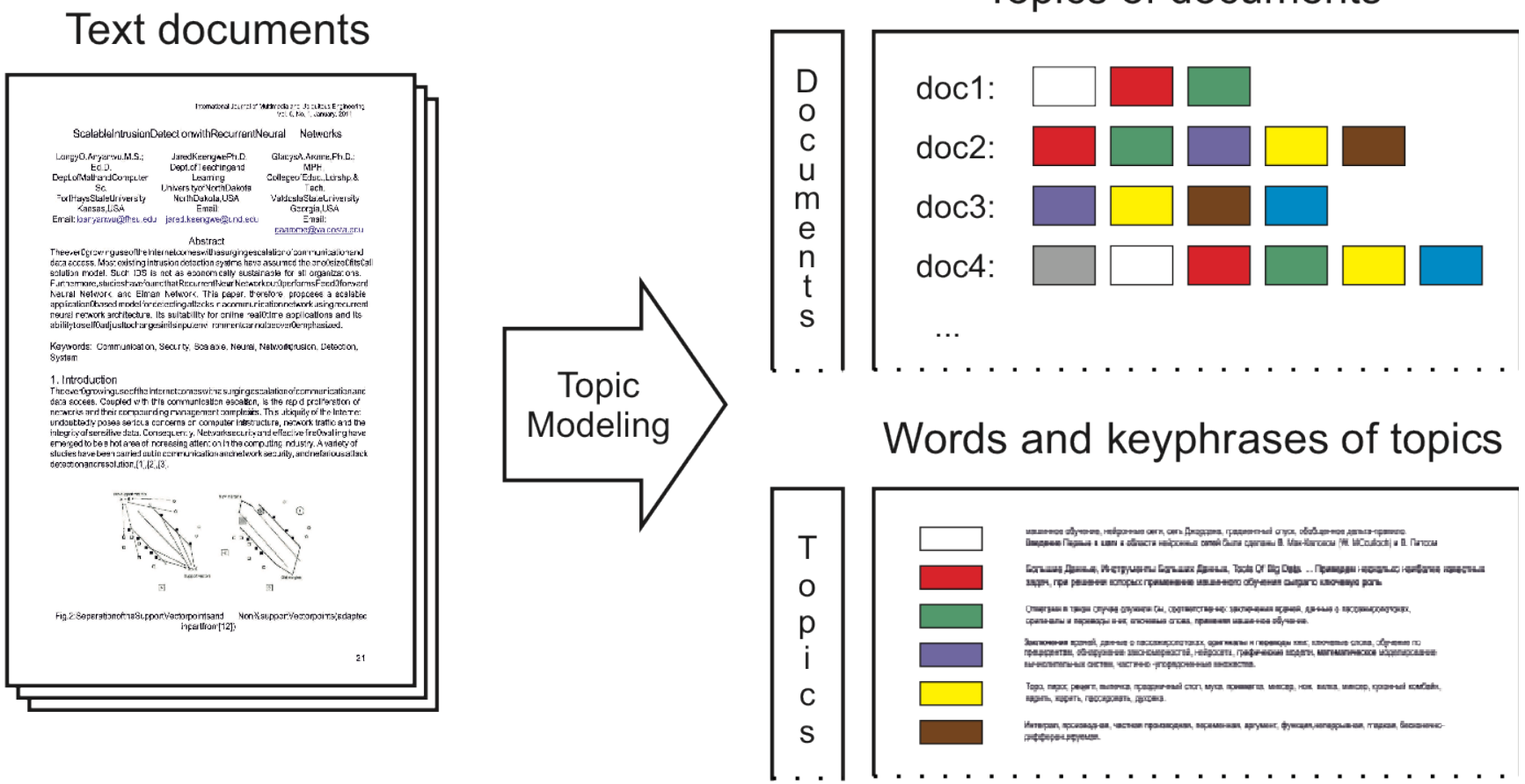
Анализ неструктурированных данных

Topic modeling: a way to navigate through text collections

Потапенко Анна Александровна

26 сентября 2018

From texts to topics



The formal task

Given:

- Collection of texts as bags-of-words:

  n_{wd} is a count of the word *w* in the document *d*

Find:

- Probabilities of word in topics:

  $\phi_{wt} = p(w|t)$

- Probabilities of topics in documents:

  $\theta_{td} = p(t|d)$

The formal task

Given:

- Collection of texts as bags-of-words:

  n_{wd} is a count of the word *w* in the document *d*

Find:

- Probabilities of word in topics:

  $\phi_{wt} = p(w|t)$ ⟵ **Definition of a topic!**

- Probabilities of topics in documents:

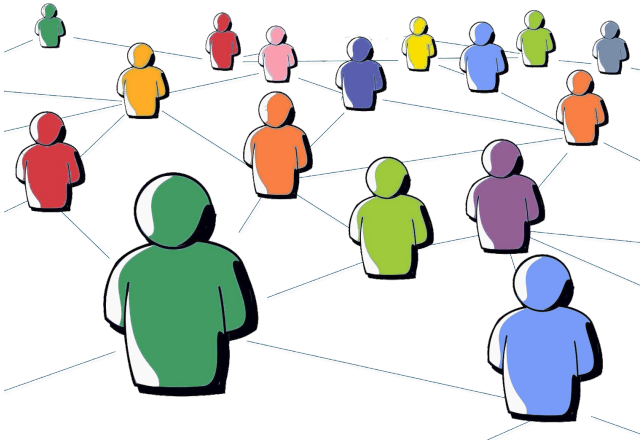
  $\theta_{td} = p(t|d)$

Where do we need that?

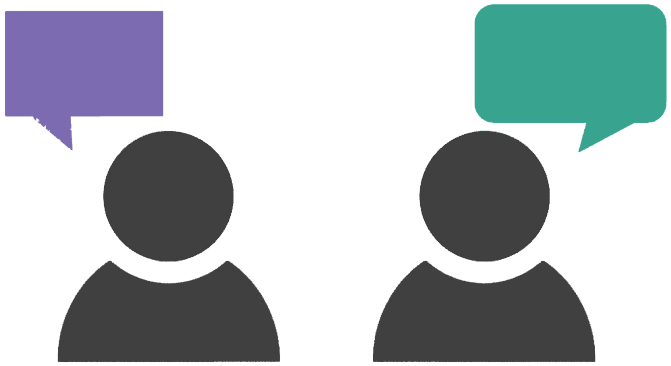
Exploration and navigation through large text collections

Where do we need that?

- Social network analysis



- Dialogue manager in chat-bots



Why do we need it?

Topic models provide hidden semantic representation of texts.

Many more applications:

- Categorization and classification of texts
- Document segmentation and summarization
- News flows aggregation and analysis
- Recommender systems
- Annotations for music, video and music
- Bioinformatics (genome annotation)
- Exploratory search
- …

Generative model of texts

Probabilistic Latent Semantic Analysis (PLSA):

$$p(w|d) = \sum_{t \in T} p(w|t,d)\, p(t|d) = \sum_{t \in T} p(w|t)\, p(t|d)$$

Notation:

- *w – word*
- *d – document*
- *t – topic*

Generative model of texts

Probabilistic Latent Semantic Analysis (PLSA):

$$p(w|d) = \sum_{t \in T} p(w|t,d)\, p(t|d) = \sum_{t \in T} p(w|t)\, p(t|d)$$

Law of total probability

$$p(w) = \sum_{t \in T} p(w|t)\, p(t)$$

Notation:

- *w – word*
- *d – document*
- *t – topic*

Generative model of texts

Probabilistic Latent Semantic Analysis (PLSA):

$$p(w|d) = \sum_{t \in T} p(w|t,d)\, p(t|d) = \sum_{t \in T} p(w|t)\, p(t|d)$$

Law of total probability

$$p(w) = \sum_{t \in T} p(w|t)\, p(t)$$

Assumption of conditional independence

$$p(w|t,d) = p(t|d)$$

Notation:

- w – *word*
- d – *document*
- t – *topic*

Generative model of texts

Topics

gene	0.04
dna	0.02
genetic	0.01
. . .	

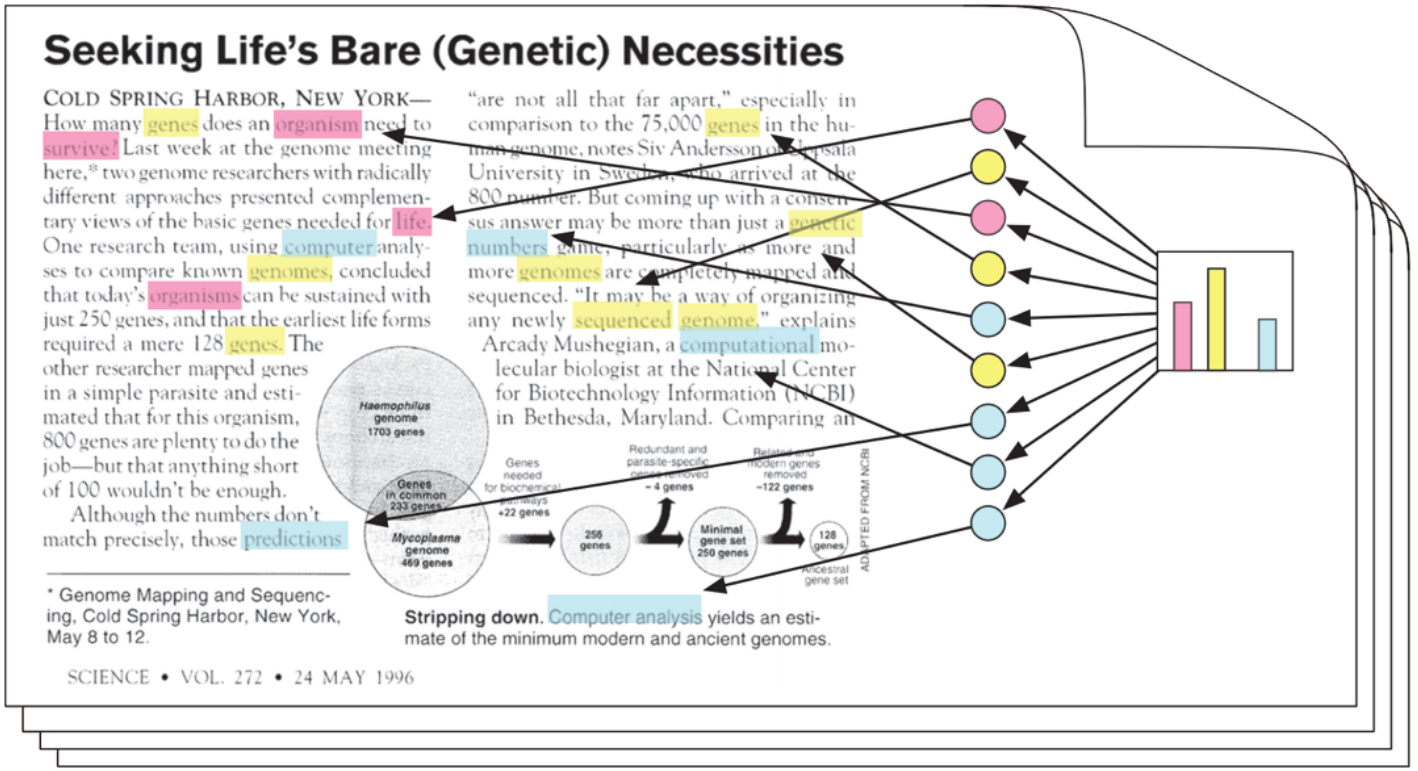
life	0.02
evolve	0.01
organism	0.01
. . .	

brain	0.04
neuron	0.02
nerve	0.01
. . .	

data	0.02
number	0.02
computer	0.01
. . .	

Documents

Topic proportions and assignments



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

* 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.

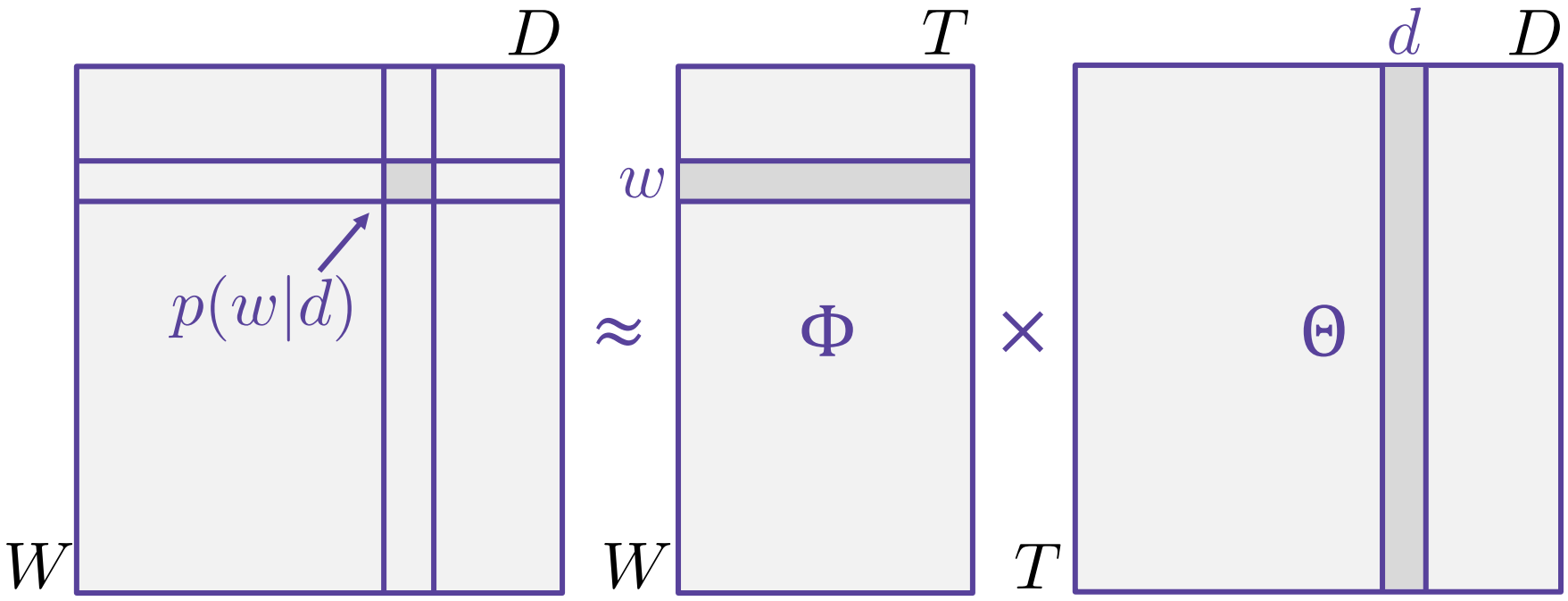
SCIENCE • VOL. 272 • 24 MAY 1996

David Blei, Probabilistic topic models, 2012.

Matrix way of thinking

Probabilistic Latent Semantic Analysis:

$$p(w|d) = \sum_{t \in T} p(w|t)\, p(t|d) = \sum_{t \in T} \phi_{wt} \theta_{td}$$



How to train PLSA?

How would you train the model?

Probabilistic Latent Semantic Analysis:

$$p(w|d) = \sum_{t \in T} p(w|t)\, p(t|d) = \sum_{t \in T} \phi_{wt} \theta_{td}$$

Parameters of the model:

- ϕ_{wt} – *probability of word* ***w*** *in topic* ***t***
- θ_{td} – *probability of topic* ***t*** *in document* ***d***

How would you train the model?

Log-likelihood optimization:

$$\log \prod_{d \in D} p(d) \prod_{w \in d} p(w|d)^{n_{dw}} \to \max_{\Phi, \Theta}$$

$$\Updownarrow$$

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \log \sum_{t \in T} \phi_{wt} \theta_{td} \to \max_{\Phi, \Theta}$$

Given non-negativity and normalization constraints:

$$\phi_{wt} \geq 0 \qquad \sum_{w \in W} \phi_{wt} = 1 \qquad \sum_{t \in T} \theta_{td} = 1$$

$$\theta_{td} \geq 0$$

How would you train the model?

Log-likelihood optimization:

$$\log \prod_{d \in D} p(d) \prod_{w \in d} p(w|d)^{n_{dw}} \to \max_{\Phi, \Theta}$$

$$\Updownarrow$$

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \log \sum_{t \in T} \phi_{wt} \theta_{td} \to \max_{\Phi, \Theta}$$

Given non-negativity and normalization constraints:

$$\phi_{wt} \geq 0 \qquad \sum_{w \in W} \phi_{wt} = 1 \qquad \sum_{t \in T} \theta_{td} = 1$$

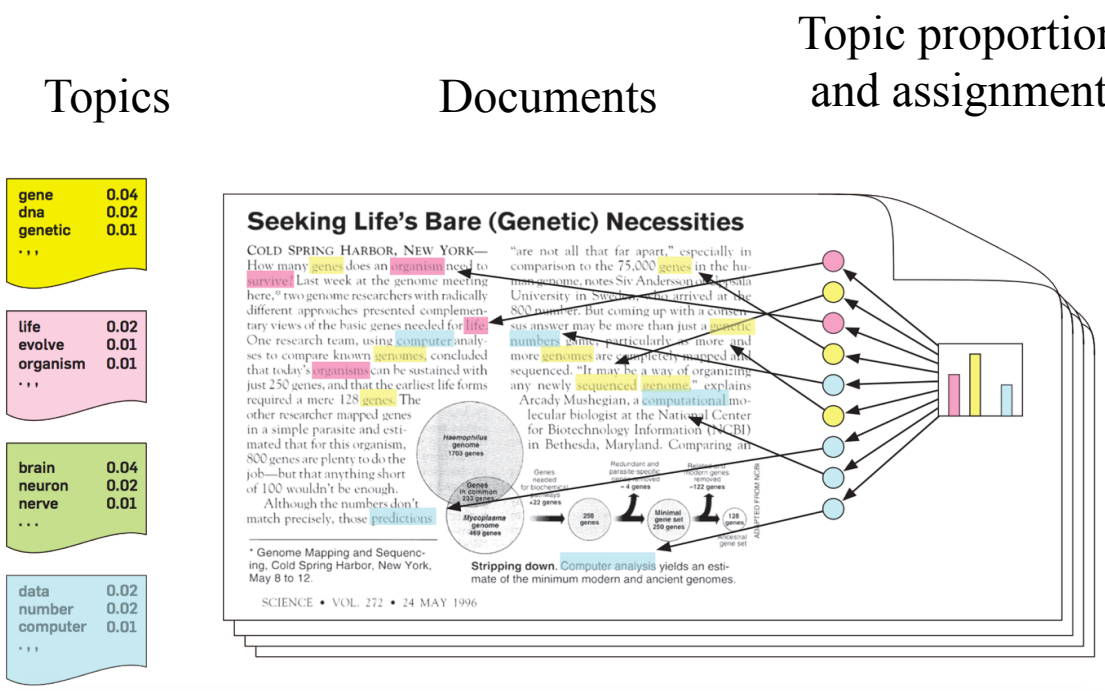
$$\theta_{td} \geq 0$$

We have just plain texts

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…

We have just plain texts

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…



If we knew topic assignments…

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…

If we knew topic assignments…

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…

We would just count:

$$p(w = sky|t) = \frac{n_{wt}}{\sum_w n_{wt}} = \frac{1}{4}$$

If we knew topic assignments…

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…

We would just count:

$$p(w = sky|t) = \frac{n_{wt}}{\sum_w n_{wt}} = \frac{1}{4}$$

$$p(t = t|d) = \frac{n_{td}}{\sum_t n_{td}} = \frac{4}{54}$$

But we have just plain texts

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…

But we have just plain texts

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened up, and said that, if it were raining already, the Heffalump would be looking at the sky wondering if it would clear up, and so he wouldn't see the Very Deep Pit until he was half-way down…

Idea! Let's estimate the topic assignment probabilities!

$$p(t|d,w) = \frac{p(w,t|d)}{p(w|d)} = \frac{p(w|t)p(t|d)}{p(w|d)}$$

Bayes rule *Product rule*

Put everything together: EM-algorithm

E-step:

$$p(t|d,w) = \frac{p(w|t)p(t|d)}{p(w|d)} = \frac{\phi_{wt}\theta_{td}}{\sum_{s\in T}\phi_{ws}\theta_{sd}}$$

M-step:

$$\phi_{wt} = \frac{n_{wt}}{\sum_w n_{wt}} \qquad n_{wt} = \sum_d n_{dw}\, p(t|d,w)$$

$$\theta_{td} = \frac{n_{td}}{\sum_t n_{td}} \qquad n_{td} = \sum_w n_{dw}\, p(t|d,w)$$

The zoo of topic models

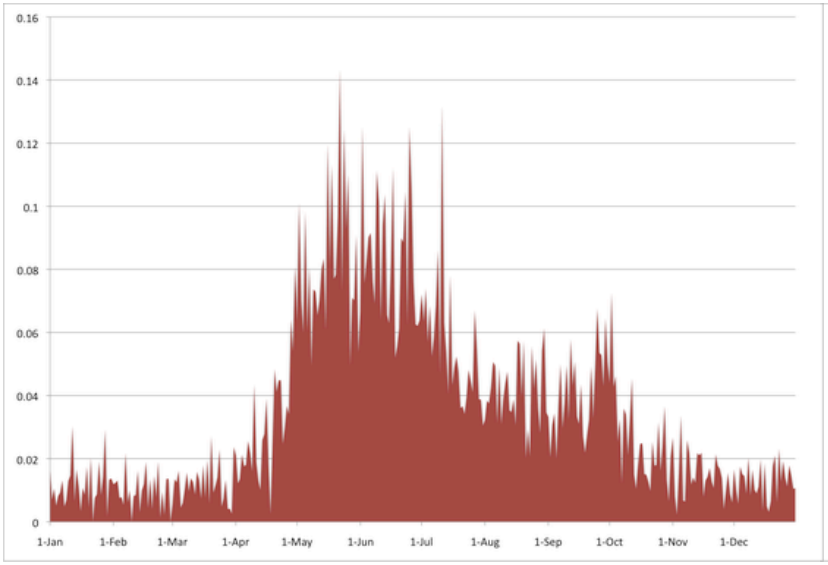
Martha Ballard's diary

- Diary had daily entries over the course of 27 years
- Topic modeling helps to analyze it
- Revealed topics (the most probable words):

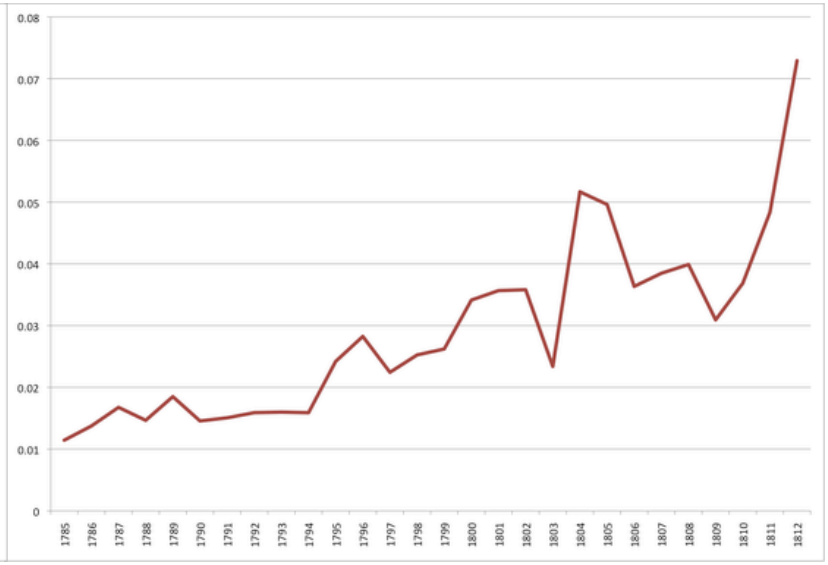
- ***GARDENING:*** *garden worked clear beans corn warm planted matters cucumbers potatoes plants*
- ***CHURCH:*** *meeting attended afternoon reverend worship foren mr famely st lecture discoarst administered*
- ***DEATH:*** *day yesterday informed morn years death ye hear expired expired weak dead*
- ***SHOPPING:*** *butter sugar carried candles wheat store flower*

Martha Ballard's diary

- Diary had daily entries over the course of 27 years
- Topic modeling helps to analyze it
- How topics are developing through time:



Gardening (average year)



Emotions (1785-1812)

http://www.cameronblevins.org/posts/topic-modeling-martha-ballards-diary/

Latent Dirichlet Allocation

Dirichlet priors for $\phi_t = (\phi_{wt})_{w \in W}$ and $\theta_d = (\theta_{td})_{t \in T}$:

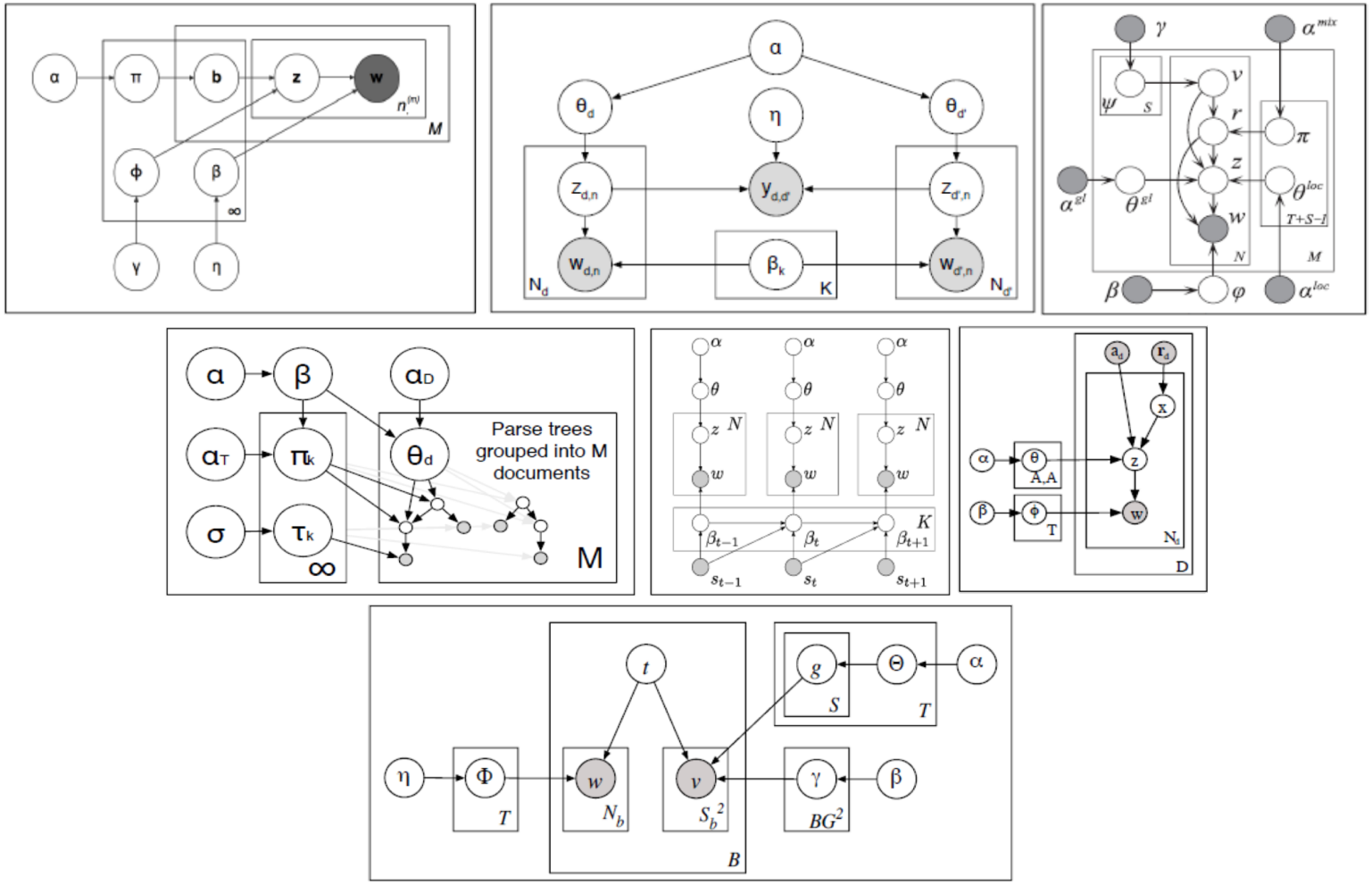
$$Dir(\phi_t|\beta) = \frac{\Gamma(\beta_0)}{\prod_w \Gamma(\beta_w)} \prod_w \phi_{wt}^{\beta_w - 1} \quad \beta_0 = \sum_w \beta_w, \beta_t > 0$$

- **Inference:**
  - Vibrational Bayes
  - Gibbs Sampling

- **Output:**
  - Posterior probabilities for parameters (also Dirichlet!).

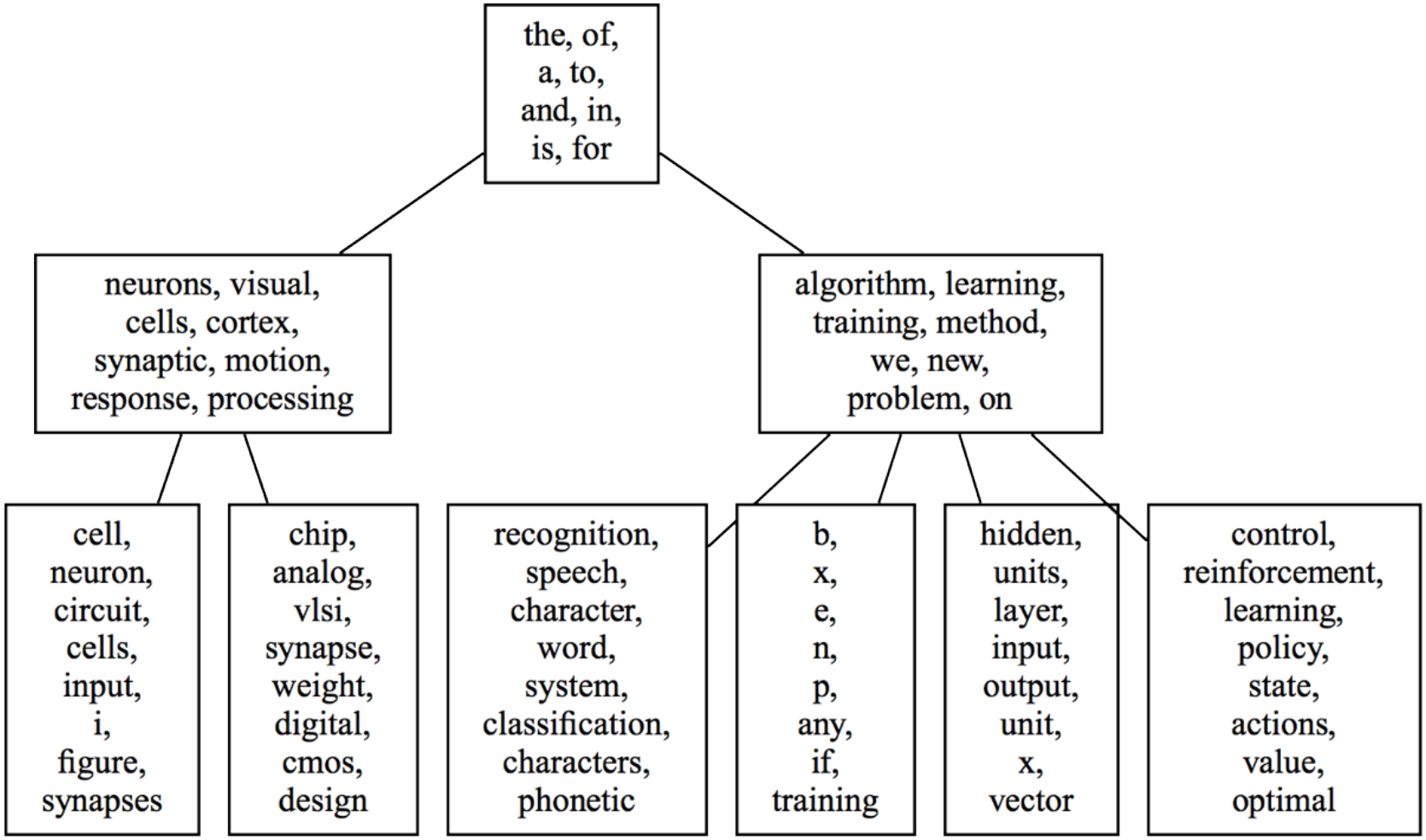
Asuncion A., Welling M., Smyth P., Teh Y. W. On smoothing and inference for topic models, 2009.

Bayesian methods and graphical models



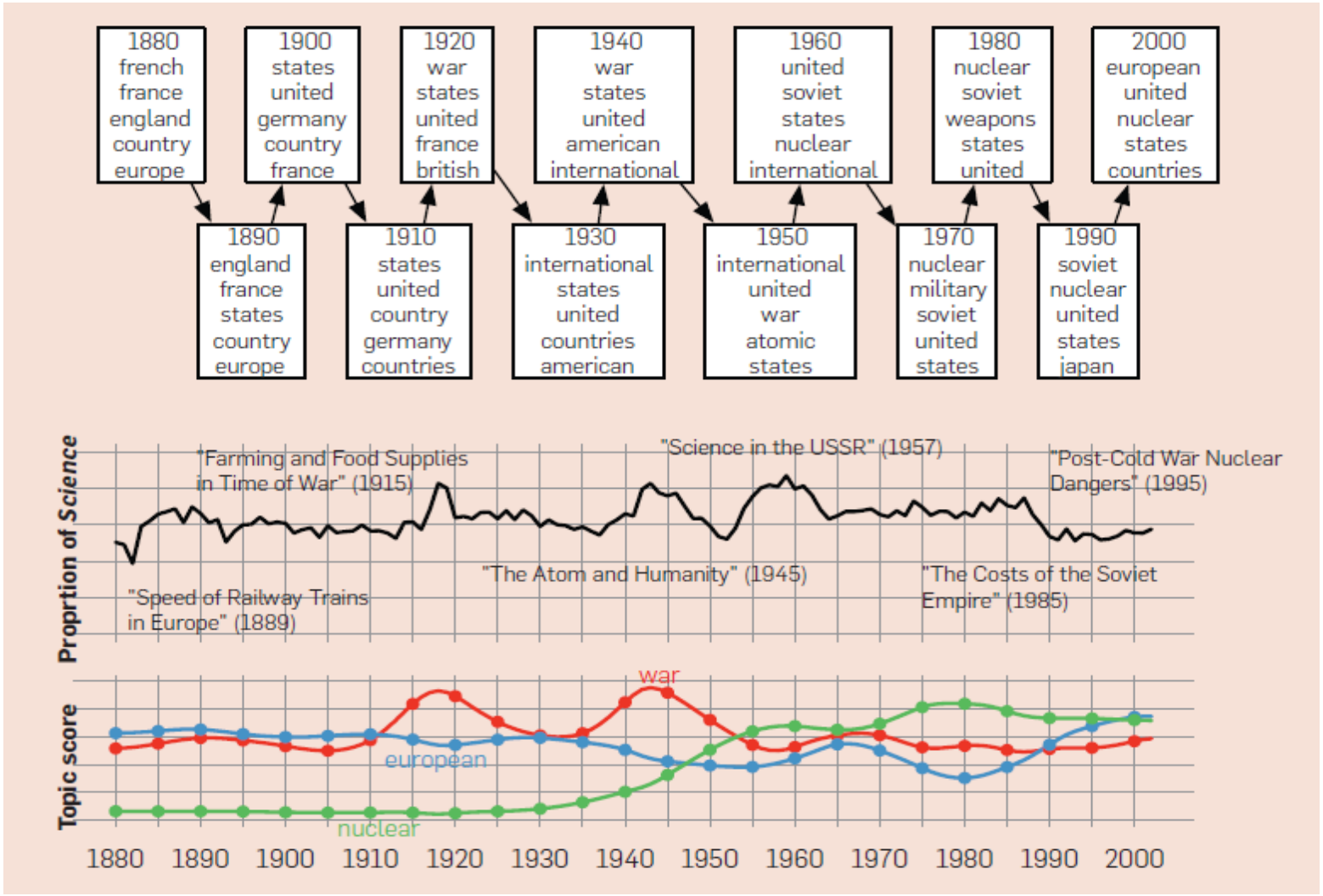
Ali Daud, Juanzi Li, Lizhu Zhou, Faqir Muhammad.
Knowledge discovery through directed probabilistic topic models: a survey, 2010.

Hierarchical topic models



D. Blei et. al. Hierarchical Topic Models and the Nested Chinese Restaurant Process, NIPS-2003.

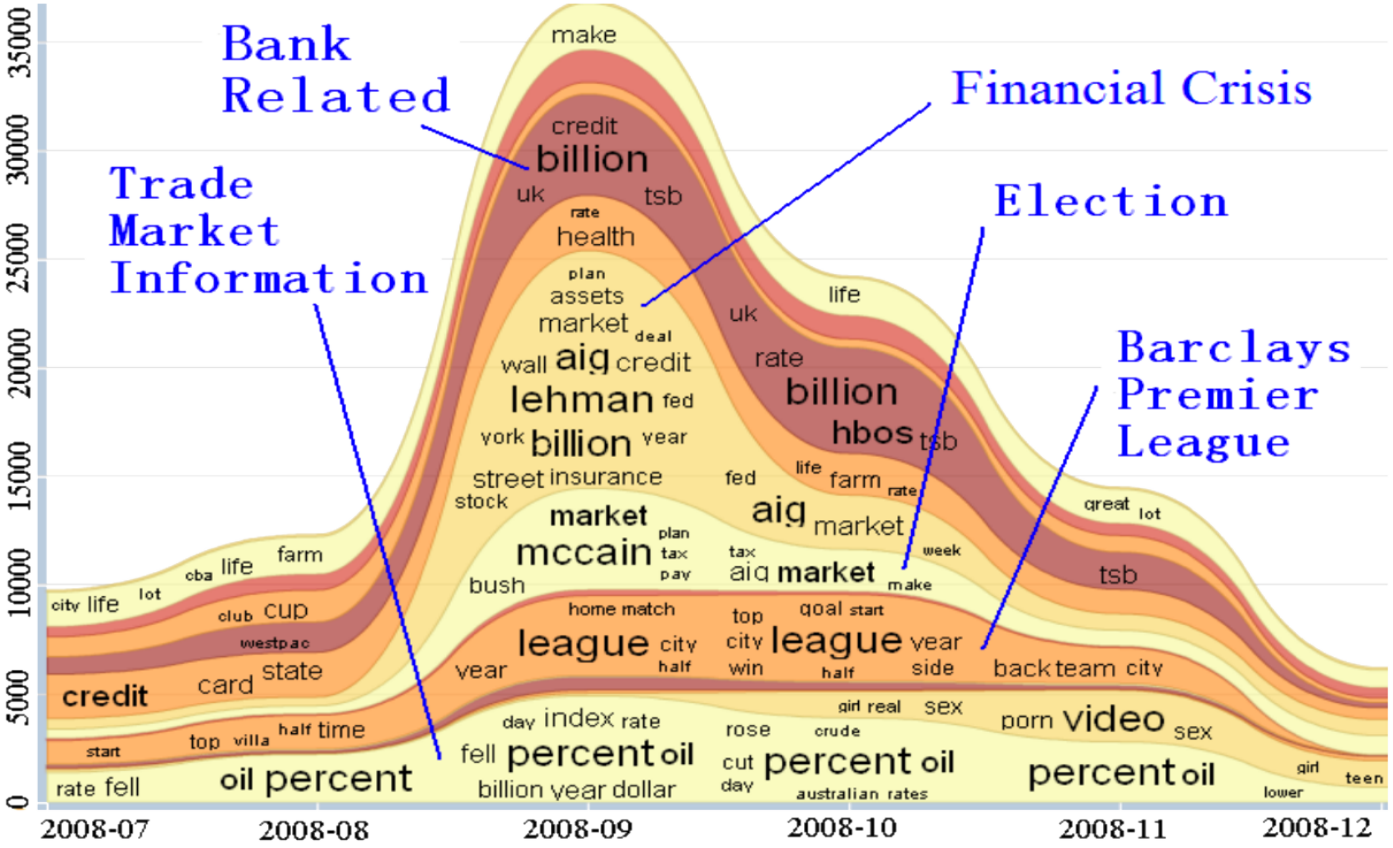
Dynamic topic models



David Blei, Probabilistic Topic Models, 2012.

Dynamic topic models

Topic detection and analysis of news flows:

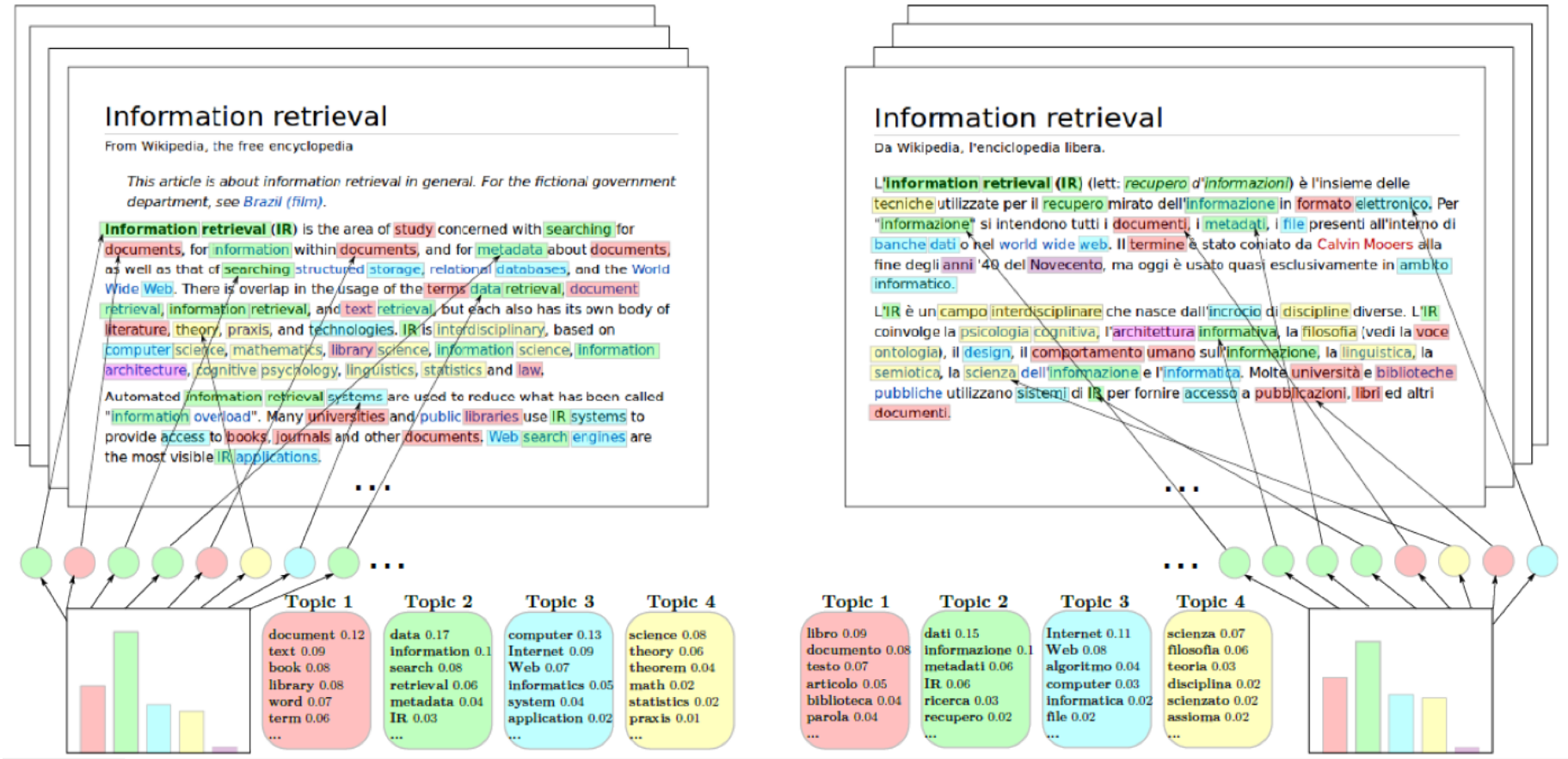


Jianwen Zhang, Yangqiu Song, Changshui Zhang, Shixia Liu Evolutionary Hierarchical Dirichlet Processes for Multiple Correlated Time-varying, KDD-2010.

Multilingual topic models

English corpus

Italian corpus



I. Vulic, W. De Smet, J. Tang, M.-F. Moens. Probabilistic topic modeling in multilingual settings: a short overview of its methodology with applications, NIPS-2012.

Additive Regularization for Topic Models

How to combine all those extensions in one model?

PLSA: $$\mathcal{L} = \sum_{d \in D} \sum_{w \in W} n_{dw} \log \sum_{t \in T} \phi_{wt} \theta_{td} \to \max_{\Phi, \Theta}$$

ARTM: $$\mathcal{L} + \sum_{i=1}^{n} \tau_i R_i(\Phi, \theta) \to \max_{\Phi, \Theta}$$

Example of a regularizer – diversity of topics:

$$R_i(\Phi) = -\sum_{t \neq s} \sum_{w} \phi_{wt} \phi_{ws}$$

K. Vorontsov, A. Potapenko Additive Regularization of Topic Models, 2015.

Regularized EM-algorithm

E-step:

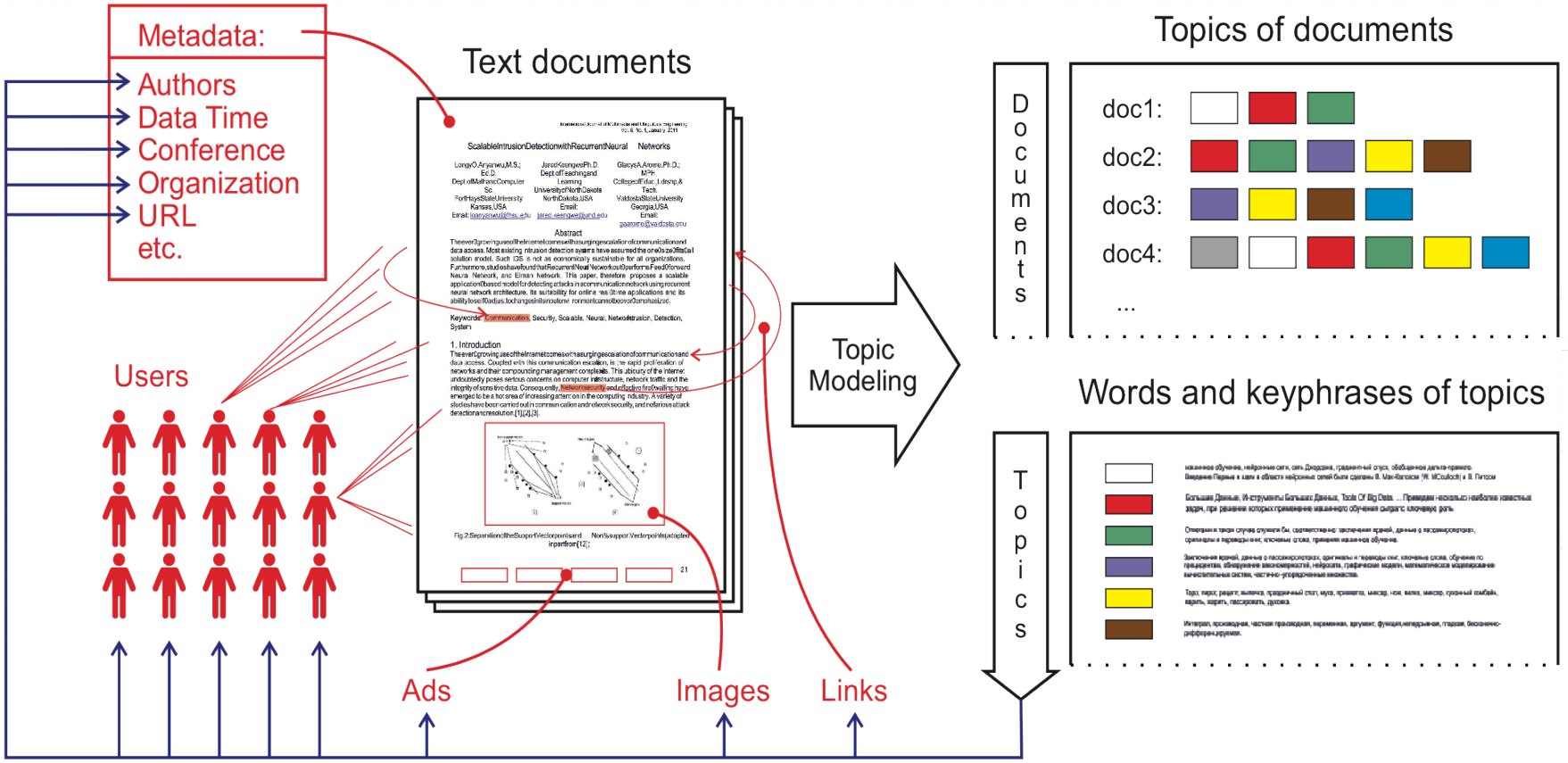
$$p(t|d,w) = \frac{p(w|t)p(t|d)}{p(w|d)} = \frac{\phi_{wt}\theta_{td}}{\sum_{s\in T}\phi_{ws}\theta_{sd}}$$

M-step:

$$\phi_{wt} = \underset{w\in W}{\text{norm}}\left(\sum_{d\in D} n_{dw}\, p(t|d,w) + \phi_{wt}\frac{\partial R}{\partial \phi_{wt}}\right)$$

$$\theta_{td} = \underset{t\in T}{\text{norm}}\left(\sum_{w\in d} n_{dw}\, p(t|d,w) + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right)$$

Multimodal topic models



K. Vorontsov et. al. BigARTM: Open Source Library for Regularized Multimodal Topic Modeling of Large Collections, 2015.

Multi-ARTM

How to incorporate tokens of additional modalities?

PLSA: $$\mathcal{L} = \sum_{d \in D} \sum_{w \in W} n_{dw} \log \sum_{t \in T} \phi_{wt} \theta_{td} \to \max_{\Phi, \Theta}$$

Multi-ARTM:

$$\sum_{m \in M} \lambda_m \sum_{d \in D} \sum_{w \in W^m} n_{dw} \log \sum_{t \in T} \phi_{wt} \theta_{td} \to \max_{\Phi, \Theta}$$

- Each topic is characterized by several probability distributions
- More parameters, still trained with EM-algorithm

Inter-modality similarities

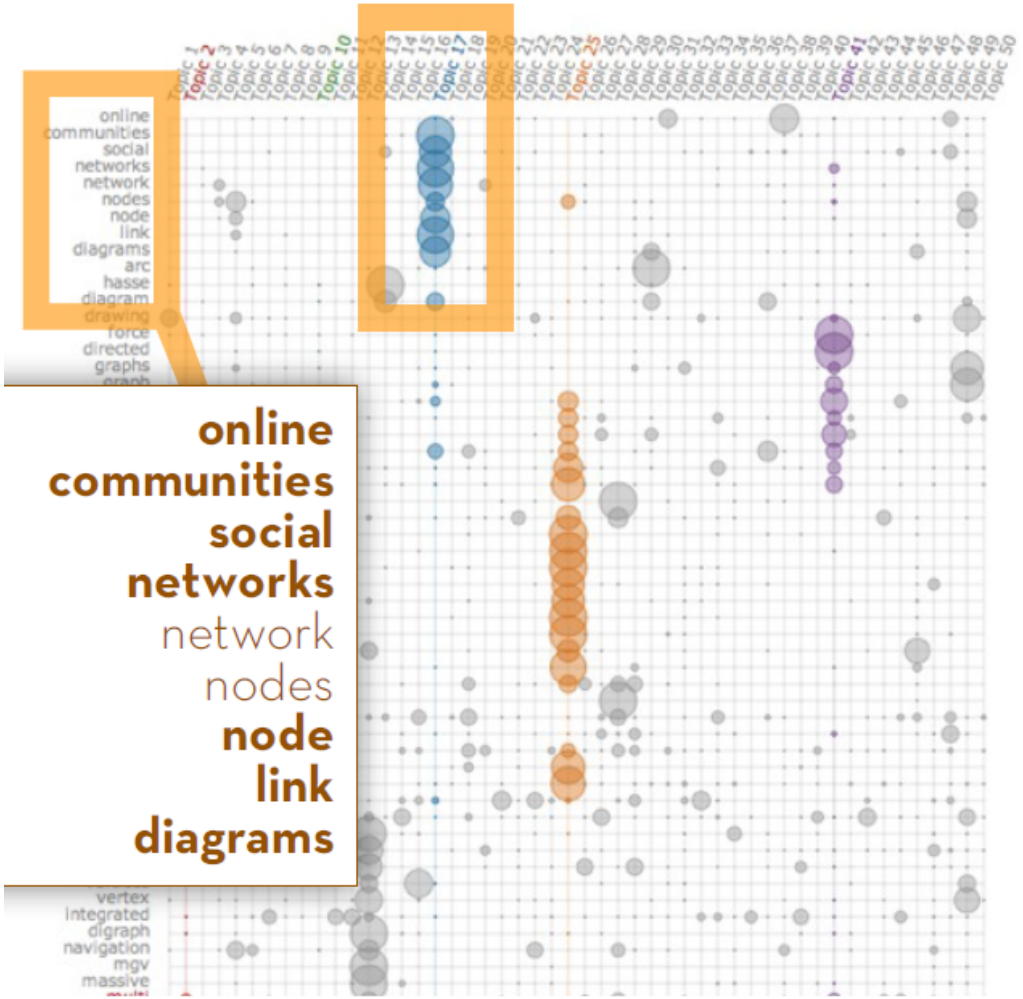
2015-12-18 Star Wars Release	2016-02-29 The Oscars	2015-05-09 Victory Day
jedi	statuette	great
sith	award	anniversary
fett	nomination	normandy
anakin	linklater	parade
chewbacca	oscar	demonstration
film series	birdman	vladimir
hamill	win	celebration
prequel	criticism	concentration
awaken	director	auschwitz
boyega	lubezki	photograph

Potapenko, Popov, Vorontsov: Interpretable probabilistic embeddings: bridging the gap between topic models and neural networks, 2017.

Libraries for topic modeling

- **BigARTM** is an open-source library for Additive Regularization of Topic Models, bigartm.org
- **Gensim** is a library of text analysis for Python, radimrehurek.com/gensim
- **MALLET** is a library of text analysis for Java mallet.cs.umass.edu
- **Vowpal Wabbit** has a fast implementation of online LDA hunch.net/~vw/

A few words about visualization



J. Chuang, C. D. Manning, J. Heer – Termite: Visualization Techniques For Assessing Textual Topic Models, 2012

380 ways to visualize: textvis.lnu.se