Twitter Graph - Synergic Partners

Dec. 14, 2015

Casey Huang, Claire Liu, Jordan Rosenblum, and Steve Royce

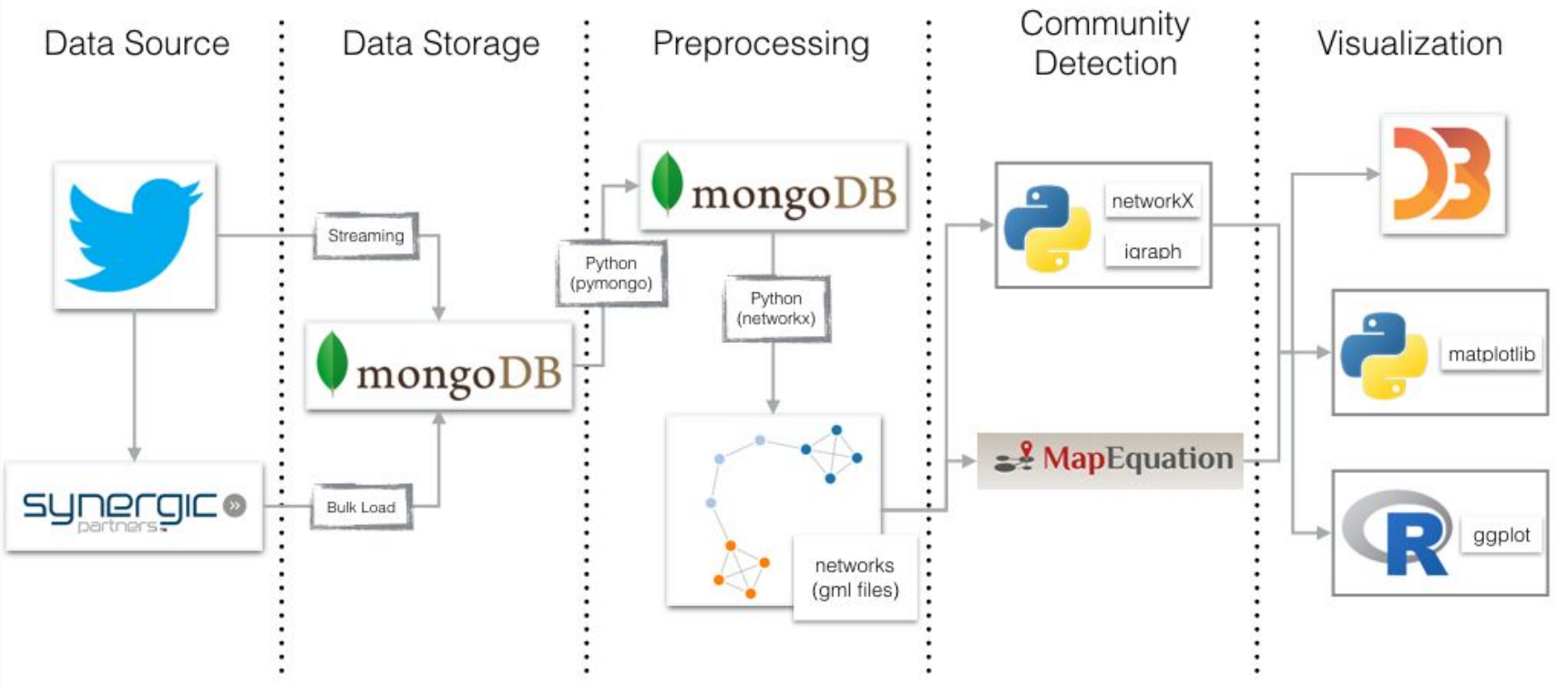
Presentation Outline

- **Project Overview**
  - Background
  - Project Flow
- Tweets to Network
- Community Detection
- Exploratory area: modeling communities as documents
- Influencers and Visualization

Background

- Original problem definition
  - Characterize and map the online conversation surrounding Columbia University on Twitter
  - 3 interaction layers: followers, mentions, retweets
- Updated problem definition
  - Characterize and map the online conversation surrounding ‘Data Science’ and ‘Big Data’ on Twitter
  - 2 interaction layers: mentions, retweets
- Goals
  - Understand the topology and structure of the **network** (e.g. visualization)
  - Investigate the **community** structure of the network
  - Find the **influencers** of the network so that they can be targeted for marketing campaigns

Project Flow



Additional tools: Gephi, python (tweepy, regex, nltk, gensim), R (lda)

Presentation Outline

- Project Overview
- **Tweets to Network**
  - Constructing the network (retweets, mentions and followers)
  - Projection (directed and weighted)
  - Data summary
- Community Detection
- Exploratory area: modeling communities as documents
- Influencers and Visualization

Constructing the network

- Mention network
  - Used mention field contained in streaming data (any user in tweet prefaced with ‘@’)
  - Edge from person mentioning to person mentioned (in direction of influence)
- Retweet network
  - Used retweet field contained in streaming data (any tweet prefaced with ‘RT’)
  - Edge from original user to user retweeting (in direction of influence)
- Follower network - not used given API and computational limitations
- The graphs for mentions and retweets were combined to form the single-layer projected network
  - Nodes: Union of the nodes in the mention and retweet networks
    - Removed nodes with 0 degree or with edges only to itself
  - Edges: If an edge is present between two nodes in at least one of the networks, it will be added to the projected network. Directionality is conserved
    - $\text{weight}_{\text{projected}} = \text{weight}_{\text{mention}} + \text{weight}_{\text{retweet}}$

Dataset

- Columbia dataset had too much noise
  - Two approaches to cleaning were unsuccessful
    - Filtering out irrelevant keywords (e.g. South Carolina)
    - Reachability to central node (e.g. '@Columbia')
  - Some form of probabilistic modeling would be required
- New dataset: data science, big data
  - 394,545 total tweets from 169,017 distinct users
  - 125,894 nodes and 169,866 edges
  - Ranging from Oct. 6 - Nov. 8, 2015
  - Doesn't require cleaning
    - Litmus test: Hashtags for 'DS'+'BD' make sense while those for Columbia don't

Top hashtags*

'Columbia' (after filtering)	**'Data Science' + 'Big Data'**
columbia	bigdata
scflood	analytics
columbiagivingday	data
elxn42	datascience
cdnpoli	iot

* Hashtag only counted once for any given user. Columbia network is after filtering out irrelevant tweets based on any node not reachable from central nodes (e.g. '@Columbia', '@DSI_Columbia', etc)

Presentation Outline

- Project Overview
- Tweets to Network
- **Community Detection**
  - Overview of algorithms
  - Comparison of Communities
- Exploratory area: modeling communities as documents
- Influencers and Visualization

Community detection algorithms

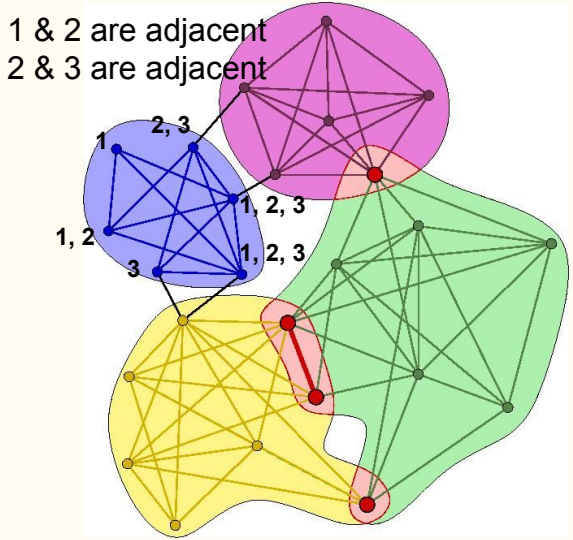
- Community in a network:
  - Groups of densely connected vertices, with sparser connections between groups
  - We want to detect groups of users who talk to each other about ‘Big Data’ and ‘Data Science’
- We explored 4 methods thus far:
  - K-clique percolation
  - Modularity using the Louvain method
  - Random Walk (3 implementations: multilevel, nonoverlapping, and overlapping)
  - Mixed Membership Stochastic Blockmodels (MMSB)

Algorithm	Package	Directed?	Weighted?	Overlapping?
K-Clique	networkX (python)	Undirected	Unweighted	Overlapping
Modularity	iGraph (python)	Directed	Weighted	Non-overlapping
Random Walk	Map Equation	Directed	Weighted	Both
MMSB (ongoing)	lda (R)	Undirected	Weighted	Overlapping

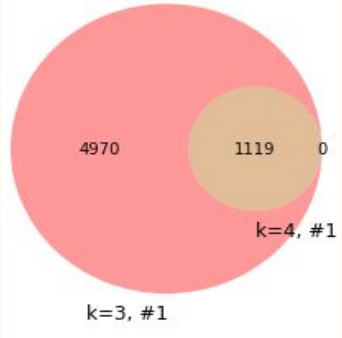
Community detection: K-clique

- How it works
  - Finds all cliques of size k (i.e. fully connected sets of k nodes).
  - A community is formed by combining all adjacent k-cliques. Two k-cliques are defined as adjacent if they share k-1 nodes.
- Undirected and unweighted network but allows for overlapping of communities
- For our network, we chose k=3 given the sparsity

Illustration of k-clique community with k=4*



Largest k-clique community of k=3 vs. k=4



*https://en.wikipedia.org/wiki/Clique_percolation_method

Community detection: K-clique

- K-clique method captures 1 larger community and numerous smaller, more closely-knit communities.
- One example consists of twitter users in specific region of the Netherlands discussing big data technology.

Characteristics of 2nd largest community (with 43 nodes) from K-clique

Location (user description)	count	Top words (user description)	count	Top hashtags (from tweets)	count
Amsterdam	5	van	9	#bigdataBK	112
Noord-Brabant	4	noord-brabant	8	#CloudComputing	41
Tilburg	3	brabant	8	#Hadoop	27
Europe	2	provincie	6	#BigData	26
Eindhoven	2	data	5	#NoSQL	21

Community detection: Modularity

- Modularity is a metric for quantifying the goodness of a community assignment based on internal (within-community) edge density
- Intuitively, maximizing high internal edge density will yield communities in our network
- However, global maximization of modularity is known to be NP-hard [Brandes et al., 2006]
- The Louvain method:
  - Is a heuristic algorithm that greedily optimizes modularity
  - It is one of the more popular algorithms for community detection in large networks

Brandes et al., 2006: “Maximizing Modularity is Hard”, arXiv:physics/0608255

Community detection: Modularity

- The user characteristics of one interesting community detected by Modularity are shown below:

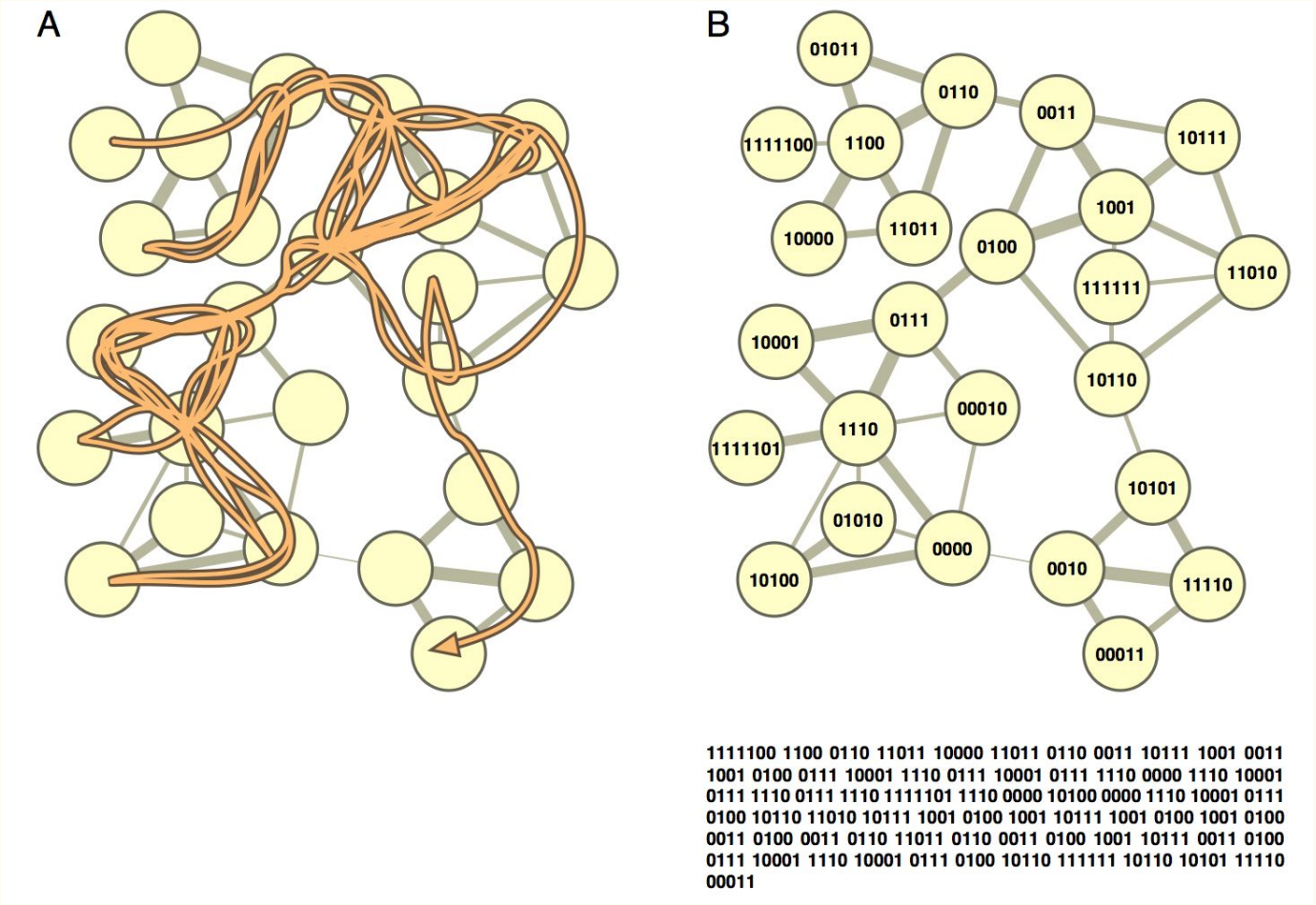
Location (user description)	count	Top words (user description)	count	Top hashtags (from tweets)	count
Madrid	347	marketing	356	#BigData	1173
Barcelona	217	digital	209	#BDS15	361
España	121	tecnolog	178	#coebigdata	216
México	51	data	165	#Marketing	114
Spain	41	comunicaci	159	#OtraformadeverTV	111

- It captures a large group of individuals (3,613 nodes) in Spanish speaking locations interested in big data technology, marketing and communication

Community detection: Random walk

- Random walker:
  - a points that traverses a graph with the probability of picking an edge proportional to the edge weights, and a small probability to jump to any other random node (as “random surfer” in PageRank)
- Recall: Huffman Encoding algorithm
  - Based on the probability distribution of a character occurring in a string, try to find an encoding that minimizes the length of the string.
  - In this case, a path representation.
- How do random walk and encoding theory relate to community detection?

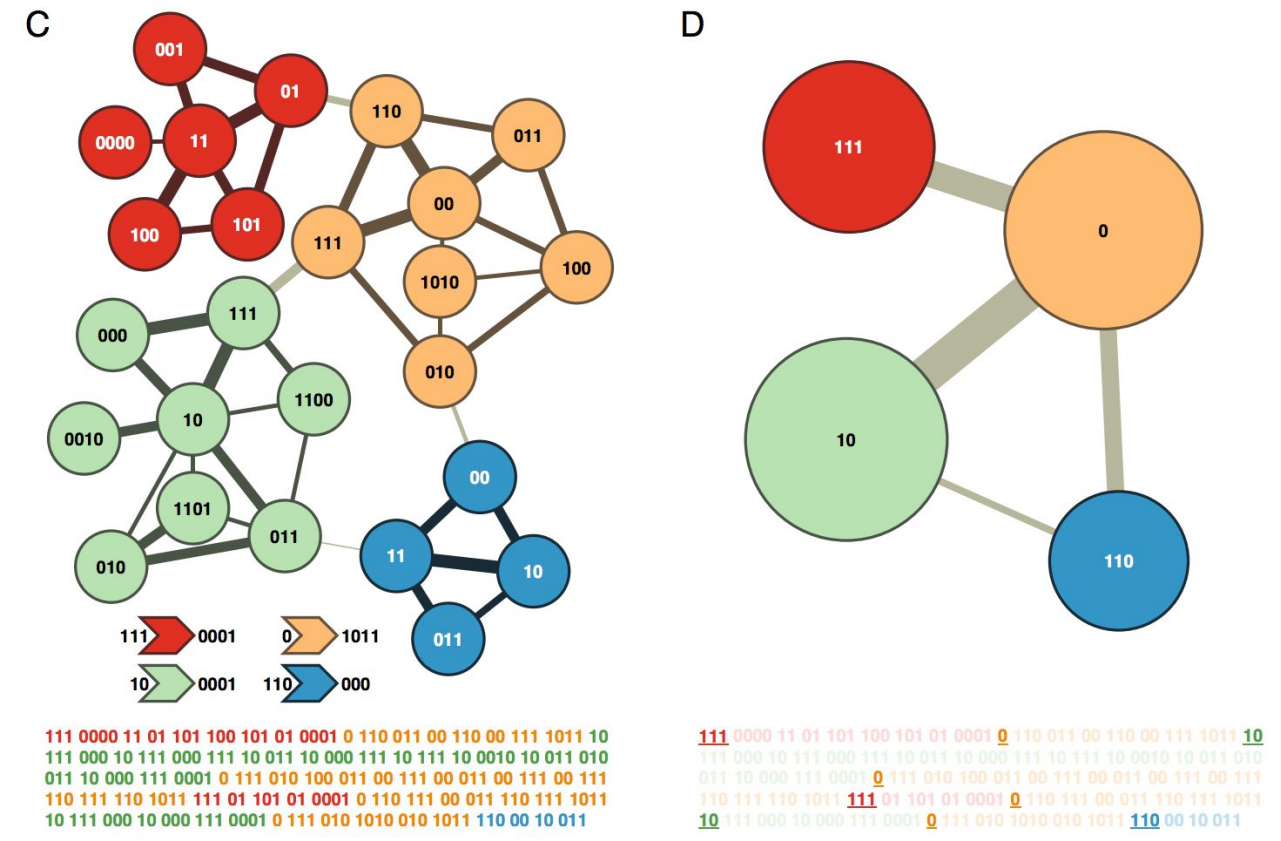
Simple description of random walk



Two layers of description of random walk

An optimal encoding would be able to:

- obtain shortest bits;
- separate the important graph structures from the inner-community details. i. e. community detection (animation)



Community Detection: Mixed Membership Stochastic Blockmodel (In process)

- Why we use it?
  - Bayesian probabilistic model for discovering overlapping communities
  - Detect influencers who bridge different communities
- How it works?
  - Assumes each user is generated from a membership over communities, the probability of two users would be connected is high if they share similar membership vectors
- Where are we now?
  - Community membership for each user.
  - Bridgeness
    - How strong a user bridges different communities
    - Function of distance between membership vector and reference vector
    - Potential improvement
  - In process: influencers within top communities, understand conversation between communities.

Top influencers bridge communities	
Users	**Bridgeness**
BigDataTweetBot	0.9965
recuweb	0.9764
Ronald_vanLoon	0.9484
KirkDBorne	0.9399
analyticbridge	0.9058
innova_scape	0.8980
ClearGrip	0.8975
7wdata	0.8818
EvanSinar	0.6508
BernardMarr	0.6431

Comparing community detection algorithms

We used the following measures to understand how communities are formed by the different algorithms:

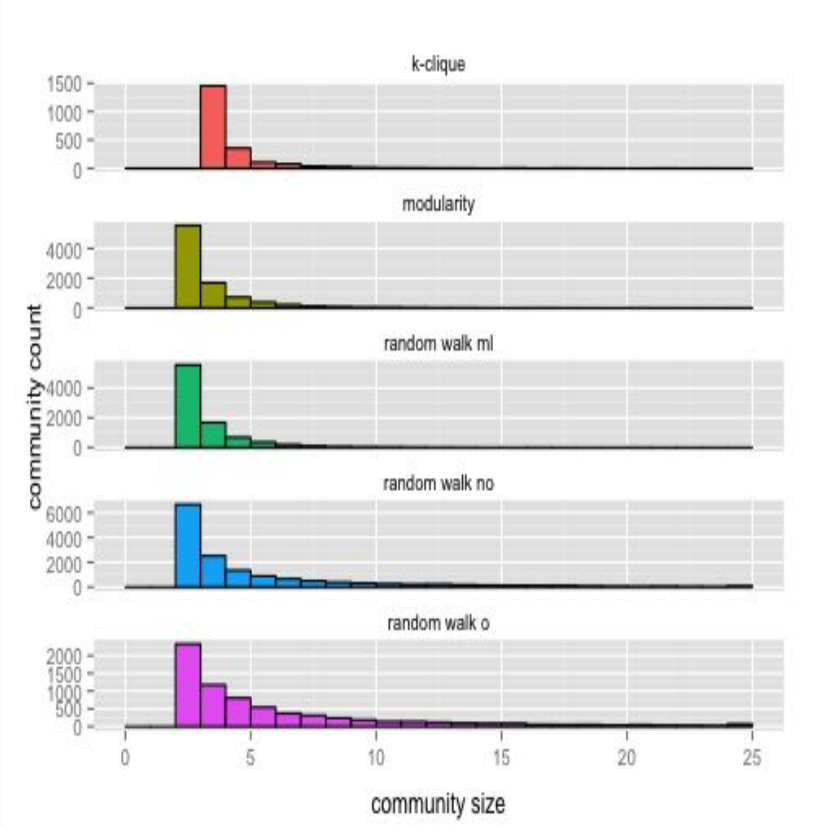
1. The size distributions of communities:
    - Do these algorithms produce communities of equal size or are they skewed?
2. Percent of outgoing edges (i.e. to another community):
    - Measures interactions across different communities
3. The overlap of communities between different methods:
    - Do these algorithms find similar communities?
4. Number of communities each node belongs to (overlapping methods only):
    - Who serves as a bridge between communities?

Comparison: Community size distribution

- Community size distribution shown in Figure
- Looking at the data from a different perspective:
  - What percent of users belong to the top N communities?
- Communities formed by modularity and random walk ml are top-heavy: the top communities in each accounts for the majority of nodes in the network

Algorithm	# Communities	75% of users	90% of users
K-clique	2,135	696 (32%)	1,556 (72%)
modularity	9,133	87 (1%)	3,085 (33%)
random walk ml	9,124	196 (2%)	3,077 (33%)
random walk no	15,872	4,327 (27%)	9,581 (60%)
random walk o	7,574	1,797 (24%)	4,159 (55%)

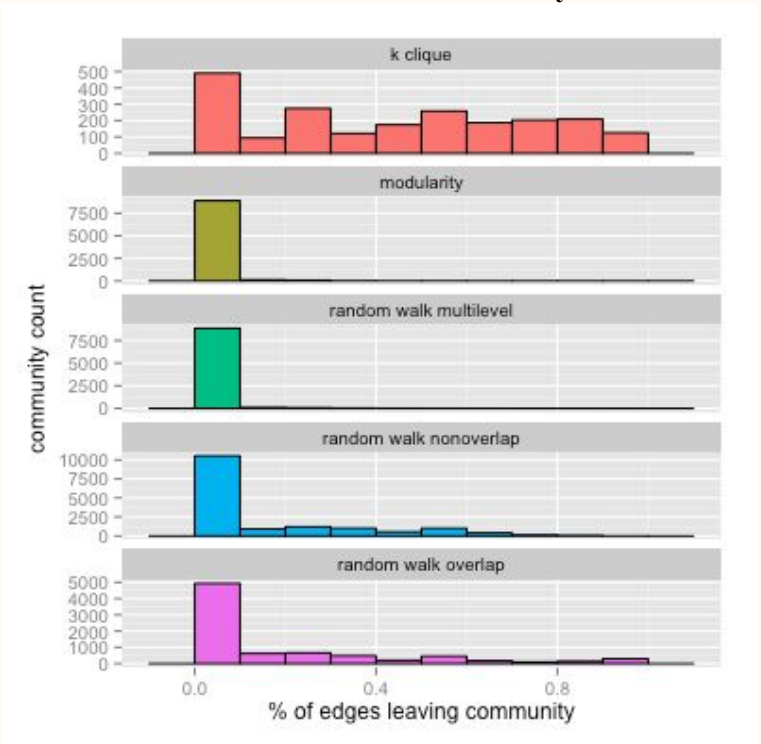
*total users (for each algorithm) is defined as the total number of distinct users across all communities



Comparison: Community interconnectedness

- Counted the number of edges leaving each community as % of total edges in community
  - By definition, we want this percentage to be lower
- K-clique method has more communities with a high percentage of outgoing edges
  - Nodes can be connected to many others outside of community but not form an adjacent clique
  - Intuitively: cliques are hard to form so nodes will talk to other people outside of their cliques as well
- K-clique does not work well for our sparse graph

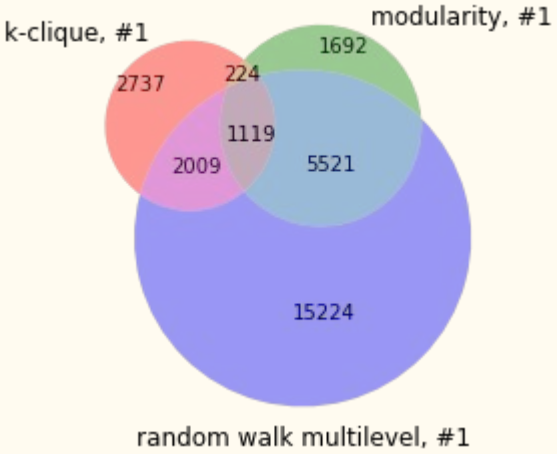
Histogram of % of outgoing edges from each community*



*Communities with zero outgoing edges are not included in the figure

Overlapping communities of different methods

Largest community overlap



Largest random walk (19%) vs. top 2 in modularity

France community forms in modularity and random walk

Number of communities each node belongs to

Overlapping Method	> 0 communities	> 1 community	> 2 communities
K-clique	12,653 (10.1%)	11,53 (0.9%)	281 (0.2%)
Random Walk Overlap	125,894 (100.0%)*	11,338 (9.0%)	4,220 (3.4%)

- Who serves as a bridge between communities?
  - In both methods, @BigDataTweetBot belongs to the most number of communities
    - 55 communities for K-clique
    - 3,122 communities for random walk overlapping
  - Others are less consistent between methods but include:
    - LinkedIn, Forbes, and YouTube

* Many communities have only 1 member

Presentation Outline

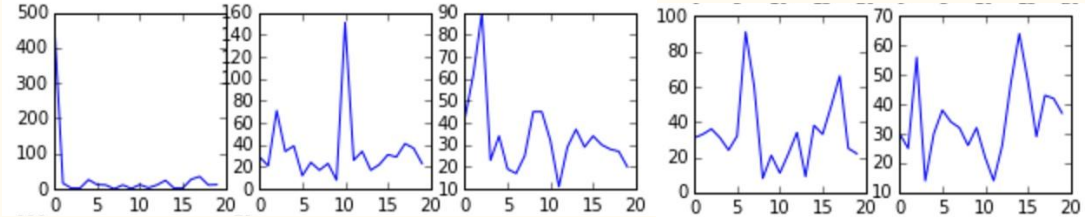
- Project Overview
- Tweets to Network
- Community Detection
- **Exploratory area: modeling communities as documents**
  - Why LDA?
  - Topic modeling results
- Influencers and Visualization

Exploratory work: LDA for tweets

- Why use LDA? Our data contains textual data and LDA can help us understand conversations in our network
- Idea: topics (conversation points) are shared across the network, but different communities use topics in different proportions
- Modeling: python's package gensim was used for training the model (20 topics and each tweet treated as a document)
- The top 4 topics from the model are shown in the table
- The topics applied to the top 5 communities (nonoverlapping random walk) are shown in the Fig. below. Topics vary between communities!
- Promising results, but not all topics were coherent enough for us to use at this point. Need larger dataset.

Topic 1	Topic 2	Topic 3	Topic 4
hadoop	jobs	gran	business
amp	hiring	empleo	intelligence
spark	san	sistema	bigdatablogs
jose	silicon	portafolioco	central
applications	jose	marcas	artificial
computing	amp	turismo	startups
wellness	services	sectur	banking
mit	machine	herramienta	seatle
generador	predictive	informaci	marketing
solutions	engineer	datos	human

Topic distributions of top 5 communities (random walk nonoverlapping)



Presentation Outline

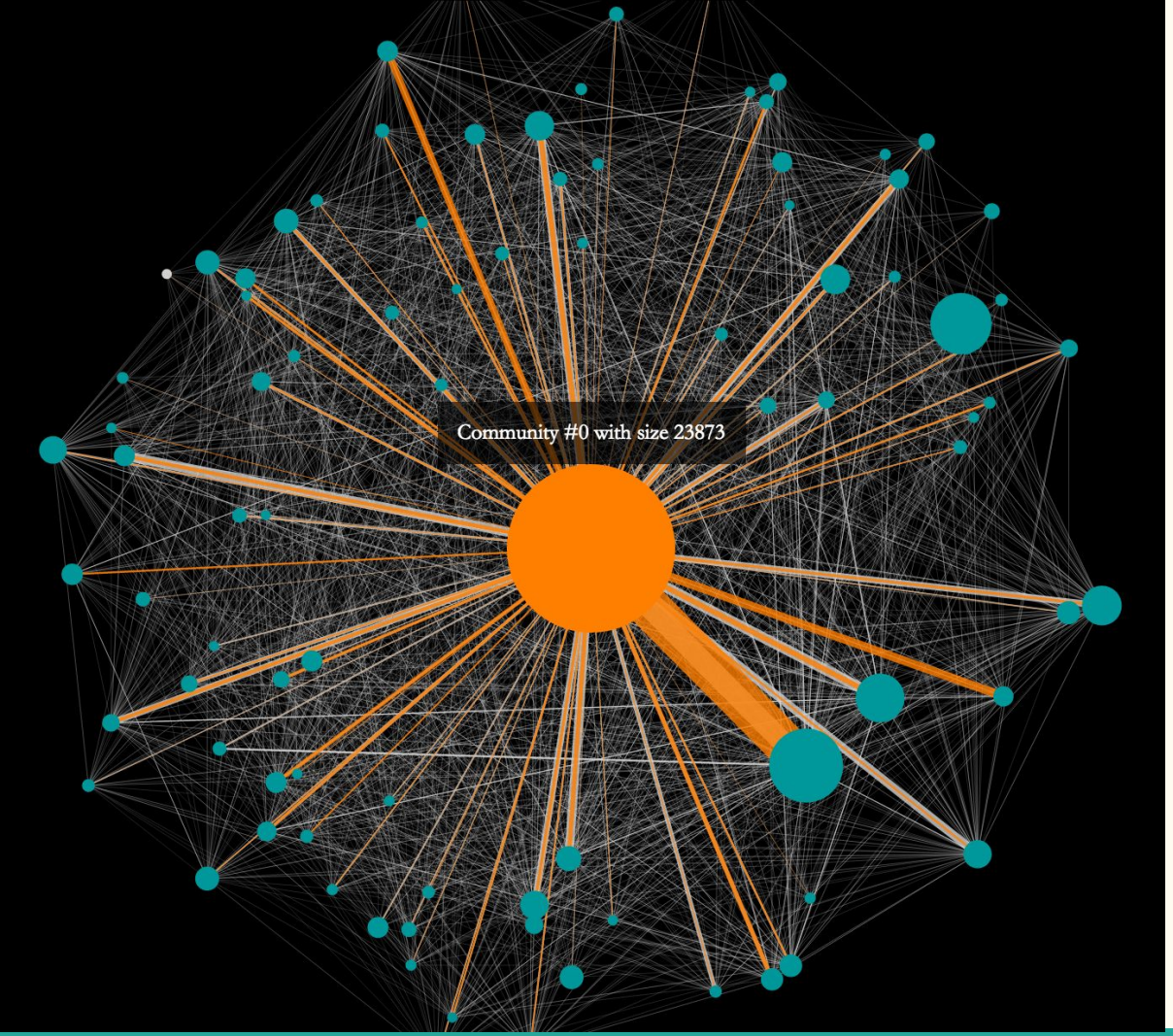
- Project Overview
- Tweets to Network
- Community Detection
- Exploratory area: modeling communities as documents
- **Influencers and Visualization**
  - Influencers for Random Walk Multilevel
  - D3 interactive visualization

Influencers: Random Walk Multilevel

Community	Degree	Betweenness	Closeness	Eigenvector	Notes
1	marcrojek[1]	Central_Colo[2]	marcrojek[1]	marcrojek[1]	[1]IoT enthusiast [2]Data centers
2	siemensindustry[1]	Remnant_Shadows[2]	giweqojyviku	RT_insurance	[1]Industrial tech [2]Programmer
3	o2mc[1]	snsharma[2]	o2mc[1]	cmphksar	[1]Big data ops [2]Applied stats
4	RonZimmernPHG	GautierLudovic[1]	GautierLudovic[1]	runitbymom	[1]Big data at Oracle France
5	jeremybowers[1]	Cassand00141441	Cassand00141441	Cassand00141441	[1]Interactive news at NYT

Degree - “How many connections does this person have?”
Betweenness - “How likely is this person to be in the shortest path between two other people in the network?”
Closeness - “How fast can this person reach anyone in the network?”
Eigenvector - “How well is this person connected to other well-connected people?“

Definitions from: High-Performance Big-Data Analytics: Computing Systems and Approaches (pg. 380)



Visualizing the communities

by Multi-level Random Walk

by Modularity

by K-clique

Conclusions and Future Work

- Data is too small. We need ideally a year worth of data.
- Conversations of top communities
  - LDA
    - Comparison of (meaningful) topic distributions for top communities.
- Community detection methods
  - Mixed Membership Stochastic Block Models
    - Probabilistic
    - Improve influencer detection (bridgeness) by integrating centrality measurement
  - CESNA by SNAP
    - Capture both community structure and node features in community detection algorithm
- Streamline data collection and processing workflow
  - Real-time data processing
  - Scalable algorithms

Acknowledgements and Questions

- Thank You!
  - Synergic Partners
    - Javier Borondo, Carmen Reina
  - Columbia Data Science Institute
    - Eleni Drinea, Sreeram Joopudi

- Questions...

Appendix

Influencers: Modularity

Community	Degree	Betweenness	Closeness	Eigenvector	Notes
1	orangebusiness	orangebusiness	orangebusiness	orangebusiness	[1]IT and communications services provider
2	infosecjerk[1]	JavierHAres	infosecjerk[1]	infosecjerk[1]	[1]Cyber security
3	FMarlidio[1]	Teligo[1]	The_Shelby_Neil	DrJessupS	[1]Spanish speakers
4	sitebymack[1]	sitebymack[1]	craigwilson	excontinent	[1]Web development and consulting
5	EmSixTeen	umbrant[1]	umbrant[1]	chris__nellis	[1]Software enginner at Cloudera

Degree - “How many connections does this person have?”
Betweenness - “How likely is this person to be in the shortest path between two other people in the network?”
Closeness - “How fast can this person reach anyone in the network?”
Eigenvector - “How well is this person connected to other well-connected people?“

Definitions from: High-Performance Big-Data Analytics: Computing Systems and Approaches (pg. 380)

Influencers: Random Walk Nonoverlap

Community	Degree	Betweenness	Closeness	Eigenvector	Notes
1	MichelleOckers	ClementineWong	MichelleOckers	ClementineWong	Both from Sydney, AU
2	claudiusvr	claudiusvr	claudiusvr	claudiusvr	Big data and analytics technology advisor
3	Flavia_Flavv	Flavia_Flavv	Flavia_Flavv	MaddieWeirrrrrr	N/A
4	2MikeGarr[1]	MartaVilaRigat	2MikeGarr[1]	2MikeGarr[1]	[1]Digital media in natural sciences
5	readflowerchild	readflowerchild	readflowerchild	readflowerchild	Scientific healing

Degree - “How many connections does this person have?”
Betweenness - “How likely is this person to be in the shortest path between two other people in the network?”
Closeness - “How fast can this person reach anyone in the network?”
Eigenvector - “How well is this person connected to other well-connected people?“

Definitions from: High-Performance Big-Data Analytics: Computing Systems and Approaches (pg. 380)

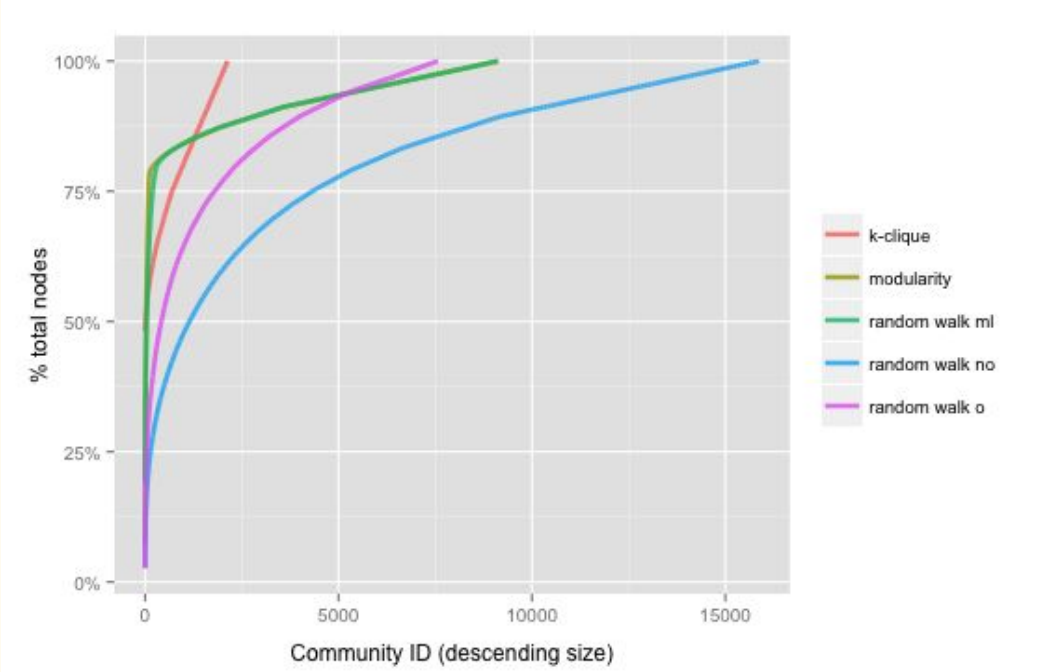
Influencers: Random Walk Overlap

Community	Degree	Betweenness	Closeness	Eigenvector	Notes
1	moha_doha[1]	moha_doha[1]	moha_doha[1]	multimedia_2016	[1]Author of novels and nonfiction
2	MediaMetricsGal	MediaMetricsGal	MediaMetricsGal	MediaMetricsGal	Analytics leader at Visa
3	multitaction	multitaction	multitaction	multitaction	Manufacturer of multitouch tech.
4	ofereliassaf	ofereliassaf	ofereliassaf	ofereliassaf	Software and Bitcoin enthusiast
5	NETSERPENTS	NETSERPENTS	NETSERPENTS	NETSERPENTS	Software dev. company

Degree - “How many connections does this person have?”
Betweenness - “How likely is this person to be in the shortest path between two other people in the network?”
Closeness - “How fast can this person reach anyone in the network?”
Eigenvector - “How well is this person connected to other well-connected people?“

Definitions from: High-Performance Big-Data Analytics: Computing Systems and Approaches (pg. 380)

Cumulative Community Size as % of Total Network*



We removed communities of size 1 (many in random walk overlapping method)