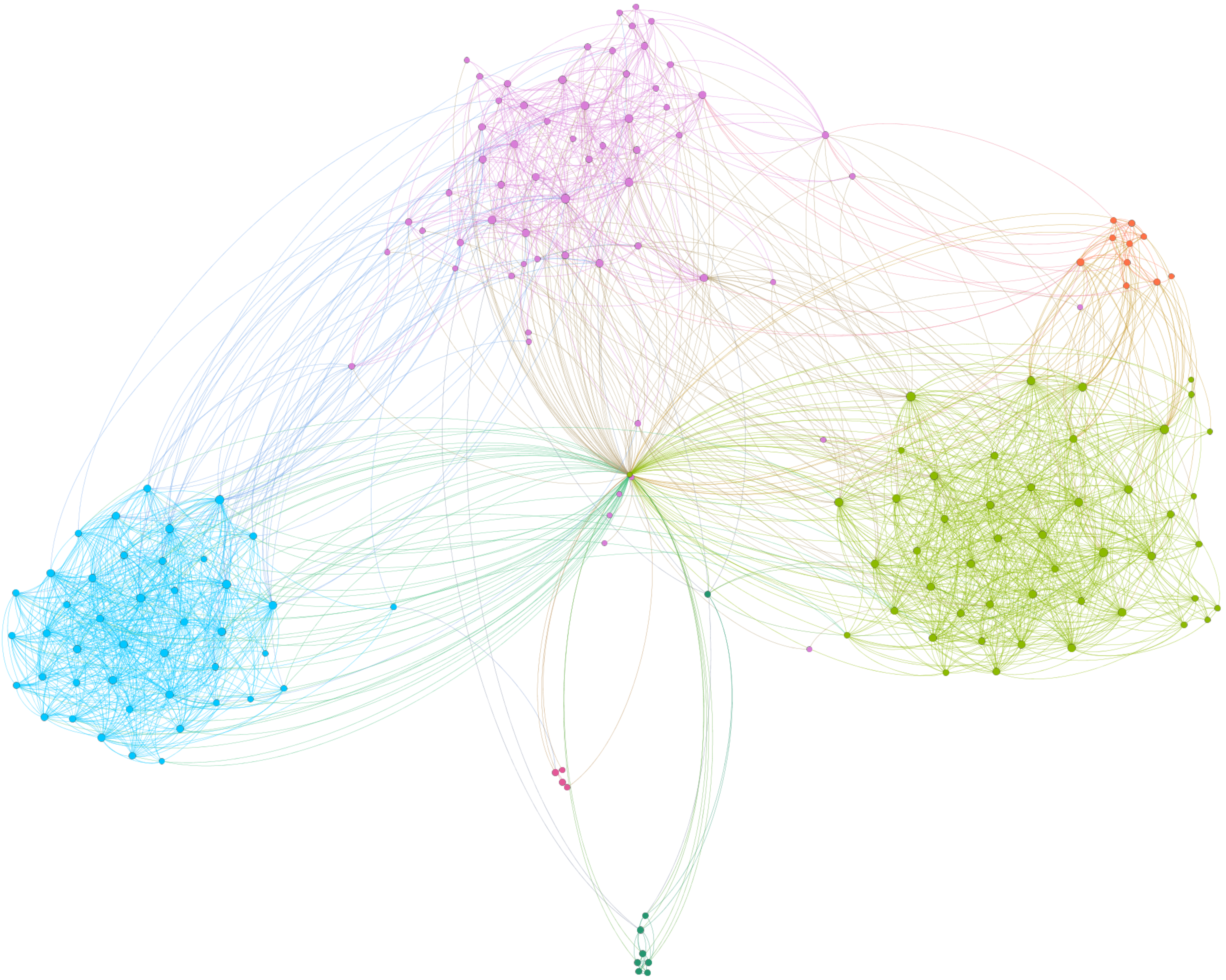
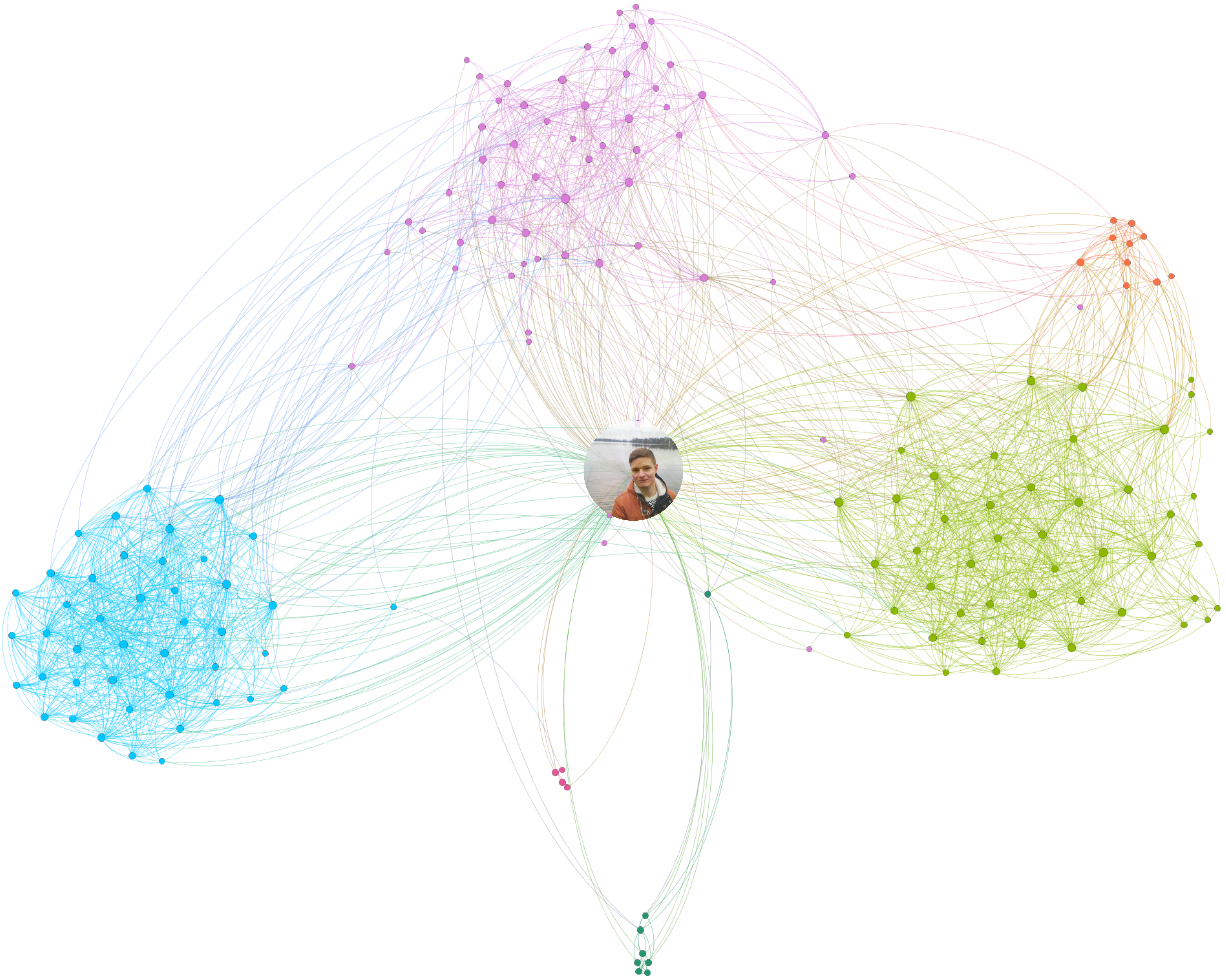
FIND INSIGHTS IN GRAPHS WITH PYTHON

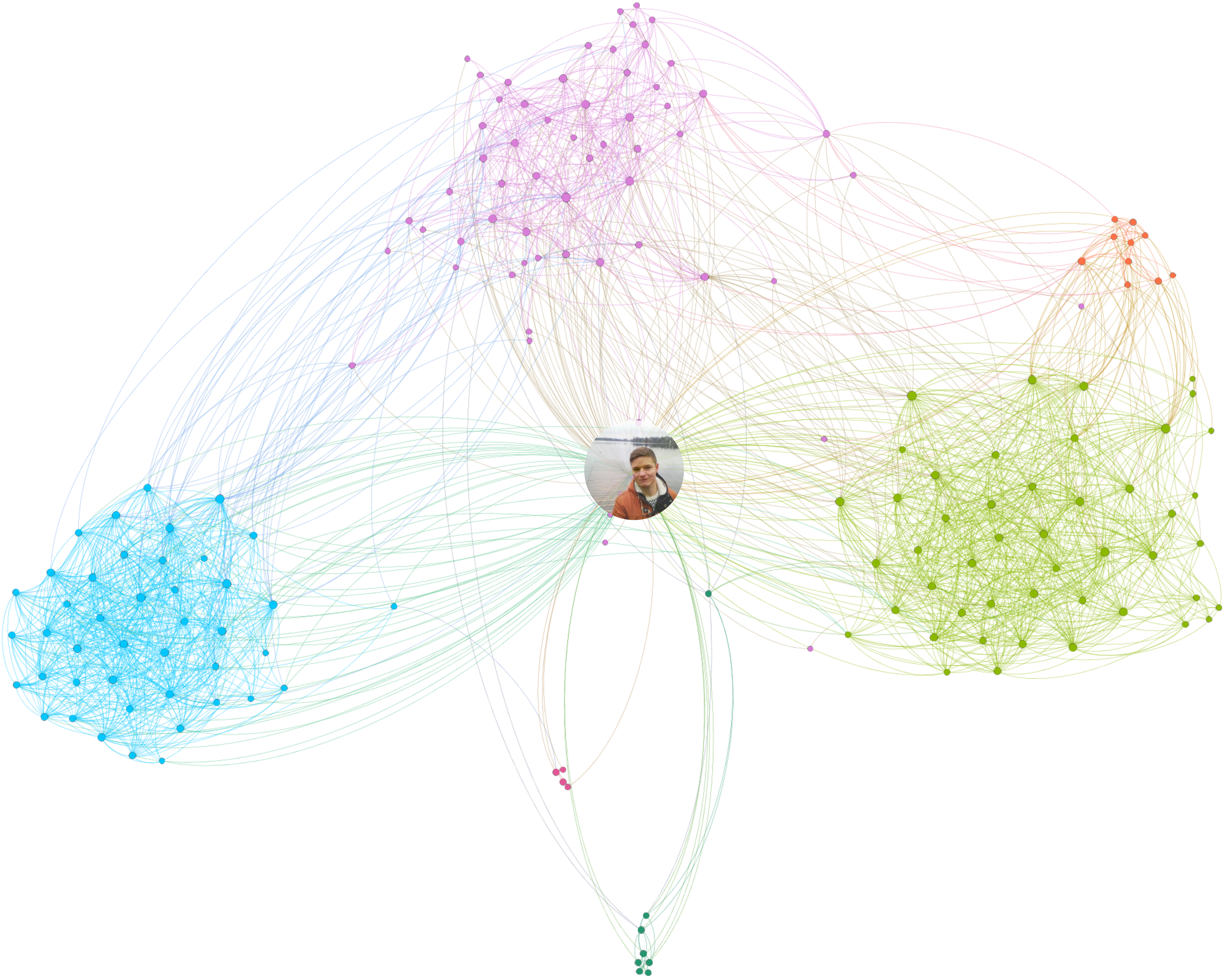
Ego network



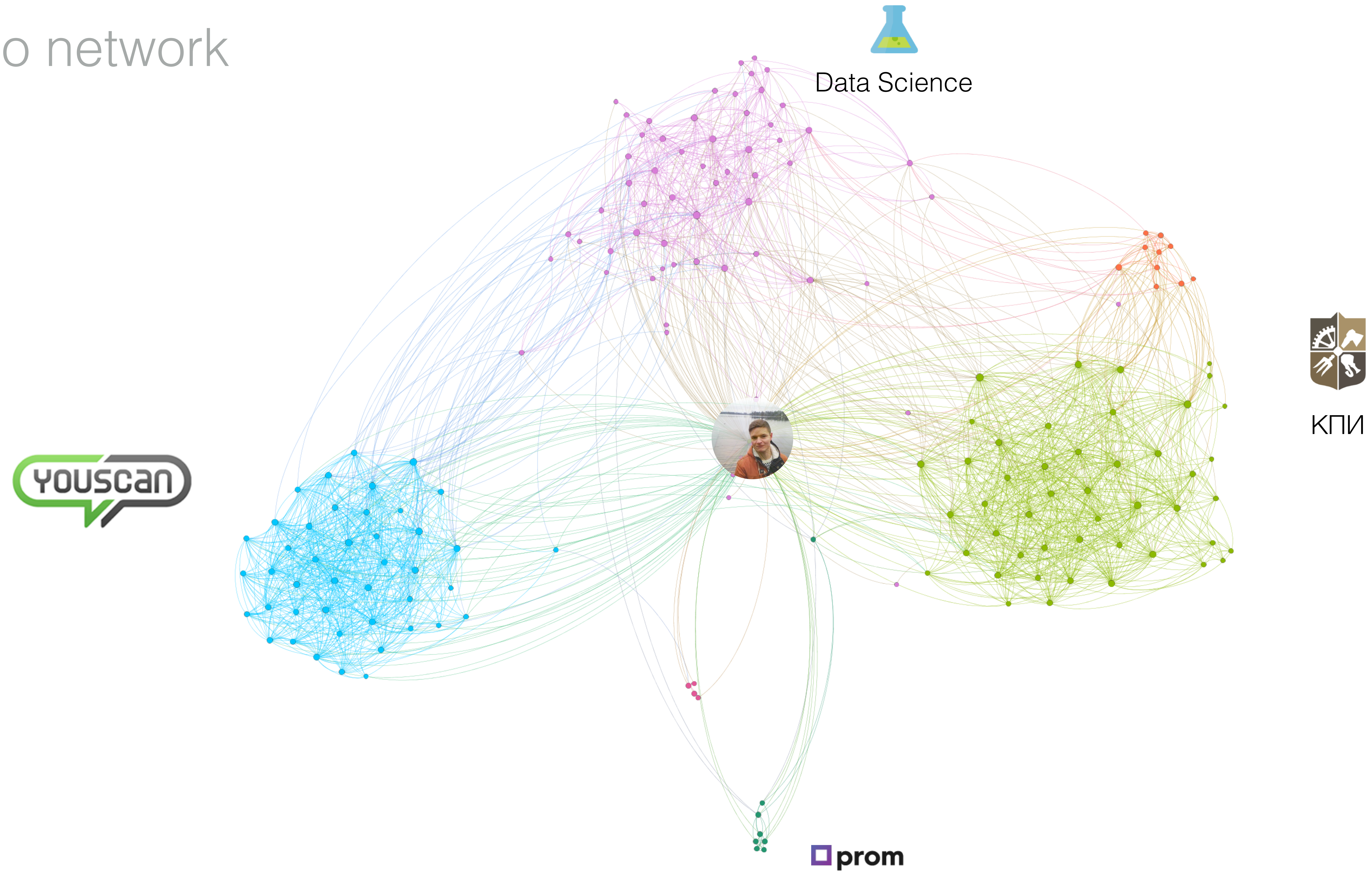
Ego network



Ego network

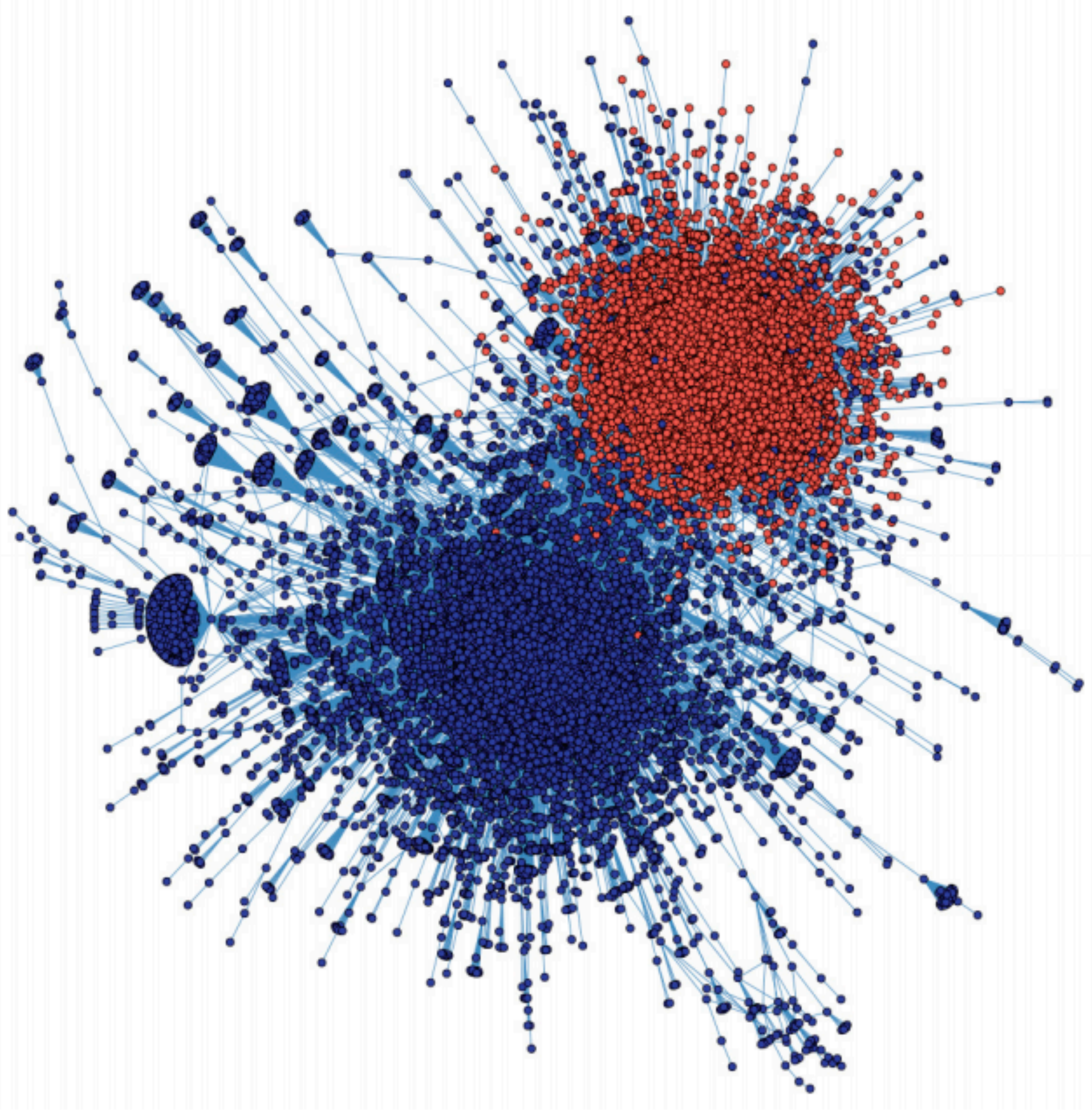


Ego network

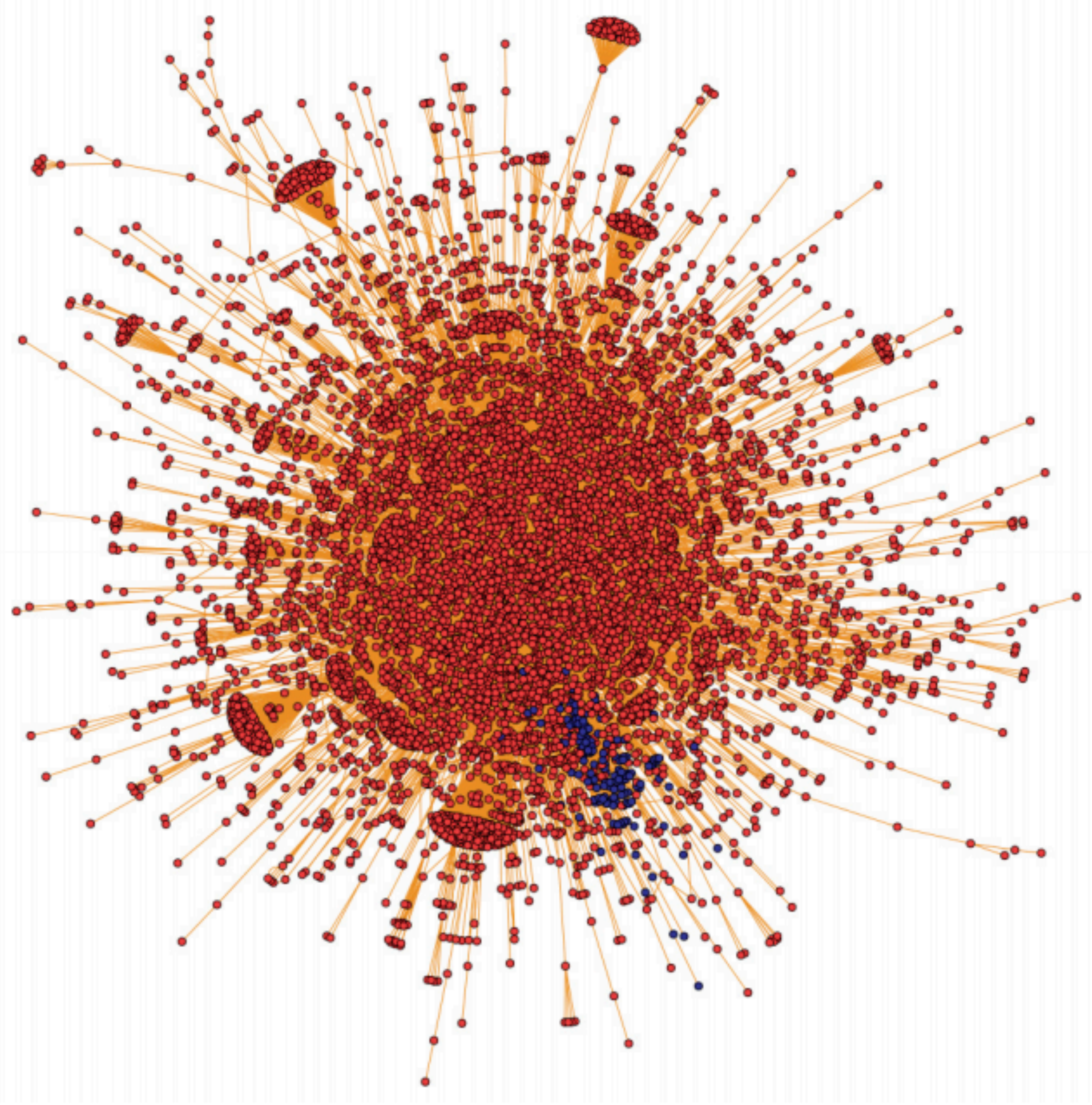


Examples

Political Polarization on Twitter



retweet network

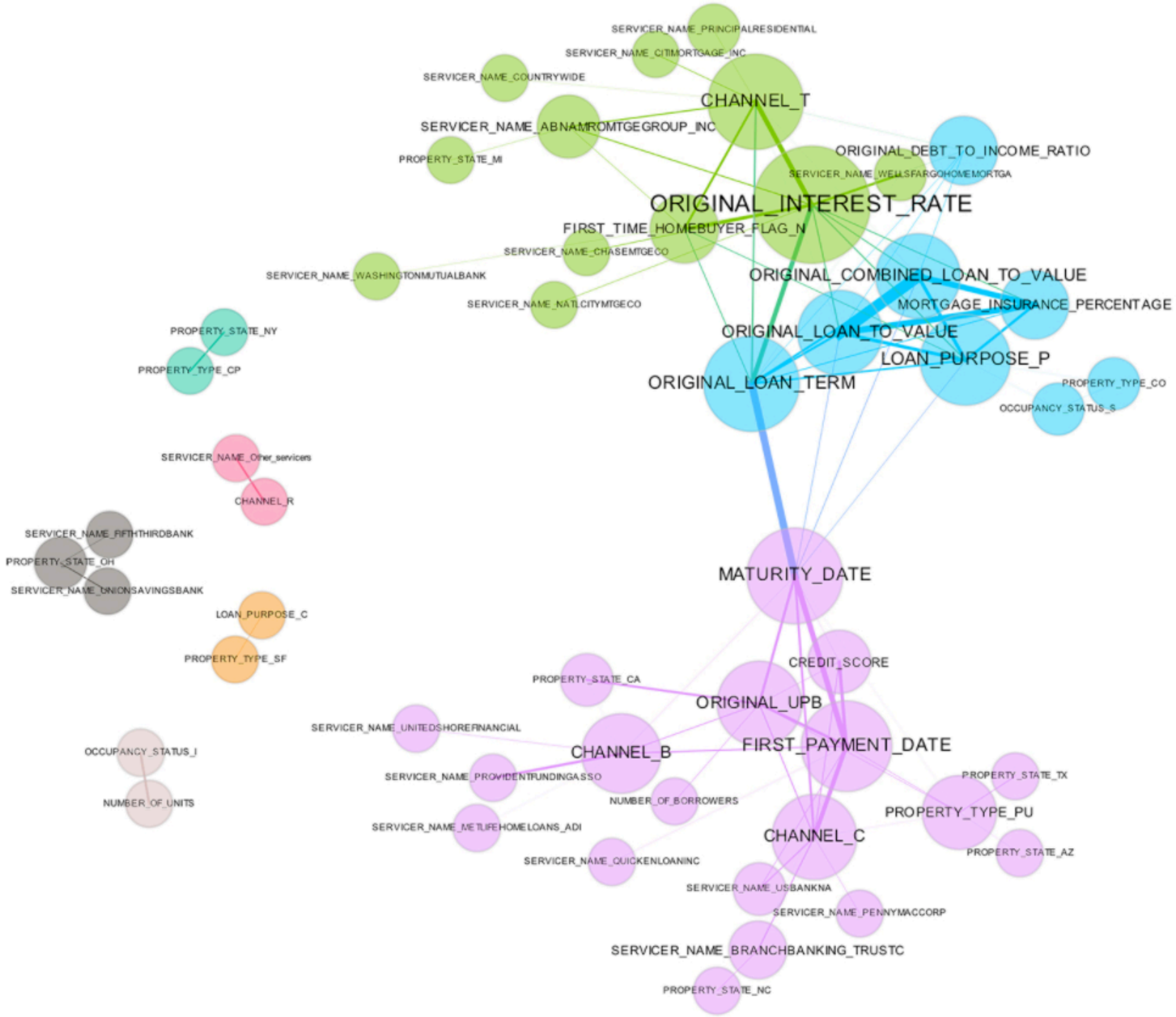


mention network

Political Polarization on Twitter

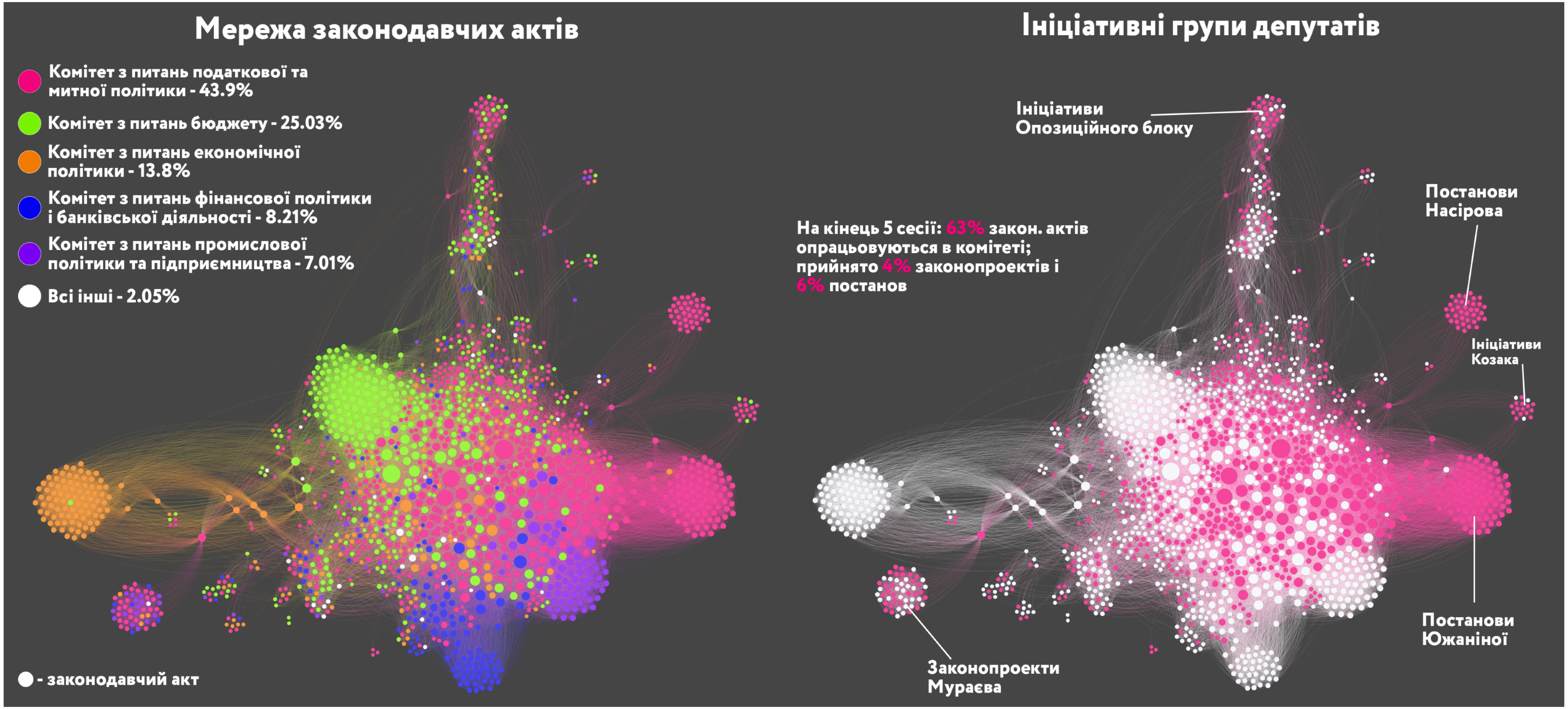
Examples

Correlation Graphs



[Ideas on interpreting machine learning]()

Examples



https://voxukraine.org/longreads/committees/article-ua.html

Terminology

network = graph

nodes = vertices, actors

links = edges, relations

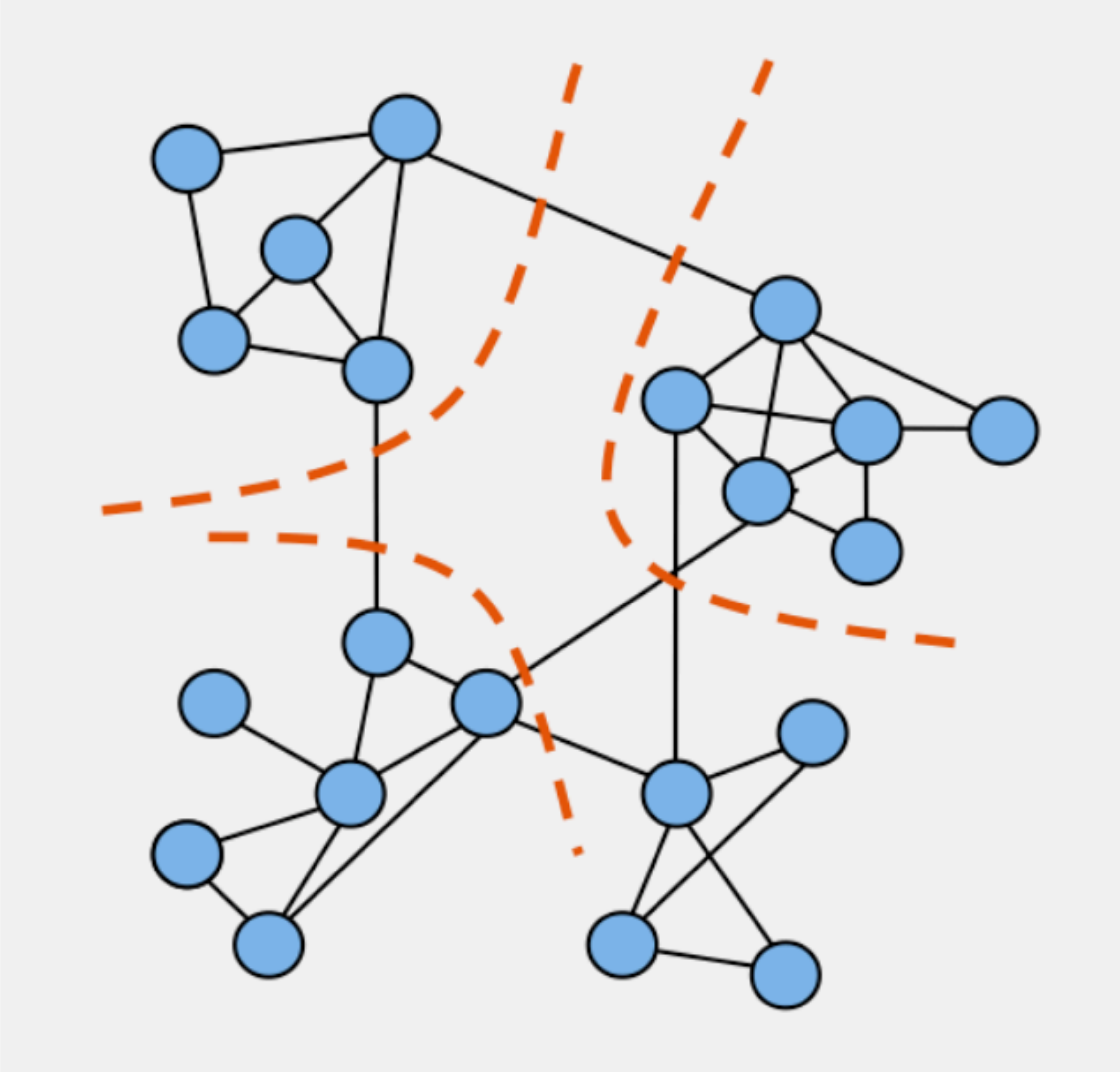
clusters = communities

directed

undirected

Complex networks

- not regular, but not random
- non-trivial topology
- universal properties
- everywhere

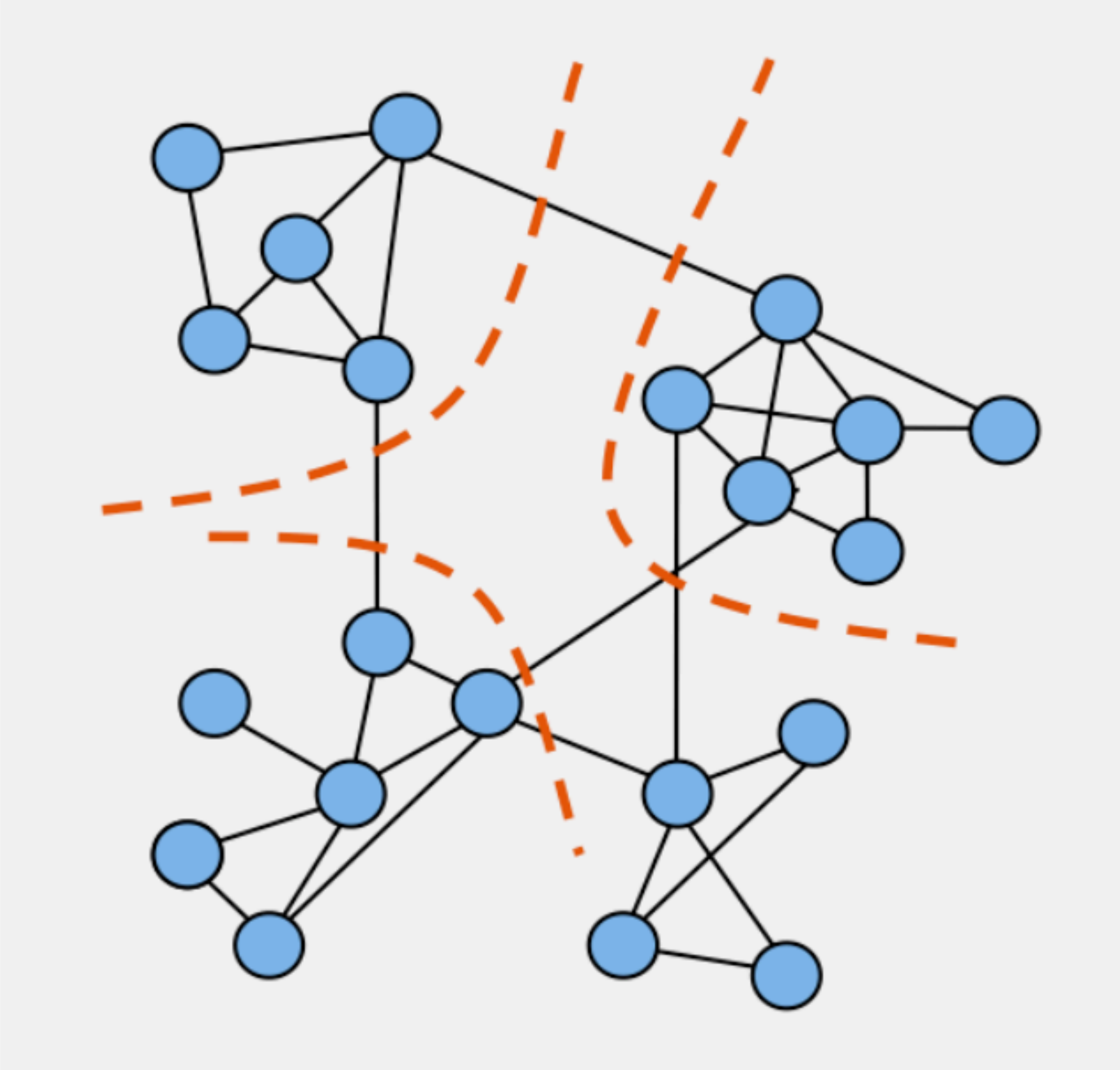


Complex networks

1. Power law node degree distribution
   ”scale-free” networks

2. Small diameter and average path length
   ”small world” networks

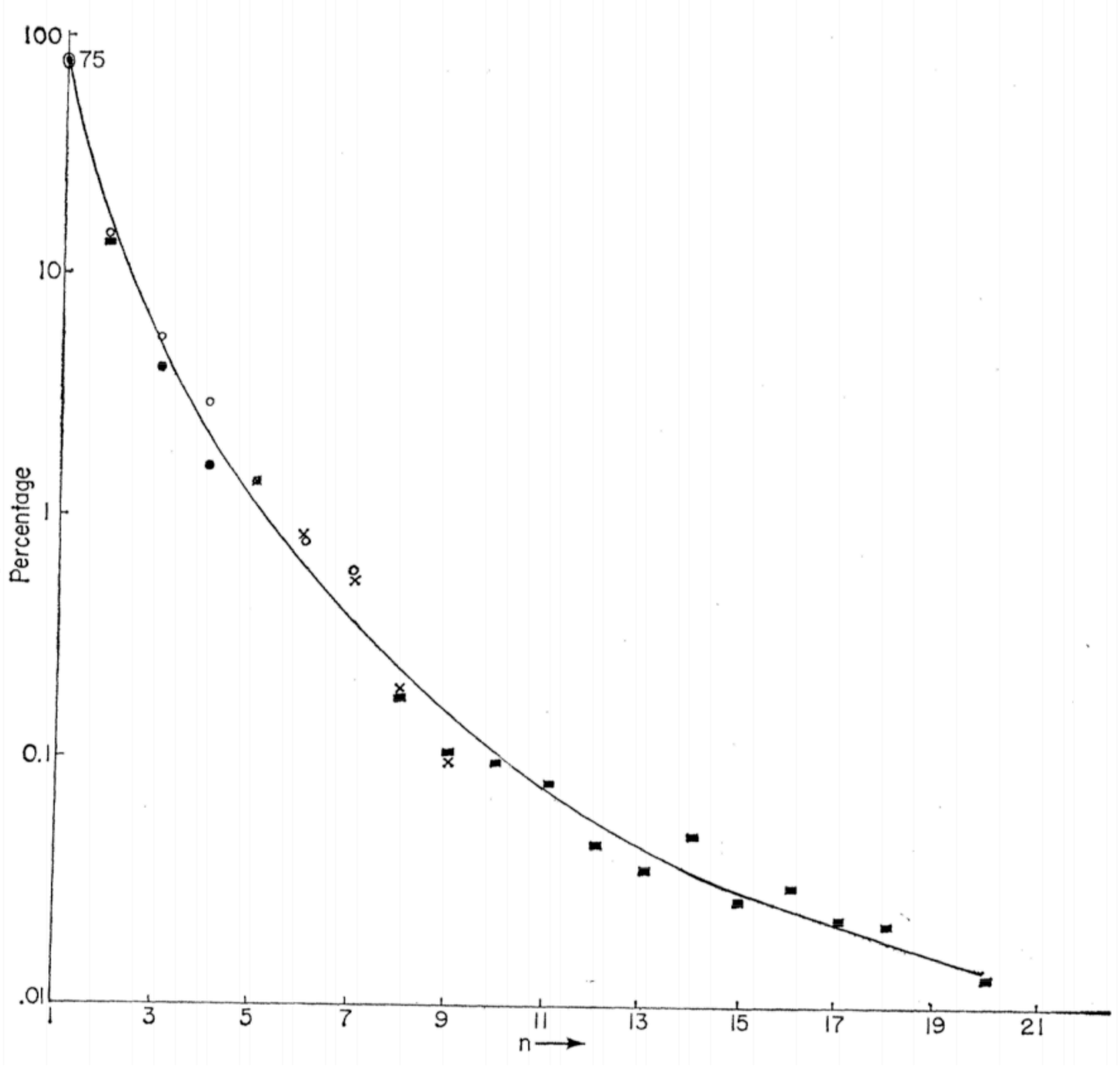
3. Hight clustering coefficient



1. Power law

Citation of scientific papers for 1961

$$f(k) \sim \frac{1}{k^\gamma}$$

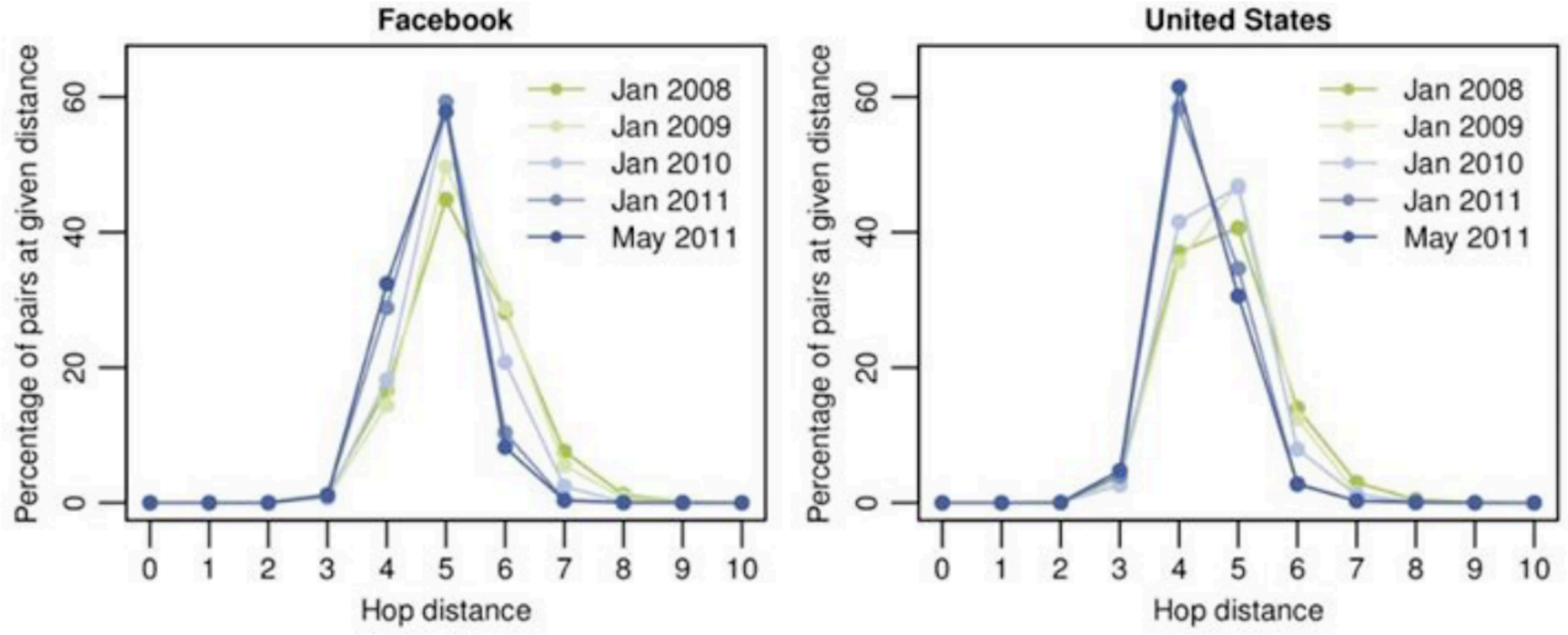


Power law node degree distribution

D.Price, 1965

2. Six degrees of separation

Any two people are on average separated no more that by six intermediate connections

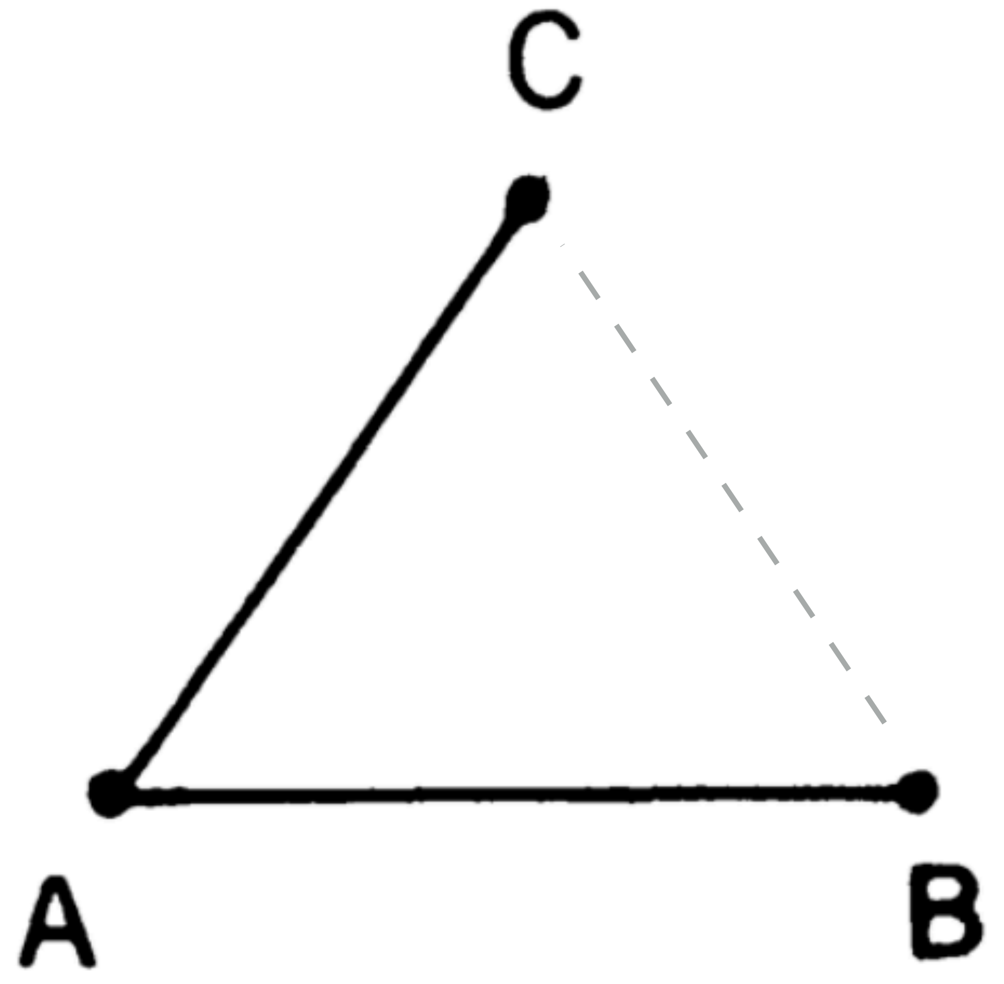


L.Backstrom, 2012

3. Triadic closure

high clustering coefficient

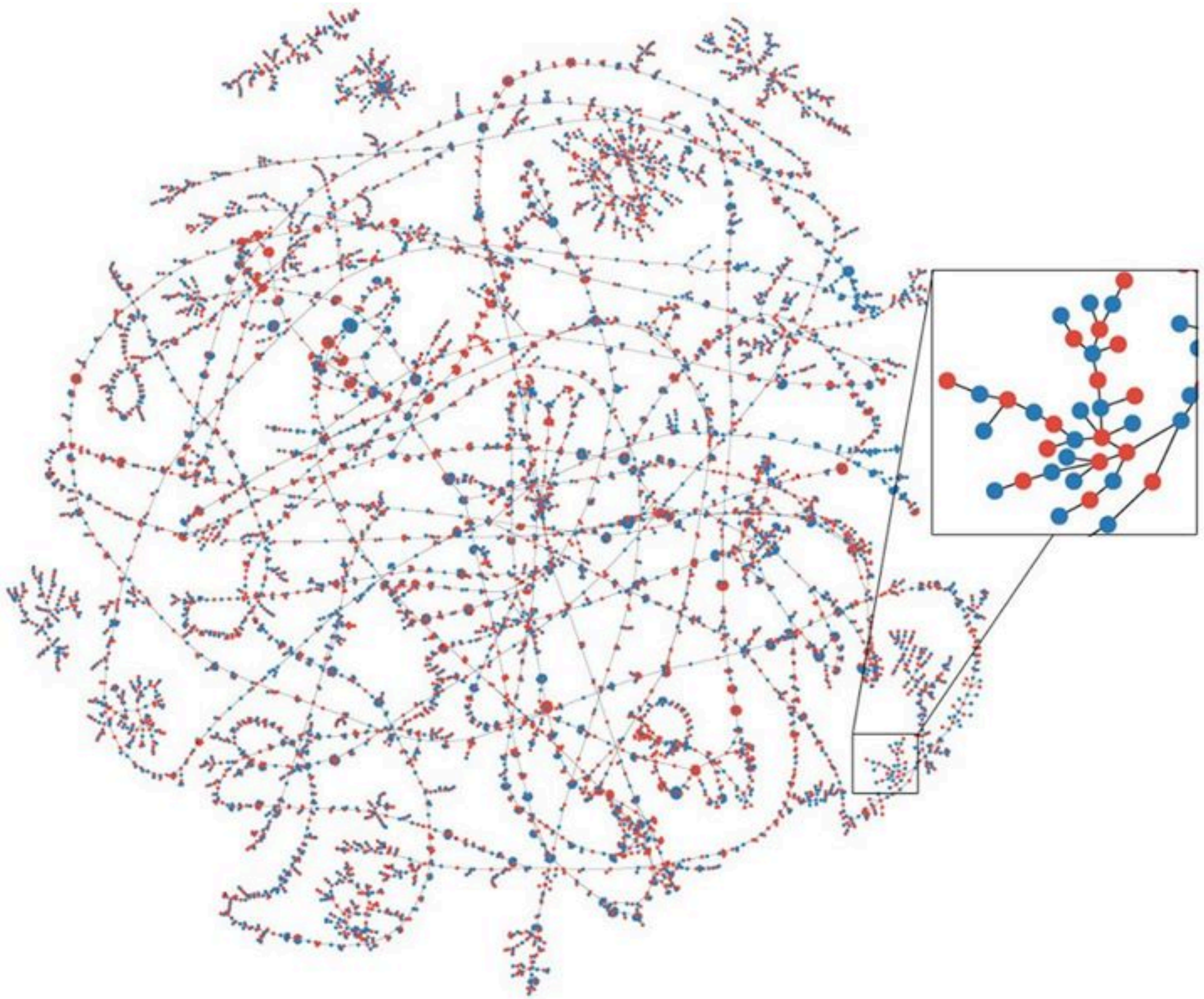
high transitivity



If A and B and B and C are strongly linked, the the tie between B and C is always present

Grannoveter, 1973

Relationship graph



The links of Love

Tools

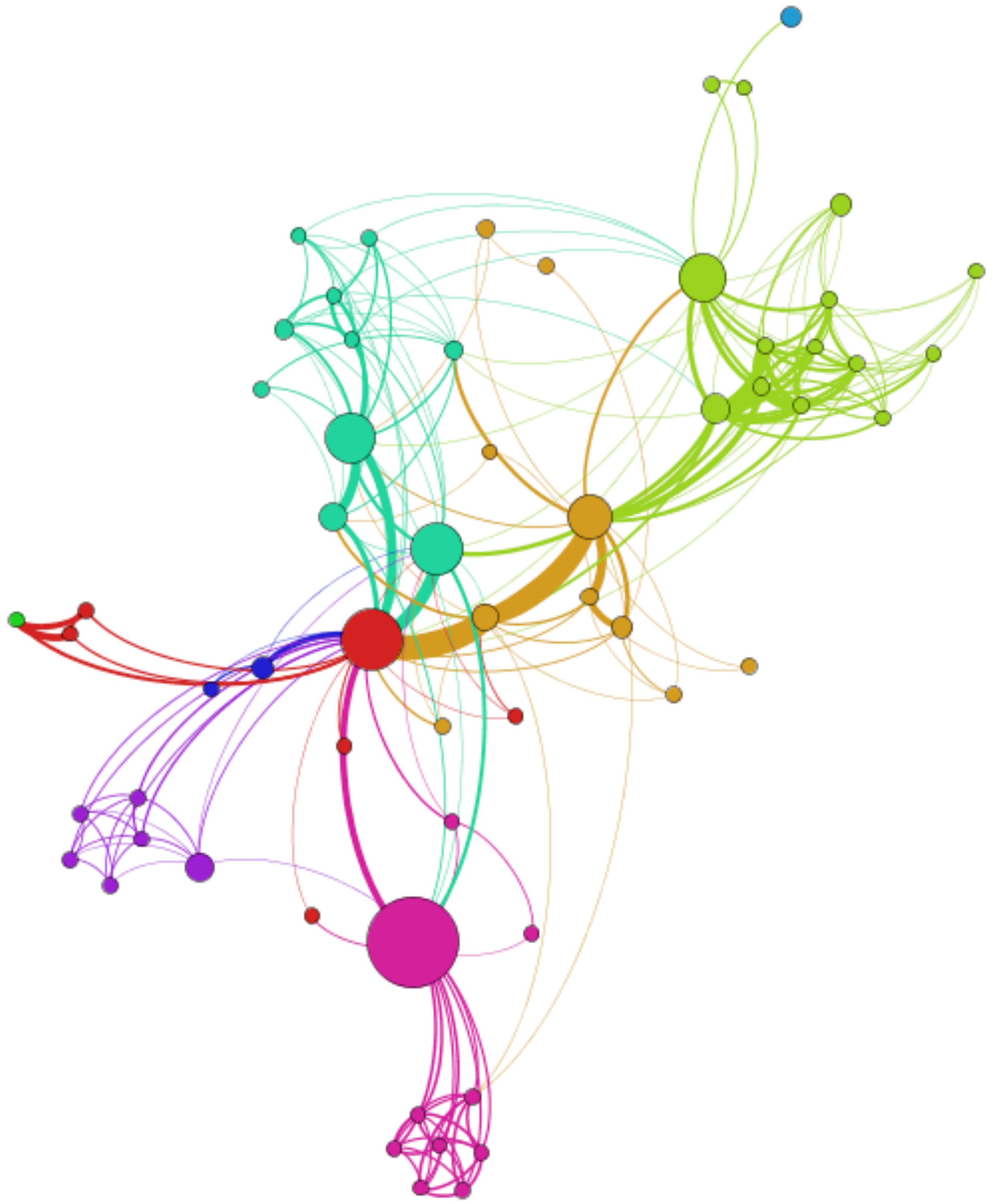
	NetworkX pure Python	igraph Python, R, C	graph-tool Python (C)
API	++	+++	+
speed		++	+++
documentation	+++	+	+++
dynamic networks	+++	+	++

Performance comparison

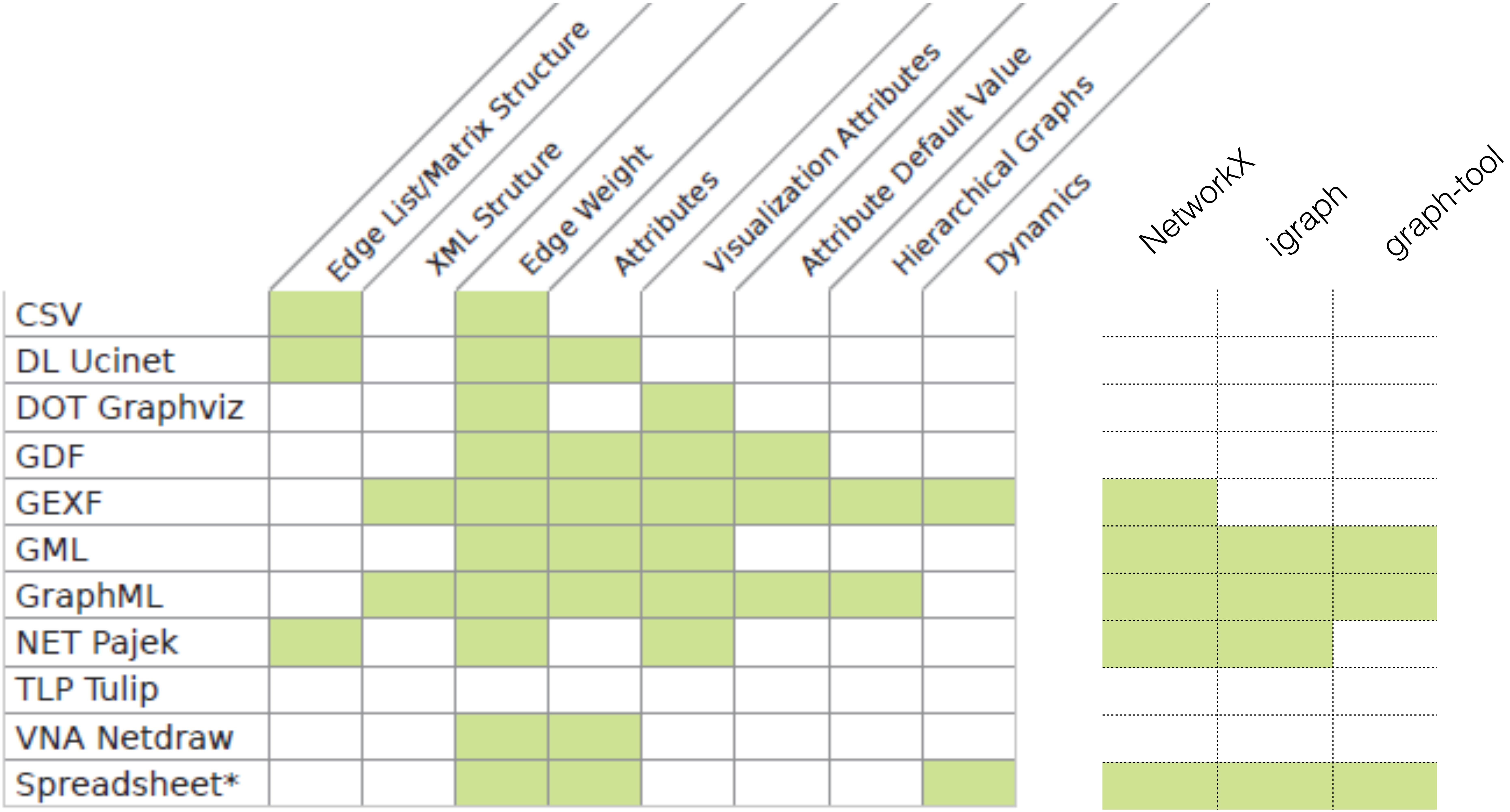
Algorithm	graph-tool (4 cores)	graph-tool (1 core)	igraph	NetworkX
Single-source shortest path	0.004 s	0.004 s	0.012 s	0.152 s
PageRank	0.029 s	0.045 s	0.093 s	3.949 s
K-core	0.014 s	0.014 s	0.022 s	0.714 s
Minimum spanning tree	0.040 s	0.031 s	0.044 s	2.045 s
Betweenness	244.3 s (~4.1 mins)	601.2 s (~10 mins)	946.8 s (edge) + 353.9 s (vertex) (~ 21.6 mins)	32676.4 s (edge) 22650.4 s (vertex) (~15.4 hours)

https://graph-tool.skewed.de/performance

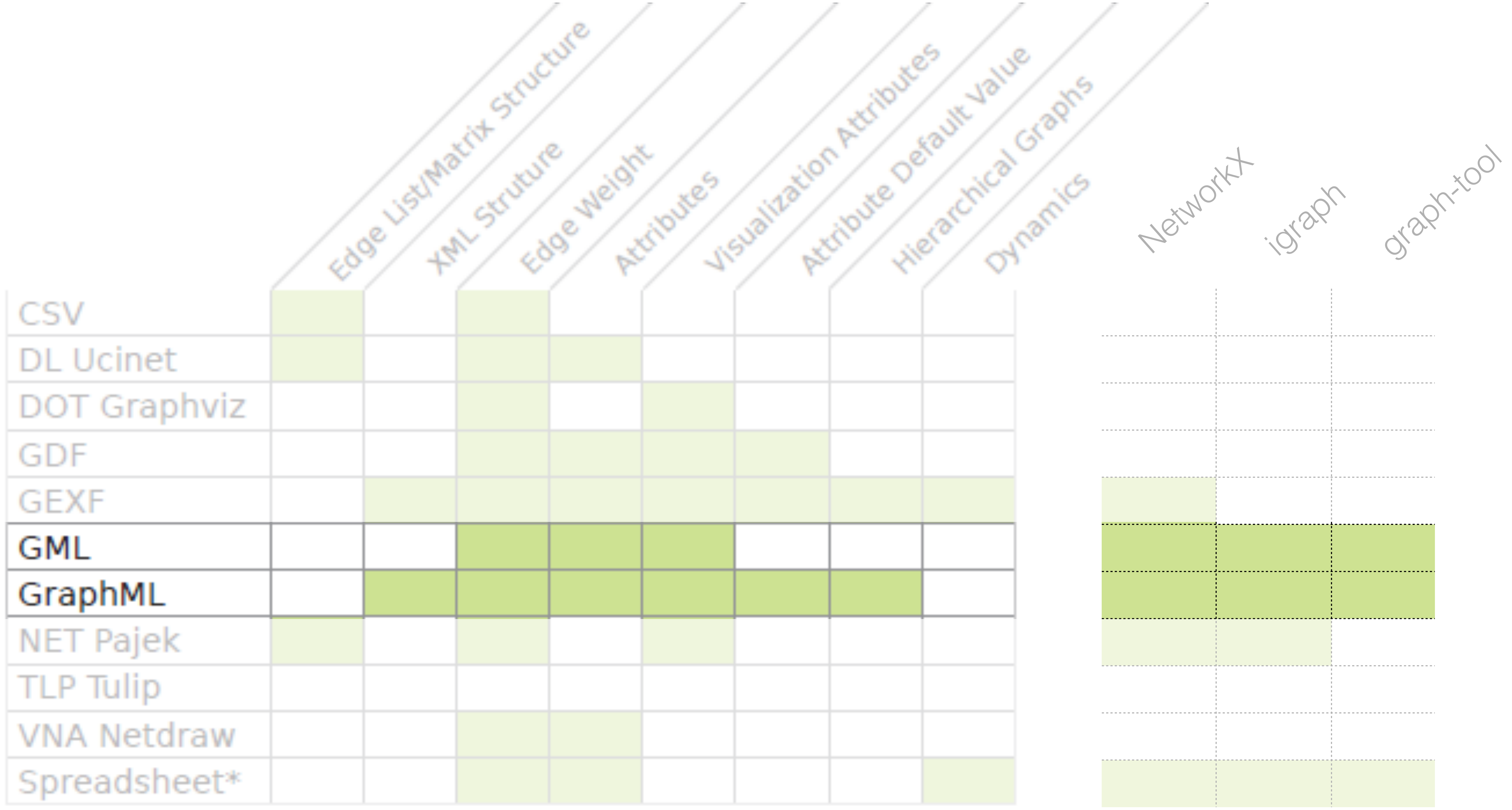
Gephi



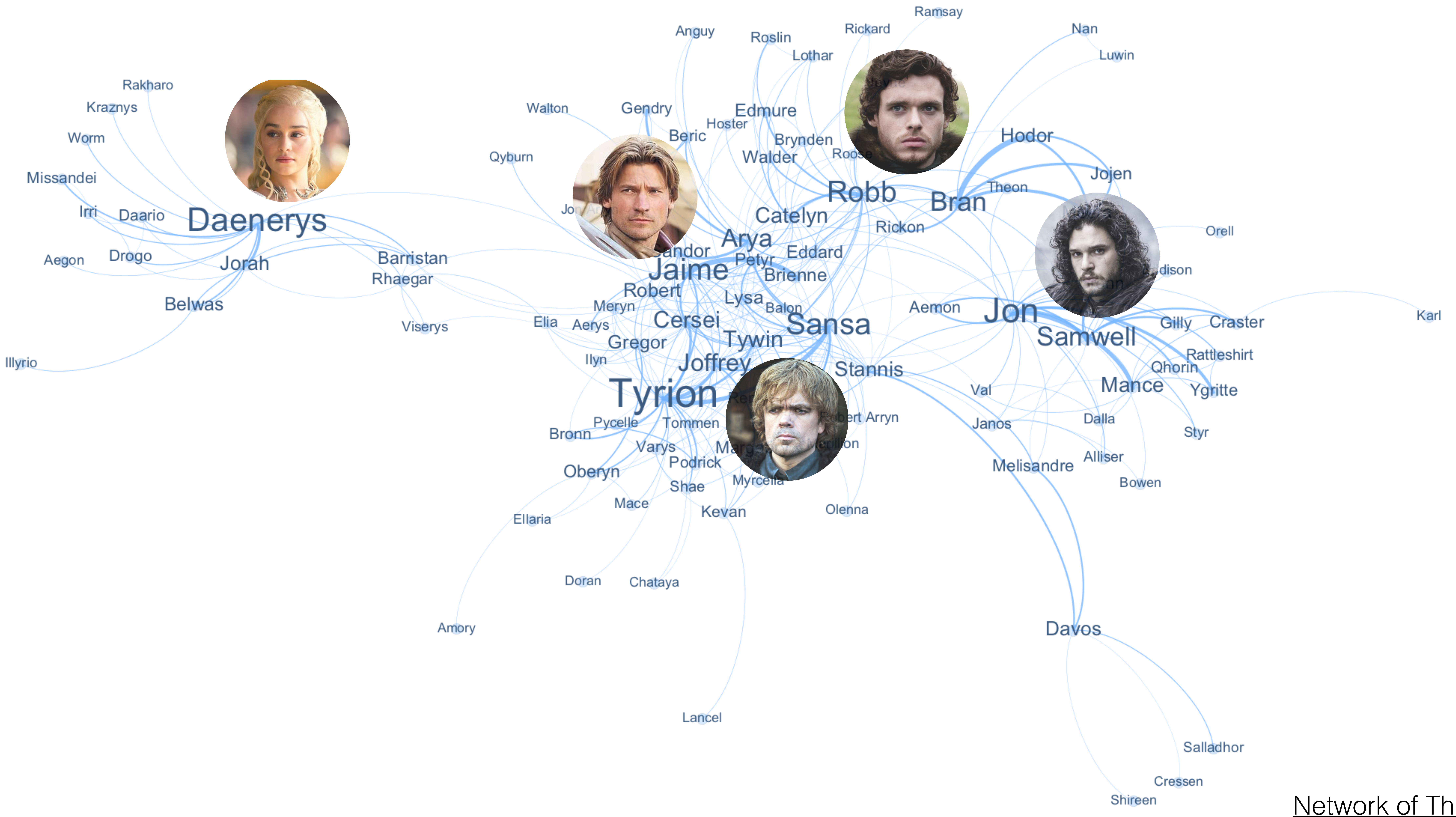
Graph formats



Graph formats

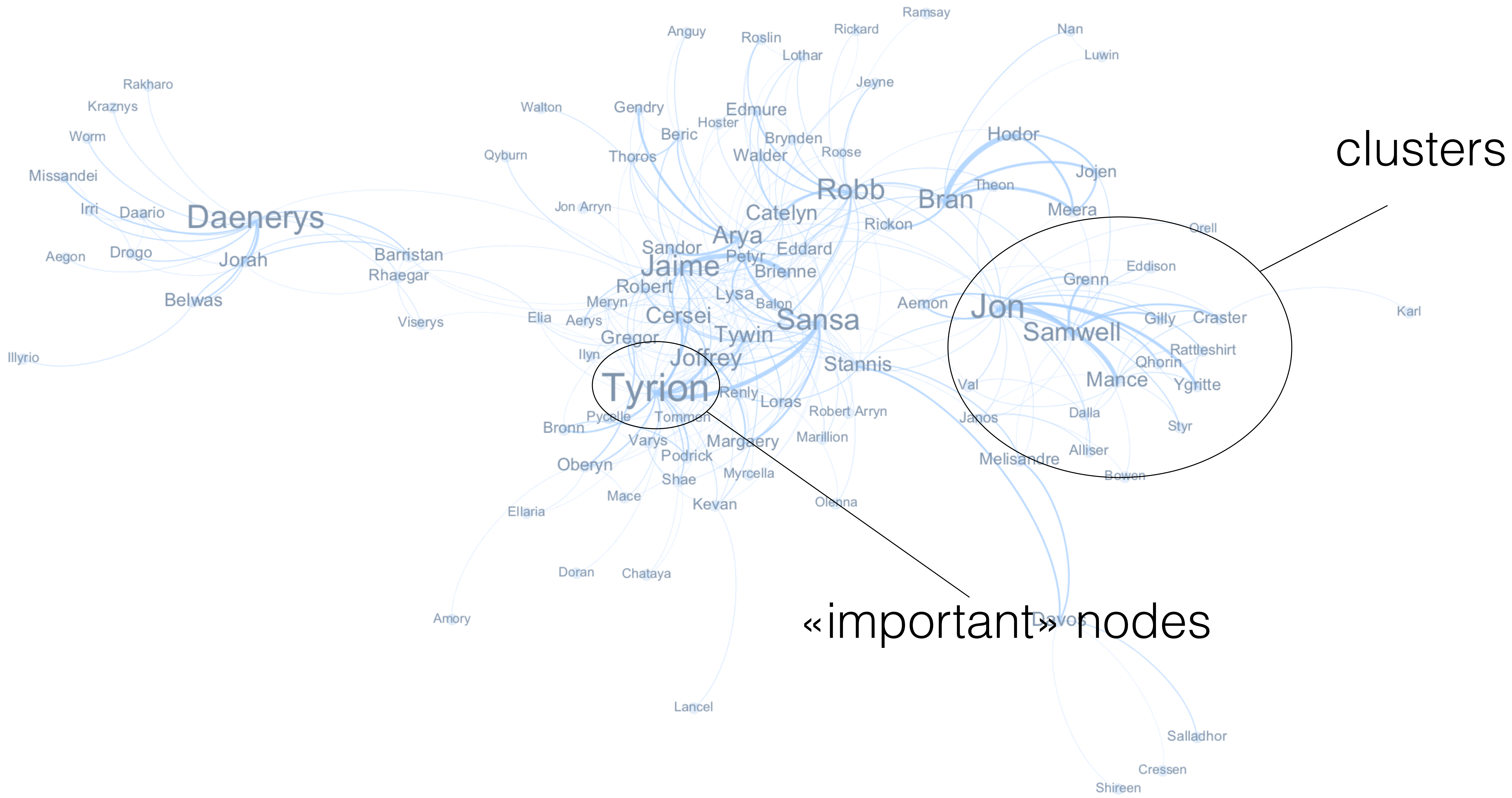



GRAPHS ARE COMING



Network of Thrones

visualisation

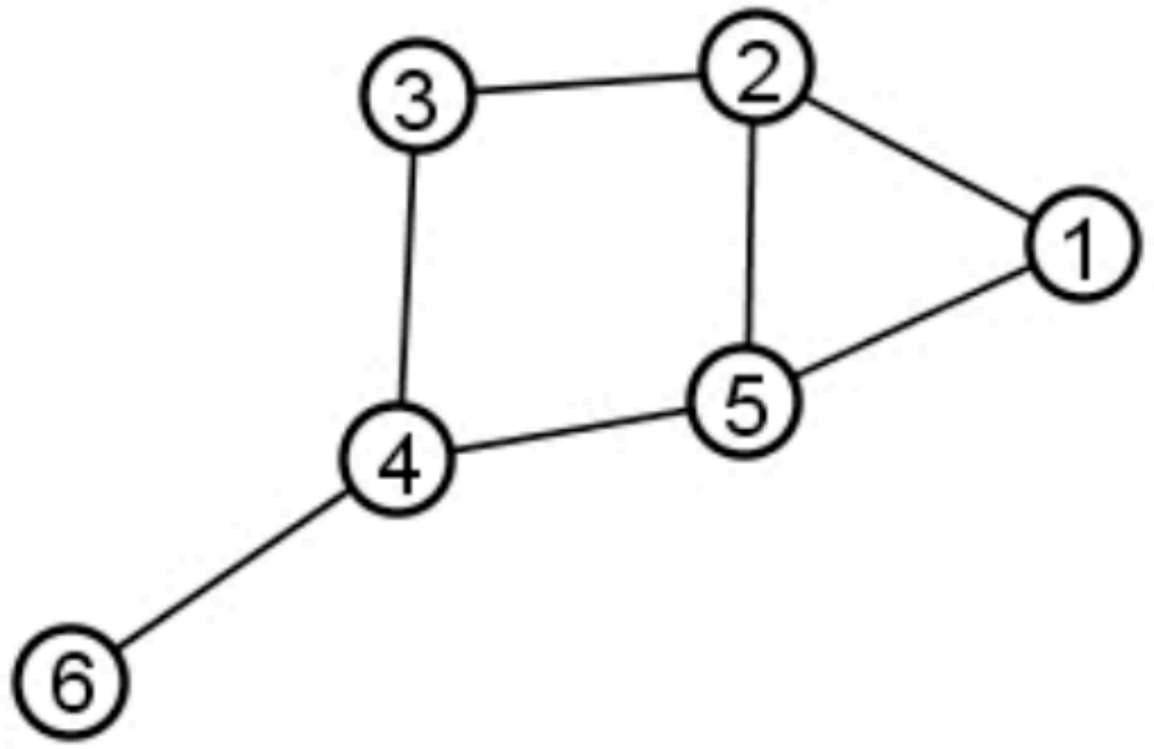


Notations

A *graph* $G = (V, E)$ is an ordered pair of sets: a set of vertices V and a set edges E, where $n = |V|$, $m = |E|$

An *edge* $e_{ij} = (v_i, v_j)$ is pair of vertices (ordered pair for directed graph)

Adjacency matrix $A^{n \times n}$ is a matrix with nonzero element a_{ij} when there is an edge e_{ij}




```
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    0    1    0    0    1    0
[2,]    1    0    1    0    1    0
[3,]    0    1    0    1    0    0
[4,]    0    0    1    0    1    1
[5,]    1    1    0    1    0    0
[6,]    0    0    0    1    0    0
```

Degree centrality

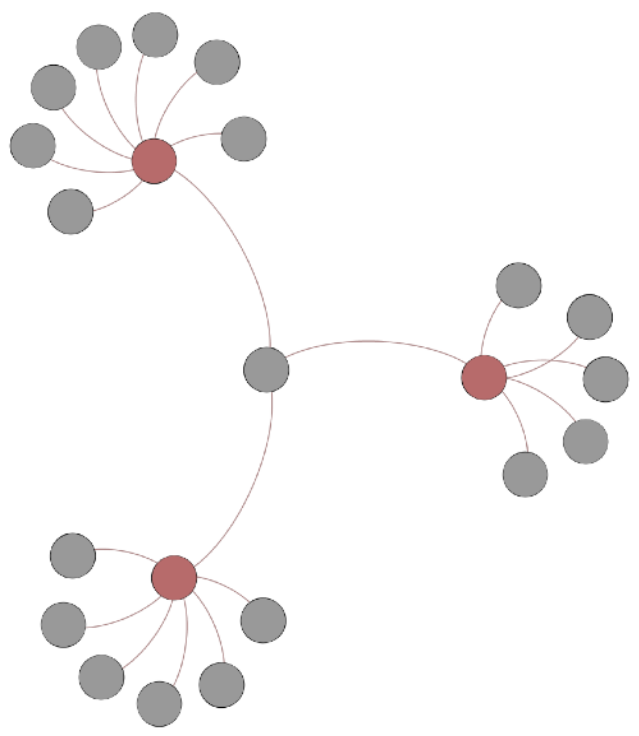
Degree centrality: number of nearest neighbors

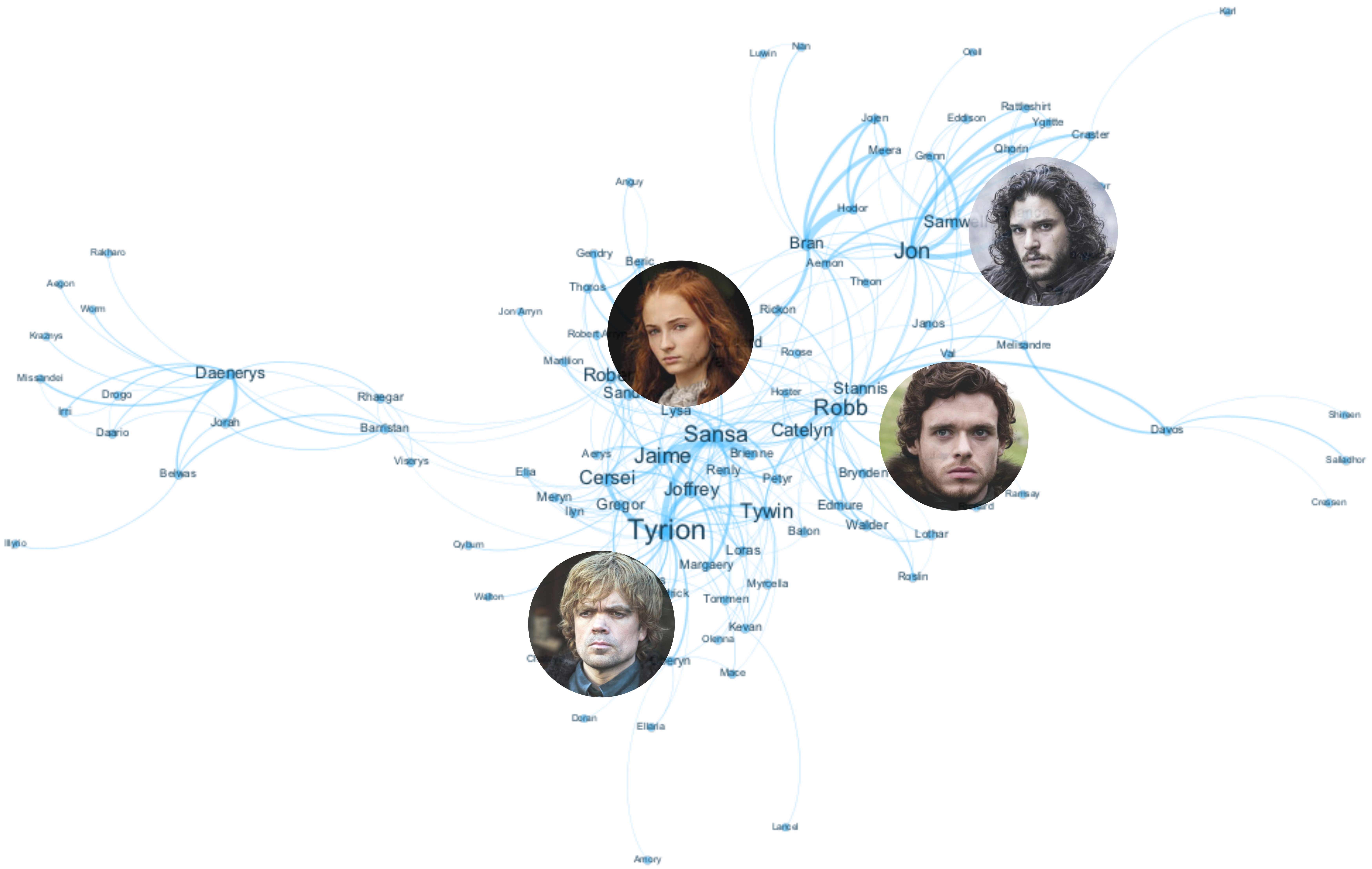
$$C_D(i) = k(i) = \sum_j A_{ij} = \sum_j A_{ji}$$

Normalized degree centrality

$$C_D^*(i) = \frac{1}{n-1} C_D(i) = \frac{k(i)}{n-1}$$

High centrality degree -direct contact with many other actors





Tyrion
Jon
Sansa
Robb
Jaime
Cersei
Joffrey
Tywin
Daenerys
Catelyn
Stannis
Bran
Samwel
Gregor
Renly
Brienne
Petyr
Brynden
Edmure
Walder
Lothar
Balon
Loras
Margaery
Myrcella
Tommen
Kevan
Olenna
Mace
Ellaria
Doran
Lancel
Amory
Walton
Qyburn
Meryn
Ilyn
Elia
Aerys
Lysa
Hoster
Roose
Rickon
Theon
Aemon
Hodor
Meera
Jojen
Luwin
Nan
Orell
Eddison
Rattleshirt
Ygritte
Craster
Qhorin
Grenn
Janos
Melisandre
Val
Davos
Shireen
Salladhor
Cressen
Ramsay
Roslin
Karl
Anguy
Gendry
Beric
Thoros
Jon Arryn
Marillion
Viserys
Rhaegar
Barristan
Jorah
Belwas
Illyrio
Daario
Irri
Drogo
Missandei
Kraznys
Worm
Aegon
Rakharo

Closeness centrality

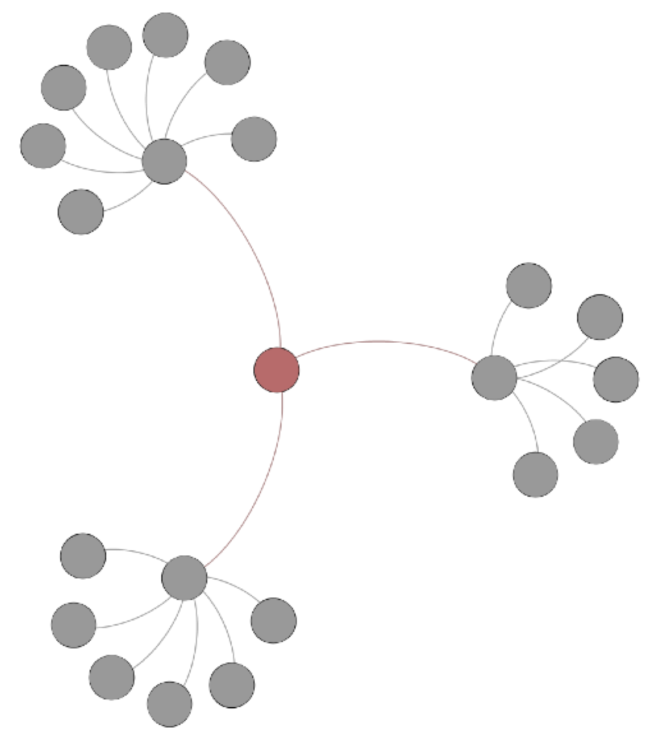
Closeness centrality: how close an actor to all the other actors in network

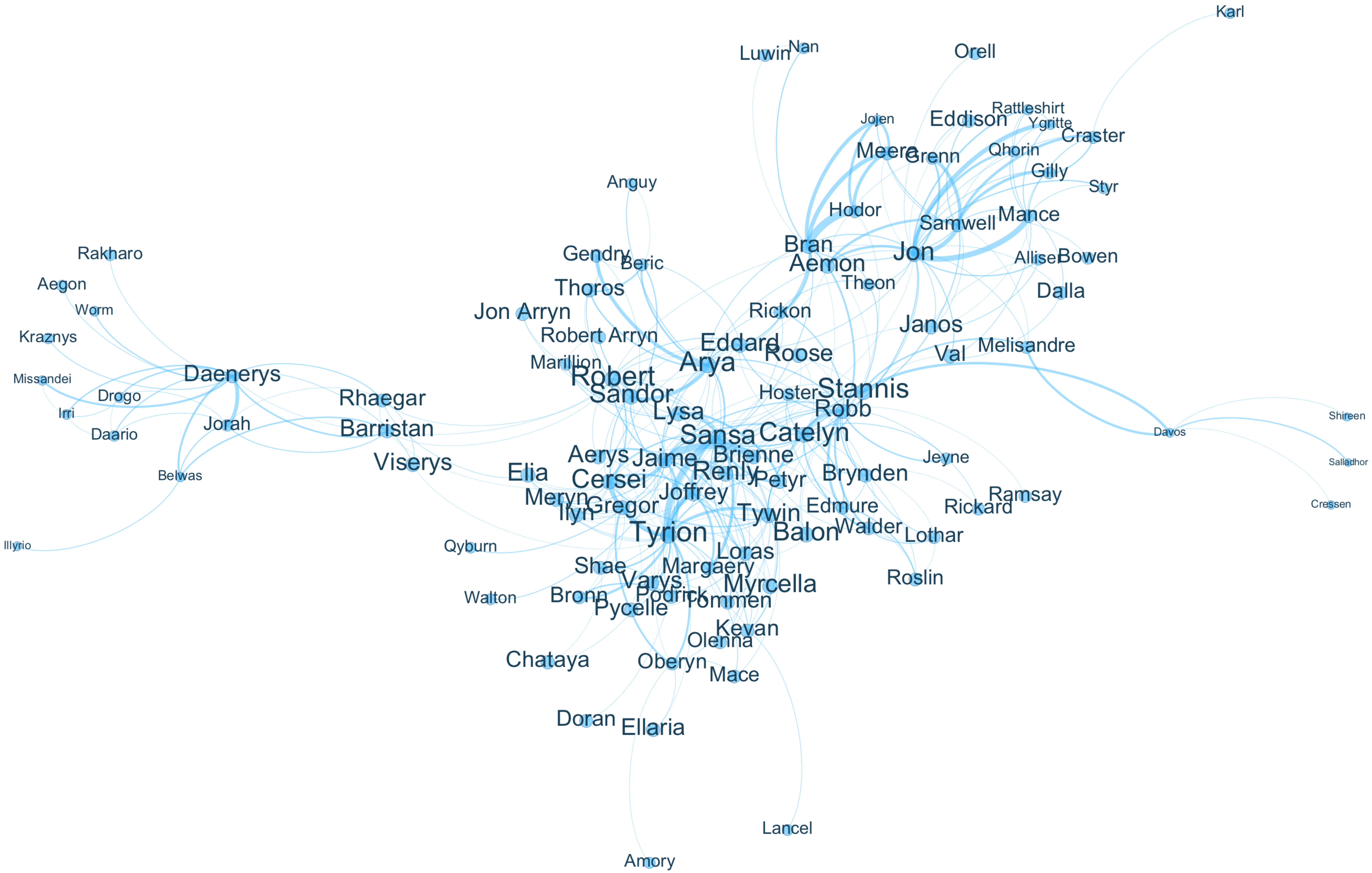
$$C_C(i) = \frac{1}{\sum_j d(i,j)}$$

Normalized closeness centrality

$$C_C^*(i) = (n-1)C_C(i) = \frac{n-1}{\sum_j d(i,j)}$$

High closeness centrality - short communication path to others, minimal number of steps to reach others





Karl
Luwin
Nan
Orell
Jojen
Eddison
Rattleshirt
Ygritte
Craster
Meera
Grenn
Qhorin
Gilly
Styr
Anguy
Hodor
Samwell
Mance
Bran
Jon
Aemon
Bowen
Gendry
Beric
Theon
Rakharo
Aegon
Thoros
Dalla
Worm
Jon Arryn
Rickon
Kraznys
Robert Arryn
Janos
Eddard
Roose
Val
Melisandre
Marillion
Missandei
Daenerys
Arya
Robert
Drogo
Rhaegar
Sandor
Hoster
Stannis
Irri
Lysa
Robb
Jorah
Daario
Barristan
Sansa
Catelyn
Davos
Shireen
Aerys
Jaime
Brienne
Jeyne
Viserys
Belwas
Elia
Renly
Petyr
Brynden
Salladhor
Cersei
Joffrey
Ramsay
Meryn
Gregor
Ilyn
Edmure
Rickard
Tywin
Cressen
Walder
Tyrion
Balon
Lothar
Illyrio
Qyburn
Loras
Shae
Margaery
Varys
Myrcella
Roslin
Walton
Bronn
Podrick
Tommen
Pycelle
Kevan
Olenna
Chataya
Oberyn
Mace
Doran
Ellaria
Lancel
Amory

Betweenness centrality

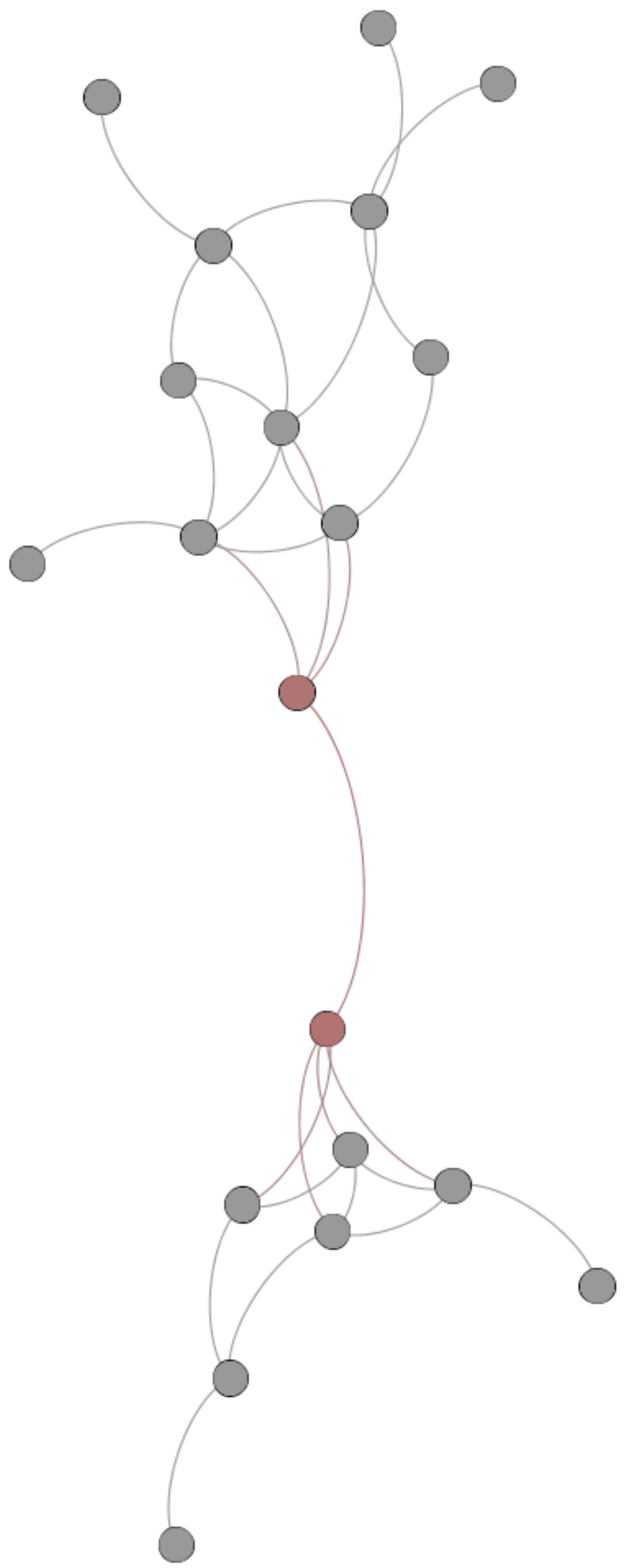
Betweenness centrality: number of shortest paths going through the actor $\sigma_{st}(i)$

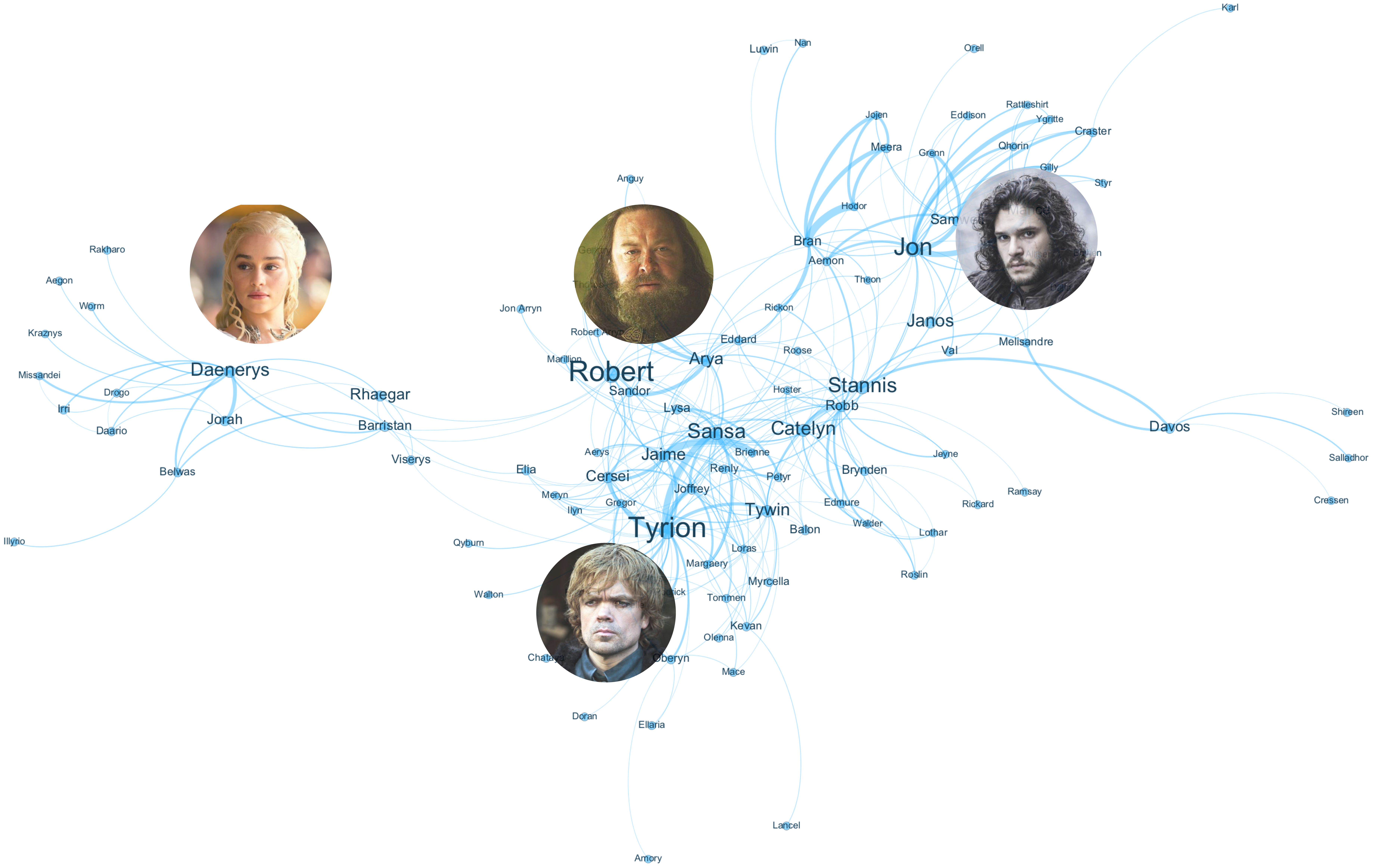
$$C_B(i) = \sum_{s \neq t \neq i} \frac{\sigma_{st}(i)}{\sigma_{st}}$$

Normalized betweenness centrality

$$C_B^*(i) = \frac{2}{(n-1)(n-2)} C_B(i) = \frac{2}{(n-1)(n-2)} \sum_{s \neq t \neq i} \frac{\sigma_{st}(i)}{\sigma_{st}}$$

Hight betweenness centrality - vertex lies on many shortest paths
Probability that a communication from s to t will go through i





Karl
Luwin
Nan
Orell
Jojen
Eddison
Rattleshirt
Ygritte
Craster
Meera
Grenn
Qhorin
Gilly
Styr
Hodor
Sam
Bran
Jon
Aemon
Theon
Rickon
Janos
Val
Melisandre
Anguy
Rakharo
Aegon
Worm
Kraznys
Missandei
Drogo
Irri
Daario
Daenerys
Jorah
Belwas
Illyrio
Rhaegar
Barristan
Viserys
Jon Arryn
Marillion
Robert
Sandor
Eddard
Arya
Roose
Hoster
Stannis
Robb
Lysa
Sansa
Catelyn
Davos
Shireen
Salladhor
Cressen
Jeyne
Ramsay
Rickard
Aerys
Jaime
Brienne
Renly
Petyr
Brynden
Elia
Cersei
Joffrey
Meryn
Ilyn
Gregor
Tywin
Edmure
Walder
Lothar
Roslin
Tyrion
Balon
Qyburn
Loras
Margaery
Myrcella
Walton
Tommen
Kevan
Olenna
Oberyn
Mace
Doran
Ellaria
Lancel
Amory

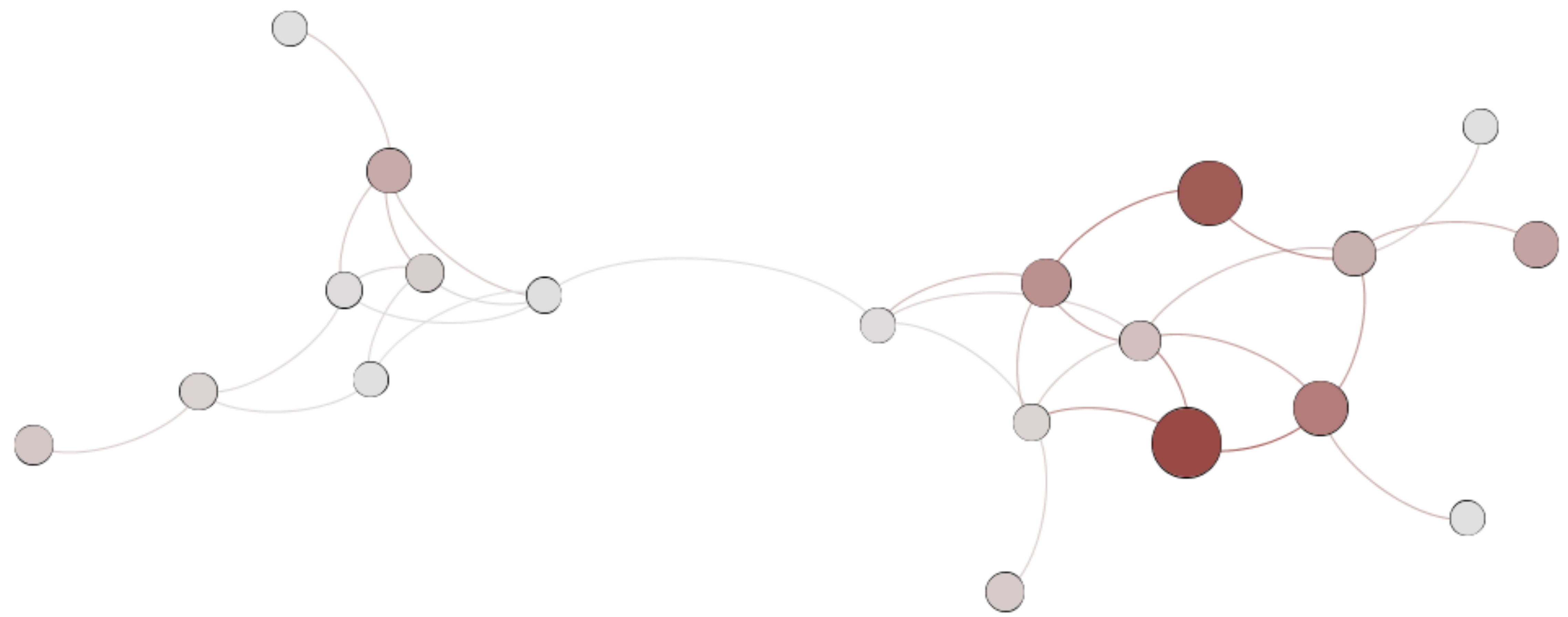
Eigenvector centrality

Importance of a node depends on the importance of its neighbors
(recursive definition)

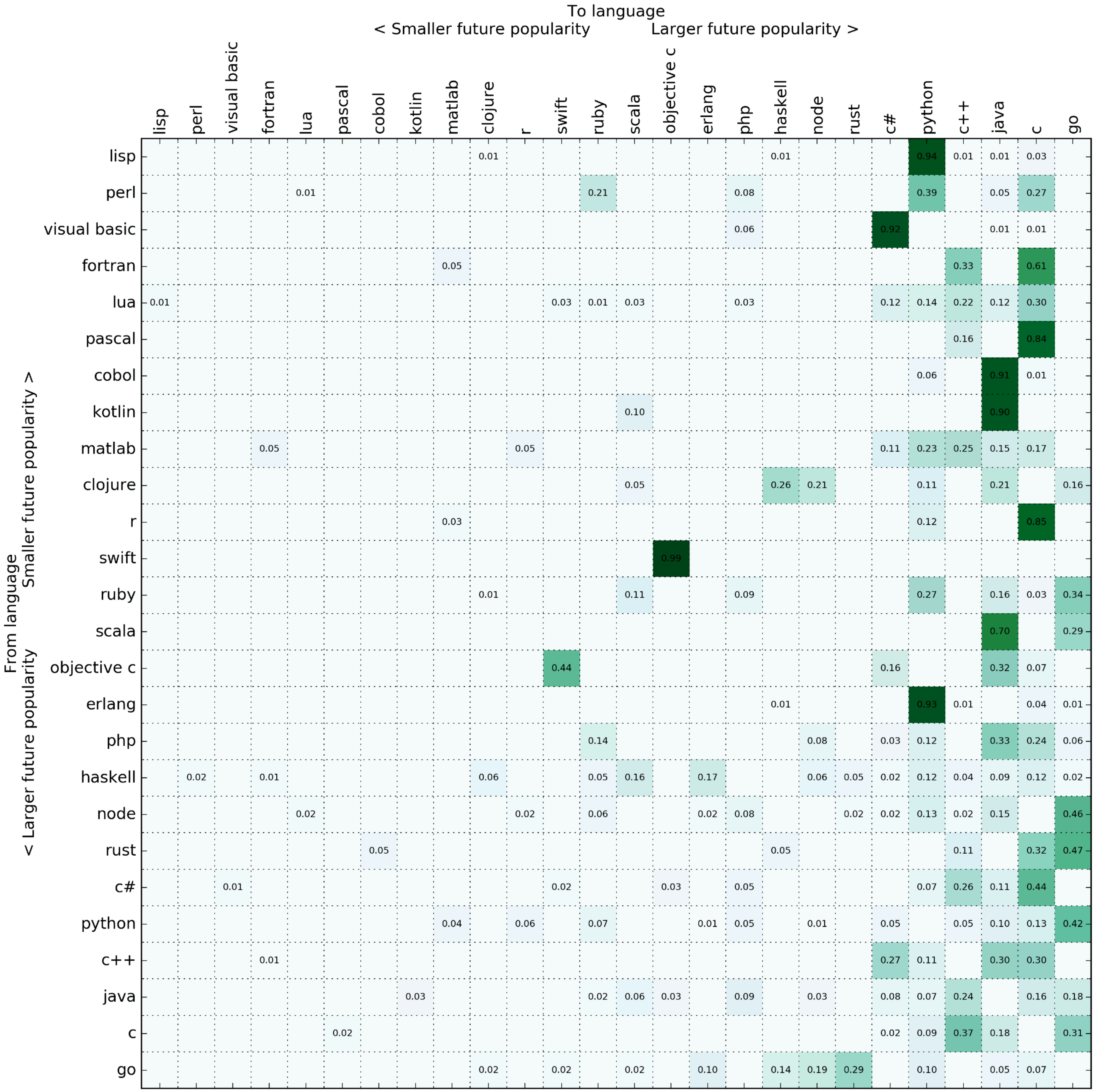
$$v_i \leftarrow \sum_j A_{ij} v_j$$

$$v_i = \frac{1}{\lambda} \sum_j A_{ij} v_j$$

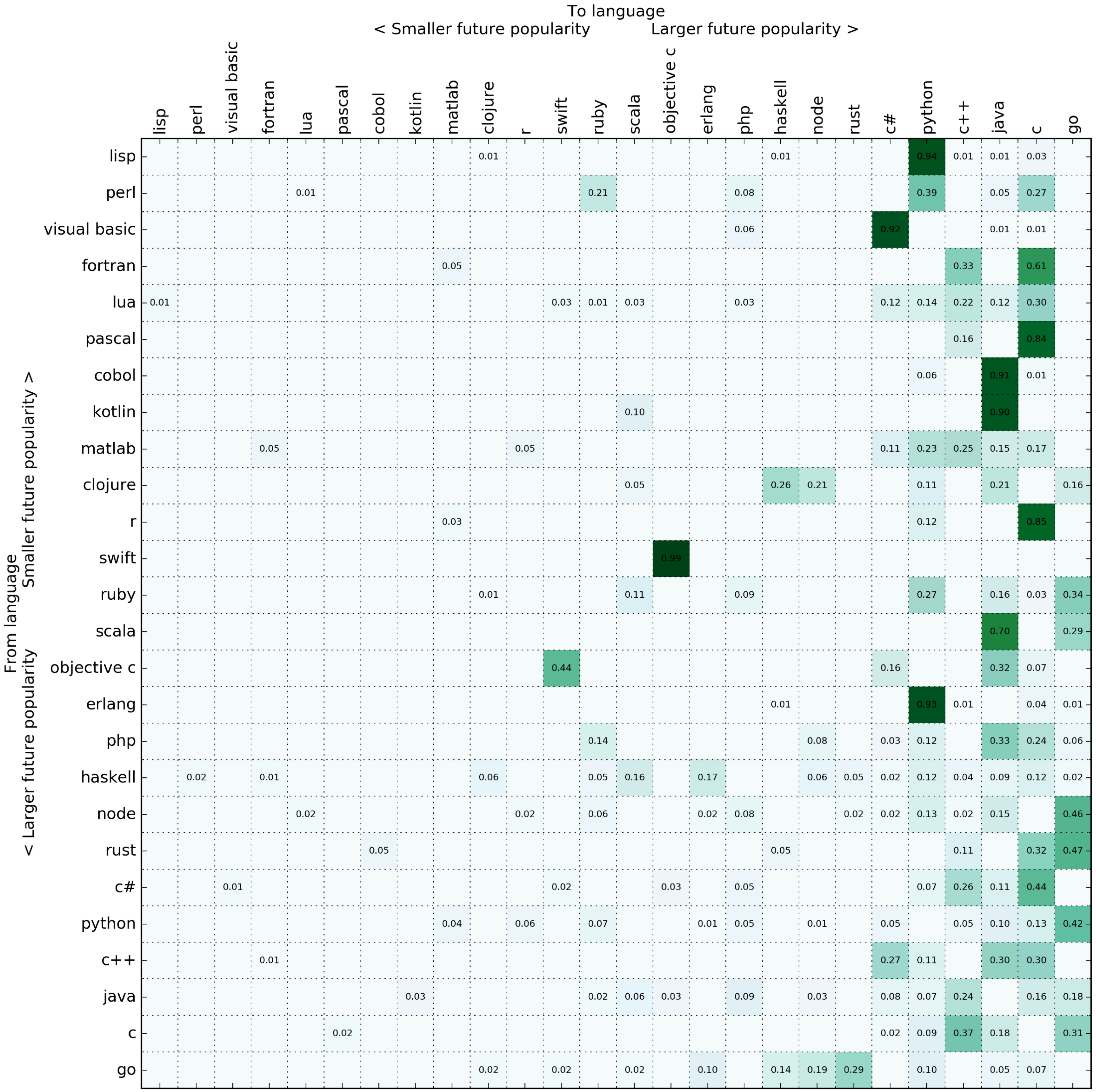
$$\mathbf{A}\mathbf{v} = \lambda \mathbf{v}$$



Select an eigenvector associated with largest eigenvalue $\lambda = \lambda_1$, $\mathbf{v} = \mathbf{v}_1$

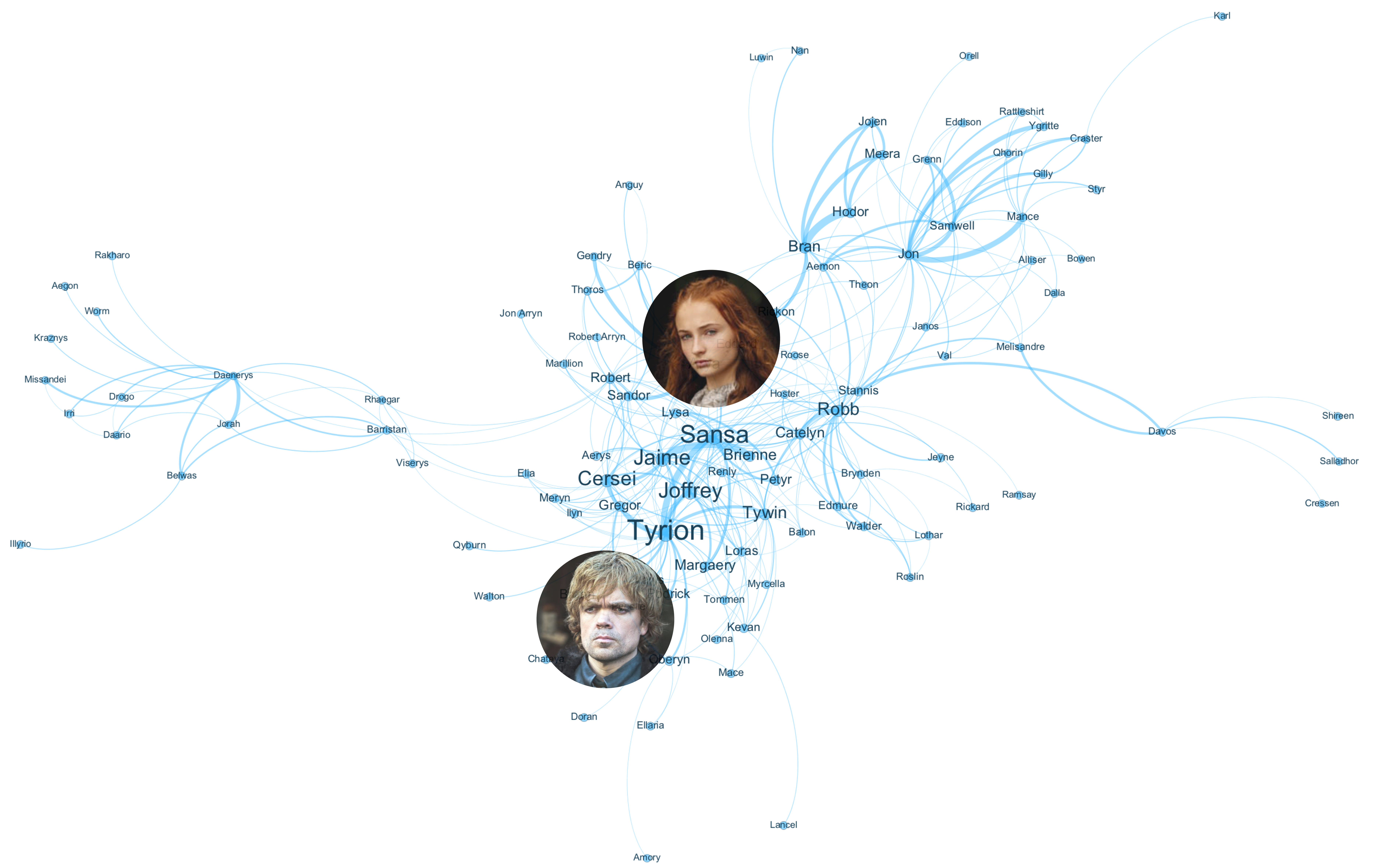


Erik Bernhardsson



Go

Erik Bernhardsson



Karl
Nan
Luwin
Orell
Jojen
Eddison
Rattleshirt
Ygritte
Craster
Meera
Grenn
Qhorin
Gilly
Styr
Anguy
Hodor
Mance
Samwell
Bran
Jon
Gendry
Beric
Aemon
Alliser
Bowen
Thoros
Theon
Dalla
Jon Arryn
Rickon
Janos
Robert Arryn
Roose
Val
Melisandre
Marillion
Rakharo
Aegon
Worm
Kraznys
Missandei
Daenerys
Drogo
Irri
Daario
Jorah
Rhaegar
Barristan
Viserys
Belwas
Illyrio
Robert
Sandor
Hoster
Stannis
Robb
Lysa
Sansa
Catelyn
Shireen
Davos
Salladhor
Cressen
Aerys
Jaime
Brienne
Jeyne
Elia
Cersei
Renly
Petyr
Brynden
Meryn
Gregor
Joffrey
Ilyn
Edmure
Rickard
Ramsay
Tywin
Walder
Lothar
Qyburn
Tyrion
Balon
Loras
Margaery
Roslin
Myrcella
Walton
Tommen
Kevan
Olenna
Oberyn
Mace
Doran
Ellaria
Lancel
Amory

Page Rank

Random walk on graph

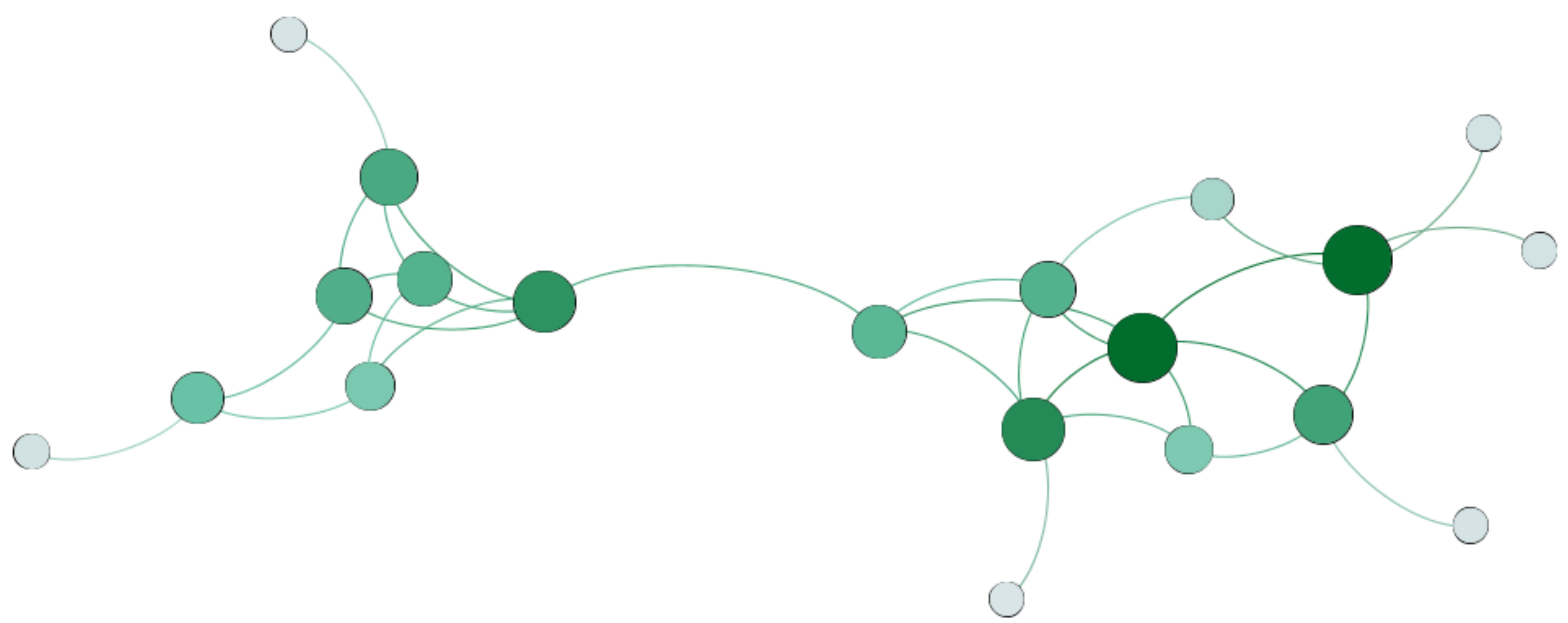
$$p_i^{t+1} = \sum_{j \in N(i)} \frac{p_j^t}{d_j^{out}} = \sum_j \frac{A_{ji}}{d_j^{out}} p_j$$

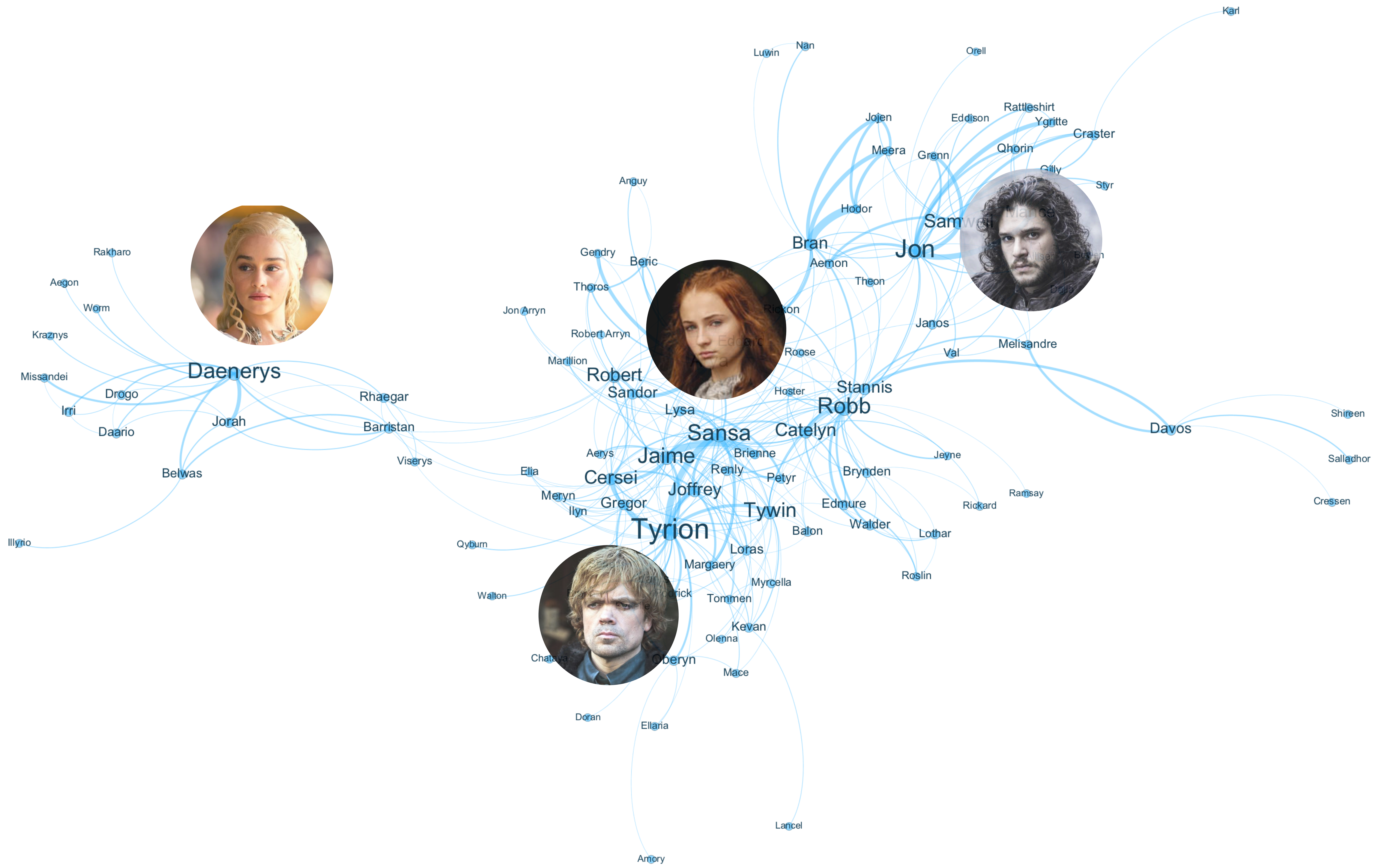
$$\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}, \quad \mathbf{D}_{ii} = diag\{d_i^{out}\}$$

$$\mathbf{p}^{t+1} = \mathbf{P}^T \mathbf{p}^t$$

with teleportation

$$\mathbf{p}^{t+1} = \alpha \mathbf{P}^T \mathbf{p}^t + (1-\alpha)\frac{\mathbf{e}}{n}$$

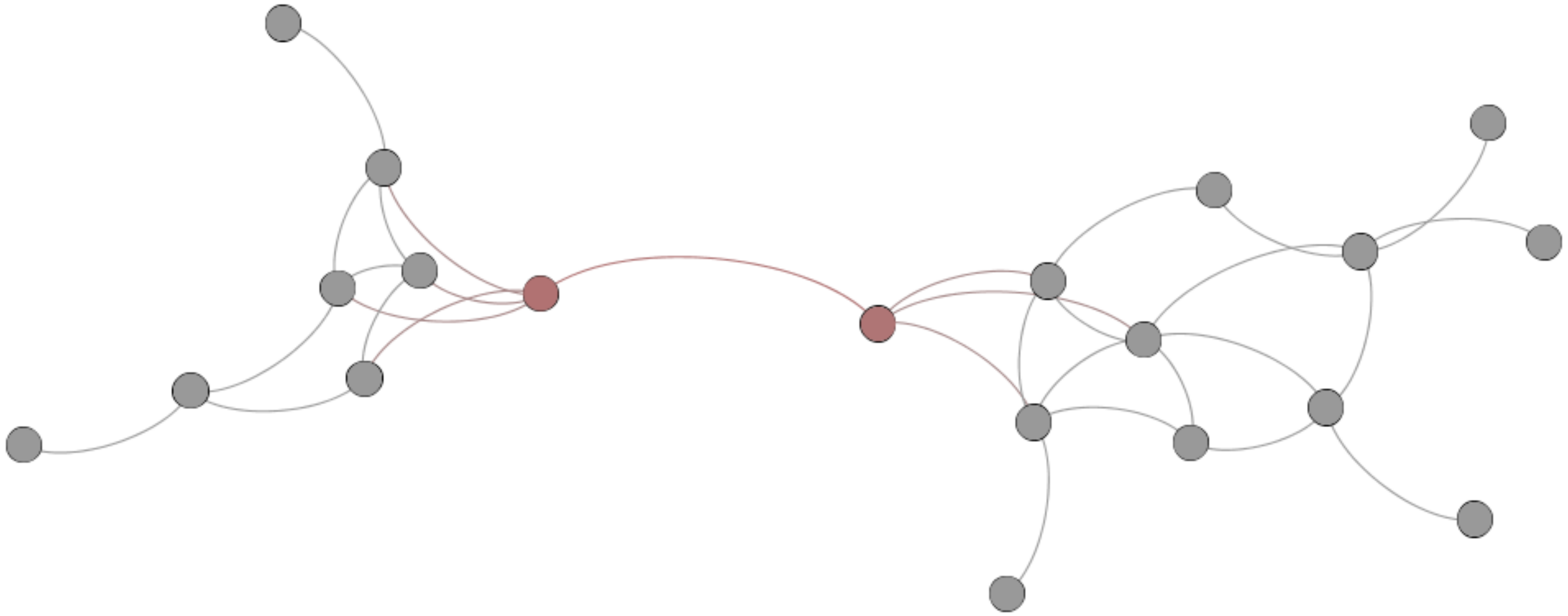



Karl
Luwin
Nan
Orell
Jojen
Eddison
Rattleshirt
Ygritte
Craster
Meera
Grenn
Qhorin
Gilly
Styr
Hodor
Sam
Bran
Jon
Aemon
Theon
Anguy
Gendry
Beric
Thoros
Jon Arryn
Robert Arryn
Roose
Janos
Val
Melisandre
Marillion
Robert
Sandor
Hoster
Stannis
Robb
Lysa
Sansa
Catelyn
Davos
Shireen
Salladhor
Cressen
Jeyne
Aerys
Jaime
Brienne
Renly
Petyr
Brynden
Ramsay
Rickard
Elia
Cersei
Meryn
Ilyn
Gregor
Joffrey
Tywin
Edmure
Walder
Lothar
Balon
Tyrion
Loras
Margaery
Myrcella
Tommen
Roslin
Kevan
Olenna
Oberyn
Mace
Qyburn
Walton
Doran
Ellaria
Lancel
Amory
Rakharo
Aegon
Worm
Kraznys
Missandei
Daenerys
Drogo
Irri
Daario
Jorah
Rhaegar
Barristan
Viserys
Belwas
Illyrio

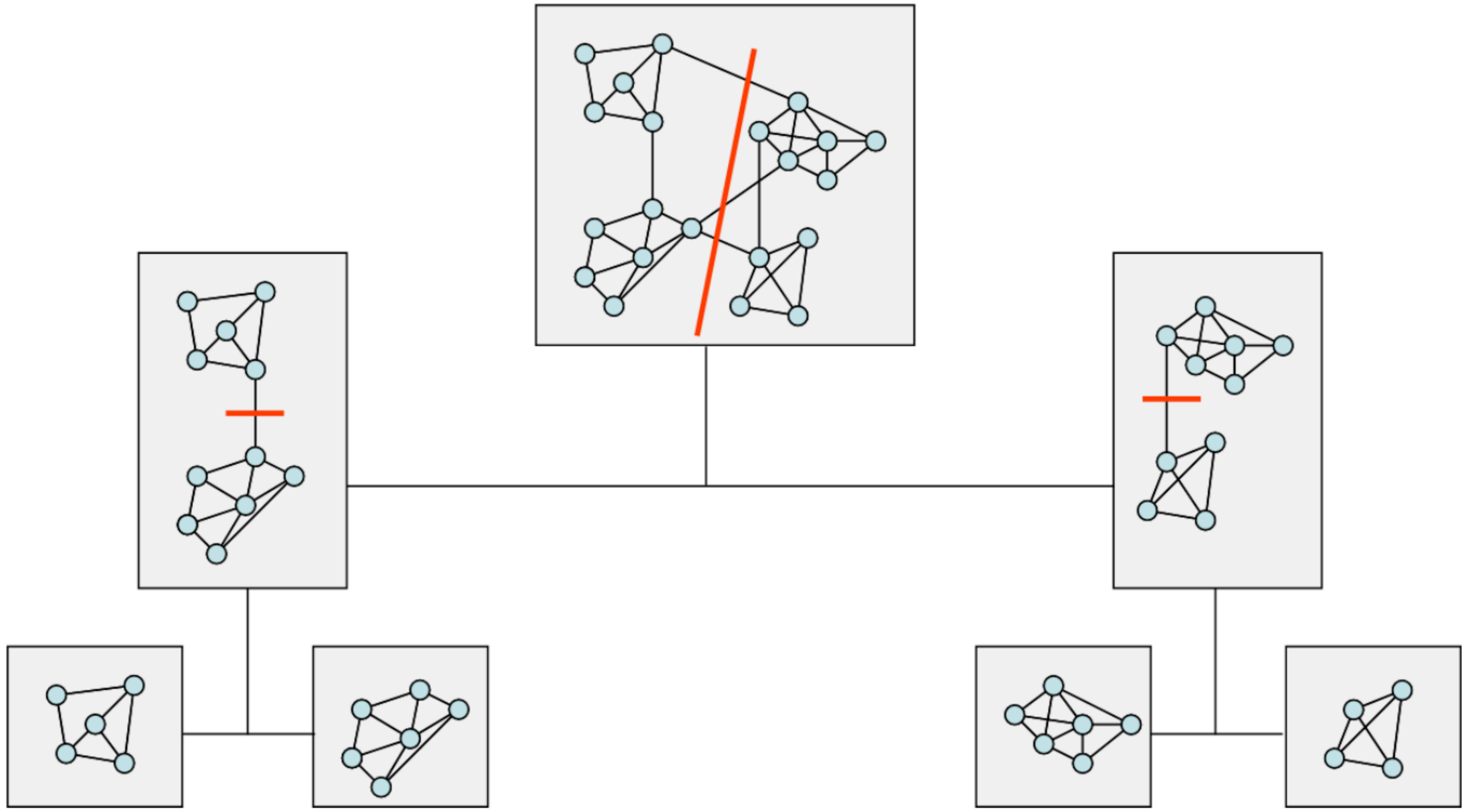
Edge betweenness

Focus on edges that connect communities.
Edge betweenness -number of shortest paths $\sigma_{st}(e)$ going through edge e

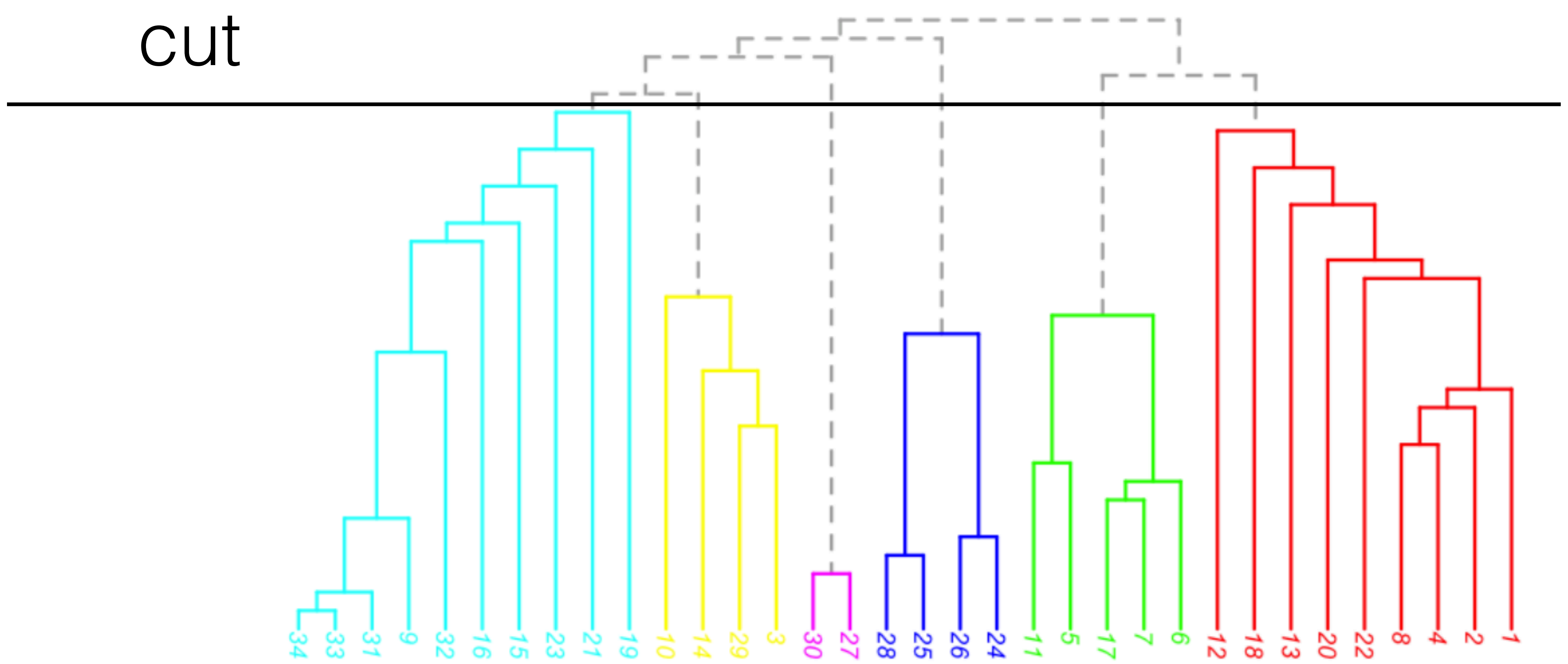
$$C_B(e) = \sum_{s \neq t} \frac{\sigma_{st}(e)}{\sigma_{st}}$$



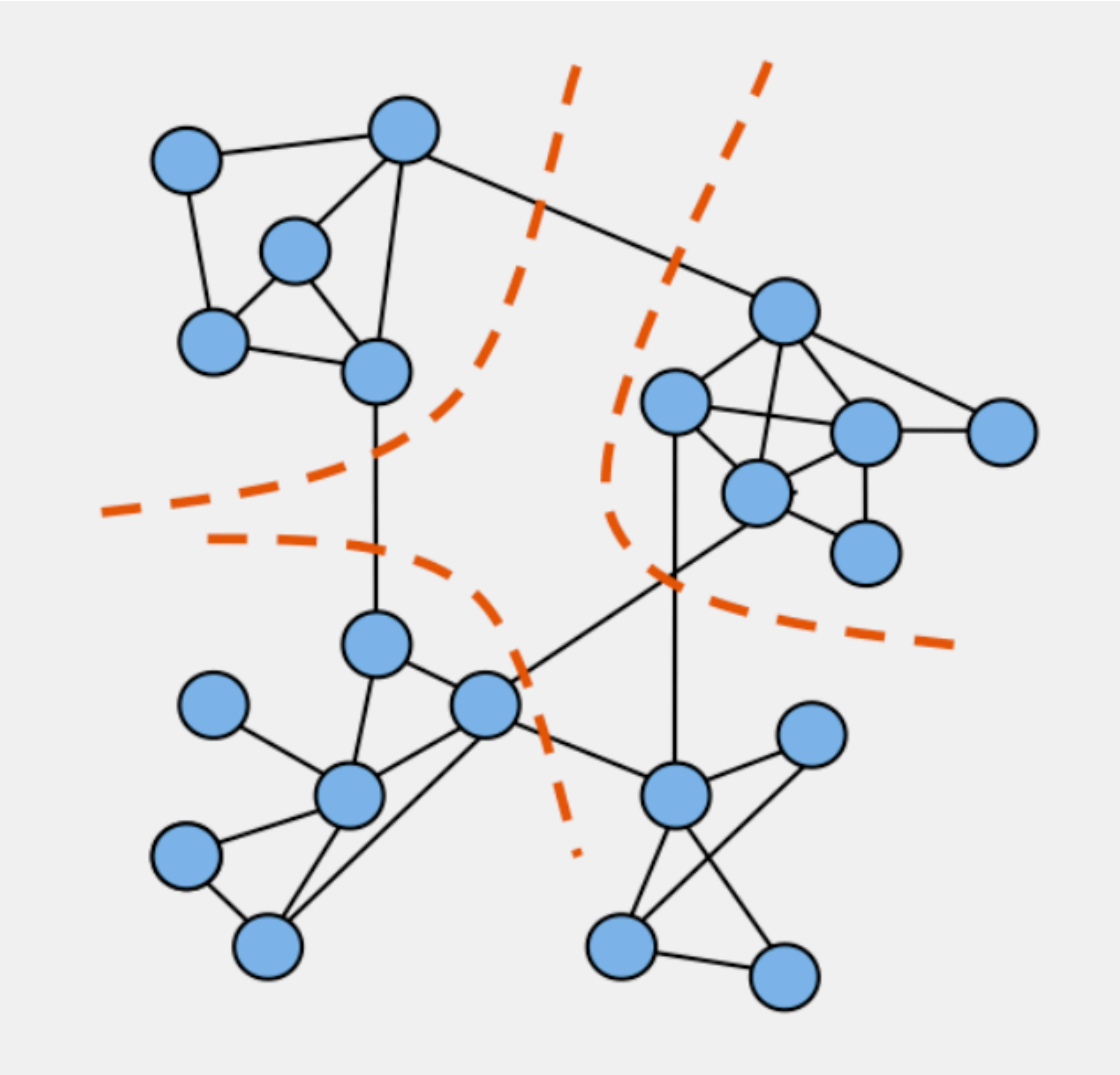
Edge betweenness



recursive partitioning



Modularity



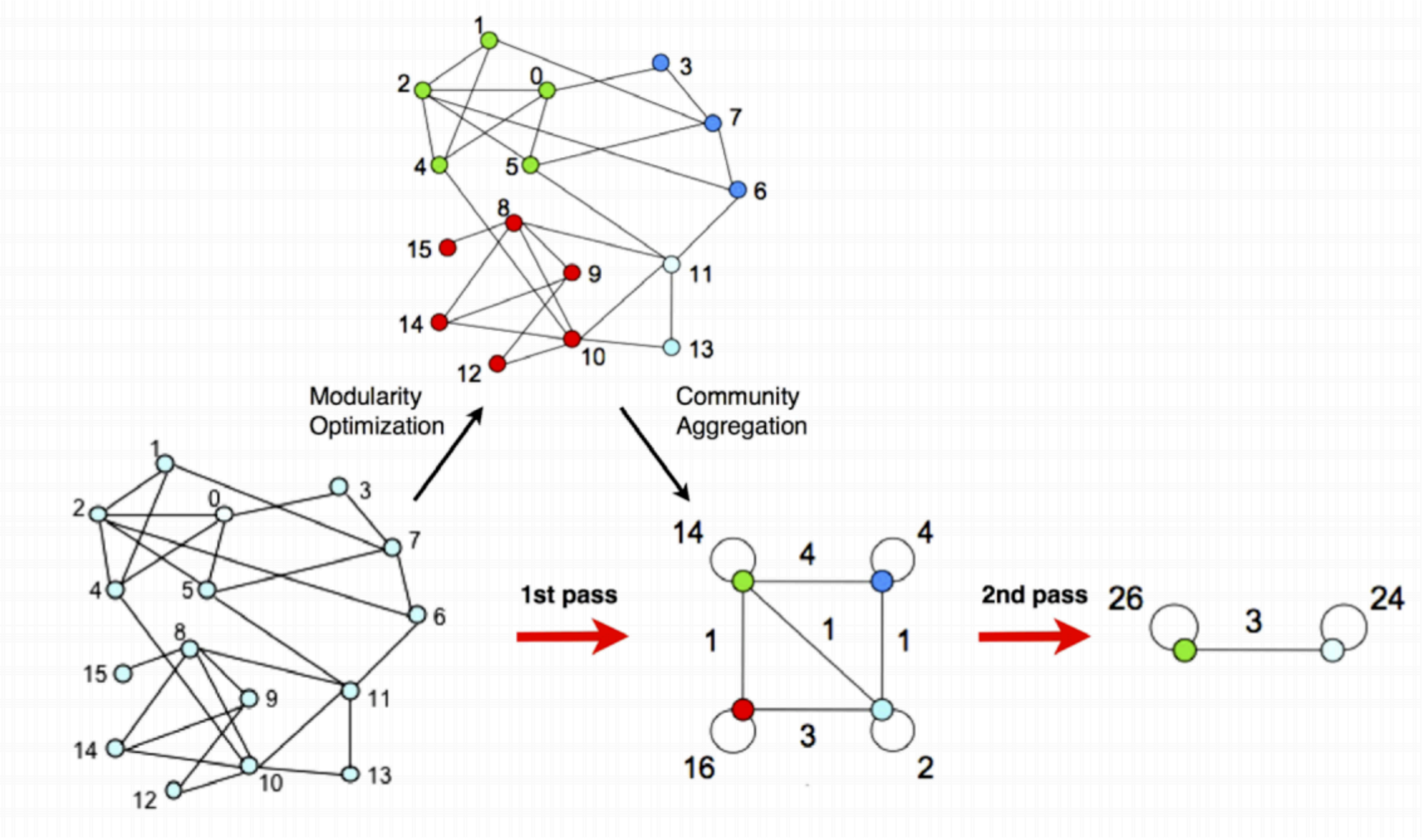
Compare fraction of edges within the cluster to expected fraction in random graph with identical degree sequence

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j),$$

The higher the modularity score - the better are communities

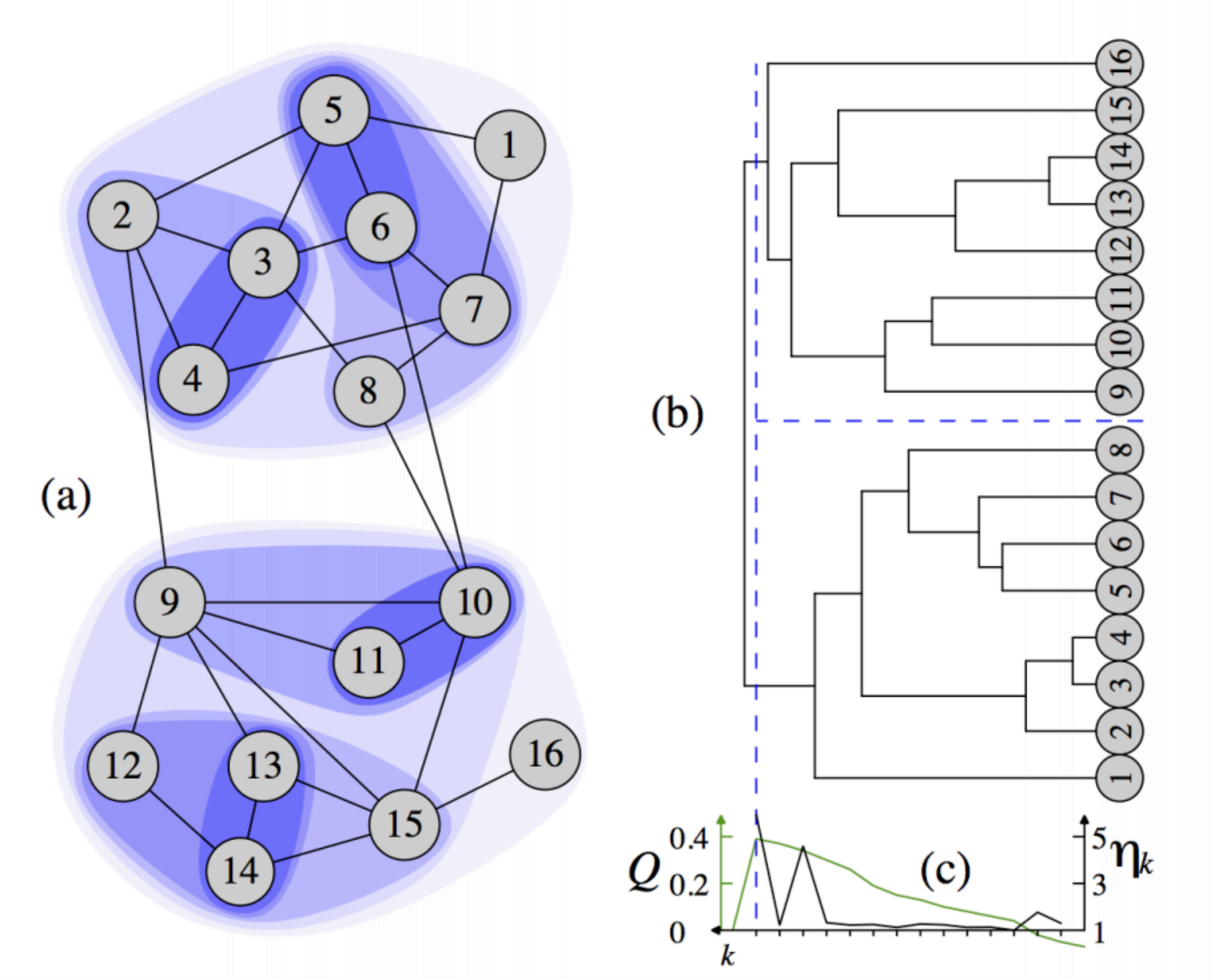
Modularity score range $Q \in [-1/2, 1)$, single community $Q = 0$

Fast community unfolding

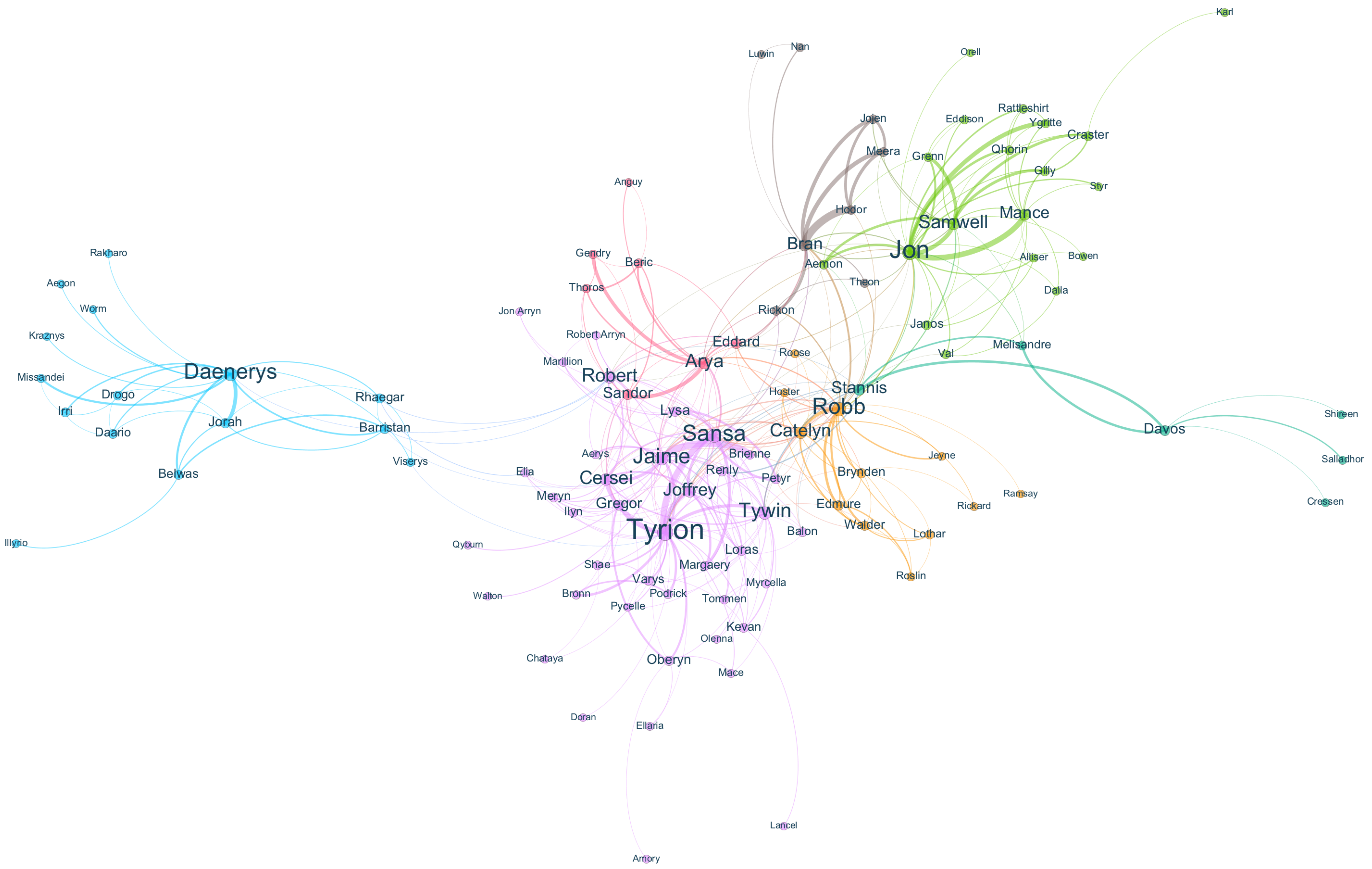


V. Blondel et.al., 2008

Walktrap



P. Pons and M. Latapy, 2006



Karl
Nan
Luwin
Orell
Jojen
Eddison
Rattleshirt
Ygritte
Craster
Meera
Grenn
Qhorin
Gilly
Styr
Anguy
Hodor
Samwell
Mance
Bran
Jon
Gendry
Beric
Aemon
Alliser
Bowen
Thoros
Theon
Dalla
Jon Arryn
Rickon
Rakharo
Aegon
Worm
Kraznys
Robert Arryn
Janos
Eddard
Melisandre
Marillion
Roose
Val
Missandei
Daenerys
Robert
Arya
Stannis
Drogo
Rhaegar
Sandor
Hoster
Robb
Shireen
Irri
Jorah
Lysa
Daario
Barristan
Sansa
Catelyn
Davos
Aerys
Brienne
Jeyne
Sallador
Jaime
Belwas
Viserys
Elia
Renly
Brynden
Cersei
Petyr
Ramsay
Meryn
Joffrey
Rickard
Cressen
Ilyn
Gregor
Edmure
Tywin
Walder
Lothar
Tyrion
Balon
Illyrio
Qyburn
Loras
Shae
Margaery
Roslin
Varys
Myrcella
Walton
Bronn
Podrick
Tommen
Pycelle
Kevan
Olenna
Chataya
Oberyn
Mace
Doran
Ellaria
Lancel
Amory

Pipeline

Data

Preprocess

Top Nodes

Communites

Filter

Visualization


DEMO