Data Mining and Analysis

Text Mining: 2

CSCE 676 :: Fall 2019
Texas A&M University
Department of Computer Science & Engineering
Prof. James Caverlee

Resources

https://web.stanford.edu/~jurafsky/slp3/ — slides and readings on Sentiment Analysis

http://www.cs.virginia.edu/~hw5x/Course/TextMining-2019Spring/_site/lectures/ — slides from Hongning Wang's course on Text Mining

http://www.cs.columbia.edu/~blei/topicmodeling.html

Opinion mining and sentiment analysis by Pang and Lee

Probabilistic Topic Models (CACM) by Blei

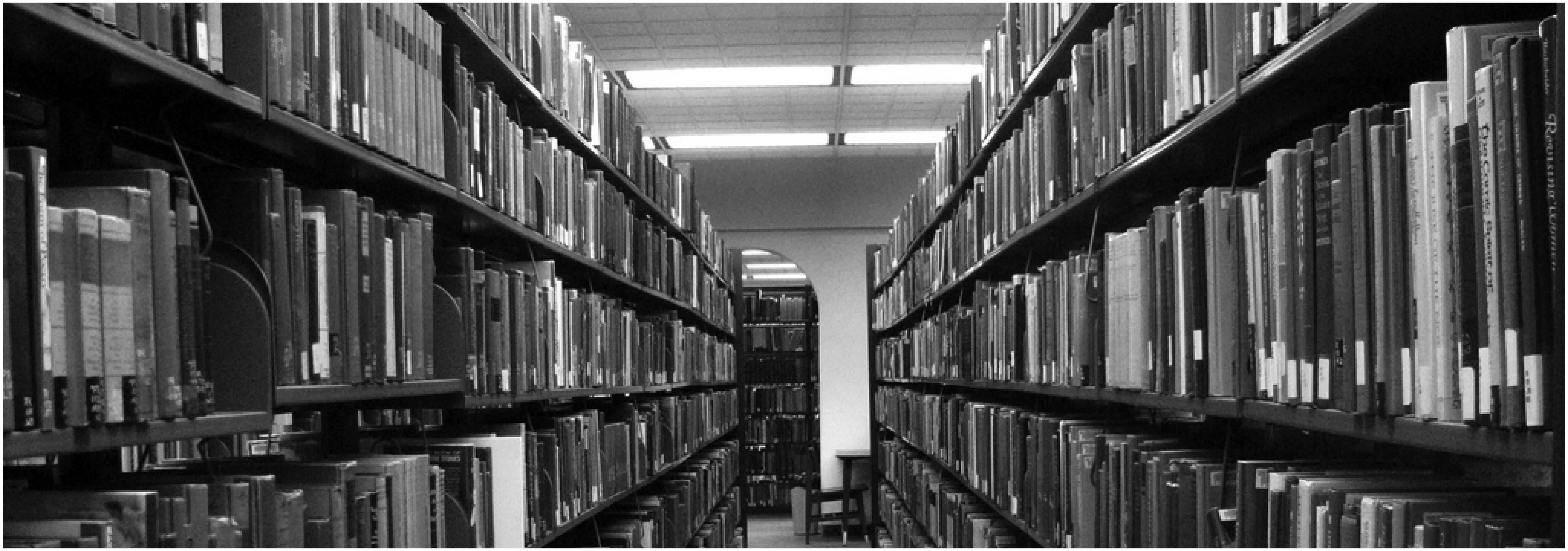
Latent dirichlet allocation by Blei, Ng, and Jordan

Probabilistic latent semantic analysis by Thomas Hofmann

Topic Models

http://www.cs.virginia.edu/~hw5x/Course/TextMining-2019Spring/_site/lectures/

http://www.cs.columbia.edu/~blei/papers/Blei2012.pdf



Input: An unorganized collection of documents

Output: An organized collection, and a description of how

Blei. Probabilistic Topic Models: Origins and Challenges (2013 Topic Modeling Workshop at NIPS)

What are Topics?

Topic = A broad semantically coherent theme, usually hidden in documents

- Examples: politics, sports, technology, etc.

How to Represent Topics?

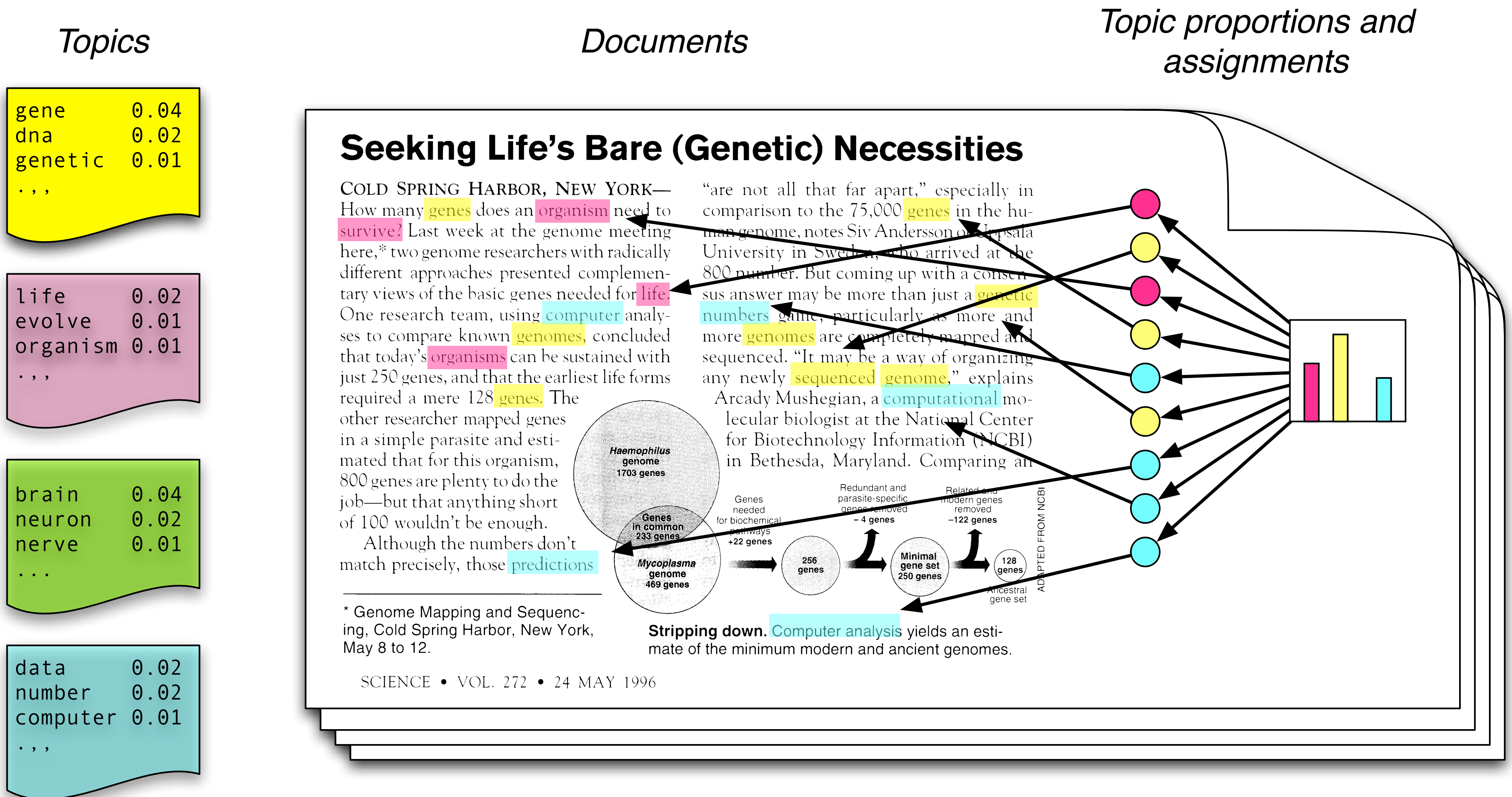
Typically as a probability distribution over words

Example for “texas a&m football”:

jimbo	**0.020**
aggies	**0.015**
touchdown	**0.011**
win	**0.009**

Remember: words could be unigrams, bigrams, phrases, …

Documents are a mix of topics



Consider the following “documents”

I like to eat broccoli and bananas.

I ate a banana and spinach smoothie for breakfast.

Chinchillas and kittens are cute.

My sister adopted a kitten yesterday.

Look at this cute hamster munching on a piece of broccoli.

= 30% broccoli, 15% bananas, 10% breakfast, 10% munching, … (at which point, you could interpret topic A to be about food)

= 20% chinchillas, 20% kittens, 20% cute, 15% hamster, … (at which point, you could interpret topic B to be about cute animals)

http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/

Motivating Question

How can we discover these topics? How about the word distributions?

Many applications would be enabled by discovering such topics

- Summarize themes/aspects
- Facilitate navigation/browsing
- Retrieve documents
- Segment documents
- Many other text mining tasks

Topic Models

Topic: a multinomial distribution over words

Document: a mixture of topics

- A document is "generated" by first sampling topics from some prior distribution
- Each time, sample a word from a corresponding topic
- Many variations of how these topics are mixed

Topic modeling

- Fitting the probabilistic model to text
- Answer topic-related questions by computing various kinds of posterior distributions
- e.g., p(topic | time), p(sentiment | topic)

Example: Scientific Topics over Time

1880	1890	1900	1910	1920	1930	1940
electric	electric	apparatus	air	apparatus	tube	air
machine	power	steam	water	tube	apparatus	tube
power	company	power	engineering	air	glass	apparatus
engine	steam	engine	apparatus	pressure	air	glass
steam	electrical	engineering	room	water	mercury	laboratory
two	machine	water	laboratory	glass	laboratory	rubber
machines	two	construction	engineer	gas	pressure	pressure
iron	system	engineer	made	made	made	small
battery	motor	room	gas	laboratory	gas	mercury
wire	engine	feet	tube	mercury	small	gas

1950	1960	1970	1980	1990	2000
tube	tube	air	high	materials	devices
apparatus	system	heat	power	high	device
glass	temperature	power	design	power	materials
air	air	system	heat	current	current
chamber	heat	temperature	system	applications	gate
instrument	chamber	chamber	systems	technology	high
small	power	high	devices	devices	light
laboratory	high	flow	instruments	design	silicon
pressure	instrument	tube	control	device	material
rubber	control	design	large	heat	technology

Simplest Case: 1 topic + 1 "background"

B

General Background English Text

Assume words in d are from <u>two</u> topics:
1 topic + 1 background

d

Text mining paper

θ_B

the 0.03
a 0.02
is 0.015
we 0.01
...
food 0.003
computer 0.00001
...
text 0.000006
...

θ_d

the 0.031
a 0.018
...
text 0.04
mining 0.035
association 0.03
clustering 0.005
computer 0.0009
...
food 0.000001
...

How can we "get rid of" the common words from the topic to make it more discriminative?

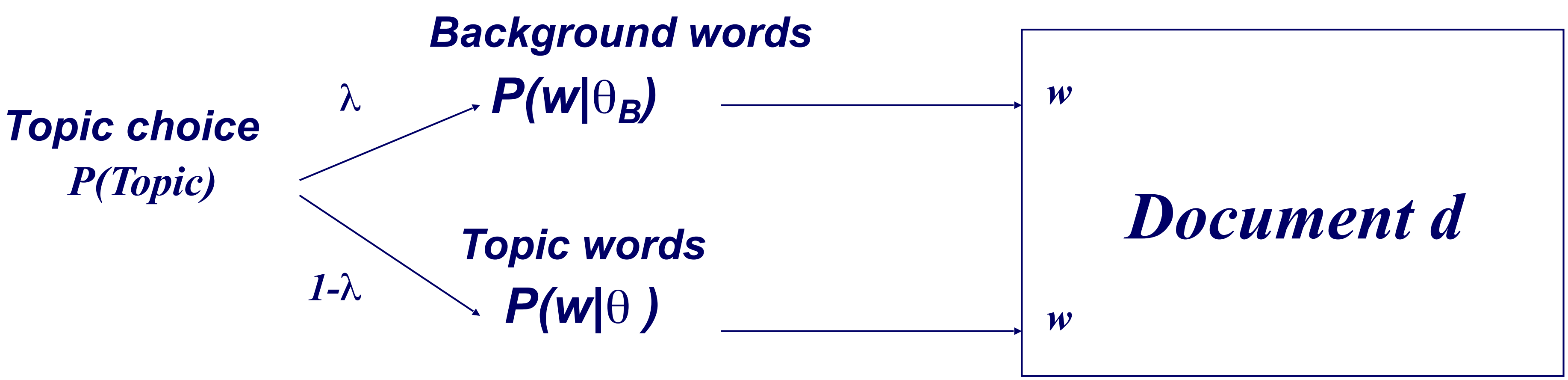
Background Topic: $p(w|\theta_B)$

Document Topic: $p(w|\theta_d)$

The Simplest Case: One Topic + One Background Model

Assume $p(w|\theta_B)$ and λ are *known*

λ = mixing proportion of background topic in d



$$p(w) = \lambda p(w|\theta_B) + (1-\lambda) p(w|\theta)$$

$$\log p(d|\theta) = \sum_{w \in V} c(w,d) \log[\lambda p(w|\theta_B) + (1-\lambda) p(w|\theta)]$$

Expectation Maximization $\hat{\theta} = \arg\max_{\theta} \log p(d|\theta)$

How to Estimate θ?

Known Background $p(w|\theta_B)$

the 0.2
a 0.1
we 0.01
to 0.02
…
text 0.0001
mining 0.00005
…

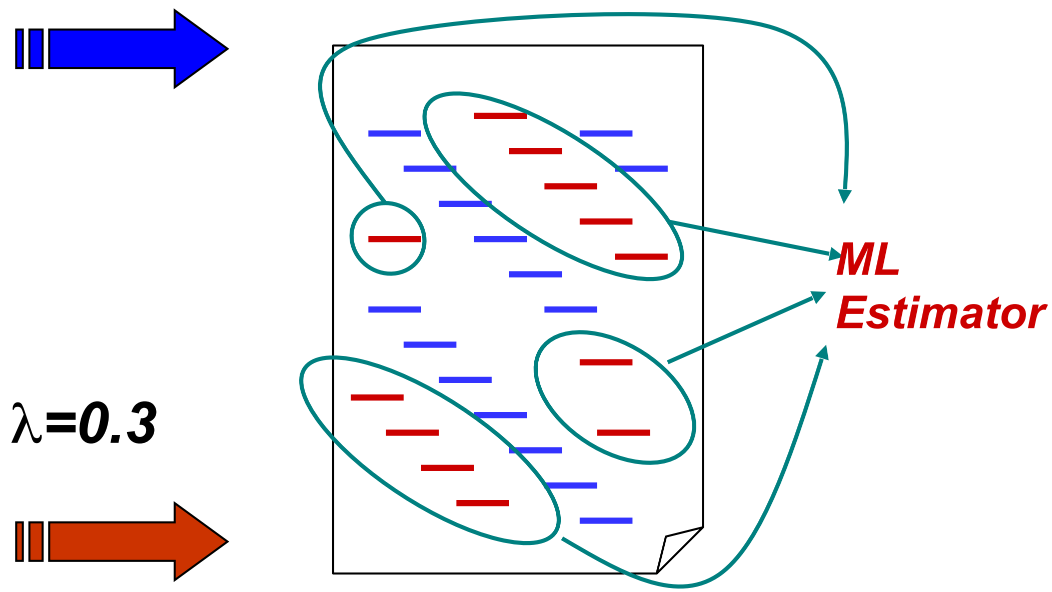
Unknown topic $p(w|\theta)$ for "Text mining"

…
text =?
mining =?
association =?
word =?
…

λ=0.7

λ=0.3

Observed words



Suppose we know the identity/label of each word ...

But we don't!

We guess the topic assignments

Assignment ("hidden") variable: $z_i \in$ {1 (background), 0(topic)}

	z_i
the	*1*
paper	*1*
presents	*1*
a	*1*
text	*0*
mining	*0*
algorithm	*0*
the	*1*
paper	*0*
…	*…*

Suppose the parameters are all known, what's a reasonable guess of z_i?

- depends on λ

- depends on $p(w|\theta_B)$ and $p(w|\theta)$

$$p(z_i=1|w_i)=\frac{p(z_i=1)p(w|z_i=1)}{p(z_i=1)p(w|z_i=1)+p(z_i=0)p(w|z_i=0)}$$

$$=\frac{\lambda p(w|\theta_B)}{\lambda p(w|\theta_B)+(1-\lambda)p^{current}(w|\theta)}$$

E-step

$$p^{new}(w_i|\theta)=\frac{c(w_i,d)(1-p(z_i=1|w_i))}{\sum_{w'\in V}c(w',d)(1-p(z_i=1|w'))}$$

M-step

θ_B and θ are competing for explaining words in document d!

Initially, set $p(w|\theta)$ to some random values, then iterate …

An example of EM computation

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

Expectation-Step:
Augmenting data by guessing hidden variables

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

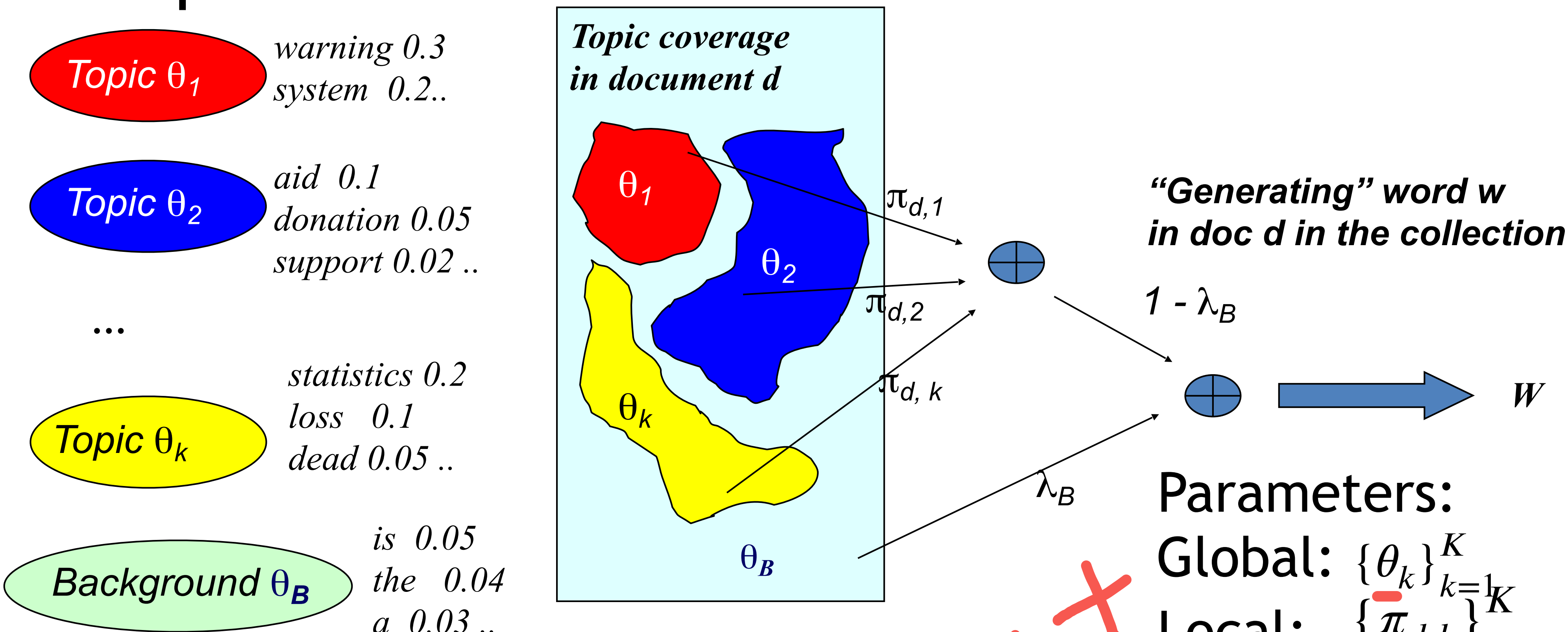
Maximization-Step
With the "augmented data", estimate parameters using maximum likelihood

Assume λ=0.5

Word	#	$P(w\|\theta_B)$	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	

Discover multiple topics in a collection

- Generalize the two topic mixture to *k* topics



Parameters:

Global: $\{\theta_k\}_{k=1}^K$

Local: $\{\pi_{d,k}\}_{d,k=1}^K$

Manual: λ_B

Probabilistic Latent Semantic Analysis
[Hofmann 99a, 99b]

- Topic: a multinomial distribution over words
- Document
  - Mixture of *k* topics
  - Mixing weights reflect the topic coverage
- Topic modeling
  - Word distribution under topic: $p(w|\theta)$
  - Topic coverage: $p(\pi|d)$

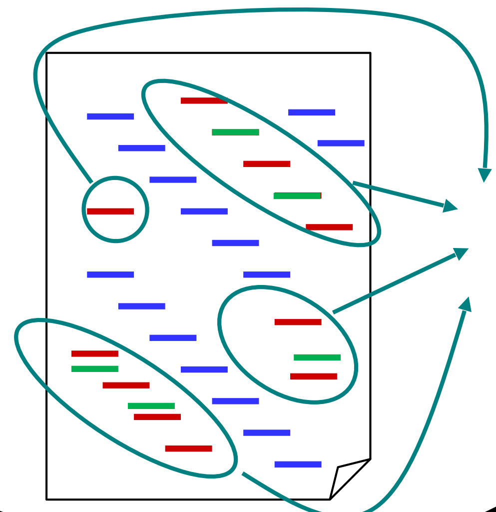
EM for estimating multiple topics

Known Background $p(w \mid \theta_B)$

the 0.2
a 0.1
we 0.01
to 0.02
…

E-Step:
Predict topic labels using Bayes Rule

Observed Words



M-Step:
ML Estimator based on "fractional counts"

Unknown topic model $p(w|\theta_1)$=?

"Text mining"

…
text =?
mining =?
association =?
word =?
…

…

Unknown topic model $p(w|\theta_2)$=?

"information retrieval"

…
information =?
retrieval =?
query =?
document =?
…

Parameter estimation

<u>*E-Step*</u>*:*

Word w in doc d is generated

- ***from topic j***
- ***from background***

Posterior: application of Bayes rule

$$p(z_{d,w}=j)=\frac{\pi_{d,j}^{(n)}p^{(n)}(w|\theta_j)}{\sum_{j'=1}^{k}\pi_{d,j'}^{(n)}p^{(n)}(w|\theta_{j'})}$$

$$p(z_{d,w}=B)=\frac{\lambda_B p(w|\theta_B)}{\lambda_B p(w|\theta_B)+(1-\lambda_B)\sum_{j=1}^{k}\pi_{d,j}^{(n)}p^{(n)}(w|\theta_j)}$$

<u>*M-Step*</u>*:*

Re-estimate

- ***mixing weights***
- ***word-topic distribution***

$$\pi_{d,j}^{(n+1)}=\frac{\sum_{w\in V}c(w,d)(1-p(z_{d,w}=B))p(z_{d,w}=j)}{\sum_{j'}\sum_{w\in V}c(w,d)(1-p(z_{d,w}=B))p(z_{d,w}=j')}$$

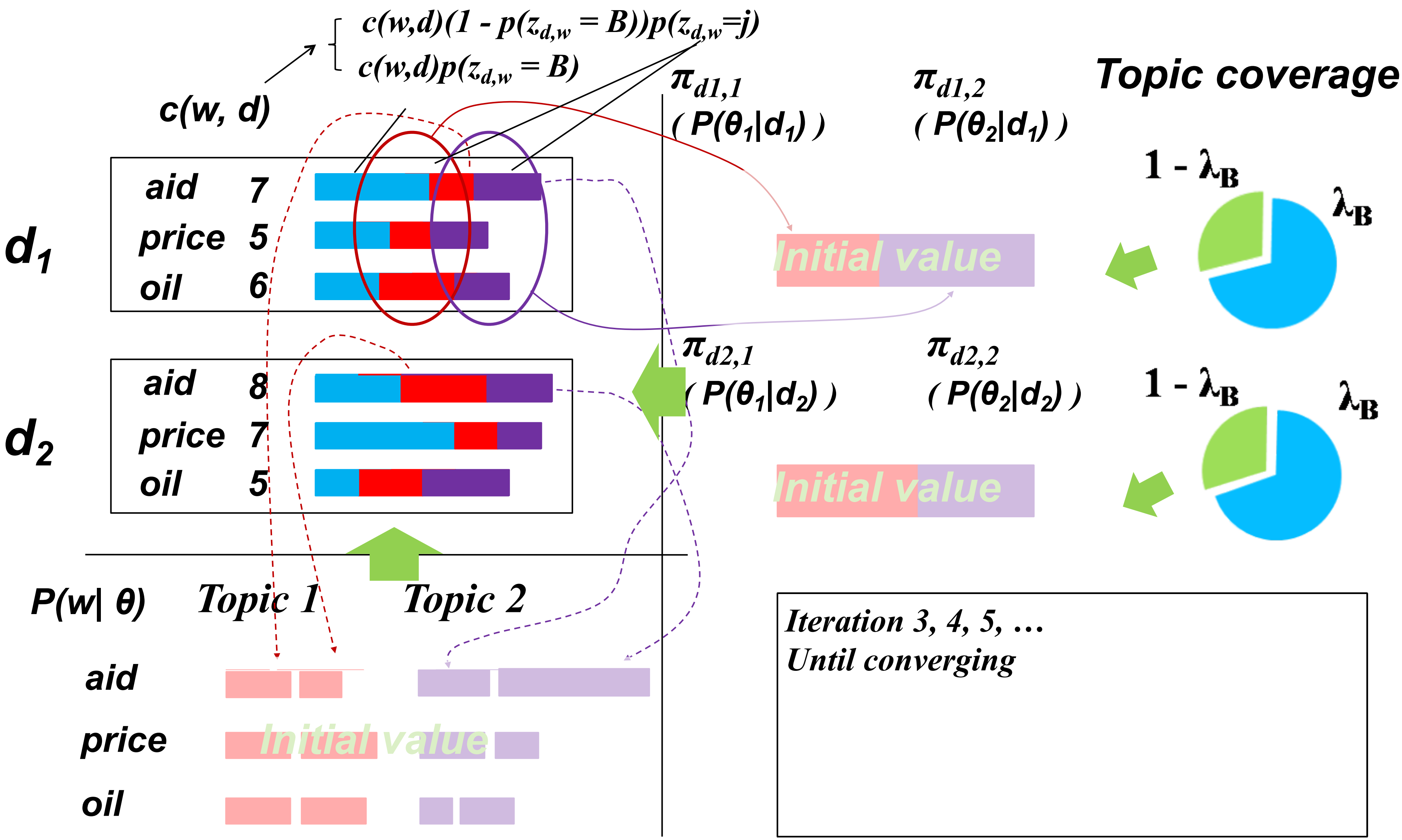
$$p^{(n+1)}(w|\theta_j)=\frac{\sum_{d\in C}c(w,d)(1-p(z_{d,w}=B))p(z_{d,w}=j)}{\sum_{w'\in V}\sum_{d\in C}c(w',d)(1-p(z_{d,w'}=B))p(z_{d,w'}=j)}$$

Sum over all docs in the collection

Fractional counts contributing to

- ***using topic j in generating d***
- ***generating w from topic j***

How the algorithm works



Sample pLSA topics from TDT Corpus [Hofmann 99b]

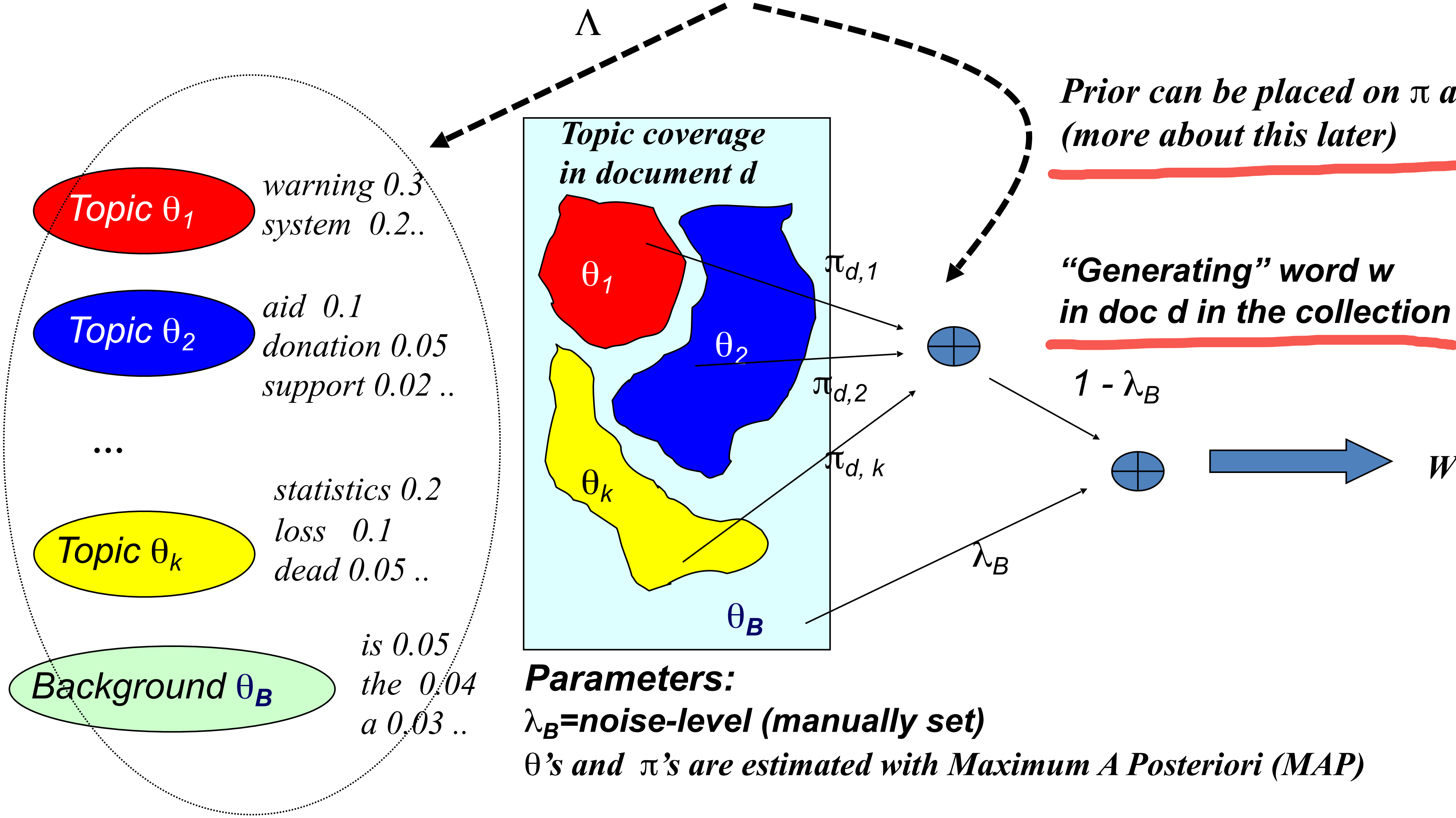
"plane"	"space shuttle"	"family"	"Hollywood"
plane	space	home	film
airport	shuttle	family	movie
crash	mission	like	music
flight	astronauts	love	new
safety	launch	kids	best
aircraft	station	mother	hollywood
air	crew	life	love
passenger	nasa	happy	actor
board	satellite	friends	entertainment
airline	earth	cnn	star

pLSA with prior knowledge

- What if we have some domain knowledge in mind
  - We want to see topics such as “battery” and “memory” for opinions about a laptop
  - We want words like “apple” and “orange” co-occur in a topic
  - One topic should be fixed to model background words (infinitely strong prior!)
- We can easily incorporate such knowledge as priors of pLSA model

Maximum a Posteriori (MAP) estimation

$$\Lambda^* = \arg\max_{\Lambda} p(\Lambda)\, p(Data \mid \Lambda)$$



Prior can be placed on π as well (more about this later)

"Generating" word w in doc d in the collection

Parameters:

λ_B=noise-level (manually set)

θ's and π's are estimated with Maximum A Posteriori (MAP)

MAP estimation

- Choosing conjugate priors
  - Dirichlet prior for multinomial distribution

Pseudo counts of w from prior θ'

$$p^{(n+1)}(w|\theta_j)=\frac{\sum_{d\in C}c(w,d)(1-p(z_{d,w}=B))p(z_{d,w}=j)+\mu p(w|\theta'_j)}{\sum_{w'\in V}\sum_{d\in C}c(w',d)(1-p(z_{d,w'}=B))p(z_{d,w'}=j)+\mu}$$

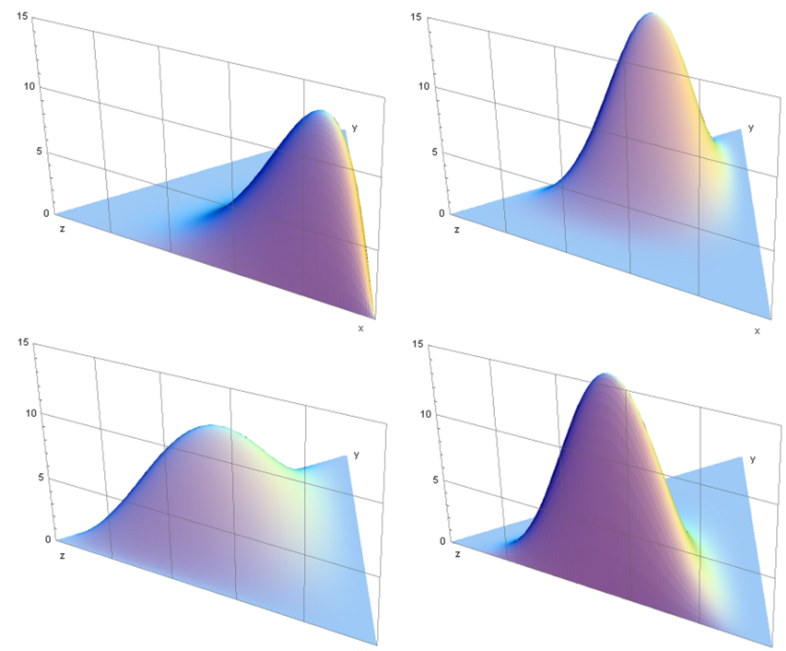
Sum of all pseudo counts

  - What if μ=0? What if μ=+∞?
  - A consequence of using conjugate prior is that the prior can be converted into “pseudo data” which can then be “merged” with the actual data for parameter estimation

Some background knowledge

- Conjugate prior
  - Posterior distribution 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



Prior as pseudo counts

Known Background $p(w \mid B)$

the 0.2
a 0.1
we 0.01
to 0.02
…

Unknown topic model $p(w|\theta_1)$=?

"Text mining"

…
text =?
mining =?
association =?
word =?
…

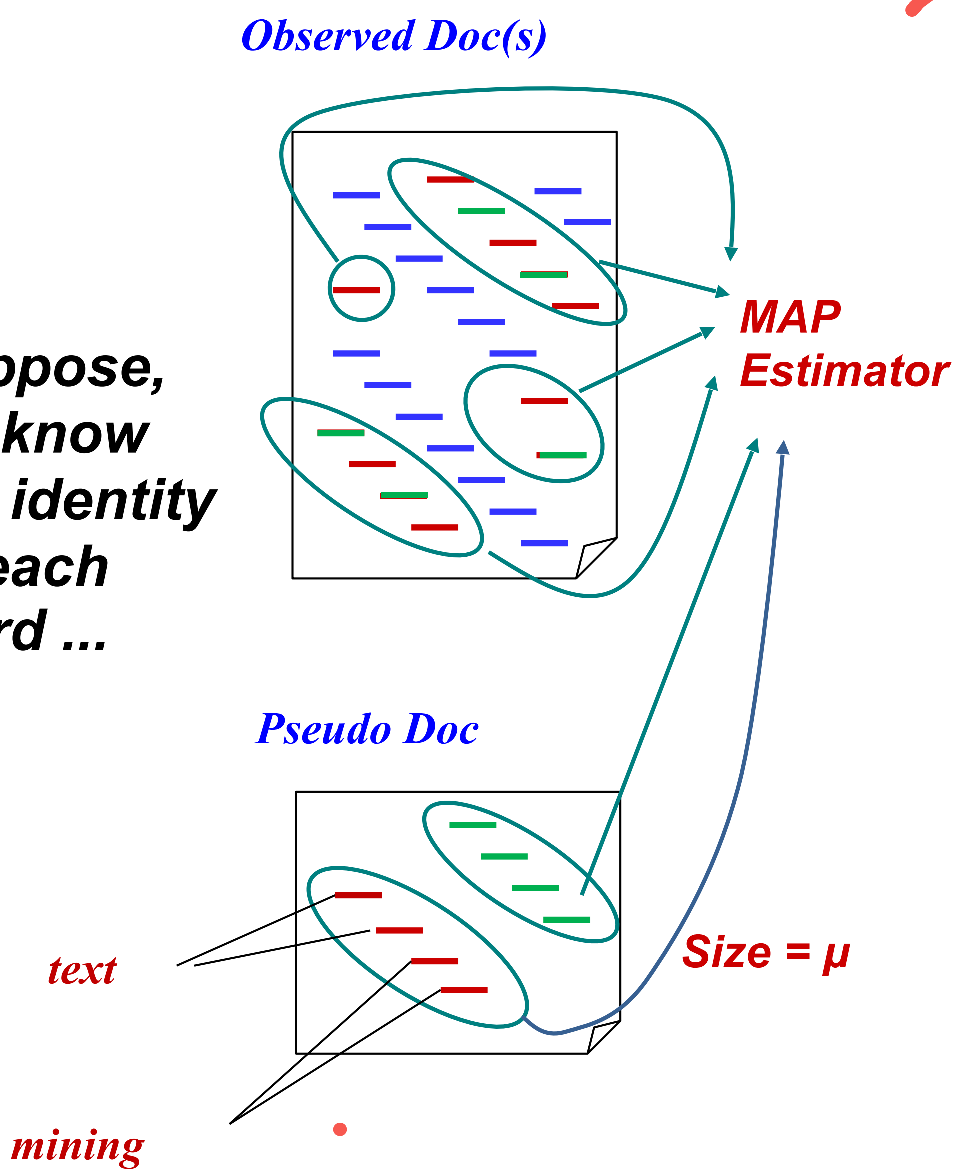
…

Unknown topic model $p(w|\theta_2)$=?

"information retrieval"

…
information =?
retrieval =?
query =?
document =?
…

Suppose, we know the identity of each word ...



Deficiency of pLSA

Then what is fully generative model???

- Not a fully generative model
  - Can't compute probability of a new document
    - Topic coverage $p(\pi|d)$ is per-document estimated
  - Heuristic workaround is possible
- Many parameters ➔ high complexity of models
  - Many local maxima
  - Prone to overfitting