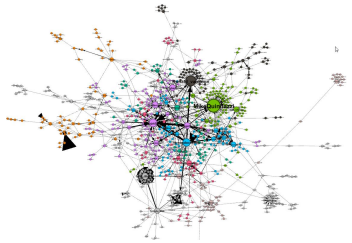
EINSTEIN CENTER Digital Future

Freie Universität Berlin

Department of Information Systems

School of Business & Economics



PhD Course on Social Network Analysis

Prof. Dr. Daniel Fürstenau

20 February 2017

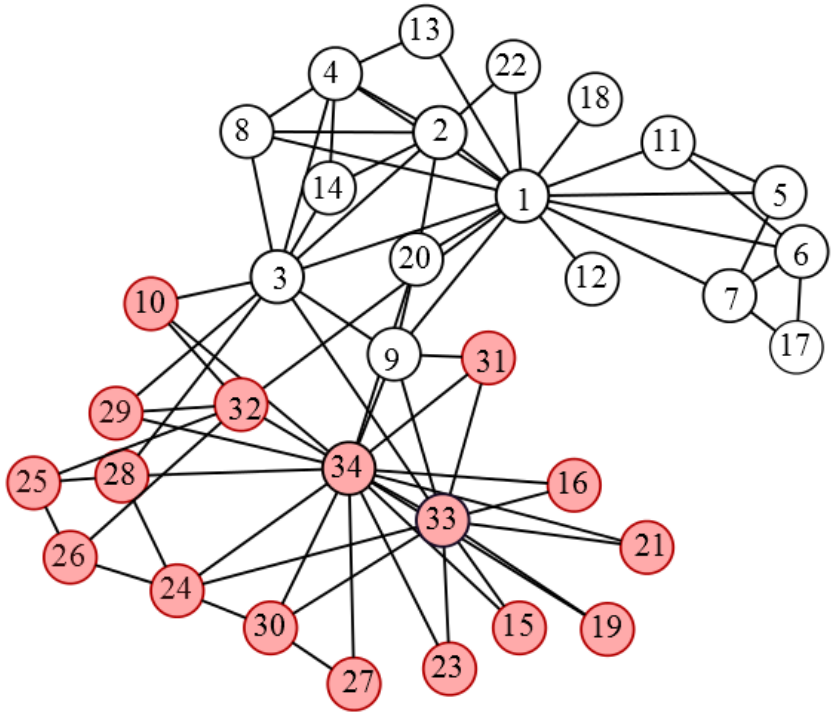
https://github.com/dfurstenau/SNA_class

Agenda

1. **Introduction and examples**
2. Fundamental concepts
    1. Creating graphs
    2. Visualizing graphs
    3. Centrality
    4. Describing the macro structure
    5. Community detection
3. Getting and handling data
4. Extended concepts
5. Discussion and conclusion

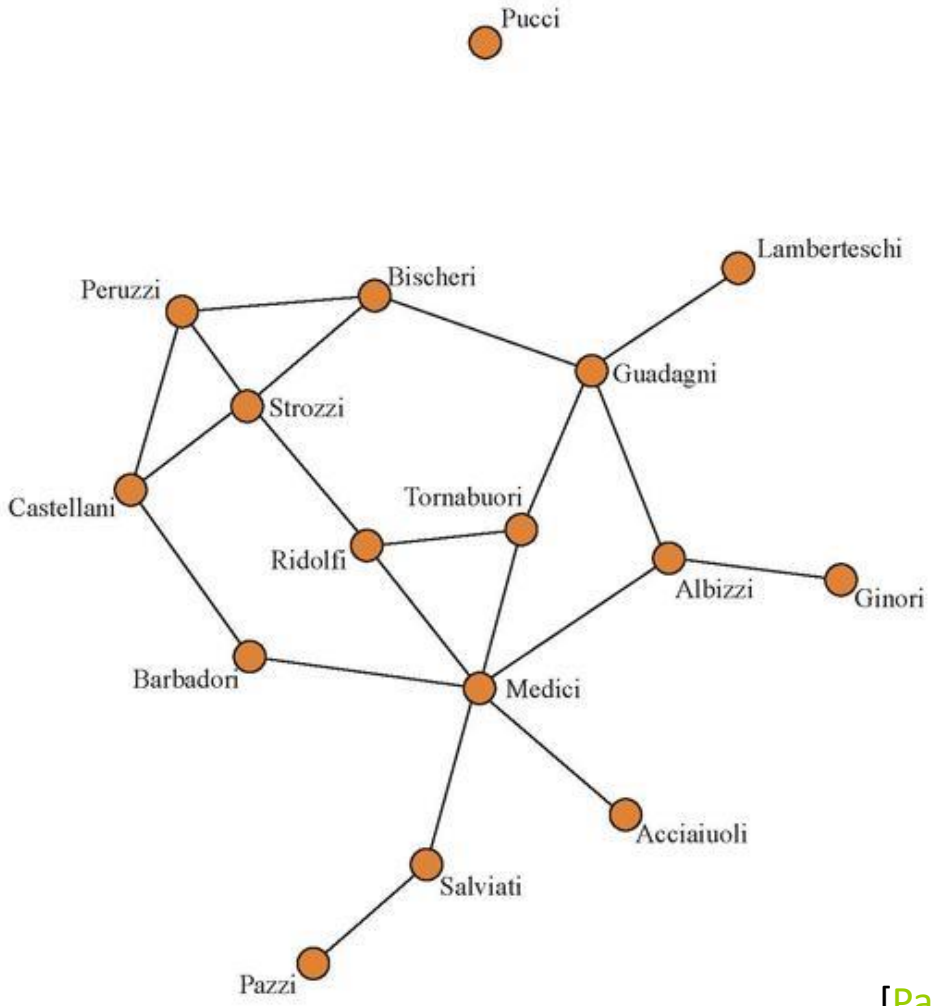
Introduction 1: Zachary‘s Karate Club Network

The Iris dataset of social network analysis



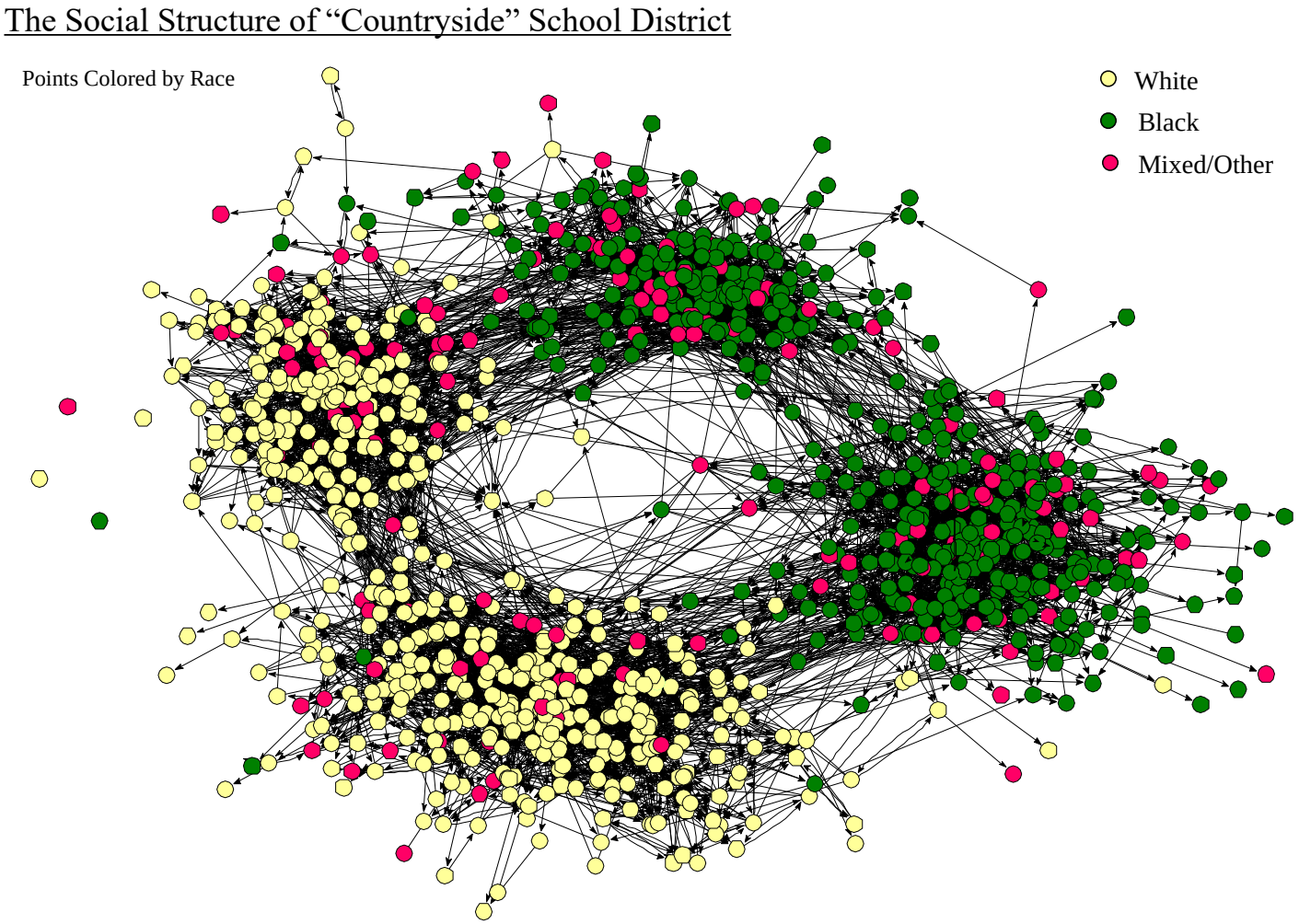
A social network of a karate club was studied by Wayne W. Zachary for a period of three years from 1970 to 1972. The network captures 34 members of a karate club, documenting 78 pairwise links between members who interacted **outside** the club. During the study a conflict arose which led to the split of the club into two. Based on collected data Zachary assigned correctly all but one member of the club to the groups they actually joined after the split.

Introduction 2: 15th Century Florentine Marriages



[Padgett and Ansell, 1993]

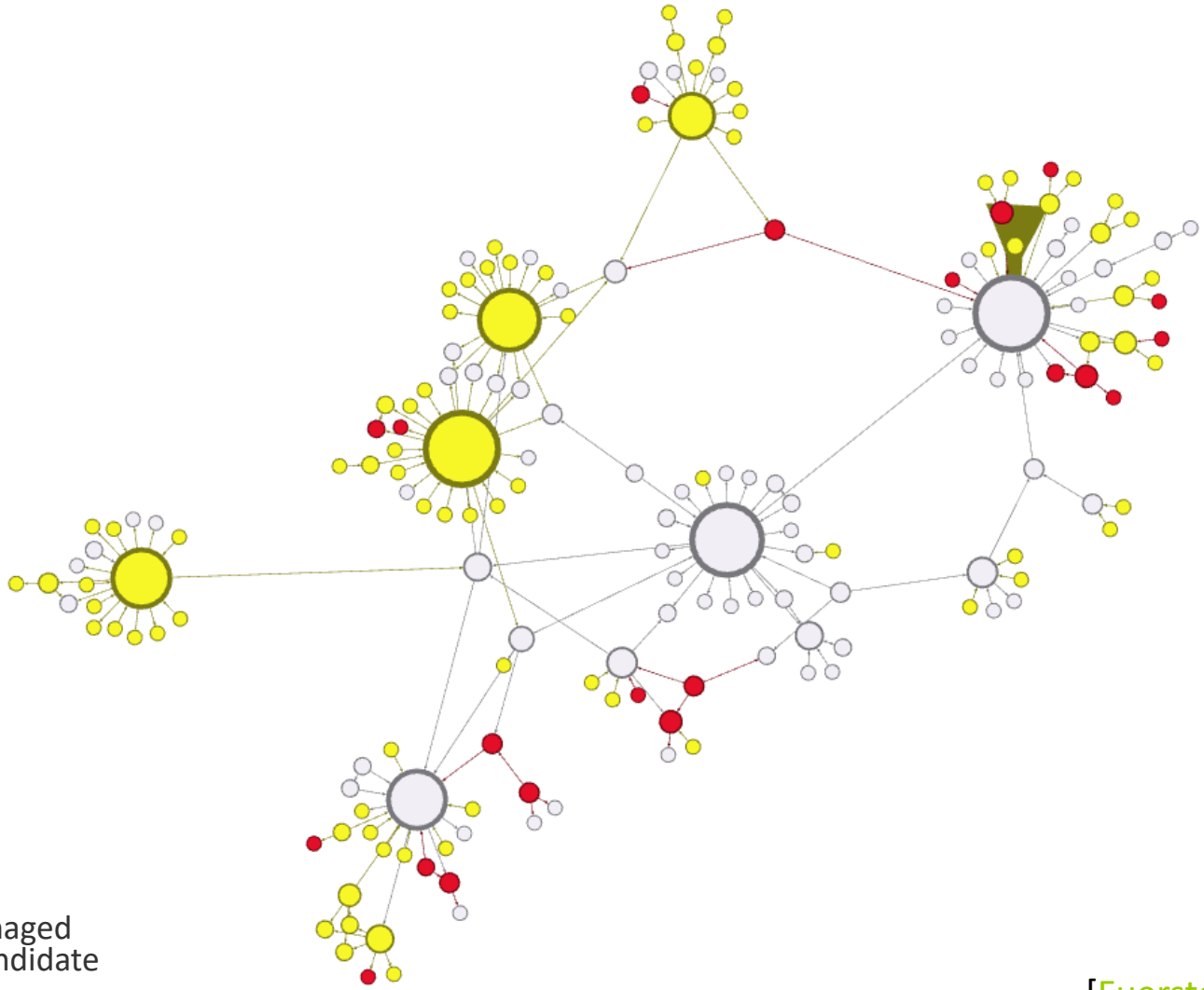
Introduction 3: Add Health School Friendship Network



Introduction 4: Retweet network from Twitter

[https://www.slideshare.net/giladlotan/pydata-gilad-networkxupload]

Introduction 5: Recyle Inc. Information Infrastructure Network



Centrally managed
Shadow IT candidate
Shadow IT

[Fuerstenau et al. 2014]

Definitions

Network: a pattern of interconnections among a set of things

Social Network: a network where the things are people and the interconnections are social interactions

Social Network Analysis (SNA): the application of graph and network theory / network science to investigate social structures.

Graph theory: a branch of mathematics concerned with modeling pairwise interactions among objects

Network theory: Theorizing networks includes at least: individual elements; pair-wise relationships between those elements; and a global or macro- patterning that can be considered as network structure (Brandes et al. 2013)

Network science: The interdisciplinary, empirical, data-driven, quantitative, computational study of network models (view of Laslo Barabasi).

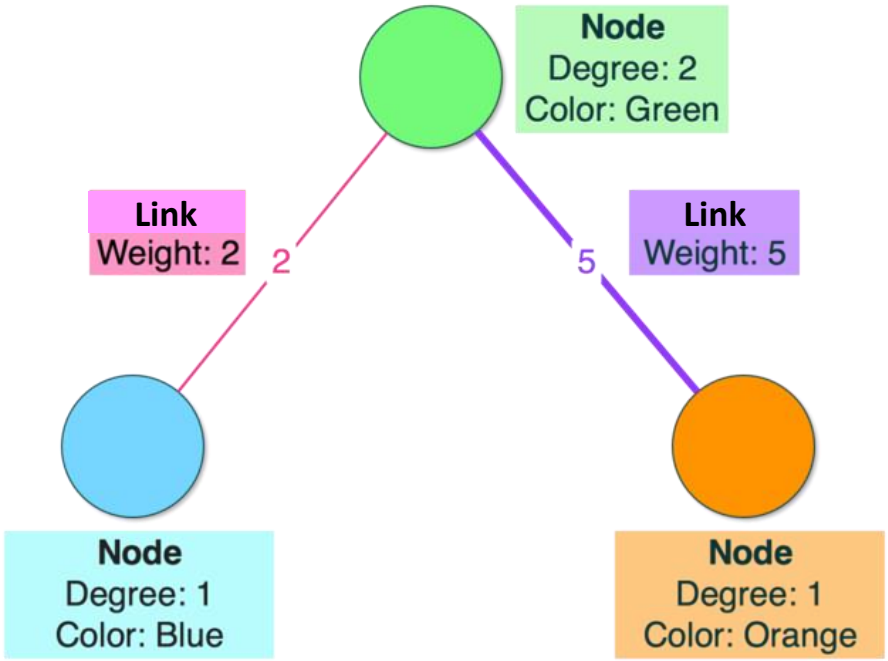
Types of social interactions




Similarities			Social Relations				Interactions	Flows
Location	**Membership**	**Attribute**	**Kinship**	**Other role**	**Affective**	**Cognitive**	e.g.,	e.g.,
e.g.,	e.g.,	e.g.,	e.g.,	e.g.,	e.g.,	e.g.,	Sex with	Information
Same spatial and temporal space	Same clubs	Same gender	Mother of	Friend of	Likes	Knows	Talked to	Beliefs
	Same events	Same attitude	Sibling of	Boss of	Hates	Knows about	Advice to	Personnel
	etc.	etc.		Student of	etc.		Helped	Resources
				Competitor of		Sees as happy	Harmed	etc.
						etc.	etc.	

[Borgatti et al. 2009]

Parts of a network: Nodes and links



Node / Vertex: The entity of analysis which has a relationship. Node is used in the network context, vertex is used in the graph theory context, but both terms are often used interchangeably.

Link / Edge / Relationship: The connections between the nodes. Link is used in the network context, edge is used in the graph theory context, and all words are used interchangably with *relationship*.

Attributes: Both nodes and links can store attributes, which contain additional data about that object.

Weight: A common *attribute* of links, used to indicate *strength* or *value* of a relationship.

Degree: Number of links a node has.

Types of networks

Networks are typically classified based on the presence of **weights** and **direction** attached to the links in a graph. The table below covers what we call each type of graph:

	Absent	Present
Weights	Unweighted	Weighted
Directionality	Undirected	Directed

Additional flavors: parallel links, self-loops, *n*-partite graphs

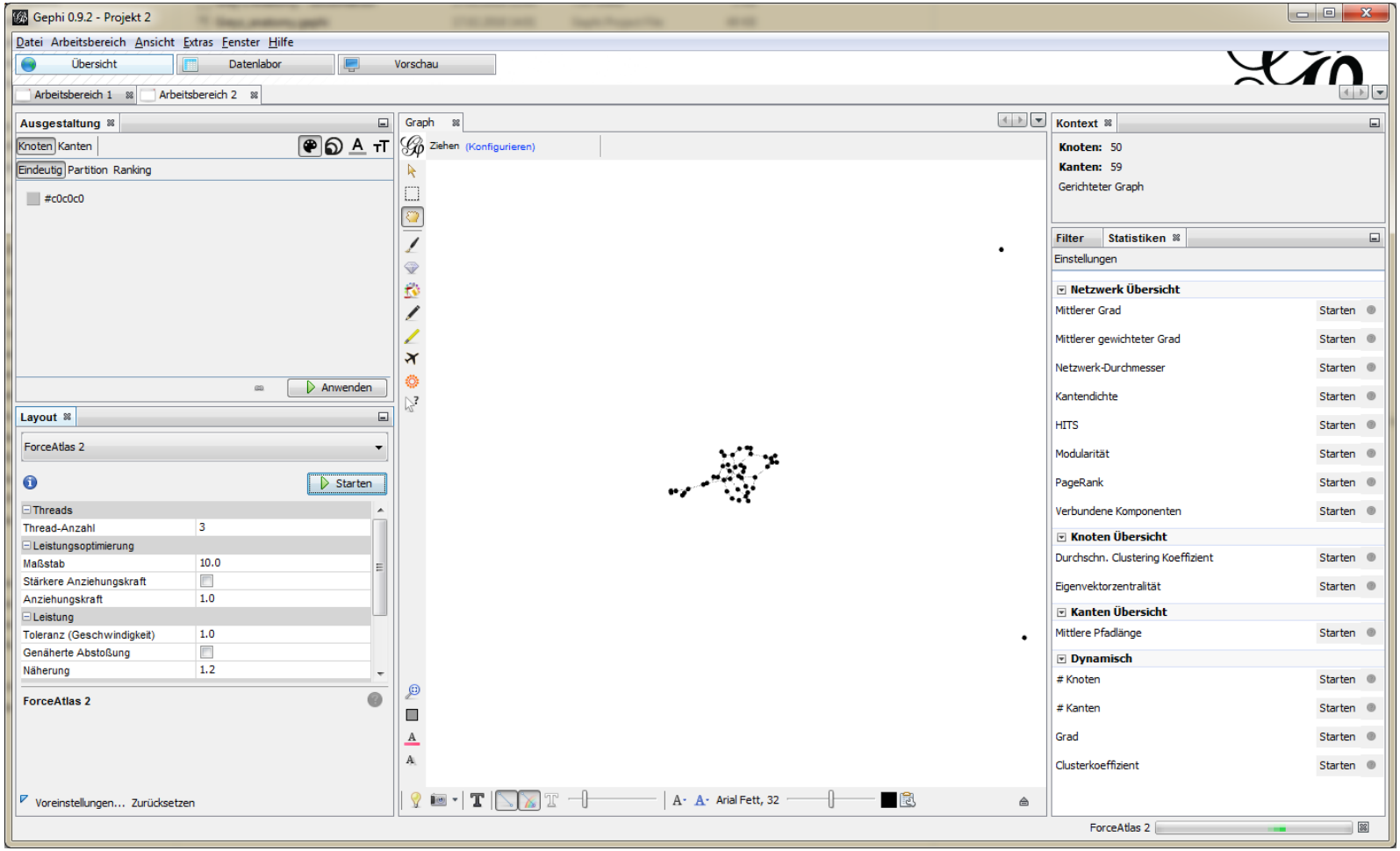
In this context:

We are talking about a(n) **[unweighted/weighted] [undirected/directed]** graph (with **[parallel edges | self loops]**).

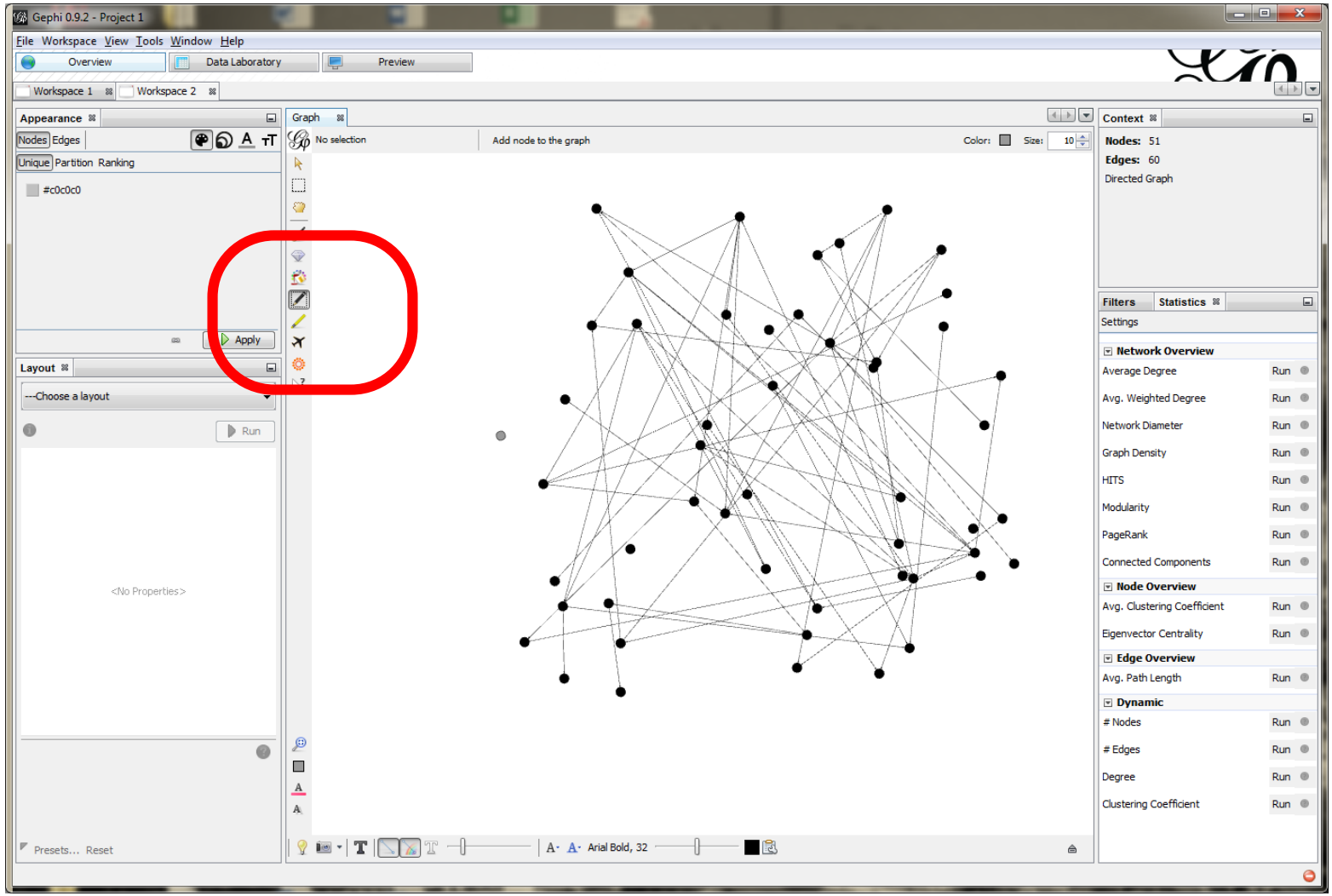
Agenda

1. Introduction and examples
2. Fundamental concepts
    1. **Creating graphs**
    2. Visualizing graphs
    3. Centrality
    4. Describing the macro structure
    5. Community detection
3. Getting and handling data
4. Extended concepts
5. Discussion and conclusion

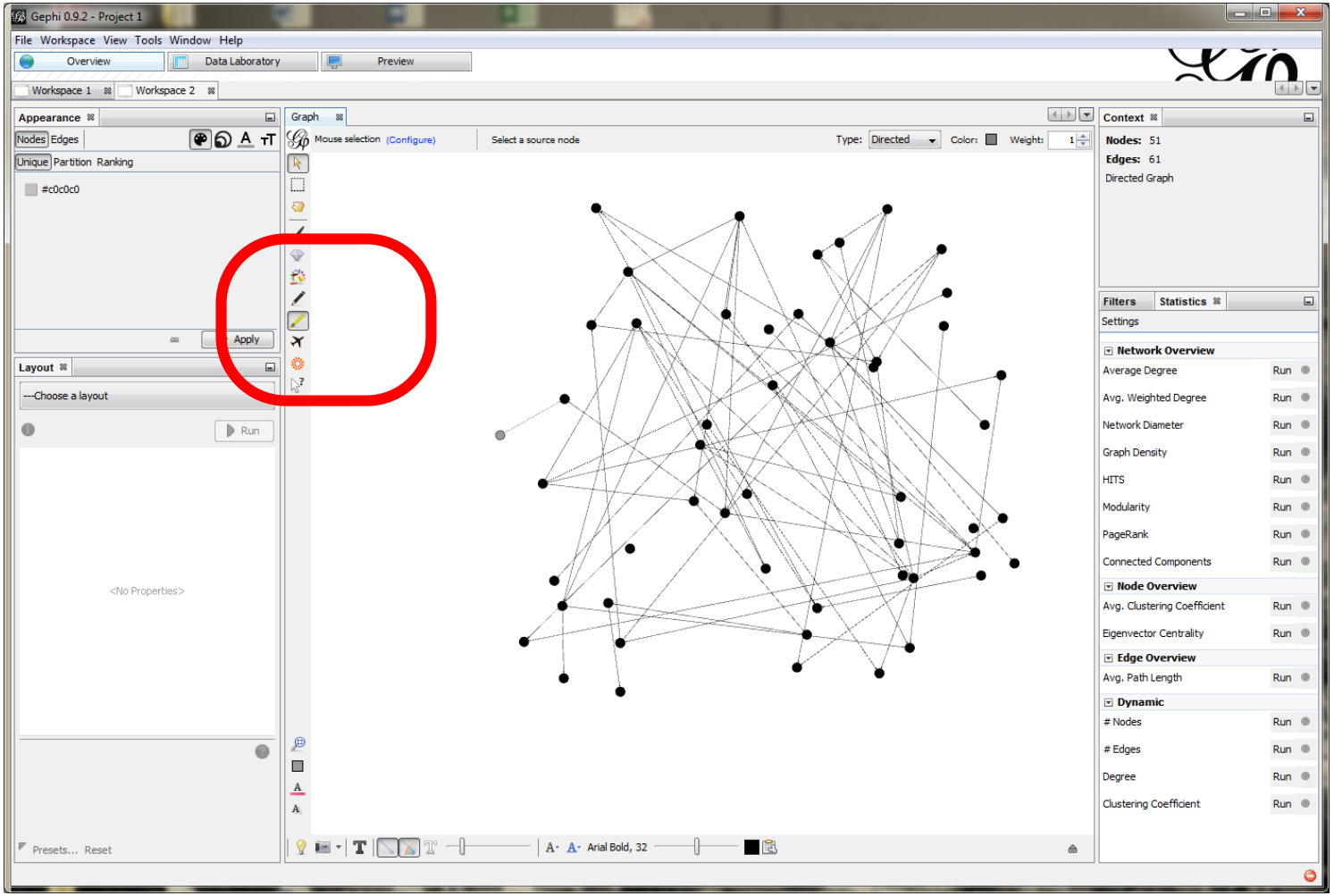
Creating a graph… in Gephi



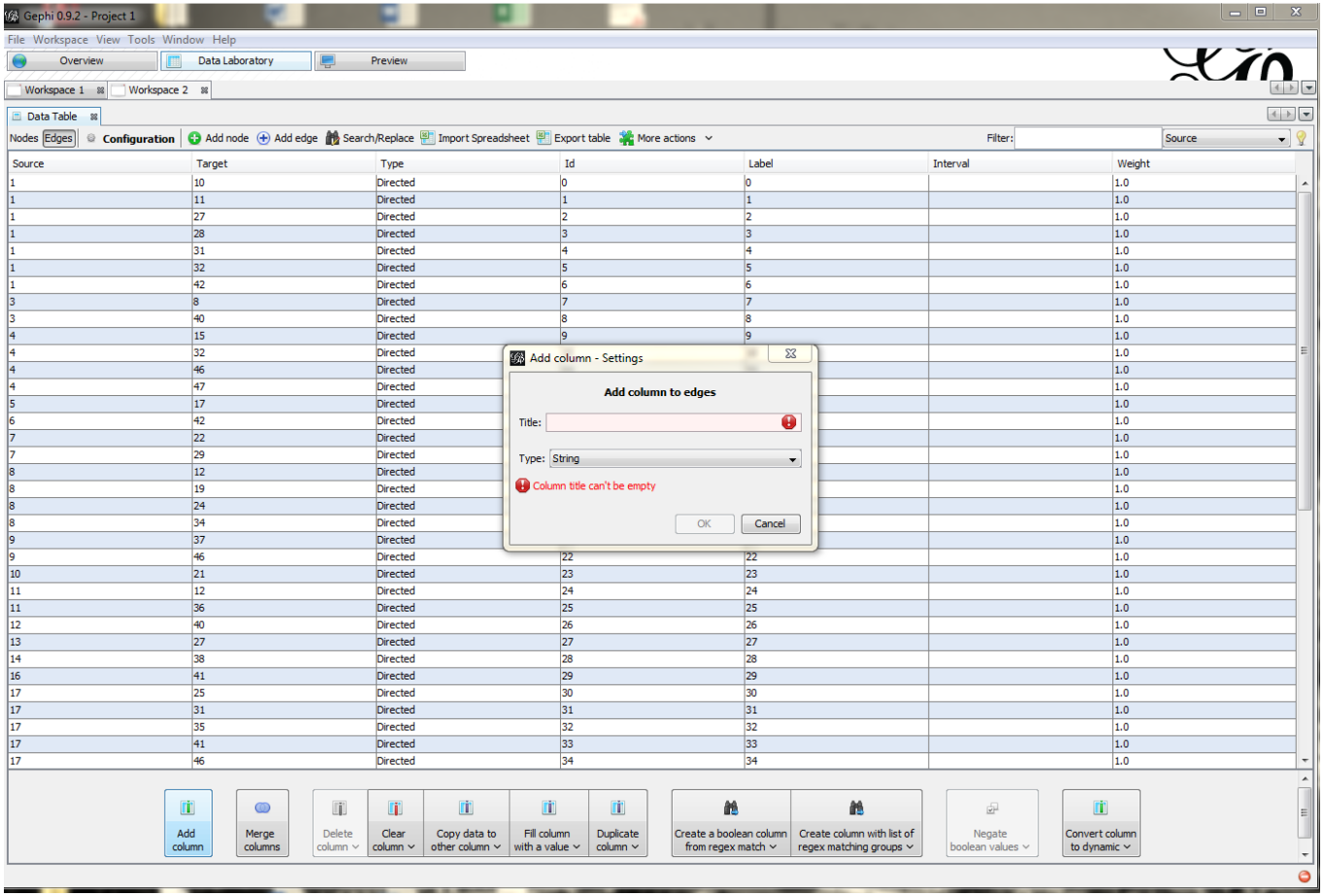
Adding nodes…



Adding links…



Adding and inspecting attributes…



This is to be done in the data laboratory.

Reading in different representations of graphs

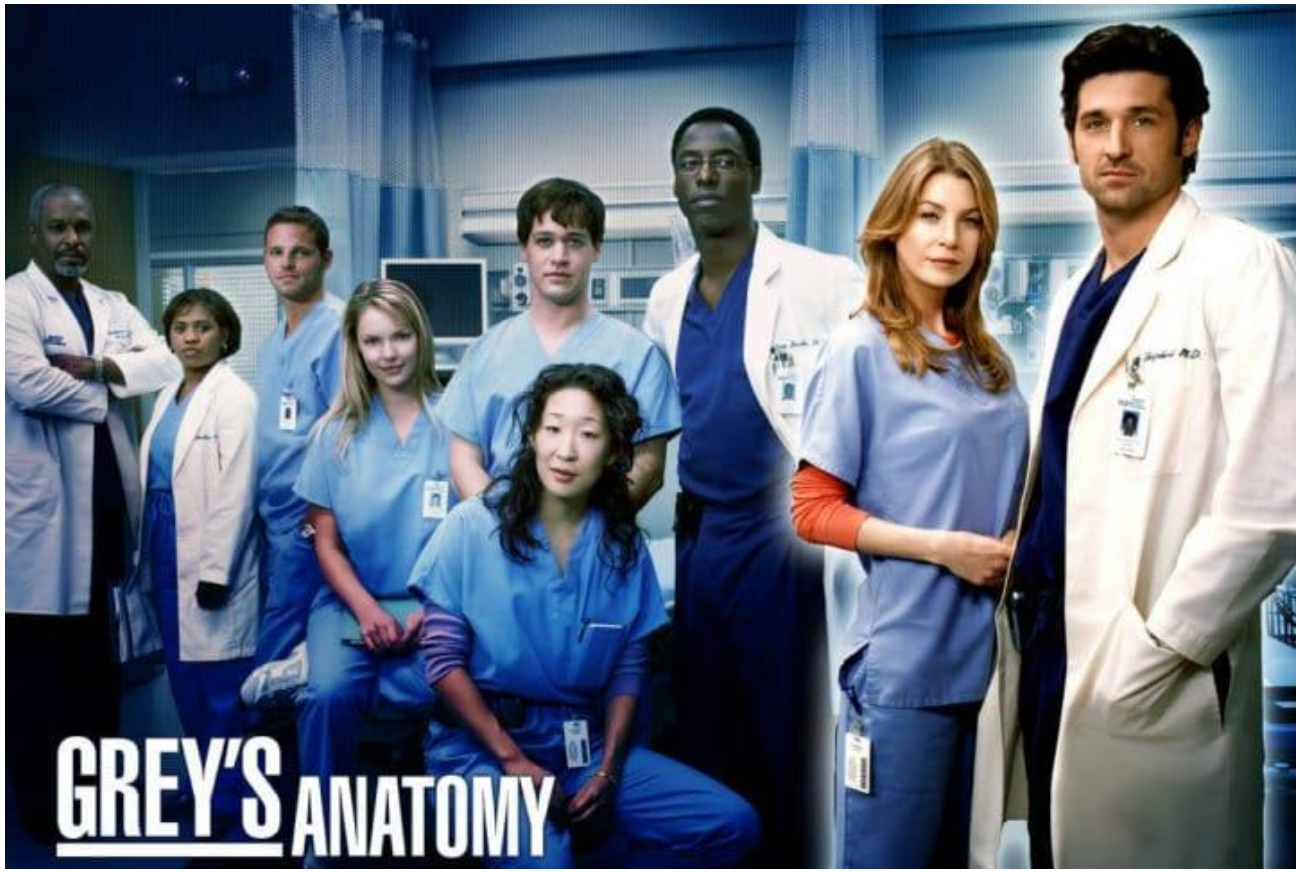
Data for graphs and networks comes in many different representations.

- **Representations:**
  - Link list
  - Adjacency matric
  - …

Note: representations are related to, but distinct from, the storage format. We will use `csv-files` for link lists and `.dl` for adjanceny matrices.

There are several other formats that we briefly touch upon such as `.gexf`, `.graphml`, and `.gephi`.

Grey‘s anatomy example data set



The dataset we'll look at is a record of all "romantic" encounters between characters on the TV show Grey's Anatomy

[Rob Chew and Peter Baumgartner]

Edge lists

An edge list (aka link list) is a common way of representing a graph. This representation can be thought of as a list of tuples, where each tuple represents an edge between two of the nodes in your graph. The nodes of the graph can be inferred by taking the set of objects from all tuples.

You can infer/determine whether a graph is directed or weighted from an edge list.

- **Weighted**: If edges appear more than once, or if an additional weight attribute is added as a 3rd column, the graph is weighted
- **Directed**: If the "From" and "To" (often seen as "Source" and "Target") of an edge in the list is not arbitrary, it's a directed graph

	A	B	C
1	Source	Target	Type
2	lexi	sloan	Undirected
3	lexi	karev	Undirected
4	owen	yang	Undirected
5	owen	altman	Undirected
6	sloan	torres	Undirected
7	sloan	altman	Undirected
8	torres	arizona	Undirected
9	torres	karev	Undirected
10	derek	grey	Undirected

01-Greys_anatomy_edgelist.csv

Adjacency matrices

A common way of representing graph data is through an `adjacency matrix` -- often referred to mathematically as `A`. This data structure is a square, `n x n` matrix where `n = number of nodes`. Each column and row in the matrix is a node. For any two nodes, `i` and `j` the value at `Aij` (row `i` and column `j`) represents the weight of the link between nodes `i` and `j`.

	denny	kepner	grey	colin	finn
denny	0.0	0.0	0.0	0.0	0.0
kepner	0.0	0.0	0.0	0.0	0.0
grey	0.0	0.0	0.0	0.0	1.0
colin	0.0	0.0	0.0	0.0	0.0
finn	0.0	0.0	1.0	0.0	0.0

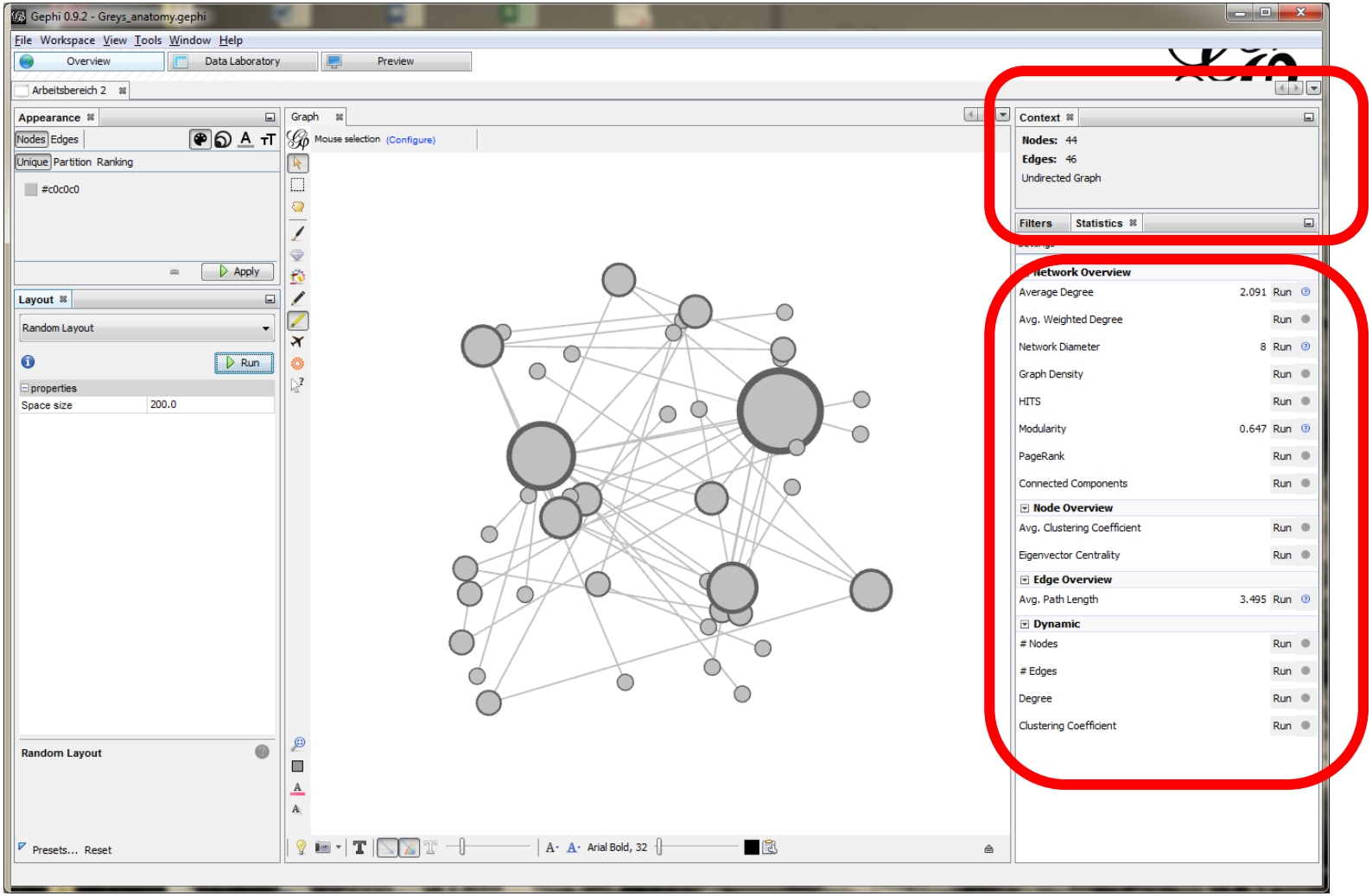
Adjacency matrices (continued): DL format

Greys_anatomy_adjacency_matrix.dl

```
dl n=44
format = fullmatrix
labels:
addison,adele,altman,ameli
data:
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0
```

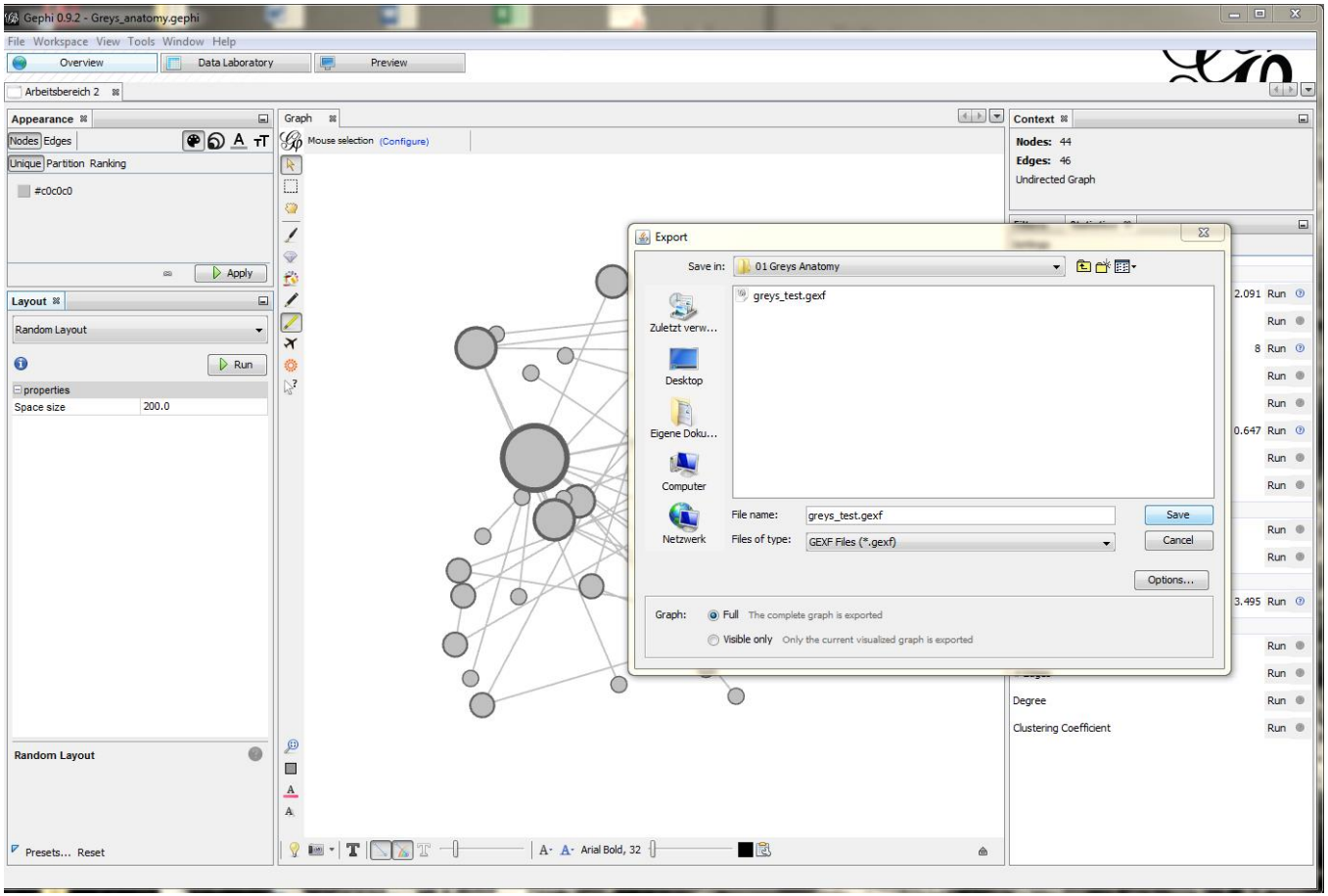
Greys_anatomy_adjacency_matrix.dl

Summary statistics on graph…



Exporting graphs

We'll export the graph in `GEXF` (Graph Exchange XML Format).



Exporting: Where to?

- **Analysis in other developer tools and packages**
  - Iphyton/Jupyter and NetworkX (my package of choice)
  - Further python packages: igraph, D3 (visualization)
  - R (SNA package etc.)
  - …
- **Analysis in other packaged software programs**
  - Ucinet
  - Pajek
  - Cytoscape
- …

Agenda

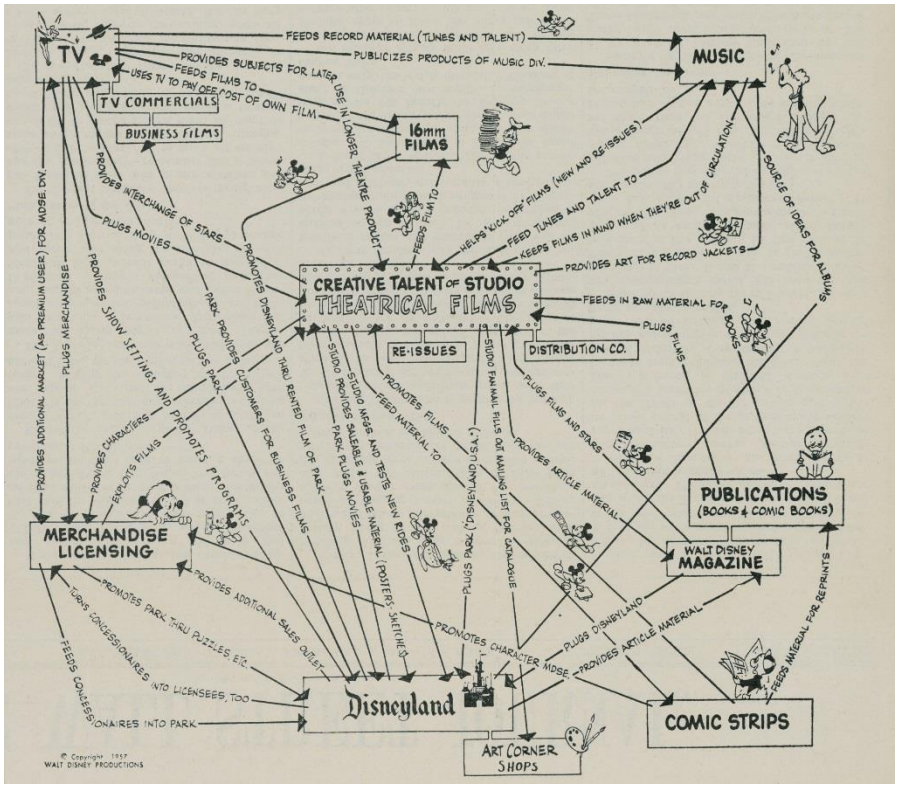
1. Introduction and examples
2. Fundamental concepts
    1. Creating graphs
    2. **Visualizing graphs**
    3. Centrality
    4. Describing the macro structure
    5. Community detection
3. Getting and handling data
4. Extended concepts
5. Discussion and conclusion

Layout algorithms

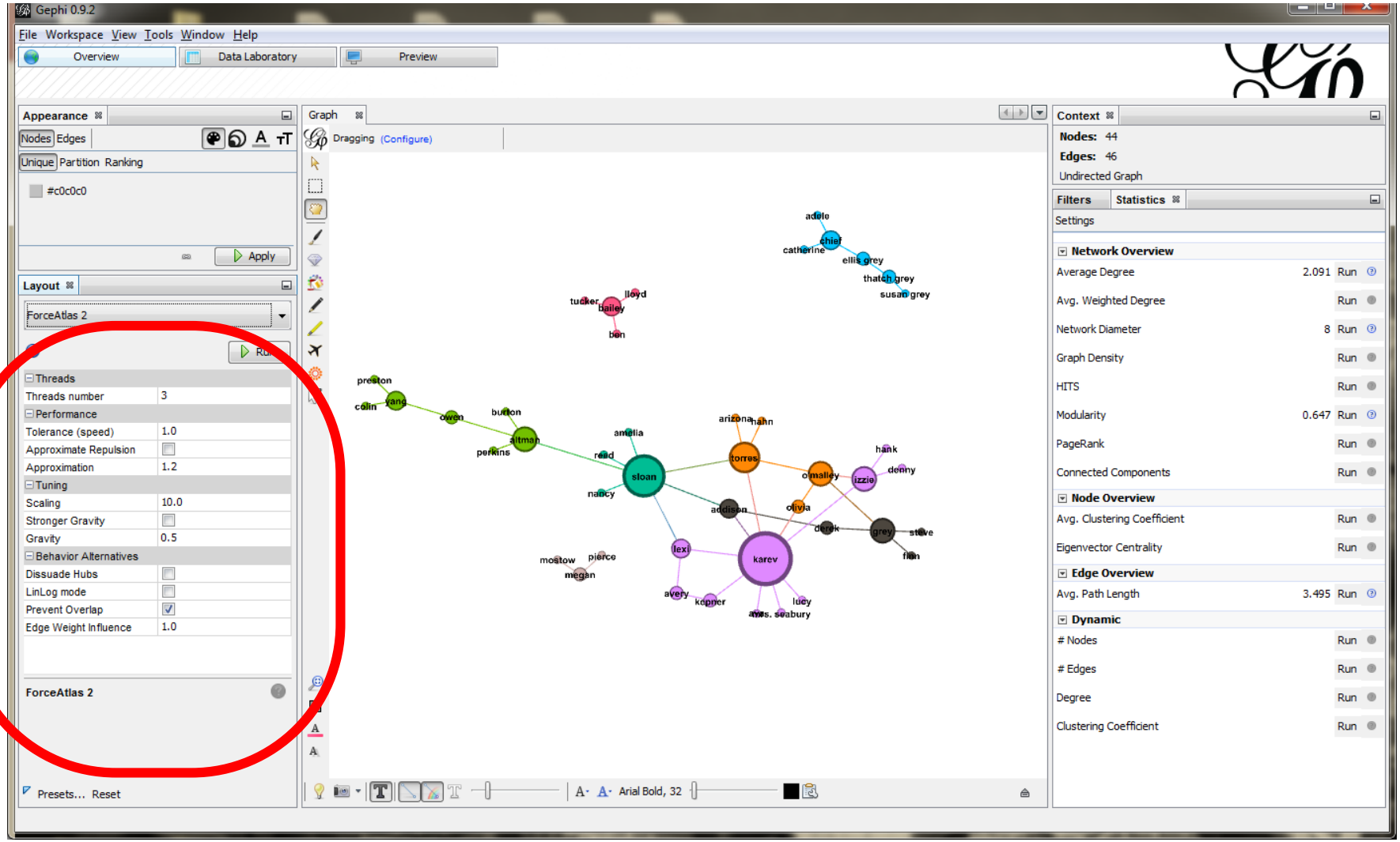
- Vizualizing networks is a complicated problem -- how do you position the nodes and edges in a way such that no nodes overlap, connected nodes are near each other, none of the labels overlap? Typically we use what is called a *layout* to plot or visualize networks. A layout is an algorithm used to position nodes and edges on a plot automatically in aesthetically and informationally satisfactory ways.

- There are several different layout algorithms, but the most common is a *force-directed* layout. These layout algorithms are based off of physical repulsion and spring systems. In general, the rule for force-directed layouts is: repel all nodes, and model connections between nodes as 'springs', with the result that more connected nodes will be closer together.

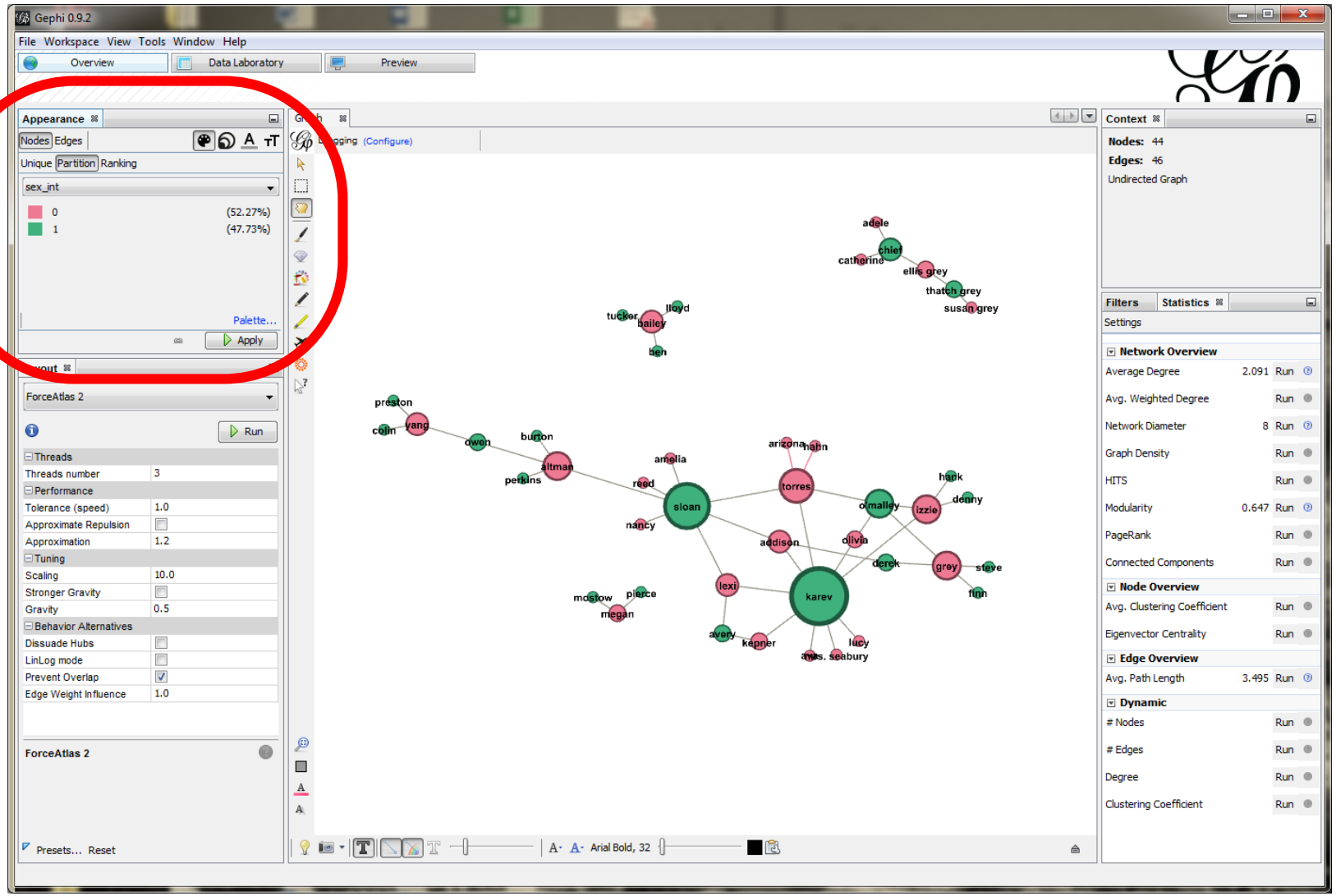
- One important issue is that each layout typically has random initial conditions. Running a plot function twice will return two different plots, both following the rules of the algorithm, but differing due to the initial conditions of the layout.



Layout algorithms



Detailed plotting with colors by attribute

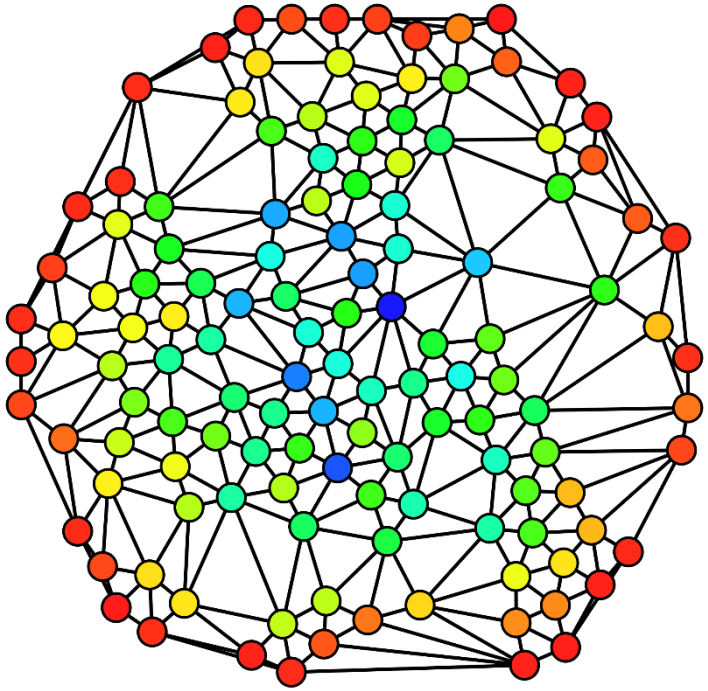


Agenda

1. Introduction and examples
2. Fundamental concepts
    1. Creating graphs
    2. Visualizing graphs
    3. **Centrality**
    4. Describing the macro structure
    5. Community detection
3. Getting and handling data
4. Extended concepts
5. Discussion and conclusion

Centrality measures – overview

- Definition of centrality
- Compare and contrast popular centrality measures on dataset
  - Degree
  - Closeness
  - Betweenness
  - Eigenvector



Degree centrality

The **degree** of a node is the number of other nodes to which it is connected.

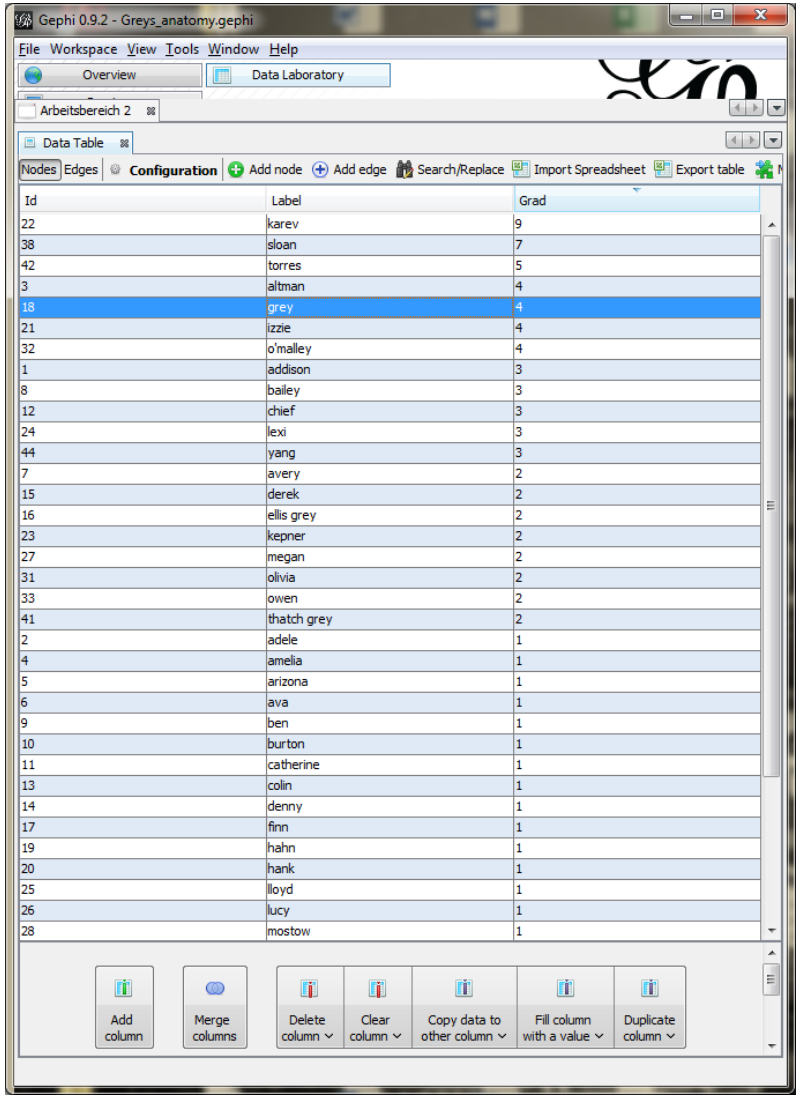
Gephi‘s degree centrality is calculated as the absolute value. Other programs are taking the degree of the node and dividing by `n-1` where `n` is the number of nodes in `G`.

$$C_D(u) = \frac{deg(u)}{n-1}$$

NOTE: In a directed graph, both in-degree and out-degree centrality can be calculated.

Let's find the degree of our main character Grey (4).

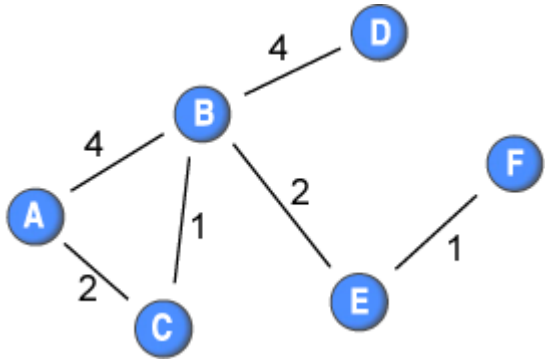
Likewise we find the degree of each cast member…



Id	Label	Grad
22	karev	9
38	sloan	7
42	torres	5
3	altman	4
18	grey	4
21	izzie	4
32	o'malley	4
1	addison	3
8	bailey	3
12	chief	3
24	lexi	3
44	yang	3
7	avery	2
15	derek	2
16	ellis grey	2
23	kepner	2
27	megan	2
31	olivia	2
33	owen	2
41	thatch grey	2
2	adele	1
4	amelia	1
5	arizona	1
6	ava	1
9	ben	1
10	burton	1
11	catherine	1
13	colin	1
14	denny	1
17	finn	1
19	hahn	1
20	hank	1
25	lloyd	1
26	lucy	1
28	mostow	1

Closeness centrality

Closeness Centrality measures how many "hops" it would take to reach every other node in a network (taking the shortest path). It can be informally thought as 'average distance' to all other nodes.



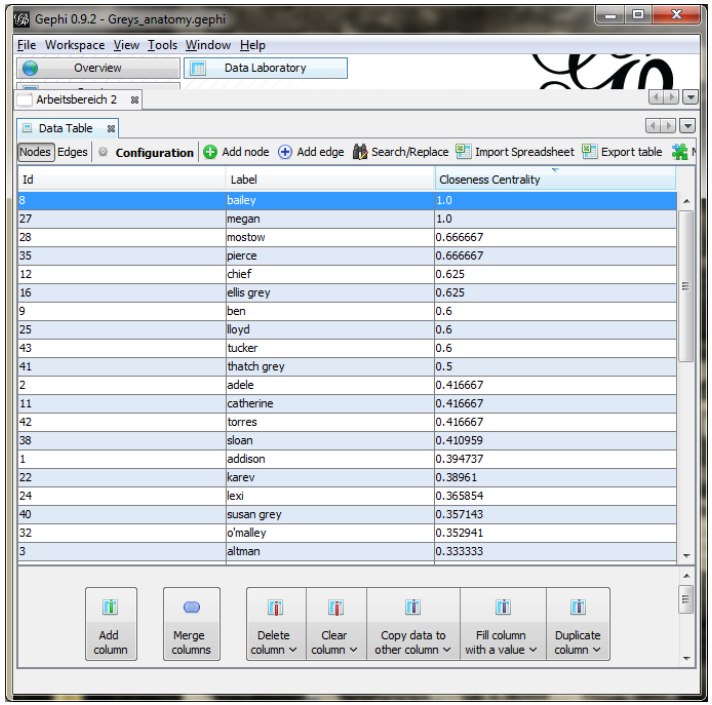
In Gephi, it the reciporical of of the *average* value, which normalizes the value in a 0 to 1 range.

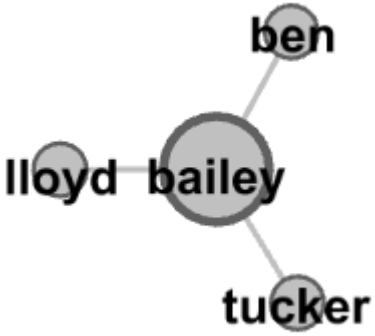
$$C_C(u) = \frac{n-1}{\sum_{v=1}^{n-1} d(v,u)}$$

Closeness centrality

$$C_C(u) = \frac{n-1}{\sum_{v=1}^{n-1} d(v,u)}$$

Why has Bailey a value of 1?





4 – 1

---------- = 1

1+1+1

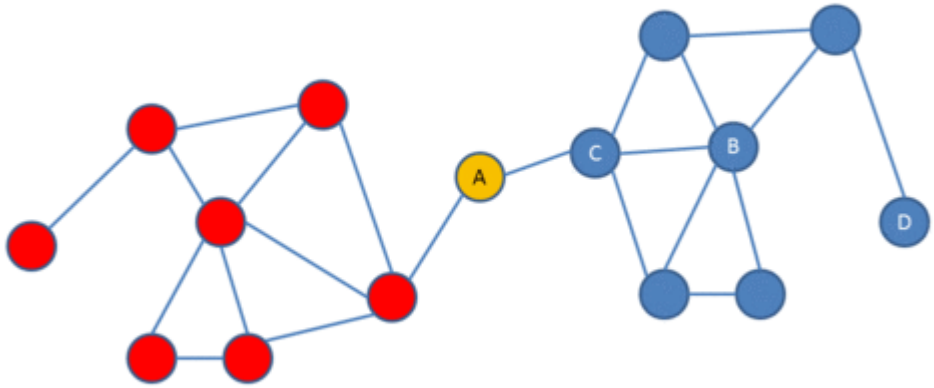
NOTE: If the graph is not completely connected, this algorithm computes the closeness centrality for each connected part separately.

Why care about closeness centrality?

- Degree centrality measures might be criticized because they only take into account the immediate ties that an actor has, or the ties of the actor's neighbors, rather than indirect ties to all others.
- One actor might be tied to a large number of others, but those others might be rather disconnected from the network as a whole.
- In a case like this, the actor could be quite central, but only in a local neighborhood.

Betweenness centrality

Betweenness centrality quantifies the number of times a node acts as a bridge (or "broker") along the shortest path between two other nodes.



In this conception, nodes that have a high frequency to occur on a shortest path between two nodes have a high betweenness.

$$C_B(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)}$$

where $\sigma(s,t)$ is total number of shortest paths from node s to node t and $\sigma(s,t|v)$ is the number of those paths that pass through v.

Why care about betweenness centrality?

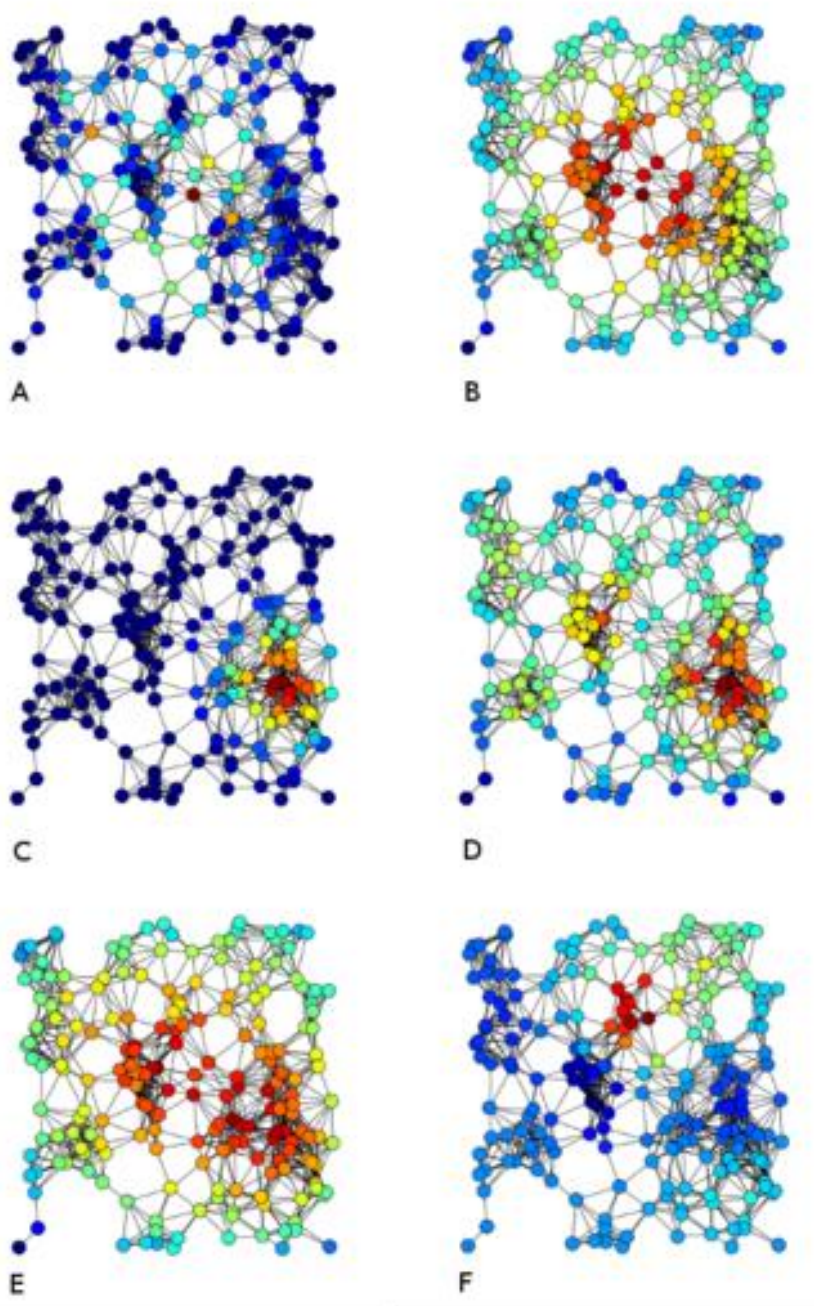
- Intermediaries occupy powerful positions
- Boundary spanning and nexus work – connecting different subgraphs

- Interpretation: Suppose you need permission from ex-partners to date. How many persons would Preston need to date Torres?

Eigenvector centrality

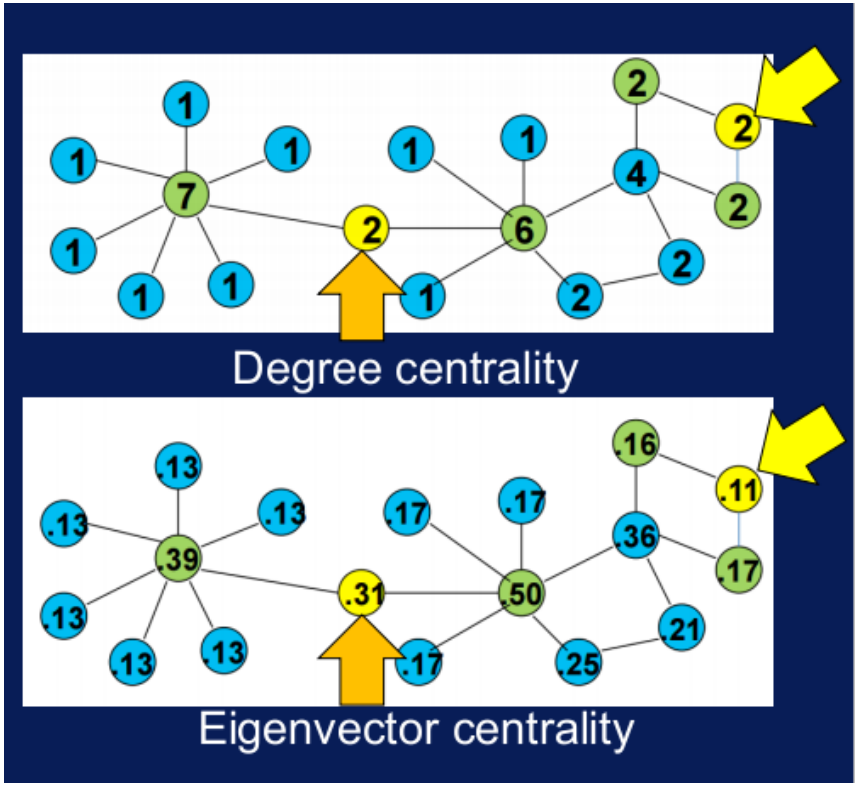
A node is high in eigenvector centrality if it is connected to many other nodes who are themselves well connected. Eigenvector centrality for each node is calculated as the proportional eigenvector values of the eigenvector with the largest eigenvalue.

Middle Left ("C"): *Eigenvector Centrality.*
Middle Right ("D"): *Degree Centrality*

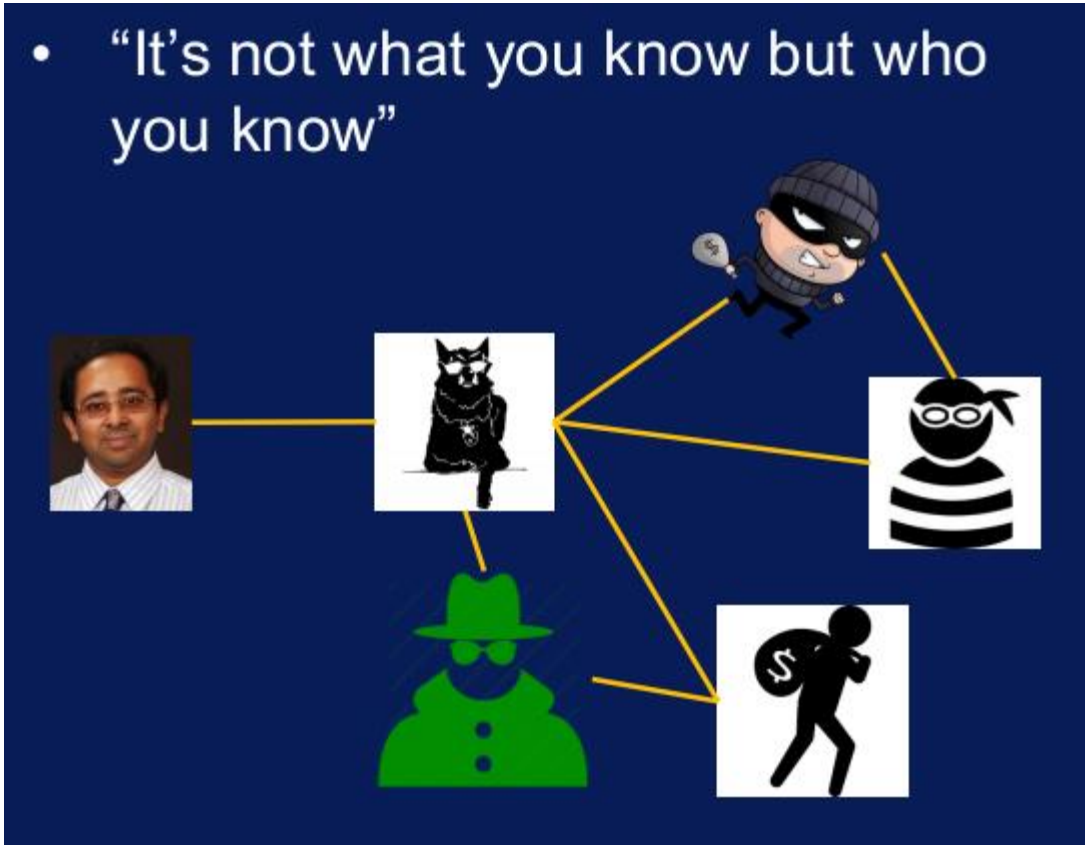


Eigenvector centrality

- Centrality is proportional to the **sum of centrality of the neighbors**
- C_i is proportional to $\sum_{j:friend\ of\ i} Cj$
- $C_i = a \sum_j g_{ij} Cj$



Eigenvector centrality: Interpretation



Pagerank is a variant of Eigenvector centrality (Intuition): A random surfer of a graph is most likely to find nodes that are highly "central" because a lot of nodes point to these nodes directly or indirectly

Pagerank

Pagerank is a variant of Eigenvector centrality

A random surfer of a graph is most likely to find nodes that are highly “central” because a lot of nodes point to these nodes directly or indirectly

Comparison of different centrality measures

Id	Label	Grad	Closeness Centrality	Betweenness Centrality	Eigenvector Centrality
22	karev	9	0.38961	0.194009	1.0
38	sloan	7	0.410959	0.257873	0.716206
42	torres	5	0.416667	0.146802	0.697546
3	altman	4	0.333333	0.169435	0.294612
18	grey	4	0.283019	0.067774	0.249274
21	izzie	4	0.319149	0.071154	0.475488
32	o'malley	4	0.352941	0.082337	0.497948
1	addison	3	0.394737	0.080362	0.522082
8	bailey	3	1.0	0.003322	0.040869
12	chief	3	0.625	0.007752	0.053202
24	lexi	3	0.365854	0.057443	0.522432
44	yang	3	0.217391	0.063123	0.072553
7	avery	2	0.275229	0.003599	0.22715
15	derek	2	0.309278	0.028073	0.215354
16	ellis grey	2	0.625	0.006645	0.046777
23	kepner	2	0.288462	0.008079	0.327235
27	megan	2	1.0	0.001107	0.02113
31	olivia	2	0.306122	0.008031	0.400993
33	owen	2	0.265487	0.089701	0.119936
41	thatch grey	2	0.5	0.00443	0.035611

Agenda

1. Introduction and examples
2. Fundamental concepts
    1. Creating graphs
    2. Visualizing graphs
    3. Centrality
    4. **Describing the macro structure**
    5. Community detection
3. Getting and handling data
4. Extended concepts
5. Discussion and conclusion

Describing complex relationships

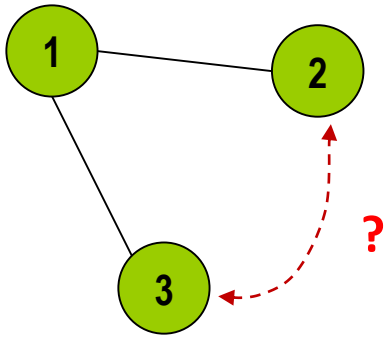
- **Global properties of the network (macro level)**
  - Degree distribution, average path length, density
- Segregation patterns (meso level)
  - Types of nodes, modularity, and homophily
- Local properties of single nodes (micro level)
  - Clustering coefficient, transitivity, …
  - Position of a node: Centrality, prestige, robustness,…

Clustering coefficient: from local to global

- How many of Grey‘s former lovers did have a romantic relationship with each other?
- **Local clustering coefficient:**

$$CI_i(g) = \frac{\#\{kj \; in \; g | k,j \; in \; N_i(g)\}}{\#\{kj | k,j \; in \; N_i(g)\}}$$

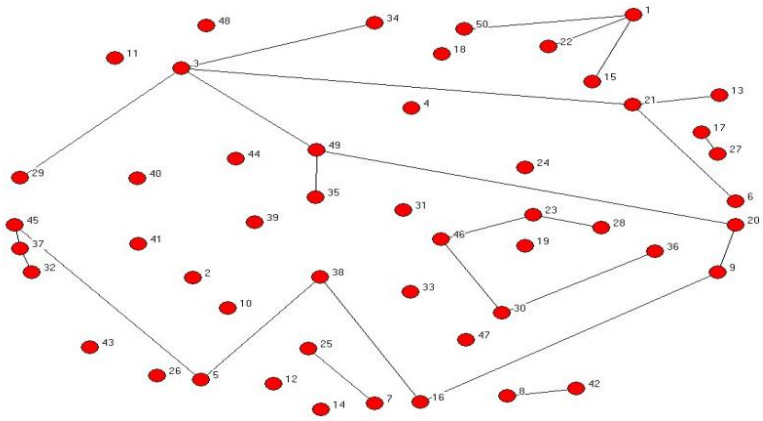
Frequency of this link?



- **Global clustering coefficient:**
  Scaled-up version averaged over all nodes in der network

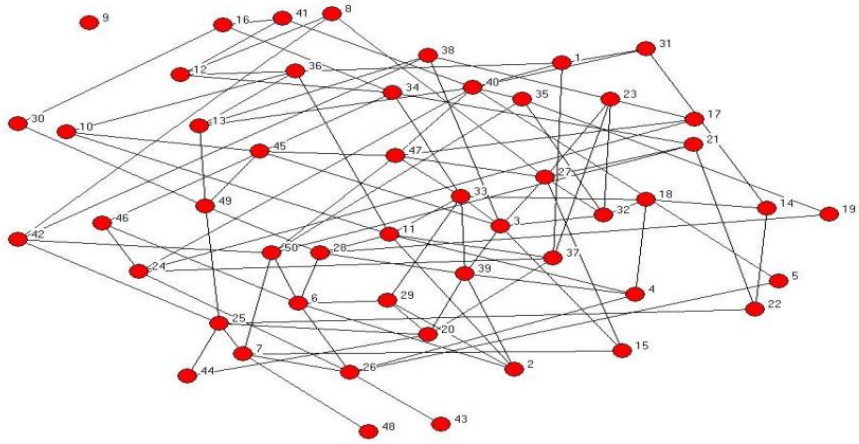
Density

A **dense network** is a **network** in which the number of links of each node is close to the maximal number of nodes. Each node is linked to almost all other nodes. The total connected case in which exactly each node is linked to each other node is called a completely connected **network**. Value ranges between 0 and 1.



Random network p=0.02, n=50

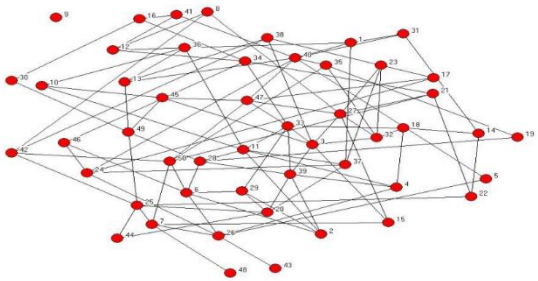
Density: 0.019

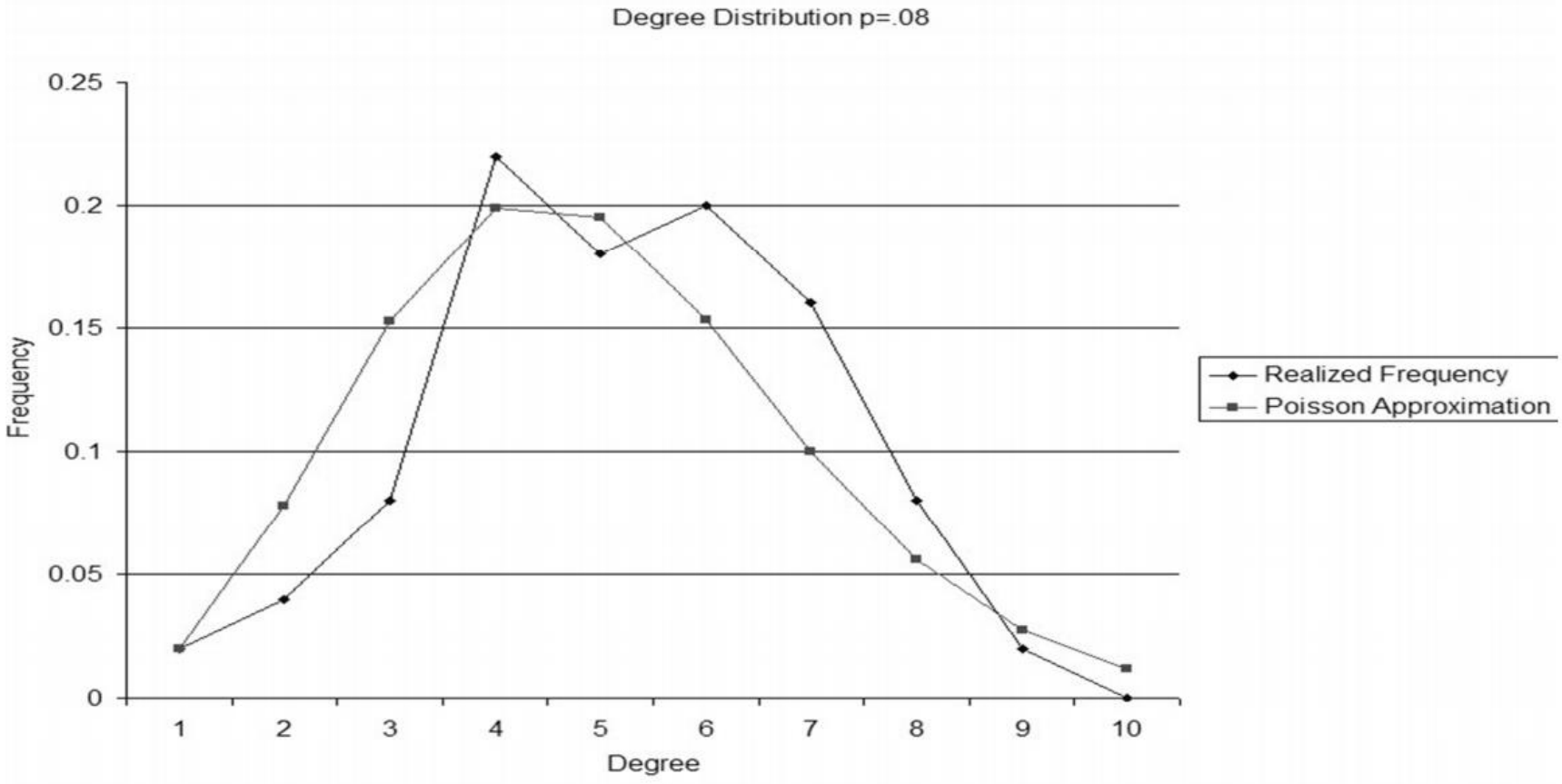


Random network p=0.05, n=50

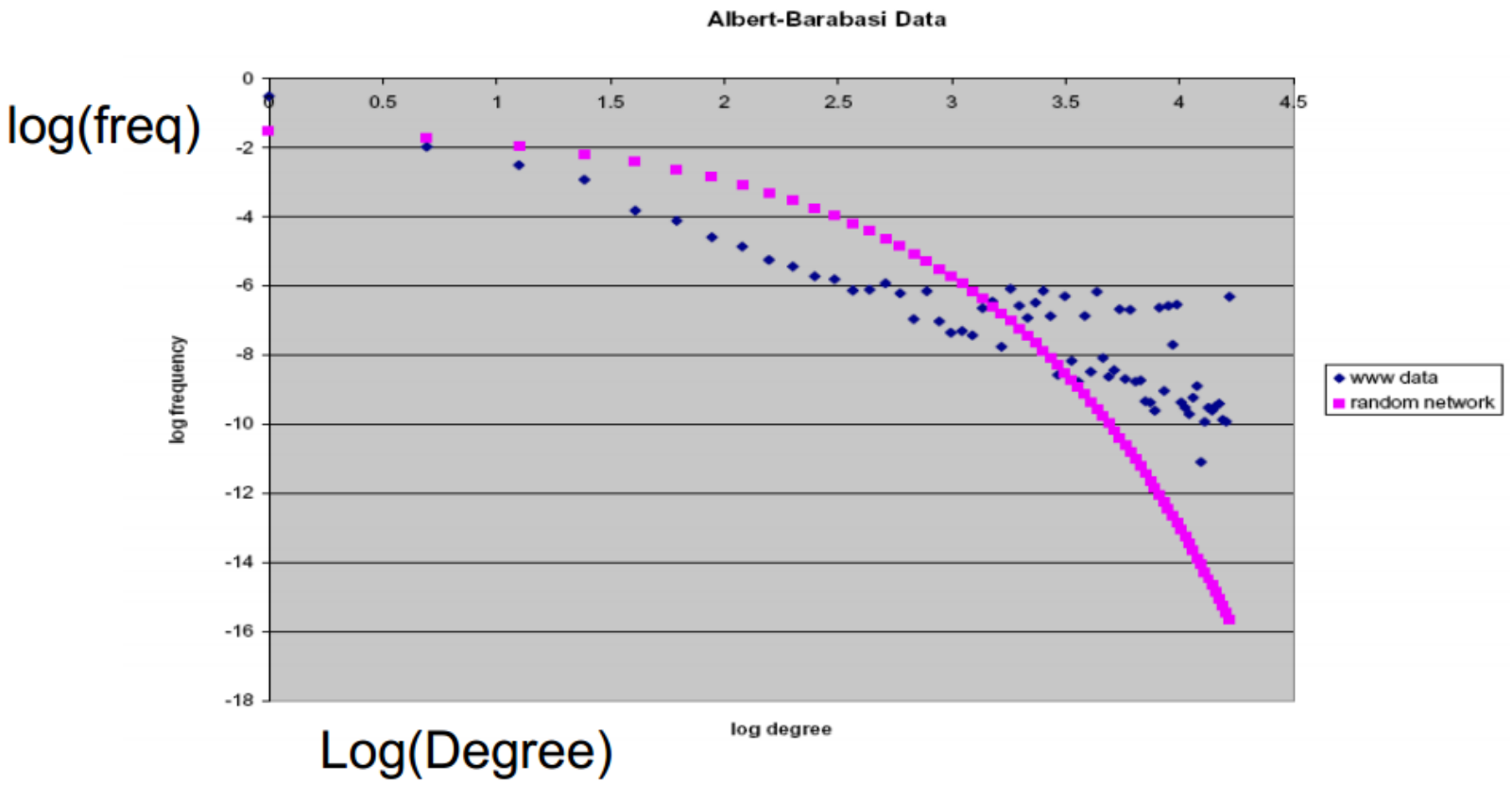
Density: 0.055

Degree distribution: Random network





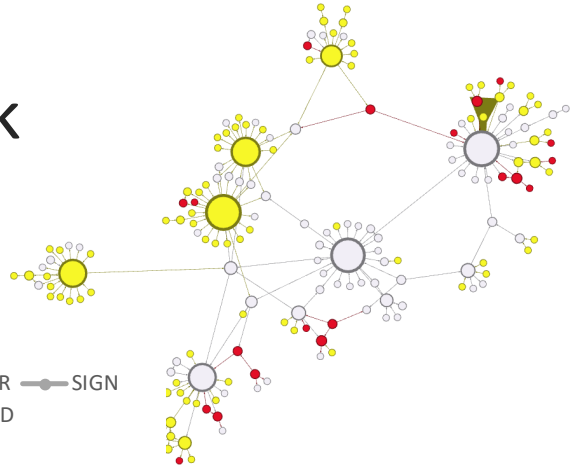
Degree distribution: Scale-free network

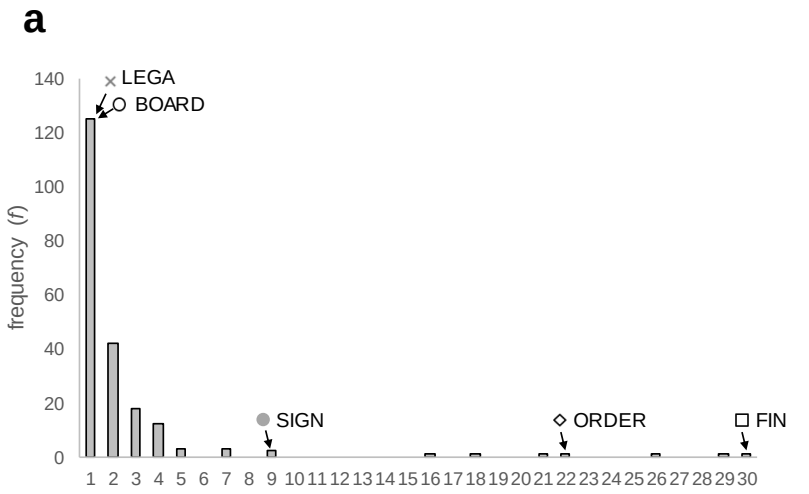


- "Fat Tails" (Price 1965), "Matthew Effect" (The rich get richer, the poor get poorer")

→ More degrees with high or low degrees, middle range ill-populated (no reg. to mean)

Degree distribution: Scale-free network



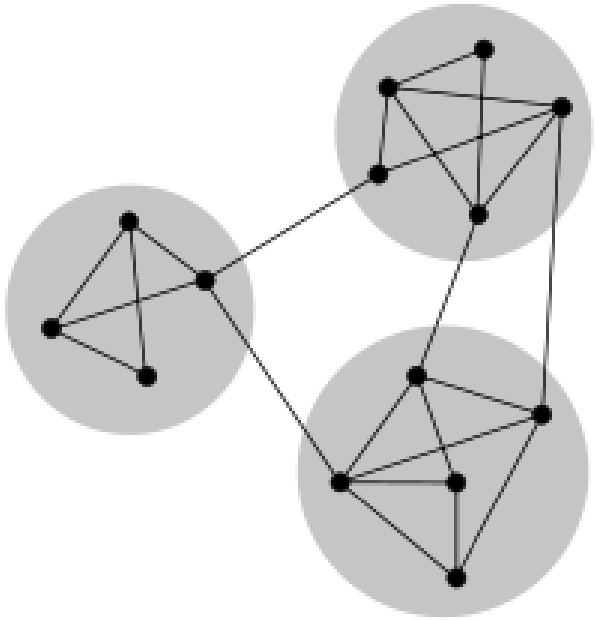


Agenda

1. Introduction and examples
2. Fundamental concepts
    1. Creating graphs
    2. Visualizing graphs
    3. Centrality
    4. Describing the macro structure
    5. **Community detection**
3. Getting and handling data
4. Extended concepts
5. Discussion and conclusion

Community detection: intuition

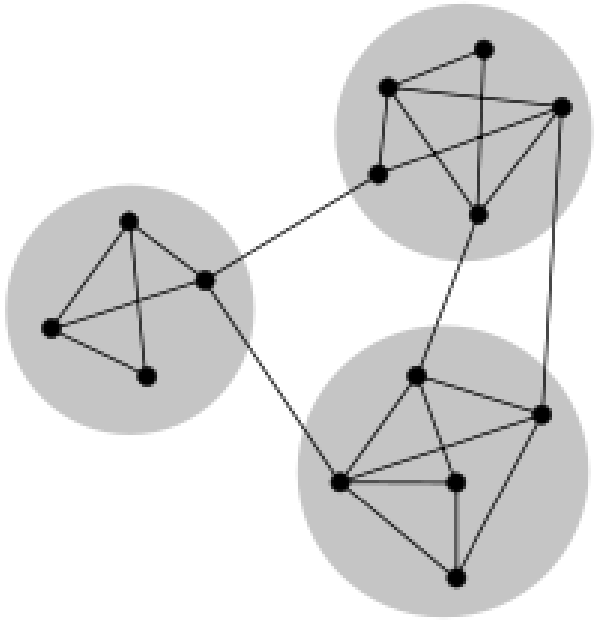
- “Where do I belong?“
  - Definitions
  - Modularity clustering
  - K-cores community detection



Community detection: intuition

Communities (also called partitions, clusters, groupings) in networks are sets of nodes that are more densely connected within a community than between other communities.

Community detection aims to reveal underlying community structure and can be handled in several different ways.



Modularity clustering: intuition

Modularity: A global measure of cluster quality

- fraction of the edges within the given groups minus the expected such fraction if edges were distributed at random.
- Value of the modularity lies in the range [−1/2,1).
- Steps of the method:

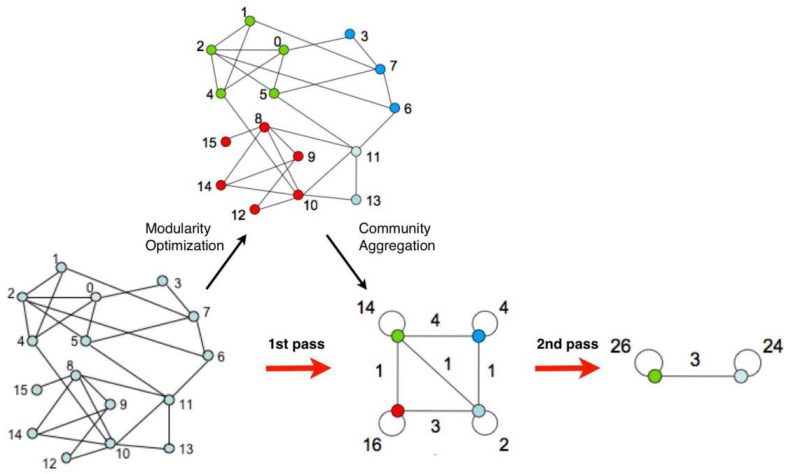
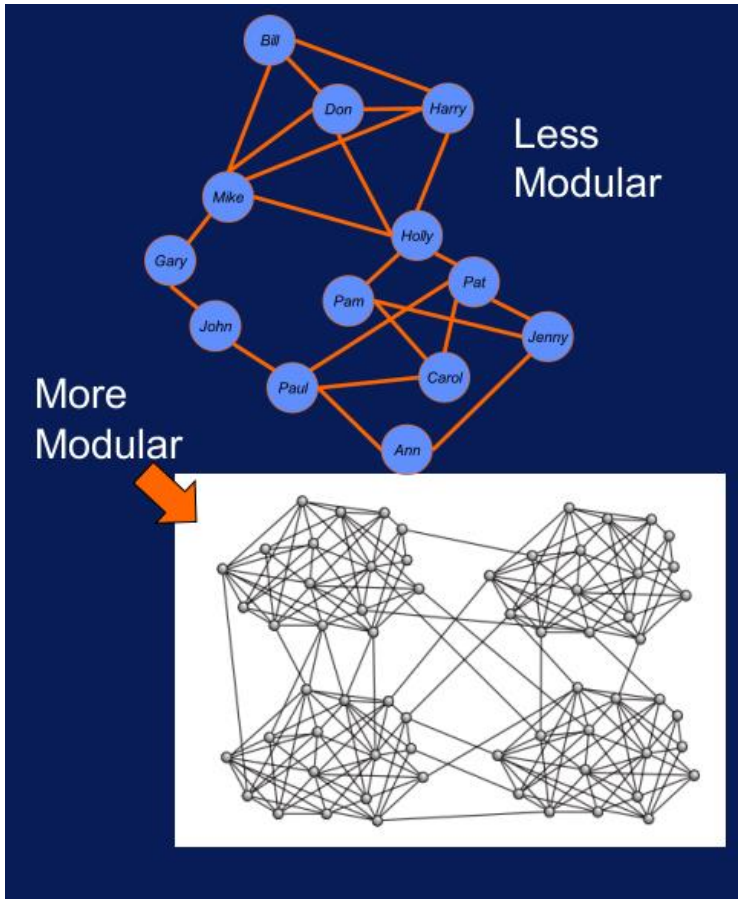
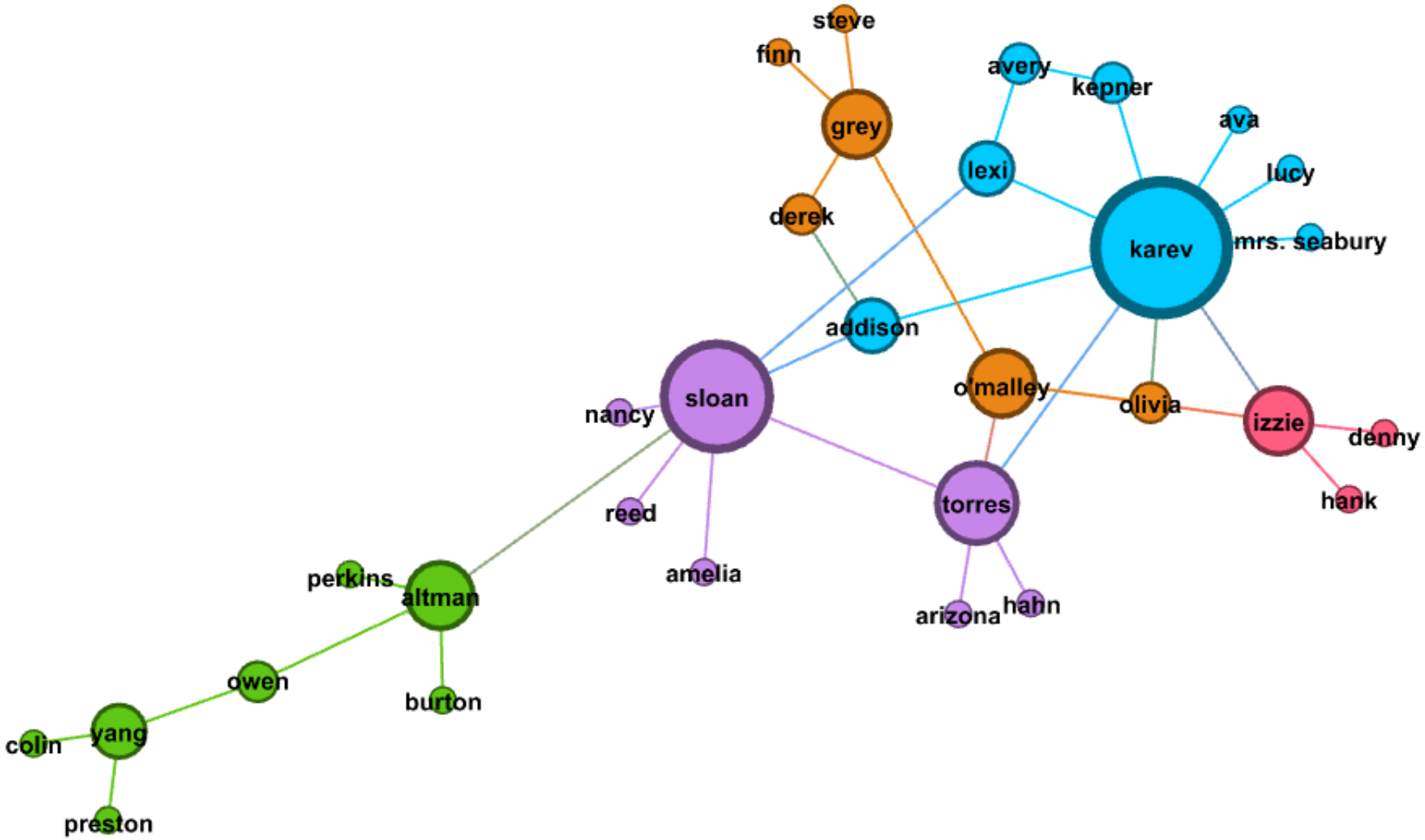


Figure 1. Visualization of the steps of our algorithm. Each pass is made of two phases: one where modularity is optimized by allowing only local changes of communities; one where the found communities are aggregated in order to build a new network of communities. The passes are repeated iteratively until no increase of modularity is possible.



[Blondel et al. 2008]

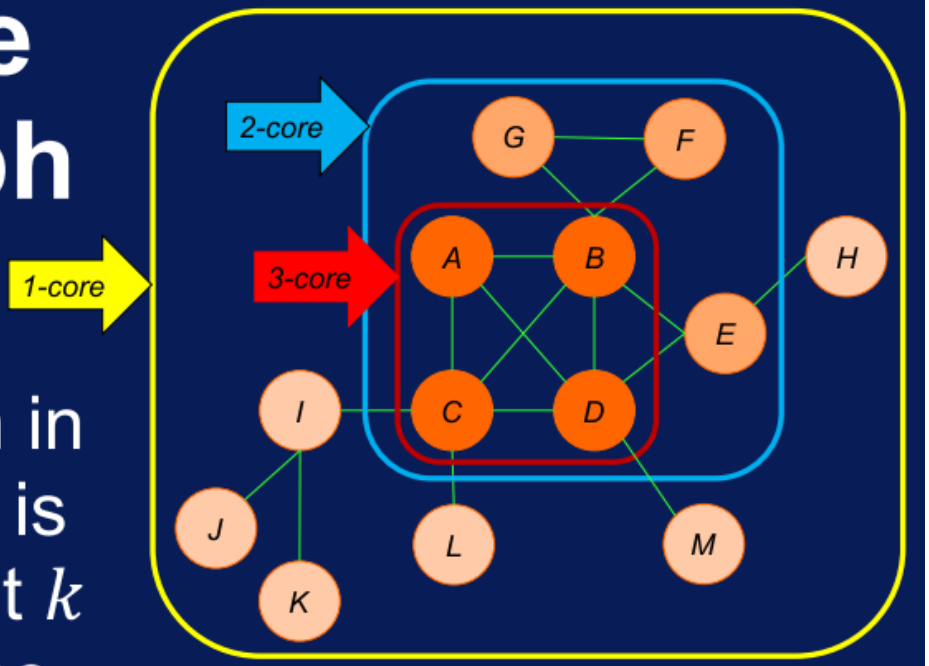
Modularity clustering: Different subgroups in the Grey‘s Anatomy cast



K-Core

Finding dense parts of a graph

- k-core
  - Maximal subgraph in which each vertex is adjacent to at least k other vertices of the subgraph



Agenda

1. Introduction and examples
2. Fundamental concepts
    1. Creating graphs
    2. Visualizing graphs
    3. Centrality
    4. Community detection
3. **Getting and handling data**
4. Extended concepts
5. Discussion and conclusion

The “classical“ workflow for an SNA study

- Formulate problem scope
- Build hypotheses
- Collect data
  - Survey data
  - Archival data
- Visualize data
- Analyze data

General ways to extract data from social media sites and other web sources

Ready-made software (e.g. Netvizz)

- Uses existing APIs such as FBs Graph API, Twitters Streaming API or GitHub's REST API
- Pros: Easy to use, intuitive
- Cons: Often limited in scope or functionality, changes in API may make the program obsolete

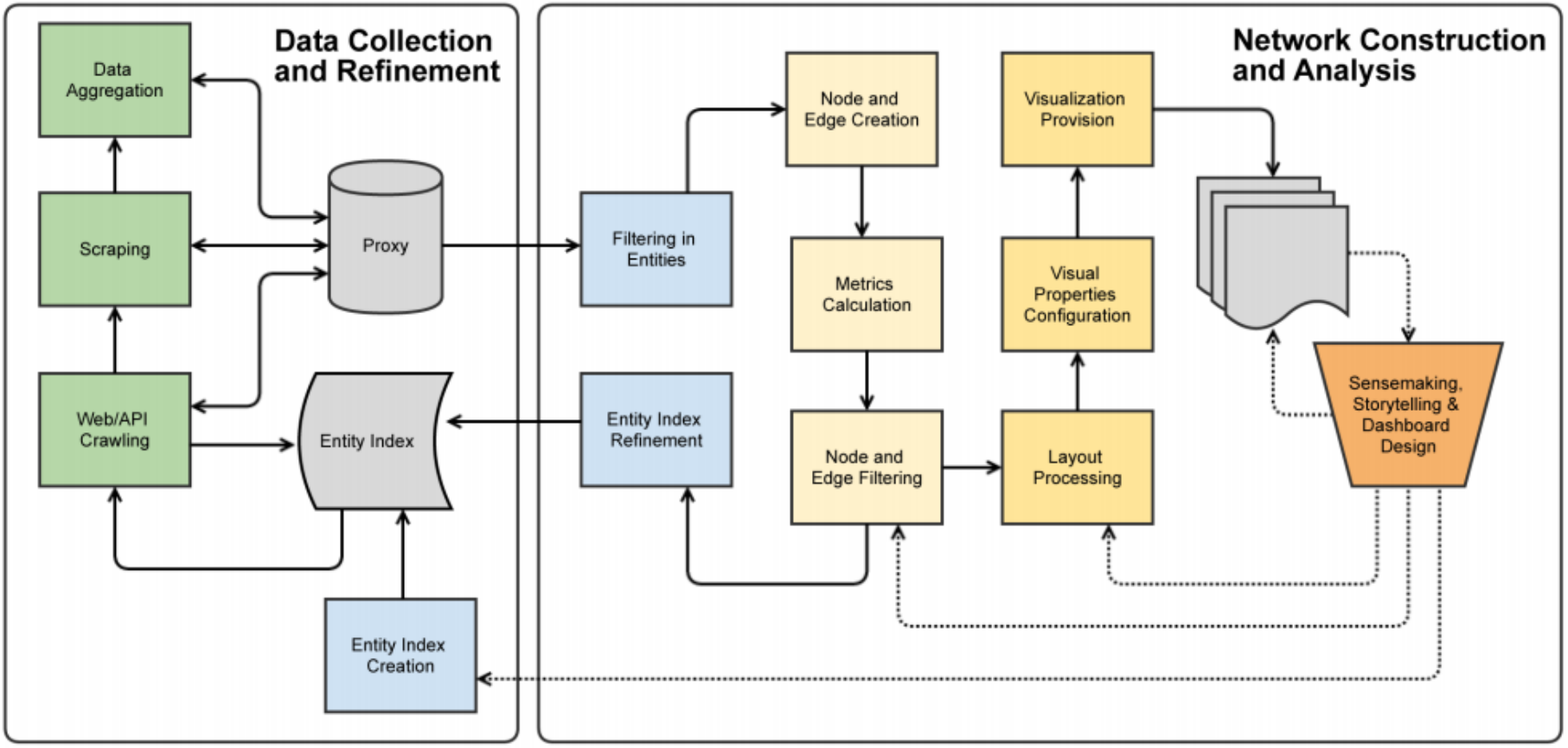
Custom-developed code (e.g. Phyton)

- Similarly builds on existing APIs but uses self-developed program code (e.g. Phyton and Project Jupyter)
- Pros: More flexible than ready-made software
- Cons: Learning curve, restricted by the APIs

Web scraping (e.g. Web scraper Chrome extension)

- Uses scrapers to extract data from website code
- Pros: Most flexible approach, in principle you can get all data that you can see or query on the web
- Cons: Often very messy websites "polluted" with Java Script, results may be mixed for various reasons

The data science paradigm to an SNA study…



[Huhtamäki 2016]

Twitter Streaming API: Understand the data structure

The JSON will be a mix of 'root-level' attributes (here we are highlighting some of the most fundamental attributes), and child objects (which are represented here with the `{}` notation):


```
{
  "created_at":"Thu Apr 06 15:24:15 +0000 2017",
  "id": 850006245121695744,
  "id_str": "850006245121695744",
  "text": "1/ Today we're sharing our vision for the future of the Twitter API platform!nhttps://t.co/XweGngmxlP",
  "user": {},
  "entities": {}
}
```

Twitter Streaming API: Possible networks to create

- **re-tweeting (retweet network)**
- replying (reply network) to existing tweets
- mentioning (mention network) other users
- friends/followers social relationships among user involved in the above activities

[http://snap.stanford.edu/data/higgs-twitter.html]

Twitter Streaming API: Getting set-up

1. Download Anaconda (includes Phyton)
2. Download Project Jupyter Notebook
3. Install the necessary packages (tweepy, simplejson)
4. Create a Twitter account
5. Create a Twitter application
6. Create the listener (next slide)

[https://github.com/jukkahuhtamaki/demo-twitter-collector]

Twitter API: Building a listener with jupyter and tweepy

jupyter Twitter Streaming API listener Last Checkpoint: 2 minutes ago (autosaved)

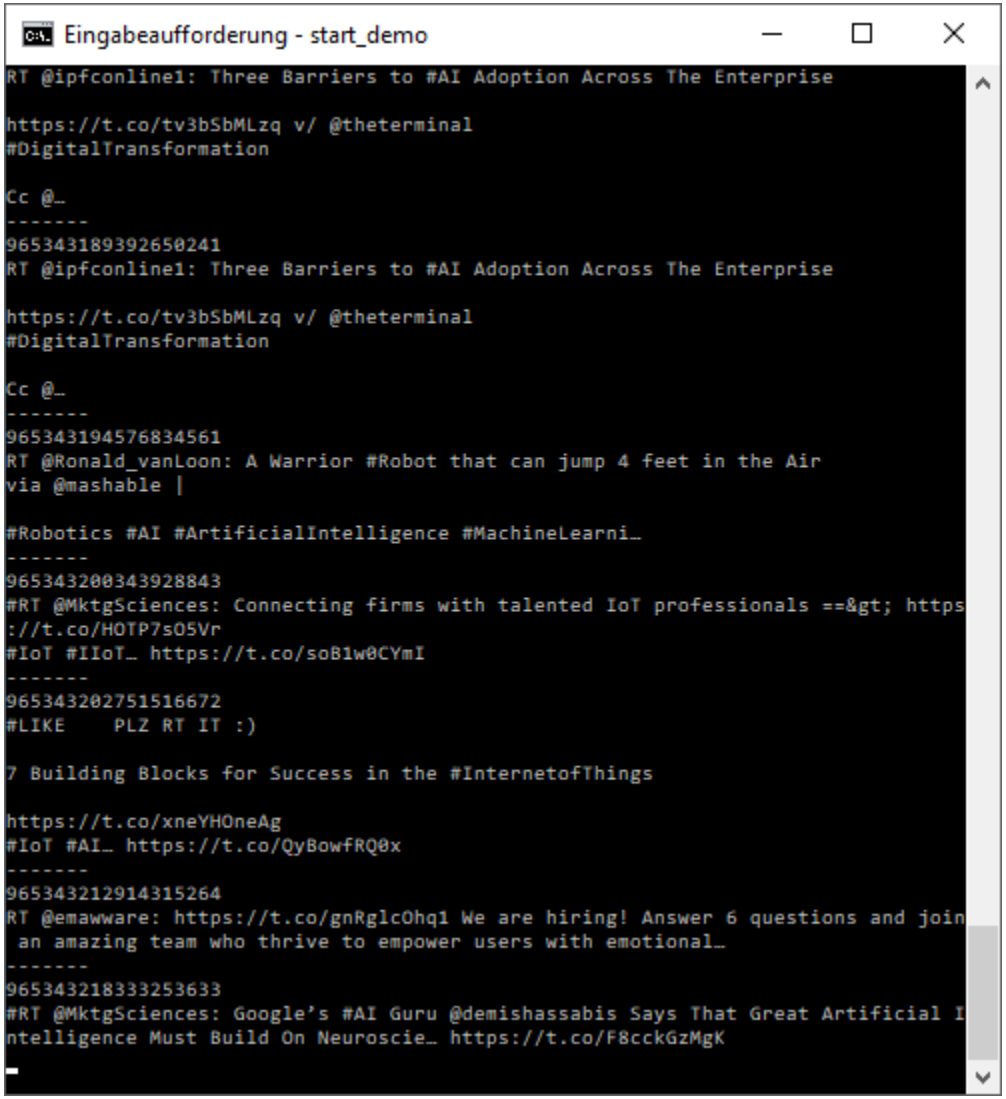
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Code

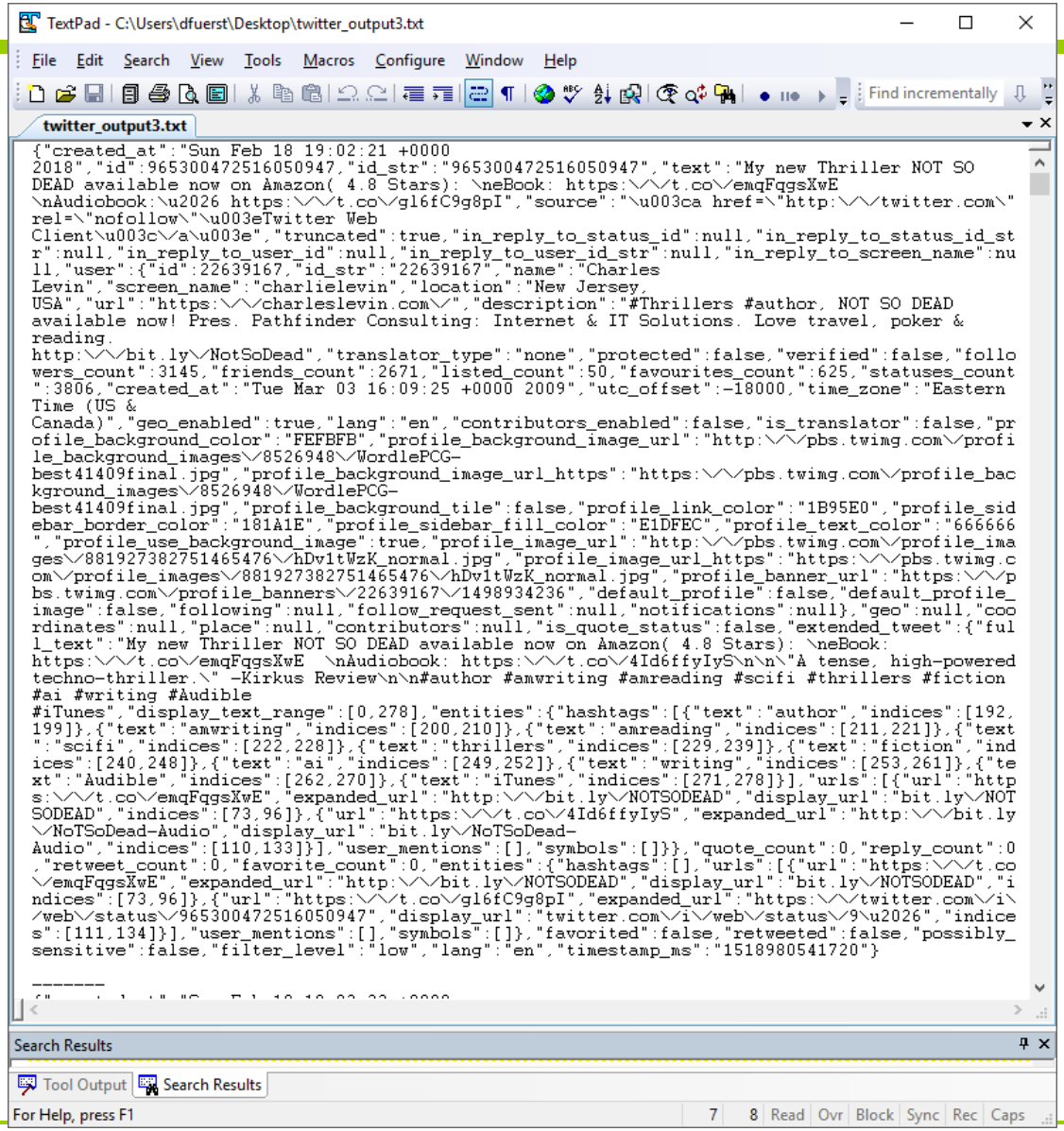
In [ ]:

```
from __future__ import absolute_import, print_function

from tweepy.streaming import StreamListener
from tweepy import OAuthHandler
from tweepy import Stream
import simplejson as json
import sys

# Go to http://apps.twitter.com and create an app.
# The consumer key and secret will be generated for you automatically
consumer_key='qvz9Op5IuQVCTClRL8XElbWwE'
consumer_secret='cwculhUloIPvllG1gE47VFpxVcHQb9ly9AwAPiFQYKS9612Ced'

# After the step above, you will be redirected to your app's page.
# Create an access token under the the "Your access token" section
access_token='10569772-r6OTgi7D5cvpERzlyWhVb50liECeIrBrwWMqZsle5'
access_token_secret='u2V85W7YRpQDM8tbYqhvVjYN1rdJlF1Nswl01RyjfPGFd'

class StdOutListener(StreamListener):
    """ A listener handles tweets that are received from the stream.
    This is a basic listener that just prints received tweets to stdout.
    """
    def on_data(self, data):
        tweet = json.loads(data)
        original = sys.stdout
        sys.stdout = open('C:/temp/twitter_output3.txt', 'a')
        print('-------')
        print(data)
        sys.stdout = original
        print('-------')
        print(tweet['id'])
        print(tweet['text'])
        return True

    def on_error(self, status):
        print(status)

if __name__ == '__main__':
    l = StdOutListener()
    auth = OAuthHandler(consumer_key, consumer_secret)
    auth.set_access_token(access_token, access_token_secret)

    stream = Stream(auth, l)
    stream.filter(track=['#ai'])
```

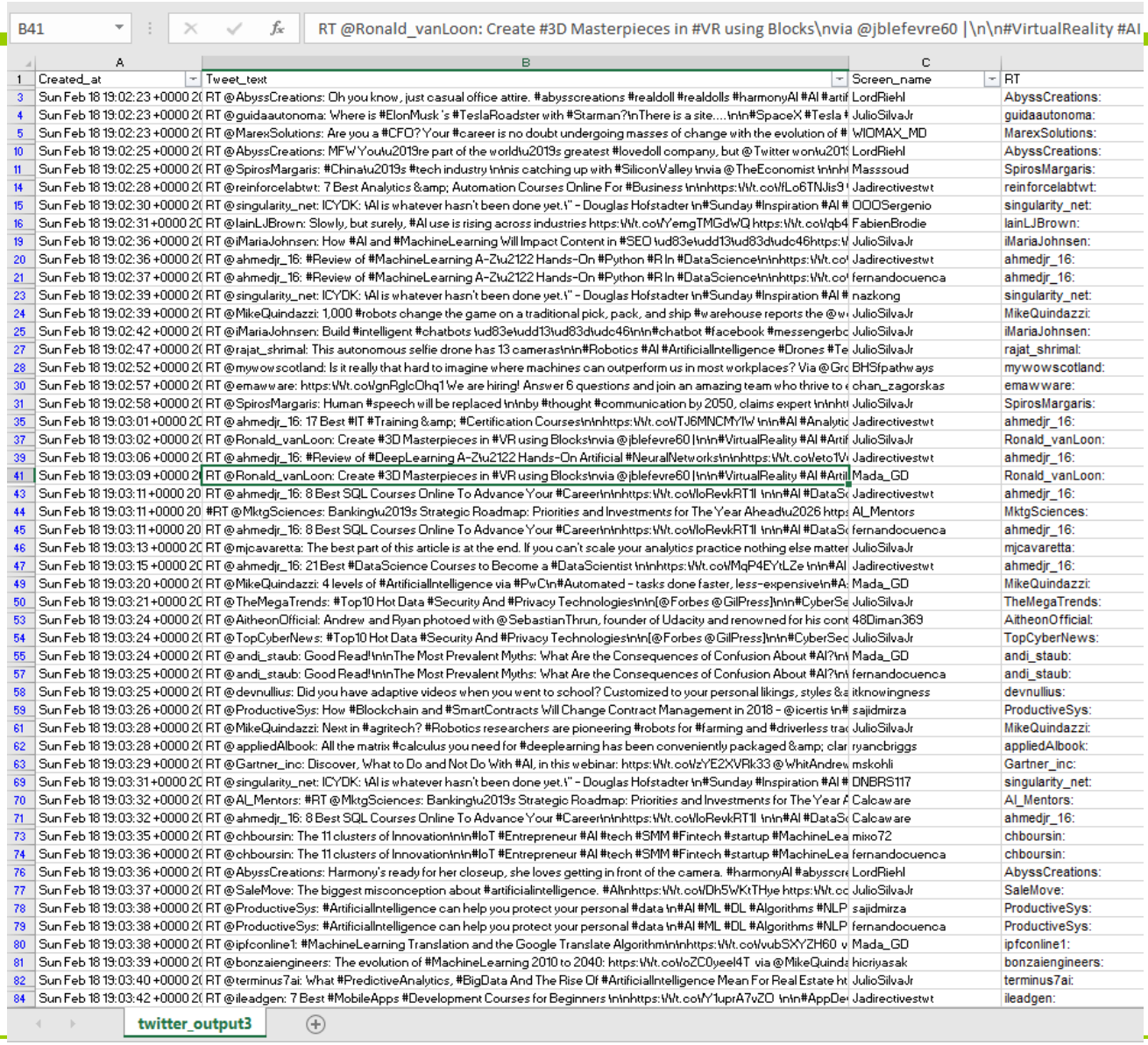
Twitter API: Writing output to stdout



Twitter API: Writing output to file



Twitter API: Parsing the file to create data set I



B41 | RT @Ronald_vanLoon: Create #3D Masterpieces in #VR using Blocks\nvia @jblefevre60 |\n\n#VirtualReality #AI

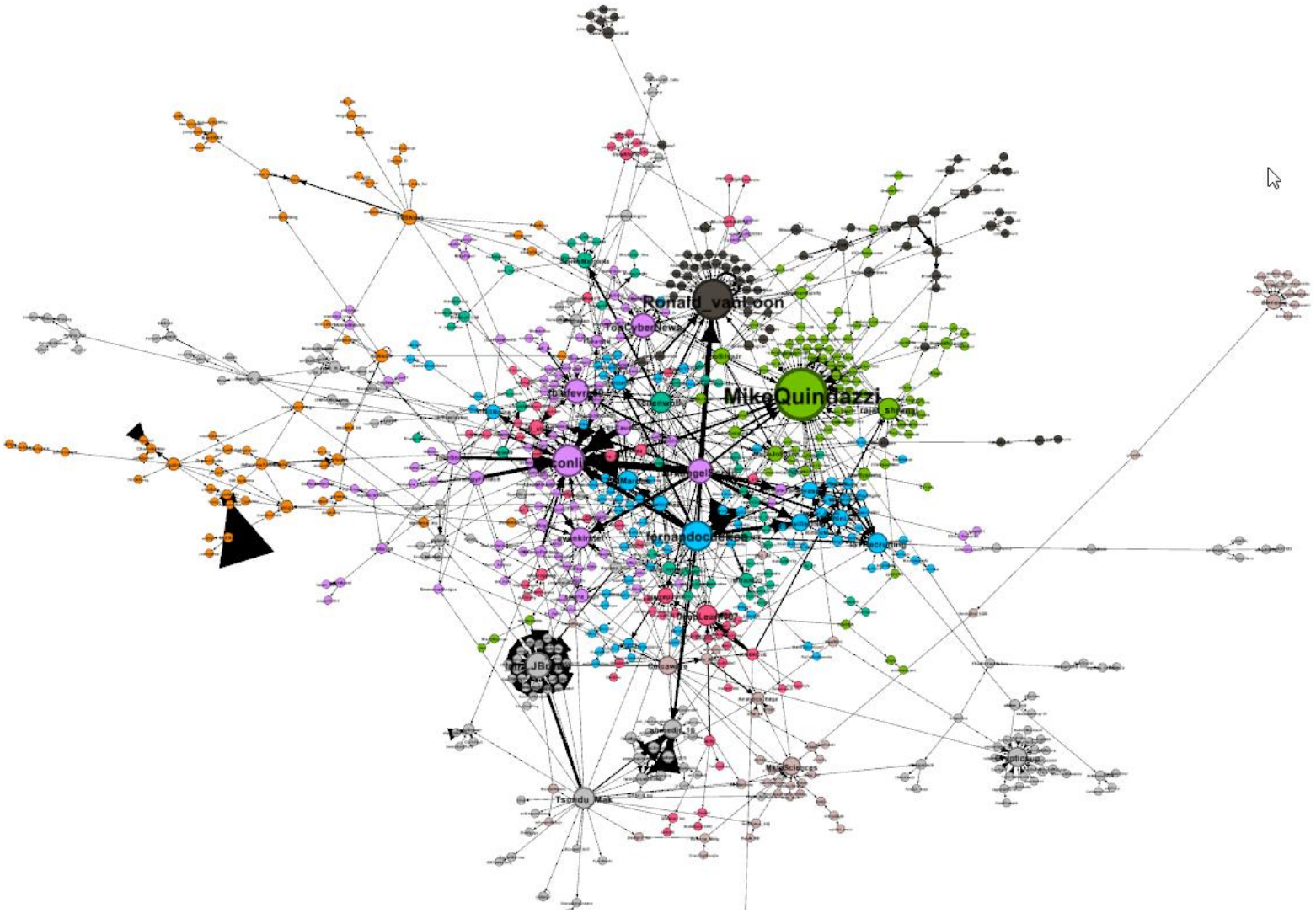
	A	B	C	
1	Created_at	Tweet_text	Screen_name	RT
3	Sun Feb 18 19:02:23 +0000 2(	RT @AbyssCreations: Oh you know, just casual office attire. #abysscreations #realdoll #realdolls #harmonyAI #AI #artif	LordRiehl	AbyssCreations:
4	Sun Feb 18 19:02:23 +0000 2(	RT @guidaautonoma: Where is #ElonMusk 's #TeslaRoadster with #Starman?\nThere is a site....\n\n#SpaceX #Tesla #	JulioSilvaJr	guidaautonoma:
5	Sun Feb 18 19:02:23 +0000 2(	RT @MarexSolutions: Are you a #CFO? Your #career is no doubt undergoing masses of change with the evolution of #	WIOMAX_MD	MarexSolutions:
10	Sun Feb 18 19:02:25 +0000 2(	RT @AbyssCreations: MFW You\u2019re part of the world\u2019s greatest #lovedoll company, but @Twitter won\u201	LordRiehl	AbyssCreations:
11	Sun Feb 18 19:02:25 +0000 2(	RT @SpirosMargaris: #China\u2019s #tech industry \n\nis catching up with #SiliconValley \nvia @TheEconomist \n\nhl	Masssoud	SpirosMargaris:
14	Sun Feb 18 19:02:28 +0000 2(	RT @reinforcelabtwt: 7 Best Analytics & Automation Courses Online For #Business \n\nhttps://t.co\/fLo6TNJis9	Jadirectivestwt	reinforcelabtwt:
15	Sun Feb 18 19:02:30 +0000 2(	RT @singularity_net: ICYDK: \tAI is whatever hasn't been done yet.\" – Douglas Hofstadter \n#Sunday #Inspiration #AI #	OOOSergenio	singularity_net:
16	Sun Feb 18 19:02:31 +0000 2(	RT @lainLJBrown: Slowly, but surely, #AI use is rising across industries https:\/\/t.co\/YemgTMGdWQ https:\/\/t.co\/qb4	FabienBrodie	lainLJBrown:
19	Sun Feb 18 19:02:36 +0000 2(	RT @iMariaJohnsen: How #AI and #MachineLearning Will Impact Content in #SEO \ud83e\udd13\ud83d\udc46https:\/	JulioSilvaJr	iMariaJohnsen:
20	Sun Feb 18 19:02:36 +0000 2(	RT @ahmedjr_16: #Review of #MachineLearning A-Z\u2122 Hands-On #Python #R In #DataScience\n\nhttps:\/\/t.co\	Jadirectivestwt	ahmedjr_16:
21	Sun Feb 18 19:02:37 +0000 2(	RT @ahmedjr_16: #Review of #MachineLearning A-Z\u2122 Hands-On #Python #R In #DataScience\n\nhttps:\/\/t.co\	fernandocuenca	ahmedjr_16:
23	Sun Feb 18 19:02:39 +0000 2(	RT @singularity_net: ICYDK: \tAI is whatever hasn't been done yet.\" – Douglas Hofstadter \n#Sunday #Inspiration #AI #	nazkong	singularity_net:
24	Sun Feb 18 19:02:39 +0000 2(	RT @MikeQuindazzi: 1,000 #robots change the game on a traditional pick, pack, and ship #warehouse reports the @w	JulioSilvaJr	MikeQuindazzi:
25	Sun Feb 18 19:02:42 +0000 2(	RT @iMariaJohnsen: Build #intelligent #chatbots \ud83e\udd13\ud83d\udc46\n\n#chatbot #facebook #messengerbc	JulioSilvaJr	iMariaJohnsen:
27	Sun Feb 18 19:02:47 +0000 2(	RT @rajat_shrimal: This autonomous selfie drone has 13 cameras\n\n#Robotics #AI #ArtificialIntelligence #Drones #Te	JulioSilvaJr	rajat_shrimal:
28	Sun Feb 18 19:02:52 +0000 2(	RT @mywowscotland: Is it really that hard to imagine where machines can outperform us in most workplaces? Via @Gr	BHSfpathways	mywowscotland:
30	Sun Feb 18 19:02:57 +0000 2(	RT @emawware: https:\/\/t.co\/gnRglcOhq1We are hiring! Answer 6 questions and join an amazing team who thrive to	chan_zagorskas	emawware:
31	Sun Feb 18 19:02:58 +0000 2(	RT @SpirosMargaris: Human #speech will be replaced \n\nby #thought #communication by 2050, claims expert \n\nht	JulioSilvaJr	SpirosMargaris:
35	Sun Feb 18 19:03:01 +0000 2(	RT @ahmedjr_16: 17 Best #IT #Training & #Certification Courses\n\nhttps:\/\/t.co\/TJ6MNCMYIW \n\n#AI #Analytic	Jadirectivestwt	ahmedjr_16:
37	Sun Feb 18 19:03:02 +0000 2(	RT @Ronald_vanLoon: Create #3D Masterpieces in #VR using Blocks\nvia @jblefevre60 |\n\n#VirtualReality #AI #Artif	JulioSilvaJr	Ronald_vanLoon:
39	Sun Feb 18 19:03:06 +0000 2(	RT @ahmedjr_16: #Review of #DeepLearning A-Z\u2122 Hands-On Artificial #NeuralNetworks\n\nhttps:\/\/t.co\/etoTV	Jadirectivestwt	ahmedjr_16:
41	Sun Feb 18 19:03:09 +0000 2(	RT @Ronald_vanLoon: Create #3D Masterpieces in #VR using Blocks\nvia @jblefevre60 |\n\n#VirtualReality #AI #Artif	Mada_GD	Ronald_vanLoon:
43	Sun Feb 18 19:03:11 +0000 20	RT @ahmedjr_16: 8 Best SQL Courses Online To Advance Your #Career\n\nhttps:\/\/t.co\/IoRevkRT1l \n\n#AI #DataSc	Jadirectivestwt	ahmedjr_16:
44	Sun Feb 18 19:03:11 +0000 20	#RT @MktgSciences: Banking\u2019s Strategic Roadmap: Priorities and Investments for The Year Ahead\u2026 https	AI_Mentors	MktgSciences:
45	Sun Feb 18 19:03:11 +0000 20	RT @ahmedjr_16: 8 Best SQL Courses Online To Advance Your #Career\n\nhttps:\/\/t.co\/IoRevkRT1l \n\n#AI #DataSc	fernandocuenca	ahmedjr_16:
46	Sun Feb 18 19:03:13 +0000 2(	RT @mjcavaretta: The best part of this article is at the end. If you can't scale your analytics practice nothing else matter	JulioSilvaJr	mjcavaretta:
47	Sun Feb 18 19:03:15 +0000 2(	RT @ahmedjr_16: 21 Best #DataScience Courses to Become a #DataScientist \n\nhttps:\/\/t.co\/MqP4EYtLZe \n\n#AI	Jadirectivestwt	ahmedjr_16:
49	Sun Feb 18 19:03:20 +0000 2(	RT @MikeQuindazzi: 4 levels of #ArtificialIntelligence via #PwC\n\n#Automated – tasks done faster, less-expensive\n\n#A	Mada_GD	MikeQuindazzi:
50	Sun Feb 18 19:03:21 +0000 2(	RT @TheMegaTrends: #Top10 Hot Data #Security And #Privacy Technologies\n\n[@Forbes @GilPress]\n\n#CyberSe	JulioSilvaJr	TheMegaTrends:
53	Sun Feb 18 19:03:24 +0000 2(	RT @AitheonOfficial: Andrew and Ryan photoed with @SebastianThrun, founder of Udacity and renowned for his cont	48Diman369	AitheonOfficial:
54	Sun Feb 18 19:03:24 +0000 2(	RT @TopCyberNews: #Top10 Hot Data #Security And #Privacy Technologies\n\n[@Forbes @GilPress]\n\n#CyberSec	JulioSilvaJr	TopCyberNews:
55	Sun Feb 18 19:03:24 +0000 2(	RT @andi_staub: Good Read!\n\nThe Most Prevalent Myths: What Are the Consequences of Confusion About #AI?\n\	Mada_GD	andi_staub:
57	Sun Feb 18 19:03:25 +0000 2(	RT @andi_staub: Good Read!\n\nThe Most Prevalent Myths: What Are the Consequences of Confusion About #AI?\n\	fernandocuenca	andi_staub:
58	Sun Feb 18 19:03:25 +0000 2(	RT @devnullius: Did you have adaptive videos when you went to school? Customized to your personal likings, styles &a	itknowingness	devnullius:
59	Sun Feb 18 19:03:26 +0000 2(	RT @ProductiveSys: How #Blockchain and #SmartContracts Will Change Contract Management in 2018 – @icertis \n#	sajidmirza	ProductiveSys:
61	Sun Feb 18 19:03:28 +0000 2(	RT @MikeQuindazzi: Next in #agritech? #Robotics researchers are pioneering #robots for #farming and #driverless tra	JulioSilvaJr	MikeQuindazzi:
62	Sun Feb 18 19:03:28 +0000 2(	RT @appliedAIbook: All the matrix #calculus you need for #deeplearning has been conveniently packaged & clar	ryancbriggs	appliedAIbook:
63	Sun Feb 18 19:03:29 +0000 2(	RT @Gartner_inc: Discover, What to Do and Not Do With #AI, in this webinar: https:\/\/t.co\/zYE2XVRk33 @WhitAndrew	mskohli	Gartner_inc:
69	Sun Feb 18 19:03:31 +0000 2(	RT @singularity_net: ICYDK: \tAI is whatever hasn't been done yet.\" – Douglas Hofstadter \n#Sunday #Inspiration #AI #	DNBRS117	singularity_net:
70	Sun Feb 18 19:03:32 +0000 2(	RT @AI_Mentors: #RT @MktgSciences: Banking\u2019s Strategic Roadmap: Priorities and Investments for The Year A	Calcaware	AI_Mentors:
71	Sun Feb 18 19:03:32 +0000 2(	RT @ahmedjr_16: 8 Best SQL Courses Online To Advance Your #Career\n\nhttps:\/\/t.co\/IoRevkRT1l \n\n#AI #DataSc	Calcaware	ahmedjr_16:
73	Sun Feb 18 19:03:35 +0000 2(	RT @chboursin: The 11 clusters of Innovation\n\n#IoT #Entrepreneur #AI #tech #SMM #Fintech #startup #MachineLea	mixo72	chboursin:
74	Sun Feb 18 19:03:36 +0000 2(	RT @chboursin: The 11 clusters of Innovation\n\n#IoT #Entrepreneur #AI #tech #SMM #Fintech #startup #MachineLea	fernandocuenca	chboursin:
76	Sun Feb 18 19:03:36 +0000 2(	RT @AbyssCreations: Harmony's ready for her closeup, she loves getting in front of the camera. #harmonyAI #abysscre	LordRiehl	AbyssCreations:
77	Sun Feb 18 19:03:37 +0000 2(	RT @SaleMove: The biggest misconception about #artificialintelligence. #AI\nhttps:\/\/t.co\/lDh5WKtTHye https:\/\/t.co	JulioSilvaJr	SaleMove:
78	Sun Feb 18 19:03:38 +0000 2(	RT @ProductiveSys: #ArtificialIntelligence can help you protect your personal #data \n#AI #ML #DL #Algorithms #NLP	sajidmirza	ProductiveSys:
79	Sun Feb 18 19:03:38 +0000 2(	RT @ProductiveSys: #ArtificialIntelligence can help you protect your personal #data \n#AI #ML #DL #Algorithms #NLP	fernandocuenca	ProductiveSys:
80	Sun Feb 18 19:03:38 +0000 2(	RT @ipfconline1: #MachineLearning Translation and the Google Translate Algorithm\n\nhttps:\/\/t.co\/vubSXYZH60 v	Mada_GD	ipfconline1:
81	Sun Feb 18 19:03:39 +0000 2(	RT @bonzaiengineers: The evolution of #MachineLearning 2010 to 2040: https:\/\/t.co\/oZC0yeel4T via @MikeQuinda	hiciyasak	bonzaiengineers:
82	Sun Feb 18 19:03:40 +0000 2(	RT @terminus7ai: What #PredictiveAnalytics, #BigData And The Rise Of #ArtificialIntelligence Mean For Real Estate ht	JulioSilvaJr	terminus7ai:
84	Sun Feb 18 19:03:42 +0000 2(	RT @ileadgen: 7 Best #MobileApps #Development Courses for Beginners \n\nhttps:\/\/t.co\/Y1uprA7vZO \n\n#AppDe	Jadirectivestwt	ileadgen:

twitter_output3

Twitter API: Parsing the file to create data set II

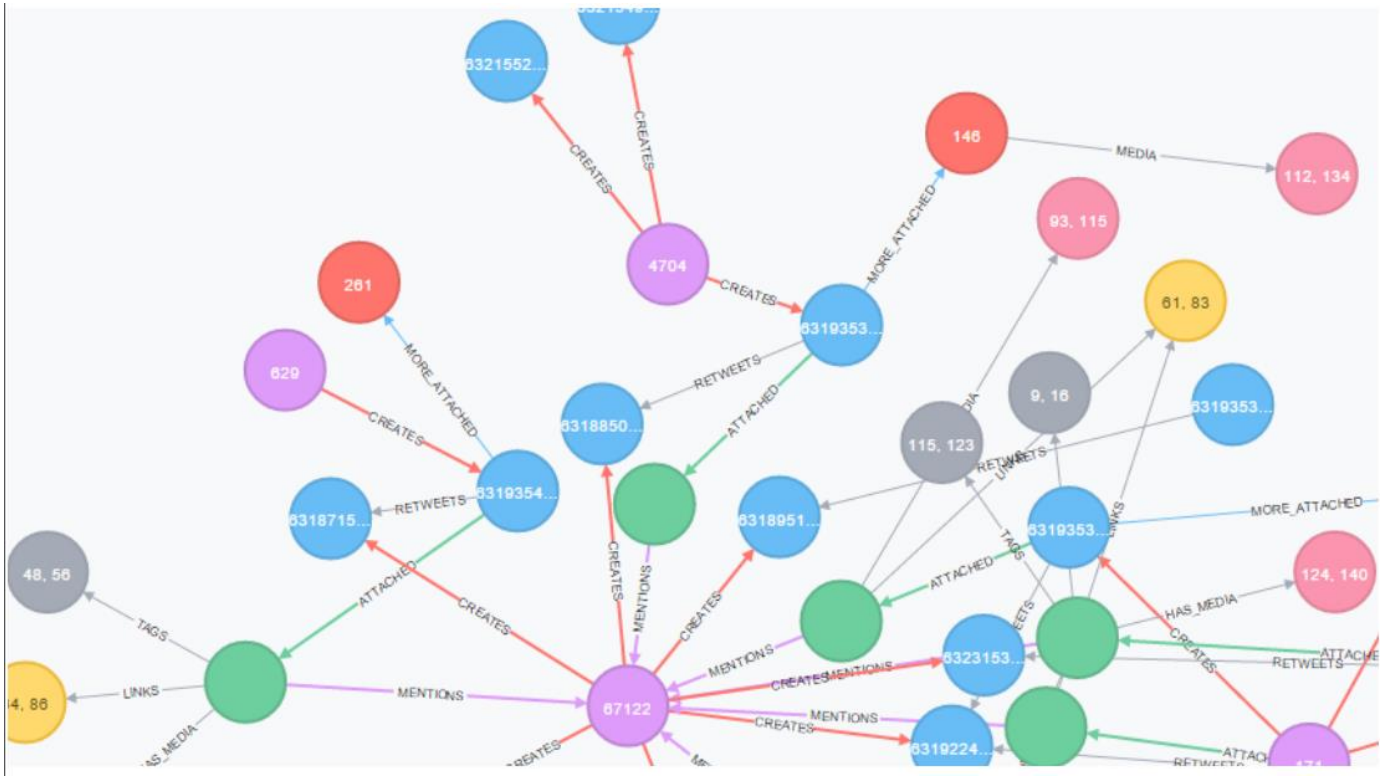
	A	B	C
1	Source	Target	Type
2	LordRiehl	AbyssCreations	Directed
3	JulioSilvaJr	guidaautonoma	Directed
4	WIOMAX_MD	MarexSolutions	Directed
5	LordRiehl	AbyssCreations	Directed
6	Masssoud	SpirosMargaris	Directed
7	Jadirectivestwt	reinforcelabtwt	Directed
8	OOOSergenio	singularity_net	Directed
9	FabienBrodie	lainLJBrown	Directed
10	JulioSilvaJr	iMariaJohnsen	Directed
11	Jadirectivestwt	ahmedjr_16	Directed
12	fernandocuenca	ahmedjr_16	Directed
13	nazkong	singularity_net	Directed
14	JulioSilvaJr	MikeQuindazzi	Directed
15	JulioSilvaJr	iMariaJohnsen	Directed
16	JulioSilvaJr	rajat_shrimal	Directed
17	BHSfpathways	mywowscotland	Directed
18	chan_zagorskas	emawware	Directed
19	JulioSilvaJr	SpirosMargaris	Directed
20	Jadirectivestwt	ahmedjr_16	Directed
21	JulioSilvaJr	Ronald_vanLoon	Directed
22	Jadirectivestwt	ahmedjr_16	Directed
23	Mada_GD	Ronald_vanLoon	Directed
24	Jadirectivestwt	ahmedjr_16	Directed
25	AI_Mentors	MktgSciences	Directed
26	fernandocuenca	ahmedjr_16	Directed
27	JulioSilvaJr	mjcavaretta	Directed
28	Jadirectivestwt	ahmedjr_16	Directed
29	Mada_GD	MikeQuindazzi	Directed
30	JulioSilvaJr	TheMegaTrends	Directed
31	48Diman369	AitheonOfficial	Directed
32	JulioSilvaJr	TopCyberNews	Directed
33	Mada_GD	andi_staub	Directed
34	fernandocuenca	andi_staub	Directed
35	itknowingness	devnullius	Directed
36	sajidmirza	ProductiveSys	Directed
37	JulioSilvaJr	MikeQuindazzi	Directed

Import to Gephi for vizualization



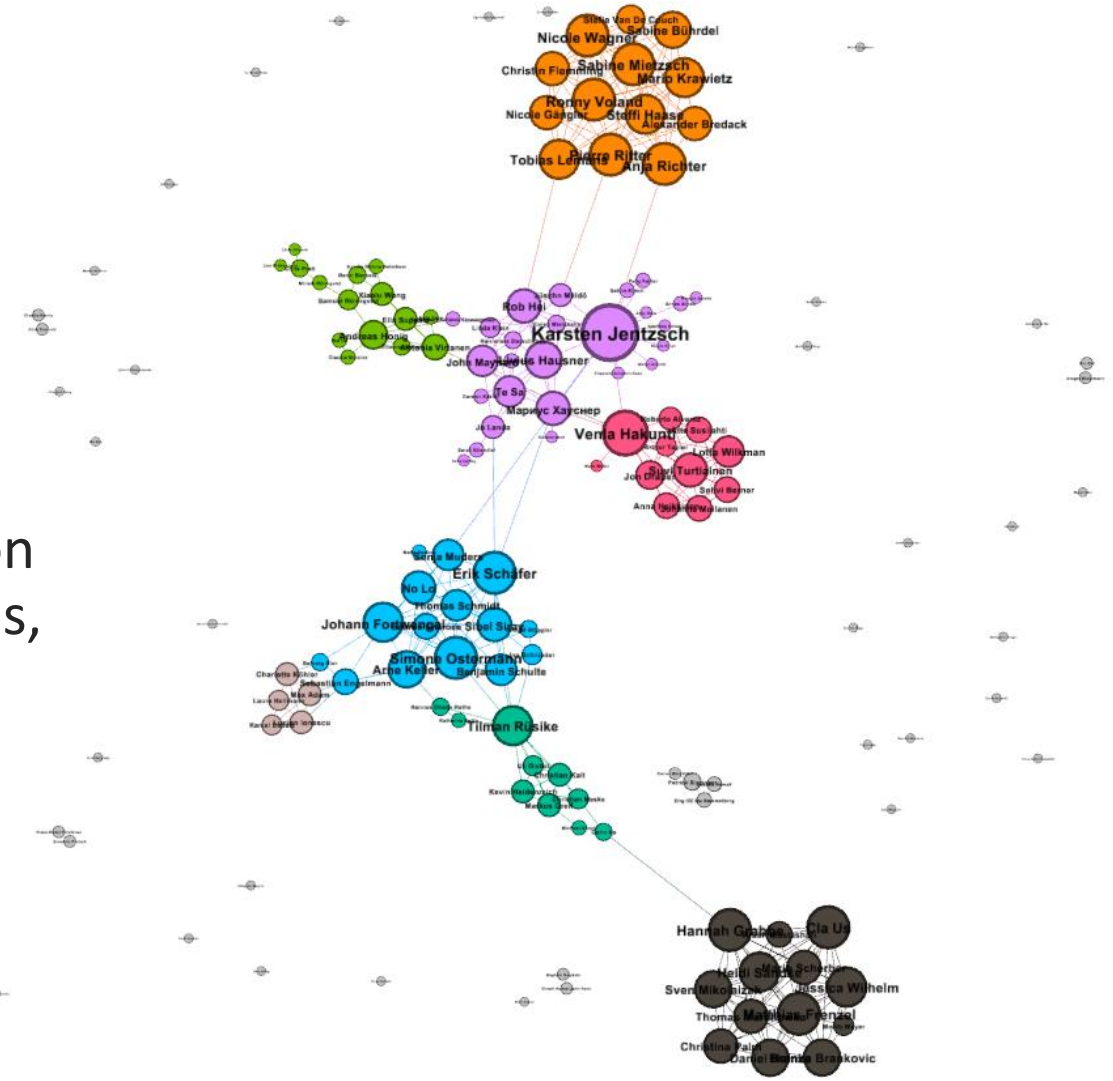
Extensions: Multipartite networks: Use a graph database (e.g. Neo4J)

Many kinds of nodes: Users, Tweets, URLs, Hashtags, Media
Many kinds of edges: User creates Tweets, A Tweet is in response to another Tweet, A Tweet retweets another, User mentions User, Tweet contains Hashtag, User follows User



One word about Facebook data…

- Facebook Graph API has been constantly closed currently very hard to extract relevant information
- Programs such as Netvizz help to extract information but limited to open groups, pages,…
- Possibly you have to use web scrapers to extract relevant data



One word about Github data…

- GitHub is a good source for information about socio-technical systems
- GitHub API reasonably easy to use to extract relevant information
- Currently less restricted with regards to extracting data from previous points in time

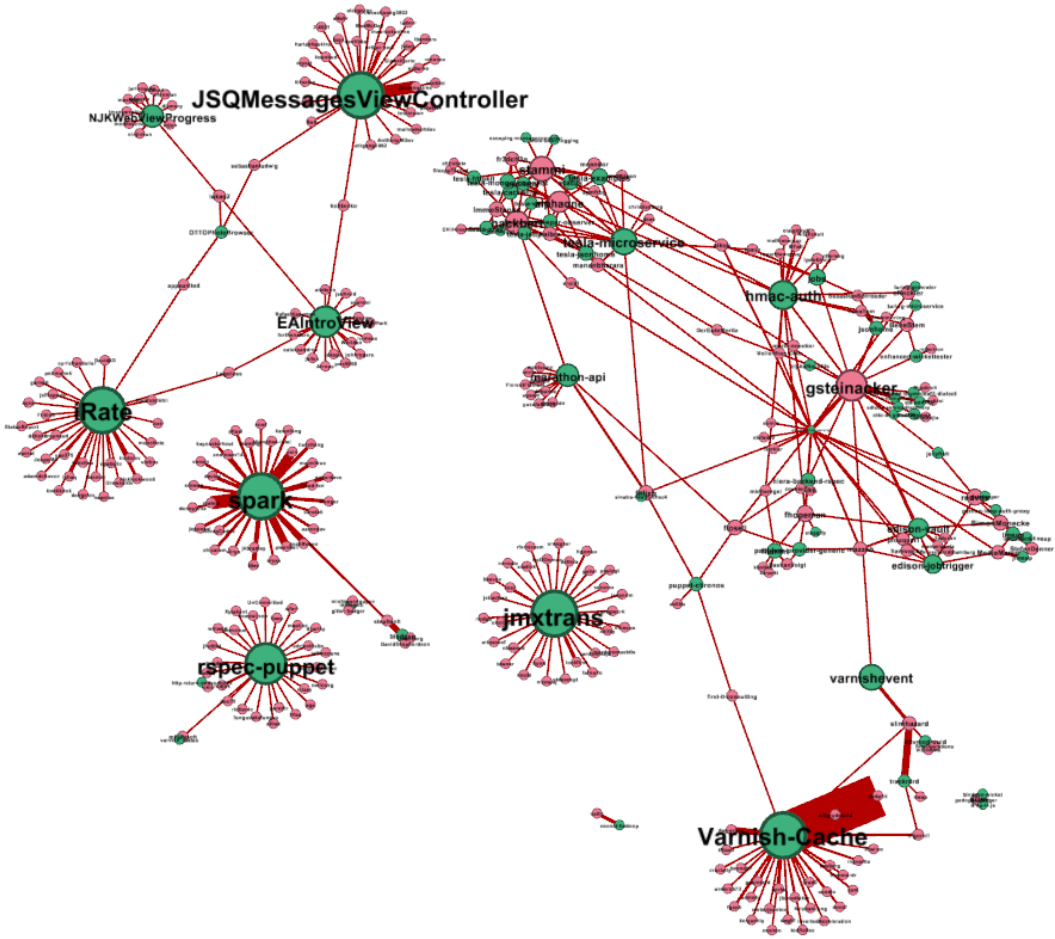


Fig.1: Otto.de open source repositorities & developer communities

Other potential sources

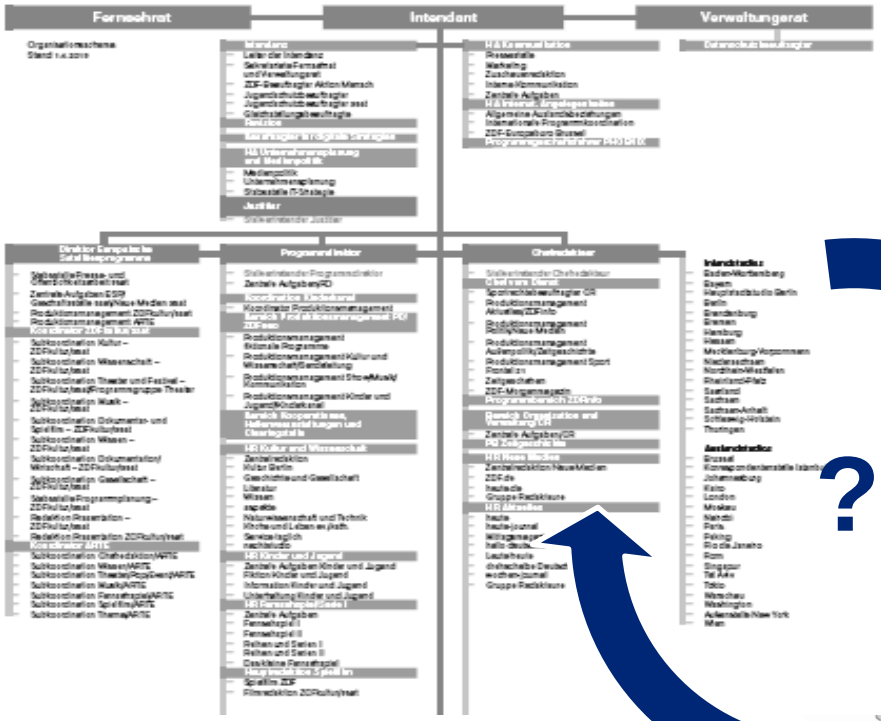
- Programmable Web provides information about available mashaups and API ecosystems
- LinkedIn has a phyton API and some tools to do automatic analysis (e.g. Socilab)

Agenda

1. Introduction and examples
2. Fundamental concepts
   1. Creating graphs
   2. Visualizing graphs
   3. Centrality
   4. Community detection
3. Getting and handling data
4. **Extended concepts**
5. Discussion and conclusion

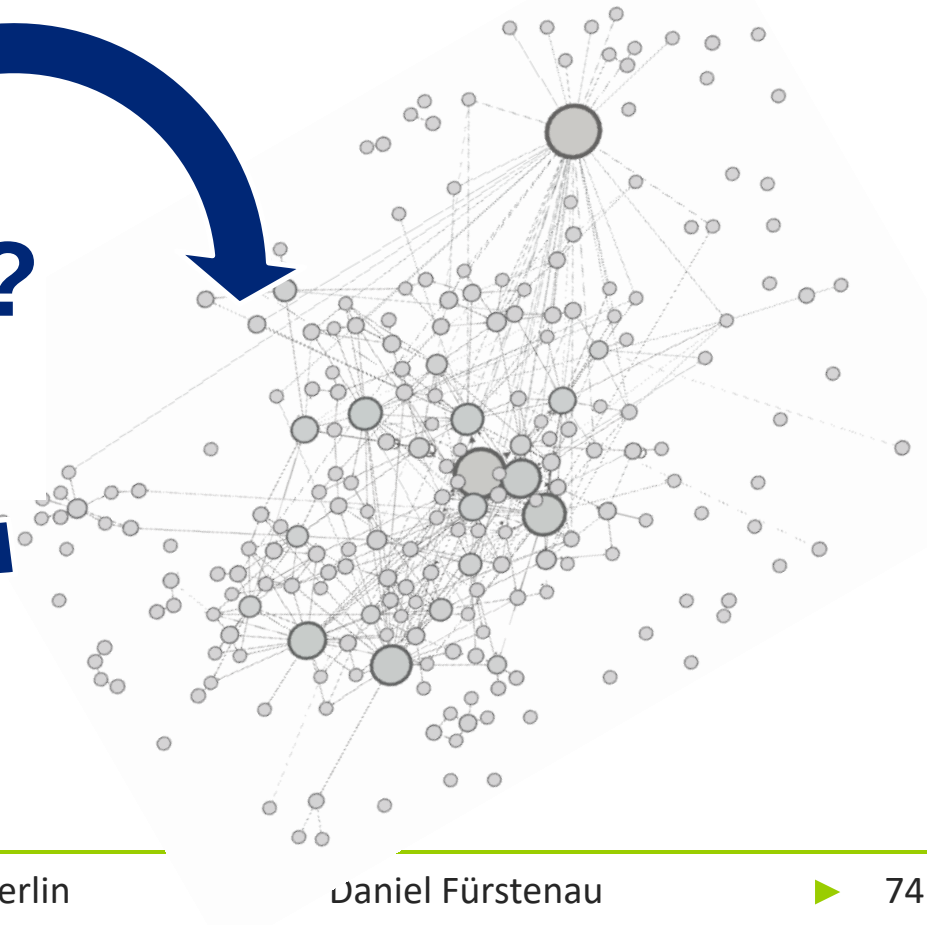
Designing a network study: Motivation

“Organizational Architecture”

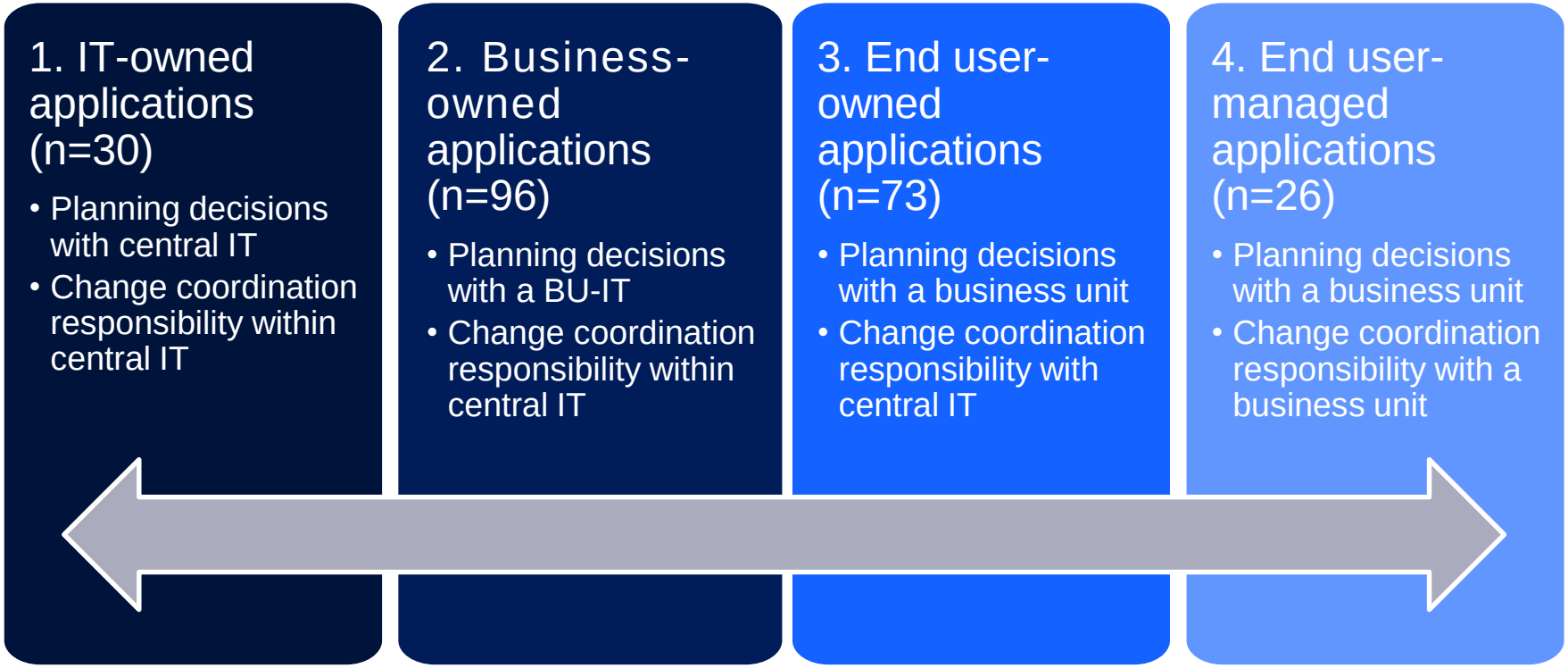


?

“IT Architecture”



Dependent Variable: Application Governance (Horizontal Allocation of Decision Rights)



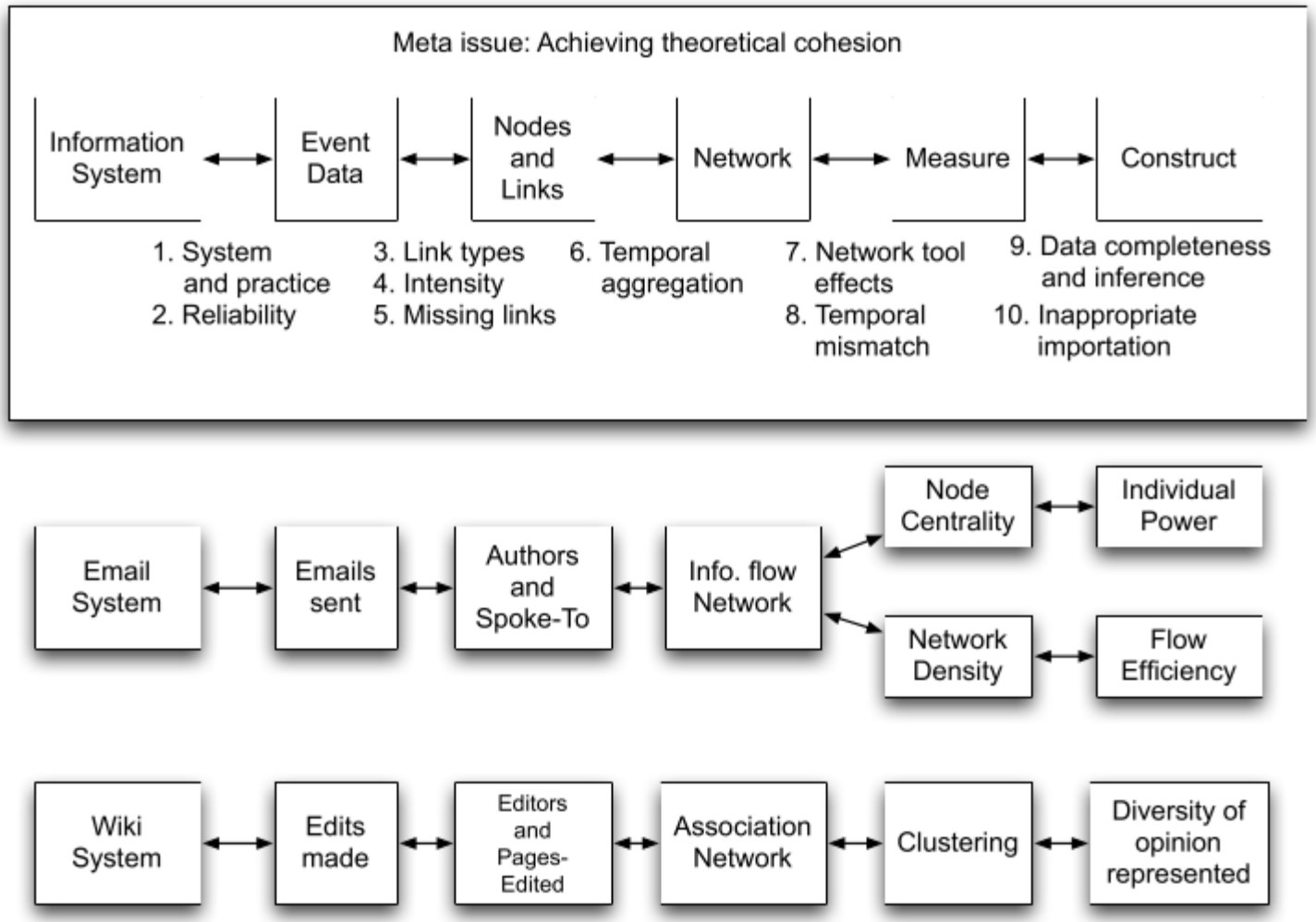
Dependent Variable: Application Governance (Horizontal Allocation of Decision Rights)

	Est.	Sig.
Scope of Use • The breadth to which the system is used throughout the organization • Operationalization: #users (binned into 5 classes) • H1: Higher scope of use → more IT decision rights	**-.325**	.002
Technical Specificity • The degree to which the system is customized to the organization • 5 classes: cloud sw, standard sw, mod. ssw., custom-modular, custom • H2: Higher technical specificity → more IT decision rights	-.189	.179
Positional Value • The number of other systems depending on this system • Operationalization: out-degree (numerical) • H3: Higher positional value → more IT decision rights	**-.098**	.001
Neighbor Governance • The extent to which neighboring systems are planned by business or IT units • Avg. planning decision rights of direct neighbors in graph (numerical) • H4: decision rights of neighbors → decision rights (positive correlation)	**1.023**	.000

Conclusions & Next Steps

- Evidence for an application-level complementarity between „organizational architecture“ and “IT architecture”
- Linking two formerly separated strands of literature
  - IT Governance (traditionally on „monolithic“ org-level)
  - IT Architecture (young stream of quantitative network views)
- Practical and theoretical implications: More differentiated, pluralistic view at IT Governance
- Caveat: Lack of criterion / limited prescriptiveness
- Next steps
  - Qualitative analysis of within-case outliers
  - Exploring reasons for deviating from statistical patterns

Chain of Reasoning and Validity Issues in Network Analysis with Digital Trace Data



[Howison et al. 2011]

Group exercise: What research design has been used in these studies?

Group 1:
Aral et al. (2009)

Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks

Sinan Aral[a,b,1], Lev Muchnik[a], and Arun Sundararajan[a]

[a]Information, Operations and Management Sciences Department, Stern School of Business, New York University, Kaufmann Management Center, 44 West 4th Street, New York, NY 10012; and [b]Center for Digital Business, Sloan School of Management, Massachusetts Institute of Technology, 5 Cambridge Center–NE25, Cambridge, MA 02142

Edited by Matthew O. Jackson, Stanford University, Stanford, CA, and accepted by the Editorial Board October 6, 2009 (received for review August 4, 2009)

Node characteristics and behaviors are often correlated with the structure of social networks over time. While evidence of this type of assortative mixing and temporal clustering of behaviors among linked nodes is used to support claims of peer influence and social contagion in networks, homophily may also explain such evidence. Here we develop a dynamic matched sample estimation framework to distinguish influence and homophily effects in dynamic networks, and we apply this framework to a global instant messaging network of 27.4 million users, using data on the day-by-day adoption of a mobile service application and users' longitudinal behavioral, demographic, and geographic data. We find that previous methods overestimate peer influence in product adoption decisions in this network by 300–700%, and that homophily explains >50% of the perceived behavioral contagion. These findings and methods are essential to both our understanding of the mechanisms that drive contagions in networks and our knowledge of how to propagate or counteract them in domains as diverse as epidemiology, marketing, development economics, and public health.

dynamic matching estimation | peer influence | social networks | identification

The recent availability of massive networked data sets has enabled studies of population-level human interaction at unprecedented scale (1–3). Such studies document the persistent structural properties of networks (4), how they form, evolve, and dissolve (5), and how their structure is correlated with social interaction (1, 6, 7), individual and collaborative team performance (8–11), health outcomes (12–14), and global product demand patterns (15). Networks of interactions among individuals also provide the primary pathways along which viral contagions spread in social, biological, technological, and economic systems (16–18), which may explain why network structure is correlated with such a variety of outcomes. Yet although many studies model the dynamics of viral spreading by using assumptions about susceptibility rates, transition probabilities, and their relationships to network structure, few large-scale empirical observations of networked contagions exist to validate these assumptions (16–18).

We analyze a new, large scale dataset which comprehensively captures the diffusion of a mobile service product over a social network for 5 months after its launch date. A key challenge in identifying true contagions in such data is to distinguish peer-to-peer influence, in which a node influences or causes outcomes in its neighbors, from homophily, in which dyadic similarities between nodes create correlated outcome patterns among neighbors that merely mimic viral contagions without direct causal influence (19). Although the diffusion patterns created by peer influence-driven contagions and homophilous diffusion are similar, they are likely to result in significantly different dynamics. Influence-driven contagions are self-reinforcing and display rapid, exponential, and less predictable diffusion as they evolve (18, 20), whereas homophily-driven diffusion processes are governed by the distributions of characteristics over nodes. These distinctions make distinguishing true contagions from homophilous diffusions at early stages important for the success or failure of contagion management efforts.

As more of a perceived contagion is explained by homophily rather than peer influence, intervention strategies should shift from peer-to-peer methods based on network structure to outreach based on population segmentation across individuals' characteristics. Formal procedures for separating influence and homophily are therefore essential to support policies that encourage or discourage the spread of behaviors in networks, from health interventions to viral marketing campaigns.

Contagions and homophilous diffusion are both typified by correlations between network structure and individual outcomes over time (1–3, 5–11, 17, 21, 22). Two empirical patterns have been used to substantiate claims of peer influence and contagion in networks (*i*) *assortative mixing*—correlations of behaviors among linked nodes (23, 24)—and (*ii*) *temporal clustering*—temporal interdependence of behaviors among linked nodes (12–14, 25–27). Because peer influence is likely to lead to assortative mixing, some studies claim assortative mixing is evidence of peer influence (12–14, 25–27). Evidence of temporal clustering is used to corroborate these claims because as Anagnostopoulos et al. (25) argue "if influence does not play a role, even though an agent's probability of activation could depend on her friends, the timing of such activation should be independent of the timing of other agents." Yet, while evidence of assortative mixing and temporal clustering in outcomes may indicate peer influence, social contagion, and viral spreading, such outcomes may also be explained by homophily—the demographic, technological, behavioral, and biological similarities of linked nodes (28). If ties are more likely between similar nodes, their outcomes could be correlated because of inherent similarities in their characteristics rather than as a consequence of their interactions. On one hand, linked nodes may directly influence one another to exhibit similar outcomes, creating viral contagions. On the other hand, linked nodes may simply have greater likelihoods of displaying correlated outcomes, in time and in network space, as a consequence of their similarities.

Here we develop a matched sample estimation framework to distinguish influence and homophily effects in dynamic networks, and we apply this framework to a unique dataset documenting product adoption in a large network. We find that previous methods significantly overestimate peer influence in this network, mistakenly identifying homophilous diffusion as influence-driven contagion.

Data

We apply our statistical framework to a longitudinal dataset that combines: (*i*) the global network of daily instant messaging (IM) traffic among 27.4 million users of Yahoo.com (Fig. 1) with (*ii*) data on the day-by-day adoption of a mobile service application

Author contributions: S.A., L.M., and A.S. designed research, performed research, contributed new reagents/analytic tools, analyzed data, and wrote the paper.

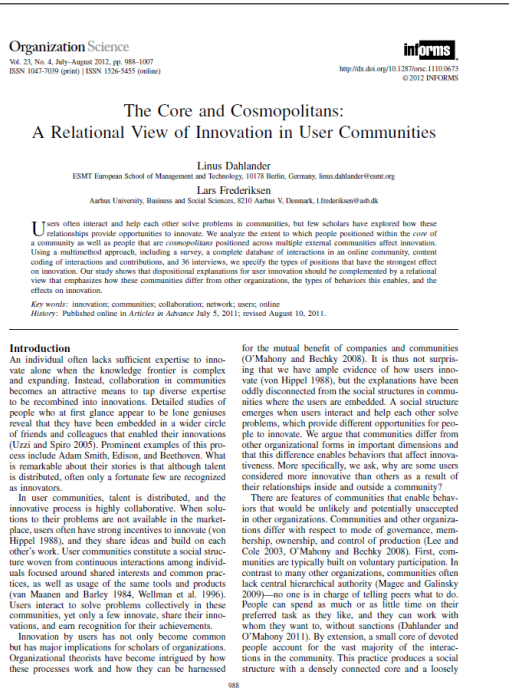
The authors declare no conflict of interest.

This article is a PNAS Direct Submission. M.O.J. is a guest editor invited by the Editorial Board.

[1]To whom correspondence should be addressed. E-mail: sinan@stern.nyu.edu.

This article contains supporting information online at www.pnas.org/cgi/content/full/0908800106/DCSupplemental.

Group 2:
Dahlander & Fredriksen (2012)



Organization Science
Vol. 23, No. 4, July–August 2012, pp. 988–1007
ISSN 1047-7039 (print) | ISSN 1526-5455 (online)
http://dx.doi.org/10.1287/orsc.1110.0673

The Core and Cosmopolitans:
A Relational View of Innovation in User Communities

Linus Dahlander
ESMT European School of Management and Technology, 10178 Berlin, Germany, linus.dahlander@esmt.org
Lars Frederiksen
Aarhus University, Business and Social Sciences, 8210 Aarhus V, Denmark, l.frederiksen@asb.dk

Users often interact and help each other solve problems in communities, but few scholars have explored how these relationships provide opportunities to innovate. We analyze the extent to which people positioned within the *core* of a community as well as people that are *cosmopolitans* positioned across multiple external communities affect innovation. Using a multimethod approach, including a survey, a complete database of interactions in an online community, content coding of interactions and contributions, and 36 interviews, we specify the types of positions that have the strongest effect on innovation. Our study shows that dispositional explanations for user innovation should be complemented by a relational view that emphasizes how these communities differ from other organizations, the types of behaviors this enables, and the effects on innovation.

Key words: innovation; communities; collaboration; network; users; online
History: Published online in *Articles in Advance* July 5, 2011; revised August 10, 2011.

Introduction

An individual often lacks sufficient expertise to innovate alone when the knowledge frontier is complex and expanding. Instead, collaboration in communities becomes an attractive means to tap diverse expertise to be recombined into innovations. Detailed studies of people who at first glance appear to be lone geniuses reveal that they have been embedded in a wider circle of friends and colleagues that enabled their innovations (Uzzi and Spiro 2005). Prominent examples of this process include Adam Smith, Edison, and Beethoven. What is remarkable about their stories is that although talent is distributed, often only a fortunate few are recognized as innovators.

In user communities, talent is distributed, and the innovative process is highly collaborative. When solutions to their problems are not available in the marketplace, users often have strong incentives to innovate (von Hippel 1988), and they share ideas and build on each other's work. User communities constitute a social structure woven from continuous interactions among individuals focused around shared interests and common practices, as well as usage of the same tools and products (van Maanen and Barley 1984, Wellman et al. 1996). Users interact to solve problems collectively in these communities, yet only a few innovate, share their innovations, and earn recognition for their achievements.

Innovation by users has not only become common but has major implications for scholars of organizations. Organizational theorists have become intrigued by how these processes work and how they can be harnessed for the mutual benefit of companies and communities (O'Mahony and Bechky 2008). It is thus not surprising that we have ample evidence of how users innovate (von Hippel 1988), but the explanations have been oddly disconnected from the social structures in communities where the users are embedded. A social structure emerges when users interact and help each other solve problems, which provide different opportunities for people to innovate. We argue that communities differ from other organizational forms in important dimensions and that this difference enables behaviors that affect innovativeness. More specifically, we ask, why are some users considered more innovative than others as a result of their relationships inside and outside a community?

There are features of communities that enable behaviors that would be unlikely and potentially unaccepted in other organizations. Communities and other organizations differ with respect to mode of governance, membership, ownership, and control of production (Lee and Cole 2003, O'Mahony and Bechky 2008). First, communities are typically built on voluntary participation. In contrast to many other organizations, communities often lack central hierarchical authority (Magee and Galinsky 2009)—no one is in charge of telling peers what to do. People can spend as much or as little time on their preferred task as they like, and they can work with whom they want to, without sanctions (Dahlander and O'Mahony 2011). By extension, a small core of devoted people account for the vast majority of the interactions in the community. This practice produces a social structure with a densely connected core and a loosely

Group 3:
Basole et al. (2015)

Understanding Business Ecosystem Dynamics: A Data-Driven Approach

6

RAHUL C. BASOLE, Georgia Institute of Technology
MARTHA G. RUSSELL, Stanford University
JUKKA HUHTAMÄKI, Tampere University of Technology
NEIL RUBENS, University of Electro-Communications
KAISA STILL, VTT Technical Centre of Finland
HYUNWOO PARK, Georgia Institute of Technology

Business ecosystems consist of a heterogeneous and continuously evolving set of entities that are interconnected through a complex, global network of relationships. However, there is no well-established methodology to study the dynamics of this network. Traditional approaches have primarily utilized a single source of data of relatively established firms; however, these approaches ignore the vast number of relevant activities that often occur at the individual and entrepreneurial levels. We argue that a data-driven visualization approach, using both institutionally and socially curated datasets, can provide important complementary, triangulated explanatory insights into the dynamics of interorganizational networks in general and business ecosystems in particular. We develop novel visualization layouts to help decision makers systemically identify and compare ecosystems. Using traditionally disconnected data sources on deals and alliance relationships (DARs), executive and funding relationships (EFRs), and public opinion and discourse (POD), we empirically illustrate our data-driven method of data triangulation and visualization techniques through three cases in the mobile industry Google's acquisition of Motorola Mobility, the coopetitive relation between Apple and Samsung, and the strategic partnership between Nokia and Microsoft. The article concludes with implications and future research opportunities.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data mining*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Graphical user interfaces*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*

General Terms: Design, Measurement, Performance

Additional Key Words and Phrases: Data triangulation, information visualization, interorganizational networks, business ecosystem

ACM Reference Format:
Rahul C. Basole, Martha G. Russell, Jukka Huhtamäki, Neil Rubens, Kaisa Still, and Hyunwoo Park. 2015. Understanding business ecosystem dynamics: A data-driven approach. ACM Trans. Manag. Inform. Syst. 6, 2, Article 6 (May 2015), 32 pages.
DOI: http://dx.doi.org/10.1145/2724730

Authors' addresses: R. C. Basole, Georgia Institute of Technology, School of Interacting Computing & Tennenbaum Institute, Atlanta, Georgia 30332 USA; email: basole@gatech.edu; M. G. Russell, mediaX at Stanford University, Stanford, California 94305-4115, USA; email: martha.russell@stanford.edu; J. Huhtamäki, Tampere University of Technology, Department of Mathematics, Korkeakoulunkatu 3, FI-33720 Tampere, Finland; email: jukka.huhtamaki@tut.fi; N. Rubens, Graduate School of Information Systems, University of Electro-Communications, Tokyo 182-8585, Japan; email: rubens@is.uec.ac.jp; K. Still, VTT Technical Research Centre of Finland Ltd, Kaitovayla 1, FI-90571 Oulu, Finland; email: kaisa.still@vtt.fi; H. Park, Georgia Institute of Technology, School of Industrial and Systems Engineering & Tennenbaum Institute, 755 Ferst Drive, Atlanta, Georgia 30332; email: hwpark@gatech.edu.

DOI: http://dx.doi.org/10.1145/2724730

Discussion

Agenda

1. Introduction and examples
2. Fundamental concepts
    1. Creating graphs
    2. Visualizing graphs
    3. Centrality
    4. Community detection
3. Getting and handling data
4. Extended concepts
5. **Discussion and conclusion**

Exercise: Design your own network study.

Discussion

EINSTEIN CENTER Digital Future

Freie Universität Berlin

Department of Information Systems

School of Business & Economics

PhD Course on Social Network Analysis

Prof. Dr. Daniel Fürstenau

20 February 2017