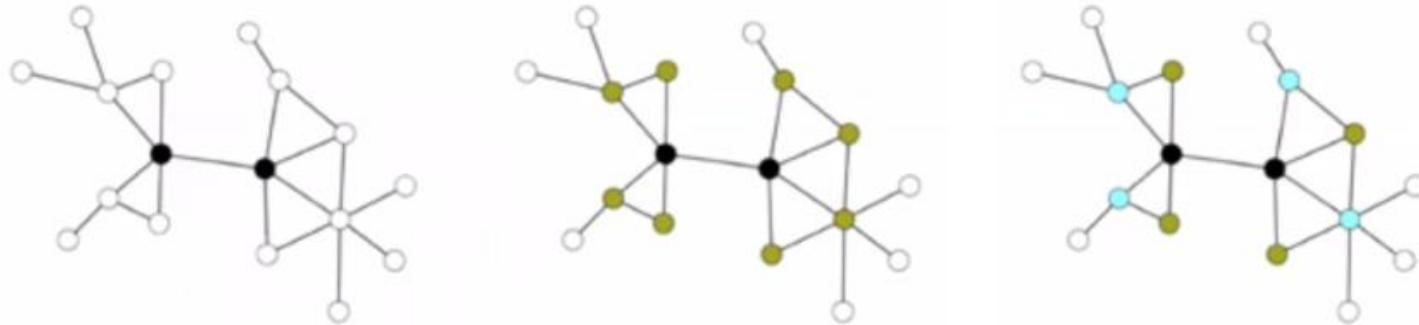
Graph



Node similarity matrix

Regular Equivalence

- Regular equivalence: two vertices are regularly equivalent if they are equally related to equivalent others

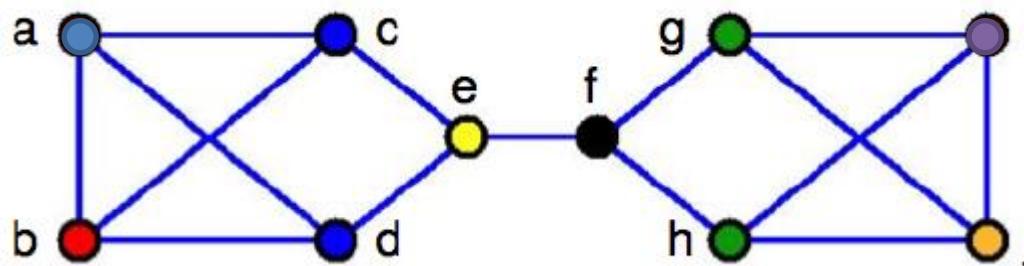


- Equivalent definition in terms of *role assignment (coloring)*: vertices that are colored the same, have the same colors of their neighborhoods

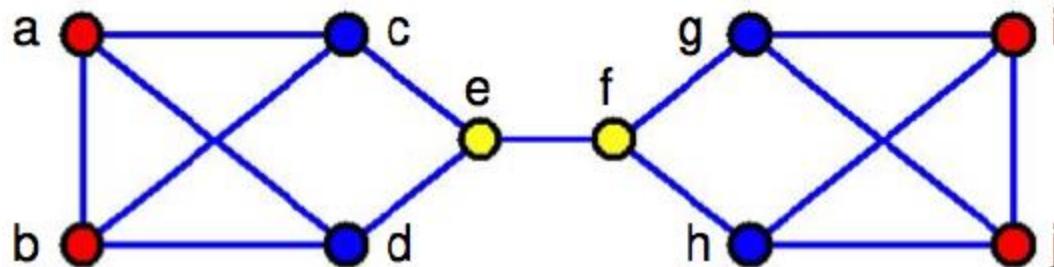
White and Reitz, 1983; Everette and Borgatti, 1991

Equivalence example

- *structural equivalence*



- *regular equivalence*



e and f play similar roles

Regular Equivalence

- **Recursive definition:** two vertices are regularly equivalent if they are equally related to equivalent others.
 - Quantitative measure of regular equivalence, σ_{ij} - similarity score

$$\sigma_{ij} = \alpha \sum_{k,l} A_{ik} A_{jl} \sigma_{kl}$$

$$\sigma = \alpha \mathbf{A} \sigma \mathbf{A}$$

- should have high σ_{ii} - self similarity

- σ_{ij} is a metric for **pair of nodes** (not for a single node like pagerank or katz centrality)
- α is used for normalization
- There are **two steps** here A_{ik} and A_{jl}

$$\sigma_{ij} = \alpha \sum_{k,l} A_{ik} A_{jl} \sigma_{kl} + \delta_{ij}$$

$$\sigma = \alpha \mathbf{A} \sigma \mathbf{A} + \mathbf{I}$$

$$\delta_{ij} = \begin{cases} 1, & \text{if } i=j \\ 0, & \text{if } i \neq j \end{cases}$$

Regular Equivalence - Revision

- A vertex j is similar to vertex i (dashed line) if i has a network neighbor v (solid line) that is itself similar to j

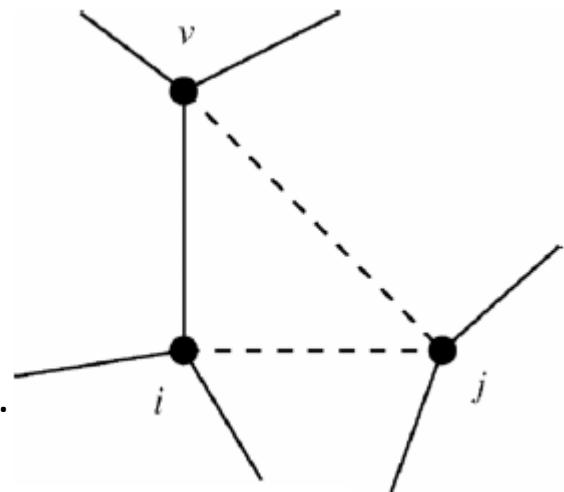
$$\sigma_{ij} = \alpha \sum_v A_{iv} \sigma_{vj} + \delta_{ij}$$

$$\boldsymbol{\sigma} = \alpha \mathbf{A} \boldsymbol{\sigma} + \mathbf{I}$$

- Closed form solution:

$$\boldsymbol{\sigma} = (\mathbf{I} - \alpha \mathbf{A})^{-1}$$

All the paths of length 1, 2, 3, ... participate in the definition.
On the previous definition only even numbered paths were included.



$$\mathbf{S} = \mathbf{I} + \phi \mathbf{A} + \phi^2 \mathbf{A}^2 + \dots$$

element $[\mathbf{A}^l]_{ij}$ is equal to the number of (possibly self-intersecting) network paths of length l from i to j .

SimRank

- Find nodes (*groups of nodes*) that *play the same/similar roles in the network*
- Values may be used as *features to predict links btw nodes*

- $s(a, b)$ – *similarity between a and b*
- $I()$ – set of *in-neighbours* (constant $c \in (0, 1)$)

$$s(a, b) = \begin{cases} 1, & a = b, \\ 0, & \text{if } I(a) = \emptyset \text{ or } I(b) = \emptyset, \\ \frac{c}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)), & \text{otherwise} \end{cases}$$

- Matrix notation:

- Previous models are based on *structural ideas* like *Katz centrality*
- This is in some way similar to *PageRank* (talking about *random walk*)

$$S_{ij} = \frac{C}{k_i k_j} \sum_{k,m} A_{ki} A_{mj} S_{km}$$

Solving SimRank

- One can find a ***solution of the system*** defined by SimRank by using the following ***iterative process***:

$$R_0(a, b) = \begin{cases} 1, & a = b, \\ 0, & \text{otherwise,} \end{cases}$$
$$R_{k+1}(a, b) = \begin{cases} 1, & a = b, \\ 0, & \text{if } I(a) = \emptyset \text{ or } I(b) = \emptyset, \\ \frac{c}{|I(a)||I(b)|} \sum_{v \in I(a)} \sum_{w \in I(b)} R_k(w, v), & \text{otherwise.} \end{cases}$$

- It is shown in Jeh and Widom that ***$R_k(a, b)$ converges to $s(a, b)$***

Example from SimRank Article

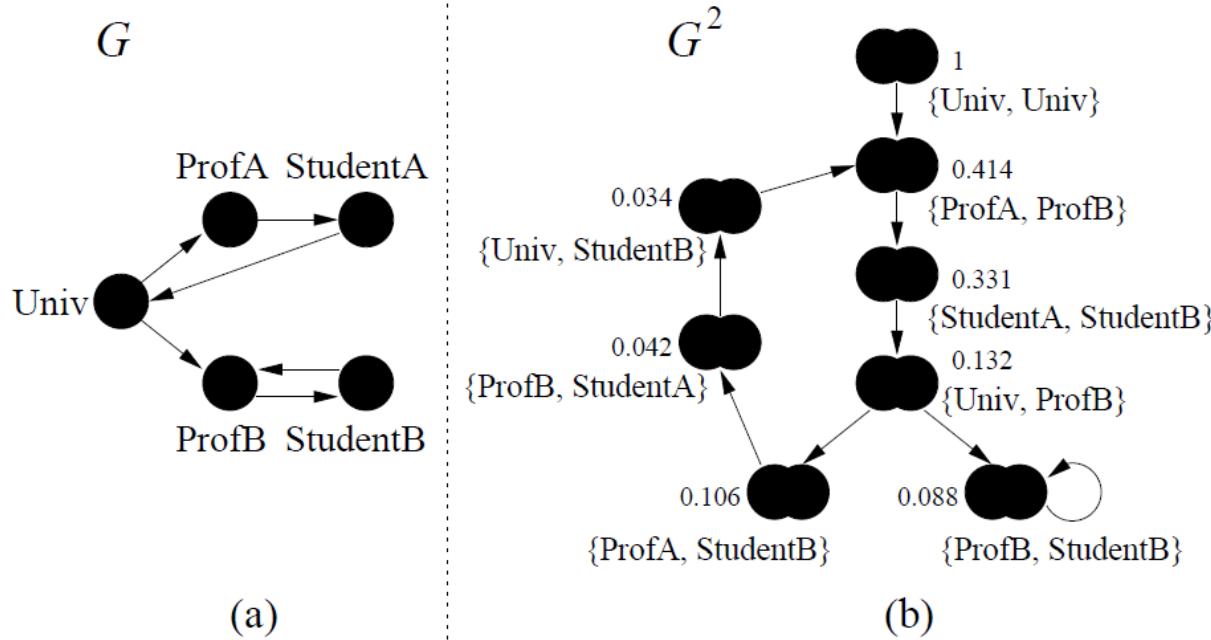


Figure 1: A small Web graph G and simplified node-pairs graph G^2 . SimRank scores using parameter $C = 0.8$ are shown for nodes in G^2 .

6.1 Experiment with Wikipedia

We used Simple English Wikipedia corpus to find semantic relatedness between concepts. We had 150495 entities(order of our matrix) which was a slightly higher than the number of articles in the given wiki at the time of writing, because some of those entities are redirection stubs. We used undirected graph representation of inter-wiki links which gave us 4454023 non zero elements in adjacency matrix. For graph that large direct computation of SimRank is infeasible. We used the following parameters for the experiment: $c = 0.3$, rank = 6000, no-oversampling and did ten iterations. While original paper [4] suggests $c = 0.8$ later it was suggested to use $c = 0.6$ for better results [6] and we choose $c = 0.3$ because it gave us subjectively better results. In experiment we used virtual server (VZ container) with 16 CPU cores and 100GiB RAM available (host node has 32 cores: 4 CPUs, each is 8-core AMD Opteron Processor 6272, 128GiB RAM). With this setup computations took roughly 40 hours.

Some examples provided in the table below. The first row is the word for which most similar words were queried, then in each of the columns most similar words are listed ordered by their SimRank score. The scores themselves would take too much space (they differ in 4-th or 5-th significant figures) and hence are omitted.

GNU	Earth (planet)	Liquid
Richard Matthew Stallman	Planet earth	Plasma (matter)
Linux operating system	SOL III	Matters
*nix	Geomagnetic	Particle theory of matter
Debian linux	Kola superdeep borehole	Potable water
Linux (kernel)	Oblate	Watery
Hurd	Guns, Germs, and Steel	Dihydrogen monoxide
Kernel (Computer science)	Ganges Plain	Hematological

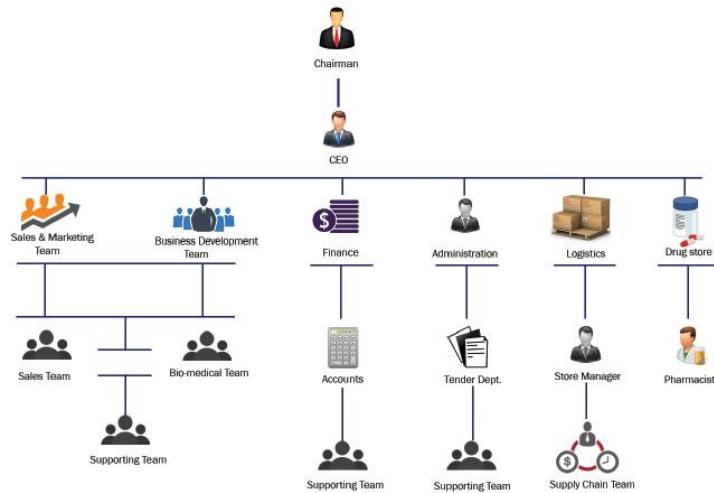
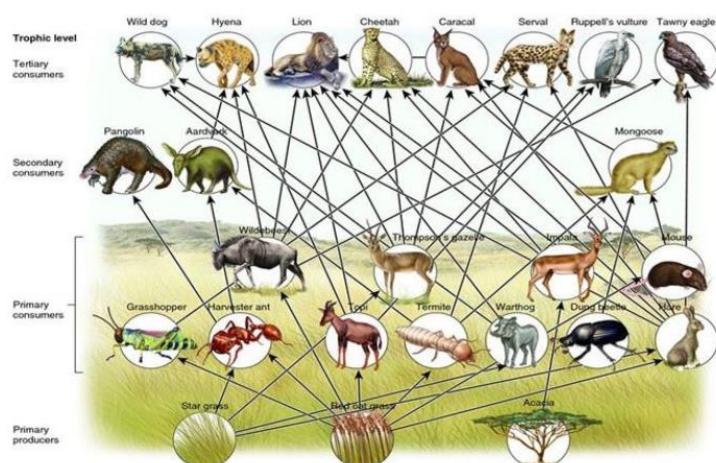
Node Similarities

- Finding of the similar network nodes are important since we many want to
 - find ***nodes*** that ***play the same role*** in the network
 - find ***similar groups of nodes*** that ***play similar roles***
- Node similarity values may also used as features to
 - ***predict links*** between nodes
 - ***cluster objects***, e.g., for ***collaborative filtering in a recommender system***
 - in which “similar” users and items are grouped based on the users’ preferences
 - ***extract the communities*** formed within the network
 - ***“find-similar-document” query*** on traditional text corpora or the World-Wide Web

Structural Roles in Networks

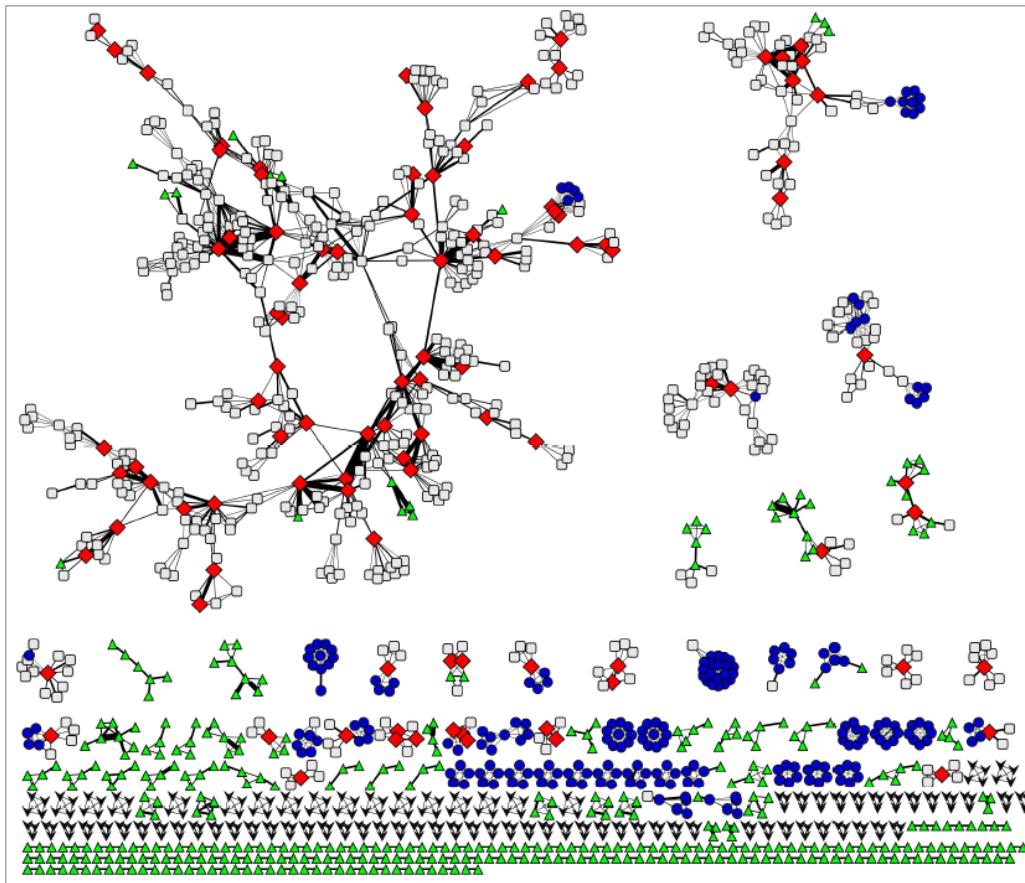
What are Roles?

- *Roles are “functions” of nodes in a network:*
 - Roles of *species* in *ecosystems*
 - Roles of *individuals* in *companies*



- Roles are measured by *structural behaviors*:
 - centers of stars
 - members of cliques
 - peripheral nodes
 - ...

Example of Roles



*Network Science Co-authorship Graph
[Newman 2006]*

Roles vs. Groups in Networks

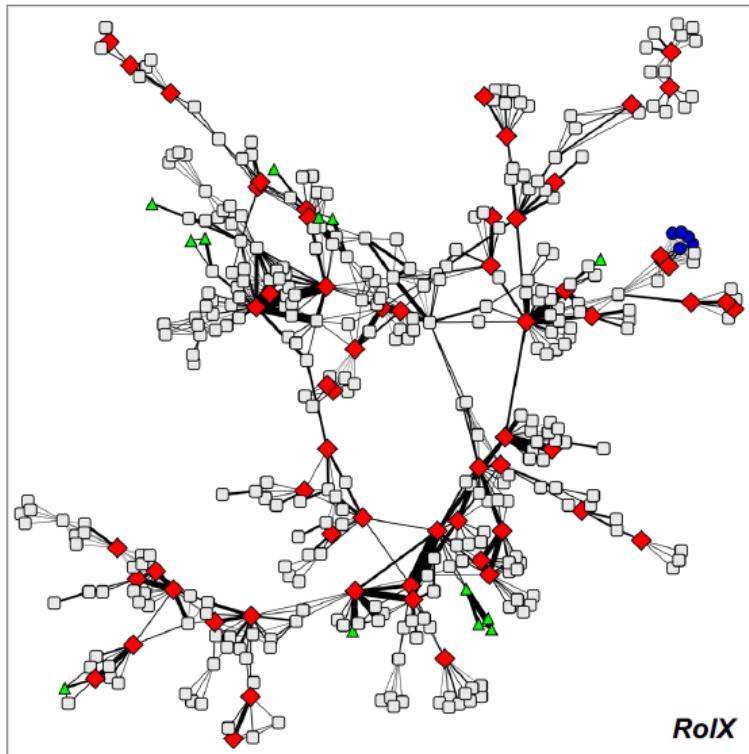
- ***Role:*** A collection of nodes which have similar positions in a network
 - Roles are based on the similarity of ties between subsets of nodes
 - Different from ***groups/communities***
 - Group is formed based on adjacency, proximity or reachability
 - typically adopted in current data mining
- ***Nodes with the same role need not be in direct, or even indirect interaction with each other***

Roles vs. Groups in Networks

- **Roles:**
 - A group of nodes with similar structural properties
- **Communities/Groups:**
 - A group of nodes that are well-connected to each other
- Roles and communities are **complementary**
- Consider the social network of a CS Dept.:
 - **Roles:** Faculty, Staff, Students
 - **Communities:** AI Lab, CyberSec. Lab, Robotics Lab

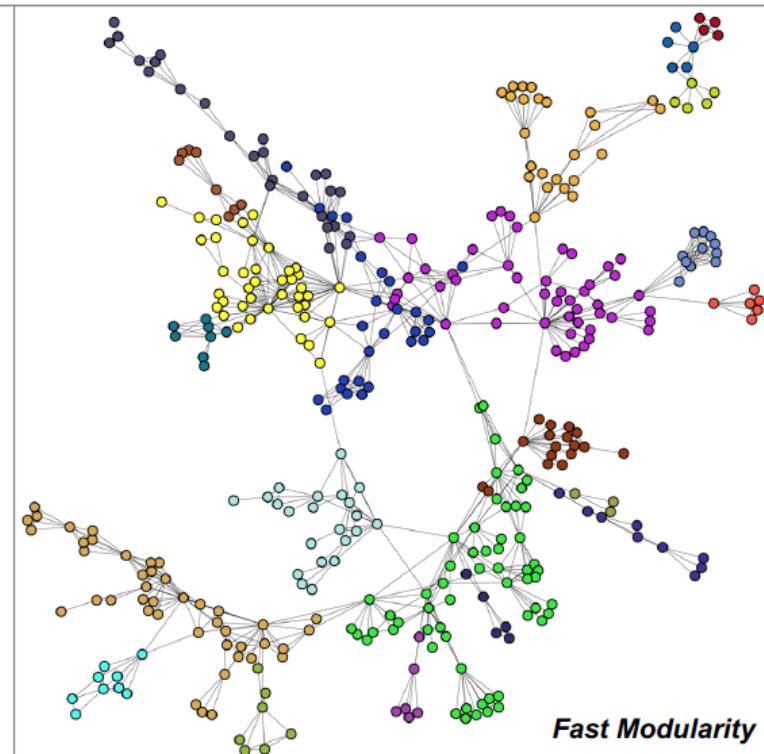
Roles vs. Groups in Networks

Roles



Henderson, et al., KDD 2012

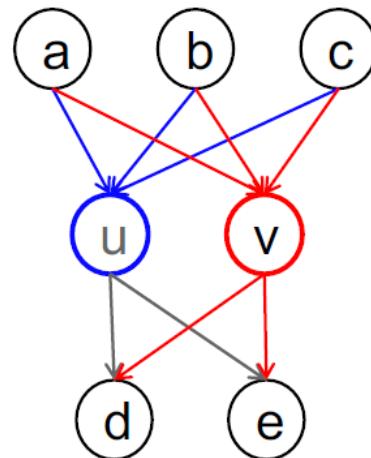
Communities



Clauset, et al., Phys. Rev. E 2004

Roles: Node Equivalence (More Formally)

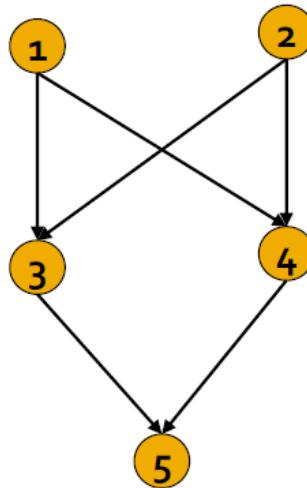
- ***Structural equivalence***: Nodes u and v are structurally equivalent if they have the same relationships ***to all other nodes*** [Lorrain & White 1971]
 - Structurally equivalent nodes are likely to be similar in other ways - i.e., friendships in social networks



Structural Equivalence

- Nodes u and v are **structurally equivalent**:
 - for all the other nodes k , node u has tie to k iff node v has tie to k

Example:



Adjacency matrix

	1	2	3	4	5
1	-	0	1	1	0
2	0	-	1	1	0
3	0	0	-	0	1
4	0	0	0	-	1
5	0	0	0	0	-

* nodes **1 and 2, 3 and 4** are structurally equivalent

Discovering Structural Roles and its applications

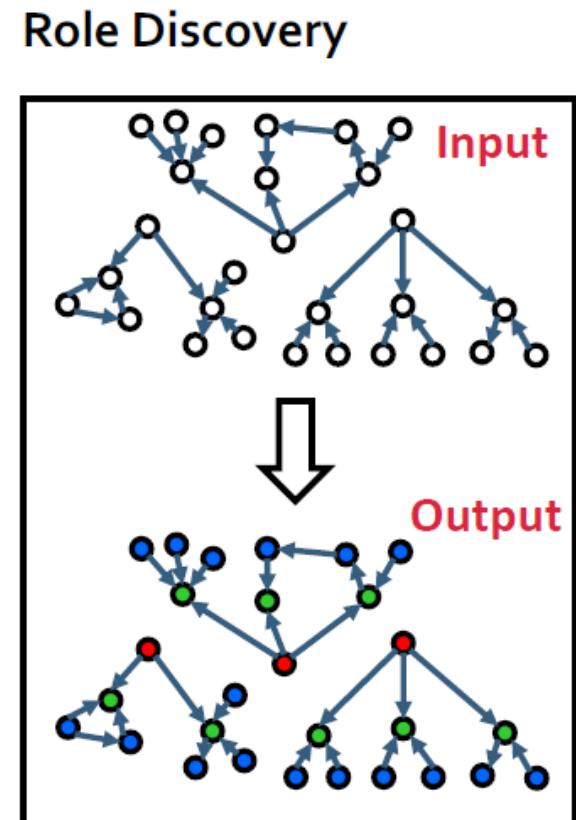
- Structural similarity
- Role generalization and transfer learning
- Making sense of roles

Importance of the Roles

Task	Example Application
Role query	Identify individuals with similar behavior to a known target
Role outliers	Identify individuals with unusual behavior
Role dynamics	Identify unusual changes in behavior
Identity resolution	Identify, de-anonymize, individuals in a new network
Role transfer	Use knowledge of one network to make predictions in another
Network comparison	Compute similarity of networks, determine compatibility for knowledge transfer

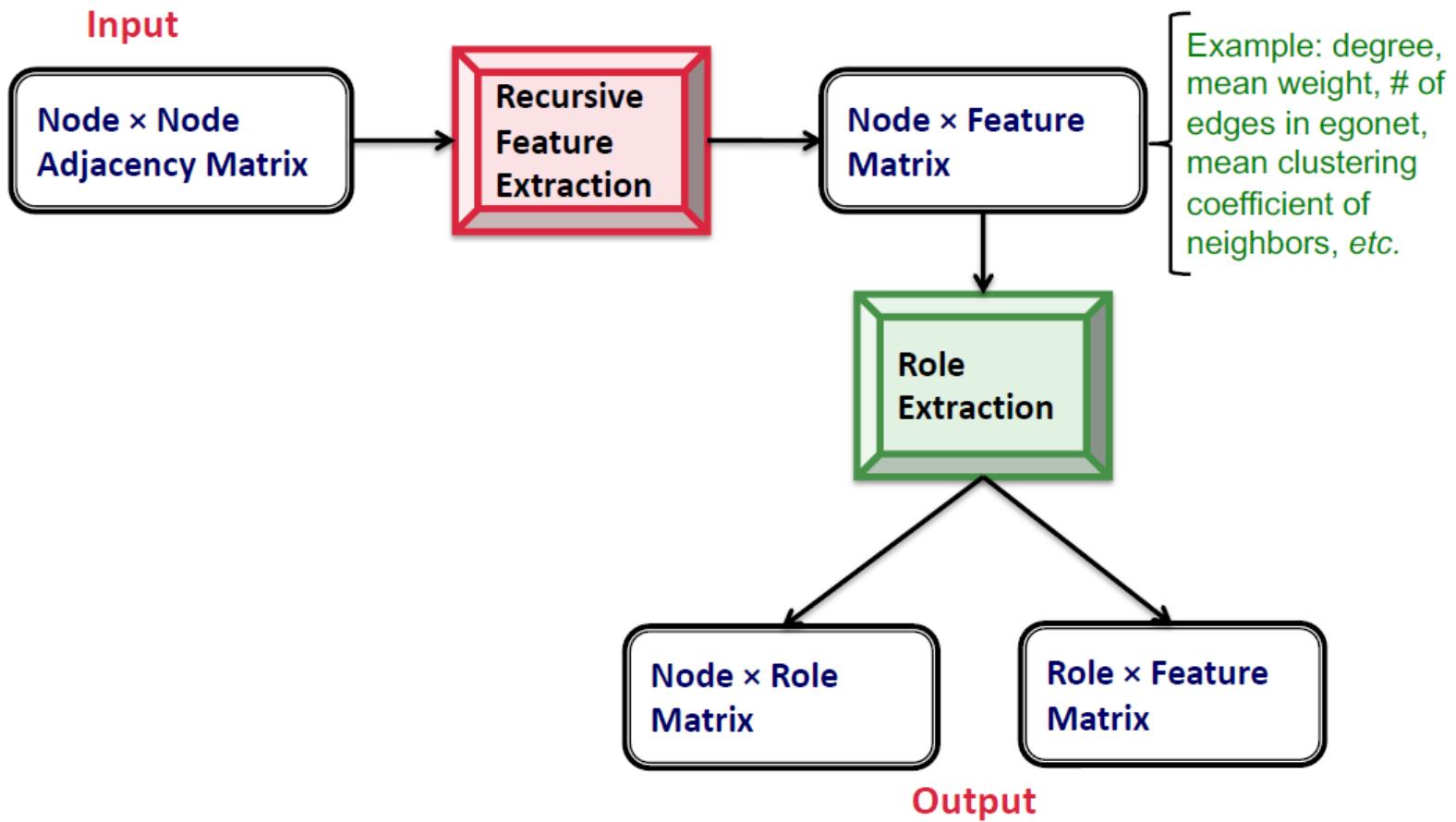
Structural Role Discovery Method

- ***RoIX***: Automatic discovery of nodes' structural roles in networks [[Henderson, et al. 2011b](#)]
 - Unsupervised learning approach
 - No prior knowledge required
 - Assigns a mixed-membership of roles to each node
 - Scales linearly in #(edges)



- ✓ Automated discovery
- ✓ Behavioral roles
- ✓ Roles generalize

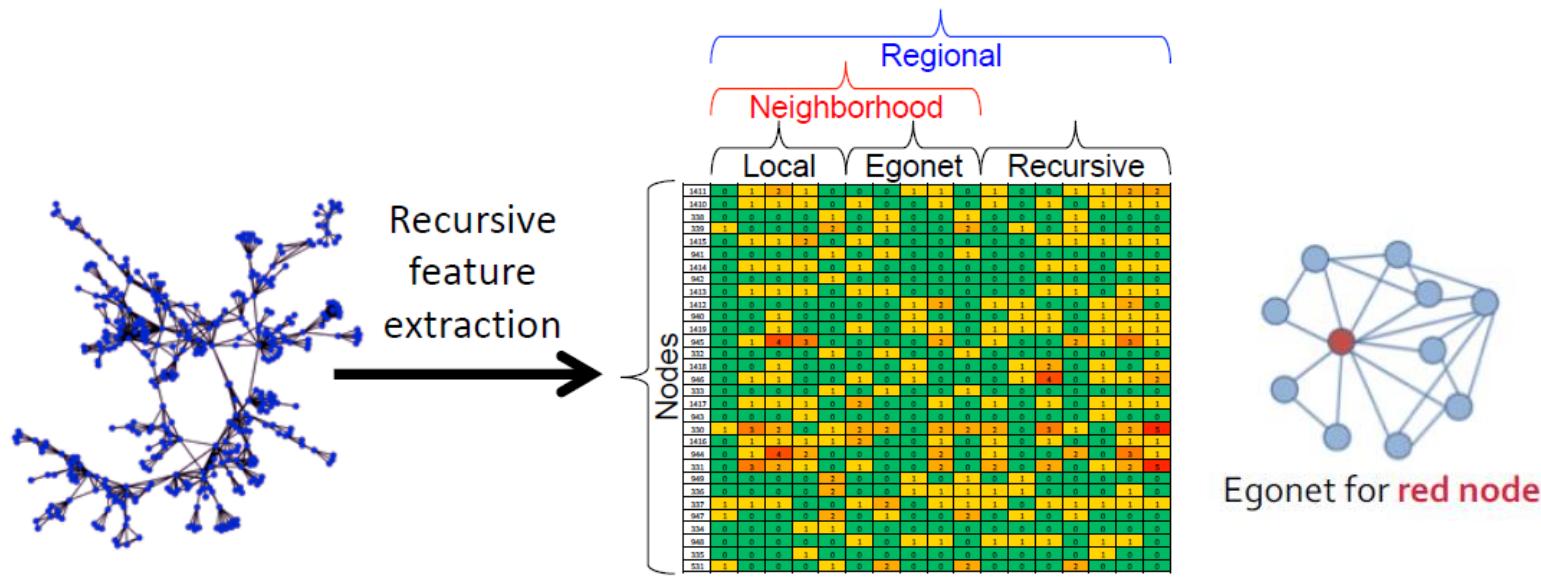
RoIX Algorithm



Role Distributions

ReFEx: Recursive Feature Extraction

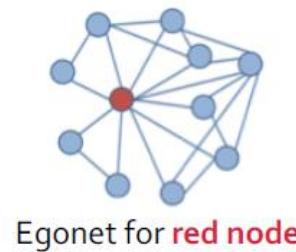
- Recursive feature extraction [[Henderson, et al. 2011a](#)] turns network connectivity into structural features



- ***Neighborhood features:*** What is a node's connectivity pattern?
 - ***Recursive features:*** To what **kinds** of nodes is a node connected?

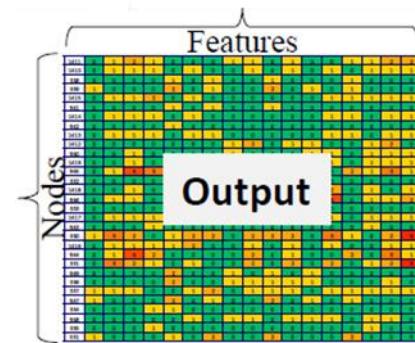
ReFEx: Recursive Feature Extraction

- **Idea:** Aggregate features of a node and use them to generate new recursive features
- ***Base set of a node's neighborhood features:***
 - **Local features:** All measures of the node degree:
 - If network is directed, include in- and out-degree, total degree
 - If network is weighted, include weighted feature versions
 - **Egonet features:** Computed on the node's egonet:
 - **Egonet** includes the node, its neighbors, and any edges in the induced subgraph on these nodes
 - #(within-egonet edges),
 - #(edges entering/leaving egonet)



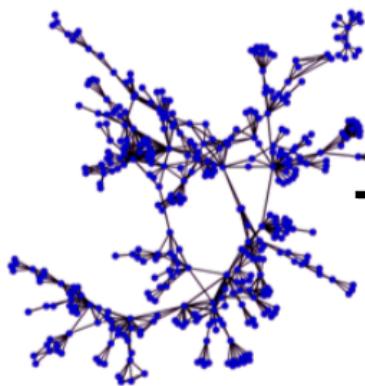
ReFEx: Recursive Feature Extraction

- Start with the ***base set of node features***
- ***Use the set of current node features to generate additional features:***
 - Two types of ***aggregate functions***: ***mean*** and ***sum***
 - e.g., mean value of “unweighted degree” feature between all neighbors of a node
 - Compute means and sums over all current features, including other recursive features
 - Repeat
- The number of possible recursive features ***grows exponentially*** with each recursive iteration:
 - Reduce the number of features using a ***pruning technique***:
 - Look for pairs of features that are highly correlated
 - Eliminate one of the features whenever two features are correlated above a user-defined threshold



Role Extraction

Input



Recursively extract features

Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	770	771	772	773	774	775	776	777	778	779	780	781	782	783	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809	800	801	802	803	804	805	806	807	808	809	810	811	812	813	814	815	816	817	818	819	810	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837	838	839	830	831	832	833	834	835	836	837	838	839	840	841	842	843	844	845	846	847	848	849	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864	865	866	867	868	869	860	861	862	863	864	865	866	867	868	869	870	871	872	873	874	875	876	877	878	879	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895	896	897	898	899	890	891	892	893	894	895	896	897	898	899	900	901	902	903	904	905	906	907	908	909	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918	919	910	911	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927	928	929	920	921	922	923	924	925	926	927	928	929	930	931	932	933	934	935	936	937	938	939	930	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945	946	947	948	949	940	941	942	943	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959	950	951	952	953	954	955	956	957	958	959	960	961	962	963	964	965	966	967	968	969	960	961	962	963	964	965	966	967	968	969	970	971	972	973	974	975	976	977	978	979	970	971	972	973	974	975	976	977	978	979	980	981	982	983	984	985	986	987	988	989	980	981	982	983	984	985	986	987	988	989	990	991	992	993	994	995	996	997	998	999	990	991	992	993	994	995	996	997	998	999	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023	1024	1025	1026	1027	1028	1029	1020	1021	1022	1023	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055	1056	1057	1058	1059	1050	1051	1052	1053	1054	1055	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071	1072	1073	1074	1075	1076	1077	1078	1079	1070	1071	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087	1088	1089	1080	1081	1082	1083	1084	1085	1086	1087	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103	1104	1105	1106	1107	1108	1109	1100	1101	1102	1103	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1120	1121	1122	1123	

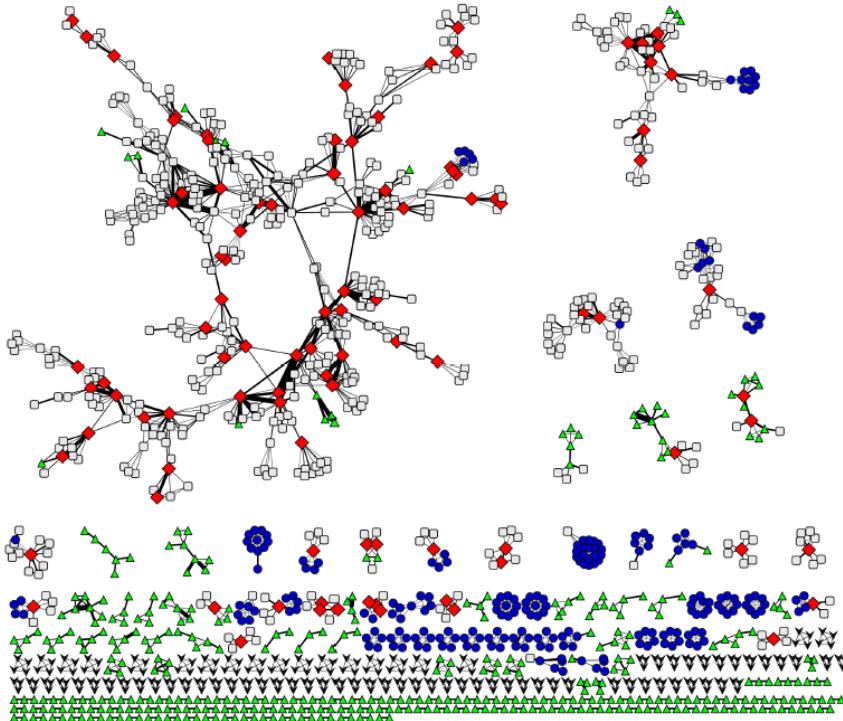
Applications

- ***Role Generalization / Transfer Learning***
 - role effectiveness for the across-network classification task (i.e. network transfer learning)
- ***Structural Similarity***
 - grouping nodes based on their structural similarity
- ***Sensemaking***
 - make “sense” of roles
 - ***NodeSense***: based on node measurements
 - ***NeighborSense***: based on neighbor measurements

Application: *Structural Similarity*

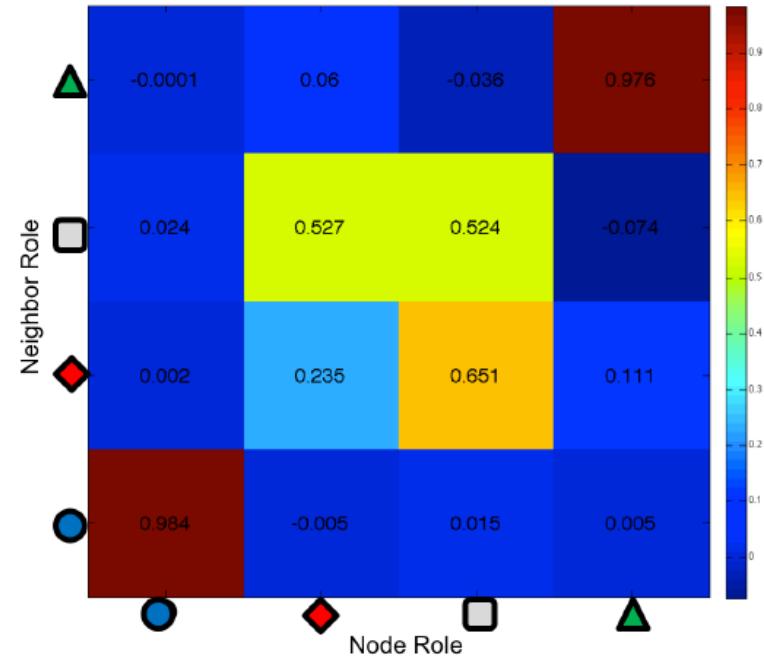
- **Task:** Cluster nodes based on their structural similarity
 - *Exploratory graph mining task (i.e., sensemaking)*
- **Two networks:**
 - ***Network science weighted co-authorship network***
 - (1589) Nodes: Network scientists
 - (2743) Edges: The number of co-authored papers
 - ***Political books co-purchasing network***
 - (105) Nodes: Political books on Amazon
 - (441) Edges: Frequent co-purchasing of books by the same buyers
- **Setup:** For each network
 - Use RolX to assign each node a distribution over the set of discovered structural roles
 - Determine similarity between nodes by comparing their ***role distributions***

Structural Similarity: Co-authorship Network



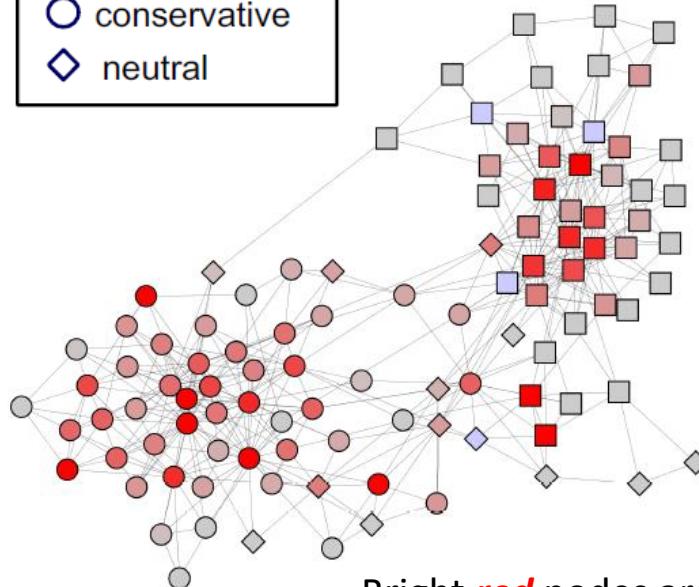
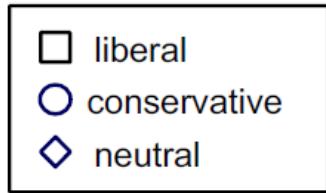
Making sense of roles:

- **Blue circle:** **Tightly knit**, nodes that participate in tightly-coupled groups
- **Red diamond:** **Bridge nodes**, that connect groups of nodes
- **Gray rectangle:** **Main-stream**, majority of nodes, neither a clique, nor a chain
- **Green triangle:** **Pathy**, nodes that belong to elongated clusters



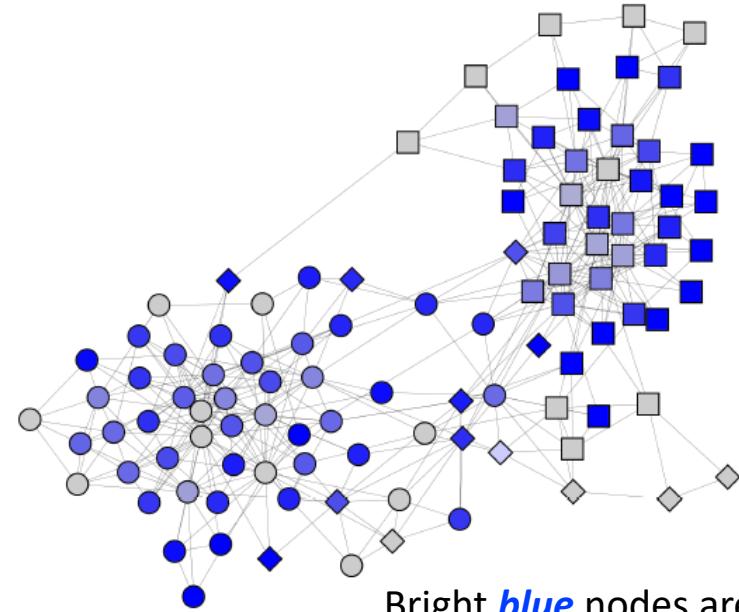
Structural Similarity: Co-purchasing Network

Book labels (i.e., liberal, conservative, neutral) were not given to role discovery algorithm



Bright red nodes are
locally central nodes

The **redness** of a node:
% membership in Role 1 (**local central-ness**)

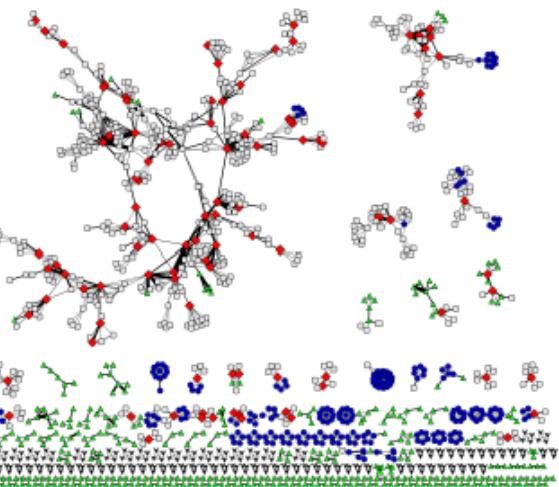
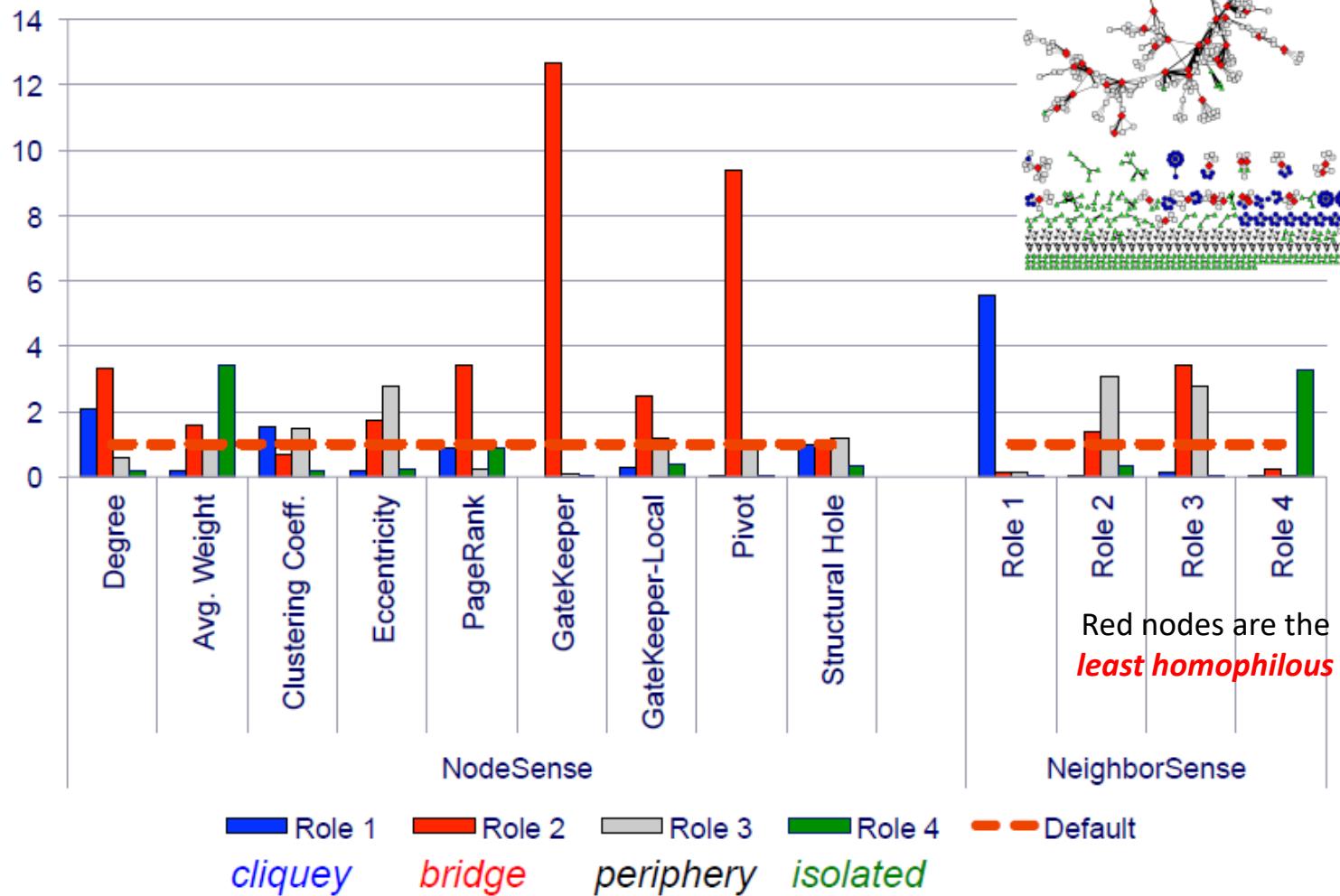


Bright blue nodes are
peripheral nodes

The **blueness** of a node:
% membership in Role 2 (**peripheral-ness**)

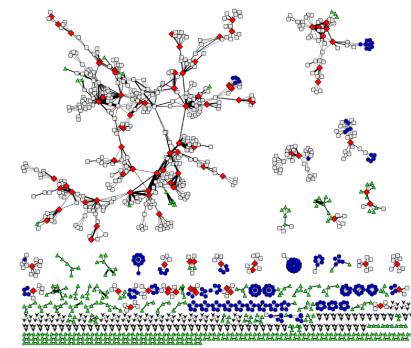
RolX's roles can be used to find similar nodes in disparate communities

Making Sense of Roles



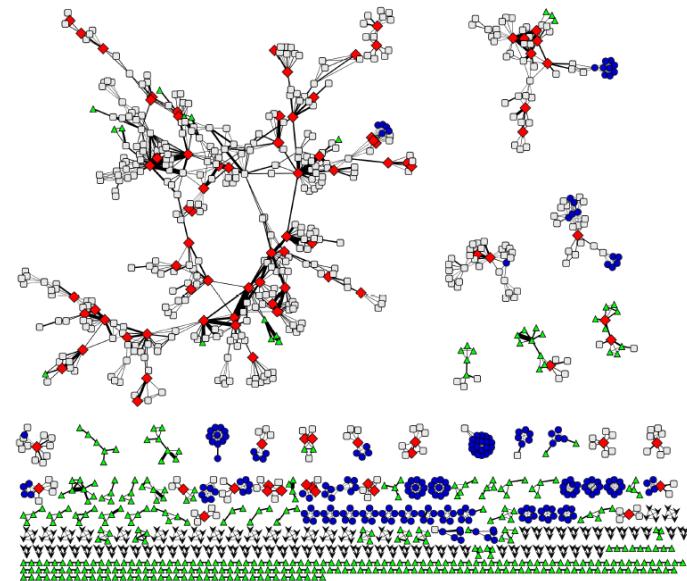
Nodes' Measurements

- **Degree**
 - number of neighbors
- **Weighted degree**
 - average weight of a node's links
- **Clustering coefficient**
- **Gatekeeper**
 - whether a node is an articulation point for some pairs of nodes
- **Local gatekeeper**
- **Pivot**
 - a node with high betweenness
- **Structural hole**
 - to what extent are a node's links redundant
- **Peripheral nodes**
 - low degree and high eccentricity
- **Eccentricity**
 - the longest geodesic from a node
- **PageRank**



Making Sense of Nodes

- **Role 1 nodes (blue circles):** authors with many coauthors and homophilic neighborhoods
 - high degree, high clustering coefficient, and high homophily
 - NOT gatekeepers (i.e. articulation points for some pairs of nodes)
 - NOT pivotal nodes (i.e., with high betweenness)
- **Role 2 nodes (red diamonds):** central and prolific authors
 - high total weight, low clustering coefficient but high degree, high PageRank
 - high affinity for Role 3 nodes (i.e., gray rectangles)
 - Removal severely interrupts graph connectivity because they are often gatekeepers and pivotal nodes
- **Role 3 nodes (gray squares):** peripheral authors
 - low degree and high eccentricity (i.e., nodes in the network periphery)
 - NOT gatekeepers, but can be pivotal
 - i.e. do not disconnect the graph, but often increase geodesic lengths when removed
- **Role 4 nodes (green triangles):** isolated authors
 - high average edge weight and homophilic neighborhoods
 - links are not redundant (w.r.t. structural holes)
 - BUT have low scores for most other measures (except homophily)



Sensemaking Analysis: useful for large networks which cannot be easily visualized

