Getting an Edge with Network Analysis in Python

A Gentle Introduction

Alon Nir
(@alonnir)

PyData Berlin
20/5/20

About Me

- Senior data scientist and a data science lead at Deliveroo.
- London based
- Haven’t had a haircut in 4 months.
- Sometimes I tweet about interesting things: **@alonnir**

My Objectives for Today

- Show how useful network analysis can be
- Show how easy it is to get started

Our Plan For Today ⏱️

What we’ll do:

1. Discuss why one should learn about network analysis + a few motivating examples

2. Brief theoretical intro (the bare bones)

3. Analyse a real network with Python

4. (time permitting) Explore useful and interesting examples

Our Plan For Today ⏱️

What we won’t do:

- In the interest of time we’ll touch on just a few of the many (many!) things there’s to learn, just enough to get you started.
  (Links to excellent further resources will be provided.)

- No mathematical formulations! No matrix representations! No Greek letters!

- Dive deep into theory (results are often intuitive but the algorithms to get them are not.)

- Use the C word

What Is a Network Anyway?

And why should we care?

Network Analysis as a Not-So-Secret Weapon

- Data scientists come from diverse backgrounds.
  Network analysis is a skill that shouldn’t be reserved for just a few.

- Humans tend to think linearly or “tabularly”.
  Adding the network perspective unlocks another dimension.

- It’s pretty cool. 😎

Motivating Examples

You are a product data scientist for a peer-2-peer payments app.

- Customer service team: who’s customer service call should we answer first?
- Marketing team: who should we spend our “influencers” marketing budget on?
- Product team: who should we recruit as a beta tester? Who are our power users?
- Customer research team: who should we interview?

Many other use-cases

Networks are everywhere, and they affect our lives daily. Some examples include:

- Finding a job
- Finding a spouse
- Finding anything on Google
- Detecting bot/troll networks on Twitter
- Detecting karma-cheaters on reddit
- Deciding which banks to bail out during a financial crisis
- Personalised recommendations
- Voting in the Eurovision

So what is it?

What Wikipedia Has to Say

In one very common sense of the term, a graph is an ordered pair G = (V, E) comprising a set V of vertices, nodes or points together with a set E of edges, arcs or lines, which are 2-element subsets of V (i.e., an edge is associated with two vertices, and the association takes the form of the unordered pair of the vertices). To avoid ambiguity, this type of graph may be described precisely as undirected and simple.

Other senses of graph stem from different conceptions of the edge set. In one more general conception, E is a set together with a relation of incidence that associates with each edge two vertices. In another generalized notion, E is a multiset of unordered pairs of (not necessarily distinct) vertices. Many authors call these types of object multigraphs or pseudographs.

Source:
https://en.wikipedia.org/wiki/Graph_(discrete_mathematics)

What *German* Wikipedia Has to Say

Ein Graph (selten auch Graf[1]) ist in der Graphentheorie eine abstrakte Struktur, die eine Menge von Objekten zusammen mit den zwischen diesen Objekten bestehenden Verbindungen repräsentiert. Die mathematischen Abstraktionen der Objekte werden dabei Knoten (auch Ecken) des Graphen genannt. Die paarweisen Verbindungen zwischen Knoten heißen Kanten (manchmal auch Bögen). Die Kanten können gerichtet oder ungerichtet sein. Häufig werden Graphen anschaulich gezeichnet, indem die Knoten durch Punkte und die Kanten durch Linien dargestellt werden.[2]

Anschauliche Beispiele für Graphen sind ein Stammbaum oder das U-Bahn-Netz einer Stadt (siehe Abbildungen). Bei einem Stammbaum stellt jeder Knoten ein Familienmitglied dar und jede Kante ist eine Verbindung zwischen einem Elternteil und einem Kind. In einem U-Bahn-Netz stellt jeder Knoten eine U-Bahn-Station dar und jede Kante eine direkte Zugverbindung zwischen zwei Stationen.

Source:
https://de.wikipedia.org/wiki/Graph_(Graphentheorie)

Keep It Simple Sugar

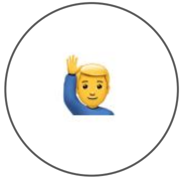
points and lines.

If you can draw a conceptual relationship between any two things, it’s likely you can model them as a network.

Points and Lines

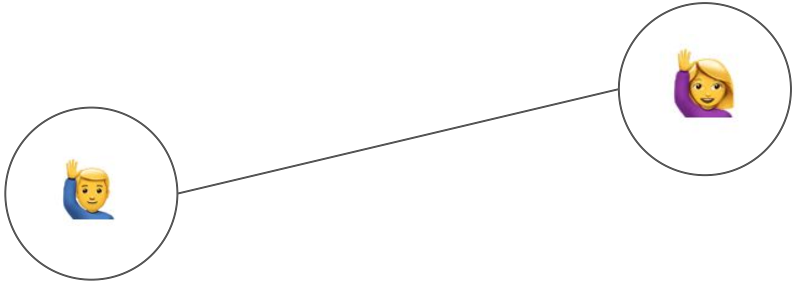
Points	Lines
People	Friendships, following
Companies	Trade
Bus stops	Buses, passengers
Bicycle sharing schemes	riders
Countries	Trade, flight routes, Eurovision points
Wikipedia articles	Internal links

Obvious Example - Facebook

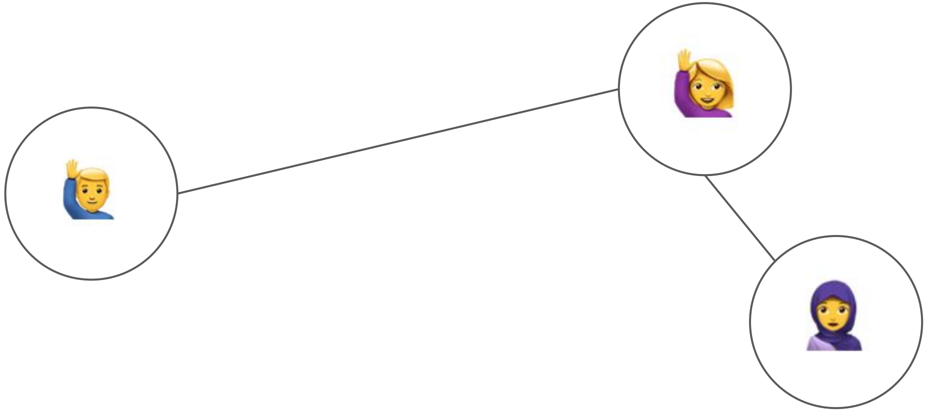


Obvious Example - Facebook

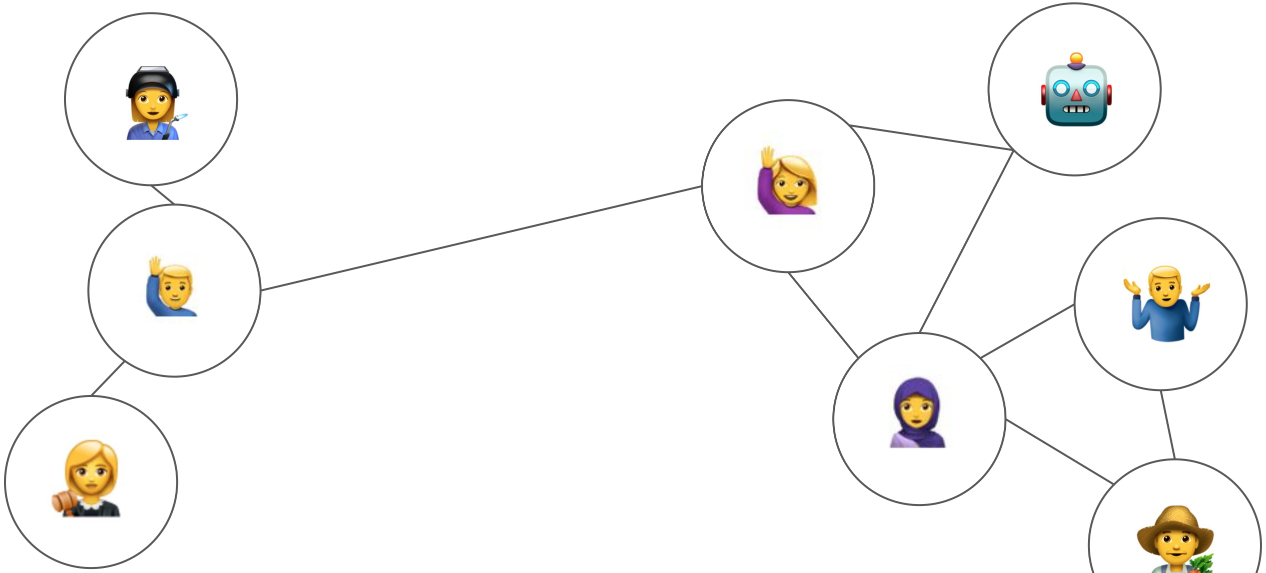
Obvious Example - Facebook



Obvious Example - Facebook



Obvious Example - Facebook



Points and Lines

Points and Lines have more formal names, which vary by discipline

Points	Lines	Discipline
Nodes	Edges, Links	Computer Science
Vertices	Edges, Arcs	Math
Sites	Bonds	Physics
Actors	Ties, Relations	Sociology

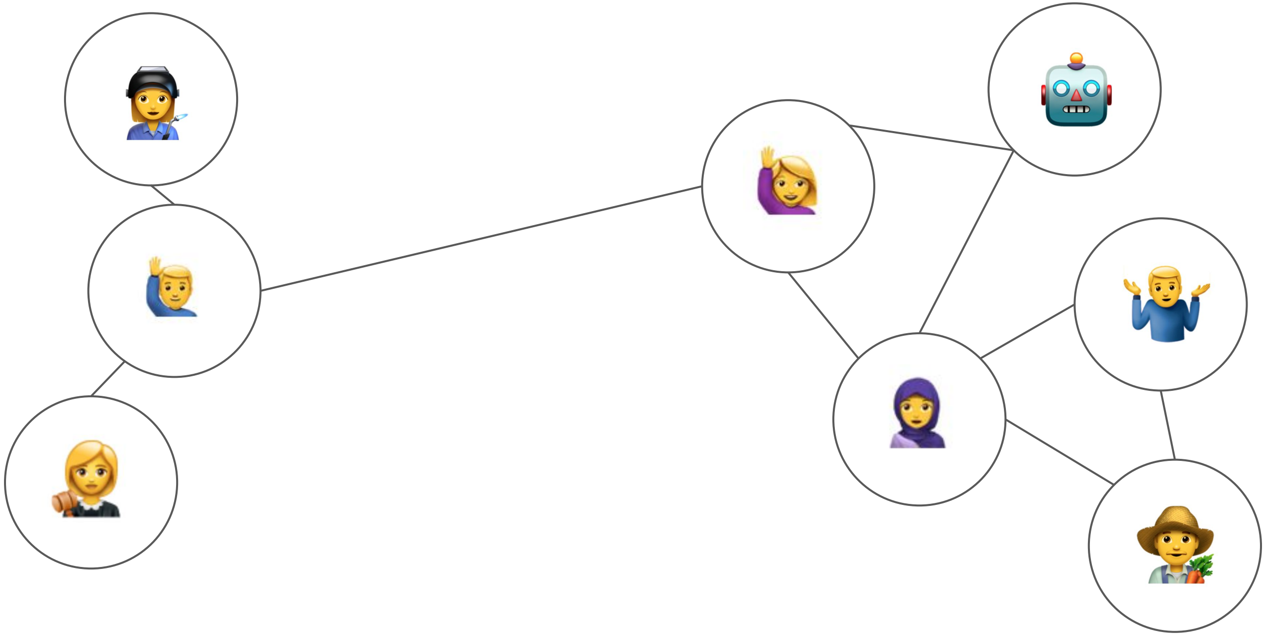
Source:
Jean Mark Gawron, Python for Social Science,
https://gawron.sdsu.edu/python_for_ss/course_core/book_draft/Social_Networks/Social_Networks.html#what-are-networks

Main Types of Networks

UnDirected	Directed
Facebook Linkedin	Twitter Snaphchat Instagram Whateverkidsusenowadays
Connections are directionless and *reciprocal* - if A is friends with B on facebook, B is also friends with A.	Connections don’t have to be reciprocal - if A follows B on Twitter, B does not necessarily follow A

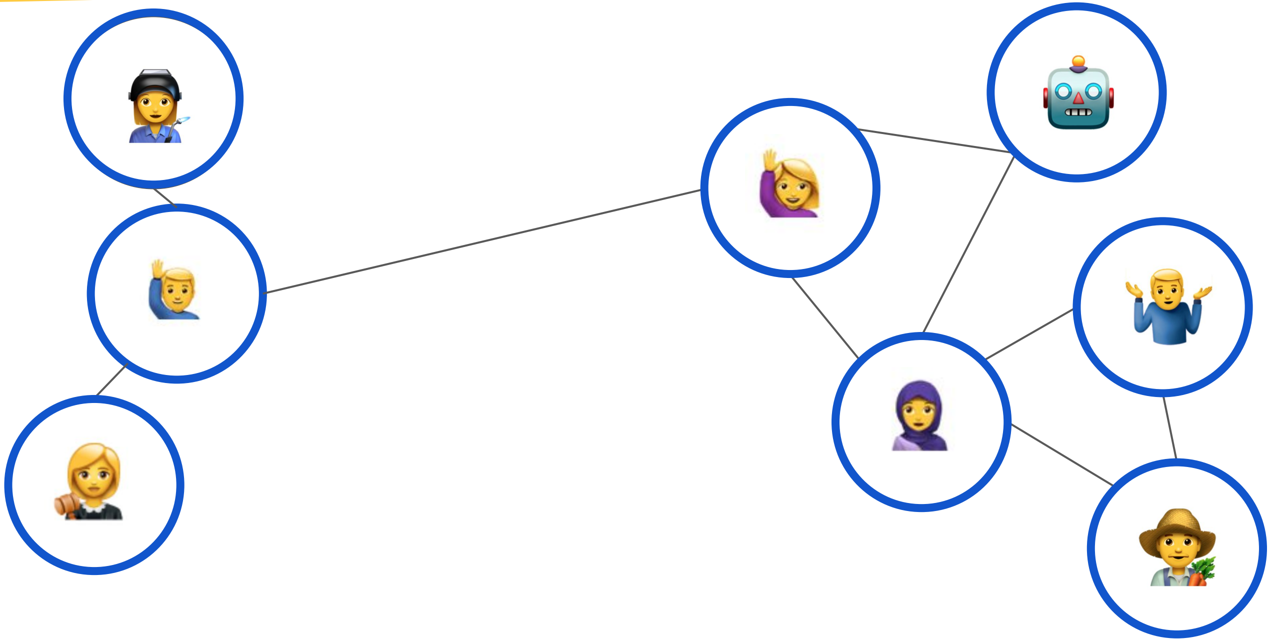
Also: MultiGraph and Multi-Directed-Graph

Back to Facebook



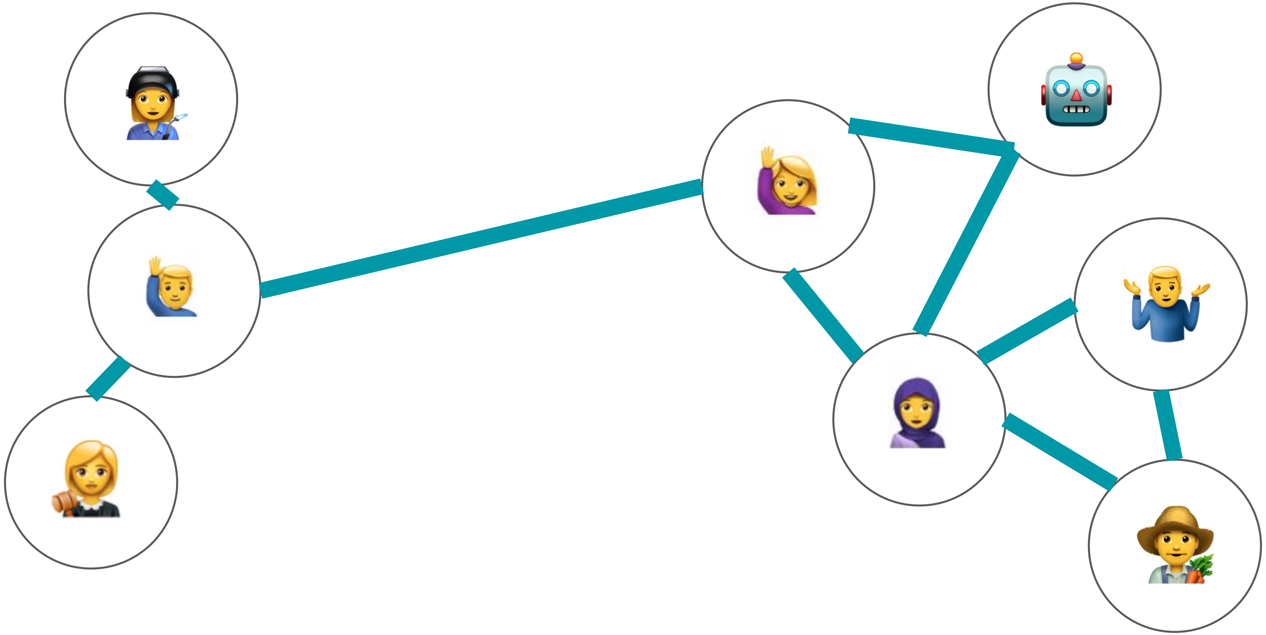
Facebook friendships → Undirected graph(s)

Back to Facebook



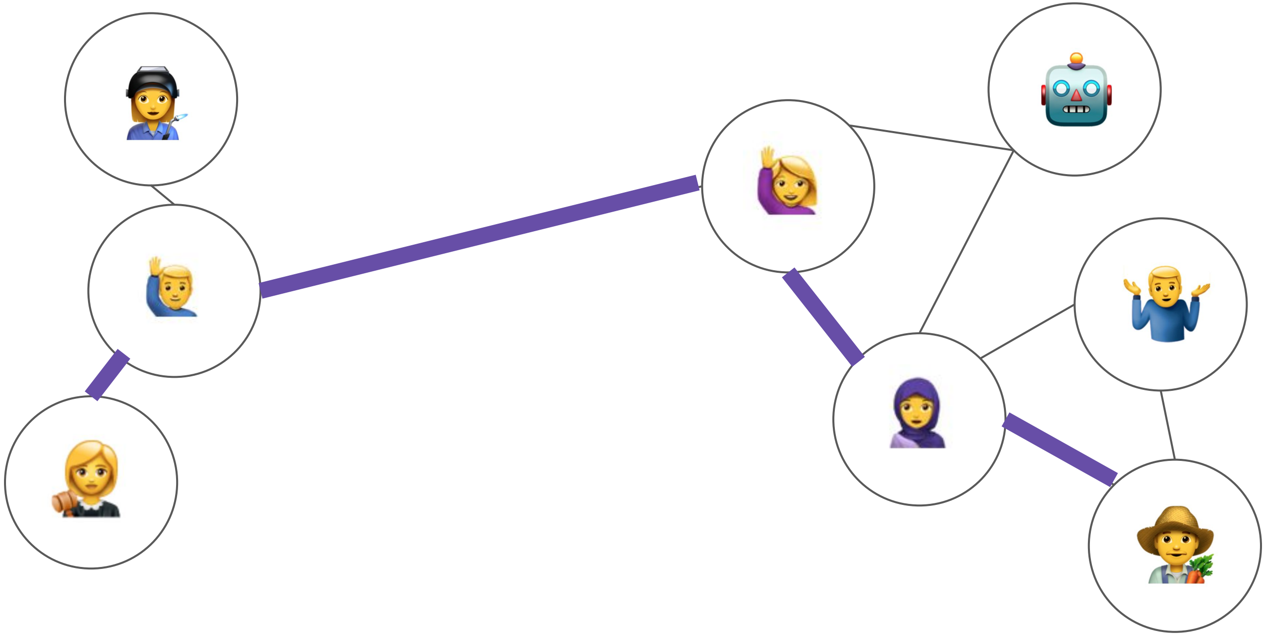
○ = nodes

Back to Facebook



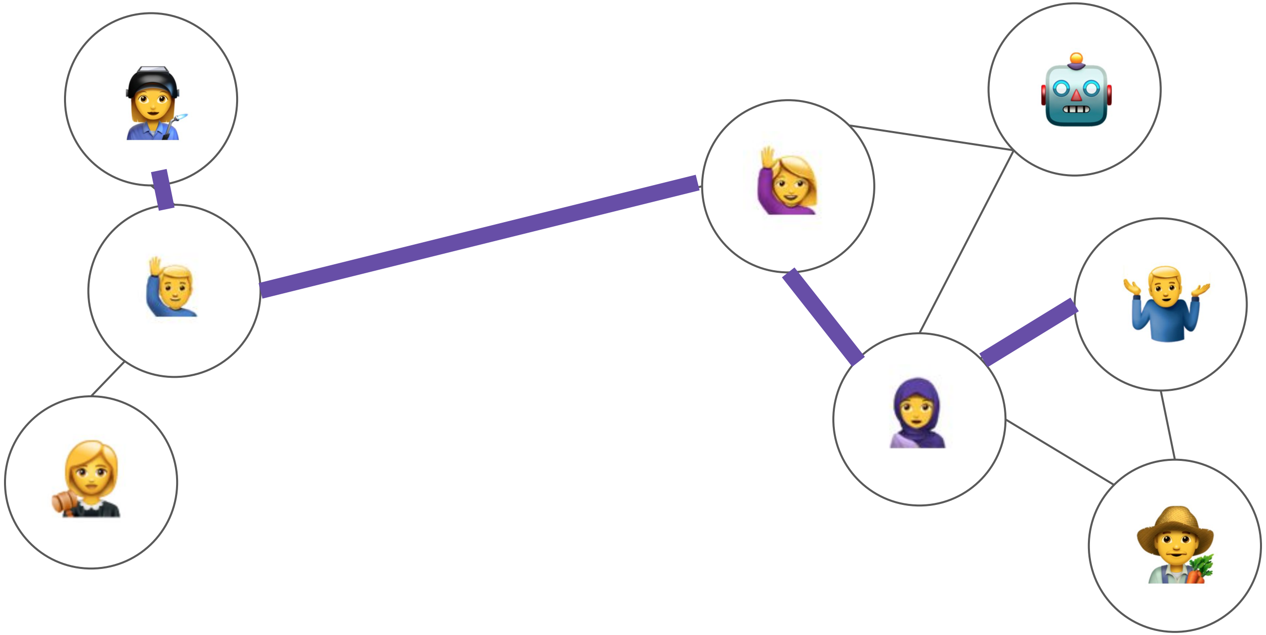
= edge

Back to Facebook



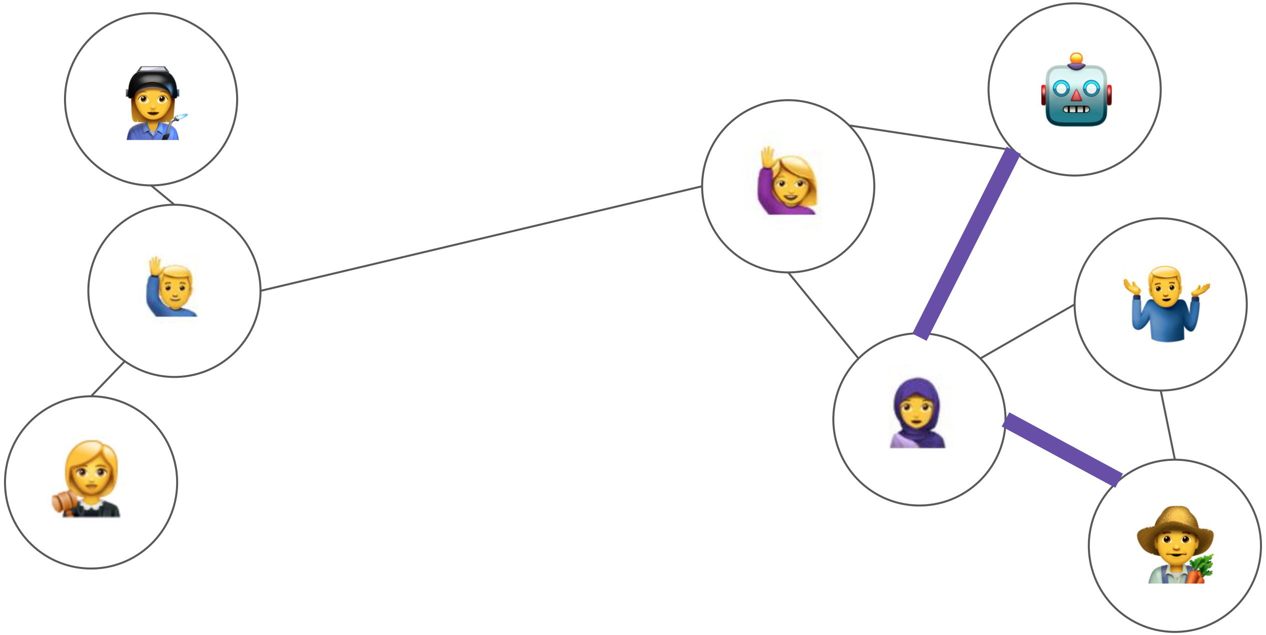
Path = edges from a node to another node, w/o repetition

Back to Facebook



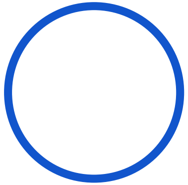
Path = edges from a node to another node, w/o repetition

Back to Facebook



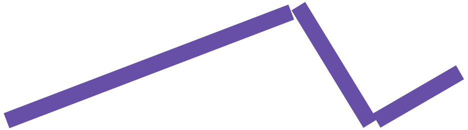
Path = edges from a node to another node, w/o repetition

Quick Recap



node

edge



path

Main Types of Networks

Multigraph

Nodes can have more than one edge between them.

e.g. messages on whatsapp

(and multi-directed-graph)

Weighted

The connection (the edge) can be quantifiable and different from edge to edge.

e.g. sending p2p payments for varying amounts.

Network Analysis in Python

The best way to learn is to get our hands dirty with real data and work out an example, so…

```
pip install networkx
```

Network Analysis in Python

Why NetworkX?

- Comprehensive (mostly)
- Friendly API
- Good documentation
- This is PyData

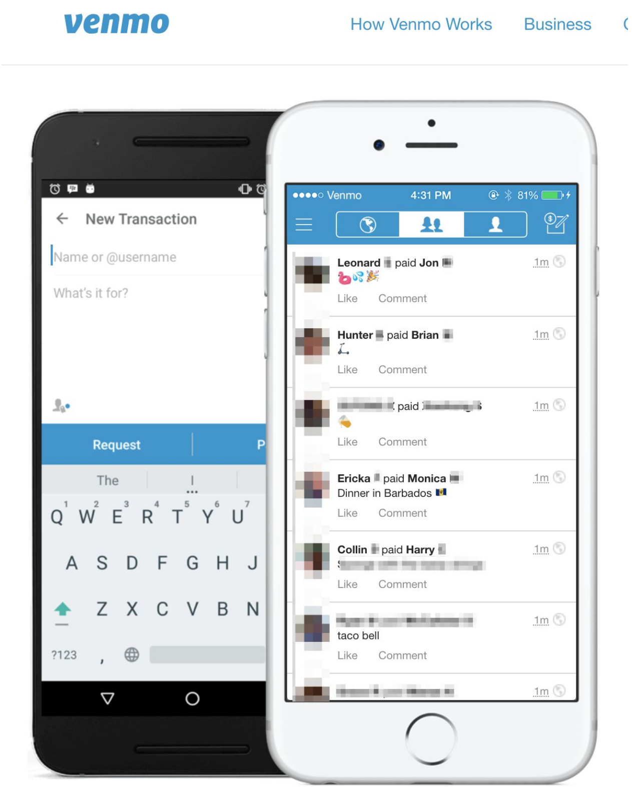
Shortcomings

- Scalability to very large graphs
- Visualization of very large graphs

(other options: snap.py, graph-tool, iGraph and Gephi)

Let’s get to work!

Example - Venmo



Leonard [redacted] paid Jon [redacted] 1m

Hunter [redacted] paid Brian [redacted] 1m

[redacted] paid [redacted] 1m

Ericka [redacted] paid Monica [redacted] 1m
Dinner in Barbados

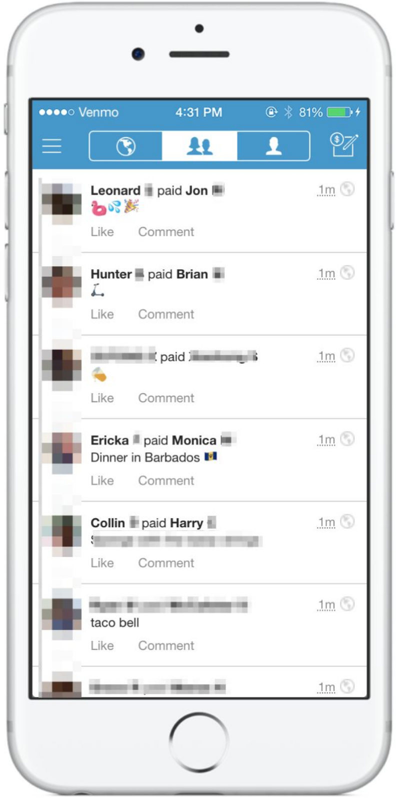
Collin [redacted] paid Harry [redacted] 1m
[redacted]

[redacted] 1m
taco bell

[redacted] 1m

Link to venmo.com

Example - Venmo



Venmo 4:31 PM 81%

Leonard [redacted] paid Jon [redacted] 1m

Like Comment

Hunter [redacted] paid Brian [redacted] 1m

Like Comment

[redacted] paid [redacted] 1m

Like Comment

Ericka [redacted] paid Monica [redacted] 1m

Dinner in Barbados

Like Comment

Collin [redacted] paid Harry [redacted] 1m

[redacted]

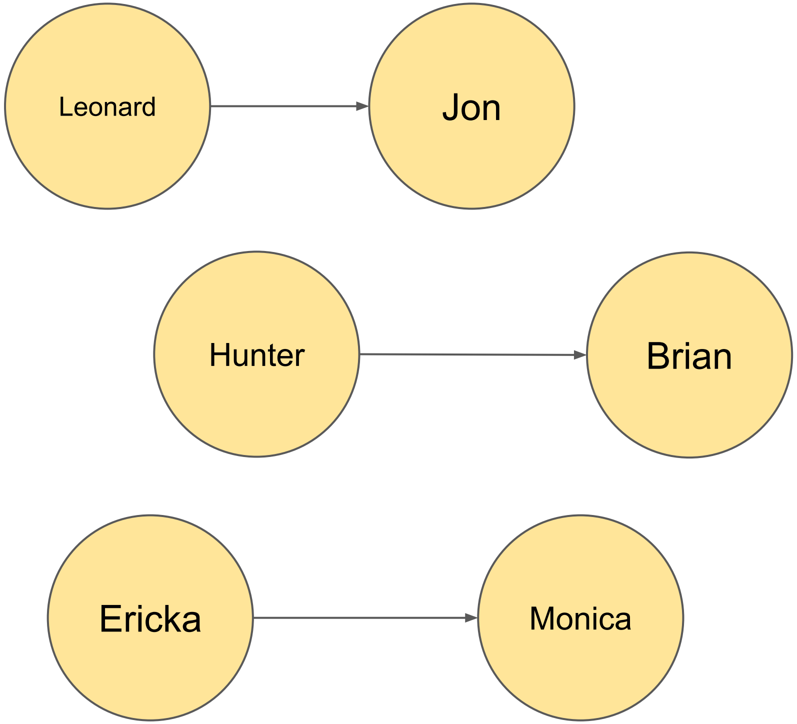
Like Comment

[redacted] 1m

taco bell

Like Comment

[redacted] 1m



Make Python Run Now

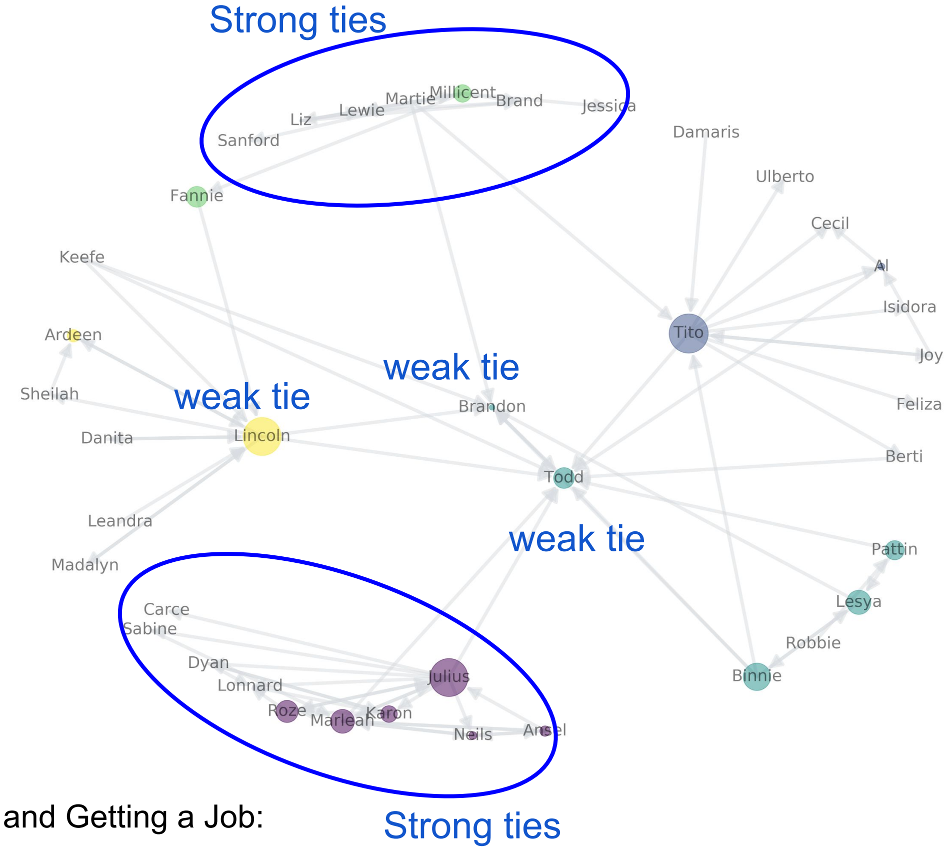
Some Examples

The Strength of Weak Ties

Seminal work by Mark Granovetter.

Cited >56K times!

- Jobs
- Spouses!



Strong ties

weak tie

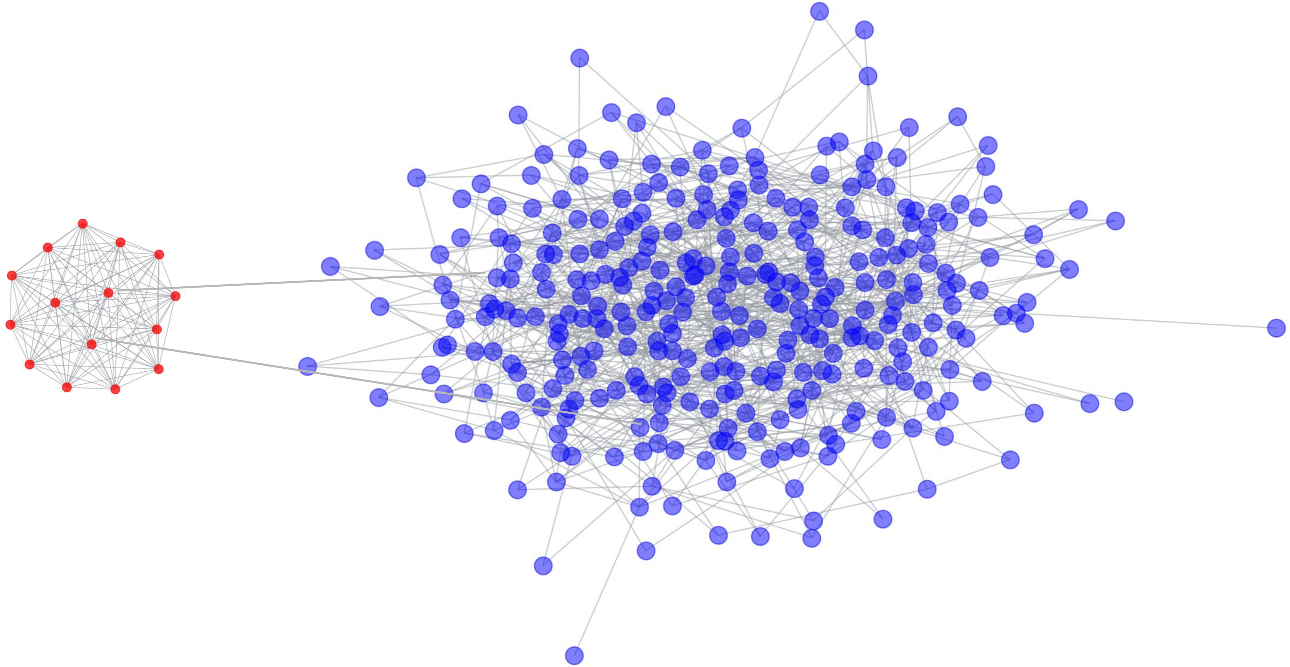
weak tie

weak tie

Strong ties

See:
Stanford Center on Poverty and Inequality: Social Networks and Getting a Job:
Mark Granovetter - link

Reddit Karma Cheaters

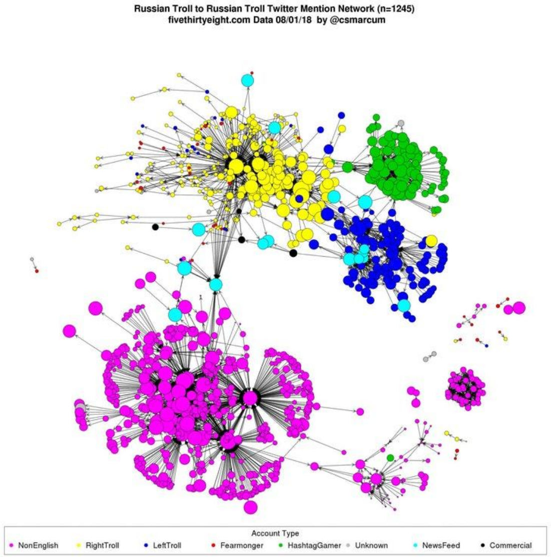


Source:

Simulated data by me.

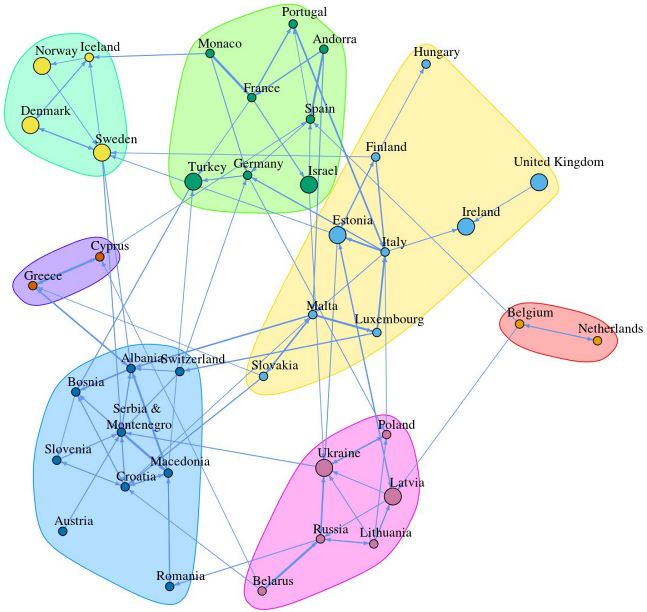
Conceptual source: Chris Slowe, Reddit's CTO, in a SFU Data Science Talk, February 3rd 2018

Twitter Bot Networks



Credit:
@csmarcum https://www.reddit.com/r/dataisbeautiful/comments/9436q2/russian_troll_2_russian_troll_twitter_mention/
FiveThirtyEight data - https://fivethirtyeight.com/features/why-were-sharing-3-million-russian-troll-tweets/

Eurovision Voting Examiner



Source:
Overthinking It - Eurovision Voting Examiner

More...

Recommendations, clustering (bi-partite graphs, which we didn’t cover)

Complex contagion

Closing Remarks

Conclusion

- We answered real world business questions using network analysis in 20 minutes.
- Network analysis is a powerful, widely applicable tool.
- We barely scratched the surface.
- NetworkX is a great place to start.
- <3 Python

Next Steps

- SNAcks is a repo that holds a highly curated list of resources on network analysis, from novice to pro.

Thank You!

Let’s *connect*:
twitter, LinkedIn