
Network Analysis of Tweets

Building and graphing networks of users and tweets

Outline

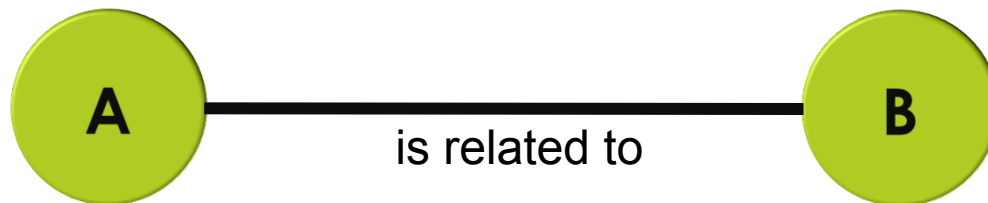
- Introduction to Graphs
 - Not "charting"
 - Constructing Network Structure
 - build_graph.py
 - Parsing tweets
 - Using networkx
 - Presentation of Graph data
 - Gephi
-

Basic Idea of a Graph

- “A” is related to “B”
 - Three things we want to represent
 - Item, person, concept, thing: “A”
 - Item, person, concept, thing: “B”
 - The “is related to” relation

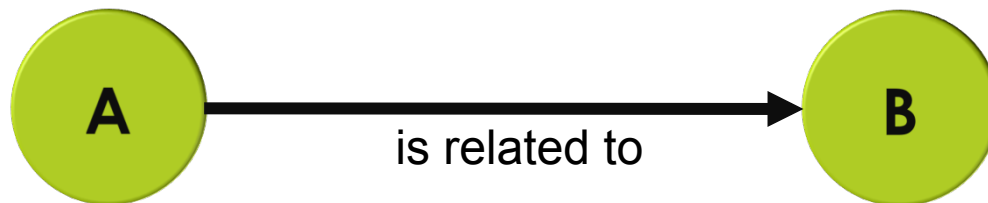
Basic Idea of a Graph

- “A” is related to “B”
 - Three things we want to represent
 - Item, person, concept, thing: “A”
 - Item, person, concept, thing: “B”
 - The “is related to” relation



Basic Idea of a Graph

- “A” is related to “B”
 - Three things we want to represent
 - Item, person, concept, thing: “A”
 - Item, person, concept, thing: “B”
 - The “is related to” relation
 - Relation can be directed, directional



Who Retweets Whom?

- AidanKellyUSA: RT @BreeSchaaf: Quick turn on @NBCSports Men's Singles Luge final! @mazdzer @AidanKellyUSA @TuckerWest1 are laying it all on the line tonig...
- Setrice93: RT @NBCOlympics: #Gold for @sagekotsenburg! First gold at #Sochi2014 and first-ever Olympic gold in snowboard slopestyle! <http://t.co/0F8ys...>
- adore_knob: RT @drdloveswater: I have waited 4 years to do this. Thank you @NBCOlympics & all your interns for such awesome coverage. #Sochi2014 <http://...>
- MattJanik: RT @NBCOlympics Yeah, it's not good for your health.
- LisaKSimone: RT @robringham: Tired of @nbc / @NBCOlympics holding the Olympics hostage. Time for them to lose exclusivity. #NBCFail
- TS_Krupa: RT @NBCOlympics: RT @RedSox: Go Team USA!! @USOlympic #Sochi2014 <http://t.co/anvneh5Mmy>

Who Retweets Whom?

- AidanKellyUSA: RT @BreeSchaaf: Quick turn on @NBCSports Men's Singles Luge final! @mazdzer @AidanKellyUSA @TuckerWest1 are laying it all on the line tonig...

AidanKellyUSA

BreeSchaaf

- Setrice93: RT @NBCOlympics: #Gold for @sagekotsenburg! First gold at #Sochi2014 and first-ever Olympic gold in snowboard slopestyle! <http://t.co/0F8ys...>

Setrice93

NBCOlympics

- adore_knob: RT @drdloveswater: I have waited 4 years to do this. Thank you @NBCOlympics & all your interns for such awesome coverage. #Sochi2014 <http://...>

adore_knob

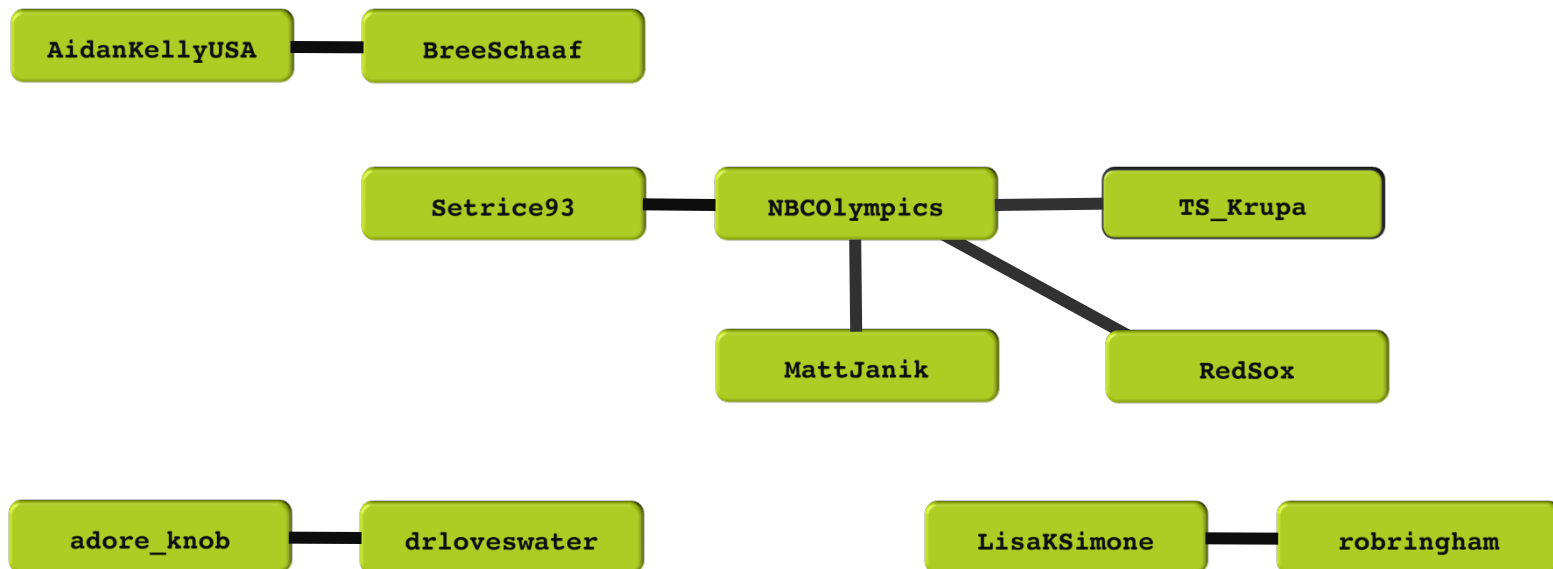
drloveswater

- MattJanik: RT @NBCOlympics Yeah, it's not good for your health.

- LisaKSimone: RT @robringham: Tired of @nbc / @NBCOlympics holding the Olympics hostage. Time for them to lose exclusivity. #NBCFail

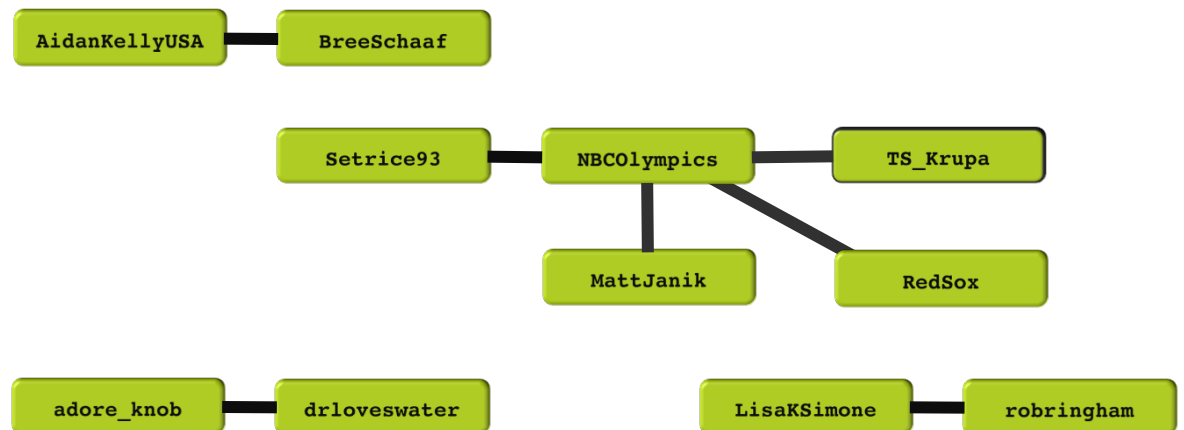
- TS_Krupa: RT @NBCOlympics: RT @RedSox: Go Team USA!! @USOlympic #Sochi2014 <http://t.co/anvneh5Mmy>

Who Retweets Whom?



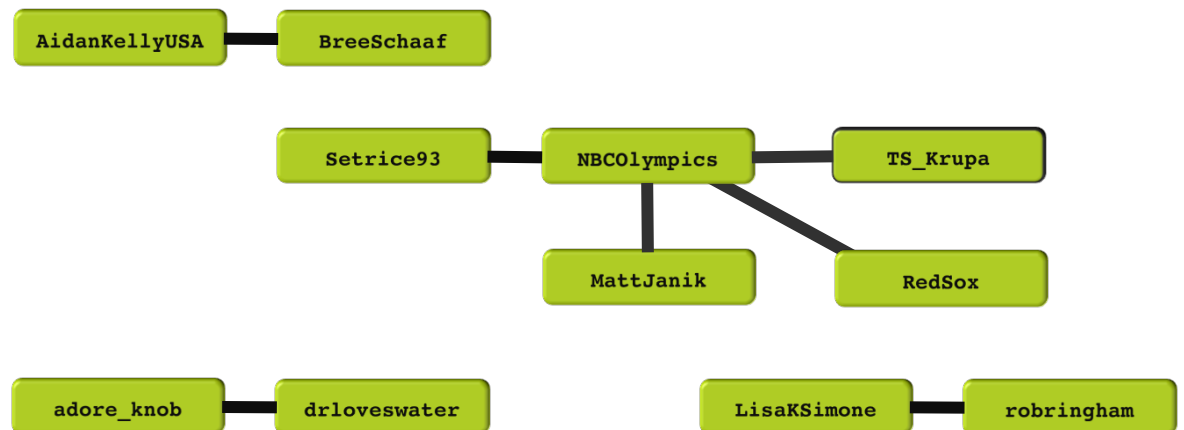
Some graph concepts

- Nodes – related items
 - Number
- Edges – the relations
 - Number



Some graph concepts

- Component (connected component)
 - A connected “chunk” of the whole thing
 - Example is one graph with four connected components
- Subgraph
 - Graph that can be found within another graph



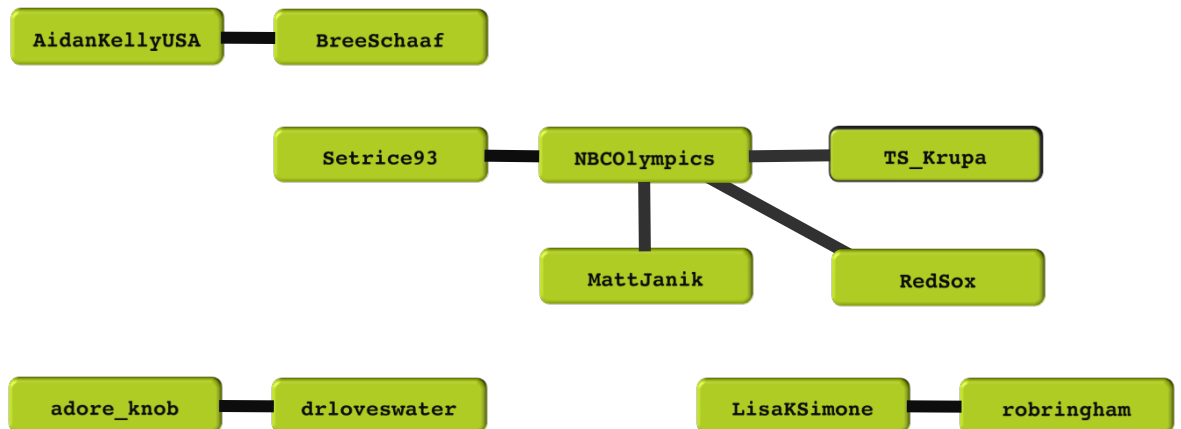
Some graph concepts

- Complete Graph

- A graph where every node is connected to every other node

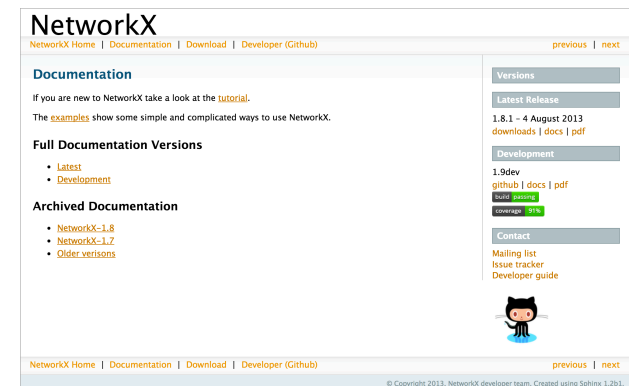
- Clique

- A complete subgraph, a complete graph found within a graph

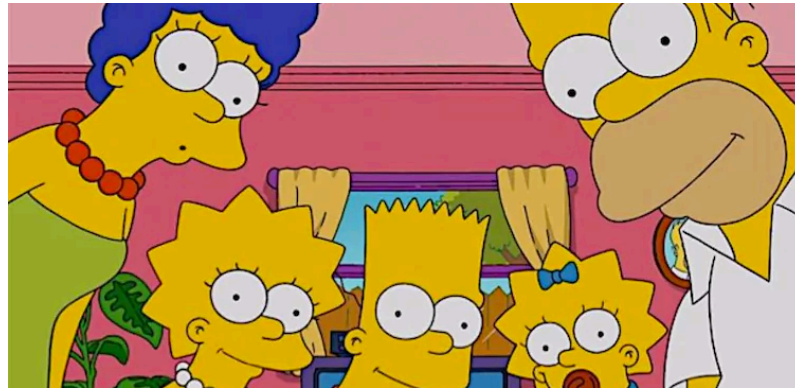


Networkx

- Networkx, a python graph creation and analysis tool
 - <http://networkx.github.io/>
- Good Documentation
 - <https://networkx.github.io/documentation/stable/>



How are we related?



Simpson's Social Network

```
import networkx as nx
g = nx.Graph() ## create a new undirected graph
# Add nodes and edges
g.add_edge("bart", "marge")
g.add_edge("homer", "marge")
g.add_edge("lisa", "marge")
g.add_edge("maggie", "marge")
g.add_edge("patty", "marge")
g.add_edge("selma", "marge")
g.add_edge("homer", "lisa")
g.add_edge("homer", "maggie")
g.add_edge("homer", "bart")
g.add_edge("ned", "todd")
g.add_edge("ned", "rod")
g.add_edge("ned", "maude")
g.add_edge("todd", "maude")
g.add_edge("rod", "maude")
```

Networks demo (nodes, edges)

```
# Print the number of nodes in the graph
print len(g.nodes())
# Print nodes - Just a list of the node names
print g.nodes()
# Print edges - A list of *node pairs*
print g.edges()
# Find all edges incident on one node - node pairs
print g.edges("marge")
# get the subgraph of all nodes around marge
nl = [ n[1] for n in g.edges("marge") ]
nl.append("marge")
sg = nx.Graph(g.subgraph(nl))
print sg.nodes()
print sg.edges()
```

Networks demo (calculations)

```
# some basic graph info
print nx.info(g)
# edge calculations
print nx.degree(g, "marge")
print nx.density(g)
# some centrality measures
print nx.degree_centrality(g)
print nx.betweenness_centrality(g)
print nx.eigenvector_centrality(g)
# find cliques
gclique = list(nx.find_cliques(g))
print gclique
# find connected components
comps = nx.connected_components(g)
print len(comps)
print comps[0]
print comps[1]
```


Tweet Networks

- Want code to build a retweet network
 - Collect tweets
 - For each tweet find if it's a retweet
 - Link the retweeting user to the retweeted user
- Based on the text retweet convention
 - RT @dwmcphd <some tweet text>
- Command line usage
- Code walk through

Command Line

- ▣ Samples are in the directories
 - ▣ hcde/data/election_2012
 - ▣ hcde/data/election_2016
 - ▣ hcde/data/oscar_2016

```
python build_graph.py
```

```
USAGE: build_graph.py -date <date> -save <filename> [-dur <days>] [-digraph] [-weighted] [-edge_cut] [-comp] [-dot | -graphml] [-report | -no_report]
```

Create a Graph (election 2012)

- ▣ `python build_graph.py -date 20121015 -save class_sample -edge_cut -graphml -report`
 - ▣ October 15, 2012 (election data)
 - ▣ Saving filename as 'class_sample'
 - ▣ Perform a single edge cut, remove singleton edges, nodes
 - ▣ Write the file in GraphML
 - ▣ Report activity to the screen (who is retweeting who)

< ... lots of text scrolls by ... >

Graph has 62751 nodes and 92841 edges.

Performing recursive single edge cut.

Made 6 passes through the graph, cut 45223 edges and 50905 nodes.

Graph has 11846 nodes and 47618 edges.

Writing GraphML file: `class_sample-20121015-dur01-edge_cut.graphml`

build_graph.py

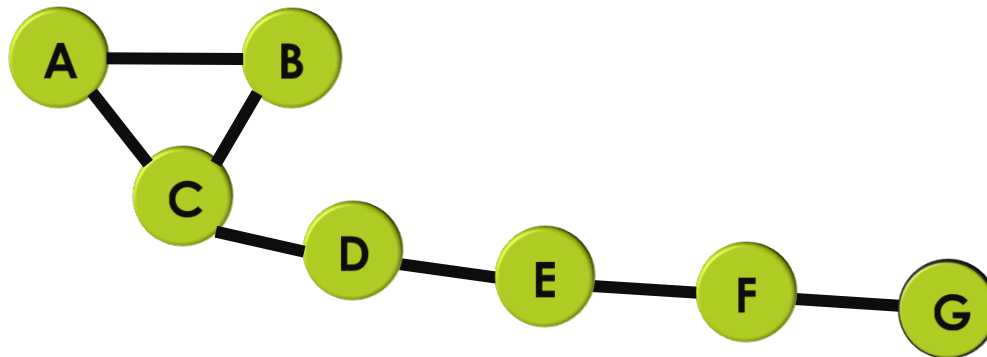
- ▣ Looking through the code

Caveats

- Can build directed or undirected graph
 - Any/all retweets create a connection between users
 - Single edge cut (recursive)
-

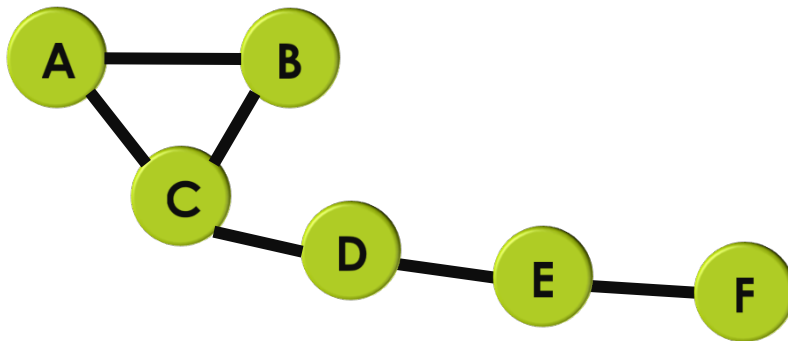
Single Edge Cut

- Do we want long chains of retweets, or main clump of people?



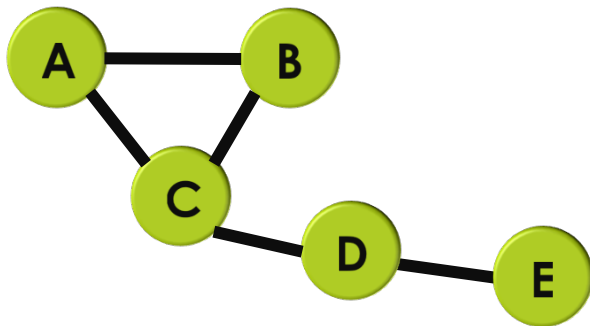
Single Edge Cut

- Do we want long chains of retweets, or main clump of people?



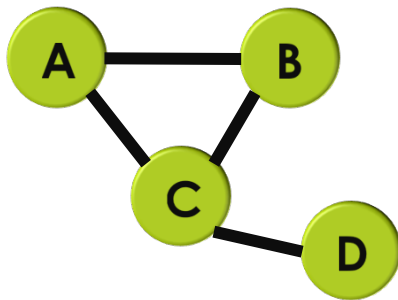
Single Edge Cut

- Do we want long chains of retweets, or main clump of people?



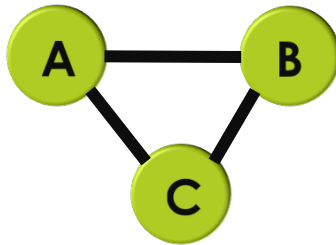
Single Edge Cut

- Do we want long chains of retweets, or main clump of people?



Single Edge Cut

- Do we want long chains of retweets, or main clump of people?



Visualizing graph data

- build_graph.py
 - Can dump GraphML (Graph Markup Language)
 - Good for Gephi (static picture, desktop app)
 - Can dump a “dot” file
 - Good for GraphVis (old, crufty, command line tool)
- Possible modifications to build_graph.py
 - Could be modified to use JSON output in Networkx
 - Maybe useful for Plotly or D3


Gephi

■ <https://gephi.org/>

■ Great tool – useful

■ Possible to use Plotly

■ <https://plot.ly/python/network-graphs/>

 **Gephi**
makes graphs handy

Download Blog Store Wiki Forum Support Bug tracker

Home Features Learn Develop Plugins Services Consortium

The Open Graph Viz Platform

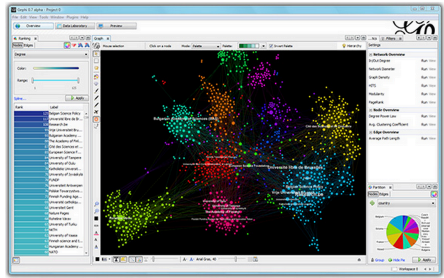
Gephi is an interactive visualization and exploration **platform** for all kinds of networks and complex systems, dynamic and hierarchical graphs.

Runs on Windows, Linux and Mac OS X. Gephi is open-source and free.

[Learn More on Gephi Platform >](#)


[Download FREE Gephi 0.8.2-beta](#)
[Release Notes](#) | [System Requirements](#)

► [Features](#) ► [Screenshots](#)
► [Quick start](#) ► [Videos](#)



Support us! We are **non-profit**. Help us to **innovate** and **empower** the community by donating only 8€:

[Donate](#)



APPLICATIONS

- ✓ **Exploratory Data Analysis:** intuition-oriented analysis by networks manipulations in real time.
- ✓ **Link Analysis:** revealing the underlying structures of associations between objects, in particular in scale-free networks.
- ✓ **Social Network Analysis:** easy creation of social data connectors to map community organizations and small-world networks.
- ✓ **Biological Network analysis:** representing patterns of biological data.
- ✓ **Poster creation:** scientific work promotion with hi-quality printable maps.

[Learn More >](#)

METRICS READY

- ✓ **Centrality:** used in sociology to indicate how well a node is connected. Available: degree (power-law), betweenness, closeness.

[Learn More >](#)

“Like Photoshop™ for graphs.”


— the Community

LATEST NEWS

- [Google Summer of Code 2013](#)
April 15, 2013
- [Rebuilding Gephi's core for the 0.9 versi..](#)
March 5, 2013
- [A month of Gephi Marketplace](#)
February 20, 2013
- [rgexf: An R library to work with GEXF graph fil..](#)
February 12, 2013
- [Graph visualization meet-up in Paris](#)
January 9, 2013

[See All >](#)

PAPERS



Bastian M., Heymann S., Jacomy M. (2009). *Gephi: an open source software for exploring and manipulating networks*. International AAAI Conference on Weblogs and Social Media. From AAAI [PDF].

LATEST FORUM TOPICS

- [Create two different clusters](#)
February 09, 2014
- [Create non-uniform clusters](#)

Gephi Tutorial

■ <https://gephi.org/users/>

■ Tutorial is very good

■ You should do it



The screenshot shows the Gephi website homepage. The header includes the Gephi logo and tagline 'makes graphs handy', along with navigation links: Home, Features, Learn, Develop, Plugins, Services, Consortium. A sidebar on the right contains buttons for 'Download Now', 'Stay informed', 'Contribute', 'Report a Bug', 'Request a Feature', and 'Share your ideas'. The main content area is titled 'Learn how to use Gephi' and includes a welcome message, a 'Getting Started' section with links to 'Quick Start Guide' and 'Supported Graph File Formats', and an 'Official Tutorials' section with three tutorial cards: 'Quick Start Guide', 'Tutorial Visualization', and 'Tutorial Layouts'. Below these are links to 'Popular Tutorials by the Community' and 'Various Tutorials in Video'.

Learn how to use Gephi

Welcome to Gephi! Gephi is an open-source software for visualizing and analysing large networks graphs. Gephi uses a 3D render engine to display graphs in real-time and speed up the exploration. You can use it to explore, analyse, spatialise, filter, clusterize, manipulate and export all types of graphs.

Getting Started

New to Gephi? These guides are for you. The following pages should introduce you to the Gephi project, basic features and installation.

- ▶ **Quick Start Guide**
- ▶ **Supported Graph File Formats**
 - Goals and Principles of the project
 - System Requirements, Installation, Release Notes
 - User interface description
 - Exported File Formats

Official Tutorials

Gephi is really easy to handle if you learn the basics. Let's follow these tutorials to quickly manage the main features!

Quick Start Guide	Tutorial Visualization	Tutorial Layouts

and others:

- How to Import Spreadsheet (Excel) Data // video
- How to Import Dynamic Data

Popular Tutorials by the Community:

- Using Netviz & Gephi to Analyze a Facebook Network
- Getting Started With The Gephi – My Facebook Network
- Dynamic Networks in Gephi: From Twapperkeeper to GEXF
- Generating graphs of retweets and @-messages on Twitter using R and Gephi
- Text Network Analysis
- Visualize keywords and landing pages from Google Analytics

Various Tutorials in Video:

Visualization

Possible Modifications

- build_graph.py
 - Network of @mentions (who mentions who)
 - Possibly directed graph
 - Network of #hashtag use (who uses which hashtags)
 - Is what we call a 2 mode network
 - Extract, save Component
 - Extract, save Clique

