TEXT ANALYSIS

UNIVERSITY OF
CALGARY

RELATED ACTIVITY
DOWNLOAD THE
“Exercise - Text Analysis.zip”
NOTEBOOK & DATASET

RELATIONAL DATA MODEL

Represent data as a TABLE (*relation*)

ROW (*tuple*) represents a single record
Each record is a fixed-length tuple

COLUMN (*attribute*) represents a single *variable*
Each has a name and a data type

SCHEMA – table's set of names and data types

DATABASE - a collection of tables

Month	Treatment	Pressure
March	Control	165
March	Placebo	163
March	300 mg	166
March	450 mg	168
April	Control	162
April	Placebo	159
April	300 mg	161
April	450 mg	163
May	Control	164

Blood Pressure Study (4 treatments, 6 months)

TEXT IS DIFFERENT

COMMON

UNSTRUCTURED (MOSTLY)

HIGH-DIMENSIONAL (10 000+)

BIG!

WHY ANALYZE TEXT?

WHY ANALYZE TEXT?

UNDERSTANDING: Examine patterns in word use.
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 – for example, correlate with social network.

Google Books Ngram Viewer

Graph these comma-separated phrases: Albert Einstein,Sherlock Holmes,Frankenstein case-insensitive

between 1800 and 2000 from the corpus English with smoothing of 3. Search lots of books

(click on line/label for focus)

Bible Verse Length, Keim & Oelke, VAST '07

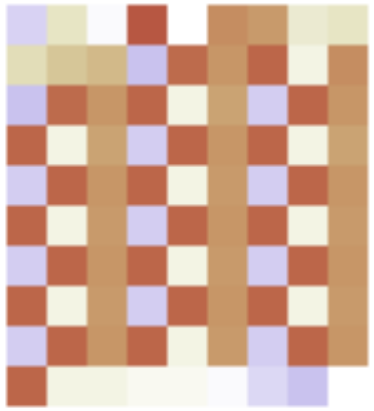
Neh 10

Those who sealed it were:
Nehemiah the governor, the son of Hakaliah.
Zedekiah, 2 Seraiah, Azariah, Jeremiah,
3 Pashhur, Amariah, Malkijah,
4 Hattush, Shebaniah, Malluk,
5 Harim, Meremoth, Obadiah,
6 Daniel, Ginnethon, Baruch,
7 Meshullam, Abijah, Mijamin,
8 Maaziah, Bilgai and Shemaiah.
These were the priests.

9 The Levites:
Jeshua son of Azaniah, Binnui of the sons of Henadad, Kadmiel,
10 and their associates: Shebaniah,
Hodiah, Kelita, Pelaiah, Hanan,
11 Mika, Rehob, Hashabiah,
12 Zakkur, Sherebiah, Shebaniah,
13 Hodiah, Bani and Beninu.

The one who brought his offering on the first day was Nahshon son of Amminadab of the tribe of Judah.

Num 7



13 His offering was one silver plate weighing a hundred and thirty shekels[a] and one silver sprinkling bowl weighing seventy shekels,[b] both according to the sanctuary shekel, each filled with the finest flour mixed with olive oil as a grain offering; 14 one gold dish weighing ten shekels,[c] filled with incense; 15 one young bull, one ram and one male lamb a year old for a burnt offering; 16 one male goat for a sin offering[d]; 17 and two oxen, five rams, five male goats and five male lambs a year old to be sacrificed as a fellowship offering. This was the offering of Nahshon son of Amminadab.

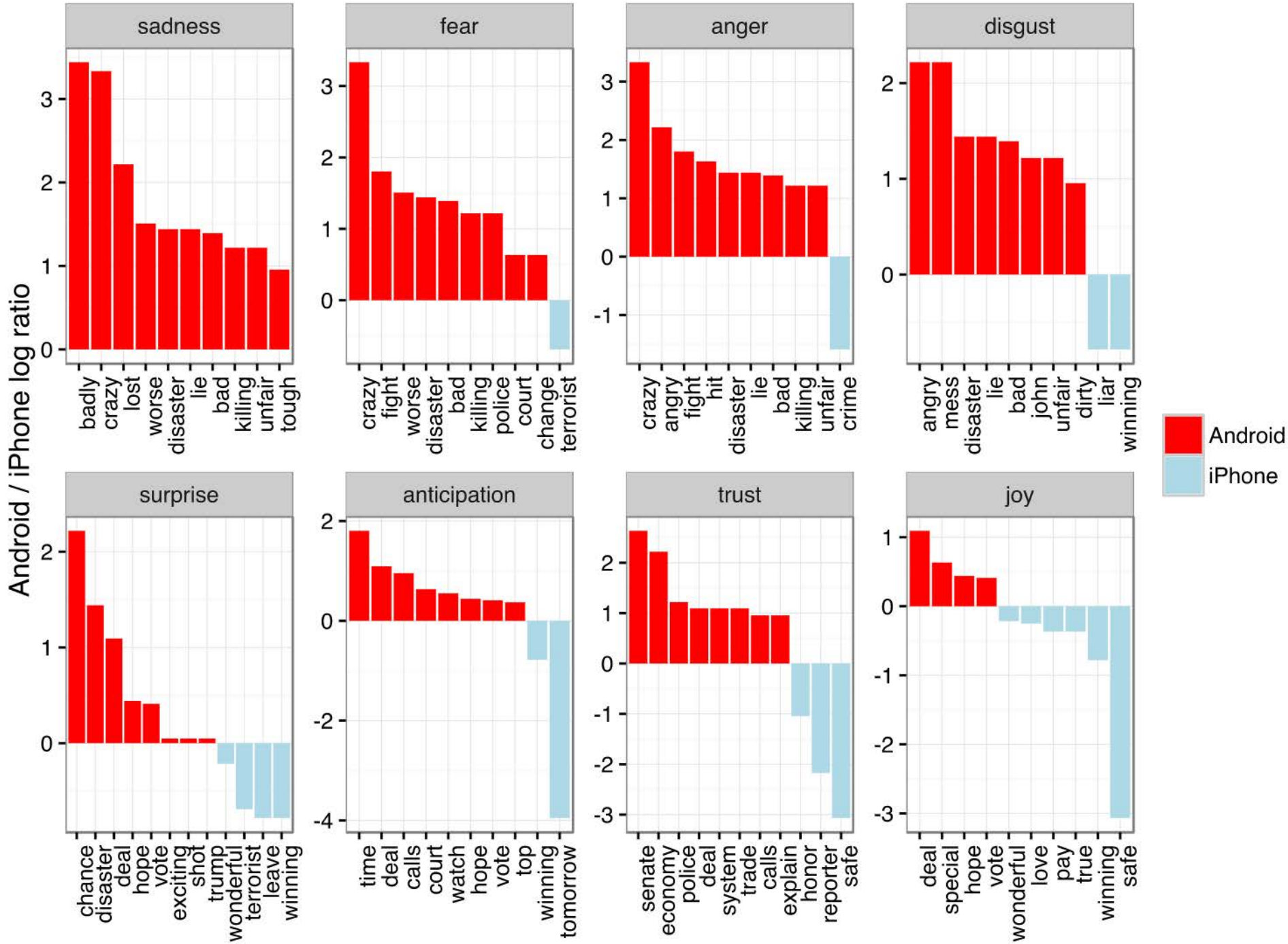
18 On the second day Nethanel son of Zuar, the leader of Issachar, brought his offering.

19 The offering he brought was one silver plate weighing a hundred and thirty shekels and one silver sprinkling bowl weighing seventy shekels, both according to the sanctuary shekel, each filled with the finest flour

Text analysis of Trump's tweets confirms he writes only the (angrier) Android half

I don't normally post about politics (I'm not particularly savvy about polling, which is where data science has had the largest impact on politics). But this weekend I saw a hypothesis about Donald Trump's twitter account that simply begged to be investigated with data:

David Robinson (2016)
http://varianceexplained.org/r/trump-tweets/


sadness
fear
anger
disgust
surprise
anticipation
trust
joy
Android / iPhone log ratio
Android
iPhone
badly crazy lost worse disaster lie bad killing unfair tough
crazy fight worse disaster bad killing police court change terrorist
crazy angry fight hit disaster lie bad killing unfair crime
angry mess disaster lie bad john unfair dirty liar winning
chance disaster deal hope vote exciting shot trump wonderful terrorist leave winning
time deal calls court watch hope vote top winning tomorrow
senate economy police deal system trade calls explain honor reporter safe
deal special hope vote wonderful love pay true winning safe

WHAT IS TEXT DATA?

DOCUMENTS

Articles, books, and novels
Computer programs
E-mails, web pages, blogs
Messages, posts, tags, comments

COLLECTION OF DOCUMENTS

Messages (e-mail, blogs, tags, comments)
Social networks (personal profiles)
Academic collaborations (publications)
Even whole libraries, websites, social networks

TEXT AS DATA

Words are the basic unit of data.

DOCUMENT-LEVEL ATTRIBUTES & METADATA

LENGTH
DATE(S)
AUTHOR(S)
FORMAT

WORD-LEVEL ATTRIBUTES

WORD LENGTH
PART OF SPEECH (noun, verb, adjective, etc.)
FORMAT (*italic*, underline, etc.)
LANGUAGE (English? Latin? Japanese?)
FREQUENCY / DIFFICULTY (is it common?)
SENTIMENT (positive or negative connotation)
SYNONYMS / ANTONYMS / ETYMOLOGY (other meanings? roots?)
ENTITIES ("Calgary", "Obama", "Telus", ...)

... AND MANY MORE

AGGREGATION

REPETITION
PLAGARISM
SHARED ENTITIES
AUTHOR STYLE

COLLECTION

DOCUMENT

SECTION

PAGE

PARAGRAPH

SENTENCE

WORD

TENSE
SENTIMENT
SENTENCE LENGTH
READING LEVEL

WHAT ABOUT THESE WORDS?

automate
automates
automatic
automation

automat

~~a, an, the, to, ...~~

“New York”
“Ban Ki-moon”
“Manchester United”

TEXT PROCESSING **PIPELINE**

TOKENIZATION: SEGMENT TEXT INTO TERMS

Entities? "San Francisco", "O'Connor", "U.S.A."

Remove stop words? "a", "an", "the", "to", "be"

N-grams? Can take words in 2-word groups (bi-grams), 3-word (tri-grams), etc.

STEMMING: GROUP TOGETHER DIFFERENT FORMS

Roots: visualization(s), visualize(s), visually visual

lemmatization: goes, went, gone go

For visualization, sometimes need to reverse stemming for labels

Simple solution: map from stem to the most frequent word

RESULT: ORDERED STREAM OF TERMS

TEXT PROCESSING **PIPELINE**

“The quick brown fox jumps over the lazy dog.”

TOKENIZE (N=1)
[The], [quick], [brown], [fox], [jumps], [over], [the], [lazy], [dog].

TOKENIZE (N=1), REMOVE STOPWORDS, STEM
[quick], [brown], [fox], [jump], [over], [lazy], [dog]

TOKENIZE (N=2)
[the quick], [quick brown], [brown fox], [fox jumps], [jumps over], [over the]

TOKENIZE (N=5)
[the quick brown fox jumps], [quick brown fox jumps over], [brown fox jumps over

PARTS-OF-SPEECH

Tag	Meaning	English Examples
ADJ	adjective	*new, good, high, special, big, local*
ADP	adposition	*on, of, at, with, by, into, under*
ADV	adverb	*really, already, still, early, now*
CONJ	conjunction	*and, or, but, if, while, although*
DET	determiner, article	*the, a, some, most, every, no, which*
NOUN	noun	*year, home, costs, time, Africa*
NUM	numeral	*twenty-four, fourth, 1991, 14:24*
PRT	particle	*at, on, out, over per, that, up, with*
PRON	pronoun	*he, their, her, its, my, I, us*
VERB	verb	*is, say, told, given, playing, would*
.	punctuation marks	*. , ; !*
X	other	*ersatz, esprit, dunno, gr8, univeristy*

NAMED ENTITY RECOGNITION

IDENTIFY AND CLASSIFY NAMED ENTITIES IN TEXT:

JOHN SMITH IS A PERSON
SOVIET UNION IS A COUNTRY
2500 UNIVERSITY DR IS AN ADDRESS
(555) 867-5309 IS A PHONE NUMBER

ENTITY RELATIONS: HOW DO THE ENTITIES RELATE?

DO THEY CO-OCCUR IN A DOCUMENT? IN A SENTENCE?

BUT THIS CAN BE DIFFICULT

KEEP UP ON *Vietnam* YOUR READING *UK* WITH AUDIO BOOKS *Louisiana, USA*

Audio books *Louisiana, USA* are highly popular *S.Carolina, USA* with library *Pennsylvania, USA* patrons in the town *Mass., USA*

of *Turkey* Springfield, *Virginia, USA* Greene *Maine, USA* County, MO. *Norway* "People are mobile *Alabama, USA*

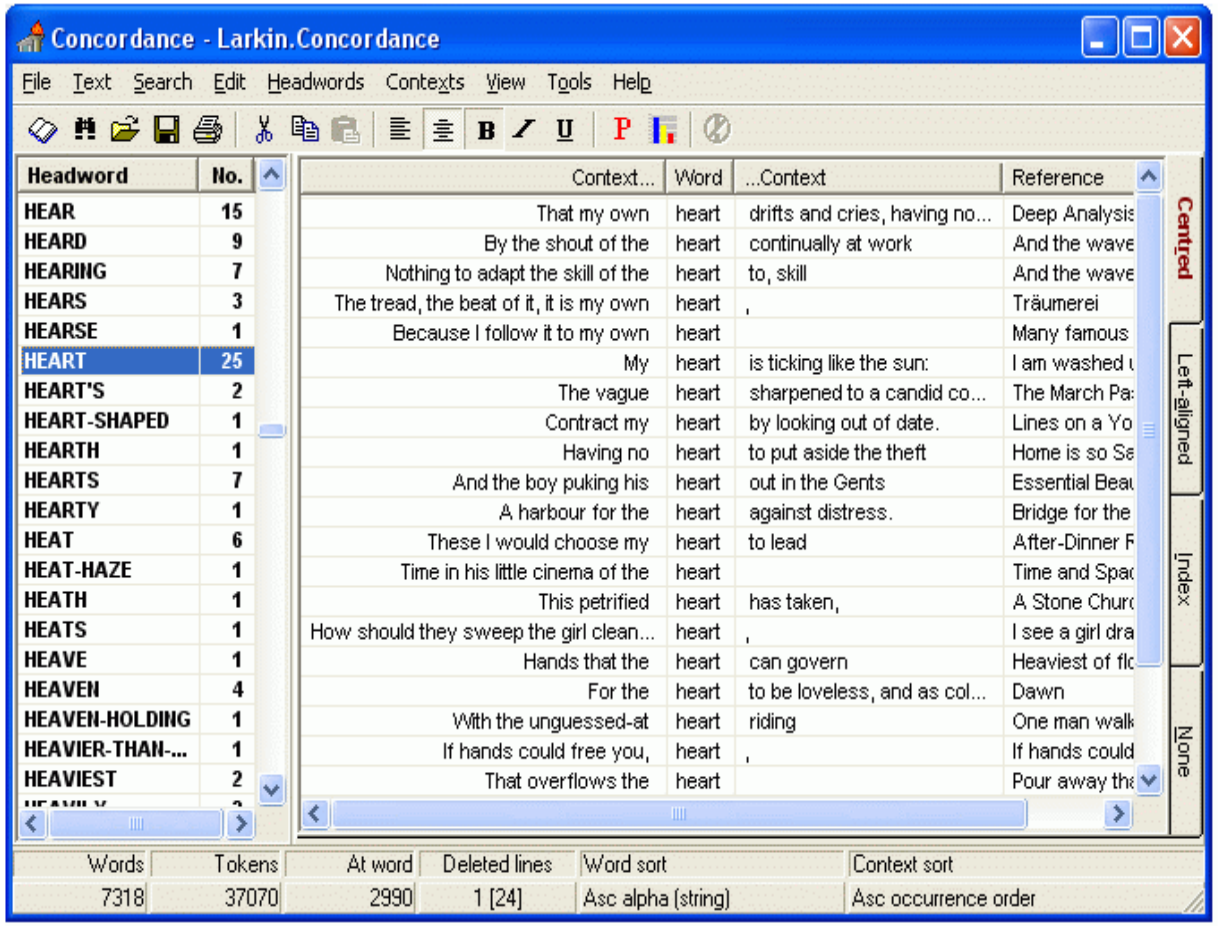
and busier, and audio books *Louisiana, USA* fit into that lifestyle" says Gary *Indiana, USA*

Sanchez, *Dominican Republic* who oversees the library's *Pennsylvania, USA* $2 million *Kentucky, USA* budget...

WORDS AND PHRASES

WORD RELATIONSHIPS

"Concordance" - Words plus the context in which they appear.



WORD TREES

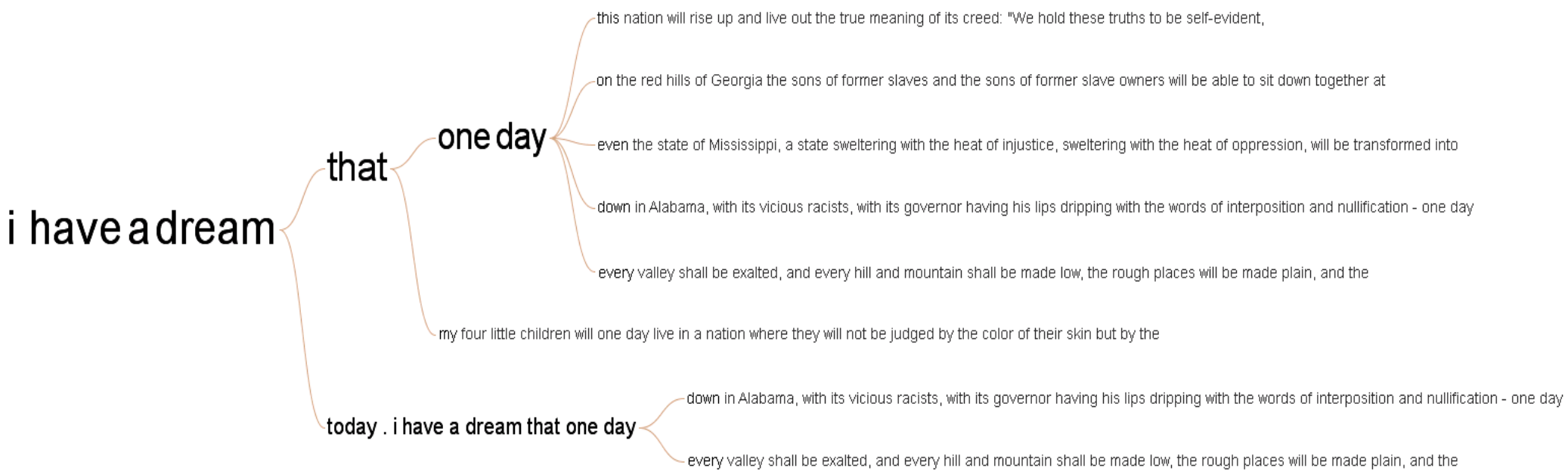
- cats are better than dogs
- cats eat kibble
- cats are better than hamsters
- cats are awesome
- cats are people too
- cats eat mice
- cats meowing
- cats in the cradle
- cats eat mice
- cats in the cradle lyrics
- cats eat kibble
- cats for adoption
- cats are family
- cats eat mice
- cats are better than kittens
- cats are evil
- cats are weird
- cats eat mice

WATTENBERG & VIÉGAS 2008

31
hits


love the
lord
thy god
with all
thine heart , and with all thy soul ,
and with all thy might .
that thou mayest live .
thy heart , and with all thy soul , and with all thy
mind
strength , an
,
and
keep his charge , and his statutes , and his judgments , and his commandments , alway .
to walk ever in his ways ; then shalt thou add three cities more for thee , beside these three : 19 :
that thou mayest obey his voice , and that thou mayest cleave unto him : for he is thy life , and the
to walk in his ways , and to keep his commandments and his statutes and his judgments , that thou mayest live
your god
,
and to
serve him with all your heart and with all your soul , 11 : 14 that i will give you the rain of your land
walk in all his ways , and to keep his commandments , and to cleave unto him , and to serve him
to walk in all his ways , and to cleave unto him ; 11 : 23 then will the lord drive out all these nations from
with all your heart and with all your soul .
.
,
all ye his saints : for the lord preserveth the faithful , and plentifully rewardeth the proud doer .
hate evil : he preserveth the souls of his saints ; he delivereth them out of the hand of the wicked .
because he hath heard my voice and my supplications .
name of the lord , to be his servants , every one that keepeth the sabbath from polluting it , and taketh hold of my covenant
good , and establish judgment in the gate : it may be that the lord god of hosts will be gracious unto the remnant of joseph
evil ; who pluck off their skin from off them , and their flesh from off their bones ; 3 : 3 who also eat the
truth and peace .
other ; or else he will hold to the one , and despise the other . ye cannot serve god and mammon .
6 : 25 therefore i say unto y
16 : 14 and the pharisees a
uppermost
rooms at feasts , and the chief seats in the synagogues , 23 : 7 and greetings in the markets , and to be called of
seats in the synagogues , and greetings in the markets .
father
; and as the father gave me commandment , even so i do .
hath bestowed upon us , that we should be called the sons of god : therefore the world knoweth us not , because it knew him
brotherhood .
world , the love of the father is not in him .
brethren .
children of god , when we love god , and keep his commandments .

RECURRENT THEMES IN SPEECH



GLIMPSES OF STRUCTURE

Concordances show local, repeated structure,
but what about other types of patterns?

FOR EXAMPLE

LEXICAL: <A> at <B>

SYNTACTIC: <Noun> <Verb> <Object>

PHRASE NETS

LOOK FOR SPECIFIC LINKING PATTERNS IN THE TEXT:

‘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 = NODE SIZE

PATTERN POSITION = EDGE DIRECTION

van Ham et al

Select a phrase

word1 and word2
word1 's word2
word1 of the word2
word1 the word2
word1 a word2
word1 at word2
word1 is word2
word1 [space] word2

or enter your own

* and * Submit

Filters

Show top: 100

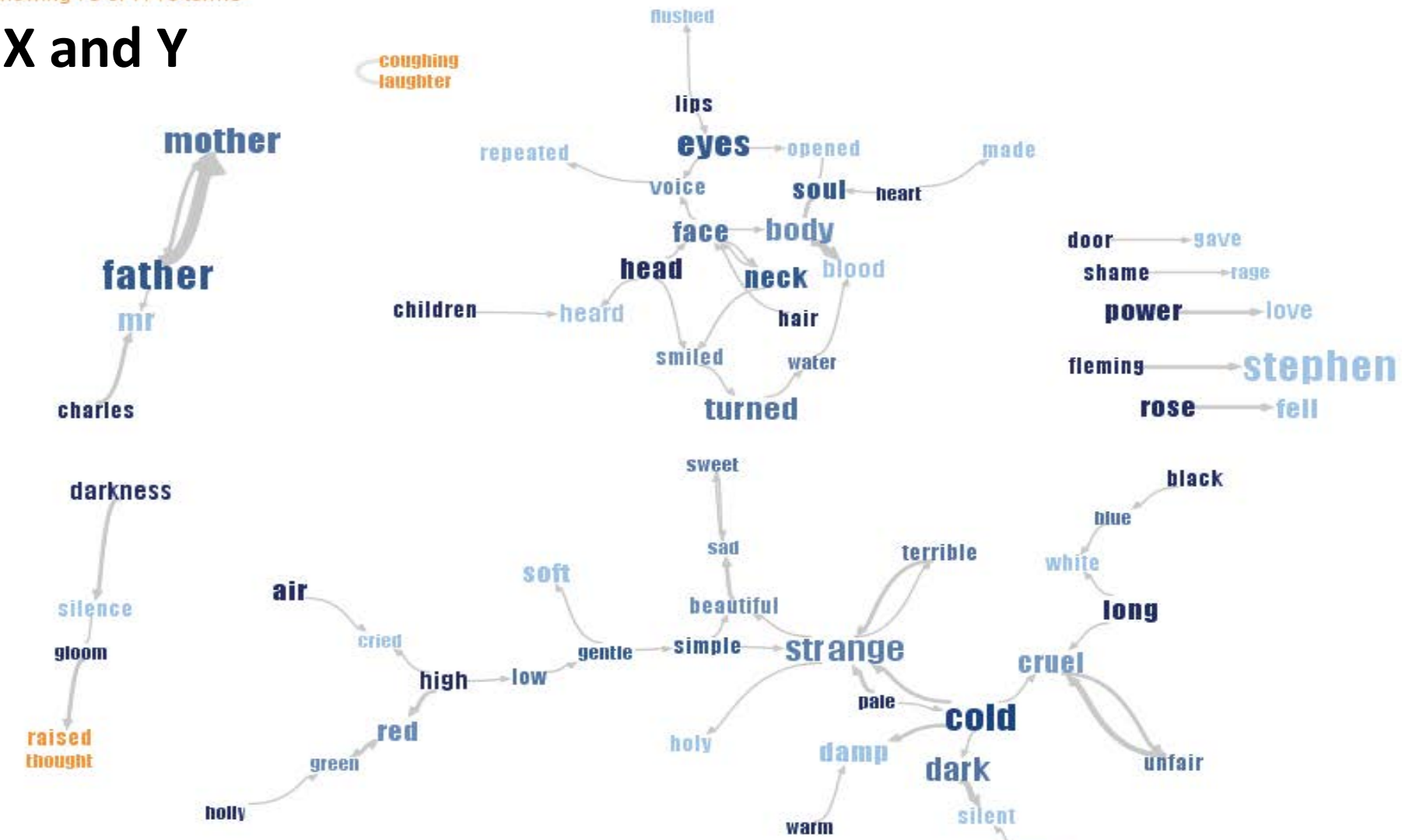
Hide common words

Zoom

In Out Reset

Showing 73 of 1719 terms

X and Y

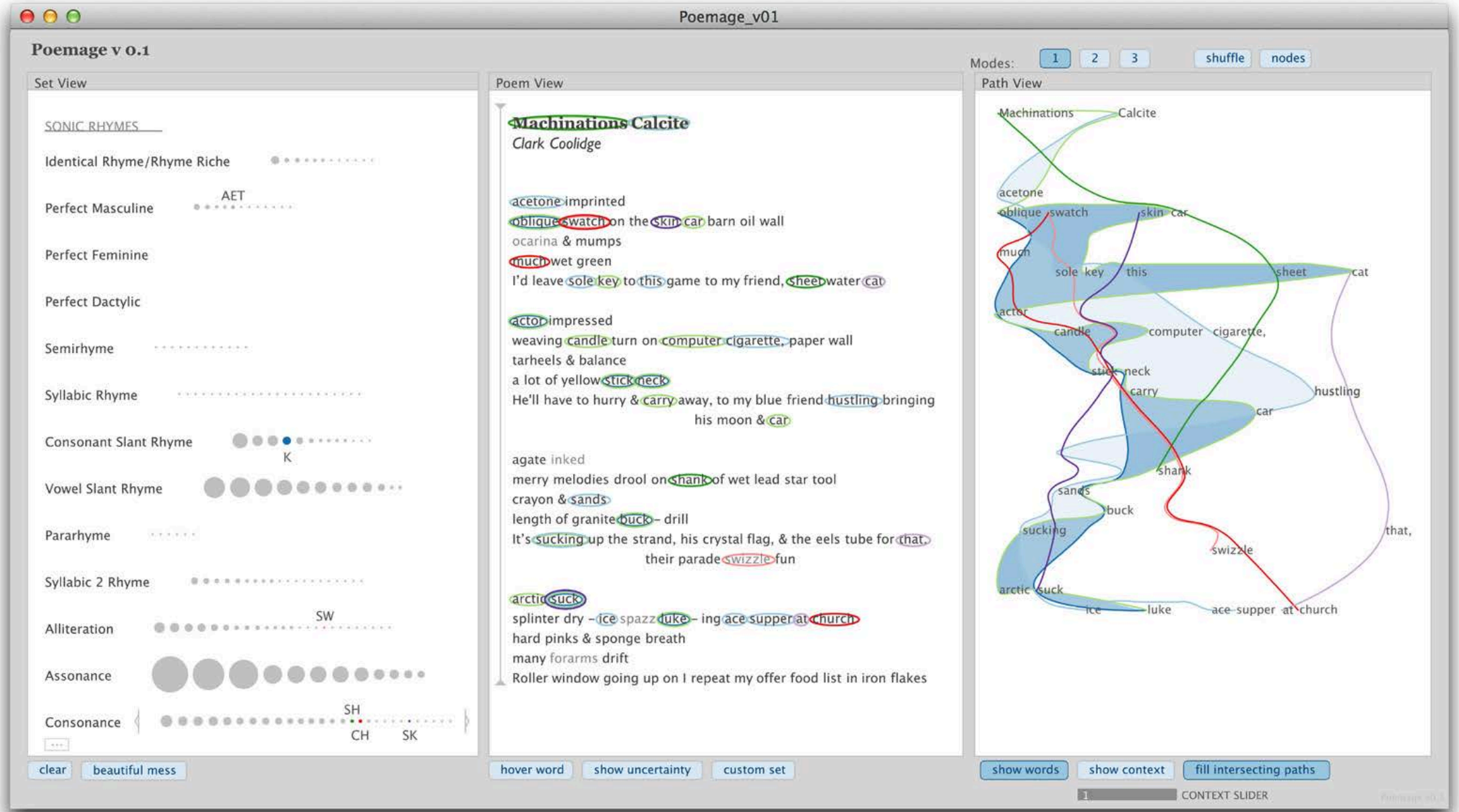


PORTRAIT OF THE ARTIST AS A YOUNG MAN

JAMES JOYCE

RHYME, SPEECH, ETC.

POEMAGE McCurdy Et al. 2016



NLTK (Natural Language ToolKit)

Tokenize and tag some text:

```
>>> import nltk
>>> sentence = """At eight o'clock on Thursday morning
... Arthur didn't feel very good."""
>>> tokens = nltk.word_tokenize(sentence)
>>> tokens
['At', 'eight', "o'clock", 'on', 'Thursday', 'morning',
'Arthur', 'did', "n't", 'feel', 'very', 'good', '.']
>>> tagged = nltk.pos_tag(tokens)
>>> tagged[0:6]
[('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'), ('on', 'IN'),
('Thursday', 'NNP'), ('morning', 'NN')]
```

Identify named entities:

```
>>> entities = nltk.chunk.ne_chunk(tagged)
>>> entities
Tree('S', [('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'),
           ('on', 'IN'), ('Thursday', 'NNP'), ('morning', 'NN'),
      Tree('PERSON', [('Arthur', 'NNP')]),
           ('did', 'VBD'), ("n't", 'RB'), ('feel', 'VB'),
           ('very', 'RB'), ('good', 'JJ'), ('.', '.')])
```

LET'S TRY SOME TEXT ANALYSIS

DOCUMENTS AND COLLECTIONS

COMMON DOCUMENT-LEVEL METRICS

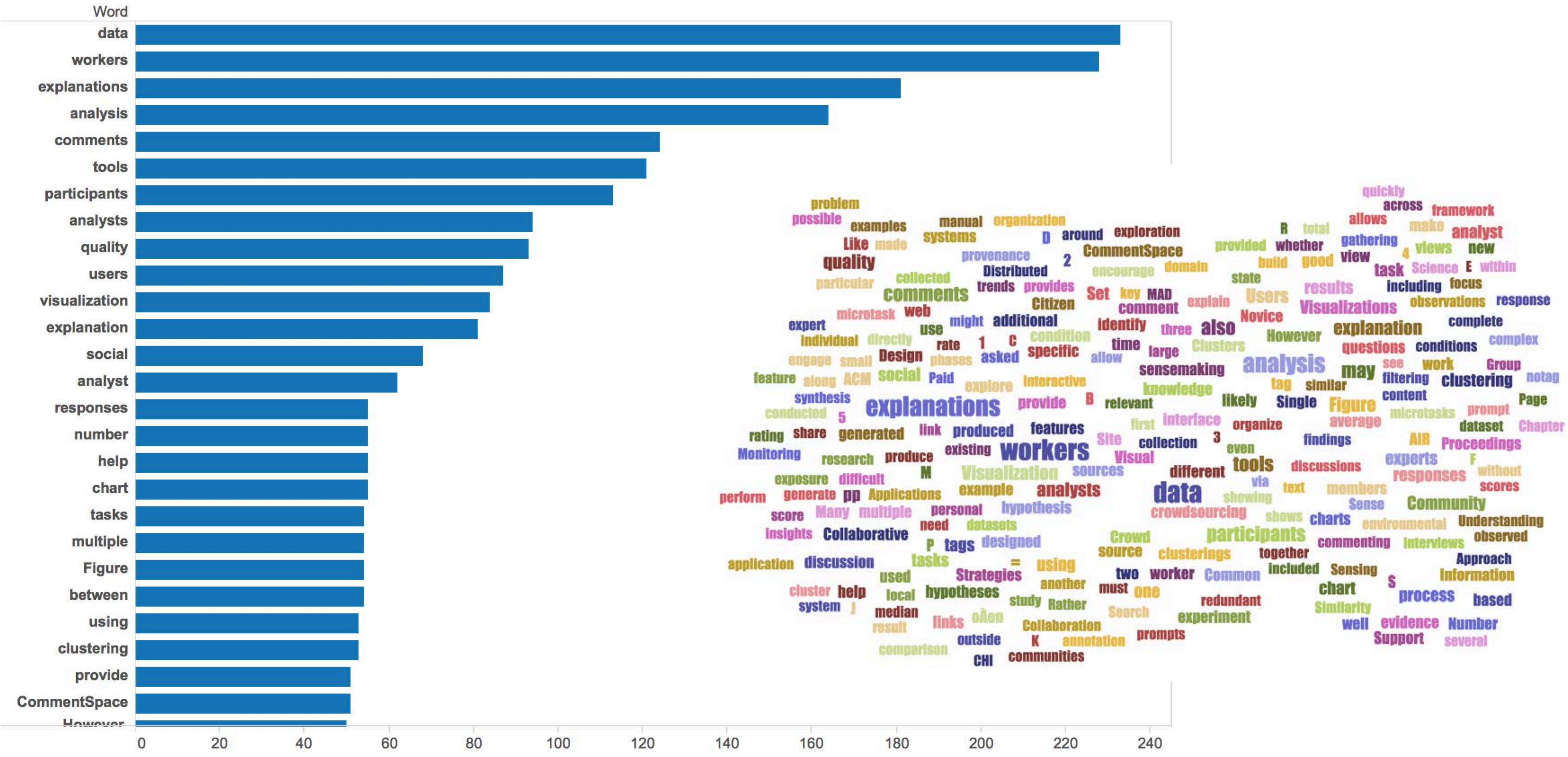
Measuing text complexity/author identity
(often used in literary analysis)

- Word length
- Syllables per word
- Average sentence length
- Percentage by parts of speech (nouns, verbs, etc.)
- Frequencies of specific words
- Language diversity (number of words used)
- Hapax Legomena (words that appear only once)

HOW OFTEN DOES A GIVEN WORD APPEAR IN A DOUMENT?

WHAT TERMS ARE THE MOST REPRESENTATIVE OF OR MOST UNIQUE TO THIS DOCUMENT?

WORD COUNTS


Word
data
workers
explanations
analysis
comments
tools
participants
analysts
quality
users
visualization
explanation
social
analyst
responses
number
help
chart
tasks
multiple
Figure
between
using
clustering
provide
CommentSpace
0
20
40
60
80
100
120
140
160
180
200
220
240

~~WORD CLOUDS~~

... are usually a bad idea

- Size and word length are conflated
- Hard to visually search/compare
- Color skews salience
- No context

“The mullets of the internet.”

Google Books Ngram Viewer

Graph these comma-separated phrases: Albert Einstein,Sherlock Holmes,Frankenstein case-insensitive

between 1800 and 2000 from the corpus English with smoothing of 3. Search lots of books

(click on line/label for focus)

IDENTIFYING IMPORTANT WORDS

Raw frequency isn't always appropriate

Stop words can pollute counts.
We often want to consider how counts compare to our expectations.

Weighting schemes

TF-IDF

weight by number of times word appears in collection

Probabilities

weight by probability of seeing the word

Etc.

TF-IDF (Term Frequency - Inverse Document Frequency)

Term Frequency

How **common** is this term in this doc.

for a term *t* in a document *d* in a corpus of N documents

tf_{td} = count(t) in d

$\log(1+ tf_{td})$ ← log frequency to smooth

$tf_{td} / \Sigma_t tf_{td}$ ← normalize based on total number of terms in doc

TF-IDF

How **unique** is this term to this doc.

$tf.idf_{td} = \log(1+ tf_{td}) \times \log(N/df_t)$ ← weight by the **inverse document frequency**
(fraction of the total docs that contain the term)

OTHER FREQUENCY STATISTICS

[Chuang et al 2012]

Table III. Frequency Statistics

Statistic	Definition
log(tf)	$\log(t_{\text{Doc}})$
tf.idf	$(t_{\text{Doc}}/t_{\text{Ref}}) \cdot \log(N/D)$
G^2	$2\left(t_{\text{Doc}} \log \frac{t_{\text{Doc}} \cdot T_{\text{Ref}}}{T_{\text{Doc}} \cdot T_{\text{Doc}}} + t_{\overline{\text{Doc}}} \log \frac{t_{\overline{\text{Doc}}} \cdot T_{\text{Ref}}}{T_{\overline{\text{Doc}}} \cdot T_{\text{Doc}}}\right)$
BM25	$3 \cdot t_{\text{Doc}} / (t_{\text{Doc}} + 2(0.25 + 0.75 \cdot T_{\text{Doc}}/r)) \cdot \log(N/D)$
WordScore	$(t_{\text{Doc}} - t_{\text{Ref}}) / \left(T_{\overline{\text{Doc}}} - T_{\overline{\text{Ref}}}\right)$
log-odds ratio (weighted)	$\left(\log \frac{t'_{\text{Doc}}}{t'_{\overline{\text{Doc}}}} - \log \frac{T'_{\text{Doc}}}{T'_{\overline{\text{Doc}}}}\right) / \sqrt{\frac{1}{t'_{\text{Doc}}} + \frac{1}{t'_{\overline{\text{Doc}}}}}$

FINDING SIMILAR DOCUMENTS...
AND IDENTIFYING GROUPS

MANY APPROACHES EXIST (WE'LL LOOK AT 2)

(1) COMPUTE SIMILARITY BETWEEN DOCUMENTS

Based on the words they share.

For example, use words' TF-IDF scores to compute similarity.

(2) TOPIC MODELING

Assume documents mixtures of multiple topics.

Model topics based on co-occurring terms

SIMILARITY - TREAT WORDS AS FEATURES

Rex O'Saurus @UCRex · Oct 20

HUGE good luck, positive vibes, major butt kicking feels to our Lady Dinos Rugby team - competing for the CanWest title this w/end! #GoDinos

Rex O'Saurus @UCRex · Oct 29

MAJOR GOOD LUCK to ⚽⚽⚽ in playoffs today! Get out and support them! They're wayyyy better than this I promise 👇 #GoDinos #WeAreAllDinos

Rex O'Saurus @UCRex · Sep 25

Your @UCDinos record this weekend was 6-3! Looking to go 8-0 next weekend! #DinosPride #WeAreAllDinos #GoDinos #ucalgary50 🏈⚽🏉🏑

CAN USE BINARY 0/1, COUNTS, NORMALIZED, TF-IDF SCORES, ETC.

IF YOU WANT CLUSTERS, JUST APPLY YOUR FAVORITE CLUSTERING TECHNIQUE!

Tweet #	good	luck	Dinos	Rugby	weekend	…
1	1	1	1	1	0	…
2	1	1	0	0	0	…
3	0	0	0	0	2	

SOME CHALLENGES

Space of all words is **really large.**

Similarity matrices can be **very sparse.**

Synonyms, homonyms, and **multiple definitions** (mean, lead, etc.).

Language is nuanced! Discrete clusters may not make sense.

TOPIC MODELING

Identify a set of "topics" that describe documents.

(Weighted combinations of terms.)

CAT = (0.25*"cat" + 0.15*"meow" + 0.004*"toy" + …)

TOYS = (0.10*"toy" + 0.04*"child" + 0.002*"game" + …)

Lost Cat! ---------- ---------- ---------- -------

30% CAT

…

New Fall Toys -- ---------- ---------- ----------

40% TOYS

…

Frolicat---------- ---------- ------- ---------- ------

18% CAT

22% TOYS

…

LOTS OF TECHNIQUES

LSA - Latent Semantic Analysis

NMF - Non-Negative Matrix Factorization

LDA - Latent Dirichlet Allocation

...

Less rigid than other classification methods. Documents can match **multiple topics simultaneously** and to **varying degrees**.

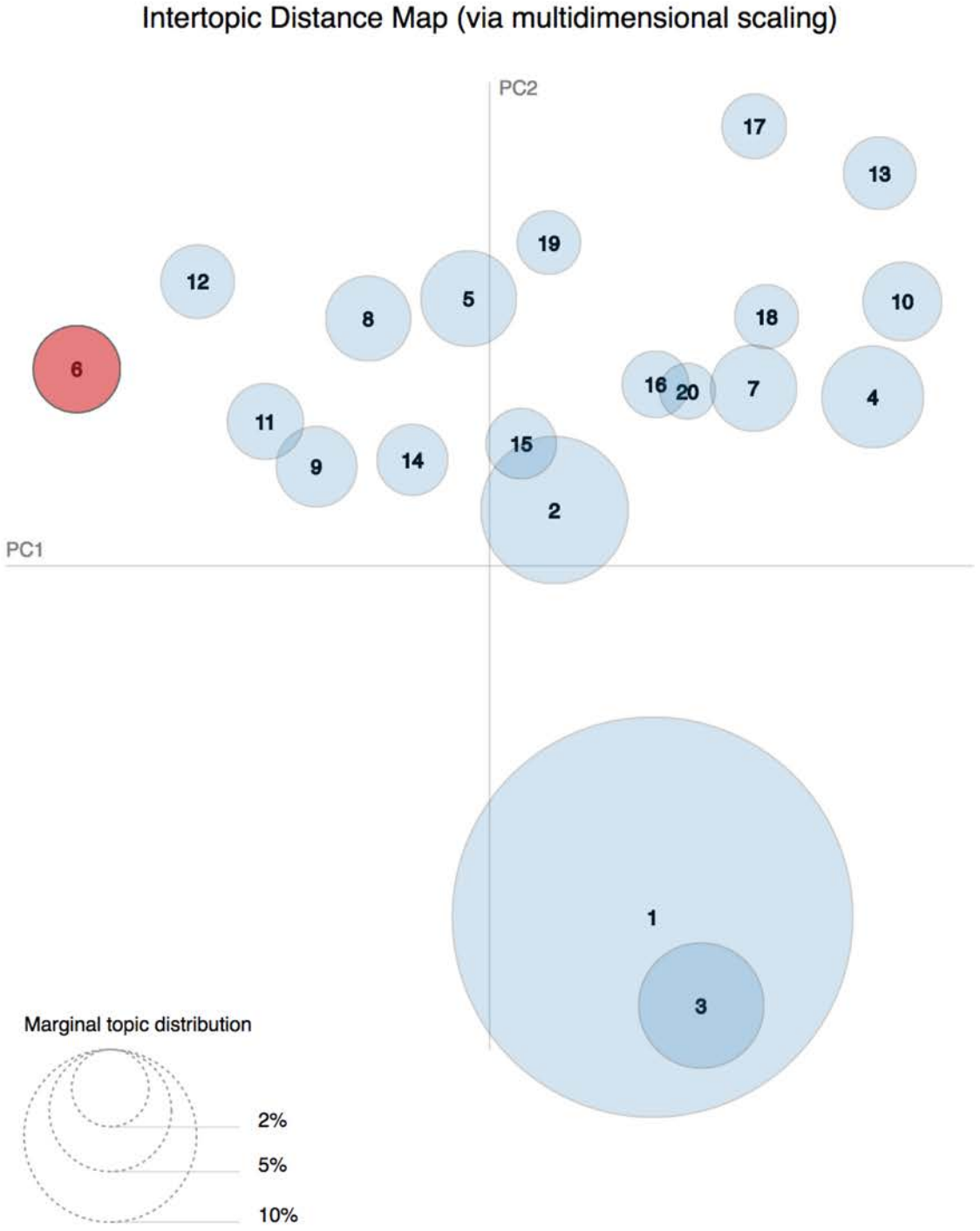
TOPIC MODELING WITH LDA
(A SIMPLIFIED EXPLANATION)

- Decide how many topics you want to model.
- For each document, randomly assign each word to a topic.
  (Gives initial word/topic and doc/topic mappings … but really bad ones.)
- **For each word** in each **document** compute:

  P(topic | document) = fraction of words in a doc assigned to that topic

  P(word | topic) = fraction of assignments to topic due to that word

  Assign **word** a **new topic** based on **P(topic | document) * P(word | topic)**
  (Basically assumes other word-topic mappings are right and we're just fixing this one.)
- **Repeat** until topics converge.



20 topics based on the text of 2000 movie reviews.

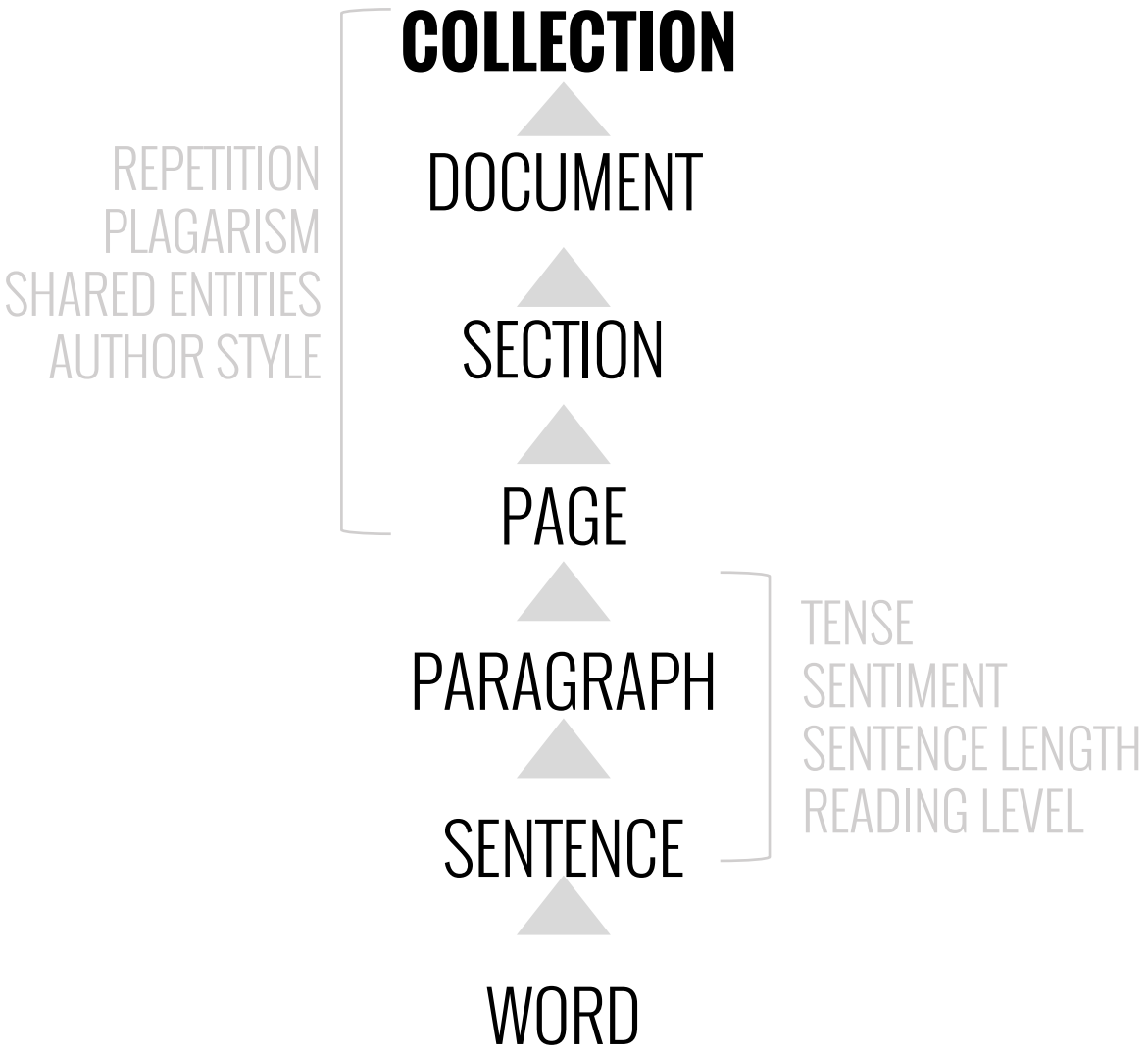
SUMMARY

Text can be a really **rich** data source.

(Mostly) **unstructured** nature means analysis is sometimes more complex.

Can work at **many different levels**
Word→Sentence→Doc→Collection

Lots of **analysis approaches** and lots of **visualization techniques**.



LOTS OF TEXT VISUALIZATION TECHNIQUES