Visualizing Twitter Data Using Time-Varying Graphs

John Ensley
Mika Sumida

Mentor:
Dr. James Abello
Why Twitter?

• Twitter can tell us how information is being spread
  • Travel patterns
  • Flu outbreaks
  • Emergency response to disasters
  • Marketing
  • Current events
  • Public opinion
Our project

Approach has been 2-fold:

1. **Word co-occurrence graph**
   Goal: identify topics of conversation

2. **User-follower graph**
   Goal: identify influential/well-connected users
Twitter Datasets

• Hurricane Sandy and Irene
  o Irene: 3 million tweets over 2 weeks
  o Sandy: 7 million tweets over 2.5 weeks

• 15k user network

• live Twitter stream
Tweet text:

"Shocking news: George Zimmerman is acquitted in Trayvon Martin killing"

"shocking news george zimmerman acquitted trayvon martin martin killing"
shocking news george zimmerman acquitted

trayvon martin killing
shocking news george zimmerman acquitted

trayvon martin killing
Word Co-occurrence Graph
colors : recency
sizes : rate
Word Co-occurrence Graph

• Screen has capacity (say 2,000 nodes)

• Assign a screen time to each word
  o If the word does not occur in a tweet within that time frame, we remove it from the screen
  o otherwise, update the word's screen time

• How do we determine the amount of screen time to give a word?
  o use the rate of the system = total # of edges seen
    time
Persistence Diagram

- a second visualization
  GOAL: identify persistent nodes

- each node that passes through the co-occurrence graph has
  
  \[ x = \text{time it appeared} \]
  \[ y = \text{time it disappeared} \]

  plot the nodes in the persistence diagram at position \((x, y)\)
Followers Graph

- Also considered user/follower network
**k-Cores of a Graph**

\[ G = (V, E) \]. A subgraph \( H=(W, E|W) \) is a \( k \)-core iff \( \forall v \in W : \deg_H(v) \geq k \)
Peeling Algorithm

Can assign each vertex the value of the highest core it belongs to in linear time (Bagatelj & Zaversnik 2003) by repeatedly removing vertices of smallest degree.
Peeling Algorithm
$k$-Core Values for Edges

- Find vertices with highest core value, $N$
- Assign all edges between these vertices a value of $N$
- Remove these edges, recalculate vertex core values
- Repeat until no edges remain

Result is a partition of edges in original graph
Edge Partitions
Future Work

Integrate the two approaches

- use peeling on tweet text data to find most central topics
References

- Thanks to Adam Feldman
- Thanks to William Rand and Jeffrey Herrmann (UMaryland) for Twitter datasets