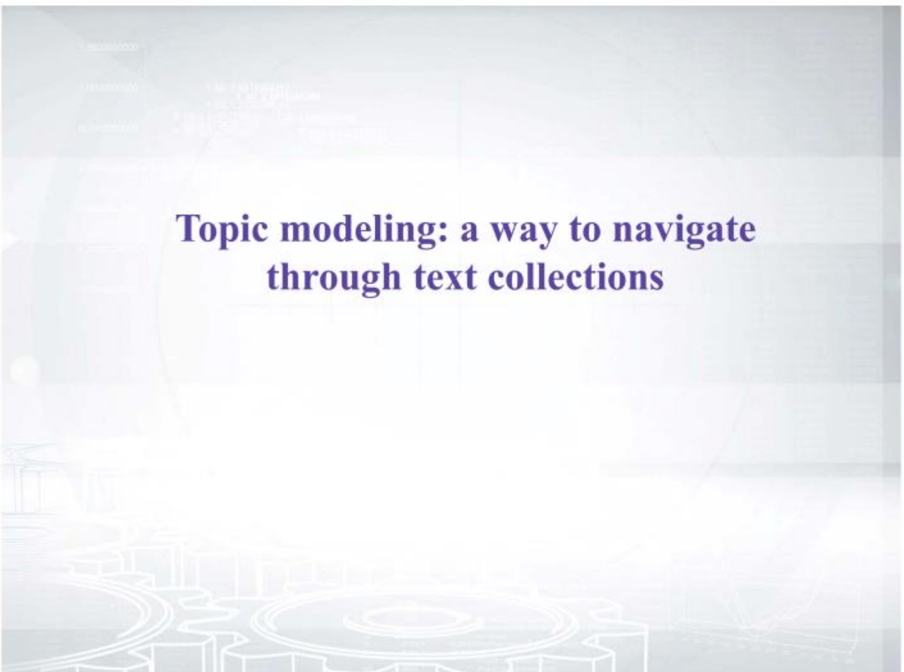
Topic Models

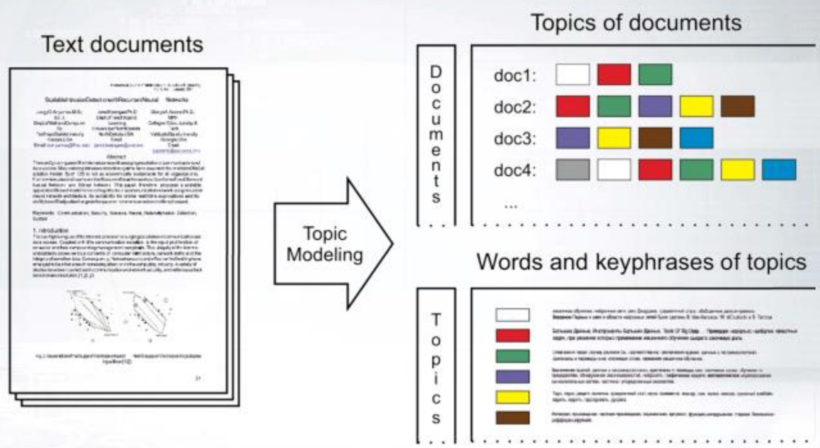
Saturday, June 16, 2018 1:01 PM

topicmodel1



soft ← some probability distribution
b-clustering

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

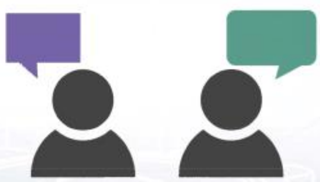
Where do we need that?

Exploration and navigation through large text collections

Where do we need that?

- Social network analysis

- Dialogue manager in chat-bots



text generation dependent on certain topic.

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
- Image captioning
- Bioinformatics (genome annotation)
- Exploratory search
- …

from some doc to other doc.

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*

text generated w prob.

toss ~~of~~ coin

=> get topic

toss a coin

=> get word

think of it as factorization

prob. distribution over word

prob. distribution of topic

topicmodel2

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:

forget about this get rid of it.

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

$$\Updownarrow$$

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

use EM algorithm

Given non-negativity and normalization constraints:

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

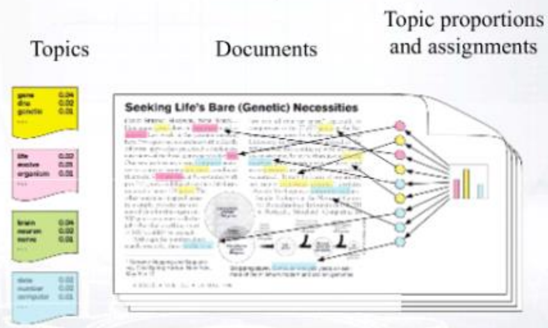
$$\theta_{td} \geq 0$$

We have just plain texts

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened again, 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 again, 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 again, 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 again, 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 again, 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:

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

← # of words in the topic

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

← # of words

But we have just plain texts

Pooh rubbed his nose again, and said that he hadn't thought of that. And then he brightened again, 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 again, 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!

(b) assumption independent

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

Bayes rule *Product rule*

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

$$= \frac{\phi_{wt}\,\theta_{td}}{\sum_{s \in T} \phi_{ws}\,\theta_{sd}}$$

$$= \frac{0.8 \times 0.7}{0.1 \times 0.2 + 0.8 \times 0.7 + 0.2 \times 0.1}$$

$$= 0.93$$

Φ matrix (many rows omitted, every column sums up to 1):

	topic 1	topic 2	topic 3
raining	0.01	0.1	0.05
would	0.1	0.2	0.1
...	...	...	...

Θ column for the document (sums up to 1):

	document
topic 1	0.1
topic 2	0.5
topic 3	0.4

The next E-step will compute posterior topic probabilities $p(t|d,w)$ for all words in the document. The next M-step will aggregate them to compute counts n_{wt} and n_{td}. Then it will normalize them to produce probabilities (new matrices Φ and Θ).

Do computations for n_{wt} count for the word **would** and **topic 2.**

Hints:

- First, compute $p(t|\,d,w)$ for the word **would** and **topic 2** .
- Recall that $n_{wt} = \sum_d n_{dw} p(t|\,d,w)$, where n_{dw} is the number of the word occurrences in the document.
- Assume that there is only one document in our toy corpus.

Enter the result with 2 digits after the decimal point.

2.000

Correct Response

Exactly! Let's review the formulas together.

E-step:

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

In our case we have $p(t|\,d,w) = \frac{0.2 \cdot 0.5}{0.1 \cdot 0.1 + 0.2 \cdot 0.5 + 0.1 \cdot 0.4}$ = 0,666666.....

M-step:

Assuming there is only one document, $n_{wt} = n_{dw} p(t|\,d,w)$, where n_{dw} is the number of the word **would** occurrences in the document, which is 3.

Thus, $n_{wt} = 3 \cdot 0.(6) = 2.0.$

$n_{dw} = 3$

p(topic 2 | d, "would")

$$= \frac{0.2 \times 0.5}{(0.1 \times 0.1 + 0.2 \times 0.5 + 0.1 \times 0.4}$$

Put everything together: EM-algorithm

E-step:

$$p(t|d,w) = \frac{p(w|t)p(t|d)}{p(w|d)} = \frac{\phi_{wt}\theta_{td}}{\sum \ldots}$$

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

所有topic

M-step:

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

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

$$= \frac{0.2 \times 0.5}{\sum (0.1\times 0.1 + 0.2\times 0.5 + 0.1\times 0.4)}$$

$$= \frac{0.1}{0.01 + 0.1 + 0.04}$$

$$\approx 0.6667$$

$$n_{wt} = \sum_{d\in D} n_{dw}\, p(t|d,w)$$

$$\approx 3 \times 0.6667$$

$$\approx 2.0001$$

topicmodel3

The zoo of topic models

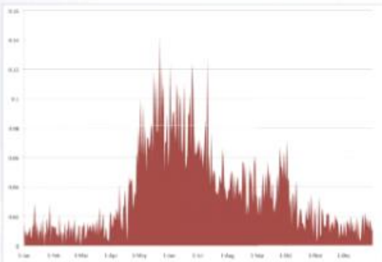
Martha Ballard's diary

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

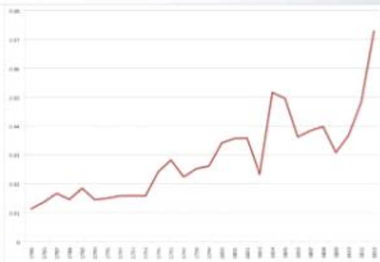
- ***GARDENING:*** *garden worked clear beans corn warm planted matters cucumbers potatoes plants*
- ***CHURCH:*** *meeting attended afternoon reverend worship foren mr famely st lecture discoarst administered*
- ***DEATH:*** *day yesterday informed morn years death ye hear expired expired weak dead*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

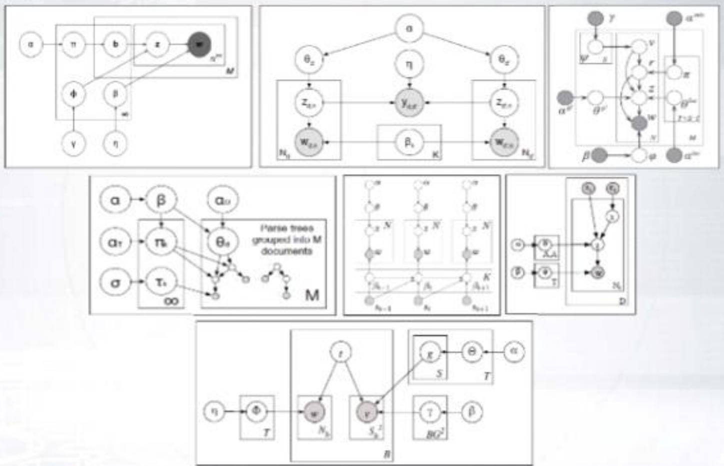
Dirichlet priors for $\phi_t = (\phi_{wt})_{w \in W}$ **and** $\theta_d = (\theta_{td})_{t \in T}$ **:**

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

- **Inference:**
  - Variational 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

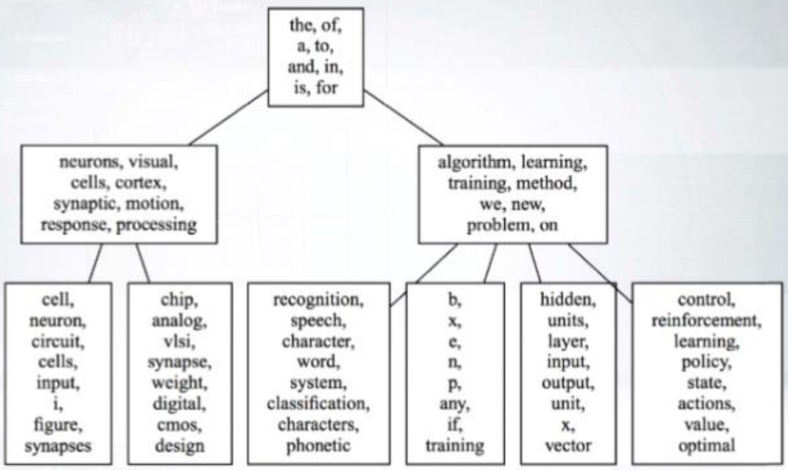


Ali Daud, Juanzi Li, Lizhu Zhou, Faqir Muhammad.
Knowledge discovery through directed probabilistic topic models: a survey, 2010.

How to develop new topic models.
(in general)

lots of extensions

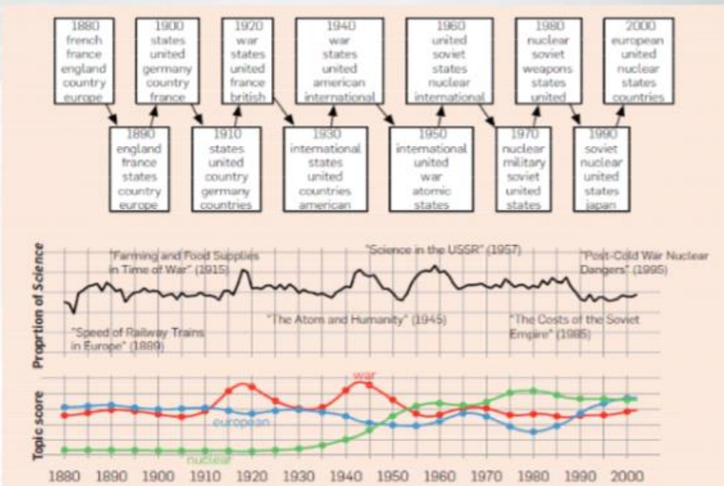
Hierarchical topic models



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

want topic to have ~~to~~ hierarchy.

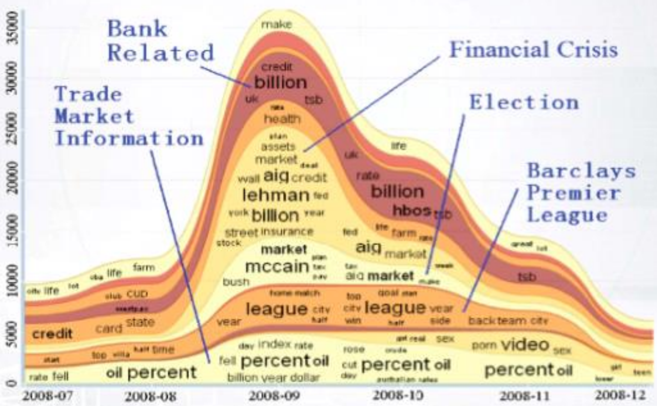
Dynamic topic models



David Blei, Probabilistic Topic Models, 2012.

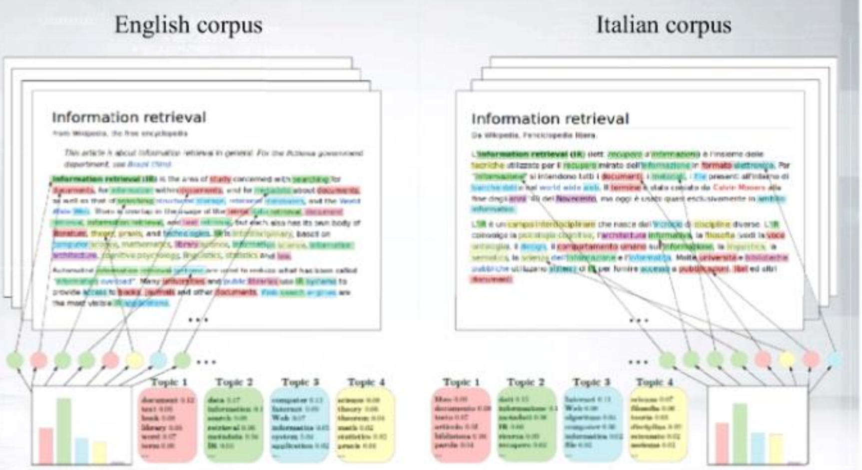
Dynamic topic models

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



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.

Is there a way to combine multiple topic models?

Additive Regularization for Topic Models

How to combine all those extensions in one model?

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

As different topics as possible

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:

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

Same as the PLSA model

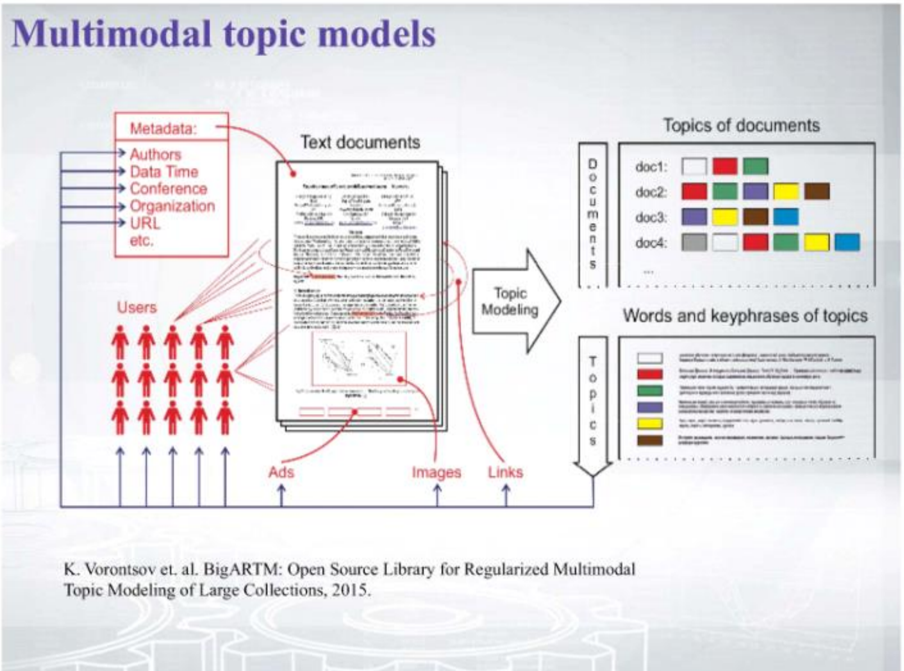
M-step:

Derivative of regularizer.

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

t.

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



need to model not only text, but also meta data

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

topic ~ distribution over authors

represent every entity in this hidden space of topic

time stamp.
words

topic

Inter-modality similarities

words popular in spa

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

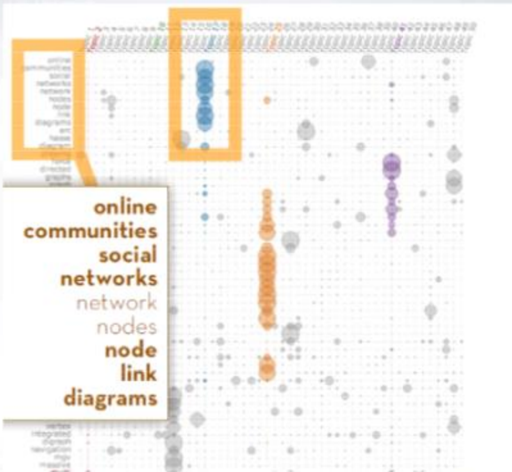
Potapenko, Popov, Vorontsov: Interpretable probabilistic embeddings: bridging the gap between topic models and neural networks, 2017.

A way to embed all your modality into a space

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/ → very fast

A few words about visualization



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