Topic Models

CS 534: Machine Learning

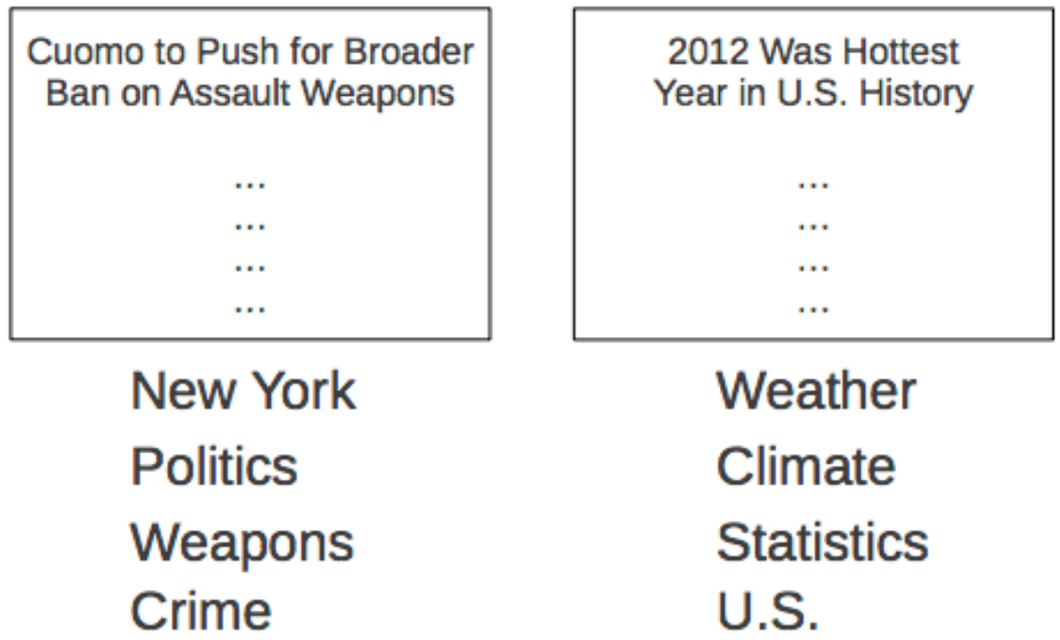
Slides adapted from David Sontag, David Blei, Nicholas Ruozzi, Guillaume Obozinski, and Ankur Moitra

Topic Models

- Discover themes (topics) from electronic archives (e.g., newspaper articles)

- Annotates the collection according to the discovered themes

- Use annotations to organize, search, summarize, etc.



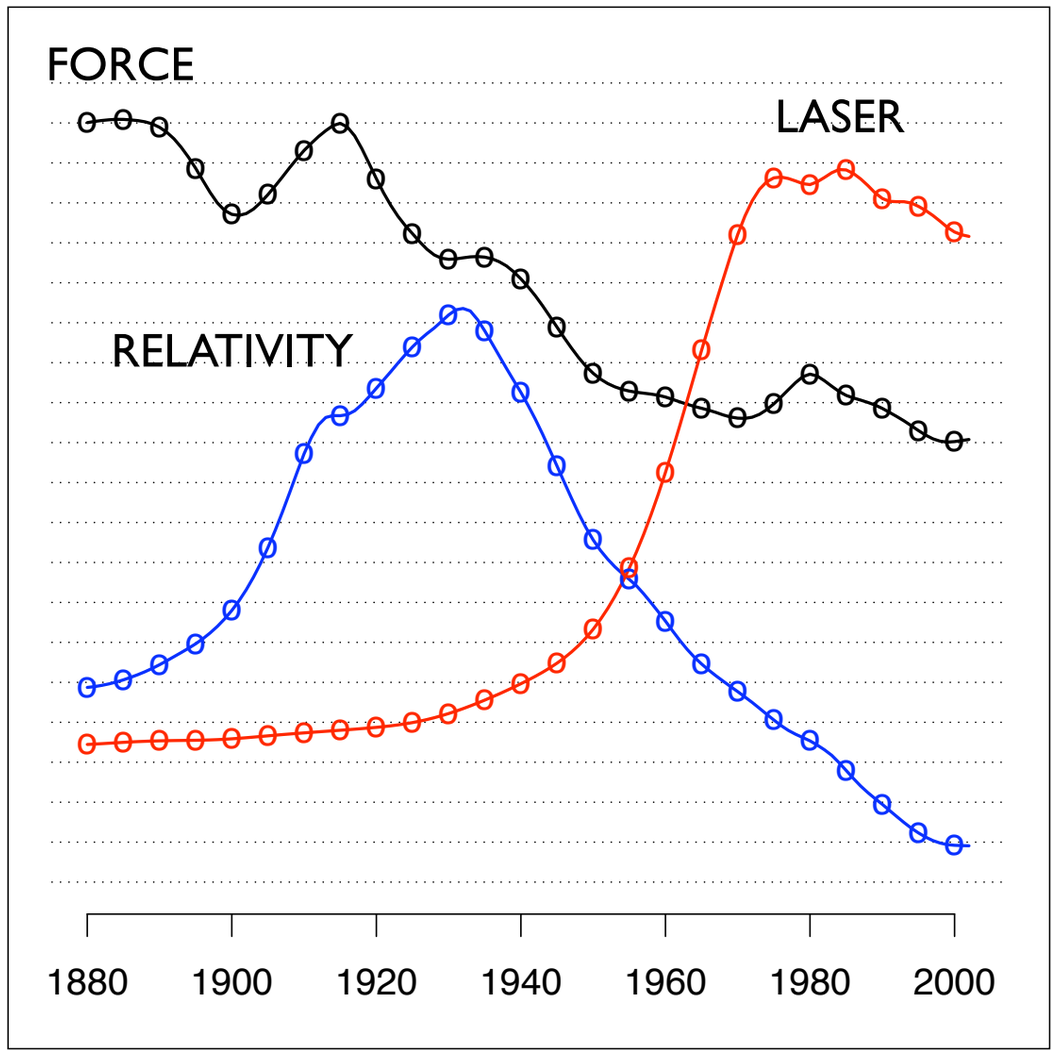
http://www.phontron.com/slides/nlp-programming-en-09-topic.pdf

Topic Models: Discover Topics

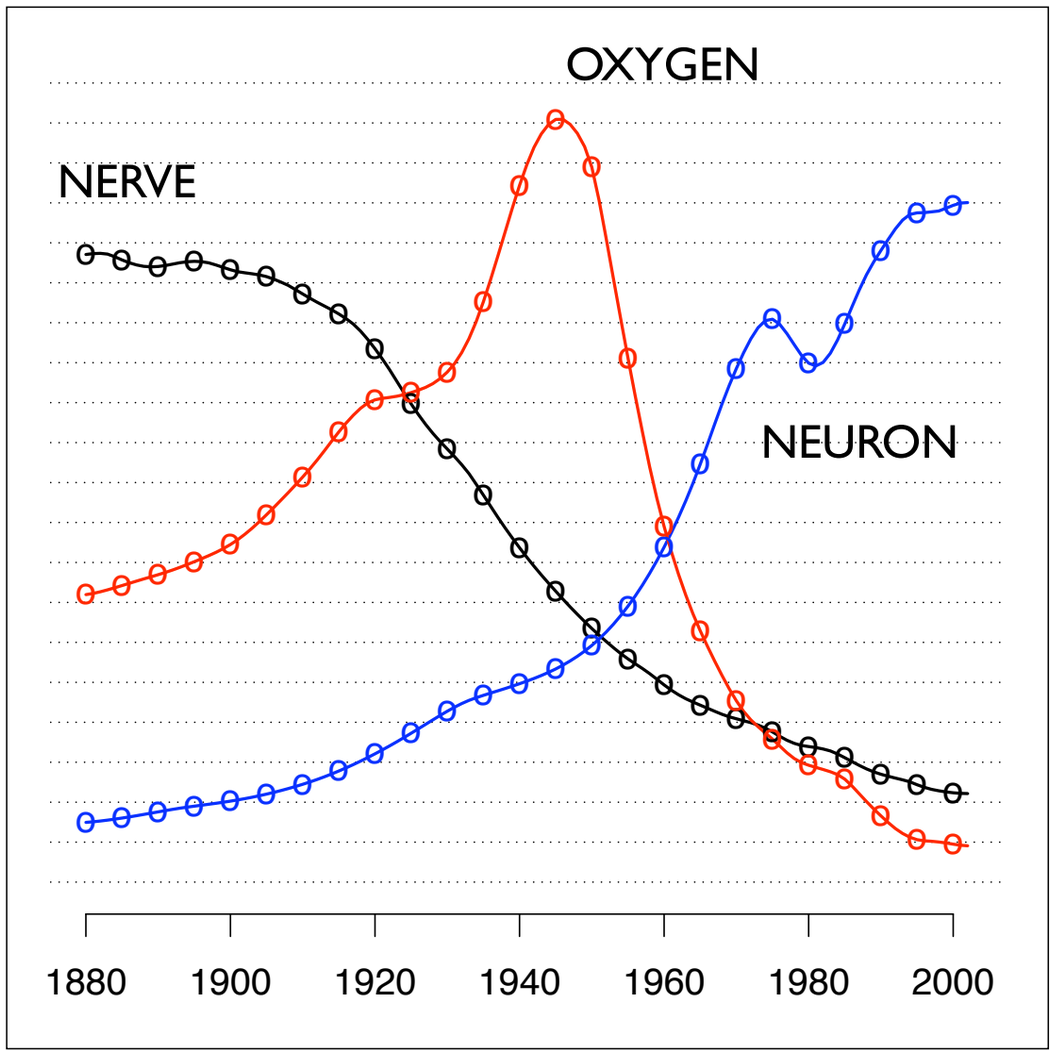
problem	model	selection	species
problems	rate	male	forest
mathematical	constant	males	ecology
number	distribution	females	fish
new	time	sex	ecological
mathematics	number	species	conservation
university	size	female	diversity
two	values	evolution	population
first	value	populations	natural
numbers	average	population	ecosystems
work	rates	sexual	populations
time	data	behavior	endangered
mathematicians	density	evolutionary	tropical
chaos	measured	genetic	forests
chaotic	models	reproductive	ecosystem

Topic Models: Evolution of Topics

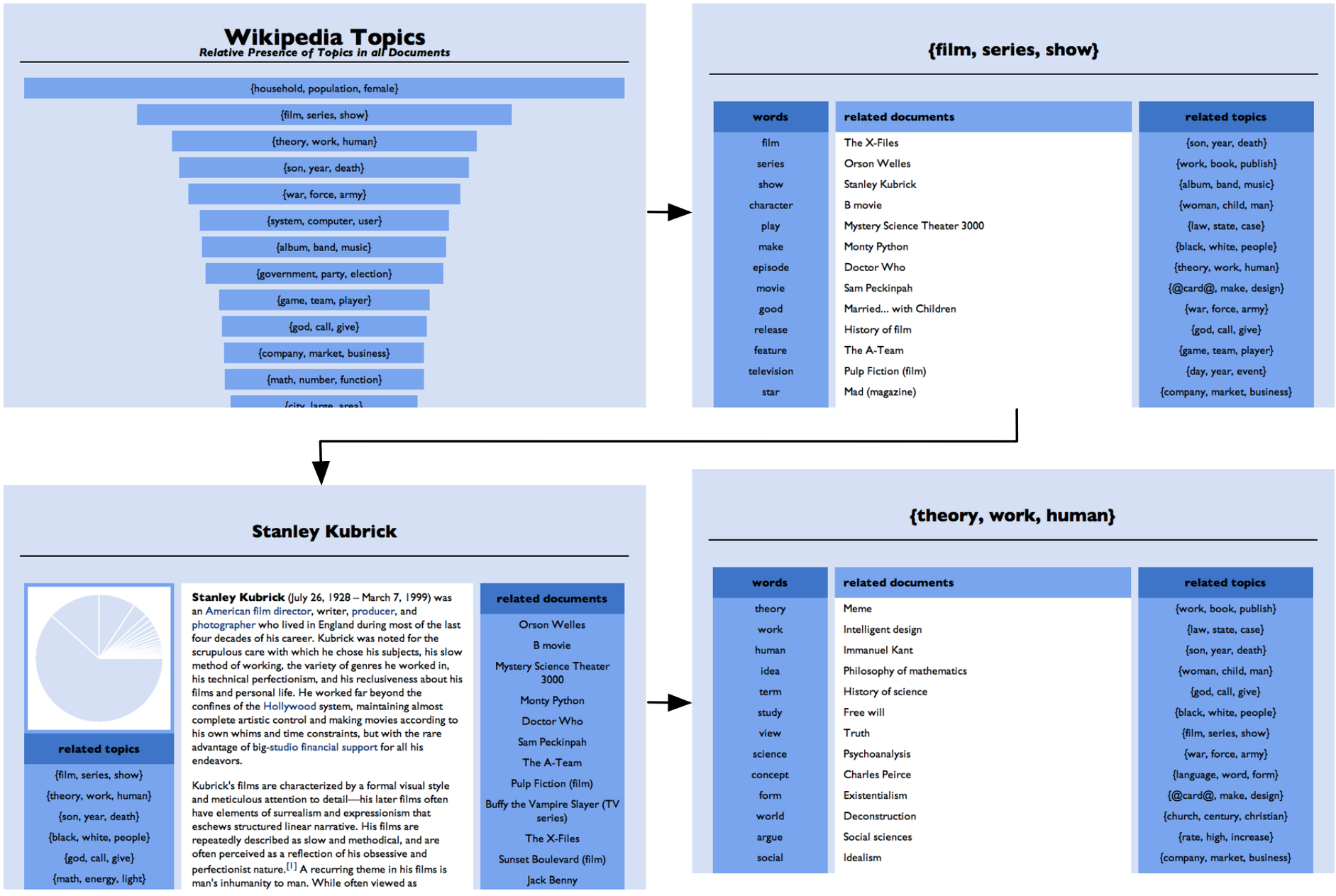
"Theoretical Physics"



"Neuroscience"



Topic Models: Organize & Browse

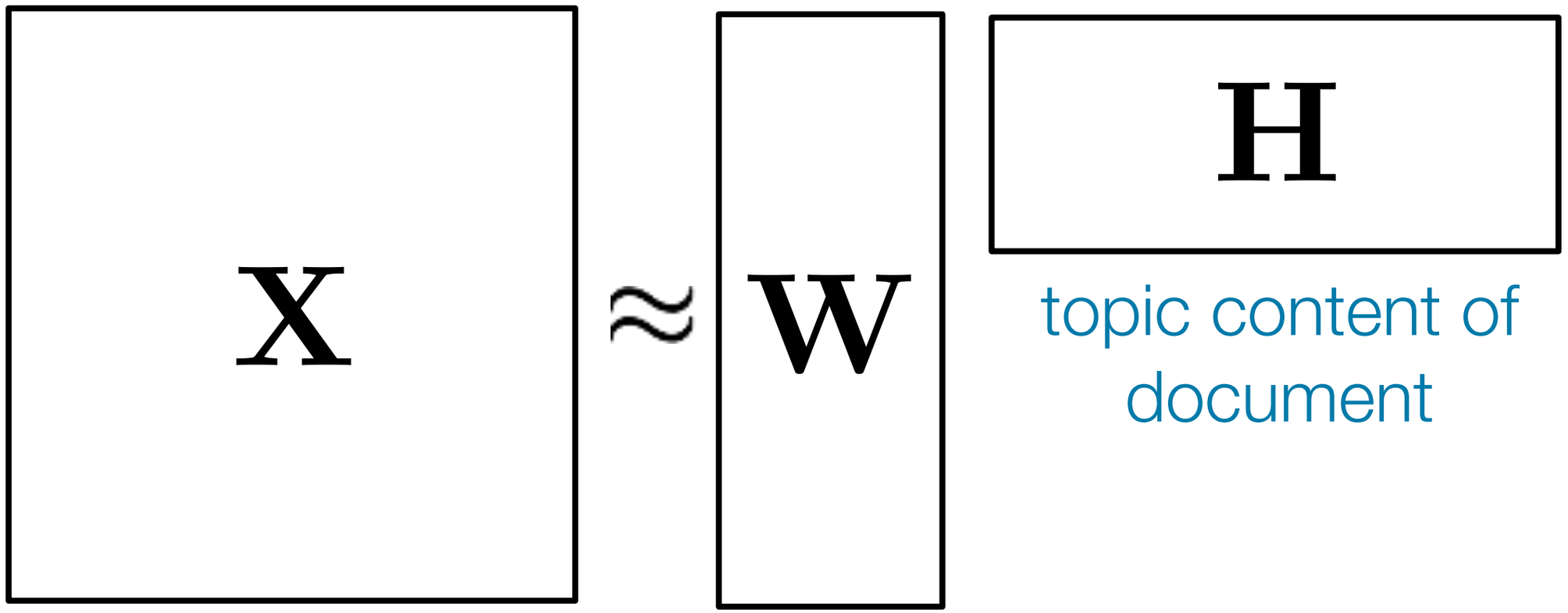


Latent Semantic Analysis (LSA)

- In information retrieval, known as Latent Semantic Indexing (LSI)
- Perform a low-rank approximation of document-term matrix
- Design mapping so that it reflects semantic association
- Similar terms map to similar location in low dimensional space

Topic Models & Matrix Factorization

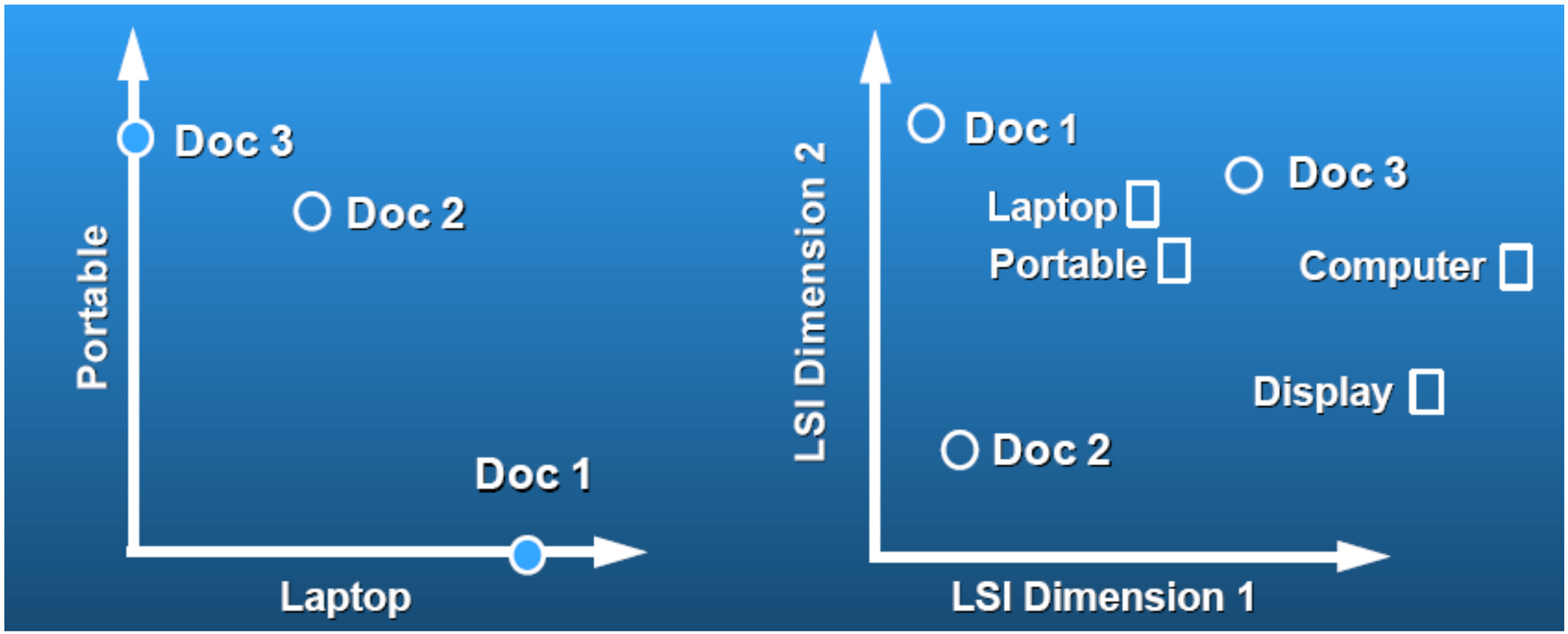
term by document matrix



topic content of document

topics matrix

Example: LSA


Doc 3
Doc 2
Doc 1
Portable
Laptop
Doc 1
Doc 3
Laptop
Portable
Computer
Display
Doc 2
LSI Dimension 2
LSI Dimension 1

LSA vs PCA

- Almost PCA on the document-term matrix
  - Find directions of high correlations between words called principal directions
- Retains projection of the data
- LSA does not center data (no specific reason)
- LSA is typically combined with term frequency–inverse document frequency (TF-IDF)

LSI: Limitations / Shortcomings

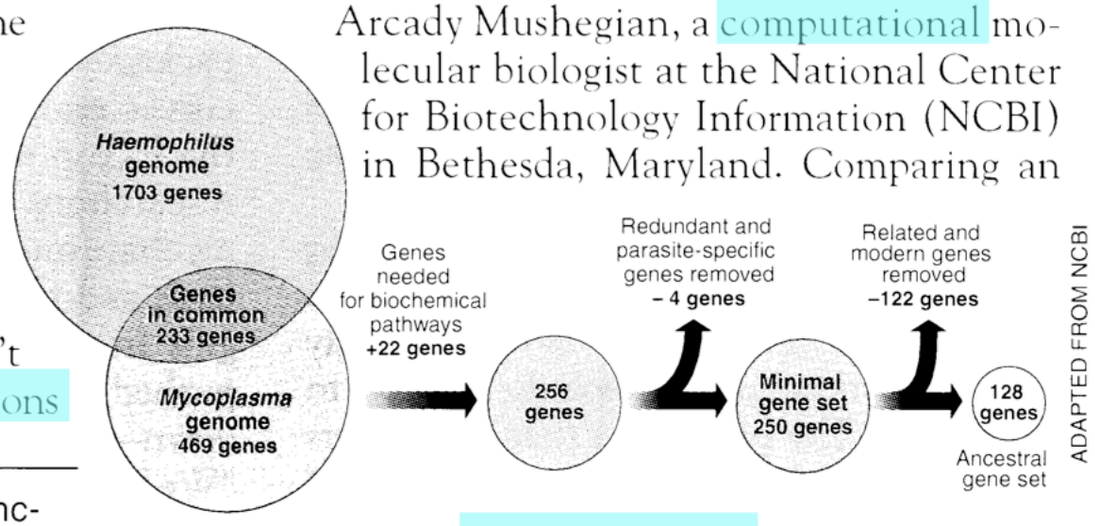
- PCA assumes data is generated from a Gaussian cloud — mismatch with data
  - Data are counts, frequencies, or TF-IDF scores
- SVD is expensive to compute on large matrix
- Context of terms is not taken into account (bag of words)
- Direction in latent space are hard to interpret

Multiple Topics: Motivation

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

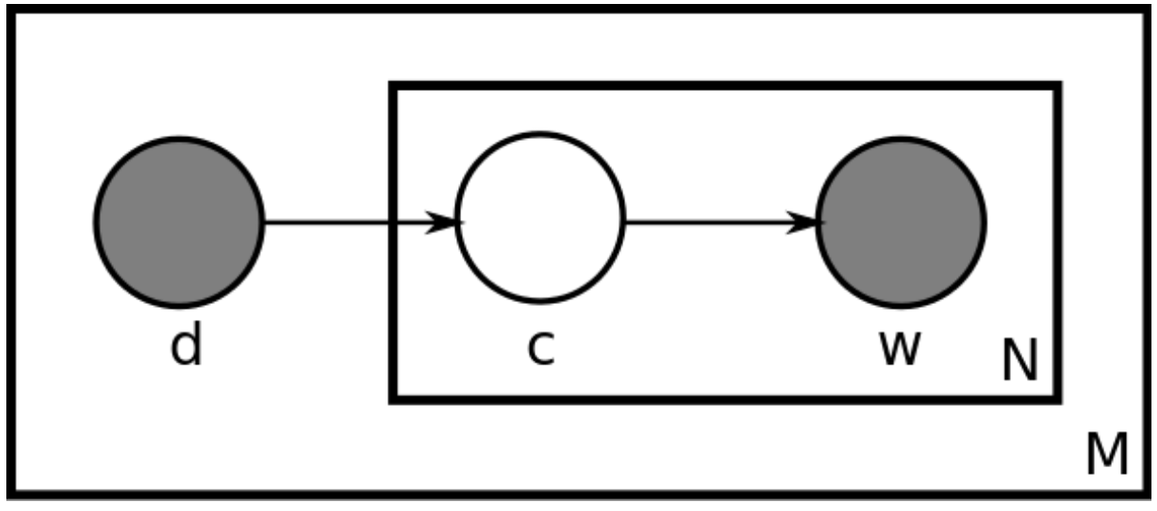
Document exhibits multiple topics

Probabilistic LSA (pLSA)

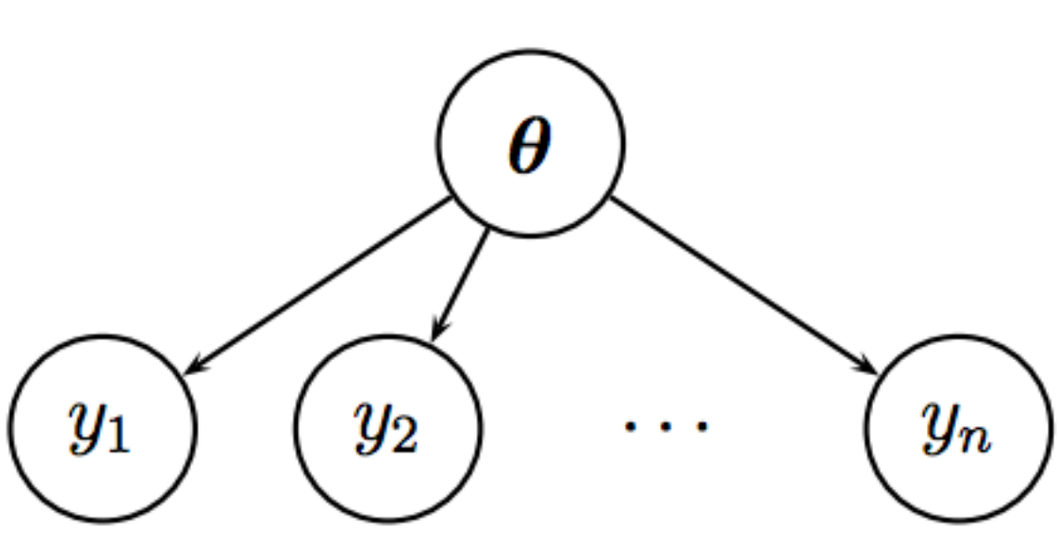
- Introduced by Hofmann, 2001

- Allows several topics per document in various proportions ($\mathbf{d}_i$)

- Each word gets its own topic ($\mathbf{c}_{in}$) drawn from multinomial distribution $\mathbf{d}_i$



Understanding Plate Notation



Graphical model with y_1, y_2, …, y_n determined by theta — note the repetitive structure

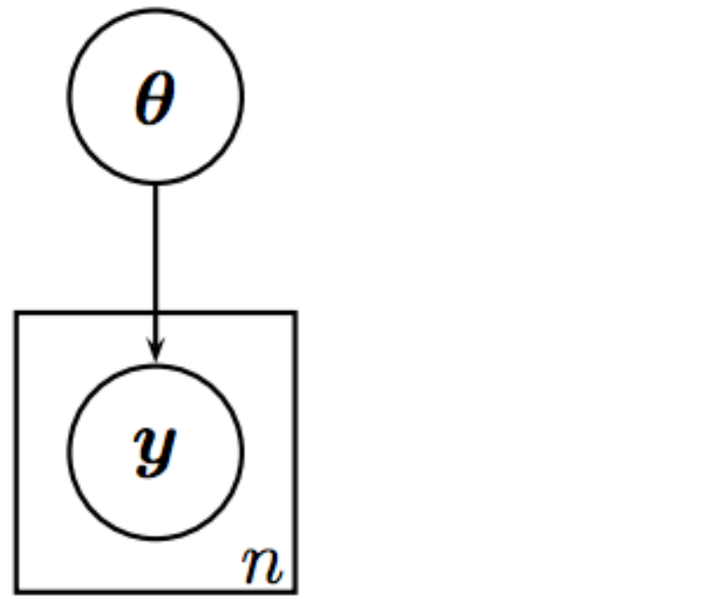
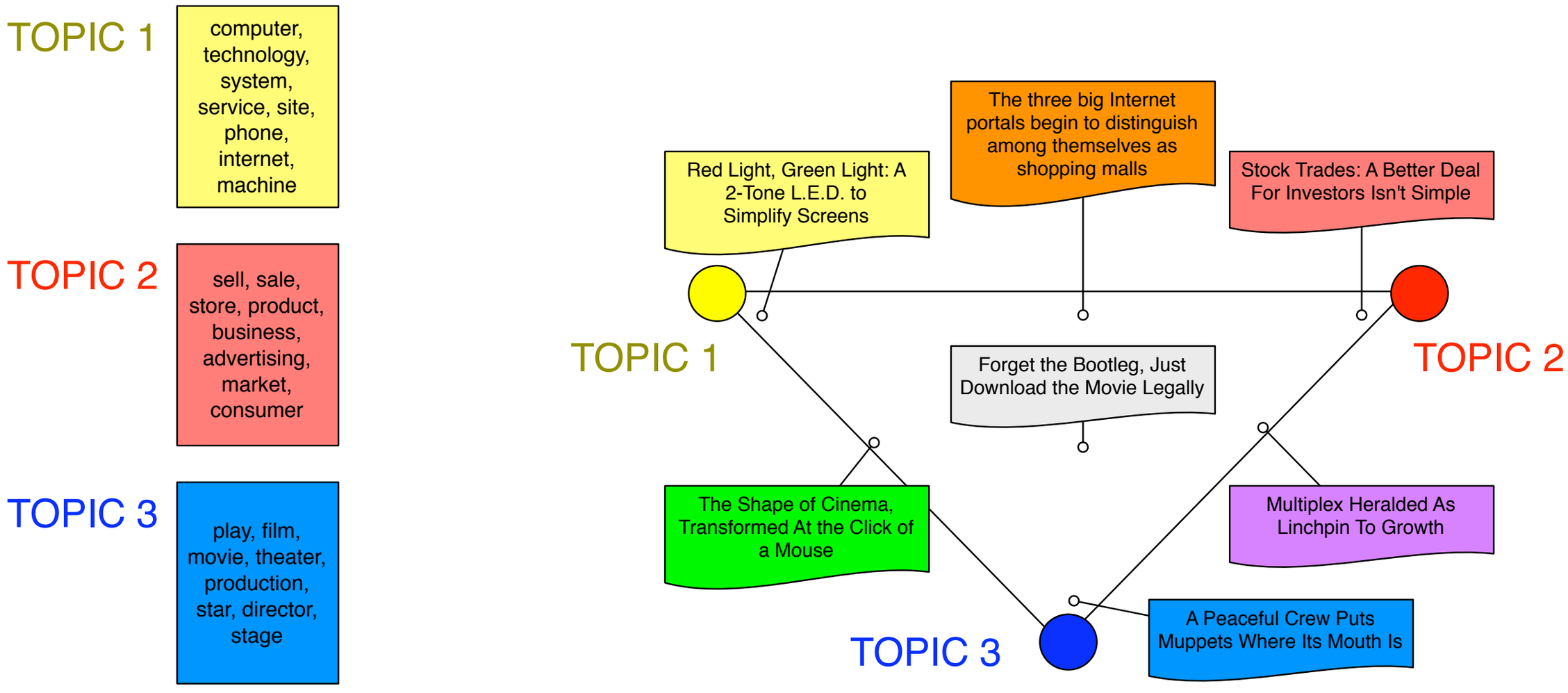


plate notation succinctly represents repetitive structure — variables within plate are replicated in conditionally independent manner

Example: pLSA



(a) Topics

(b) Document Assignments to Topics

pLSA

- Probability of each co-occurrence is a mixture of conditionally independent multinomial distributions

$$p(w,d) = \sum_c P(c)P(d|c)P(w|c) = P(d)\sum_c P(c|d)P(w|c)$$

- Document - topic probability distribution is shared by all words in a document ($p(c|d)$)

- Topic - word probability distribution shared by all documents ($p(w|c)$)

- Estimate parameters by maximum likelihood

pLSA: Expectation Maximization

- Log likelihood

$$L = \sum_{ij} n(w_j, d_i) \log(\sum_c P(c)P(d_i|c)P(w_j|c))$$

- E-step: estimate topics given words and documents

$$p(c|d,w) = \frac{P(w|c)P(d|c)P(c)}{\sum_c P(w|c)P(d|c)P(c)}$$

- M-step: estimate words per topic and topics per document

$$p(w|c) = \frac{\sum_d n(w,d)P(c|d,w)}{\sum_d \sum_w n(w,d)P(c|d,w)}$$

$$p(d|c) = \frac{\sum_w n(w,d)P(c|d,w)}{\sum_d \sum_w n(w,d)P(c|d,w)}$$

pLSA vs LSA

- Conditional independence assumption “replaces” outer product

- Class-conditional distributions “replace” left / right eigenvectors

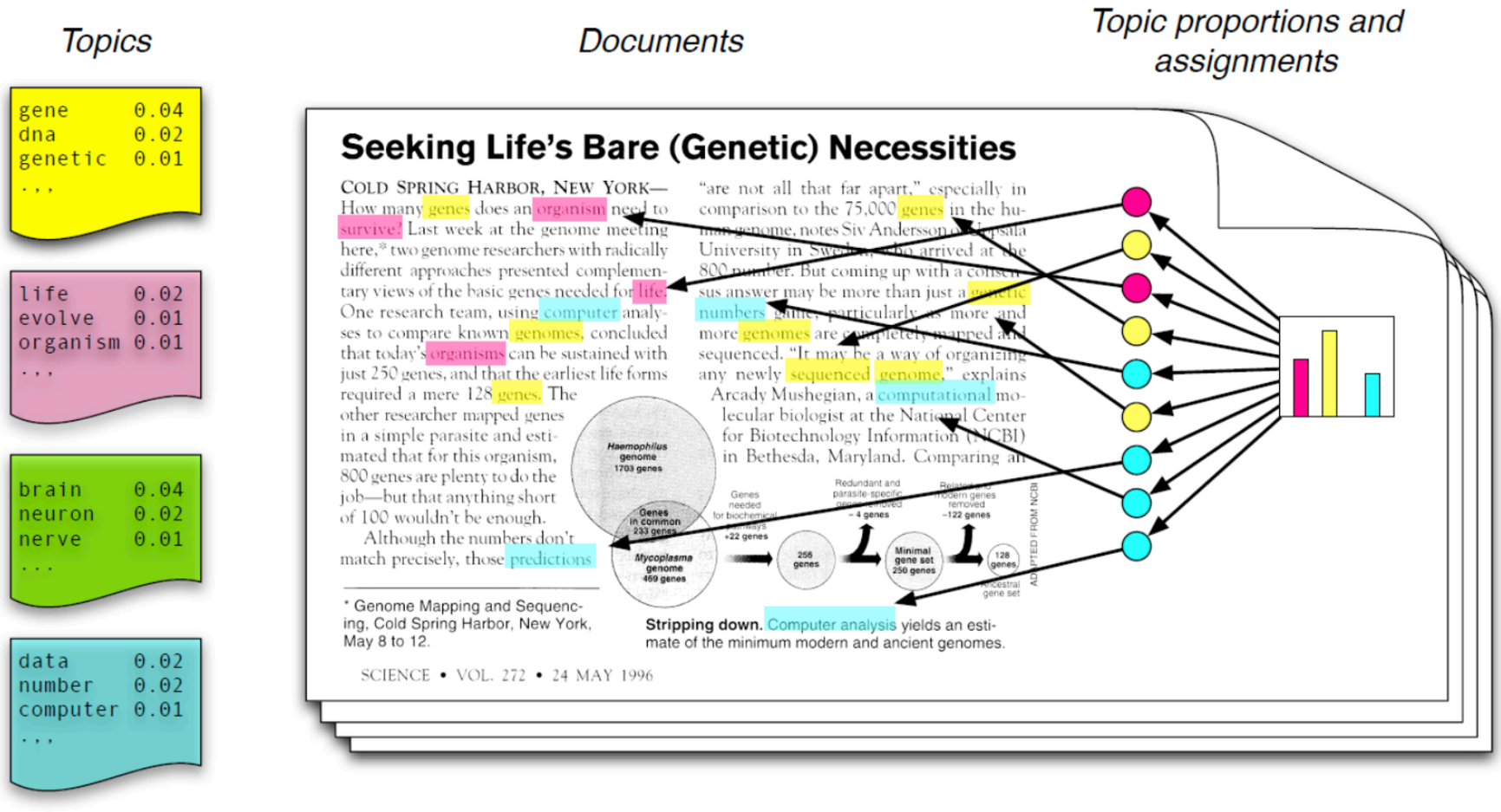
- Maximum likelihood instead of minimum L2 norm

pLSI: Limitations / Shortcomings

- Number of parameters increase linearly with number of documents (cd + wc)
  - Too many parameters —> overfitting
- No probabilistic model at the level of documents
  - Each document is represented as list of numbers (mixing proportions of topics)

Latent Dirichlet Allocation (LDA)

Each document is mixture of topics

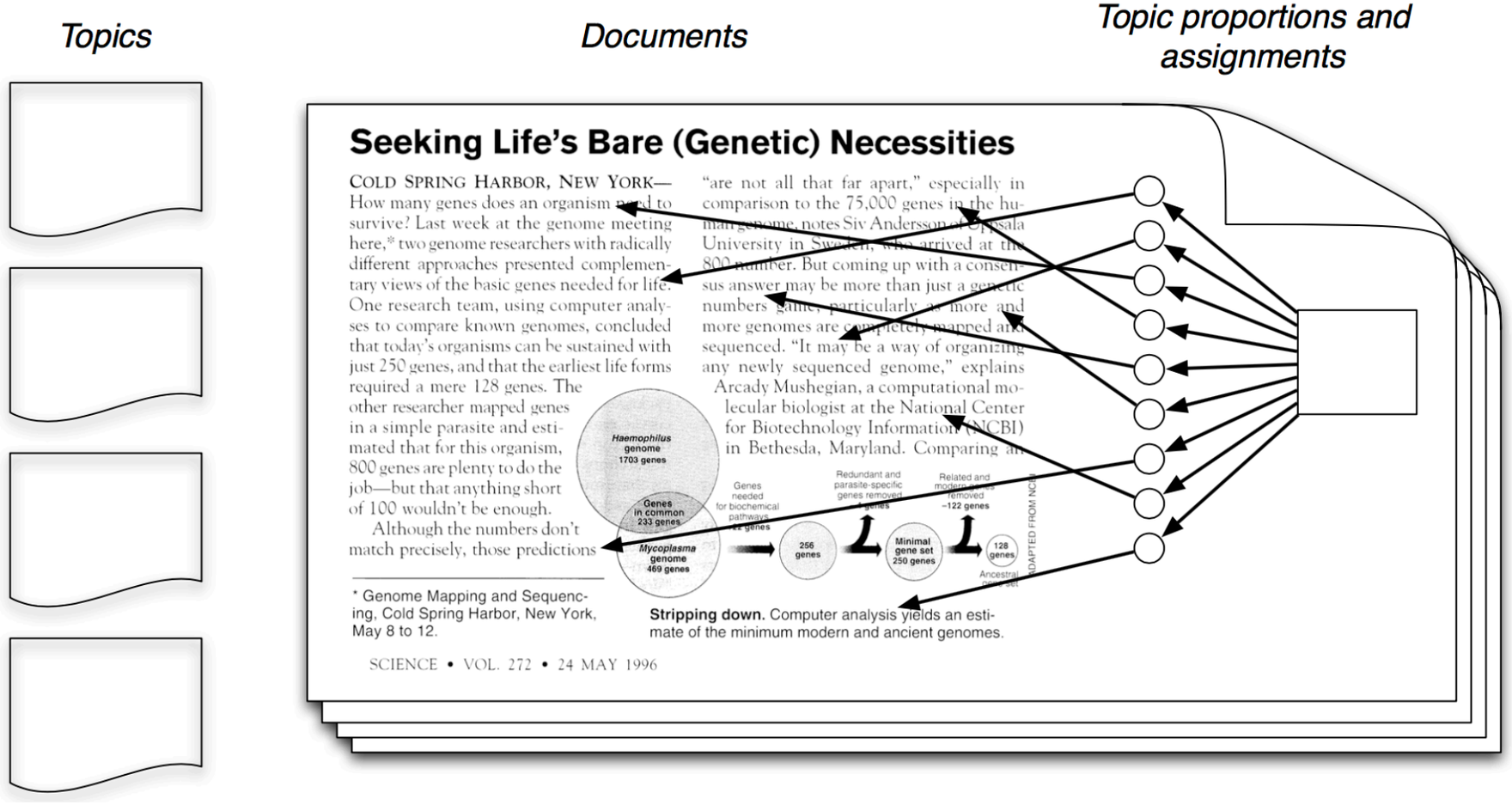


Each topic is a distribution of words

Each word is drawn from one of the topics

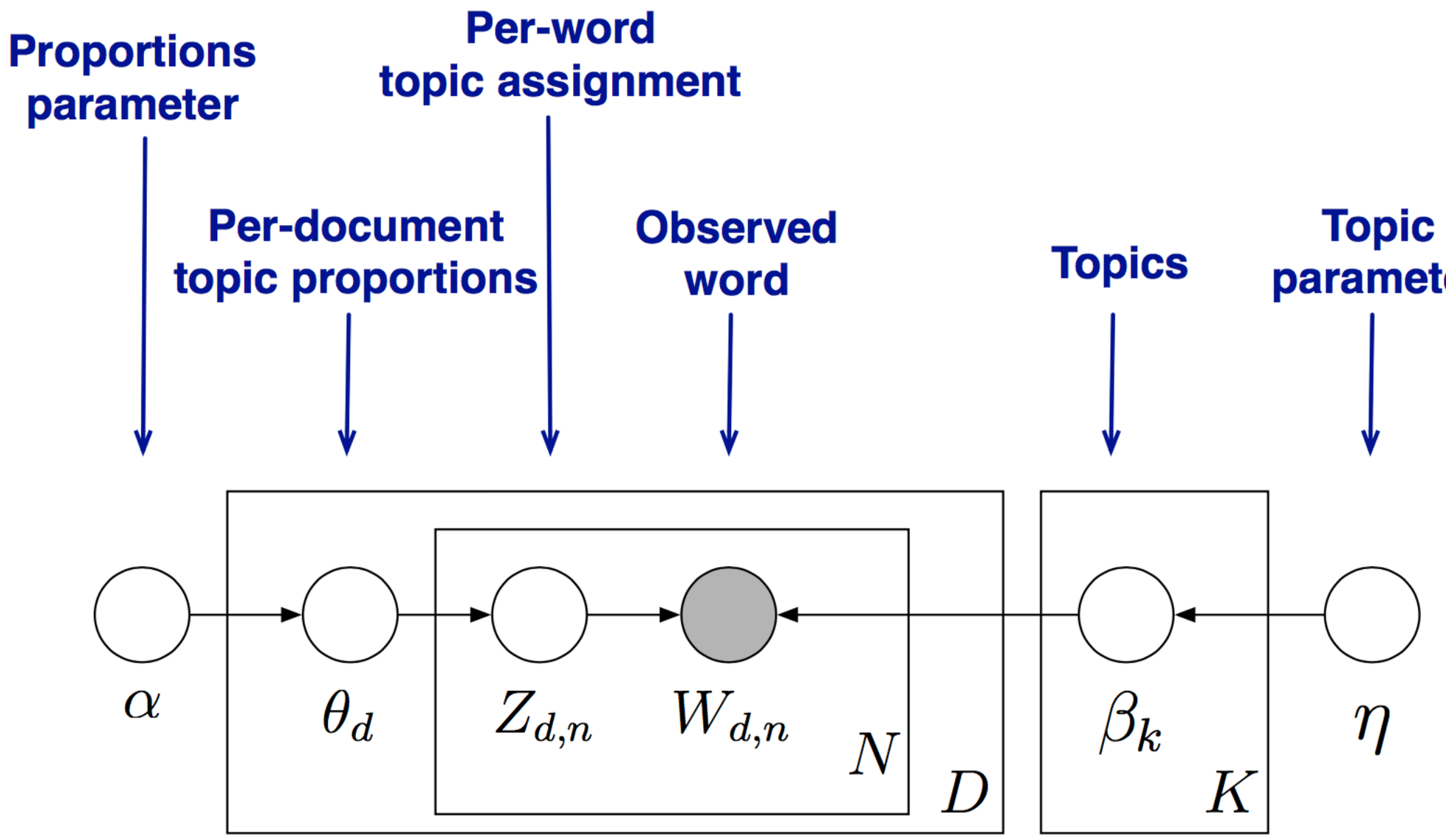
LDA: Posterior Distribution

We only observe the documents!



Everything else are hidden variables — need to be inferred / learned

LDA: Plate Model



LDA: Generative Model

- Sample document's topic distribution

$$\boldsymbol{\theta} \sim \text{Dirichlet}(\alpha_{1:T})$$

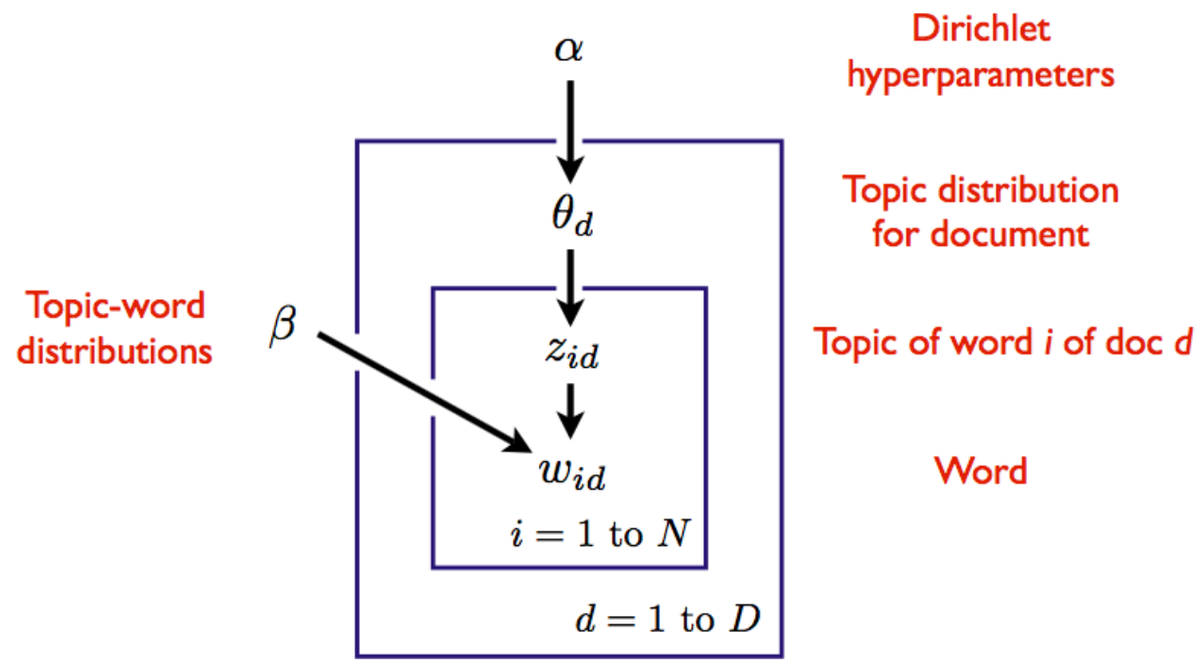
- For each word

- Sample the topic

$$z_i \mid \boldsymbol{\theta} \sim \boldsymbol{\theta}$$

- Sample actual word from topic

$$w_i | z_i \sim \beta_{z_i}$$

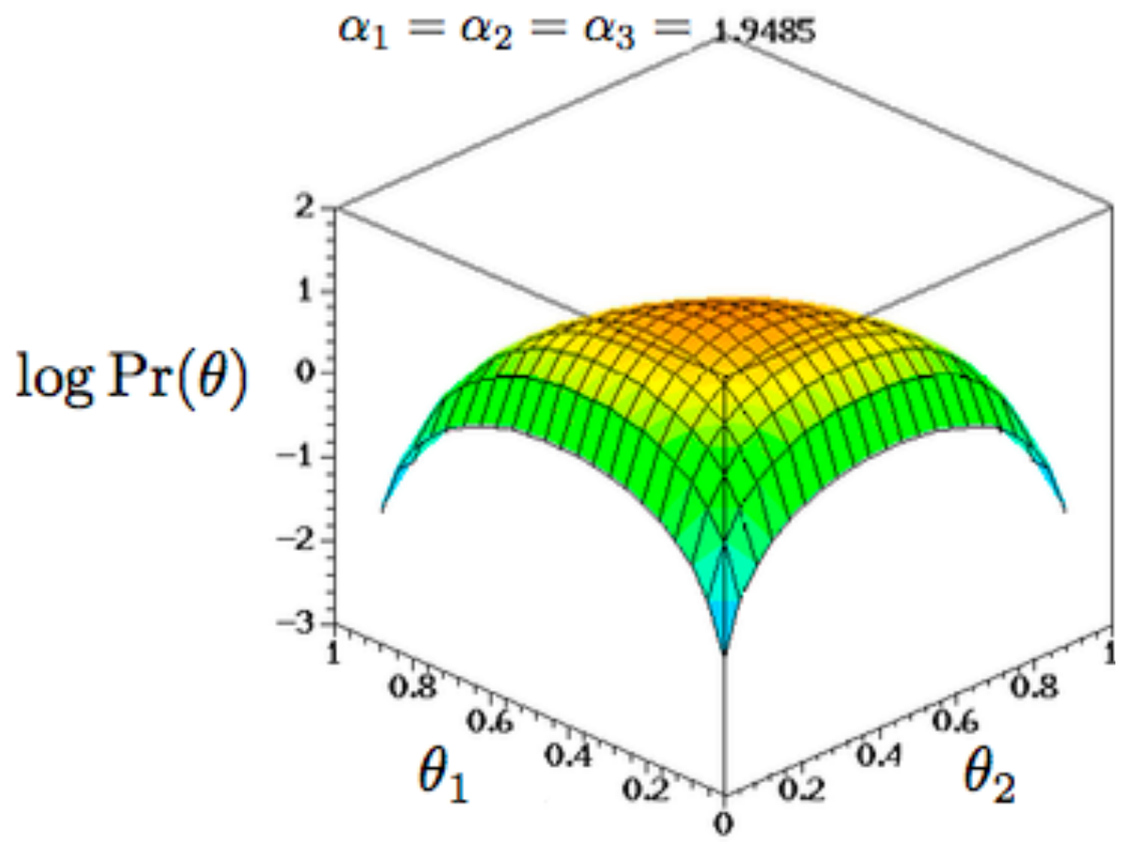


Exact inference is intractable!

LDA: Dirichlet Distribution

- Exponential family distribution over the simplex (i.e., positive vectors that sum to one)

$$p(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

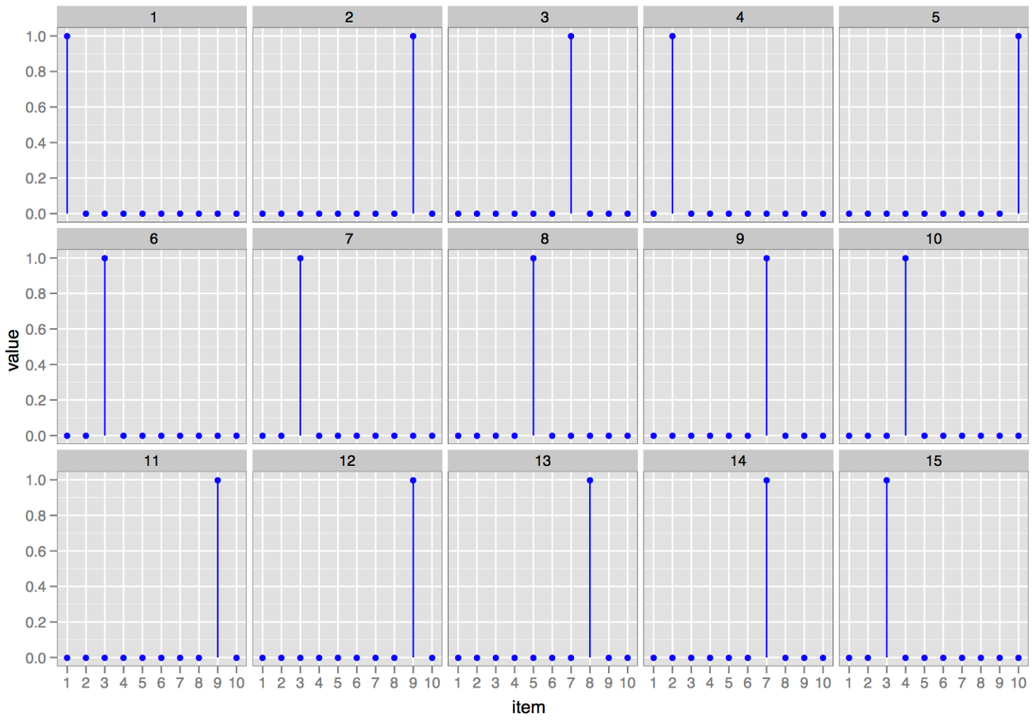


- Conjugate to the multinomial

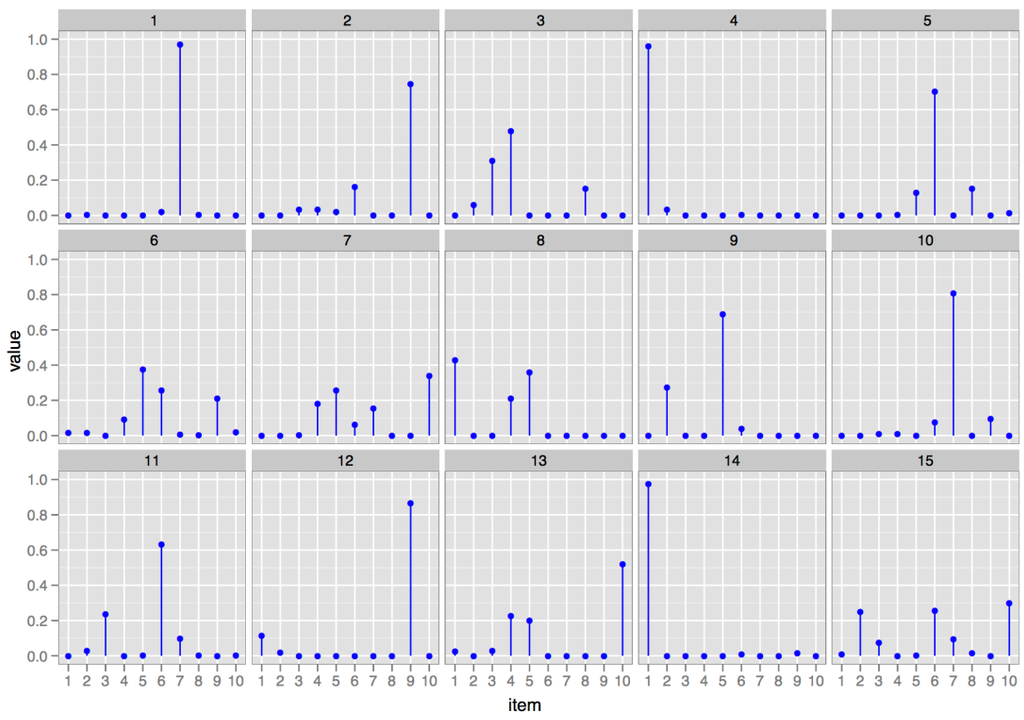
- Parameter controls mean shape and sparsity of topic proportions

LDA: Dirichlet Distribution

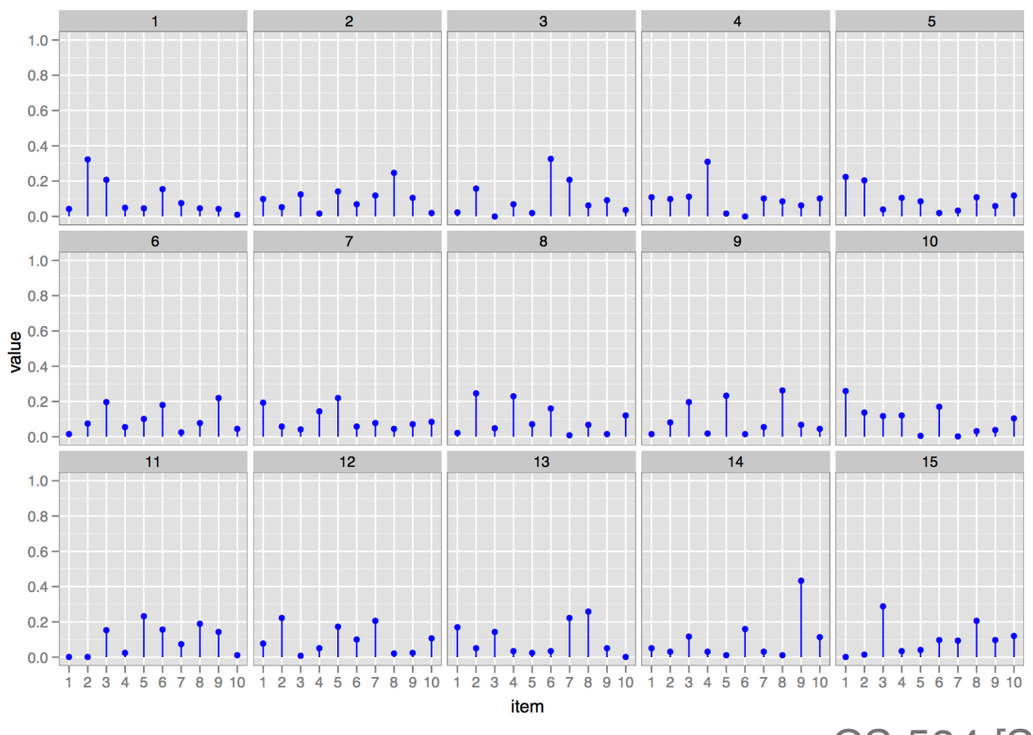
$\alpha = 0.001$



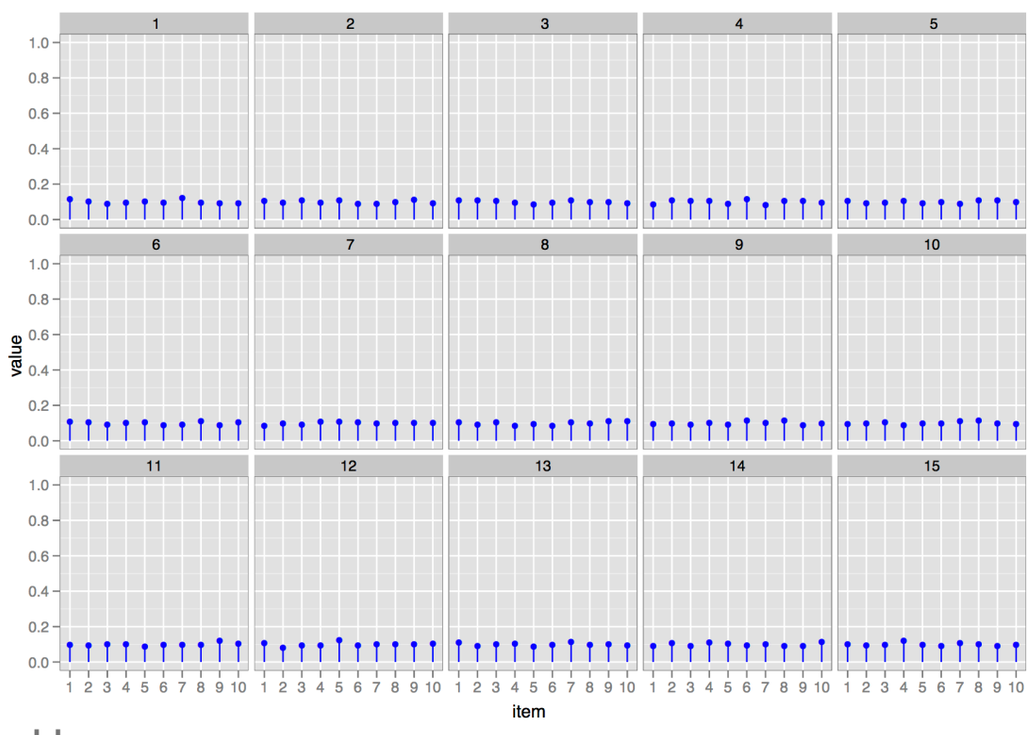
$\alpha = 0.1$



$\alpha = 1$



$\alpha = 100$



LDA: Likelihood

- Posterior distribution given document

$$p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \alpha, \boldsymbol{\beta}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta})}{p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta})}$$

- Conditional independence from graphical model:

$$p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = p(\mathbf{w}|\mathbf{z}, \boldsymbol{\beta})p(\mathbf{z}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\alpha})$$

$$p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \left(\frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1} \right) \prod_{i=1}^{N} \beta_{z_n, w_n} \theta_{z_n}$$

Exact inference is intractable!

LDA: Approximate Inference

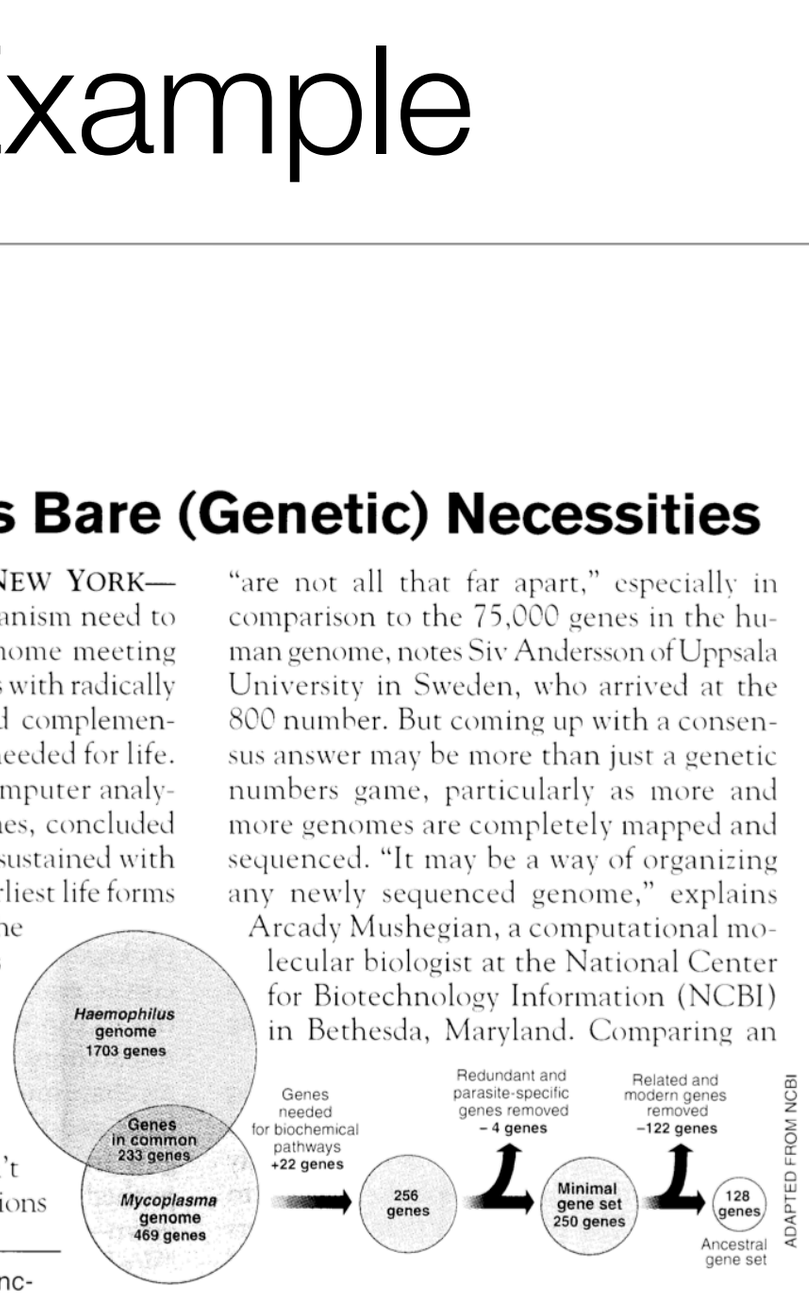
- Mean field variational methods (Blei et al., 2001, 2003)
- Expectation propagation (Minka and Lafferty, 2002)
- Collapsed Gibbs sampling (Griffiths and Steyvers, 2002)
- Collapsed variational inference (Teh et al., 2006)
- Online variational inference (Hoffman et al., 2010)

LDA: Example

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.

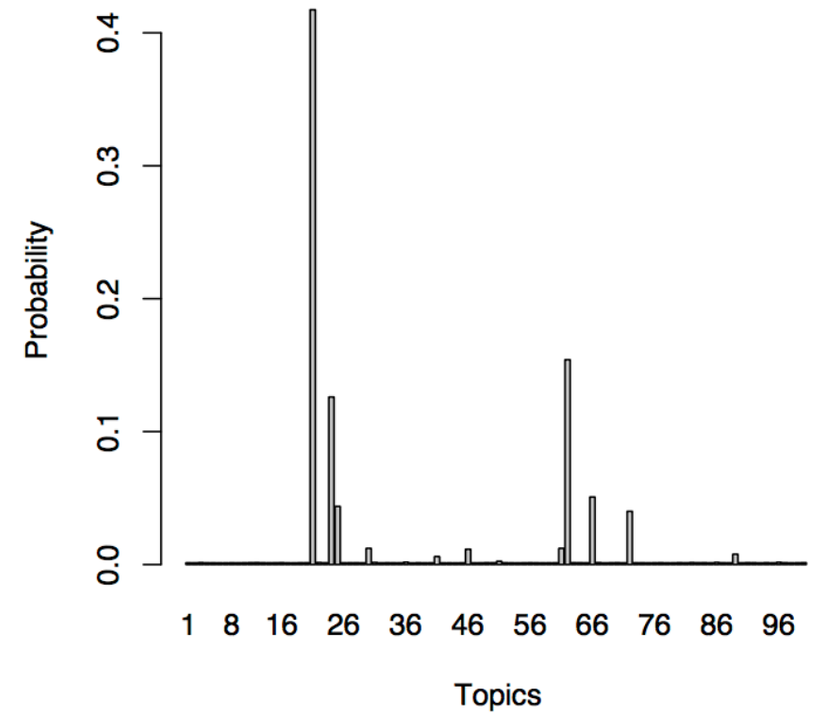
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just 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



LDA: Example

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

LDA: Example

"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

LDA: Example

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.

LDA: Why Does It Work?

- Word probabilities are maximized by dividing words among the topics
  - Enough to find clusters of co-occurring words
- Dirichlet on topic proportions can encourage sparsity — document is penalized for using many topics
  - Think of it as softening strict definition of “co-occurrence” in a mixture model
  - Leads to set of terms that more tightly co-occur

Topic Model: Frequentist Algorithms

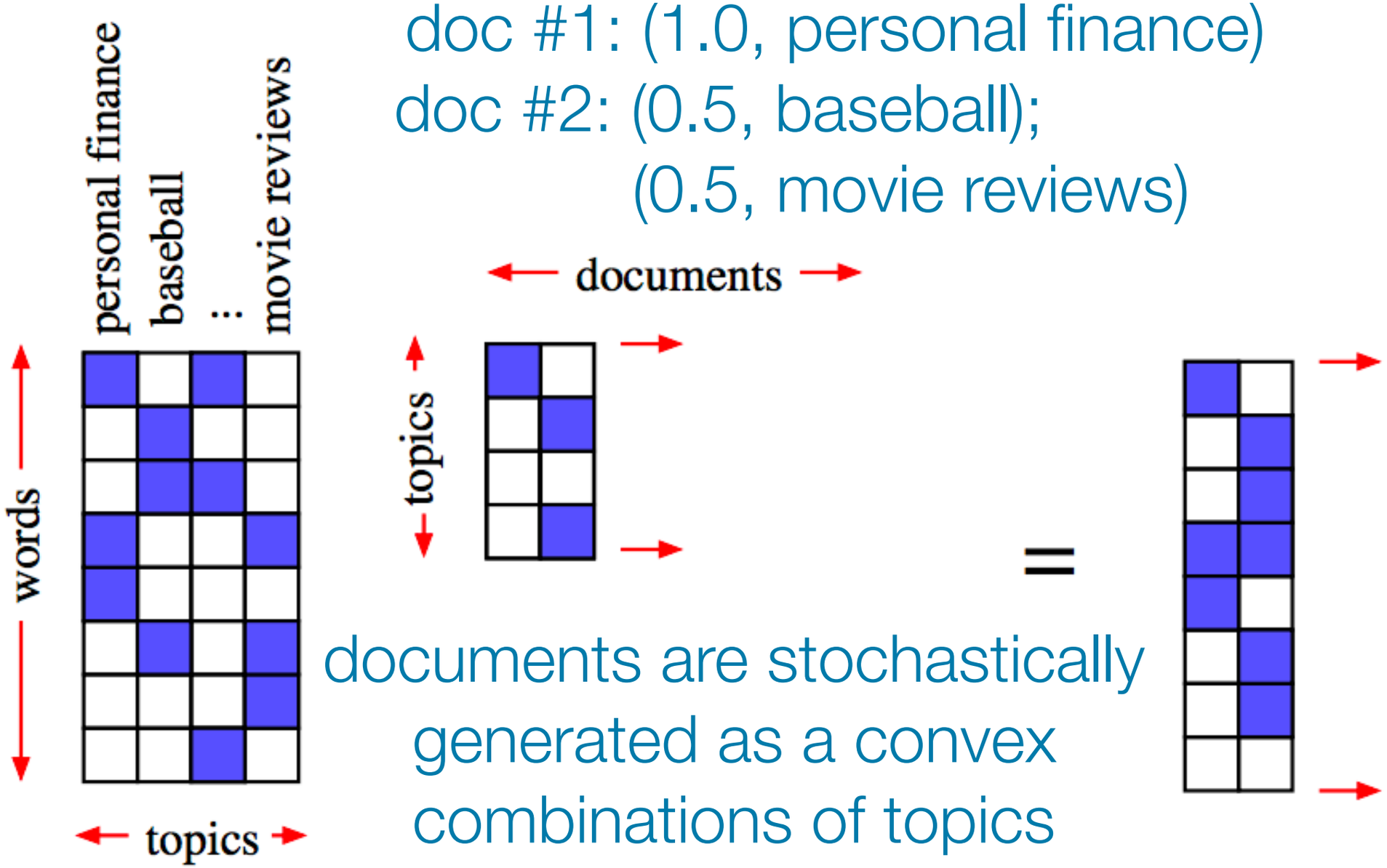
- Maximum likelihood: Find parameters that maximize the likelihood of generating the observed data — hard to compute

- Spectral method: Compute SVD of matrix — singular vectors are orthonormal

Non-negative matrix factorization?

Topic Model

doc #1: (1.0, personal finance)
doc #2: (0.5, baseball);
(0.5, movie reviews)

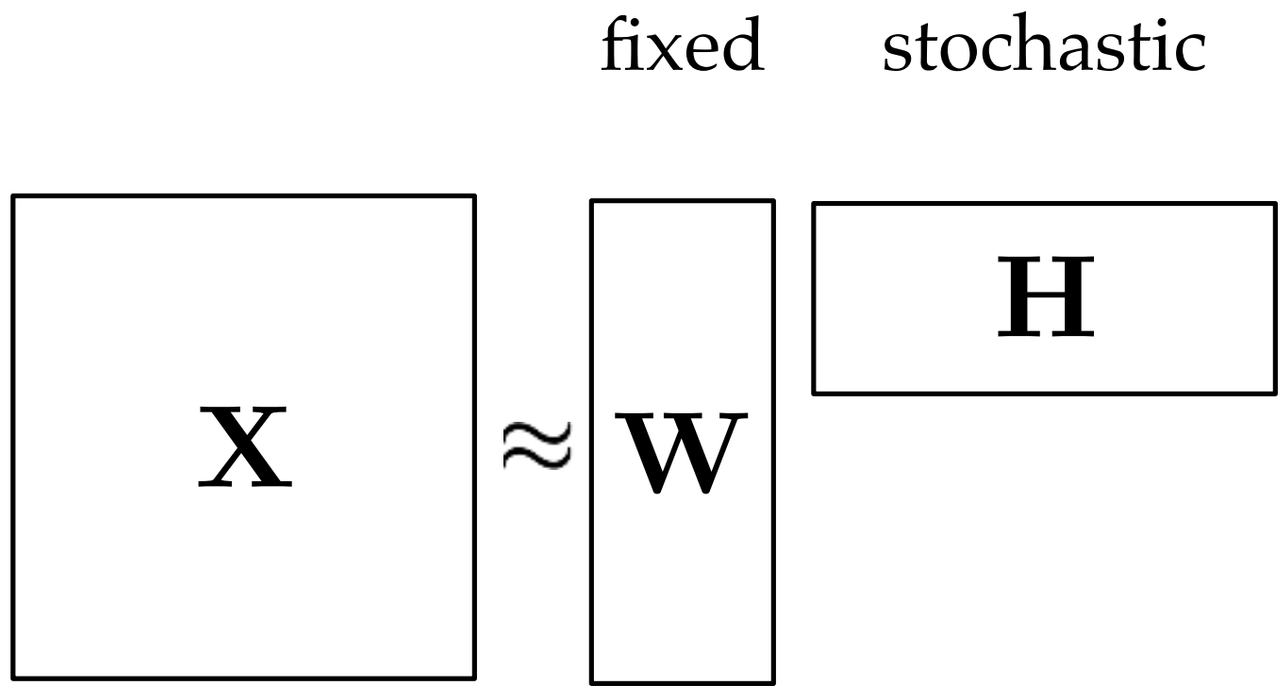


documents are stochastically generated as a convex combinations of topics

W H X

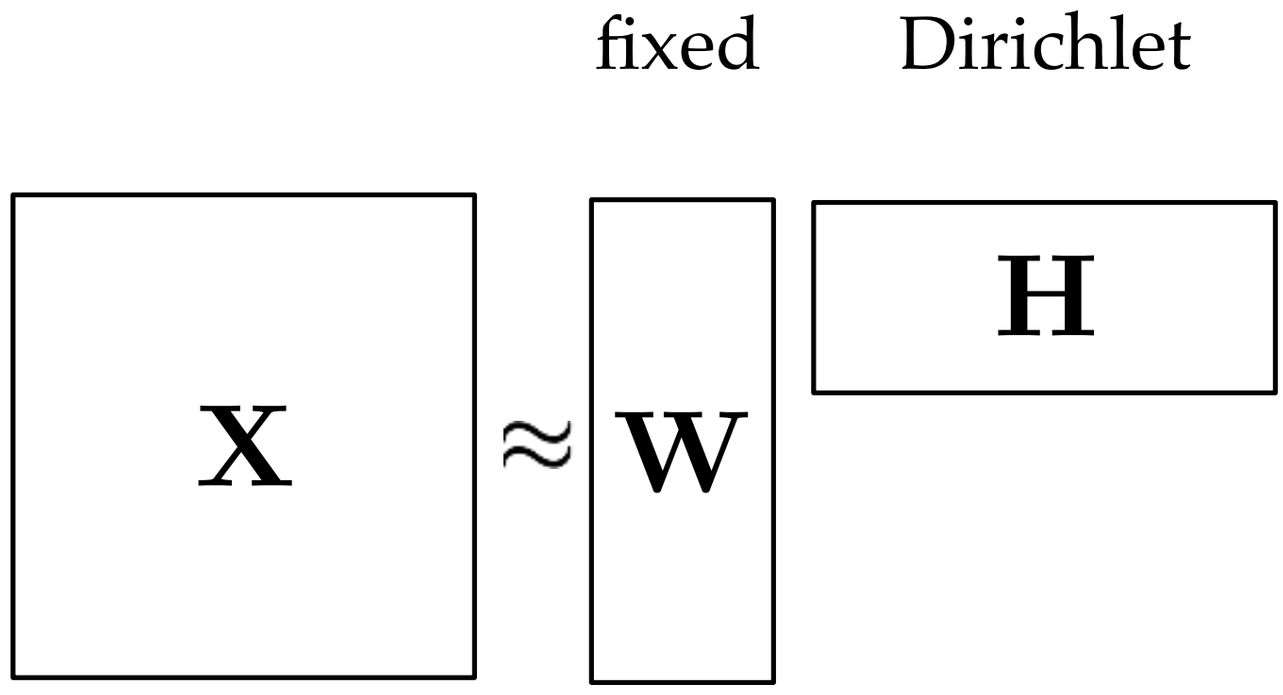
Pure Topics

- One topic per document



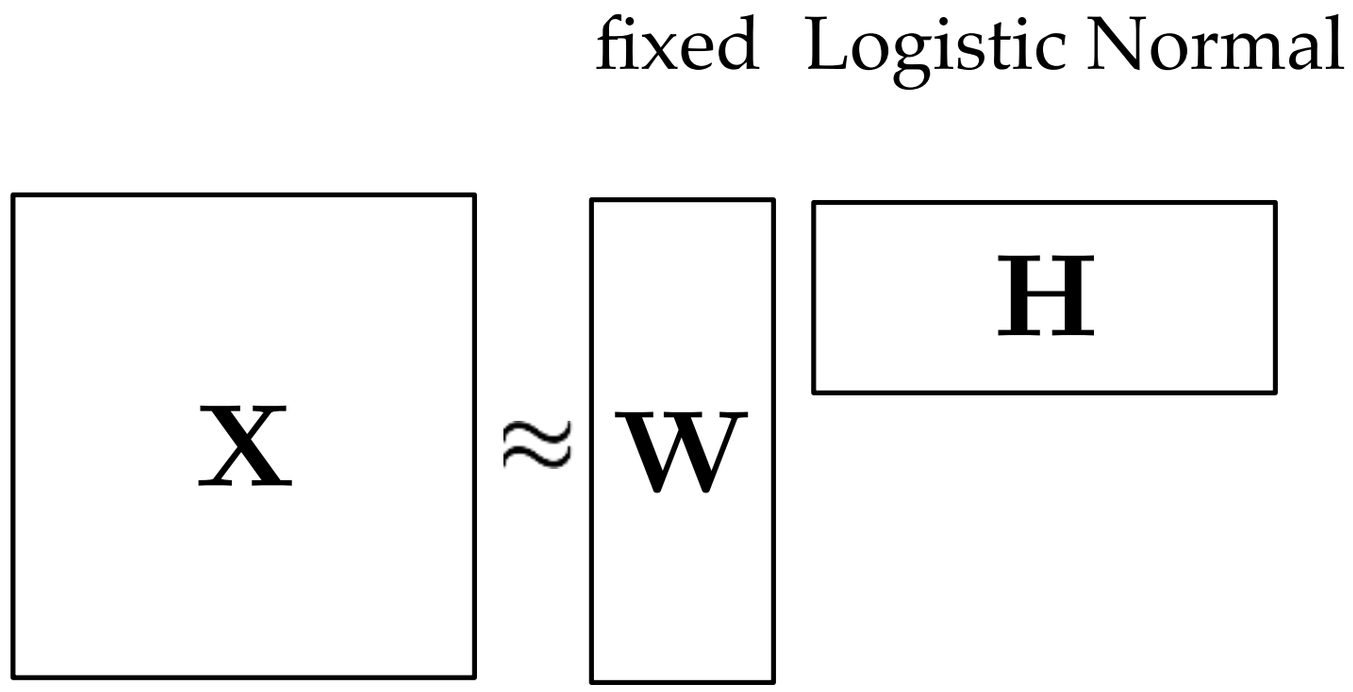
LDA

- Developed by Blei, Ng, and Jordan
- Assumes independence between topics
- Representation drawn from Dirichlet distribution



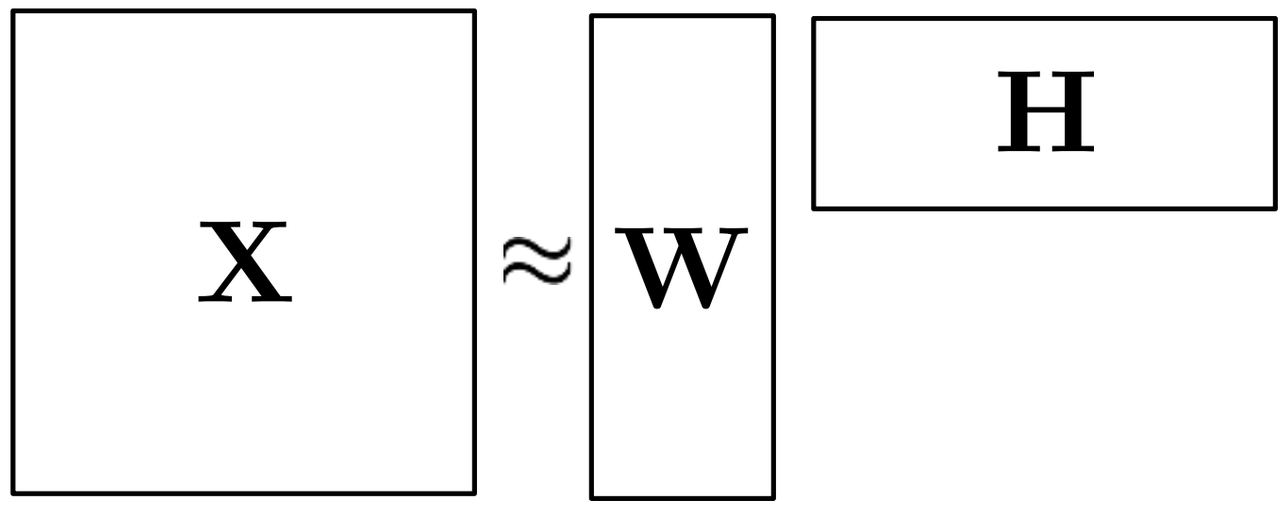
Correlated Topic Models

- Developed by Blei and Lafferty
- Allows correlation between topics
- Covariance matrix of logistic normal models topic correlations



Pachinko Allocation Model

- Developed by Li and McCallum
- Models correlations between topics by uncovering thematic structure
- Extension of LDA

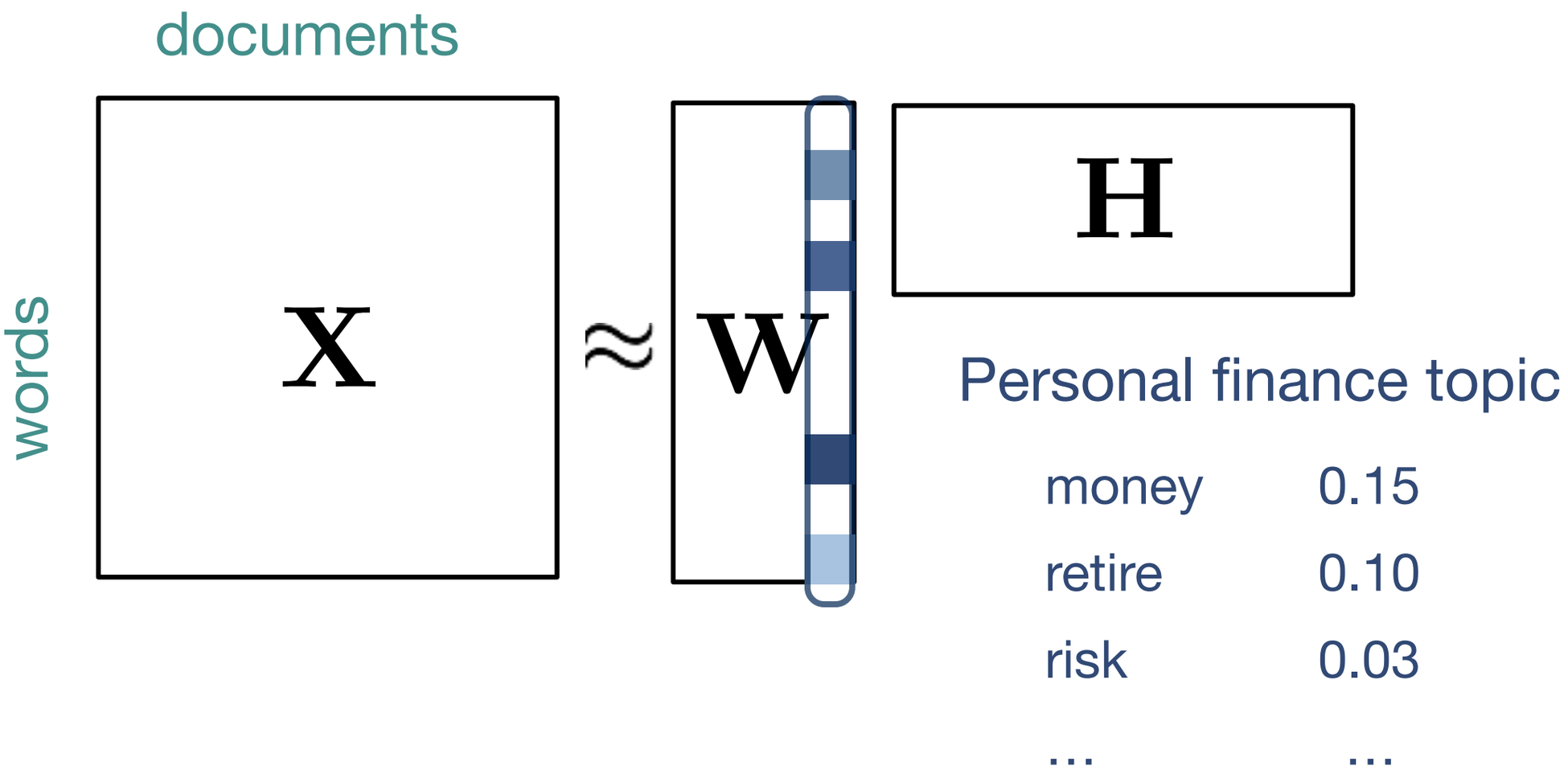


Models differ only in how H is generated!

Review: NMF

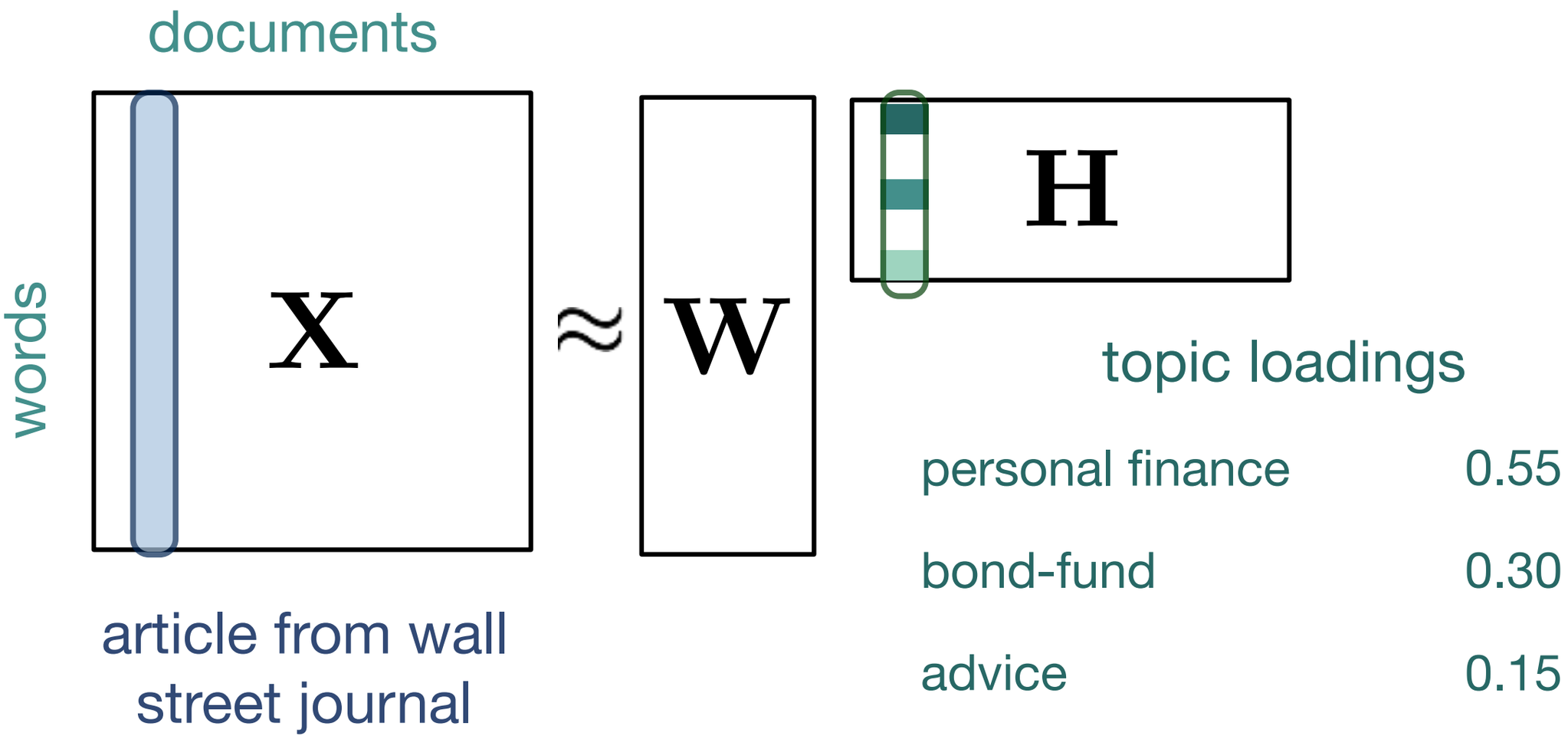
- Low rank approximation to original matrix
- Both **W** and **H** are nonnegative
- Empirically induces sparsity
- Improved interpretability (sum of parts representation)

Example: NMF



WLOG, assume columns of **W** and **H** sum to 1

Example: NMF



WLOG, assume columns of $\mathbf{W}$ and $\mathbf{H}$ sum to 1

NMF: Alternating Minimization

- Known to fail on worst-case inputs (stuck in local optima)
- Highly sensitive to:
  - Cost function
  - Update procedure
  - Regularization

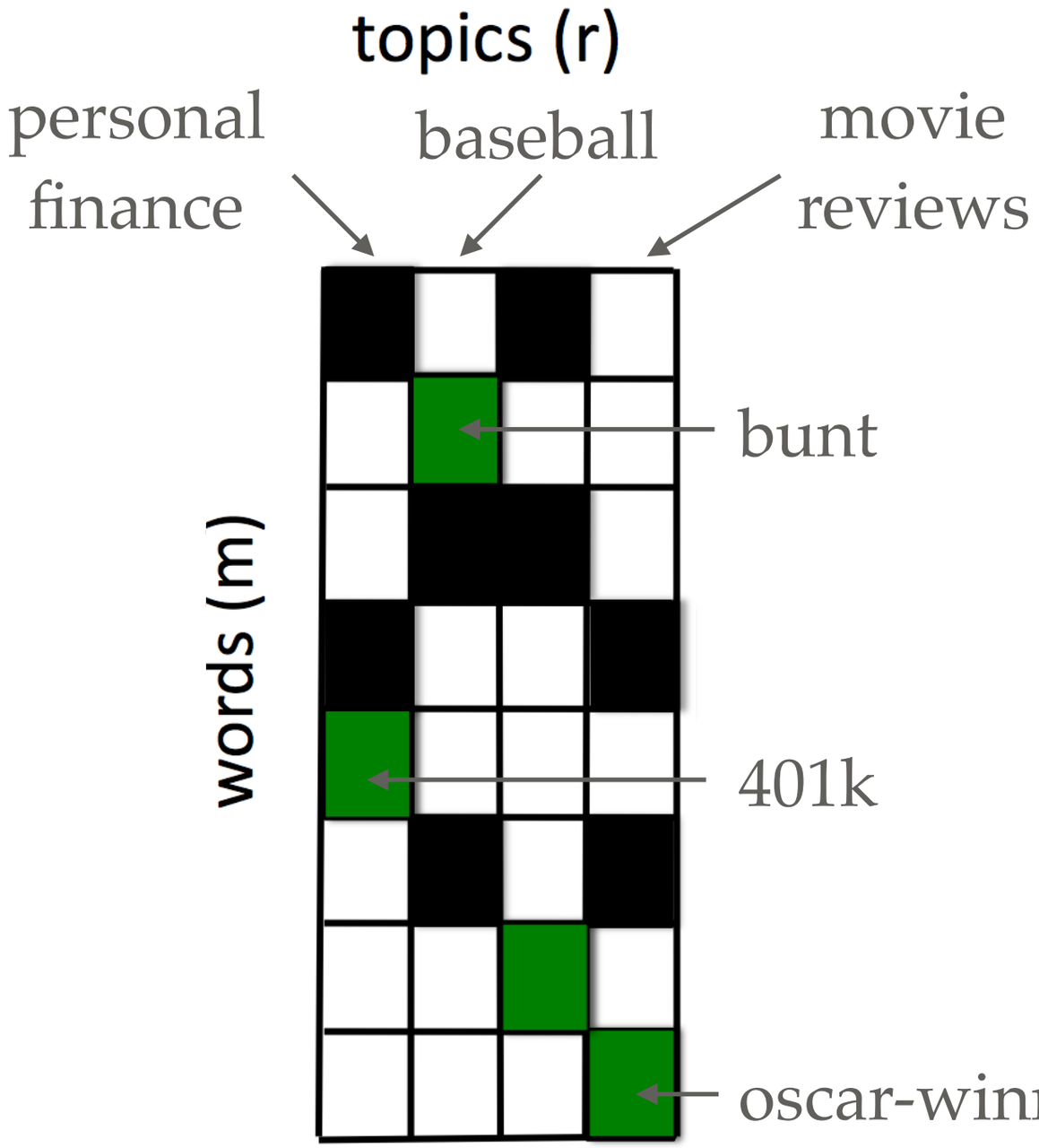
Can there be an efficient algorithm that works on all inputs?

NMF: Worst-case Complexity

- Theorem [Vavasis '09]: It is NP-hard to compute NMF
- Theorem [Cohen & Rothblum '13]: Can solve NMF in time $(nm)^{O(nr+mr)}$
- Theorem [Arora, Ge, Kanna, Moitra 2012]: Can solve NMF in time $(nm)^{O(r^2)}$ yet any algorithm that runs in time $(nm)^{o(r)}$ would yield a $2^{o(n)}$ algorithm for 3-SAT

Are the instances we want to solve somewhat easier?

NMF: Separability and Anchor Words



If an **anchor word** occurs then the document is at least partially about the topic

W is **p-separable** if each topic has an anchor word that occurs with probability at least p

NMF: Complexity for Separability

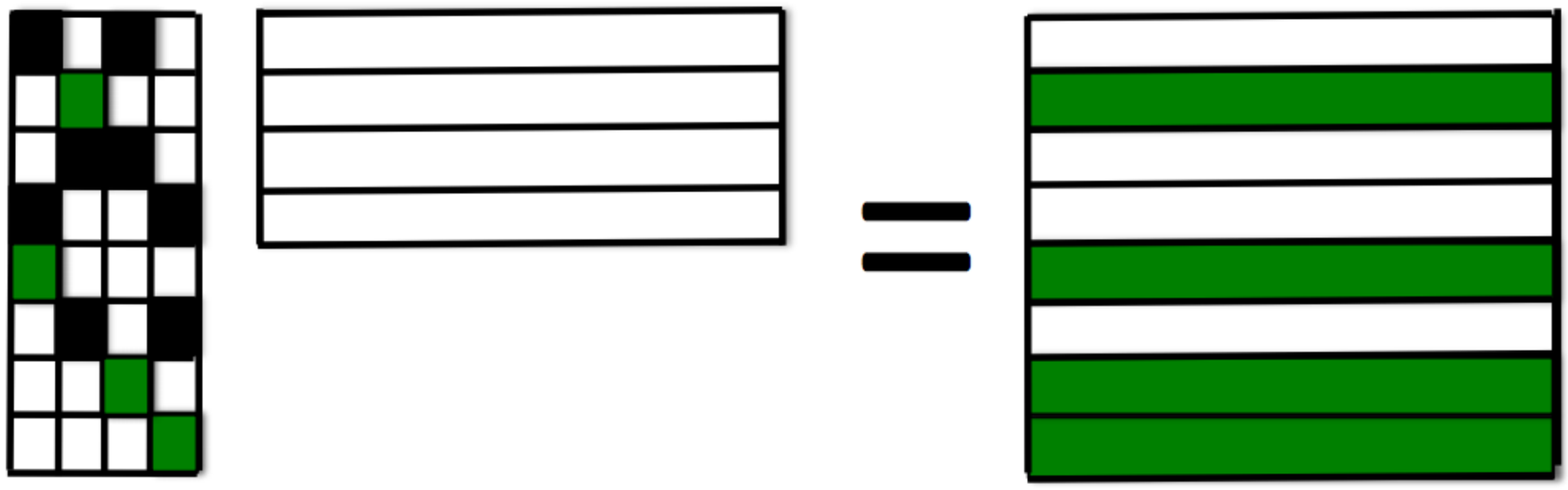
- Theorem [Arora, Ge, Kanna, Moitra 2012]: There is an $O(nmr+mr^{3.5})$ time algorithm for NMF when the topic matrix **W** is separable

- Theorem [Arora, Ge, Moitra 2012]: There is a polynomial time algorithm that learns the parameters of any topic model provided that the topic matrix **W** is p-separable

Algorithm is highly practical and runs orders of magnitude faster with nearly-identical performance as the current best (Gibbs sampling)

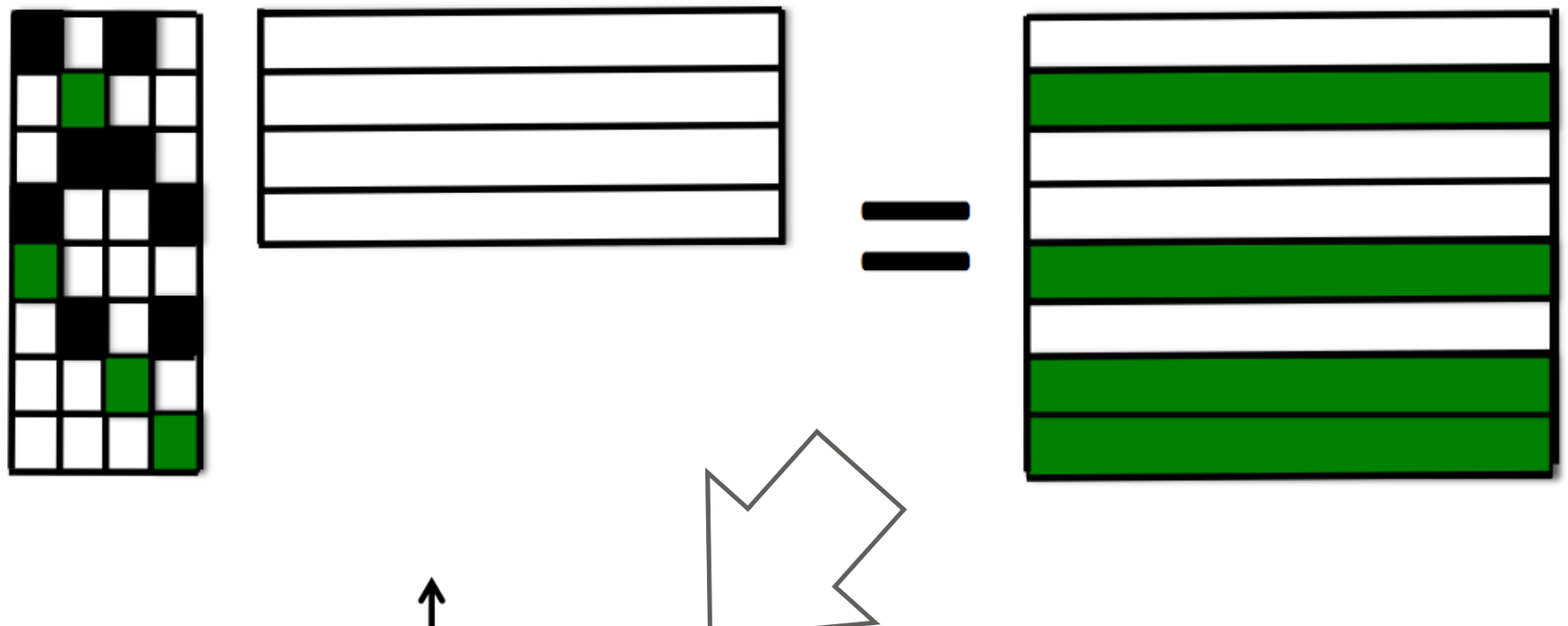
NMF: Anchor Words

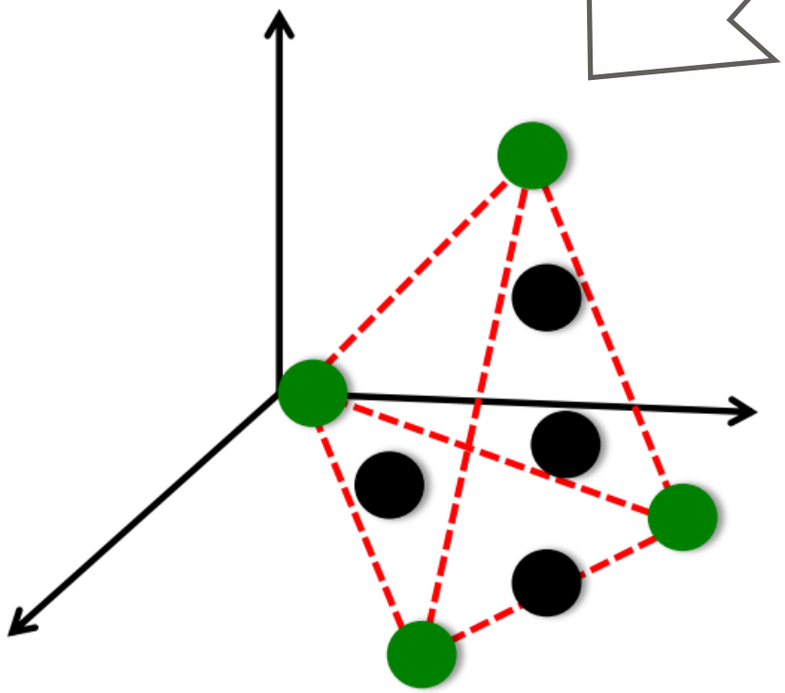
Observation: If **W** is separable, the rows of **H** appear as rows of **X**, and we just need to find the anchor words



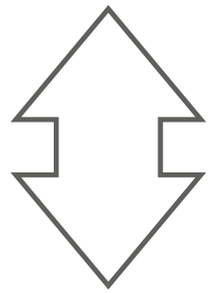
How to find anchor words?

NMF: Anchor Words <—> Vertices





Deleting a word changes the convex hull



Anchor word!

Topic Model: Anchor Word Algorithm

- Find anchor words: linear programming or a combinatorial distance-based algorithm

- Paste these vectors in as rows in **H**

- Find nonnegative **W** so that **WH = X** (convex programming)

Topic Model: Anchor Word Algorithm

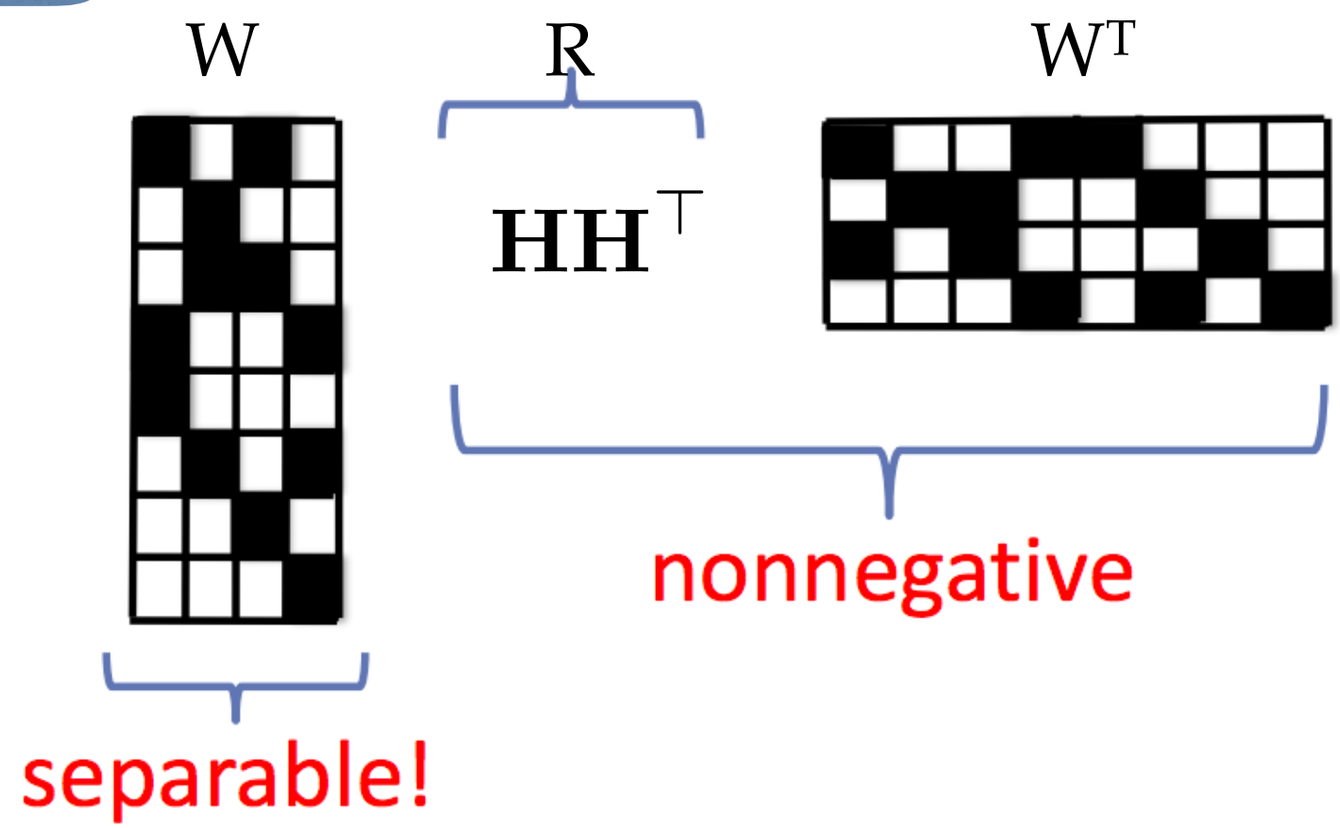
- What if documents are short? Can we still uncover **W**?
  - Given enough documents we can still find anchor words
  - Work with Gram matrix
- How do we use anchor words to find the rest of **W**?

Topic Model: Gram Matrix

Gram Matrix

$\hat{X}\hat{X}^\top$

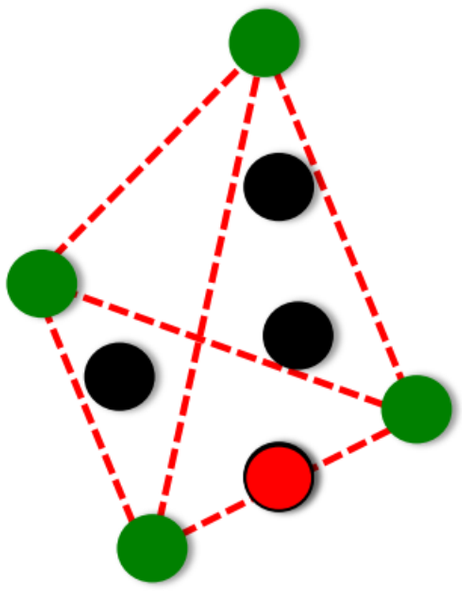
$$E[XX^\top] = WE[HH^\top]W^\top \rightarrow WRW^\top$$



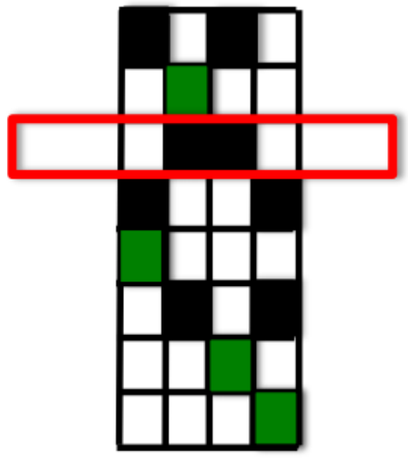
Anchor words are extreme rows of Gram matrix

Bayes Rule: Using Anchor Words

points are (normalized) rows of Gram matrix



w



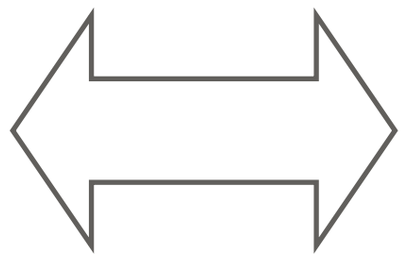
word #3: (0.5, anchor #2); (0.5, anchor #3)

Pr[topic|word #3]: (0.5, topic #2); (0.5, topic #3)

Bayes Rule: Using Anchor Words

What we have

Pr[topic | word]



What we want

Pr[word | topic]

Bayes rule!

$$\Pr[\text{word} \mid \text{topic}] = \frac{\Pr[\text{topic} \mid \text{word}] \Pr[\text{word}]}{\sum_{\text{word}'} \Pr[\text{topic} \mid \text{word}'] \Pr[\text{word}']}$$

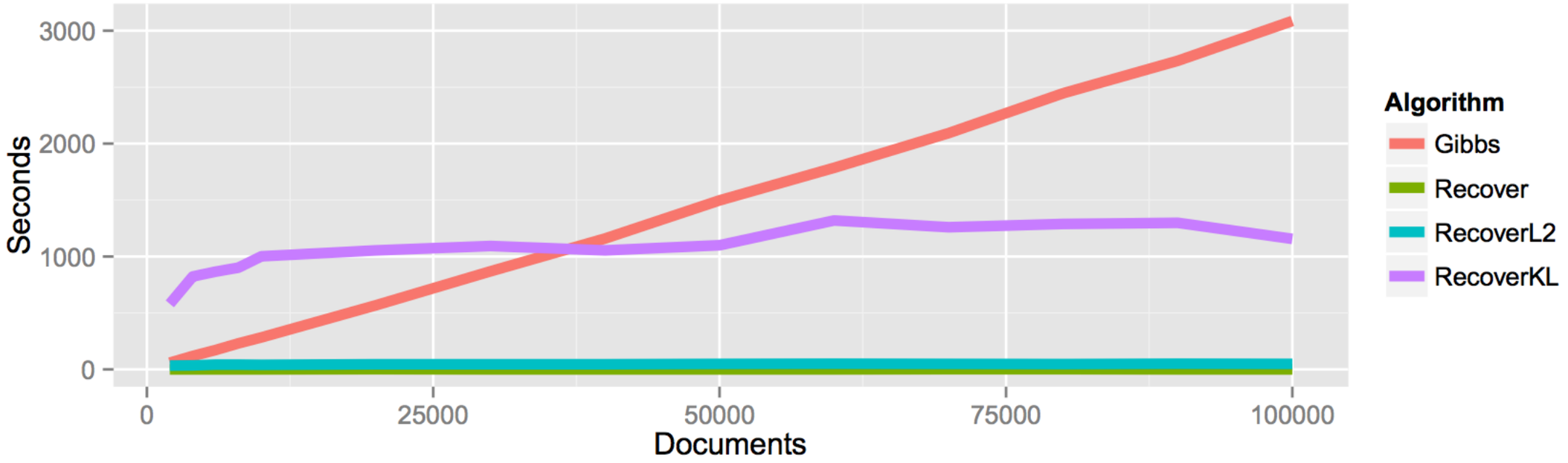
Topic Model: Anchor Word Algorithm

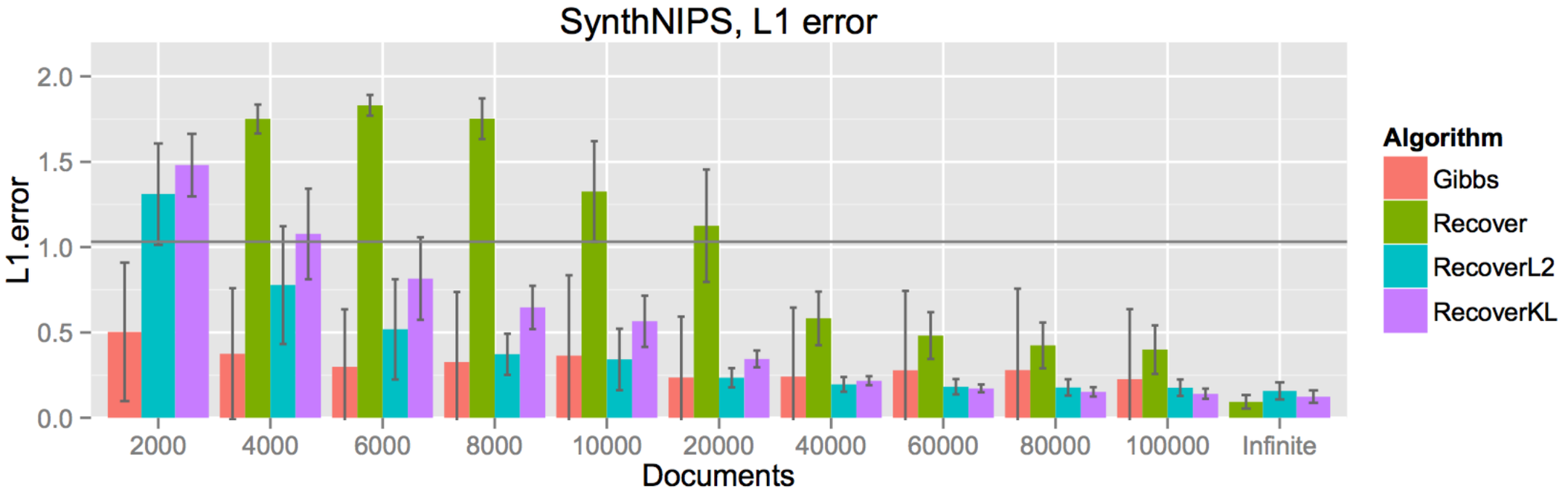
- Form Gram matrix and find anchor words
- Write each word as a convex combination of the anchor words to find P[topic | word]
- Compute W from the formula above
- This provably works on any topic model provided W is separable and R is non-singular

Experiment: Synthetic NIPS documents

- Train an LDA model on 1100 NIPS abstracts
- Use this model to run experiments
- Algorithm is 50x faster and performs nearly the same on all metrics when compared to MALLET

Experiment: Synthetic NIPS documents





Experiment: UCI Collection of NYT

- 300,000 New York Times articles with 30,000 distinct words

- Run time: 12 minutes (compared to 10 hours for MALLET and other state-of-the-art topic models)

- Topics are high quality

RecoverL2	run inning game hit season zzz_anaheim_angel
Gibbs	run inning hit game ball pitch
RecoverL2	father family **zzz_elian** boy court zzz_miami
Gibbs	zzz_cuba zzz_miami cuban zzz_elian boy protest
RecoverL2	**file** sport read internet email zzz_los_angeles
Gibbs	web site com www mail zzz_internet