Text Mining for Historical Linguistics

Joseph Roy

ICHL 23

San Antonio, TX

Schedule for Today

- Session 1: 10 - noon
  - Introduction/Intro to R
  - Pre-processing
  - Basic Topic Modeling
- Session 2: 1:30 – 3pm
  - Basic Topic Modeling
  - Advanced Topic Modeling
- Coffee Break: 3-3:15
- Session 3: 3:15-4pm
  - Advanced Topic Modeling
  - Word2Vec

Introduction

Traditional Statistical Approaches to Historical Linguistics

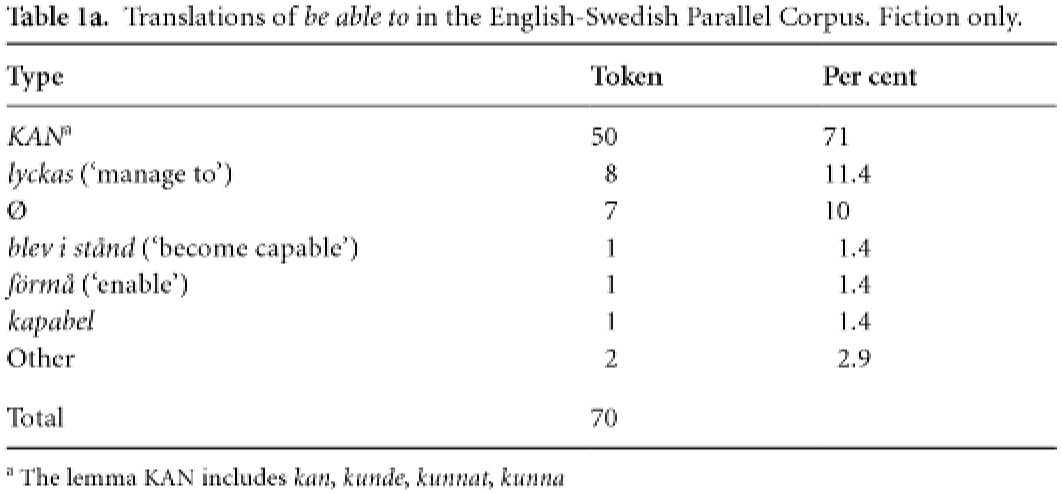


Table 1a. Translations of *be able to* in the English-Swedish Parallel Corpus. Fiction only.

Type	Token	Per cent
KAN[a]	50	71
lyckas ('manage to')	8	11.4
Ø	7	10
blev i stånd ('become capable')	1	1.4
förmå ('enable')	1	1.4
kapabel	1	1.4
Other	2	2.9
Total	70	

[a] The lemma KAN includes *kan, kunde, kunnat, kunna*

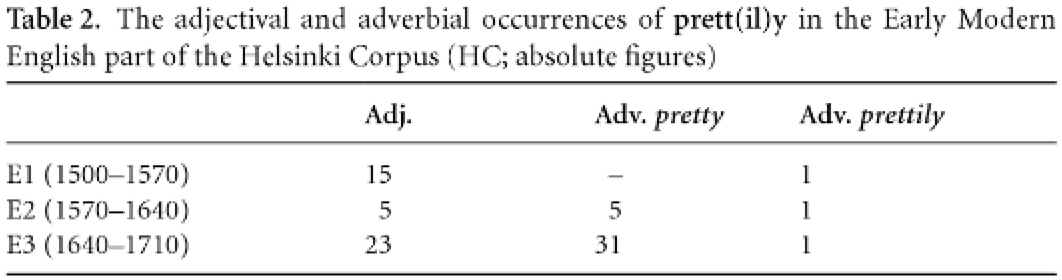
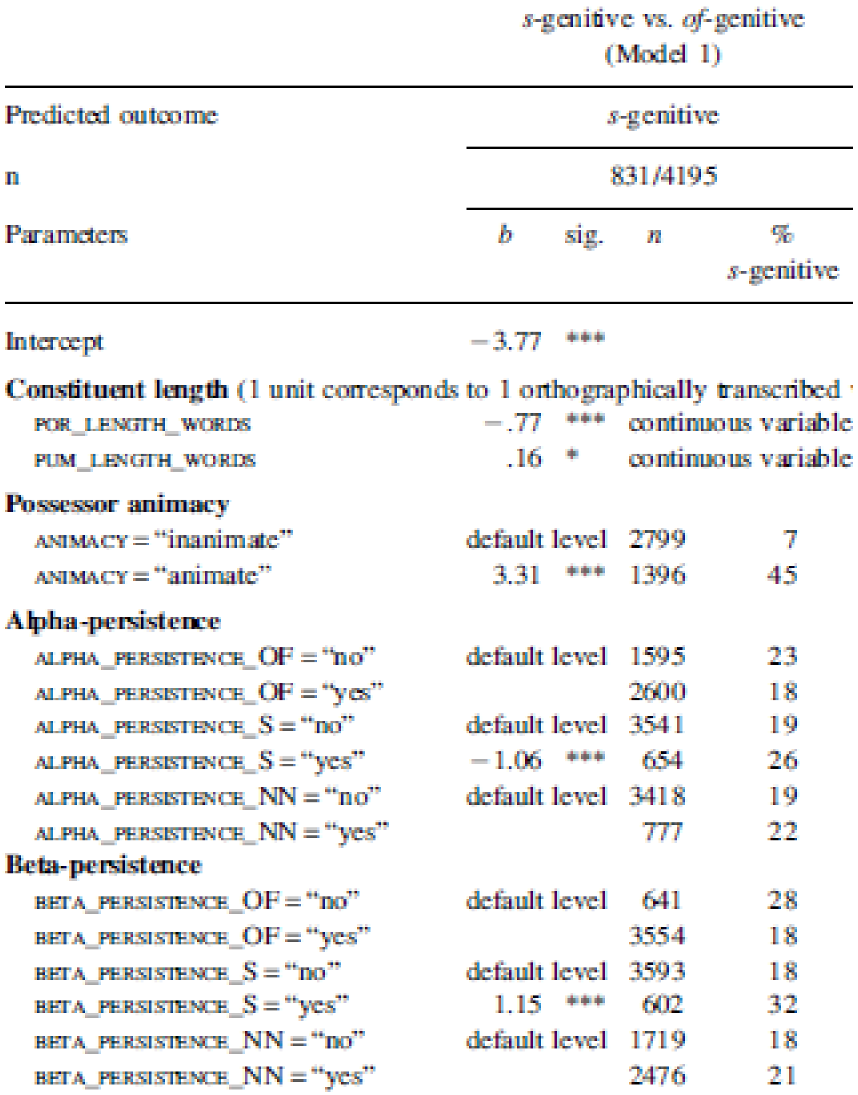


Table 2. The adjectival and adverbial occurrences of **prett(il)y** in the Early Modern English part of the Helsinki Corpus (HC; absolute figures)

	Adj.	Adv. *pretty*	Adv. *prettily*
E1 (1500–1570)	15	–	1
E2 (1570–1640)	5	5	1
E3 (1640–1710)	23	31	1



	s-genitive vs. *of*-genitive (Model 1)			
Predicted outcome	*s*-genitive			
n	831/4195			
Parameters	*b*	sig.	*n*	% *s*-genitive
Intercept	−3.77	***		
Constituent length (1 unit corresponds to 1 orthographically transcribed				
POR_LENGTH_WORDS	−.77	***	continuous variable	
PUM_LENGTH_WORDS	.16	*	continuous variable	
Possessor animacy				
ANIMACY = "inanimate"	default level		2799	7
ANIMACY = "animate"	3.31	***	1396	45
Alpha-persistence				
ALPHA_PERSISTENCE_OF = "no"	default level		1595	23
ALPHA_PERSISTENCE_OF = "yes"			2600	18
ALPHA_PERSISTENCE_S = "no"	default level		3541	19
ALPHA_PERSISTENCE_S = "yes"	−1.06	***	654	26
ALPHA_PERSISTENCE_NN = "no"	default level		3418	19
ALPHA_PERSISTENCE_NN = "yes"			777	22
Beta-persistence				
BETA_PERSISTENCE_OF = "no"	default level		641	28
BETA_PERSISTENCE_OF = "yes"			3554	18
BETA_PERSISTENCE_S = "no"	default level		3593	18
BETA_PERSISTENCE_S = "yes"	1.15	***	602	32
BETA_PERSISTENCE_NN = "no"	default level		1719	18
BETA_PERSISTENCE_NN = "yes"			2476	21

Decade	Fiction	Magazines	Newspaper	NF Books	Total	Percent fiction
1810s	641,164	88,316	0	451,542	1,181,022	0.54
1820s	3,751,204	1,714,789	0	1,461,012	6,927,005	0.54
1830s	7,590,350	3,145,575	0	3,038,062	13,773,987	0.55
1840s	8,850,886	3,554,534	0	3,641,434	16,046,854	0.55
1850s	9,094,346	4,220,558	0	3,178,922	16,493,826	0.55
1860s	9,450,562	4,437,941	262,198	2,974,401	17,125,102	0.55
1870s	10,291,968	4,452,192	1,030,560	2,835,440	18,610,160	0.55
1880s	11,215,065	4,481,568	1,355,456	3,820,766	20,872,855	0.54
1890s	11,212,219	4,679,486	1,383,948	3,907,730	21,183,383	0.53
1900s	12,029,439	5,062,650	1,433,576	4,015,567	22,541,232	0.53
1910s	11,935,701	5,694,710	1,489,942	3,534,899	22,655,252	0.53
1920s	12,539,681	5,841,678	3,552,699	3,698,353	25,632,411	0.49
1930s	11,876,996	5,910,095	3,545,527	3,080,629	24,413,247	0.49
1940s	11,946,743	5,644,216	3,497,509	3,056,010	24,144,478	0.49
1950s	11,986,437	5,796,823	3,522,545	3,092,375	24,398,180	0.49
1960s	11,578,880	5,803,276	3,404,244	3,141,582	23,927,982	0.48
1970s	11,626,911	5,755,537	3,383,924	3,002,933	23,769,305	0.49
1980s	12,152,603	5,804,320	4,113,254	3,108,775	25,178,952	0.48
1990s	13,272,162	7,440,305	4,060,570	3,104,303	27,877,340	0.48
2000s	14,590,078	7,678,830	4,088,704	3,121,839	29,479,451	0.49
Total	207,633,395	97,207,399	40,124,656	61,266,574	**406,232,024**	0.51

Table 1: Composition of COHA by genre and decade

Text Mining

“Text Mining is the discovery by computer of new, previously unknown information, by automatically extracting information from different written resources. A key element is the linking together of the extracted information together to form new facts or new hypotheses to be explored further by more conventional means of experimentation.”

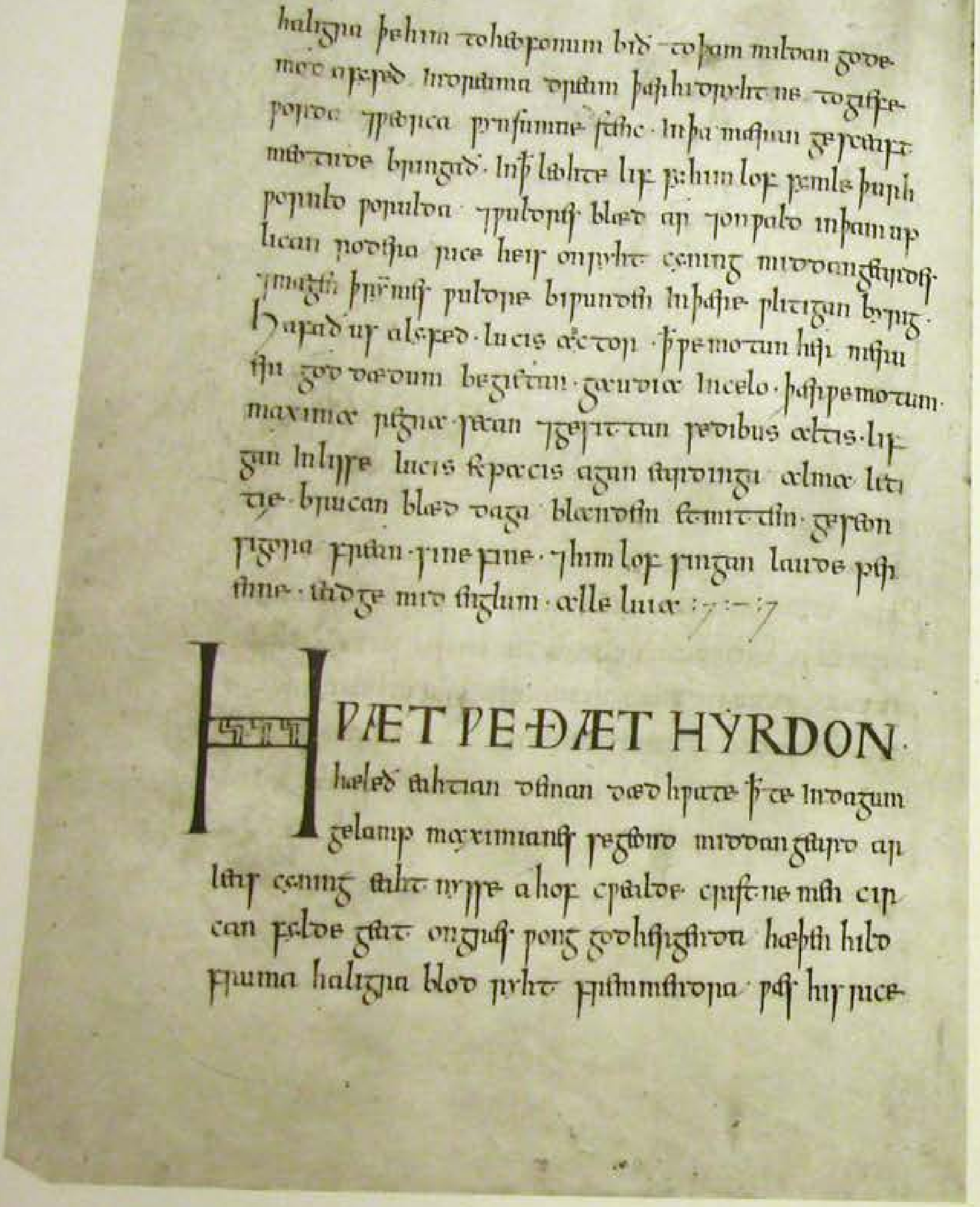
-Marti Hearst

Text Mining Procedures

- Natural Language Processing: Parsing, tagging text for part of speech, other linguistic analyses
- Sentiment Analysis: Identifying the sentiment, opinion, mood, and emotion of a text.
- Topic Modeling (Today)

Applications

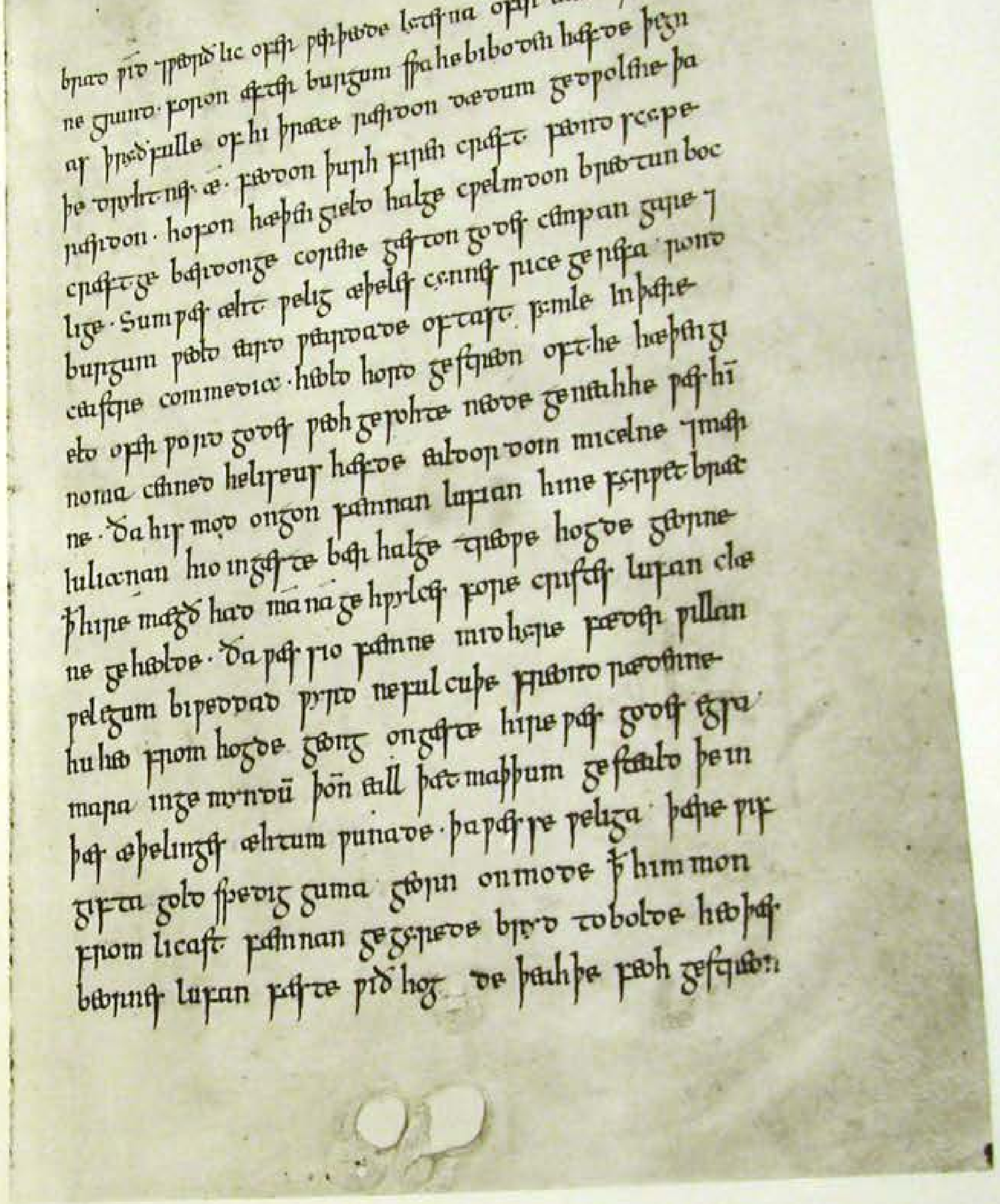
- Social Media Analysis
  - Kate Lyons – Linguistic Landscapes: https://jroy042.github.io/textmining_sociolinguistics/presentations/Lyons_Workshop_Slides.html
- English Borrowings in French
  - Gyula Zsombok– Lexical Borrowing in Newspaper Corpus/Social Media https://jroy042.github.io/textmining_sociolinguistics/presentations/ZsombokWorkshop.pdf
- Higher Education Policy
  - Corpus of Management Letters in Audited Financial Statements (2005-2015) at R1 Universities.
  - Job descriptions for Academic Professionals at UIUC 2013-2016.



haligra þe him to hiofonum bið · to þam mildan gode
mod afysed · in dreama dream þær hi dryhtne to giefe
worde ⁊ weorca wynsumne stenc · in þa mæran gesceaft
meotude bringað · in ꝥ leohte lif · sy him lof symle þurh
woruld worulda ⁊ wuldres blæd · ar ⁊ onwald in þam up
lican rodera rice · he is on ryht cyning middangeardes
⁊ mægenþrymmes wuldre biwunden in þære wlitigan byrig ·
Hafað us alyfed · lucis auctor · ꝥ we motun her merue
ri goddædum begietan · gaudia in celo · þær we motun
maxima regna · secan ⁊ gesittan sedibus altis · lif
gan in lisse lucis & pacis agan eardinga alma leti
tie · brucan blæddaga blandem & mitem · geseon
sigora frean · sine fine · ⁊ him lof singan laude per
enne · eadge mid englum · alleluia :7:–:7

HWÆT WE ÐÆT HYRDON
hæleð eahtian deman dædhwate ꝥte in dagum
gelamp maximianes se geond middangeard ar
leas cyning eahtnysse ahof cwealde cristne men cir
can fylde geat on græswong godhergendra hæþen hild
fruma haligra blod ryhtfremmendra · wæs his rice

fol. 65b
THE PHŒNIX, ll. 656-end; JULIANA, ll. 1-8.



brad wid ⁊ weorðlic ofer werþeode lytesna ofer
ne grund · foron æfter burgum swa he biboden hæfde þegn
as þryðfulle oft hi þræce rærdon dædum gedwolene þa
þe dryhtnes æ · feodon þurh firencræft · feondscipe
rærdon · hofon hæþengield halge cwelmdon breotun boc
cræftge bærndon gecorene gæston godes cempan gare ⁊
lige · Sum wæs æhtwelig æþeles cynnes rice gerefa · rond
burgum weold eard weardade oftast symle in þære
ceastre commedia · heold hordgestreon · oft he hæþengi
eld ofer word godes weoh gesohte neode geneahhe wæs hī
noma cenned heliseus hæfde ealdordom micelne ⁊ mær
ne · Ða his mod ongon fæmnan lufian hine fyrwet bræc
iulianan hio in gæste bær halge treowe hogde georne
ꝥ hire mægðhad mana gehwylces fore cristes lufan clæ
ne geheolde · Ða wæs sio fæmne mid hyre fæder willan
welegum biweddad wyrd ne ful cuþe freondrædenne
hu heo from hogde geong on gæste hire wæs godes egsa
mara in gemyndū þon̄ eall þæt maþþum gesteald þe in
þæs æþelinges æhtum wunade · þa wæs se weliga þæra wif
gifta goldspedig guma georn on mode ꝥ him mon
fromlicast fæmnan gegyrede bryd to bolde heo þæs
beornes lufan fæste wið hog de þeahþe feoh gestreon

fol. 66a
JULIANA, ll. 8-42.

Data for today

Pamphlets of the American Revolution

http://ota.ox.ac.uk/desc/2021

A Corpus of English Dialogues 1560-1760 (CED)

http://ota.ox.ac.uk/desc/2507

Penn Parsed Corpus of Modern British English (1700-1913)

https://www.ling.upenn.edu/hist-corpora/PPCMBE2-RELEASE-1/index.html

Feeding America: The Historic American Cookbook Dataset (18th-20th Century)

http://archive.lib.msu.edu/dinfo/feedingamerica/

Format of the data

- Simple text format (.txt)
  - Today: Basic roman alphabet is assumed (with accents).
    - Other alphabets, character sets principles are similar, but require other software (or added packages).
  - Text encoding: http://stat545.com/block032_character-encoding.html
    - This is a wonderful resource by a statistician at UBC, Prof. Jenny Bryan.
- Arabic: There is some support in python (https://sites.google.com/site/dyaafayedsite/)
- Chinese in R: Chinese.misc package https://cran.r-project.org/web/packages/chinese.misc/chinese.misc.pdf

Software

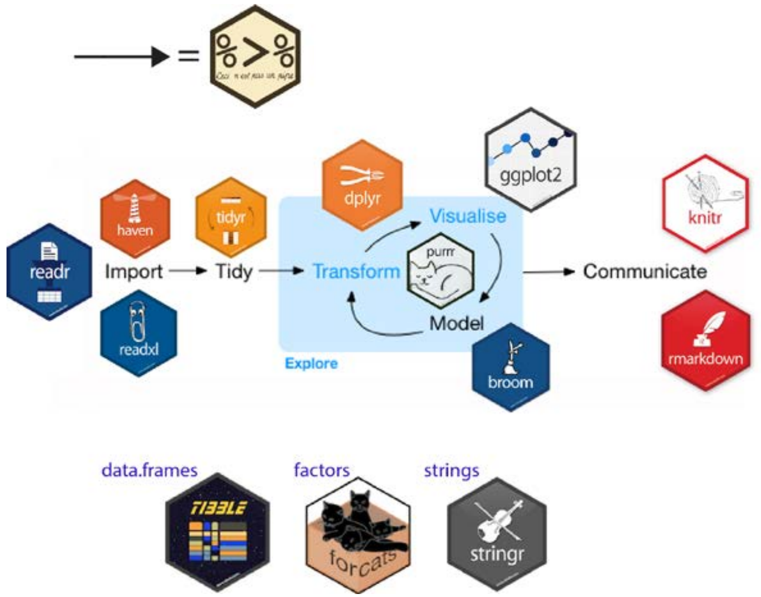
- R, Rstudio and the tidyverse
- Python
- Many other programming languages have support for these techniques (SAS, Perl, Java, C++, etc.)
- If you have “big data” (i.e. too big for a personal computer to analyze): https://www.xsede.org/xsede-call-for-humanities-art-and-sciences-projects
  - The Extreme Science and Engineering Discovery Environment (XSEDE) is seeking use cases in the humanities and social sciences. Alan Craig: acraig@ncsa.uiuc.edu
  - They also have free compute resources available that you can apply for: https://www.xsede.org/using-xsede#allocations

Software Today

R version 3.4.1 (2017-06-30) -- "Single Candle"

RStudio

Tidyverse



Pre-processing text

Pre-processing

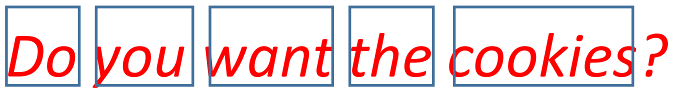
- *Pre-processing* refers to the stage where prepare the text for analysis without actually doing any analysis.
- We have a collection of texts or documents in a corpus that we need to strip all non-word items from.
- There are methodological decisions made during this phase that can effect the outcome and, at least for linguistic analysis.

First Steps

- Remove all non-lexical items.
- Any formatting characters (html, etc.)
- Any non-sentence punctuation.
- Involves using regular expressions (i.e. pattern matching shortcuts):
    - https://www.rstudio.com/wp-content/uploads/2016/09/RegExCheatsheet.pdf

Tokenizing

- Tokenizing: how we break up a text into lexical item (or other components)
- Unigrams: All one word items in a document.
- For most topic modeling:
  - *Do you want the cookies?* →



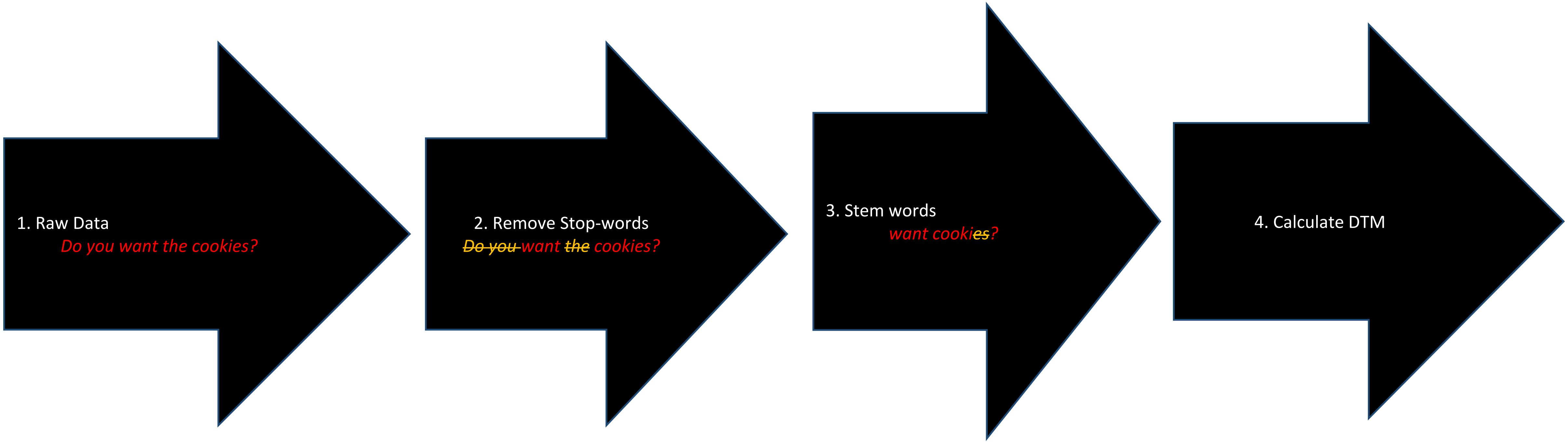
Do you want the cookies?

  - Tokenizing takes place at the word level.

N-grams

- A word and its N-1 following neighbors:
  - Do you want the cookie?
- Bi-gram (2-gram)
  - Do you
  - You want
  - Want the
  - The cookie
- Tri-gram (3-gram)
  - Do you want
  - You want the
  - Want the cookie

Standard Preprocessing Workflow



1. Raw Data

Do you want the cookies?

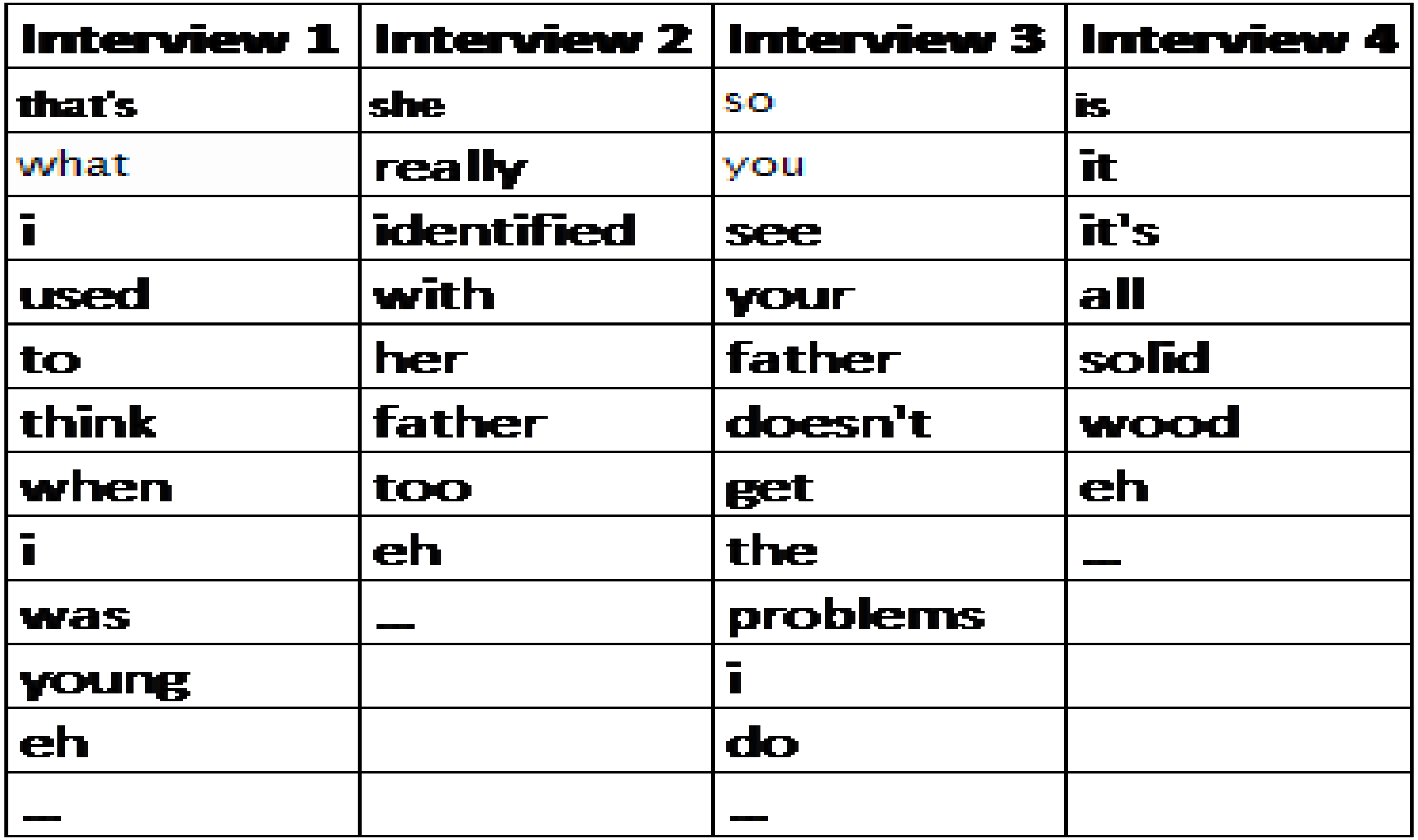
2. Remove Stop-words

~~Do you~~ want ~~the~~ cookies?

3. Stem words

want cooki~~es~~?

4. Calculate DTM



Interview 1	Interview 2	Interview 3	Interview 4
that's	she	so	is
what	really	you	it
i	identified	see	it's
used	with	your	all
to	her	father	solid
think	father	doesn't	wood
when	too	get	eh
i	eh	the	_
was	_	problems	
young		i	
eh		do	
_		_	

Stop words

- In linguistics, stop words are often function words: determiners, modals, light verbs, articles, etc. that have no content information, but do grammatical work
- Stop words in r can be changed and often have to be changed for different problems.

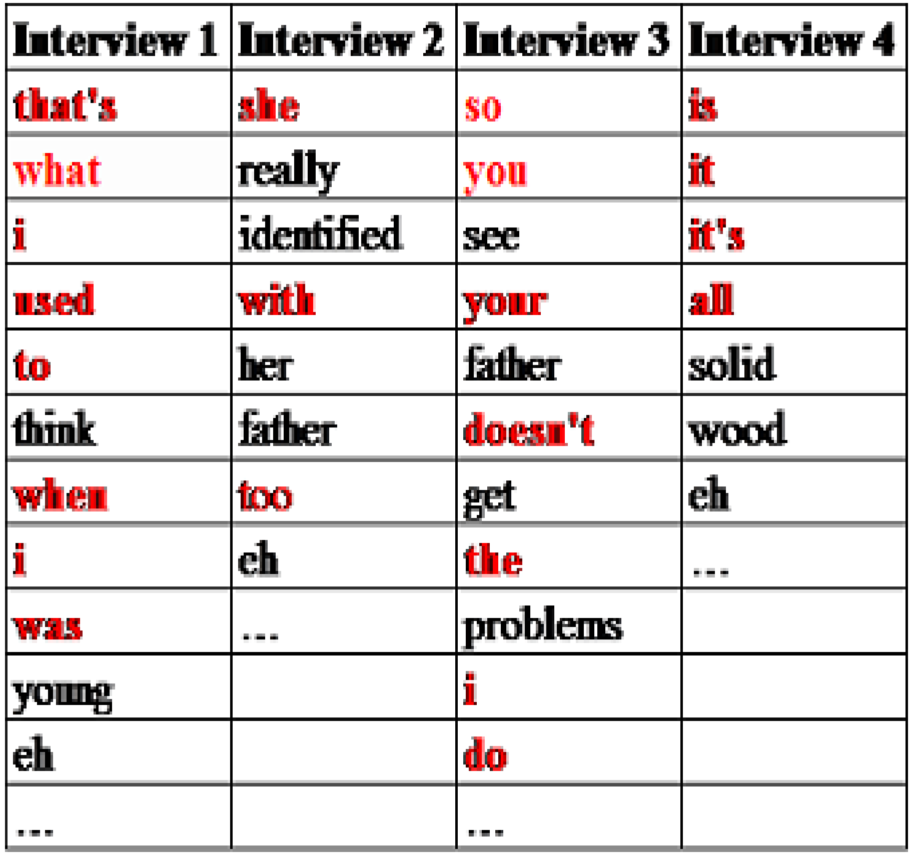
In English (tm package in R)

```
stopwords(kind="en")
  [1] "i"          "me"         "my"         "myself"     "we"         "our"        "ours"       "ourselves"  "you"        "your"
 [11] "yours"      "yourself"   "yourselves" "he"         "him"        "his"        "himself"    "she"        "her"        "hers"
 [21] "herself"    "it"         "its"        "itself"     "they"       "them"       "their"      "theirs"     "themselves" "what"
 [31] "which"      "who"        "whom"       "this"       "that"       "these"      "those"      "am"         "is"         "are"
 [41] "was"        "were"       "be"         "been"       "being"      "have"       "has"        "had"        "having"     "do"
 [51] "does"       "did"        "doing"      "would"      "should"     "could"      "ought"      "i'm"        "you're"     "he's"
 [61] "she's"      "it's"       "we're"      "they're"    "i've"       "you've"     "we've"      "they've"    "i'd"        "you'd"
 [71] "he'd"       "she'd"      "we'd"       "they'd"     "i'll"       "you'll"     "he'll"      "she'll"     "we'll"      "they'll"
 [81] "isn't"      "aren't"     "wasn't"     "weren't"    "hasn't"     "haven't"    "hadn't"     "doesn't"    "don't"      "didn't"
 [91] "won't"      "wouldn't"   "shan't"     "shouldn't"  "can't"      "cannot"     "couldn't"   "mustn't"    "let's"      "that's"
[101] "who's"      "what's"     "here's"     "there's"    "when's"     "where's"    "why's"      "how's"      "a"          "an"
[111] "the"        "and"        "but"        "if"         "or"         "because"    "as"         "until"      "while"      "of"
[121] "at"         "by"         "for"        "with"       "about"      "against"    "between"    "into"       "through"    "during"
[131] "before"     "after"      "above"      "below"      "to"         "from"       "up"         "down"       "in"         "out"
[141] "on"         "off"        "over"       "under"      "again"      "further"    "then"       "once"       "here"       "there"
[151] "when"       "where"      "why"        "how"        "all"        "any"        "both"       "each"       "few"        "more"
[161] "most"       "other"      "some"       "such"       "no"         "nor"        "not"        "only"       "own"        "same"
[171] "so"         "than"       "too"        "very"
```

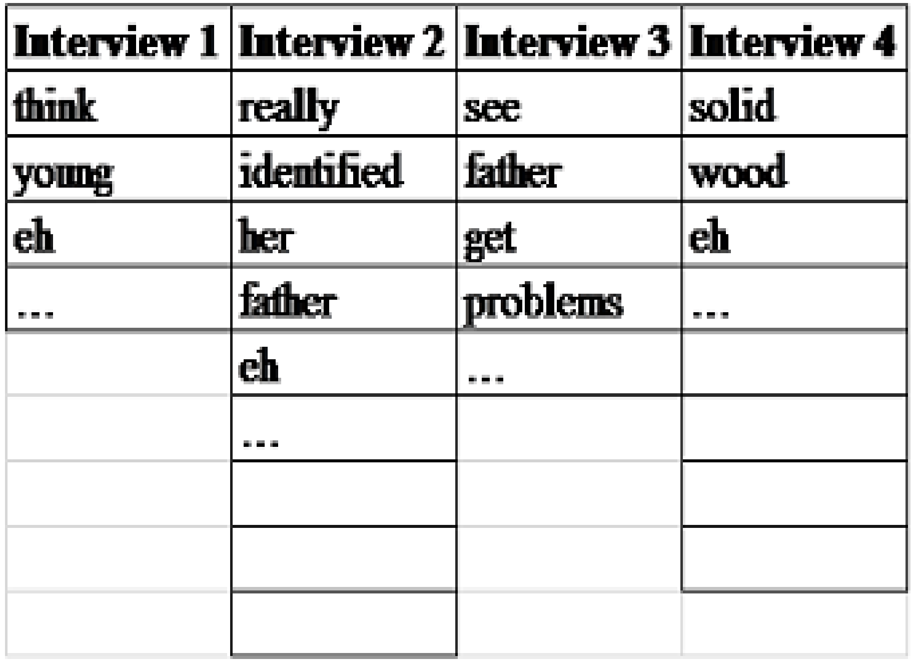
In French (tm package in R)

```
stopwords(kind="fr") stopwords(kind="fr")
  [1] "au"       "aux"      "avec"     "ce"       "ces"      "dans"     "de"       "des"      "du"       "elle"     "en"
 [12] "et"       "eux"      "il"       "je"       "la"       "le"       "leur"     "lui"      "ma"       "mais"     "me"
 [23] "même"     "mes"      "moi"      "mon"      "ne"       "nos"      "notre"    "nous"     "on"       "ou"       "par"
 [34] "pas"      "pour"     "qu"       "que"      "qui"      "sa"       "se"       "ses"      "son"      "sur"      "ta"
 [45] "te"       "tes"      "toi"      "ton"      "tu"       "un"       "une"      "vos"      "votre"    "vous"     "c"
 [56] "d"        "j"        "l"        "à"        "m"        "n"        "s"        "t"        "y"        "été"      "étée"
 [67] "étées"    "étés"     "étant"    "suis"     "es"       "est"      "sommes"   "êtes"     "sont"     "serai"    "seras"
 [78] "sera"     "serons"   "serez"    "seront"   "serais"   "serait"   "serions"  "seriez"   "seraient" "étais"    "était"
 [89] "étions"   "étiez"    "étaient"  "fus"      "fut"      "fûmes"    "fûtes"    "furent"   "sois"     "soit"     "soyons"
[100] "soyez"    "soient"   "fusse"    "fusses"   "fût"      "fussions" "fussiez"  "fussent"  "ayant"    "eu"       "eue"
[111] "eues"     "eus"      "ai"       "as"       "avons"    "avez"     "ont"      "aurai"    "auras"    "aura"     "aurons"
[122] "aurez"    "auront"   "aurais"   "aurait"   "aurions"  "auriez"   "auraient" "avais"    "avait"    "avions"   "aviez"
[133] "avaient"  "eut"      "eûmes"    "eûtes"    "eurent"   "aie"      "aies"     "ait"      "ayons"    "ayez"     "aient"
[144] "eusse"    "eusses"   "eût"      "eussions" "eussiez"  "eussent"  "ceci"     "cela"     "celà"     "cet"      "cette"
[155] "ici"      "ils"      "les"      "leurs"    "quel"     "quels"    "quelle"   "quelles"  "sans"     "soi"
```

When would we want to discard content words and not stop words?



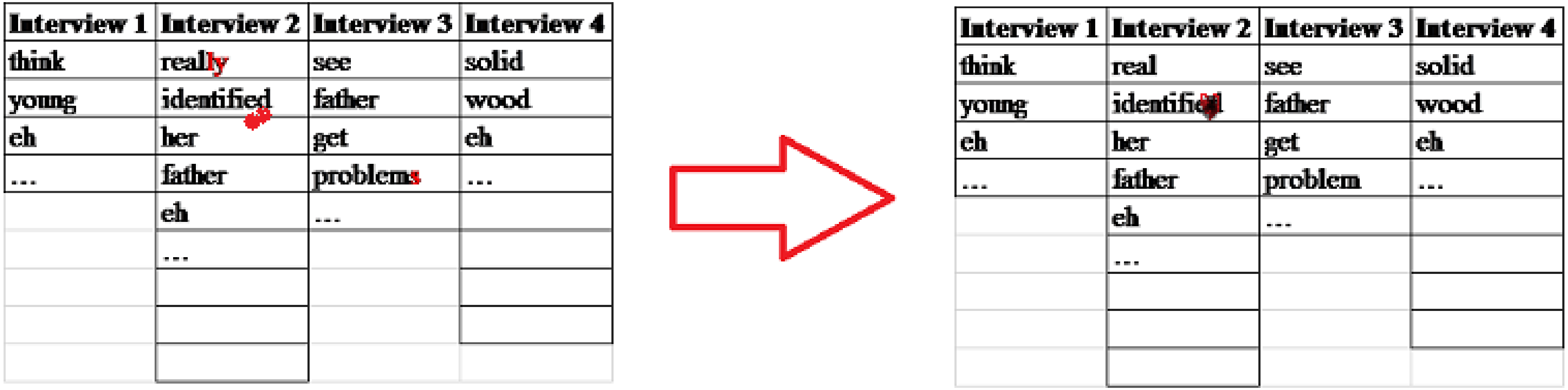
Interview 1	Interview 2	Interview 3	Interview 4
that's	she	so	is
what	really	you	it
i	identified	see	it's
used	with	your	all
to	her	father	solid
think	father	doesn't	wood
when	too	get	eh
i	eh	the	...
was	...	problems	
young		i	
eh		do	
...		...	



Interview 1	Interview 2	Interview 3	Interview 4
think	really	see	solid
young	identified	father	wood
eh	her	get	eh
...	father	problems	...
	eh	...	
	...		

Stemming

- Roughly, removes morphological endings to collapse lexical items into stem.
- NOT A LINGUISTIC ANALYSIS per se.
  - HORSES, Horse → hors
  - Running, Runs → run
- Stemming algorithms are not breaking a word into its morphemes. They are taking common know endings and knocking them off words.
- This is not the same thing as *Lemmatization*



Interview 1	Interview 2	Interview 3	Interview 4
think	really	see	solid
young	identified	father	wood
eh	her	get	eh
…	father	problems	…
	eh	…	
	…		

Interview 1	Interview 2	Interview 3	Interview 4
think	real	see	solid
young	identified	father	wood
eh	her	get	eh
…	father	problem	…
	eh	…	
	…		

Messy Historical Data

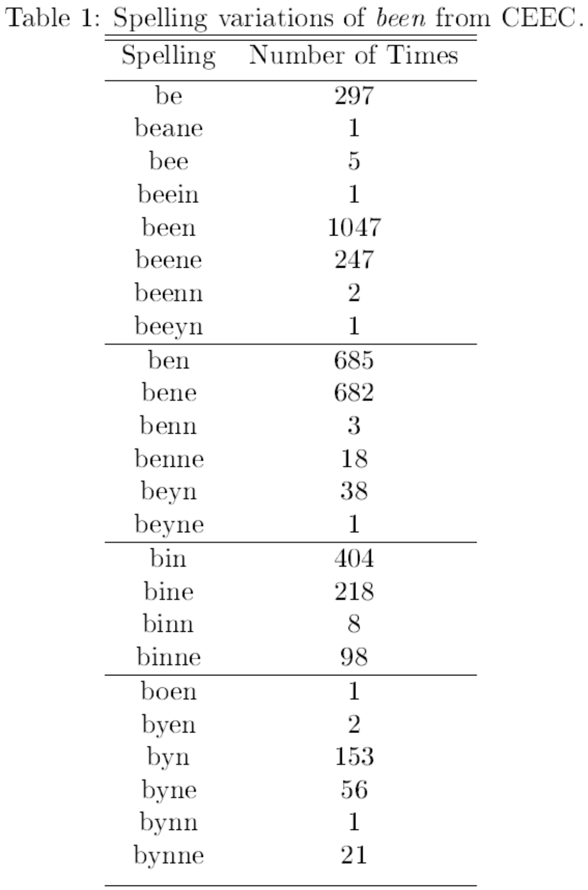
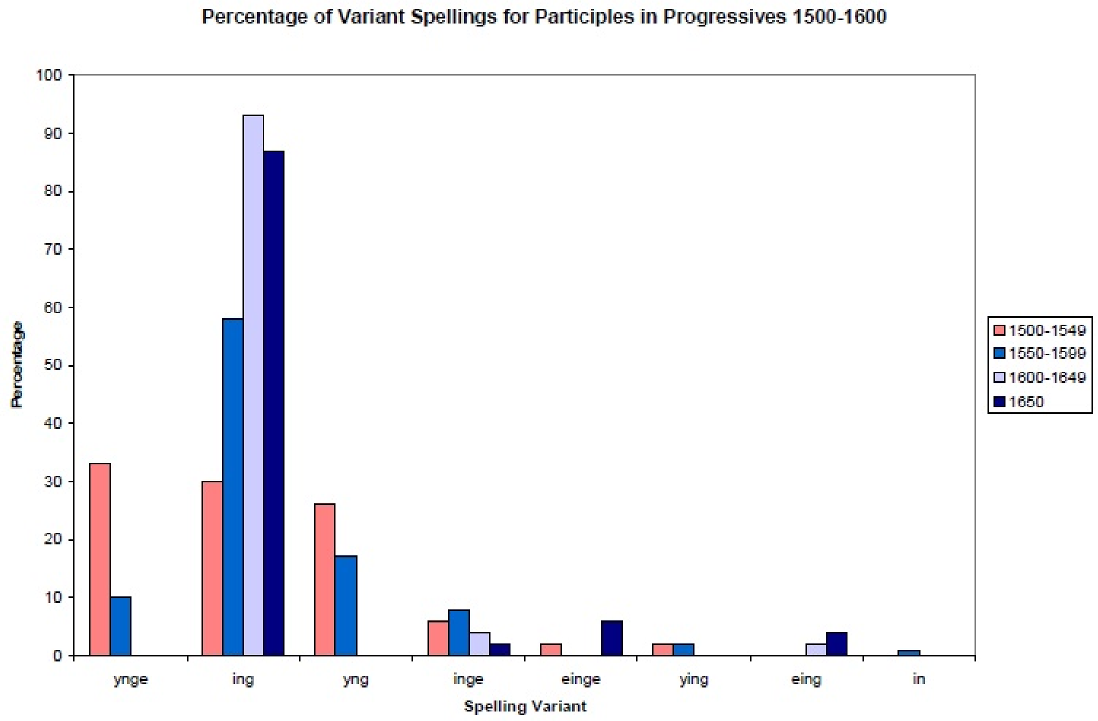
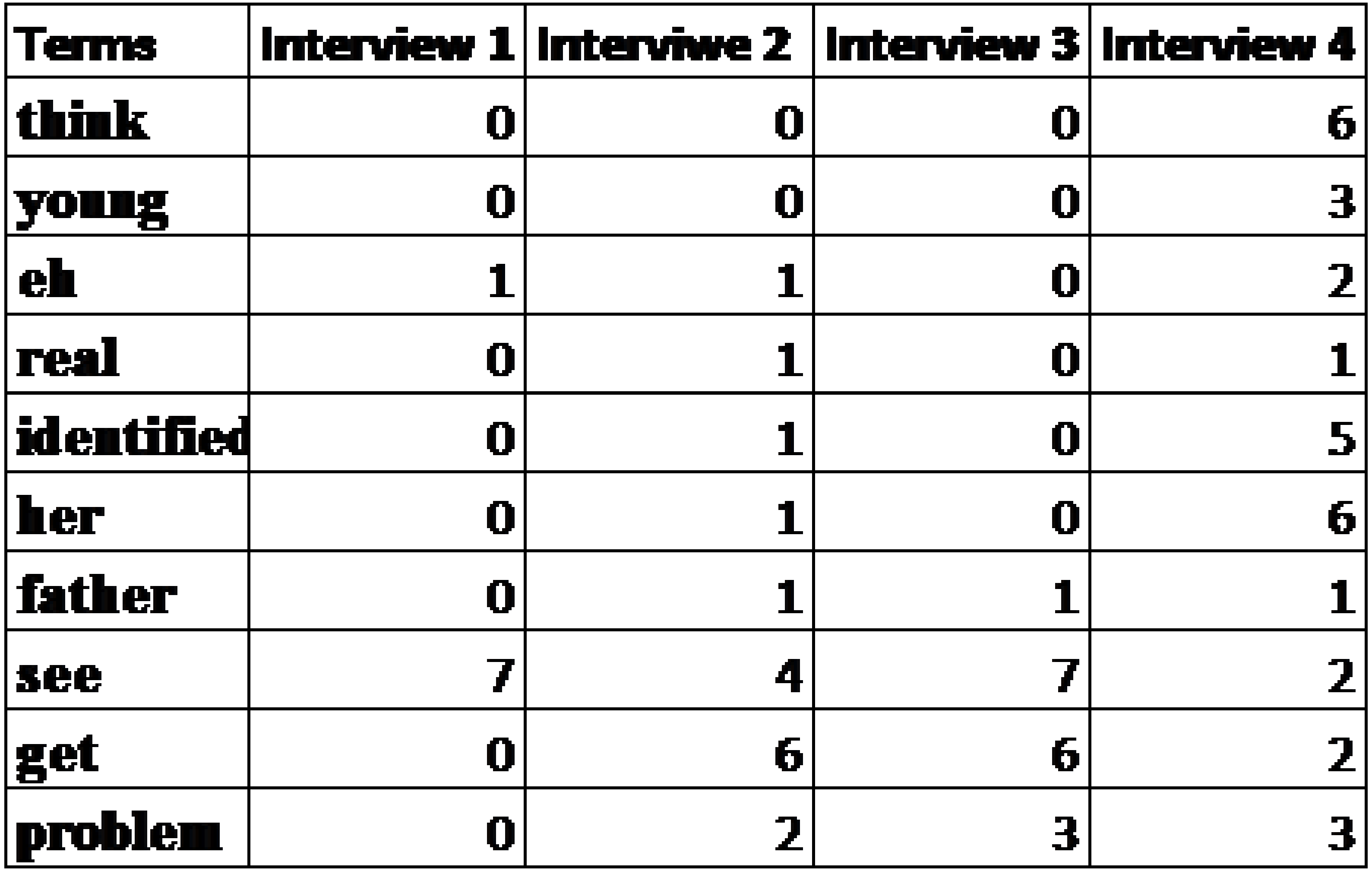


Table 1: Spelling variations of *been* from CEEC.

Spelling	Number of Times
be	297
beane	1
bee	5
beein	1
been	1047
beene	247
beenn	2
beeyn	1
ben	685
bene	682
benn	3
benne	18
beyn	38
beyne	1
bin	404
bine	218
binn	8
binne	98
boen	1
byen	2
byn	153
byne	56
bynn	1
bynne	21



Final Output Document Term Matrix



Terms	Interview 1	Interviwe 2	Interview 3	Interview 4
think	0	0	0	6
young	0	0	0	3
eh	1	1	0	2
real	0	1	0	1
identified	0	1	0	5
her	0	1	0	6
father	0	1	1	1
see	7	4	7	2
get	0	6	6	2
problem	0	2	3	3

Alternatives to TF DTM

- Term frequency DTM is just counts of terms per document
- Binary: each cell is 0 or 1 (1=the term is in the document)
- TF-IDF: Term frequency Inverse Document Frequency.

Term-Frequency Inverse Document Frequency (TD-IDF)

Given some term *i*, and a document *j,* the *term count*

is the number of times that term *i* occurs in document *j*

n_{ij}

Given a collection of *k* terms and a set *D* of documents, the *term frequency*, is:

tf_{ij}

$$tf_{ij} = \frac{n_{ij}}{\sum_{k=1}^{T} n_{kj}}$$

… considering only the terms of interest, this is the proportion of document *j* that is made up from term *i.*

TD-IDF cont.

- Term frequency tf_{ij} is a measure of the importance of term *i* in document *j*
- Inverse document frequency (which we see next) is a measure of the *general* importance of the term.
- I.e. High term frequency for “apple” means that apple is an important word in a specific document.
- But high document frequency (low inverse document frequency) for “apple”, given a particular set of documents, means that apple is not all that important overall, since it is in all of the documents.

IDF

Inverse document frequency of term *i* is:

$$idf_i = \log \frac{|D|}{\{d_j : d_j \in D\}}$$

Log of: … the number of documents in the corpus,
divided by the number of those documents that contain the term.

TF-IDF

the *i*th entry in the matrix for document *j* is:

$$tf_{ij} \times idf_i$$

Usefulness of alterative DTM

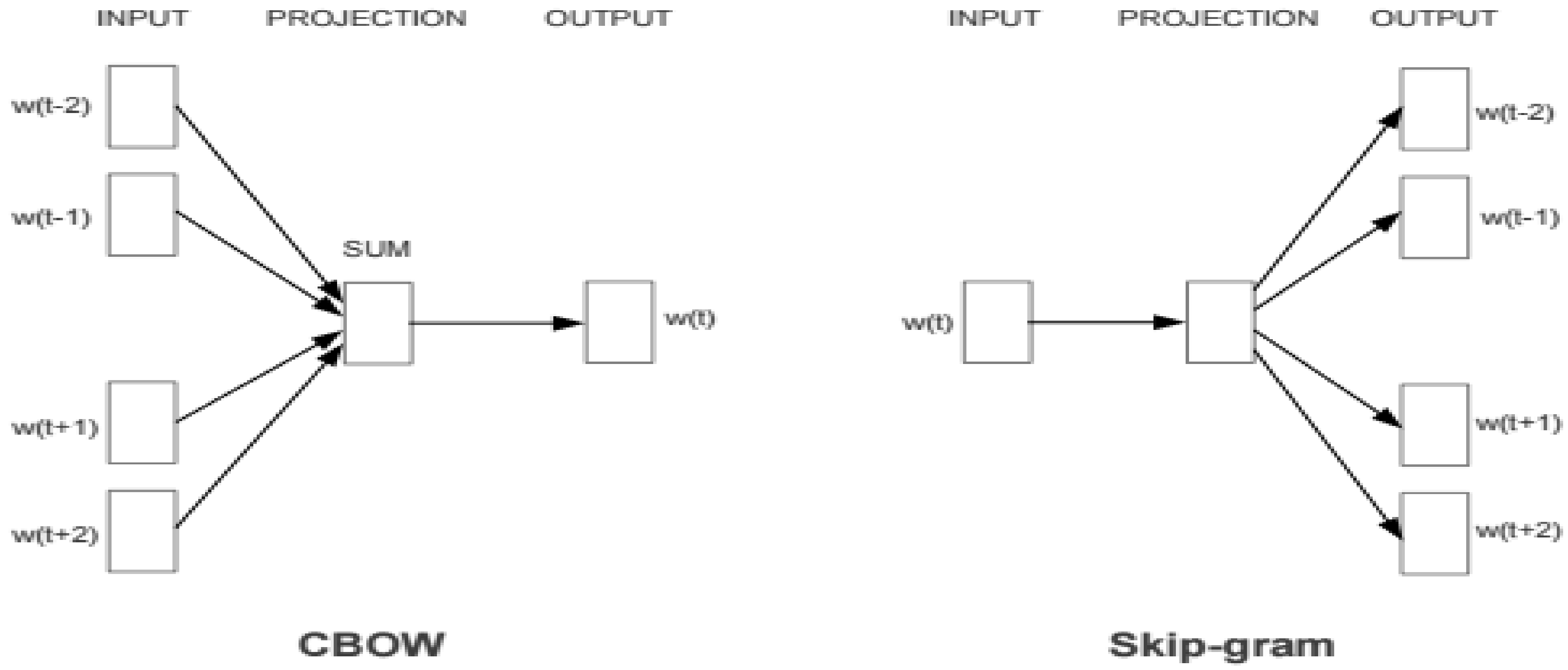
- Historically (PCA, Latent Semantic Analysis): results changed depending on how you weighted the dtm.
- Most modern text mining techniques incorporate some weighting procedure into the analysis. Everything we discuss today will only need the TF dtm.
- TF DTM -> Model

Bag of words approach

- Order of words doesn’t matter.
- Each document treated as a “bag of words” without structure.
- Computationally efficient
- Powerful predictive input for a number of applications

CBOW and Skip-gram

- CBOW stands for "continuous bag-of-words"



Reference: Efficient Estimation of Word Representations in Vector Space by Tomas Mikolov, et al.

Basic Topic Modeling

Topic Modeling

- “Topic”: A group of words that share the same context and are likely to co-occur together within one document.
- From Blei (2012: 77) *Topic models are algorithms for discovering the main themes that pervade a large and otherwise unstructured collection of documents.* Large is greater than 100 documents or texts. The number of topics, k, should be much less than the number of documents in your corpus.
- Bag of words approach: no structure accounted for and order not considered

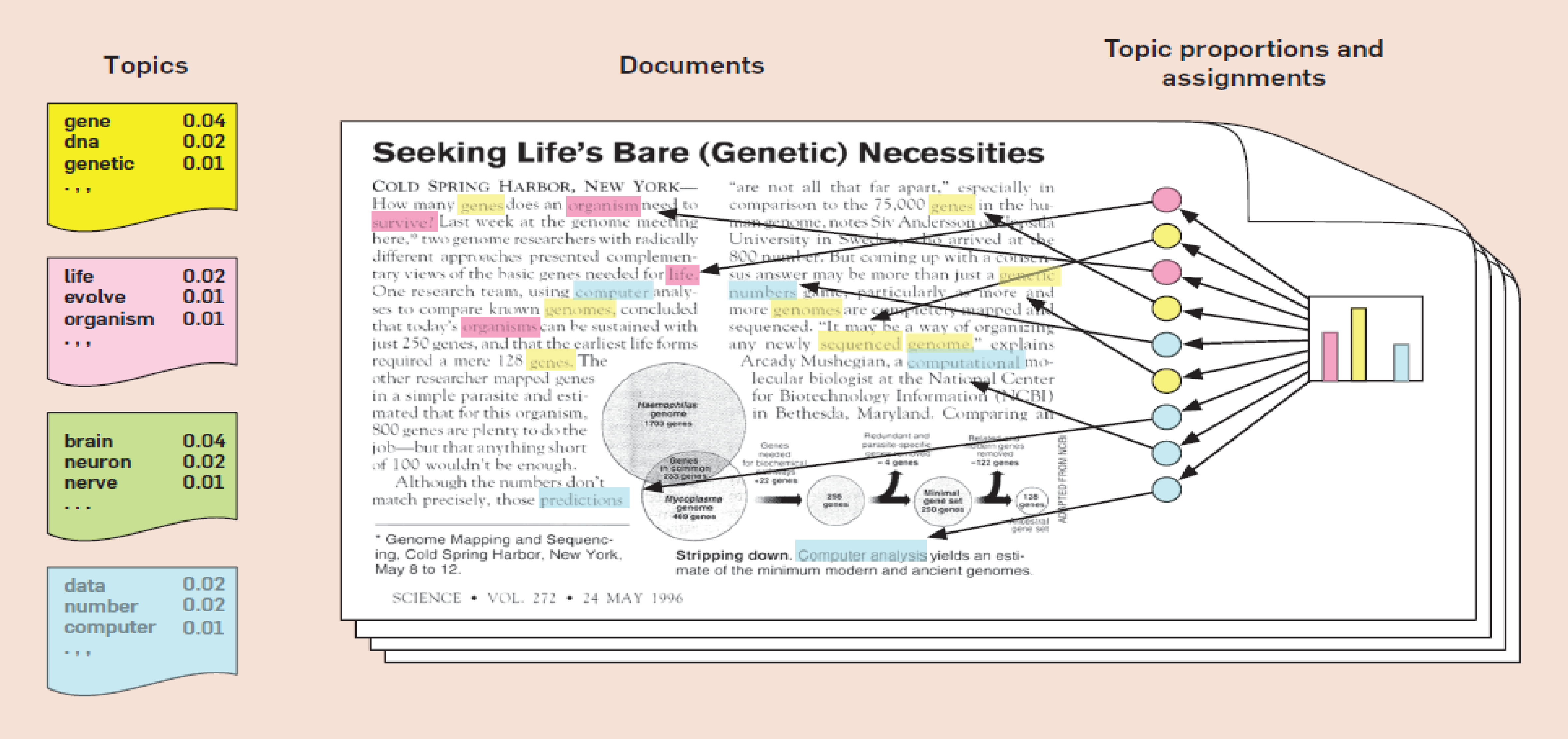
Latent Dirichlet Allocation

- Set of documents that contain different topics.
- We want to estimate these topics in each document, but each topic is made up of words associated with that topic at some probability.
- The topic structure (i.e. the topics in each document and the word probabilities associated with each topic) are hidden [not observed]
- The documents are observed.

Blei (2012)

- 17,000 articles from last 50 years in journal *Science*
- Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.
- *http://dl.acm.org/citation.cfm?id=2133826*

Figure 1. The intuitions behind latent Dirichlet allocation. We assume that some number of "topics," which are distributions over words, exist for the whole collection (far left). Each document is assumed to be generated as follows. First choose a distribution over the topics (the histogram at right); then, for each word, choose a topic assignment (the colored coins) and choose the word from the corresponding topic. The topics and topic assignments in this figure are illustrative—they are not fit from real data. See Figure 2 for topics fit from data.


Topics
gene 0.04
dna 0.02
genetic 0.01
. . .
life 0.02
evolve 0.01
organism 0.01
. . .
brain 0.04
neuron 0.02
nerve 0.01
. . .
data 0.02
number 0.02
computer 0.01
. . .
Documents
Topic proportions and assignments
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
256 genes
Redundant and parasite-specific genes removed −4 genes
Minimal gene set 250 genes
Related and modern genes removed −122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

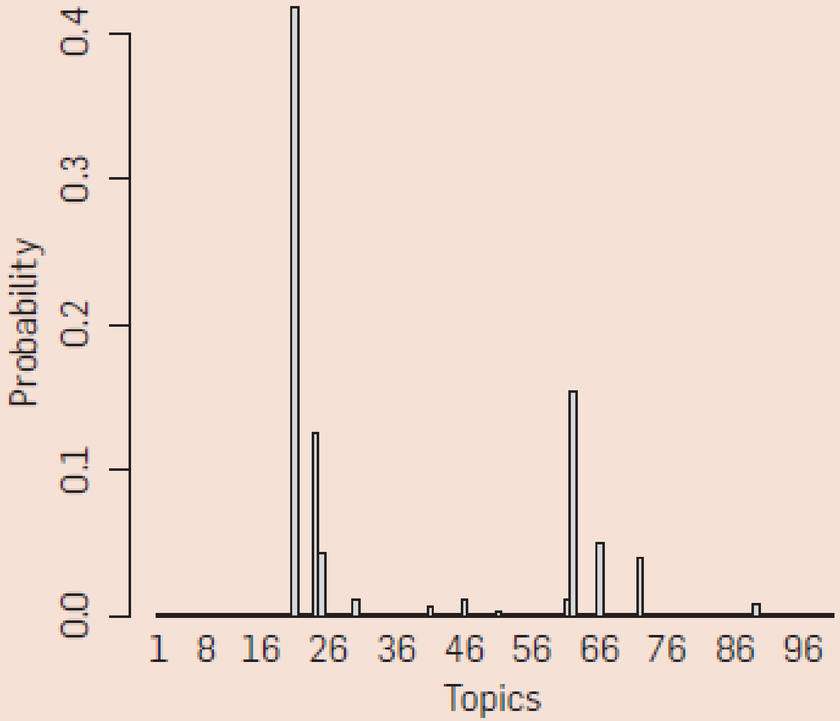
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

Tuning parameters

- http://papers.nips.cc/paper/2070-latent-dirichlet-allocation.pdf
- Setting K: https://cran.r-project.org/web/packages/ldatuning/vignettes/topics.html
- Setting alpha, beta (sparsity of topic model): a low alpha value places more weight on having each document composed of only a few dominant topics (whereas a high value will return many more relatively dominant topics). Similarly, a low beta value places more weight on having each topic composed of only a few dominant words.

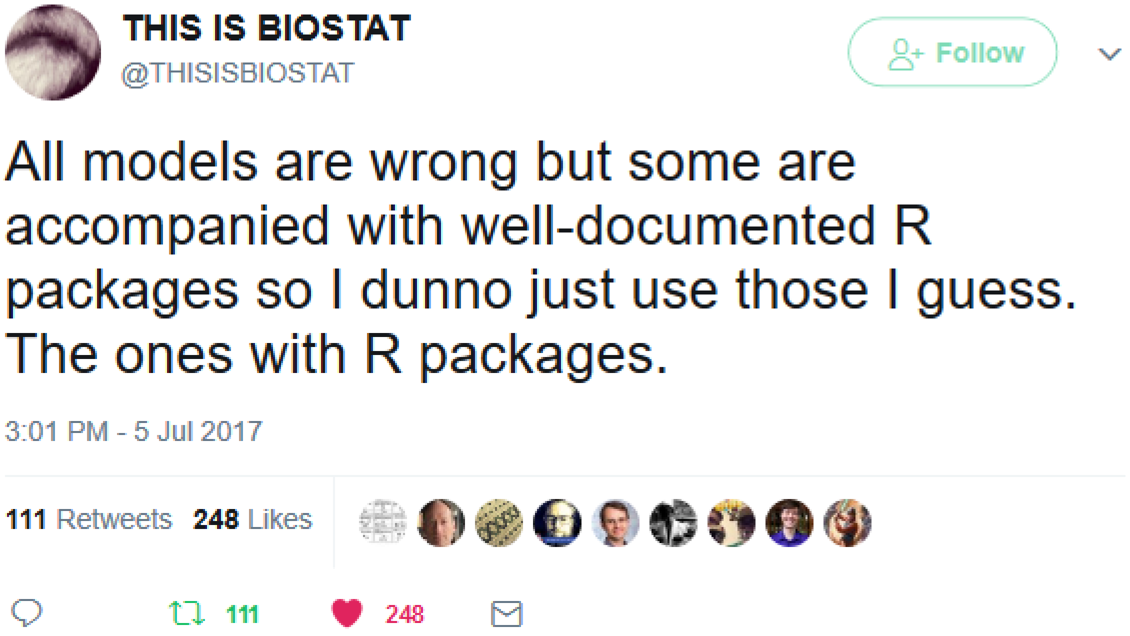
Figure 2. Real inference with LDA. We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.



"Genetics"	"Evolution"	"Disease"	"Computers"
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Advanced Topic Modeling

Structured, Hierarchical and Dynamic Models



THIS IS BIOSTAT
@THISISBIOSTAT

All models are wrong but some are accompanied with well-documented R packages so I dunno just use those I guess. The ones with R packages.

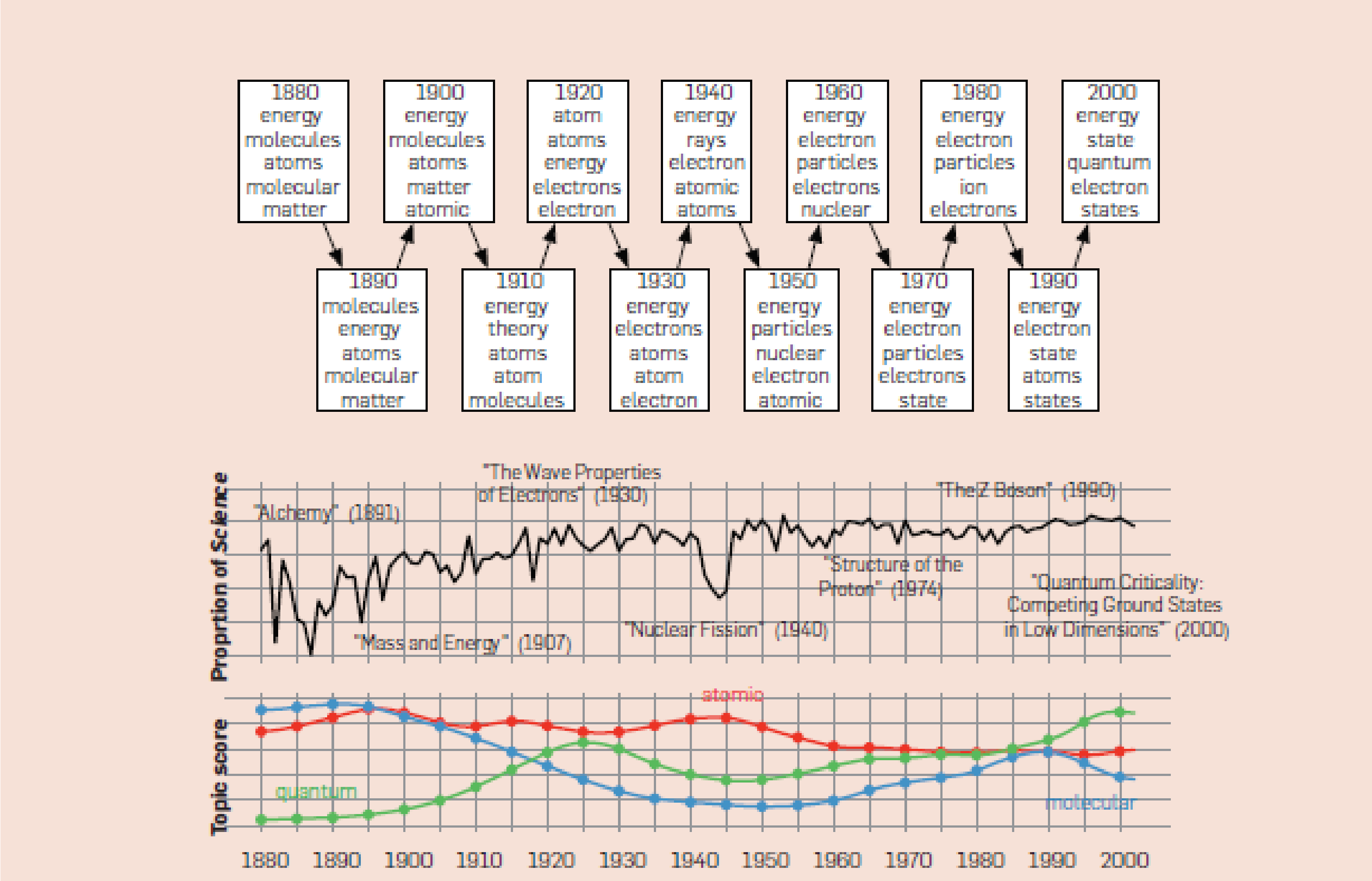
3:01 PM - 5 Jul 2017

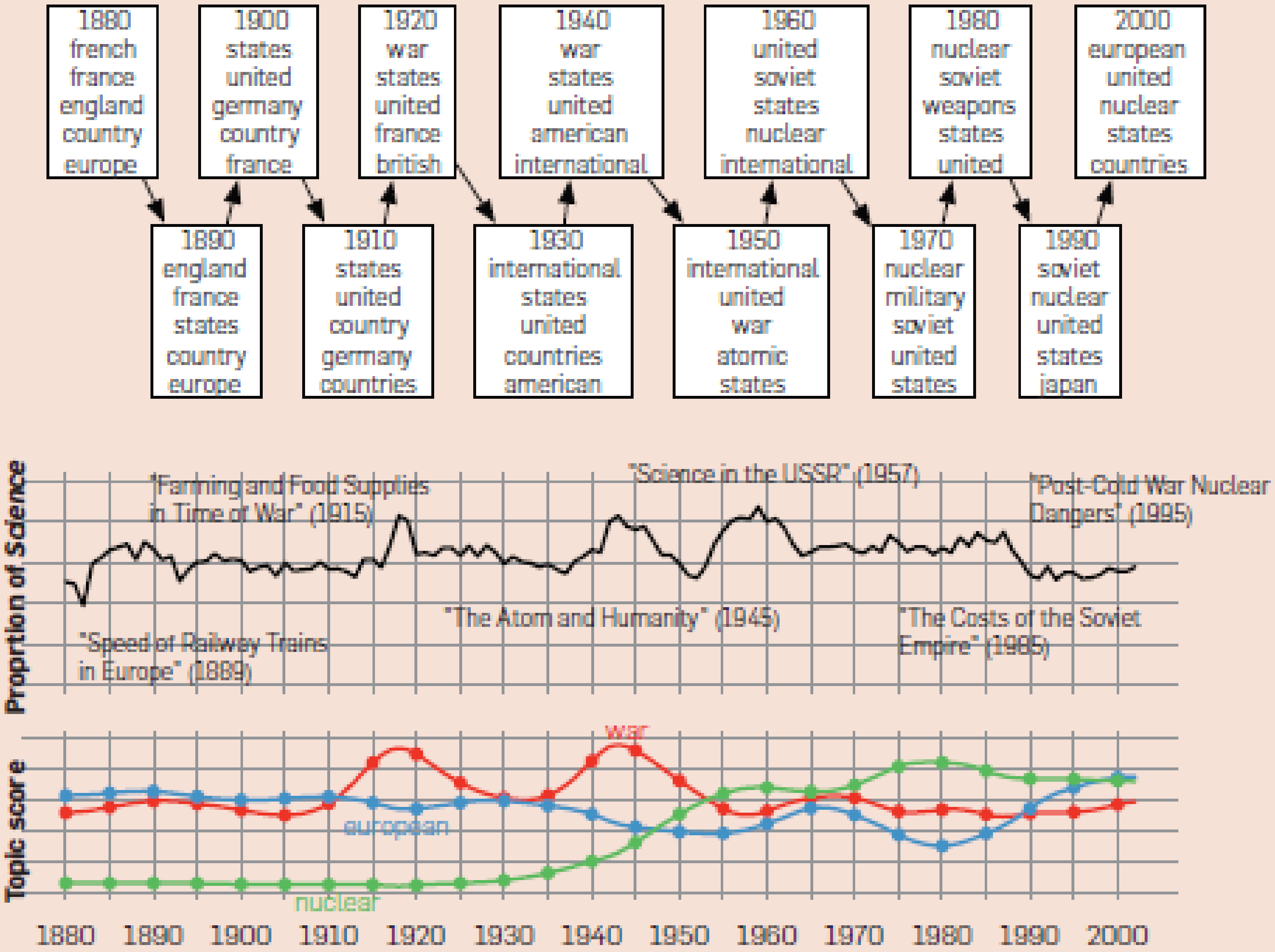
111 Retweets 248 Likes

Original quote: “All models are wrong, some are useful.”
George Box

Dynamic Topic Modeling (i.e. Time)

Figure 5. Two topics from a dynamic topic model. This model was flt to *Science* from 1880 to 2002. We have illustrated the top words at each decade.




1880
french
france
england
country
europe
1890
england
france
states
country
europe
1900
states
united
germany
country
france
1910
states
united
country
germany
countries
1920
war
states
united
france
british
1930
international
states
united
countries
american
1940
war
states
united
american
international
1950
international
united
war
atomic
states
1960
united
soviet
states
nuclear
international
1970
nuclear
military
soviet
united
states
1980
nuclear
soviet
weapons
states
united
1990
soviet
nuclear
united
states
japan
2000
european
united
nuclear
states
countries
Proprtion of Science
"Farming and Food Supplies in Time of War" (1915)
"Science in the USSR" (1957)
"Post-Cold War Nuclear Dangers" (1995)
"Speed of Railway Trains in Europe" (1889)
"The Atom and Humanity" (1945)
"The Costs of the Soviet Empire" (1985)
Topic score
war
european
nuclear
1880
1890
1900
1910
1920
1930
1940
1950
1960
1970
1980
1990
2000

Hierarchical Topic Modeling

- Topics are in a hierarchy of topics

- E.g. food -> {vegetables, meat, dairy}
  - Dairy → {cheese, cream}
- Python implementation.

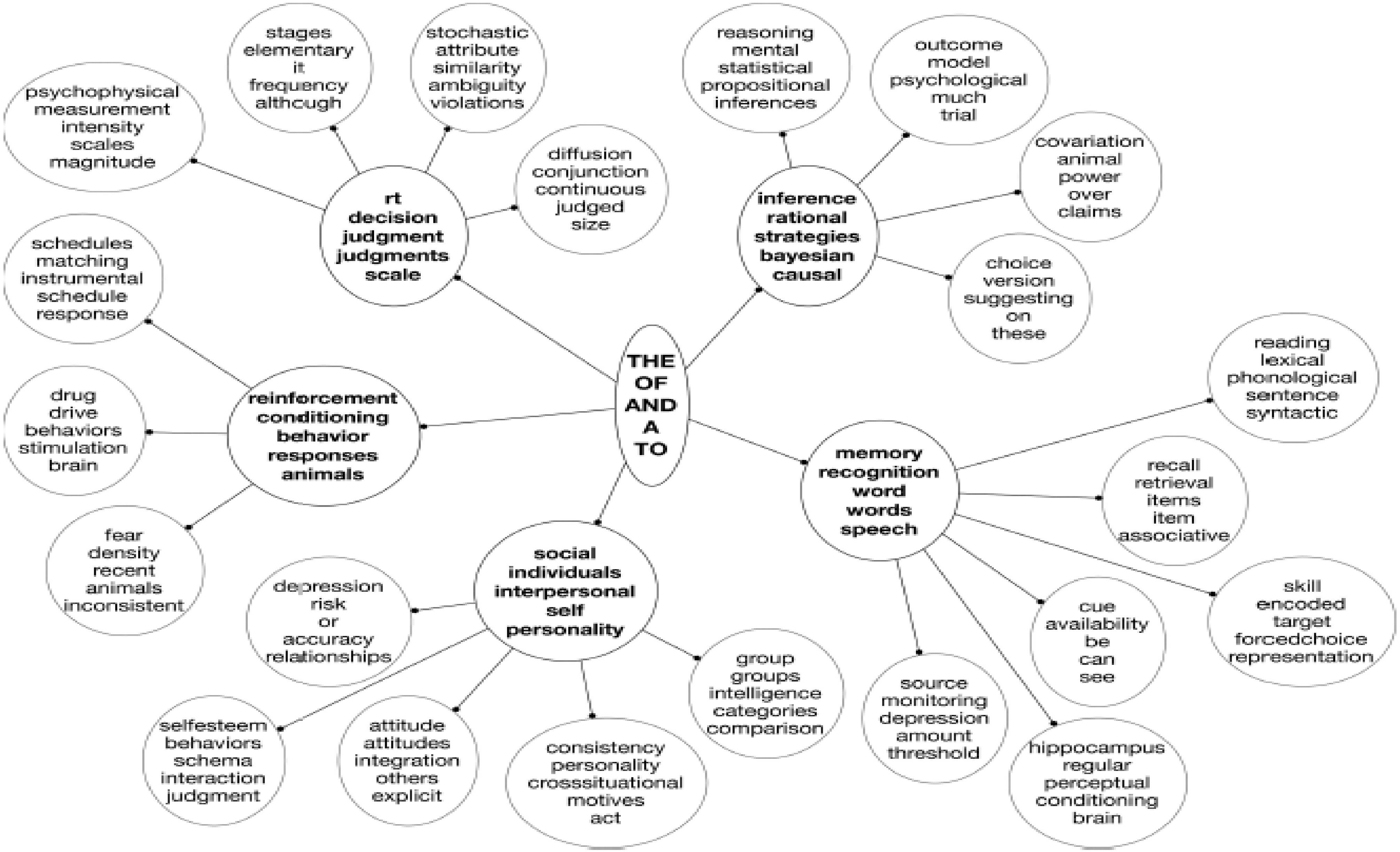

THE OF AND A TO
rt decision judgment judgments scale
psychophysical measurement intensity scales magnitude
stages elementary it frequency althcugh
stochastic attribute similarity ambiguity violations
diffusion conjunction continuous judged size
inference rational strategies bayesian causal
reasoning mental statistical propositional inferences
outcome model psychological much trial
covariation animal power over claims
choice version suggesting on these
reinforcement conditioning behavior responses animals
schedules matching instrumental schedule response
drug drive behaviors stimulation brain
fear density recent animals inconsistent
memory recognition word words speech
reading lexical phonological sentence syntactic
recall retrieval items item associative
skill encoded target forcedchoice representation
cue availability be can see
hippocampus regular perceptual conditioning brain
source monitoring depression amount threshold
social individuals interpersonal self personality
depression risk or accuracy relationships
group groups intelligence categories comparison
consistency personality crosssituational motives act
attitude attitudes integration others explicit
selfesteem behaviors schema interaction judgment

FIG. 7. A portion of the hierarchy learned from the 1,272 abstracts of *Psychological Review* from 1967–2003. The vocabulary was restricted to the 1,971 terms that occurred in more than five documents, yielding a corpus of 136K words. The learned hierarchy, of which only a portion is illustrated, contains 52 topics.

Structured Topic Modeling

- Topics are conditional on predictors
  - Historical Linguistics: Social Characteristics, Time, Variants.
- Topic is now dependent on both the words in a document and the features associated with a document.

Word2Vec

Word Embedding Models – taking into account context.

So Far

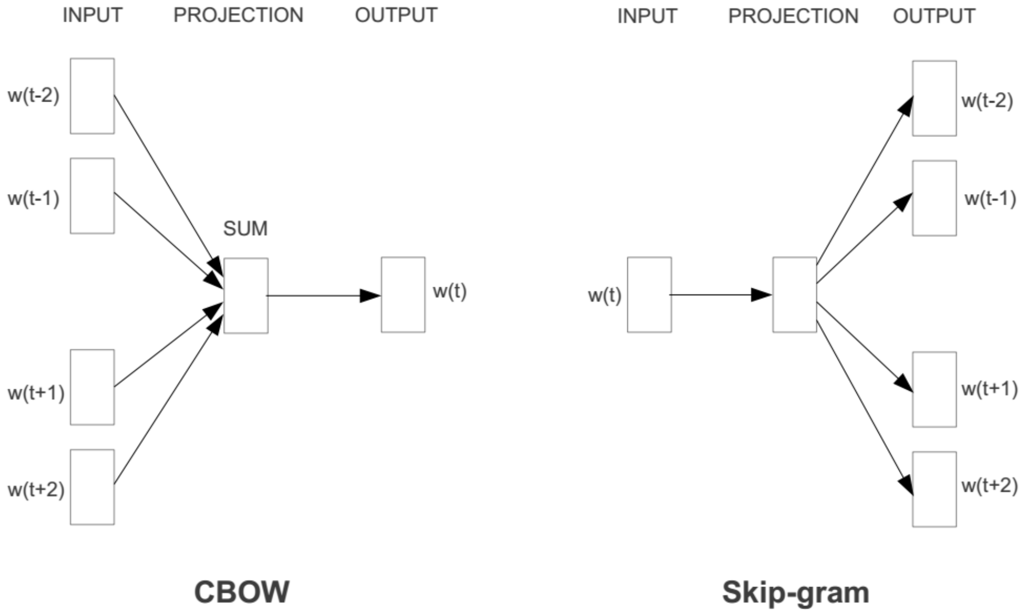
- Topic modeling requires a bag of words approach.
- Syntactic Topic Modeling is a possibility (Blei has some published work)
    - Problems: Too computationally intensive – requires parsing data as a pre-processing step.
    - No widely available implementation (that I know of…)

word2vec Approach to represent the meaning of word

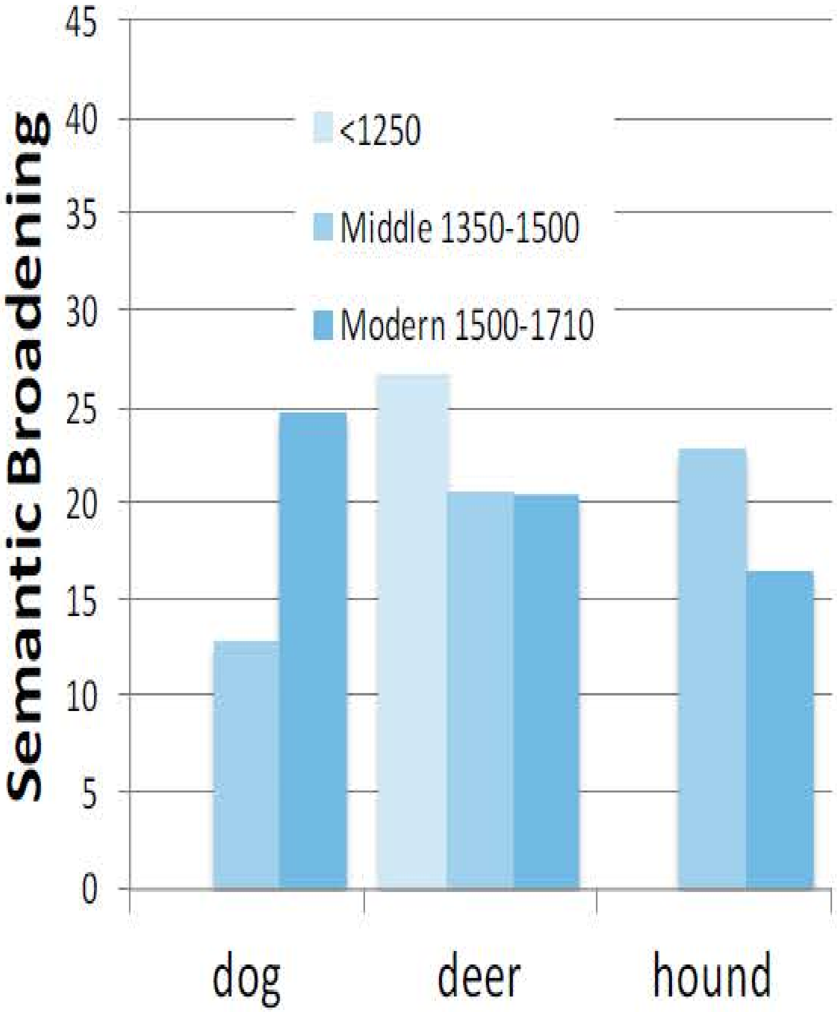
- Represent each word with a low-dimensional vector
- Word similarity = vector similarity
- Key idea: Predict surrounding words of every word
- Faster and can easily incorporate a new sentence/document or add a word to the vocabulary
- Allows context (i.e. surrounding words) to matter in output.

Represent the meaning of **word** – word2vec

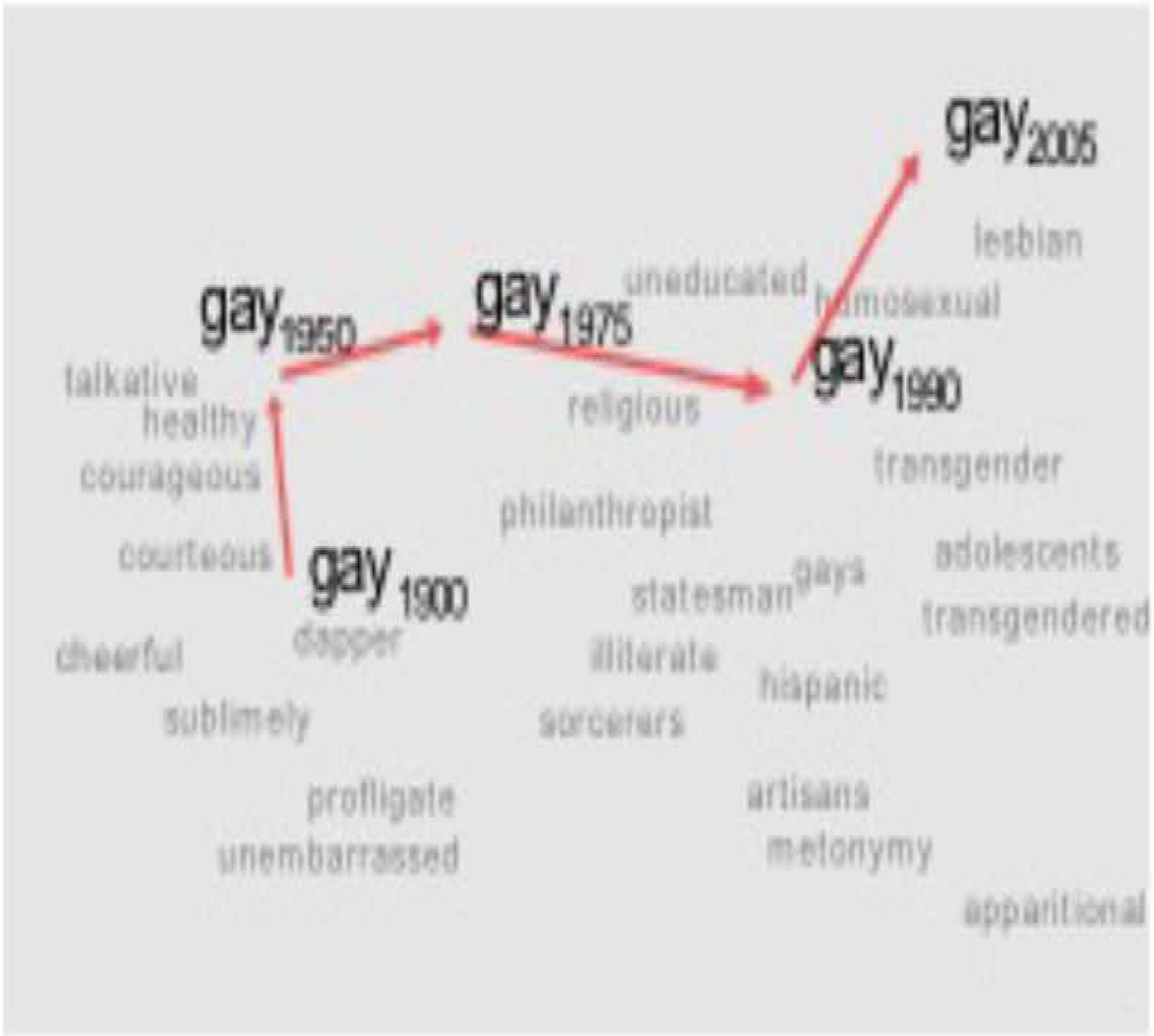
- 2 basic neural network models:
  - Continuous Bag of Word (CBOW): use a window of word to predict the middle word
  - Skip-gram (SG): use a word to predict the surrounding ones in window.



Sagi, Kaufmann Clark 2013



Kulkarni, Al-Rfou, Perozzi, Skiena 2015



Some interesting results

Word Analogies

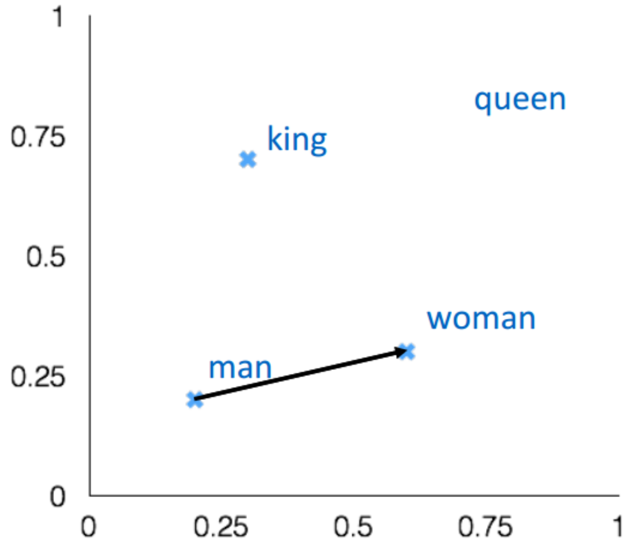
Test for linear relationships, examined by Mikolov et al. (2014)

a:b :: c:? →

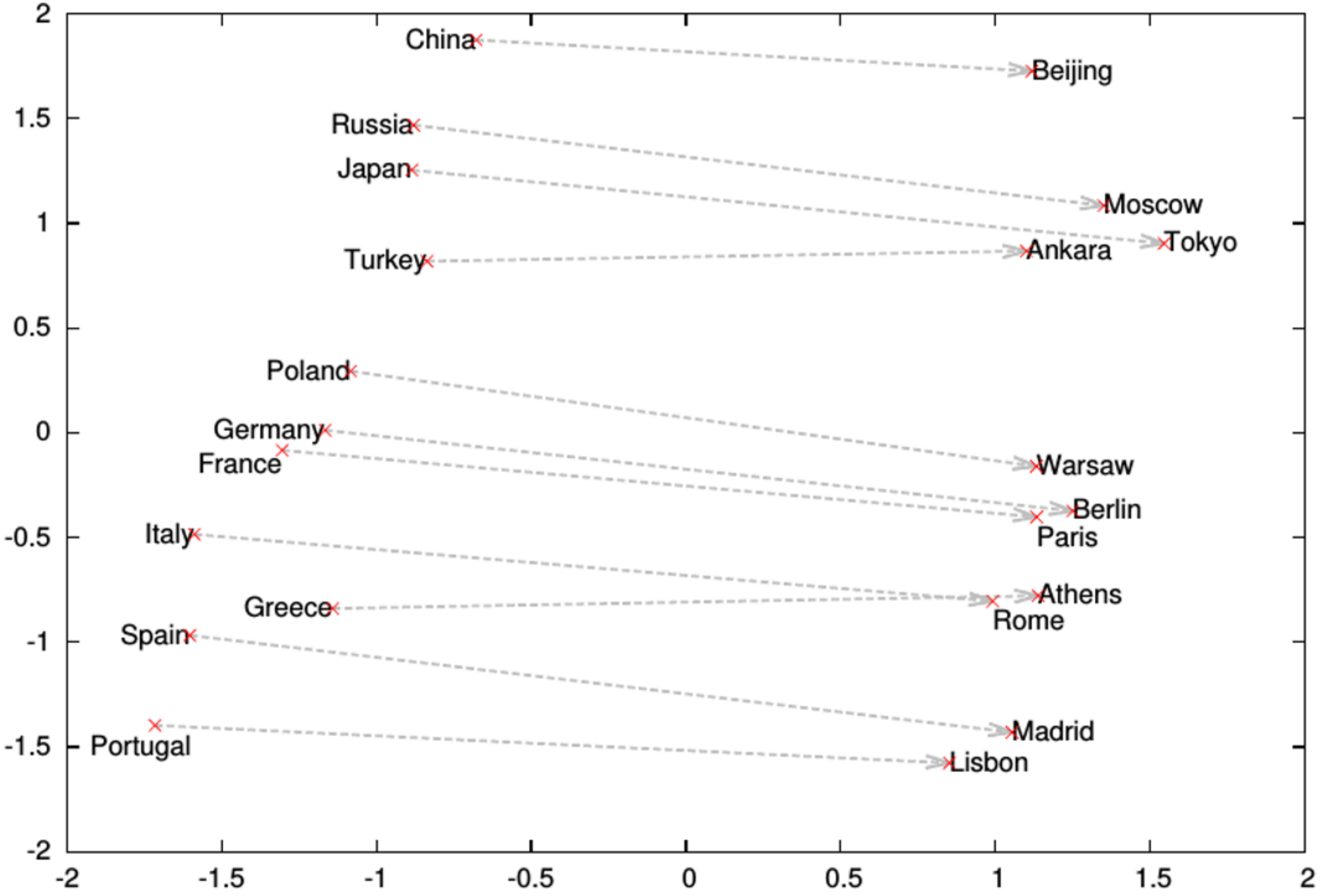
$$d = \arg\max_{x} \frac{(w_b - w_a + w_c)^T w_x}{||w_b - w_a + w_c||}$$

man:woman :: king:?

+	king	[0.30 0.70]
-	man	[0.20 0.20]
+	woman	[0.60 0.30]
	queen	[0.70 0.80]

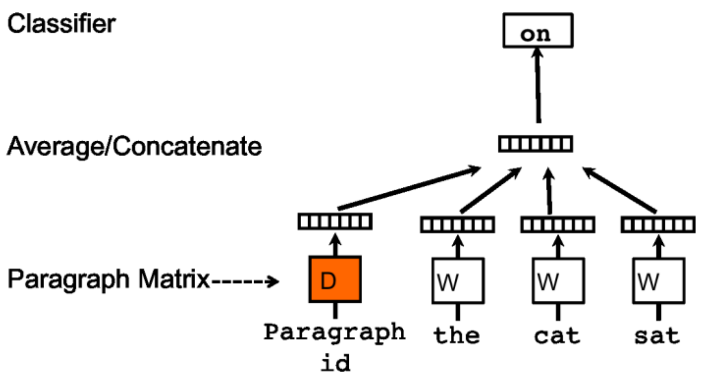


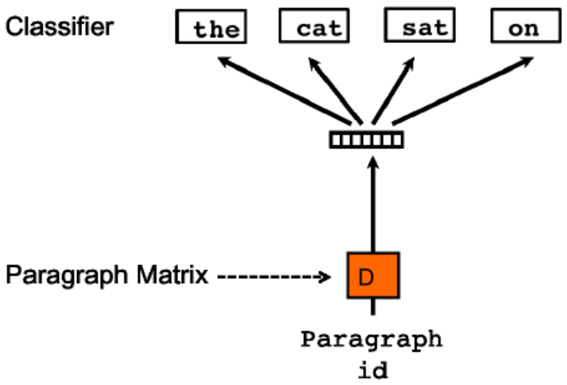
Word analogies



Represent the meaning of **sentence/text**

- Paragraph vector (2014, Quoc Le, Mikolov)
  - Extend word2vec to text level
  - Also two models: add paragraph vector as the input


Classifier
on
Average/Concatenate
Paragraph Matrix----->
D
W
W
W
Paragraph id
the
cat
sat


Classifier
the
cat
sat
on
Paragraph Matrix ---------->
D
Paragraph id