Graphs and Trees

Graphs
Model relations among data
Nodes and edges

Trees
Graphs with hierarchical structure
- Connected graph with N-1 edges
- Nodes as parents and children

Sugiyama-style graph layout

Reverse some edges to remove cycles
Assign nodes to hierarchy layers → Longest path layering
Create dummy nodes to “fill in” missing layers
Arrange nodes within layer, minimize edge crossings
Route edges – layout splines if needed

Semantic Substrates
Shneiderman 06
PivotGraph

Hierarchical Edge Bundles

Trees with Adjacency Relations

Bundle Edges along Hierarchy

Configuring Edge Tension

Use radial tree layout for inner circle
Mirror to outside
Replace inner tree with hierarchical edge bundles
Summary

Tree Layout
Indented / Node-Link / Enclosure / Layers
How to address issues of scale?
  Filtering and Focus + Context techniques

Graph Layout
Tree layout over spanning tree
Hierarchical “Sugiyama” Layout
Optimization (Force-Directed Layout)
Attribute-Driven Layout

Announcements

Final project

Design new visualization method
  Pose problem, Implement creative solution

Deliverables
  Implementation of solution
  8-12 page paper in format of conference paper submission
  2 design discussion presentations

Schedule
  Project proposal: 3/14
  Project presentation: 4/4
  Final paper and presentation: 5/3 1:30-3pm 6th floor Soda

Grading
  Groups of up to 3 people, graded individually
  Clearly report responsibilities of each member

Text Visualization

Why visualize text?
Why Visualize Text?

Understanding: get the “gist” of a document
Grouping: cluster for overview or classification
Compare: compare document collections, or inspect evolution of collection over time
Correlate: compare patterns in text to those in other data, e.g., correlate with social network

What is text data?

Documents
- Articles, books and novels
- Computer programs
- E-mails, web pages, blogs
- Tags, comments

Collection of documents
- Messages (e-mail, blogs, tags, comments)
- Social networks (personal profiles)
- Academic collaborations (publications)

Challenge: Visualize Dissertations

You have 20 years of university Ph.D. theses:
- Text
- Year
- Dept.
- Author
- Advisor
- Committee

What questions might you want to answer?
What visualizations might help?

A Concrete Example

What would help you gauge?
Topics in document?
Relationship to other docs?
...

Tag Cloud: Word Counts

Word Tree: Word Sequences

Supporting Asynchronous Collaboration for Interactive Visualization

by

Jill S. Mikkelson

M.S., University of California, Berkeley (2001)
M.S., University of California, Berkeley (2003)

A dissertation submitted in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science in the Department of the University of California, Berkeley
PhraseNet: “A the B”

A Double Gulf of Evaluation
Many (most?) text visualizations do not represent text directly, they represent a model
   - term statistics
   - clusters
   ...

Can you interpret the visualization?
   - How well does it convey the properties of the model?

Do you trust the model?
   - How does the model enable us to reason about the text?

Lessons for Text Visualization
Show (or provide access to) source text
   - Let readers assess model
   - Let readers use visualization as index into documents

Find meaningful abstractions for grouping docs
   - Are clusters interpretable?

Where possible use text to represent text... but which terms are the most descriptive?

Topics
Text as Data
Visualizing Document Content
Evolving Documents
Visualizing Conversation
Document Collections

Text as Data
Words are (not) nominal?

High dimensional (10,000+)
More than equality tests
Words have meanings and relations
- Correlations: Hong Kong, San Francisco, Bay Area
- Order: April, February, January, June, March, May
- Membership: Tennis, Running, Swimming, Hiking, Piano
- Hierarchy, antonyms & synonyms, entities, …

Text Processing Pipeline

Tokenization: segment text into terms
Special cases? e.g., “San Francisco”, “L’ensemble”, “U.S.A.”
Remove stop words? e.g., “a”, “an”, “the”, “to”, “be”?

Stemming: one means of normalizing terms
Reduce terms to their “root”: Porter’s algorithm for English
e.g., automate(s), automatic, automation all map to automat
For visualization, want to reverse stemming for labels
- Simple solution: map from stem to the most frequent word

Result: ordered stream of terms

The Bag of Words Model

Ignore ordering relationships within the text
A document = vector of term weights
Each term corresponds to a dimension (10,000+)
Each value represents the relevance
- For example, simple term counts

Aggregate into a document x term matrix
Document vector space model

Document x Term matrix

Each document is a vector of term weights
Simplest weighting is to just count occurrences

<table>
<thead>
<tr>
<th></th>
<th>Antony and Cleopatra</th>
<th>Julius Caesar</th>
<th>The Tempest</th>
<th>Hamlet</th>
<th>Othello</th>
<th>Macbeth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antony</td>
<td>157</td>
<td>73</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Brutus</td>
<td>4</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Caesar</td>
<td>232</td>
<td>227</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Calpurnia</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Cleopatra</td>
<td>57</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>energy</td>
<td>2</td>
<td>9</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>weapon</td>
<td>2</td>
<td>9</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

WordCount (Harris 2004)


http://wordcount.org
Weaknesses of Tag Clouds

- Sub-optimal visual encoding (size vs. position)
- Inaccurate size encoding (long words are bigger)
- May not facilitate comparison (unstable layout)
- Term frequency may not be meaningful
- Does not show the structure of the text

Keyword Weighting

**Term Frequency**

\[ tf_d = \text{count}(t) \text{ in } d \]

Can take log frequency: \( \log(1 + tf_d) \)

Can normalize to show proportion: \( \frac{tf_d}{\sum t_i} \)

**TF.IDF: Term Freq by Inverse Document Freq**

\[ tf.idf_d = \log(1 + tf_d) \times \log\left(\frac{N}{df_t}\right) \]

\( df_t \) = # docs containing \( t \); \( N \) = # of docs

Visualizing Document Content
What is the common local context of a term?
WordTree (Wattenberg et al)

Recurrent themes in speech

Filter infrequent runs
Glimpses of structure
Concordances show local, repeated structure
But what about other types of patterns?
For example
  Lexical:   \texttt{<A> at <B>}
  Syntactic: \texttt{<Noun> <Verb> <Object>}

Phrase Nets \cite{vanHam}: Look for specific linking patterns in the text:
  \texttt{A and B, A at B, A of B, etc.}
Could be output of regexp or parser

Visualize extracted patterns in a node-link view
  Occurrences \rightarrow Node size
  Pattern position \rightarrow Edge direction

Node Grouping

Portrait of the Artist as a Young Man
\texttt{X and Y}

The Bible
\texttt{X begat Y}

Pride & Prejudice
\texttt{X at Y}
Lexical Parser: \textless 1 sec running time
18th & 19th Century Novels
X's Y

Evolving Documents
Visualizing Revision History

How to depict contributions over time?
Example: Wikipedia history log

Animated Traces (Ben Fry)

http://benfry.com/traces/

History Flow (Viégas et al)

Wikipedia History Flow (IBM)
Visualizing Conversation

Many dimensions to consider:
- Who (senders, receivers)
- What (the content of communication)
- When (temporal patterns)

Interesting cross-products:
- What x When → Topic “Zeitgeist”
- Who x Who → Social network
- Who x Who x What x When → Information flow

Usenet Visualization (Viégas & Smith)

Show correspondence patterns in text forums
Initiate vs. reply; size and duration of discussion
Mountain (Viégas)

Conversation by person over time (who x when)

Themail (Viégas et al)

One person over time, TF.IDF weighted terms