Text Mining

Vector Space Model & Topic Modeling

Qixiang Fang

Based on Ayoub Bagheri's slides

Last week

- Neural networks
- Feed-forward neural networks
- Convolutional neural networks
- Estimation and optimization

Today

- Text mining
- Pre-processing text data
- Vector space model
  - Bag-of-Words
  - Word embedding (next week)
- Topic modeling

Introduction

Text mining

- Hearst (1999): “The discovery by computer of **new, previously unknown information**, by automatically extracting information from different written resources”.
- Wikipedia: Text mining, text data mining (TDM) or text analytics is the process of deriving high-quality information from text.

ChatGPT

ChatGPT 4o

What is text mining?

Text mining, also known as **text data mining** or **text analytics**, is the process of extracting meaningful information, patterns, and insights from unstructured text data. It involves using computational techniques to analyze text documents, emails, social media posts, or any other form of textual information to uncover trends, relationships, and actionable insights.

Why text mining?

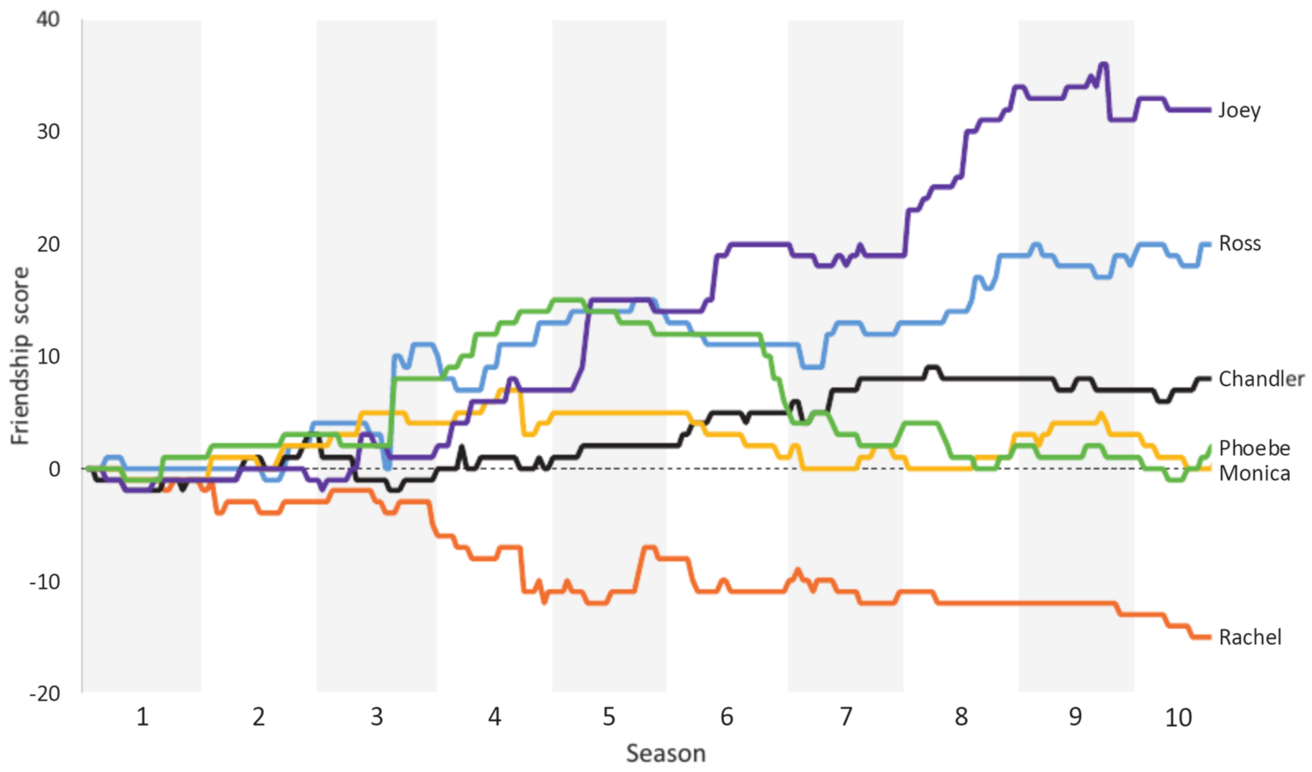
- **Text data is everywhere**, websites (e.g., news), social media (e.g., X), databases (e.g., doctors' notes), digital scans of printed materials, …
- A lot of world's data is in **unstructured text format**

Applications of Text Mining:

- **Business Intelligence**: Analyzing customer feedback, reviews, and survey responses.
- **Healthcare**: Extracting insights from medical records or research articles.
- **Social Media Monitoring**: Understanding public sentiment and trends.
- **Legal and Regulatory Compliance**: Analyzing legal documents for compliance.
- **Academic Research**: Discovering trends in scientific literature.

Who was the best Friend?

https://rss.onlinelibrary.wiley.com/doi/epdf/10.1111/1740-9713.01574

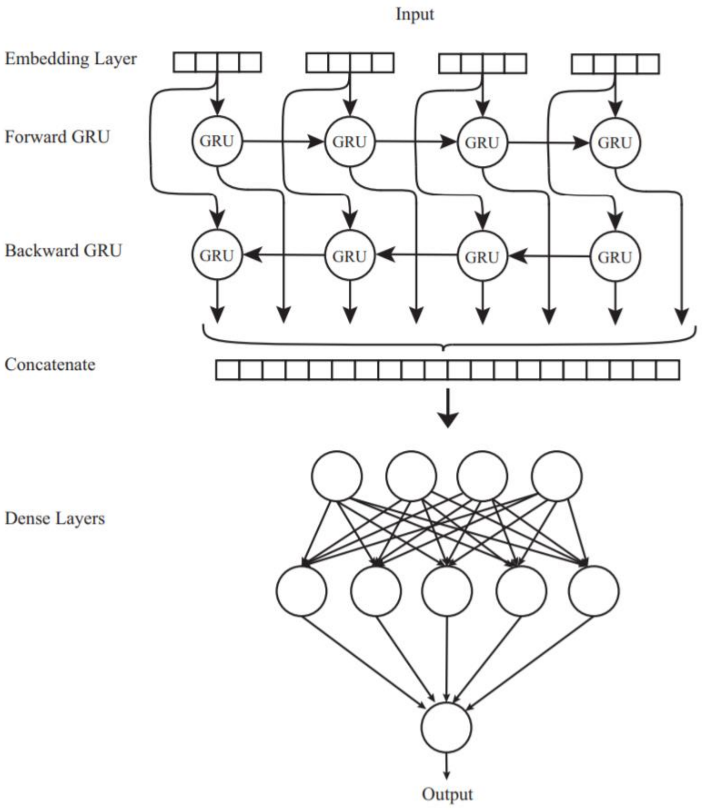


Did a poet with donkey ears write the oldest anthem in the world?

https://dh2017.adho.org/abstracts/079/079.pdf

Automatic detection of disease codes in cardiology discharge letters

https://www.nature.com/articles/s41746-021-00404-9



Box 1: An example of a Dutch discharge letter from the dataset

Bovengenoemde patiënt was opgenomen op <DATUM-1> op de <PERSOON-1> voor het specialisme Cardiologie.
***Reden van opname** STEMI inferior*
***Cardiale voorgeschiedenis**. Blanco*
***Cardiovasculaire risicofactoren**: Roken(-) Diabetes(-) Hypertensie(?) Hypercholesterolemie (?)*
***Anamnese**. Om 18.30 pijn op de borst met uitstraling naar de linkerarm, zweten, misselijk. Ambulance gebeld en bij aansluiten monitor beeld van acuut onderwandinfarct.*
AMBU overdracht. 500 mg aspegic iv, ticagrelor 180 mg oraal, heparine, zofran eenmalig, 3× NTG spray. HD stabiel gebleven.Medicatie bij presentatie.Geen.
***Lichamelijk onderzoek**. Grauw, vegetatief, Halsvenen niet gestuwd. Cor s1 s2 geen souffles.Pulm schoon. Extr warm en slank.*
***Aanvullend onderzoek**. AMBU ECG: Sinusritme, STEMI inferior III)II C/vermoedelijk RCA.*
Coronair angiografie. (…). Conclusie angio: 1-vatslijden..PCI
Conclusie en beleid
Bovengenoemde <LEEFTIJD-1> jarige man, blanco cardiale voorgeschiedenis, werd gepresenteerd vanwege een STEMI inferior waarvoor een spoed PCI werd verricht van de mid-RCA. Er bestaan geen relevante nevenletsels. Hij kon na de procedure worden overgeplaatst naar de CCU van het <INSTELLING-2>…Dank voor de snelle overname…Medicatie bij overplaatsing. Acetylsalicylzuur dispertablet 80 mg; oraal; 1× per dag 80 milligram; <DATUM-1>. Ticagrelor tablet 90 mg; oraal; 2× per dag 90 milligram; <DATUM-1>. Metoprolol tablet 50 mg; oraal; 2× per dag 25 milligram; <DATUM-1> .Atorvastatine tablet 40 mg (als ca-zout-3-water); oraal; 1× per dag 40 milligram; <DATUM-1>
Samenvatting
Hoofddiagnose: STEMI inferior wv PCI RCA. Geen nevenletsels. Nevendiagnoses: geen.
Complicaties: geen Ontslag naar: CCU <INSTELLING-2>.

Pre-processing Text Data

Text preprocessing

- is an approach for cleaning text data and removing noises in the data.

Challenges

High dimensional data

- All possible words & phrases

Complex & subtle relationships in text

- ”Jumbo merges with Hema”
- “Jumbo is bought by Hema”

Ambiguity & context sensitivity

- car = automobile = vehicle
- kapsalon (hairdresser) or kapsalon (fast food)

Homographs: same words can mean different things

- Bat (sports, animal, …)

Synonyms

Misspellings

Abbreviations

Negations

Spelling variations

LANGUAGE!

Typical steps

- Tokenization (“text”, “ming”, “is”, “the”, “best” , “!”)
- Stemming (“running”→“run”) or Lemmatization (“were”→“is”)
- Lowercasing (“And”→“and”)
- Stopword removal (“text ming is best!”)
- Punctuation removal (“text ming is the best”)
- Number removal (“infomda 2”→“infomda”)
- Spell correction (“ming”→“mining”)

Not all of these are appropriate at all times!

Example

Text mining is to identify useful information.

↓ Tokenization

‘Text’, ‘mining’, ‘is’, ‘to’, ‘identify’, ‘useful’, ‘information’, ‘.’

↓ Stemming

‘text’, ‘mine’, ‘is’, ‘to’, ‘identify’, ‘use’, ‘inform’, ‘.’

↓ Bigrams

‘text mine’, ‘mine is’, ‘is to’, ‘to identify’, ‘identify use’, ‘use inform’, ‘inform .’

↓ Stopwords & punctuations

‘text mine’, ‘to identify’, ‘identify use’, ‘use inform’

↓ Vectorization

Vector Space Model

Vector Space Model

Basic idea

- Text is “unstructured data”
- How do we get to something structured that we can compute with?
- **Text must be represented somehow**
- Represent the text as something that makes sense to a computer

Vector space model

文A 17 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

Vector space model or **term vector model** is an algebraic model for representing text documents (or more generally, items) as vectors such that the distance between vectors represents the relevance between the documents. It is used in information

Vector space model

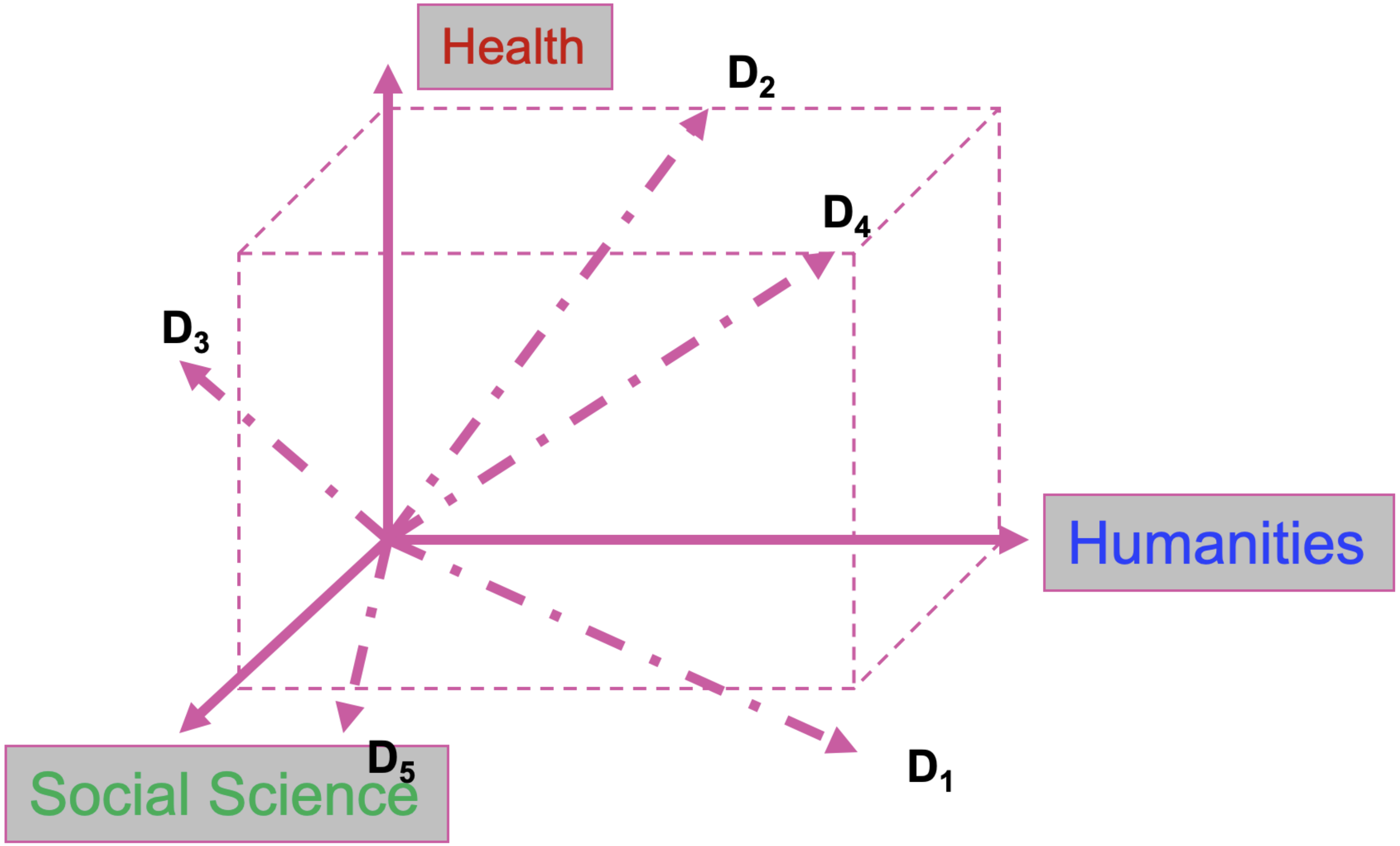
- Each document is represented as a vector
  - Each dimension corresponds to some concept
  - Each element (i.e., scalar) in the vector corresponds to a concept weight
  - A vector can be high-dimensional (e.g., > 10,000)

Simplest vector space model

- Documents are represented as vectors of terms
  - Typically, terms are single words, keywords, n-grams, or phrases
- Each dimension (concept) corresponds to a separate term

$$d = (w_1, \ldots, w_n)$$

An illustration



Vectorization

- The process of converting text into numbers is called **vectorization**

- Distance between the vectors in this concept space
  - Relationship among documents

VSM representations

1/ Bag-of-Words → TF, TFiDF → High Dimensional, Sparse

2/ Topics → Topic modeling, Clustering → Low Dimensional

3/ Embeddings → Word2Vec, fasttext, transformers → Dense representation, Distributional hypothesis

Bag-of-Words

- **Terms** are words (more generally we can use n-grams)
- **Weights** capture the occurrences/relevance of the terms in the document
  - Binary
  - Term Frequency (TF)
  - Term Frequency inverse Document Frequency (TFiDF)

Example (TF/binary)

Doc1: Text mining is to identify useful information.

Doc2: Useful information is mined from text.

Doc3: Apple is delicious.

Document-Term matrix (DTM):

	text	information	identify	mining	mined	is	useful	to	from	apple	delicious
Doc1	1	1	1	1	0	1	1	1	0	0	0
Doc2	1	1	0	0	1	1	1	0	1	0	0
Doc3	0	0	0	0	0	1	0	0	0	1	1

DTM in R

```r
library(tm)
# prepare your data
df <- data.frame(document = c("Text mining is to identify useful information.",
                              "Useful information is mined from text.",
                              "Apple is delicious."))
corpus <- VCorpus(VectorSource(df$document))

# convert to dtm
dtm <- DocumentTermMatrix(corpus,
            control = list(wordLengths = c(1, Inf),
             removePunctuation = TRUE))
```

DTM in R

```
> inspect(dtm)

<<DocumentTermMatrix (documents: 3, terms: 11)>>
Non-/sparse entries: 16/17
Sparsity           : 52%
Maximal term length: 11
Weighting          : term frequency (tf)
Sample             :
    Terms
Docs apple delicious from identify information is mined mining text useful
   1     0         0    0        1           1  1     0      1    1      1
   2     0         0    1        0           1  1     1      0    1      1
   3     1         1    0        0           0  1     0      0    0      0
```

TFiDF

- A term is more discriminative if it occurs a lot but only in fewer documents
- Relative term frequency: Let $n_{d,t}$ denote the number of times the term t appears in the document d.

$$TF_{d,t} = \frac{n_{d,t}}{\sum_i n_{d,i}}$$

- Let N denote the number of documents and N_t denote the number of documents containing term t.

$$IDF_t = log(\frac{N}{N_t})$$

TFiDF weight:

$$w_{d,t} = TF_{d,t} \cdot IDF_t$$

DTM in R (TFiDF)

```
dtm_tfidf <- DocumentTermMatrix(corpus,
                                control = list(weighting = weightTfIdf,
                                removePunctuation = TRUE,
                                wordLengths = c(1, Inf)))
```

DTM in R (TFiDF)

```
> inspect(dtm_tfidf)

<<DocumentTermMatrix (documents: 3, terms: 11)>>
Non-/sparse entries: 13/20
Sparsity           : 61%
Maximal term length: 11
Weighting          : term frequency - inverse document frequency (normalized) (tf-idf)
Sample             :
    Terms
Docs     apple delicious      from  identify information     mined    mining       text        to
useful
   1 0.0000000 0.0000000 0.0000000 0.2264232  0.08356607 0.0000000 0.2264232 0.08356607 0.2264232
0.08356607
   2 0.0000000 0.0000000 0.2641604 0.0000000  0.09749375 0.2641604 0.0000000 0.09749375 0.0000000
0.09749375
   3 0.5283208 0.5283208 0.0000000 0.0000000  0.00000000 0.0000000 0.0000000 0.00000000 0.0000000
0.00000000
```

N-grams in R

```
library(RWeka)

tokenizer <- function(x) {
  NGramTokenizer(x, Weka_control(min = 1, max = 2))
}
dtm_ngram <- DocumentTermMatrix(corpus,
           control = list(tokenize = tokenizer,
                    wordLengths = c(1, Inf)))

> colnames(dtm_ngram)
 [1] "apple"           "apple is"     "delicious"      "from"    "from text"     "identify"
 [7] "identify useful“ "information“ "information is" "is"      "is delicious" "is mined"
[13] "is to"           "mined"        "mined from"     "mining“ "mining is"    "text"
[19] "text mining"     "to"           "to identify"    "useful" "useful information"
```

Bag of words representations are often high dimensional!

Topic Modeling

Topic modeling?

- Statistical model for discovering the abstract "topics" that occur in a collection of documents.
- The goal is to uncover hidden thematic structures in large collections of texts.
  - **Latent Dirichlet Allocation (LDA)**
  - Non-negative Matrix Factorization (NMF)
  - Latent Semantic Analysis (LSA)

Applications

- Dimensionality reduction
- Clustering
- Many other text mining tasks
  - Tracking topic changes over time
  - Uncovering new topics

Latent Dirichlet Allocation

What is a topic in LDA?

- A probabilistic distribution over words
- A broad concept/theme, semantically coherent, which is hidden in documents
  - e.g., politics; sports; technology; entertainment; education etc.

What is a topic in LDA?

Steyvers & Griffiths, 2007

Topic 247

word	prob.
DRUGS	.069
DRUG	.060
MEDICINE	.027
EFFECTS	.026
BODY	.023
MEDICINES	.019
PAIN	.016
PERSON	.016
MARIJUANA	.014
LABEL	.012
ALCOHOL	.012
DANGEROUS	.011
ABUSE	.009
EFFECT	.009
KNOWN	.008
PILLS	.008

Topic 5

word	prob.
RED	.202
BLUE	.099
GREEN	.096
YELLOW	.073
WHITE	.048
COLOR	.048
BRIGHT	.030
COLORS	.029
ORANGE	.027
BROWN	.027
PINK	.017
LOOK	.017
BLACK	.016
PURPLE	.015
CROSS	.011
COLORED	.009

Topic 43

word	prob.
MIND	.081
THOUGHT	.066
REMEMBER	.064
MEMORY	.037
THINKING	.030
PROFESSOR	.028
FELT	.025
REMEMBERED	.022
THOUGHTS	.020
FORGOTTEN	.020
MOMENT	.020
THINK	.019
THING	.016
WONDER	.014
FORGET	.012
RECALL	.012

Topic 56

word	prob.
DOCTOR	.074
DR.	.063
PATIENT	.061
HOSPITAL	.049
CARE	.046
MEDICAL	.042
NURSE	.031
PATIENTS	.029
DOCTORS	.028
HEALTH	.025
MEDICINE	.017
NURSING	.017
DENTAL	.015
NURSES	.013
PHYSICIAN	.012
HOSPITALS	.011

Figure 1. An illustration of four (out of 300) topics extracted from the TASA corpus.

Document as a mixture of topics

[Criticism of government response to the hurricane primarily consisted of criticism of its response to the approach of the storm and its aftermath, specifically in the delayed response] to the [flooding of New Orleans. … 80% of the 1.3 million residents of the greater New Orleans metropolitan area evacuated] …[Over seventy countries pledged monetary donations or other assistance]. …

Topic θ_1

government 0.3
response 0.2
...

Topic θ_2

city 0.2
new 0.1
orleans 0.05
...

...

Topic θ_m

donate 0.1
relief 0.05
help 0.02
...

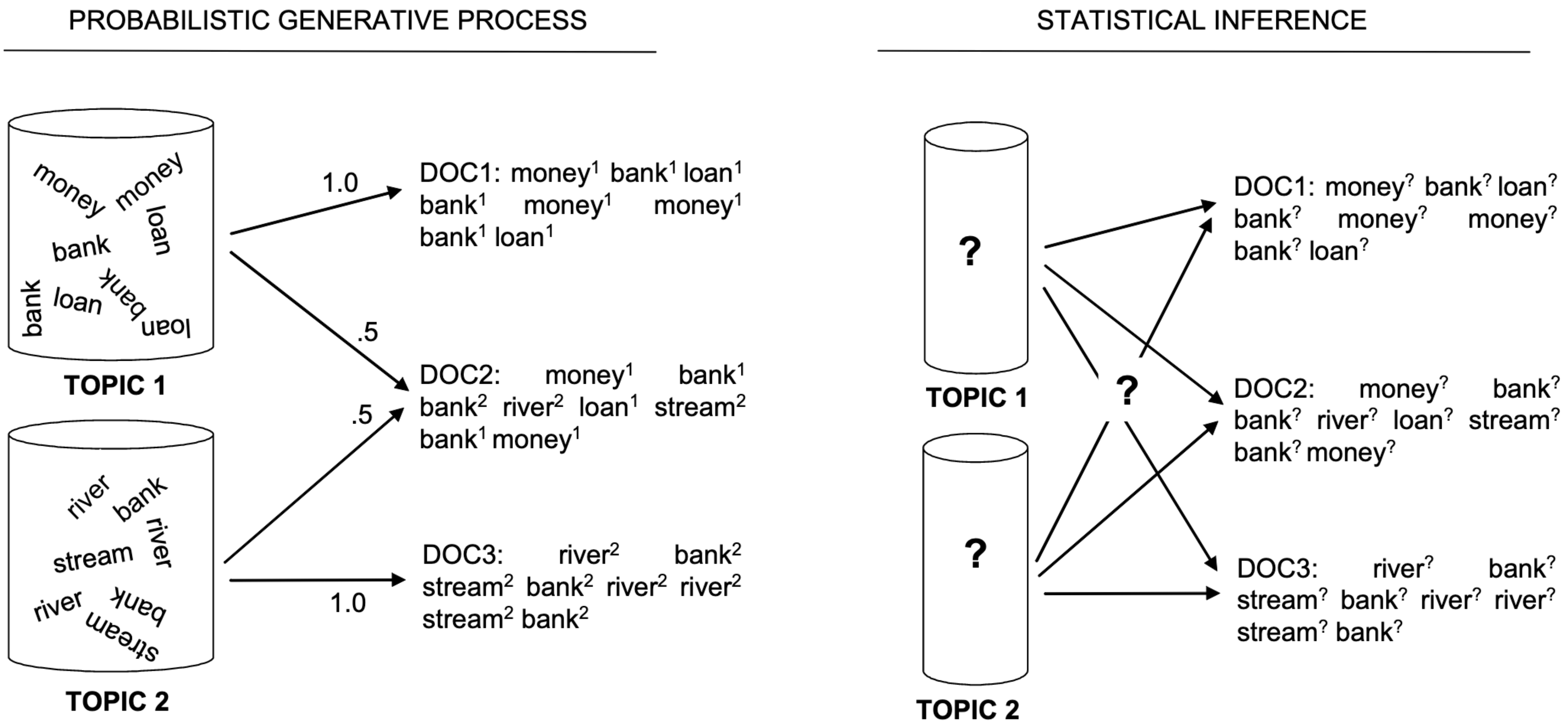
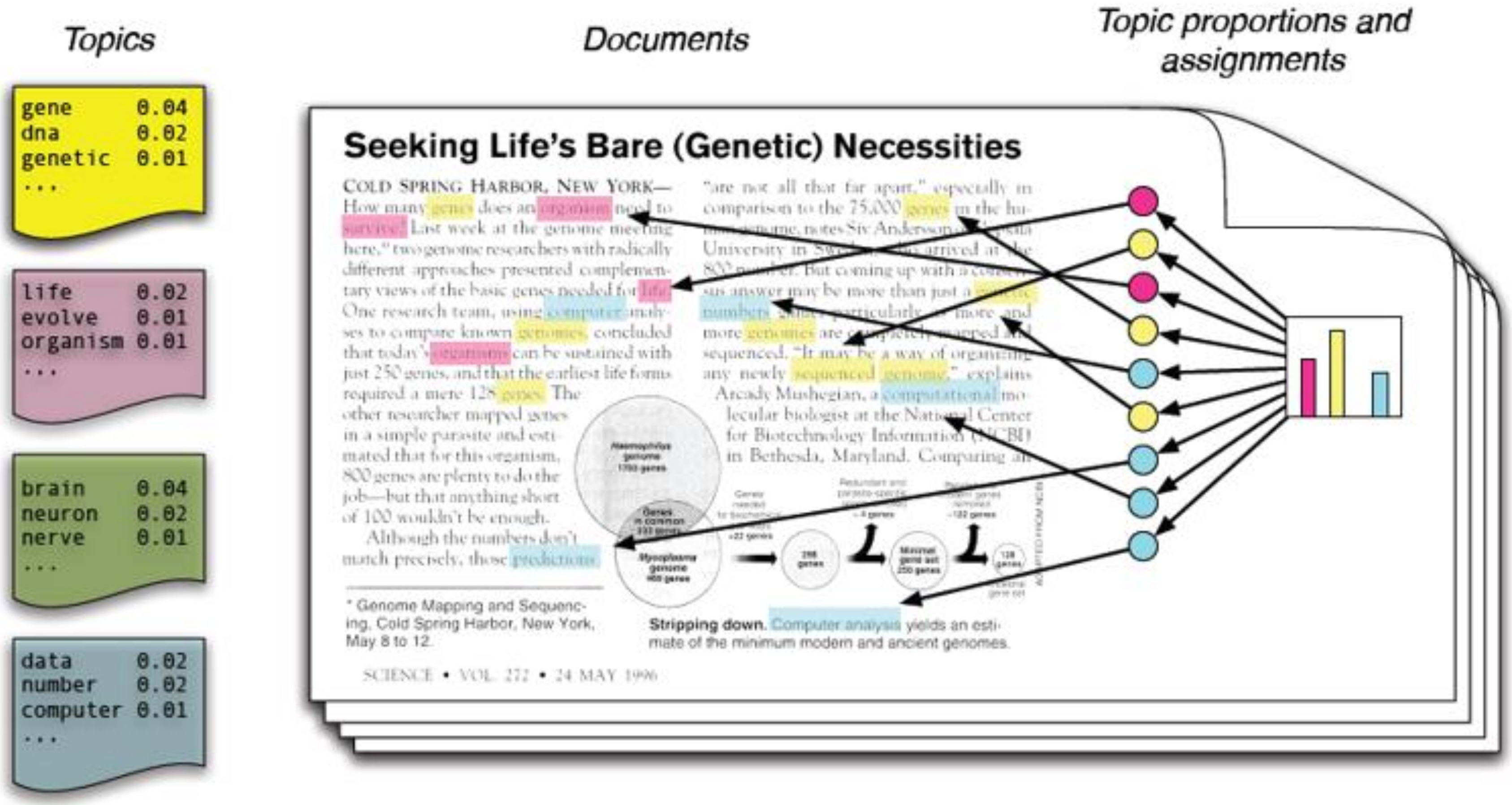


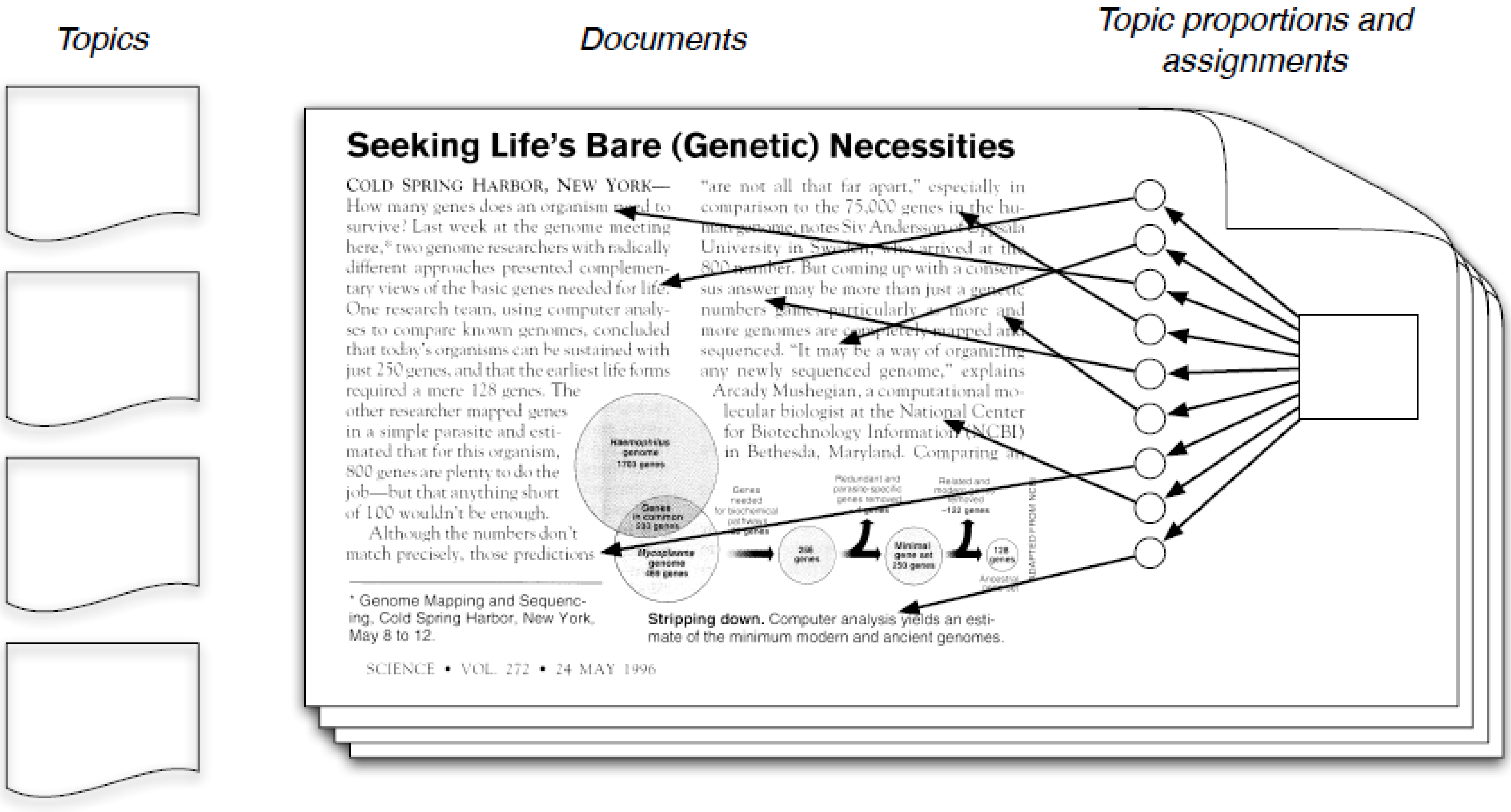
Figure 2. Illustration of the generative process and the problem of statistical inference underlying topic models

The goal



Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *the Journal of machine Learning research*, *3*, 993-1022.
https://dl.acm.org/doi/pdf/10.5555/944919.944937

Reality


Topics
Documents
Topic proportions and assignments
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
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 468 genes
Minimal gene set 250 genes
128 genes
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

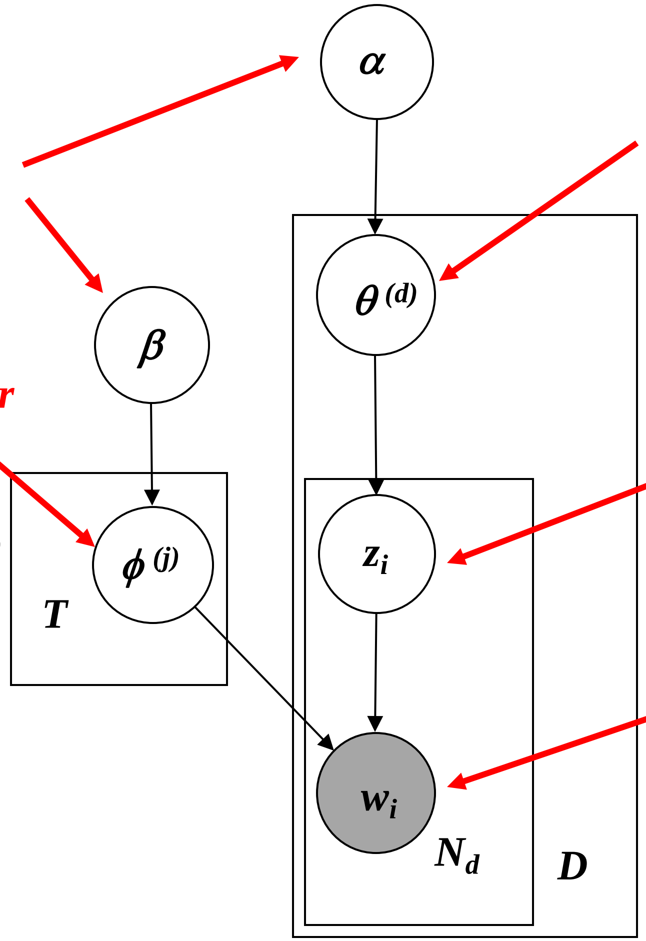
General idea of LDA

- **Key concepts**
  - **Topic**: a probabilistic distribution over words of a fixed vocab
  - **Document**: a mixture of topics
    - First, sample topics from some prior distribution
    - Second, sample words from the selected topics' distributions
- **Modelling**
  - Fit LDA to the data
    - Compare the generated documents to the actual documents
    - Improve through iterations
  - Answer topic-related questions by computing various kinds of posterior distributions
    - e.g., p(sentiment(e.g., "happy") | topic)

LDA graphical model

Dirichlet priors

distribution over topics
for each document



$\theta^{(d)} \sim \text{Dirichlet}(\alpha)$

distribution over words for
each topic

(same as θ_j on the previous slides)

$\phi^{(j)} \sim \text{Dirichlet}(\beta)$

topic assignment
for each word

$z_i \sim \text{Discrete}(\theta^{(d)})$

word generated from
assigned topic

$w_i \sim \text{Discrete}(\phi^{(z_i)})$

Illustration of Dirichlet distribution

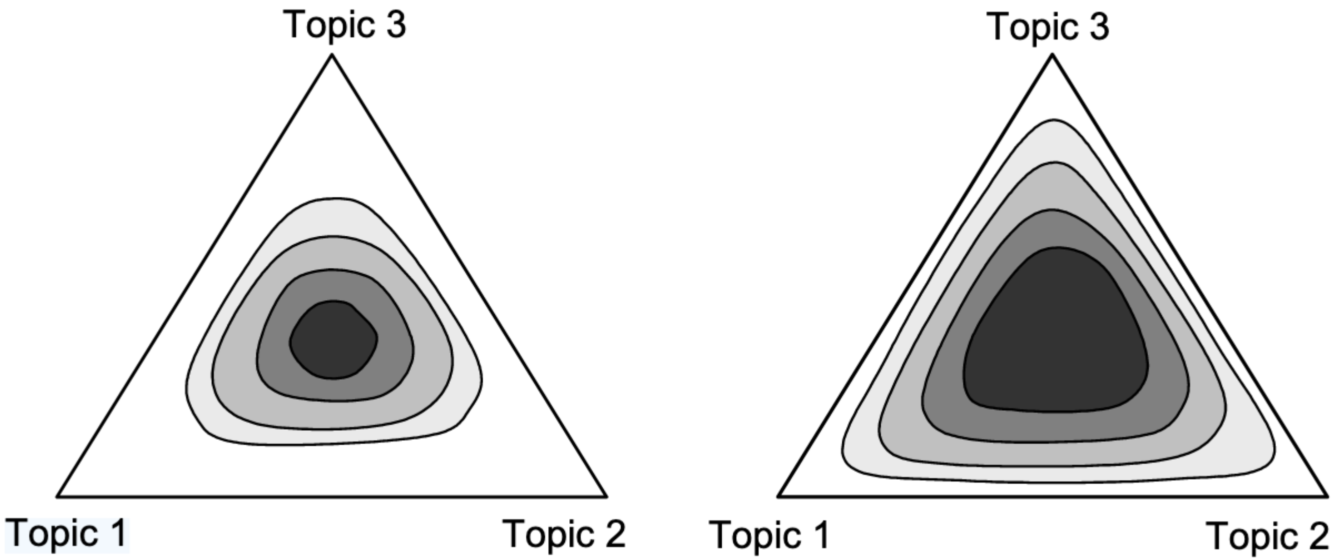
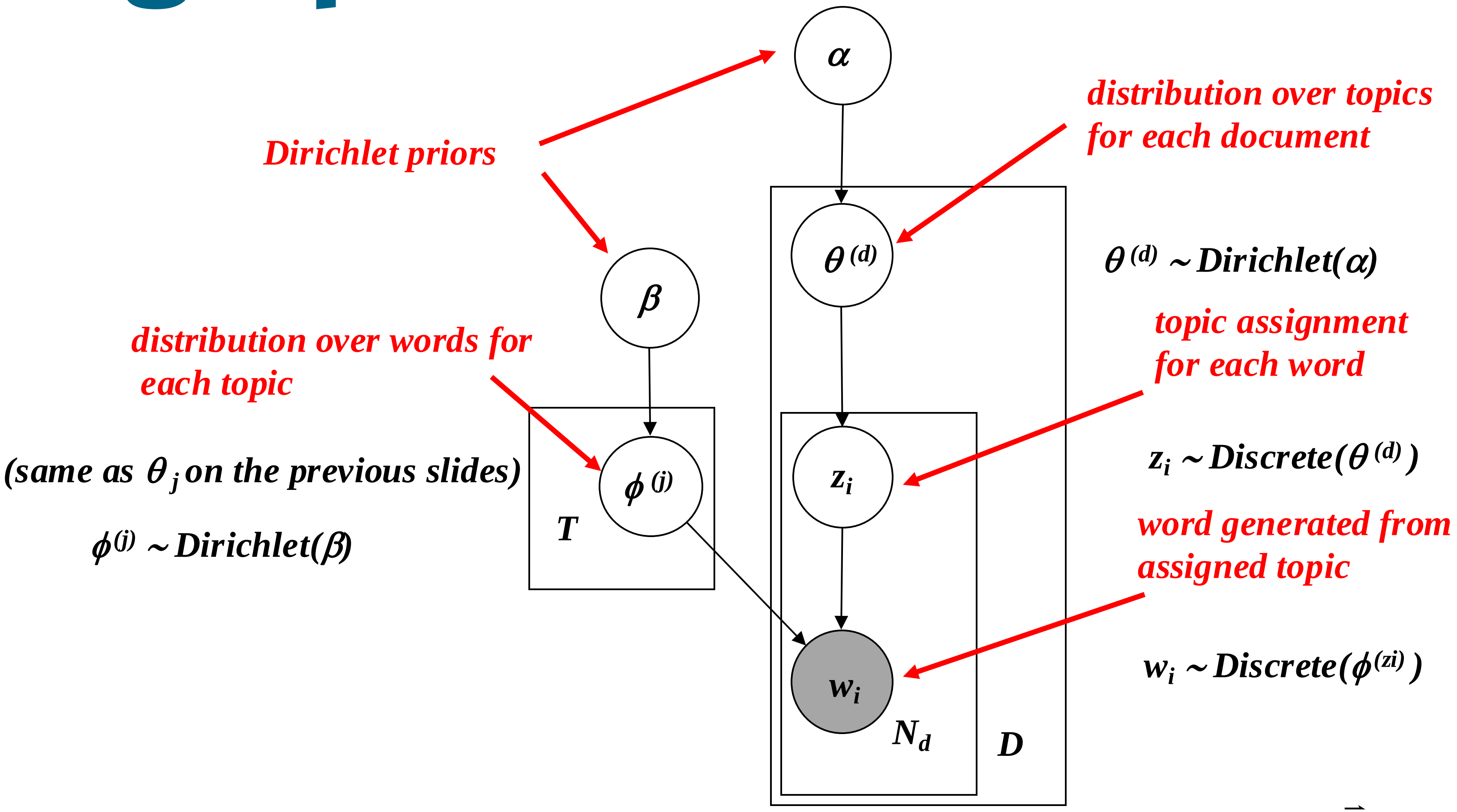


Figure 3. Illustrating the symmetric Dirichlet distribution for three topics on a two-dimensional simplex. Darker colors indicate higher probability. Left: $\alpha = 4$. Right: $\alpha = 2$.

α=50/T and β= 0.01 to work well with many different text collections.

LDA graphical model



Most approximate inference algorithms aim to infer $p(z_i \mid \vec{w}, \vec{\alpha}, \vec{\beta})$
from which other interesting variables can be easily computed

LDA geometric interpretation

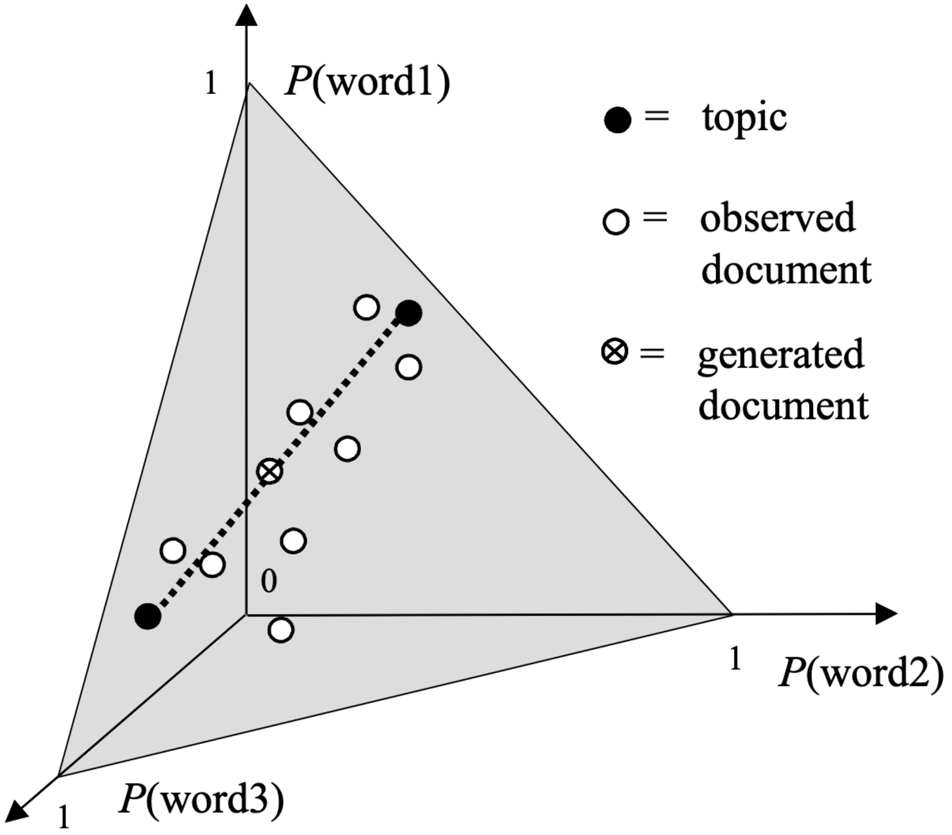


Figure 5. A geometric interpretation of the topic model.

LDA vs LSA

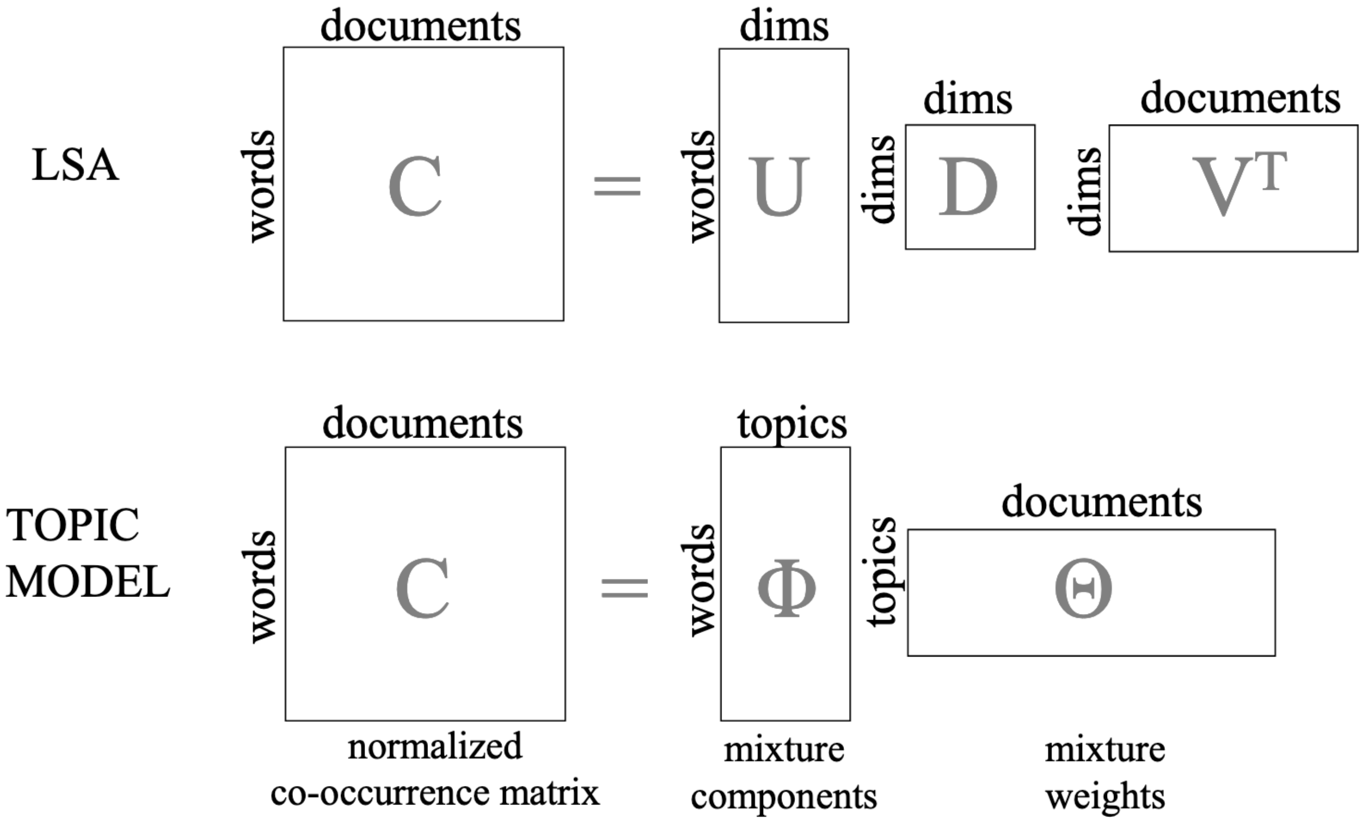


Figure 6. The matrix factorization of the LSA model compared to the matrix factorization of the topic model

Approximate inferences for LDA

- Deterministic approximation
  - Variational inference
  - Expectation propagation
- Markov chain Monte Carlo
  - Full Gibbs sampler
  - Collapsed Gibbs sampler

Topics learned by LDA

AP corpus

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

Topic assignments

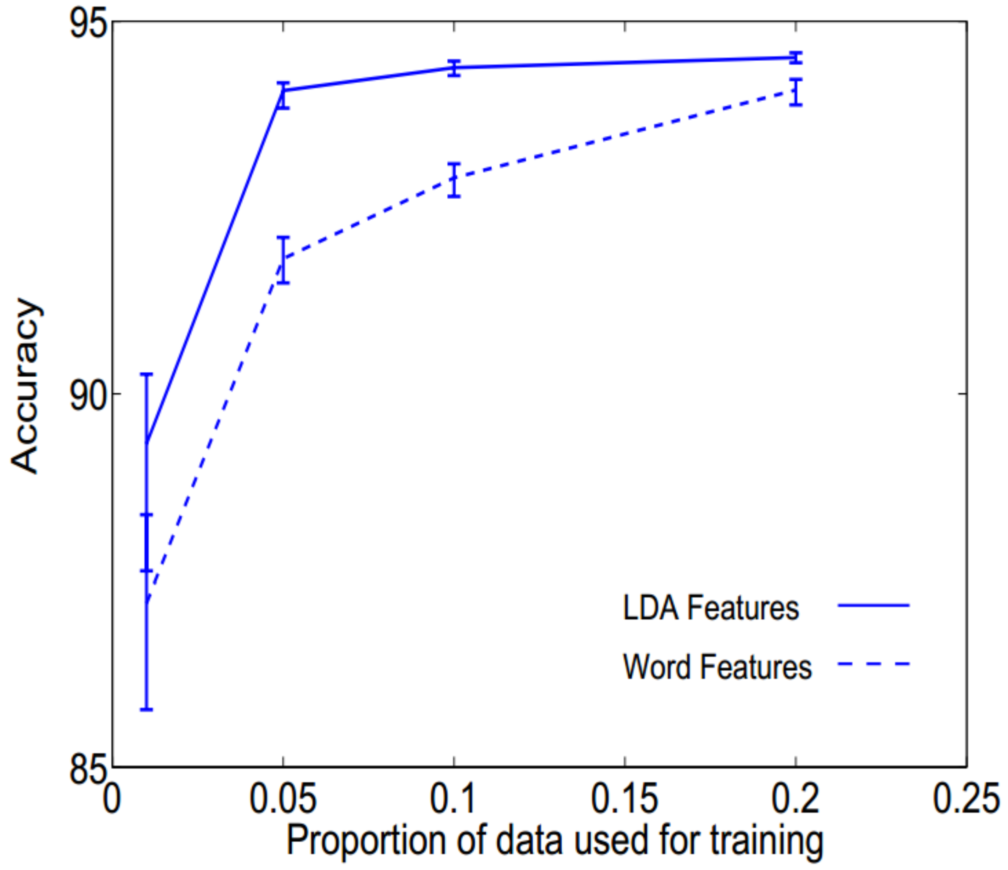
AP corpus

"Arts"	"Budgets"	"Children"	"Education"

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Application of learned topics

- Document classification



Polysemy with topics

Topic 77

word	prob.
MUSIC	.090
DANCE	.034
SONG	.033
PLAY	.030
SING	.026
SINGING	.026
BAND	.026
PLAYED	.023
SANG	.022
SONGS	.021
DANCING	.020
PIANO	.017
PLAYING	.016
RHYTHM	.015
ALBERT	.013
MUSICAL	.013

Topic 82

word	prob.
LITERATURE	.031
POEM	.028
POETRY	.027
POET	.020
PLAYS	.019
POEMS	.019
PLAY	.015
LITERARY	.013
WRITERS	.013
DRAMA	.012
WROTE	.012
POETS	.011
WRITER	.011
SHAKESPEARE	.010
WRITTEN	.009
STAGE	.009

Topic 166

word	prob.
PLAY	.136
BALL	.129
GAME	.065
PLAYING	.042
HIT	.032
PLAYED	.031
BASEBALL	.027
GAMES	.025
BAT	.019
RUN	.019
THROW	.016
BALLS	.015
TENNIS	.011
HOME	.010
CATCH	.010
FIELD	.010

Figure 9. Three topics related to the word PLAY.

Document #29795

Bix beiderbecke, at age^{060} fifteen^{207}, sat^{174} on the slope^{071} of a bluff^{055} overlooking^{027} the mississippi^{137} river^{137}. He was listening^{077} to music^{077} coming^{009} from a passing^{043} riverboat. The music^{077} had already captured^{006} his heart^{157} as well as his ear^{119}. It was jazz^{077}. Bix beiderbecke had already had music^{077} lessons^{077}. He showed^{002} promise^{134} on the piano^{077}, and his parents^{035} hoped^{268} he might consider^{118} becoming a concert^{077} pianist^{077}. But bix was interested^{268} in another kind^{050} of music^{077}. He wanted^{268} to $\boxed{\text{play}^{077}}$ the cornet. And he wanted^{268} to $\boxed{\text{play}^{077}}$ jazz^{077}...

Document #1883

There is a simple^{050} reason^{106} why there are so few periods^{078} of really great theater^{082} in our whole western^{046} world. Too many things^{300} have to come right at the very same time. The dramatists must have the right actors^{082}, the actors^{082} must have the right playhouses, the playhouses must have the right audiences^{082}. We must remember^{288} that plays^{082} exist^{143} to be performed^{077}, not merely^{050} to be read^{254}. (even when you read^{254} a $\boxed{\text{play}^{082}}$ to yourself, try^{288} to perform^{062} it, to put^{174} it on a stage^{078}, as you go along.) as soon^{028} as a $\boxed{\text{play}^{082}}$ has to be performed^{082}, then some kind^{126} of theatrical^{082}...

Document #21359

Jim^{296} has a game^{166} book^{254}. Jim^{296} reads^{254} the book^{254}. Jim^{296} sees^{081} a game^{166} for one. Jim^{296} plays^{166} the game^{166}. Jim^{296} likes^{081} the game^{166} for one. The game^{166} book^{254} helps^{081} jim^{296}. Don^{180} comes^{040} into the house^{038}. Don^{180} and jim^{296} read^{254} the game^{166} book^{254}. The boys^{020} see a game^{166} for two. The two boys^{020} $\boxed{\text{play}^{166}}$ the game^{166}. The boys^{020} $\boxed{\text{play}^{166}}$ the game^{166} for two. The boys^{020} like the game^{166}. Meg^{282} comes^{040} into the house^{282}. Meg^{282} and don^{180} and jim^{296} read^{254} the book^{254}. They see a game^{166} for three. Meg^{282} and don^{180} and jim^{296} $\boxed{\text{play}^{166}}$ the game^{166}. They $\boxed{\text{play}^{166}}$...

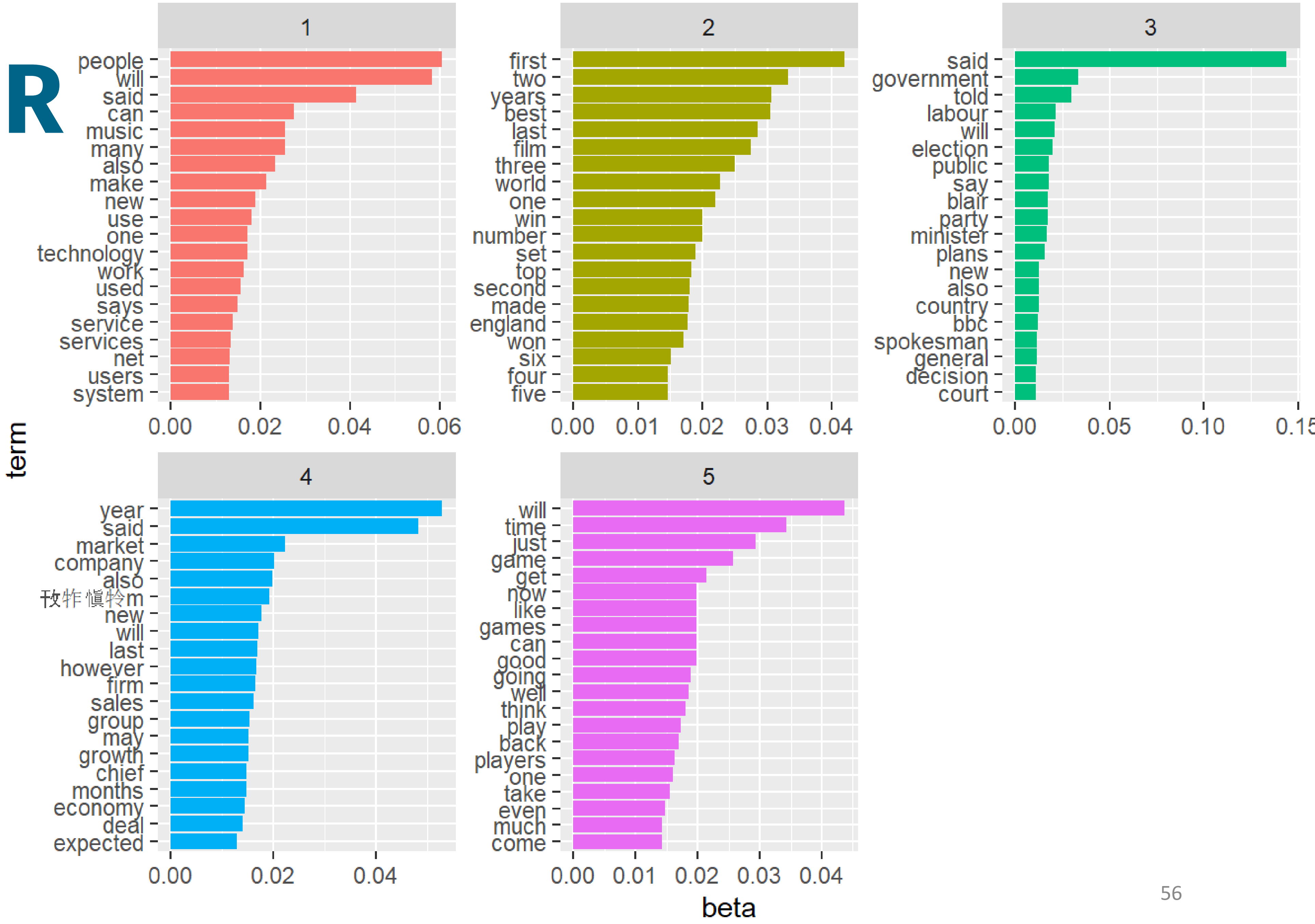
Figure 10. Three TASA documents with the word *play*.

LDA in R

```r
library(topicmodels)
# prepare your data
dtm <- DocumentTermMatrix(docs,
                          control = list(tolower = TRUE,
                                         removeNumbers = TRUE,
                                         removePunctuation = TRUE,
                                         stopwords = TRUE))

# LDA with 5 topics
out_lda <- LDA(dtm, k = 5, method= "Gibbs", control = list(seed = 321))
```

LDA in R


1
people
will
said
can
music
many
also
make
new
use
one
technology
work
used
says
service
services
net
users
system
0.00 0.02 0.04 0.06
2
first
two
years
best
last
film
three
world
one
win
number
set
top
second
made
england
won
six
four
five
0.00 0.01 0.02 0.03 0.04
3
said
government
told
labour
will
election
public
say
blair
party
minister
plans
new
also
country
bbc
spokesman
general
decision
court
0.00 0.05 0.10 0.15
4
year
said
market
company
also
new
will
last
however
firm
sales
group
may
growth
chief
months
economy
deal
expected
0.00 0.02 0.04
5
will
time
just
game
get
now
like
games
can
good
going
well
think
play
back
players
one
take
even
much
come
0.00 0.01 0.02 0.03 0.04
term
beta

Conclusions

Text representations can be high-dimensional!

Topic modelling can be a solution.

Practical

Create document-term matrices on BBC news dataset and apply LDA topic modeling.

Thanks!

q.fang@uu.nl

Additional information on LDA

Collapsed Gibbs sampling

- Sample each z_i conditioned on $\mathbf{z}_{-i}$

All the other words beside z_i

$$P(z_i \mid \mathbf{w}, \mathbf{z}_{-i}) \propto \boxed{\frac{n_{w_i}^{(z_i)} + \beta}{n_{\bullet}^{(z_i)} + W\beta}} \boxed{\frac{n_j^{(d_i)} + \alpha}{n_{\bullet}^{(d_i)} + T\alpha}}$$

Word-topic distribution **Topic proportion**

- Implementation: counts can be cached in two sparse matrices; no special functions, simple arithmetic
- Distributions on Φ and Θ can be analytic computed given z and w

Latent Dirichlet Allocation

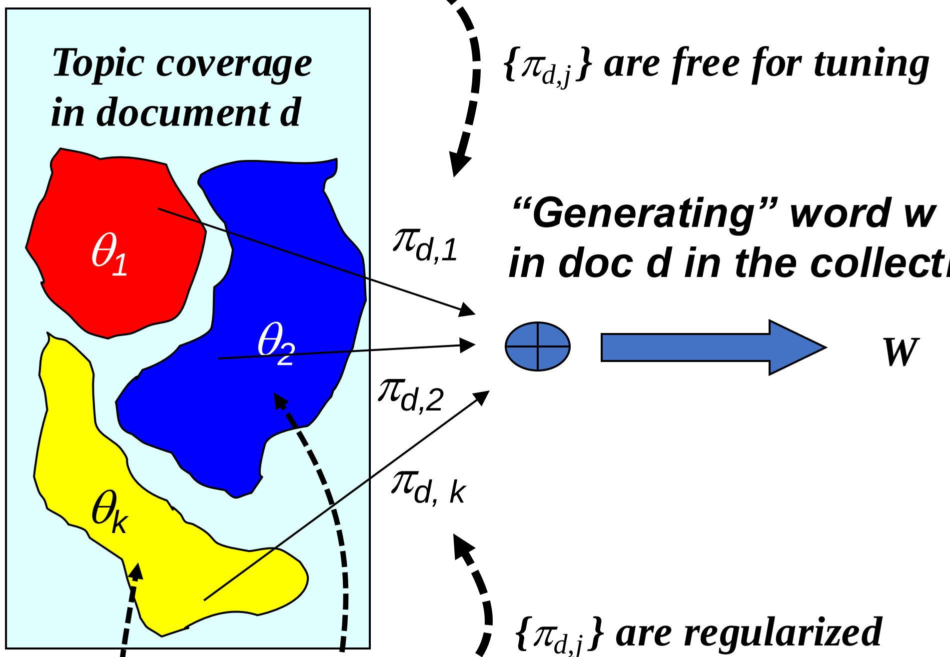
- Makes pLSA a fully generative model by imposing Dirichlet priors
  - Dirichlet priors over p(π|d)
  - Dirichlet priors over p(w|θ)
  - A Bayesian version of pLSA
- Provides mechanism to deal with new documents
  - Flexible to model many other observations in a document

LDA = Imposing Prior on PLSA

pLSA:
Topic coverage $\pi_{d,j}$ is specific to each "training document", thus can't be used to generate a new document

$\{\pi_{d,j}\}$ are free for tuning

"Generating" word w in doc d in the collection



$\{\pi_{d,j}\}$ are regularized

LDA:
Topic coverage distribution $\{\pi_{d,j}\}$ for any document is sampled from a Dirichlet distribution, allowing for generating a new doc

$$p(\vec{\pi}_d) = Dirichlet(\vec{\alpha})$$

In addition, the topic word distributions $\{\theta_j\}$ are also drawn from another Dirichlet prior

$$p(\vec{\theta}_i) = Dirichlet(\vec{\beta})$$

Magnitudes of α and β determine the variances of the prior, thus also the concentration of prior (larger α and β ➔ stronger prior)

EM computation

$$p^{(n)}(z_i = 1|w_i) = \frac{\lambda p(w_i|\theta_G)}{\lambda p(w_i|\theta_G) + (1-\lambda)p^{(n)}(w_i|\theta)}$$

Expectation-Step:
Augmenting data by guessing hidden variables

$$p^{(n+1)}(w_i|\theta) = \frac{c(w_i,d)(1-p^{(n)}(z_i = 1|w_i))}{\sum_{w_j \in vocabulary} c(w_j,d)(1-p^{(n)}(z_j = 1|w_j))}$$

Maximization-Step:
With the "augmented data", estimate parameters using maximum likelihood

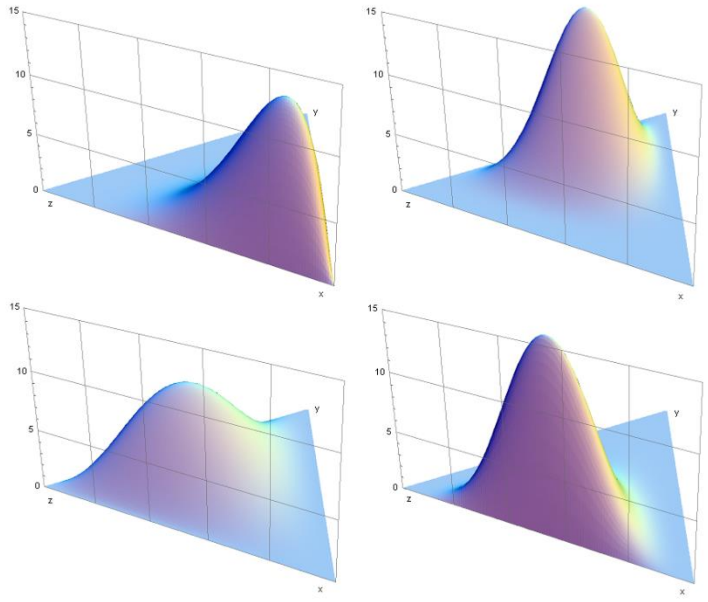
Assume $\lambda = 0.5$

Word	#	$P(w\|\theta_G)$	Iteration 1		Iteration 2		Iteration 3	
			$P(w\|\theta)$	$P(z=1)$	$P(w\|\theta)$	$P(z=1)$	$P(w\|\theta)$	$P(z=1)$
The	4	0.5	**0.25**	0.67	**0.20**	0.71	**0.18**	0.74
Paper	2	0.3	**0.25**	0.55	**0.14**	0.68	**0.10**	0.75
Text	4	0.1	**0.25**	0.29	**0.44**	0.19	**0.50**	0.17
Mining	2	0.1	**0.25**	0.29	**0.22**	0.31	**0.22**	0.31
Log-Likelihood			-16.96		-16.13		-16.02	

Some background knowledge

- Conjugate prior
  - Posterior dist in the same family as prior
- Dirichlet distribution
  - Continuous
  - Samples from it will be the parameters in a multinomial distribution

Gaussian -> Gaussian
Beta -> Binomial
Dirichlet -> Multinomial



pLSA vs LDA

pLSA

$$p_d(w|\{\theta_j\},\{\pi_{d,j}\}) = \underline{\sum_{j=1}^{k}\pi_{d,j}\,p(w|\theta_j)}$$

Core assumption in all topic models

$$\log p(d|\{\theta_j\},\{\pi_{d,j}\}) = \boxed{\sum_{w\in V} c(w,d)\log[\sum_{j=1}^{k}\pi_{d,j}\,p(w|\theta_j)]}$$

$$\log p(C|\{\theta_j\},\{\pi_{d,j}\}) = \sum_{d\in C}\log p(d|\{\theta_j\},\{\pi_{d,j}\})$$

pLSA component

LDA

$$p_d(w|\{\theta_j\},\{\pi_{d,j}\}) = \underline{\sum_{j=1}^{k}\pi_{d,j}\,p(w|\theta_j)}$$

$$\log p(d|\vec{\alpha},\{\theta_j\}) = \boxed{\int\sum_{w\in V} c(w,d)\log[\sum_{j=1}^{k}\pi_{d,j}\,p(w|\theta_j)]}\,\underline{p(\vec{\pi}_d|\vec{\alpha})d\vec{\pi}_d}$$

$$\log p(C|\vec{\alpha},\vec{\beta}) = \int\sum_{d\in C}\log p(d|\vec{\alpha},\{\theta_j\})\underline{\prod_{j=1}^{k}p(\theta_j|\vec{\beta})d\theta_1\ldots d\theta_k}$$

Regularization added by LDA

Variants of topic models

- Smoothed LDA
- Correlated Topic Models
- Hierarchical Topic Models
- Dynamic Topic Models
- Contextual Topic Models
- BERTopic
- And many more!