Regularized Matrix Factorization for Topic Modeling of Text Collections

Konstantin Vorontsov
Oleksandr Frei • Murat Apishev
Andrey Shadrikov • Alexander Plavin

(MIPT, MSU, CC RAS, Yandex • Moscow, Russia)

4th International Conference on Matrix Methods in Mathematics and Applications MMMA-2015

Moscow • 24–28 August 2015

Contents

Sparse stochastic matrix factorization under KL-loss

Given a matrix $Z = \|z_{ij}\|_{n\times m}$, $(i,j) \in \Omega \subseteq \{1..n\} \times \{1..m\}$

Find matrices $X = \|x_{it}\|_{n\times k}$ and $Y = \|y_{tj}\|_{k\times m}$ such that

$$\|Z - XY\|_{\Omega,d} = \sum_{(i,j)\in\Omega} d\Big(z_{ij}, \sum_t x_{it} y_{tj}\Big) \to \min_{X,Y}$$

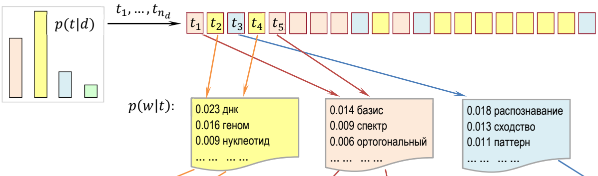
Variants of the problem:

- quadratic loss: $d(z,\hat{z}) = (z-\hat{z})^2$
- Kullback–Leibler loss: $d(z,\hat{z}) = z \ln(z/\hat{z}) - z + \hat{z}$
- nonnegative matrix factorization: $x_{it} \geqslant 0$, $y_{tj} \geqslant 0$
- stochastic matrix factorization: $x_{it} \geqslant 0$, $y_{tj} \geqslant 0$, $\sum_i x_{it} = 1$, $\sum_t y_{tj} = 1$
- sparse input data: $|\Omega| \ll nm$
- sparse output factorization X, Y

Probabilistic Topic Model (PTM) generating a text collection

Topic model explains terms w in documents d by topics t:

$$p(w|d) = \sum_t p(w|t)p(t|d)$$



$w_1, \dots, w_{n_d}$:

Разработан спектрально-аналитический подход к выявлению размытых протяженных повторов в геномных последовательностях. Метод основан на разномасштабном оценивании сходства нуклеотидных последовательностей в пространстве коэффициентов разложения фрагментов кривых GC- и GA-содержания по классическим ортогональным базисам. Найдены условия оптимальной аппроксимации, обеспечивающие автоматическое распознавание повторов различных видов (прямых и инвертированных, а также тандемных) на спектральной матрице сходства. Метод одинаково хорошо работает на разных масштабах данных. Он позволяет выявлять следы сегментных дупликаций и мегасателлитные участки в геноме, районы синтении при сравнении пары геномов. Его можно использовать для детального изучения фрагментов хромосом (поиска размытых участков с умеренной длиной повторяющегося паттерна).

Inverse problem: text collection → PTM

Given: D is a set (collection) of documents
W is a set (vocabulary) of terms
n_{dw} = how many times term w appears in document d

Find: parameters $\phi_{wt} = p(w|t)$, $\theta_{td} = p(t|d)$ of the topic model

$$p(w|d) = \sum_t \phi_{wt}\theta_{td}.$$

The problem of log-likelihood maximization under constraints:

$$\mathscr{L}(\Phi, \Theta) = \sum_{d,w} n_{dw} \ln \sum_t \phi_{wt}\theta_{td} \to \max_{\Phi,\Theta},$$

$$\phi_{wt} \geqslant 0, \quad \sum_{w\in W} \phi_{wt} = 1; \qquad \theta_{td} \geqslant 0, \quad \sum_{t\in T} \theta_{td} = 1.$$

Hofmann T. Probabilistic Latent Semantic Indexing. ACM SIGIR, 1999.

EM-algorithm for likelihood maximization [Hofmann, 1999]

From KKT conditions for the constrained maximization problem

Theorem

Maximum of $\mathscr{L}(\Phi, \Theta)$ satisfies the system of equations with model parameters ϕ_{wt}, θ_{td} and auxiliary variables p_{tdw}, n_{wt}, n_{td}:

$$
\begin{array}{rl}
\text{E-step:} & \left\{\begin{array}{l}
p_{tdw} = \dfrac{\phi_{wt}\theta_{td}}{\sum_{t'} \phi_{wt'}\theta_{t'd}}; \\
\text{M-step:} \quad \phi_{wt} = \dfrac{n_{wt}}{\sum_{w'} n_{w't}}; \quad n_{wt} = \sum\limits_{d \in D} n_{dw} p_{tdw}; \\
\theta_{td} = \dfrac{n_{td}}{\sum_{t'} n_{t'd}}; \quad n_{td} = \sum\limits_{w \in d} n_{dw} p_{tdw};
\end{array}\right.
\end{array}
$$

EM-algorithm alternates E-step and M-step until convergence.
EM-algorithm is equivalent to a simple iteration method.

LDA — Latent Dirichlet Allocation [Blei, 2003]

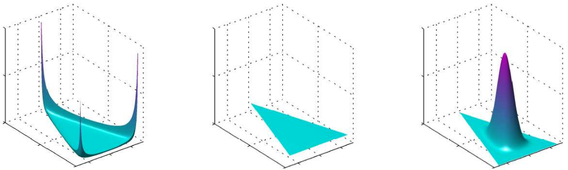
Assumption. Column vectors $\phi_t = (\phi_{wt})_{w \in W}$ and $\theta_d = (\theta_{td})_{t \in T}$ are generated from Dirichlet distributions, $\alpha \in \mathbb{R}^{|T|}$, $\beta \in \mathbb{R}^{|W|}$:

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

$$\mathrm{Dir}(\theta_d|\alpha) = \frac{\Gamma(\alpha_0)}{\prod\limits_t \Gamma(\alpha_t)} \prod_t \theta_{td}^{\alpha_t - 1}, \quad \alpha_0 = \sum_t \alpha_t, \; \alpha_t \geqslant 0;$$

Example:

$\mathrm{Dir}(\theta|\alpha)$
$|T| = 3$
$\theta, \alpha \in \mathbb{R}^3$



$\alpha_1 = \alpha_2 = \alpha_3 = 0.1$ $\quad$ $\alpha_1 = \alpha_2 = \alpha_3 = 1$ $\quad$ $\alpha_1 = \alpha_2 = \alpha_3 = 10$

The main difference between LDA and PLSA

The estimates of conditionals $\phi_{wt} \equiv p(w|t)$, $\theta_{td} \equiv p(t|d)$:

- in PLSA — unbiased maximum likelihood estimates:

$$\phi_{wt} = \frac{n_{wt}}{n_t}, \qquad \theta_{td} = \frac{n_{td}}{n_d}$$

- in LDA — smoothed Bayesian estimates:

$$\phi_{wt} = \frac{n_{wt} + \beta_w}{n_t + \beta_0}, \qquad \theta_{td} = \frac{n_{td} + \alpha_t}{n_d + \alpha_0}.$$

The difference is significant for small n_{wt}, n_{td} only.
Robust LDA and robust PLSA produce almost identical models.

Asuncion A., Welling M., Smyth P., Teh Y. W. On smoothing and inference for topic models. Int'l Conf. on Uncertainty in Artificial Intelligence, 2009.

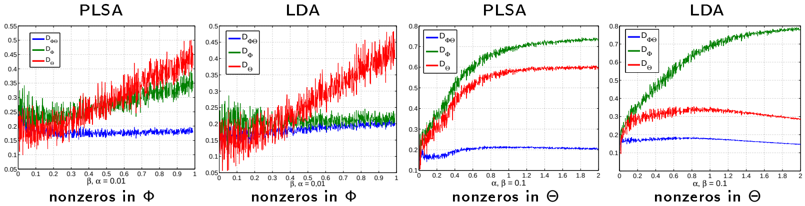
Potapenko A. A., Vorontsov K. V. Robust PLSA Performs Better Than LDA. ECIR-2013, Moscow, Russia, 24-27 March 2013. LNCS, Springer. Pp. 784–787.

Topic Modeling as an ill-posed inverse problem

The *nonuniqueness* and *instability* of matrix factorization:
$\Phi\Theta = (\Phi S)(S^{-1}\Theta) = \Phi'\Theta'$ for all S such that Φ', Θ' are stochastic.

Experiment: recovering known Φ, Θ on synthetic dataset,
$|D| = 500, \;\; |W| = 1000, \;\; |T| = 30.$

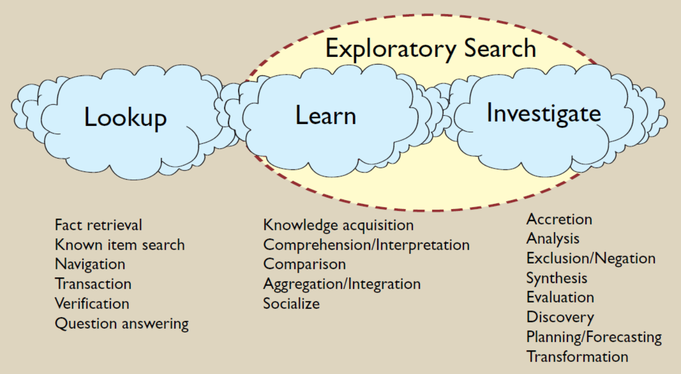
Result: product $\Phi\Theta$ is always recovered well, however
matrix Φ and matrix Θ are recovered if being highly sparse only:



Conclusions: Dirichlet prior is too weak as a regularizer;
stronger regularization is needed to ensure a stable solution.

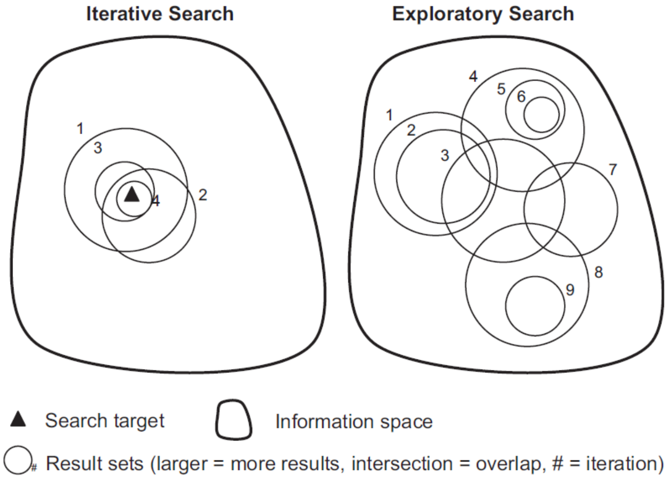
Exploratory Search for learning, knowledge acquisition and discovery

- what if the user doesn't know which keywords to use?
- what if the user isn't looking for a single answer?



Gary Marchionini. Exploratory Search: from finding to understanding. Communications of the ACM. 2006, 49(4), p. 41–46.

Iterative “query-browse-refine” search vs Exploratory Search



R.W.White, R.A.Roth. Exploratory Search: beyond the Query-Response paradigm. San Rafael, CA: Morgan and Claypool, 2009.

Exploratory search scenario

Search query:

- a document of any length or even a set of documents

Search intents:

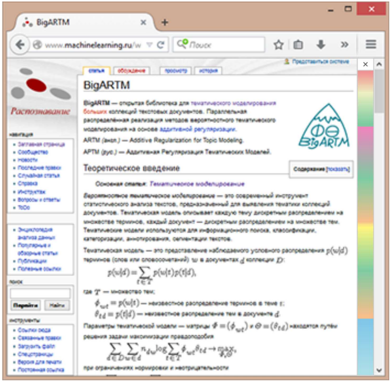
- what topics does it contain?
- what else is known on these topics?
- what is the structure of this domain area?
- what is most important, useful, popular, recent here?

Search scenario:

1. given a text (of any length) at hand (in any application)
2. identify topics and sub-topics it contains
3. show textual and graphical representations of these topics

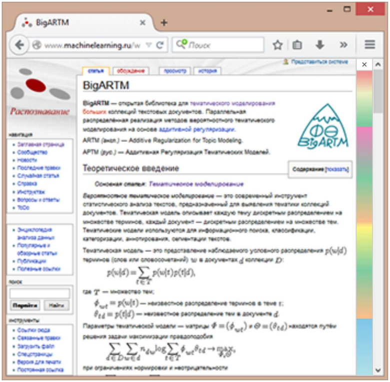
Exploratory search: the prototype of graphical user interface

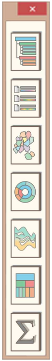
Color topic bar is a starting GUI element for exploratory search



Exploratory search: the prototype of graphical user interface

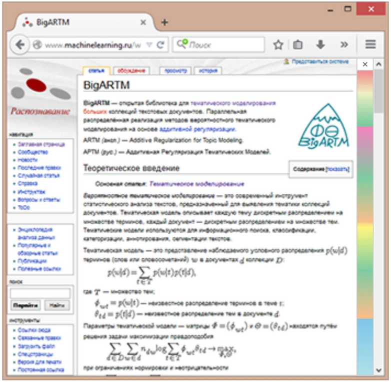
Click on the color topic bar is a topic query

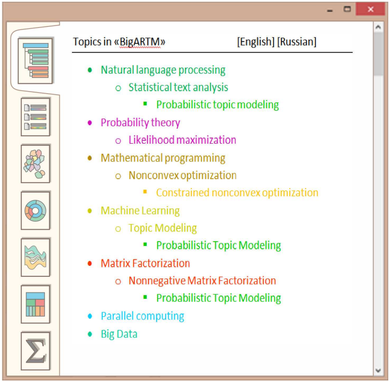




Exploratory search: the prototype of graphical user interface

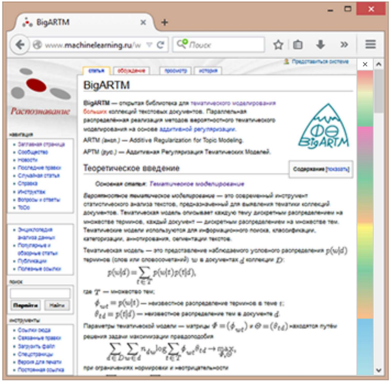
Topics of the query document

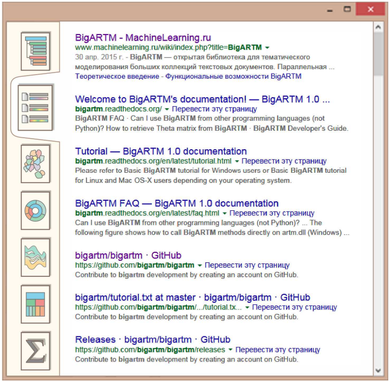




Exploratory search: the prototype of graphical user interface

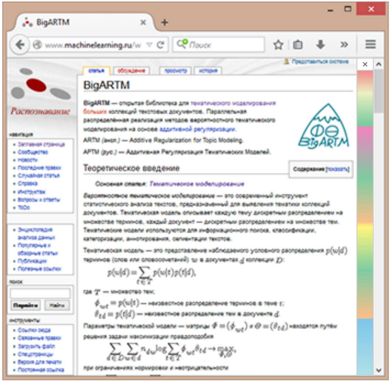
Similar documents and objects ranked by relevance

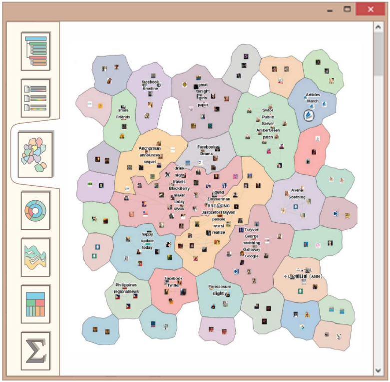




Exploratory search: the prototype of graphical user interface

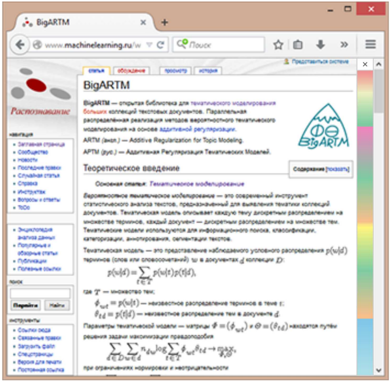
Topic roadmap: clustering of relevant documents

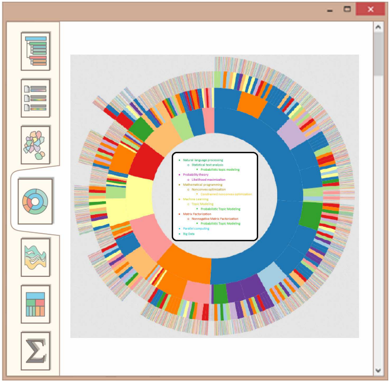




Exploratory search: the prototype of graphical user interface

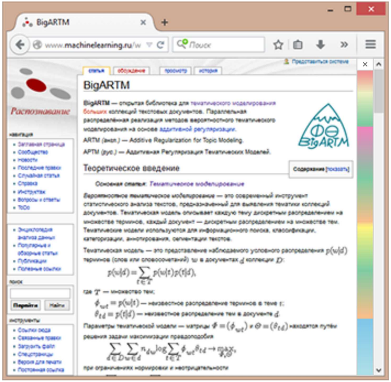
Topic hierarchy: topical structure of the domain area

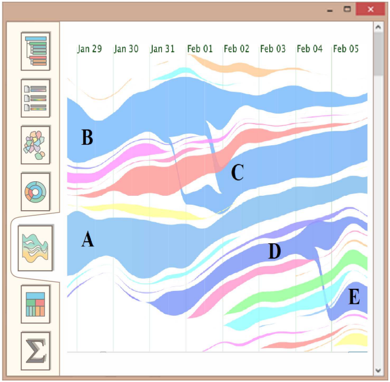




Exploratory search: the prototype of graphical user interface

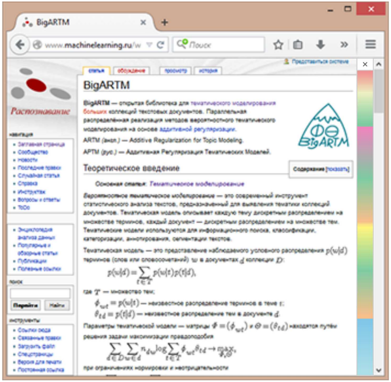
Topic river: evolution of the domain area

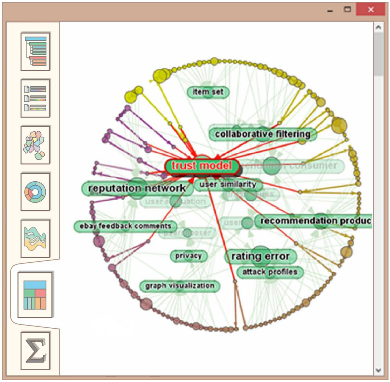




Exploratory search: the prototype of graphical user interface

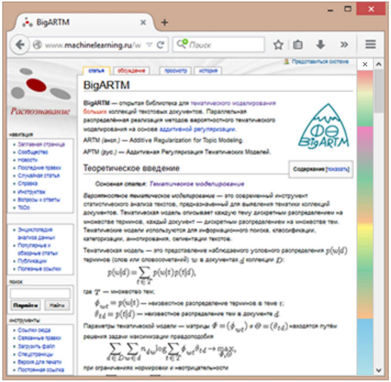
Topic bar: segmentation of the query document

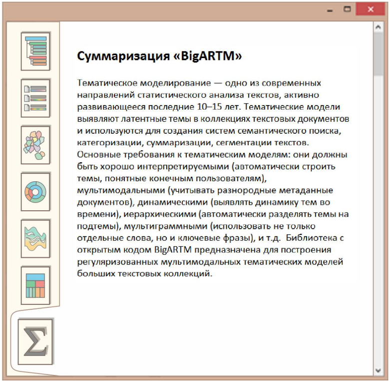




Exploratory search: the prototype of graphical user interface

Summarization of the query document





http://textvis.lnu.se

A visual survey of 170 text visualization techniques

The elements of Exploratory Search technology

1. Web crawling .... ready-made solutions
2. Content filtering .... ready-made solutions
3. Topic modeling .... **ongoing research**
4. Building the inverted index .... ready-made solutions
5. Ranking .... ready-made solutions
6. Visualization .... ready-made solutions

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically
3. **Multilingual**: cross-language and multi-language search should be supported

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically
3. **Multilingual**: cross-language and multi-language search should be supported
4. **Multimodal**: authors, categories, sources, links, tags, named entities, users, etc. should be involved in the model

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically
3. **Multilingual**: cross-language and multi-language search should be supported
4. **Multimodal**: authors, categories, sources, links, tags, named entities, users, etc. should be involved in the model
5. **Temporal**: topic dynamics over time should be identified

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically
3. **Multilingual**: cross-language and multi-language search should be supported
4. **Multimodal**: authors, categories, sources, links, tags, named entities, users, etc. should be involved in the model
5. **Temporal**: topic dynamics over time should be identified
6. **Hierarchical**: granularity of topics should be user-adjustable

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically
3. **Multilingual**: cross-language and multi-language search should be supported
4. **Multimodal**: authors, categories, sources, links, tags, named entities, users, etc. should be involved in the model
5. **Temporal**: topic dynamics over time should be identified
6. **Hierarchical**: granularity of topics should be user-adjustable
7. **Segmented**: the topical text segmentation should be supported beyond the bag-of-words (BoW) model

Exploratory Search requires that the Topic Model was...

1. **Interpretable**: each topic should be well interpretable by humans and labeled automatically
2. **Multigram**: keyphrases should be extracted automatically
3. **Multilingual**: cross-language and multi-language search should be supported
4. **Multimodal**: authors, categories, sources, links, tags, named entities, users, etc. should be involved in the model
5. **Temporal**: topic dynamics over time should be identified
6. **Hierarchical**: granularity of topics should be user-adjustable
7. **Segmented**: the topical text segmentation should be supported beyond the bag-of-words (BoW) model
8. **Semi-supervised**: the corrections from experts should be used to improve the model

Additive Regularization for Topic Modeling (ARTM)

Additional *regularization* criteria $R_i(\Phi, \Theta) \to \max$, $i = 1, \dots, n$.

The problem of regularized log-likelihood maximization under non-negativeness and normalization constraints:

$$\underbrace{\sum_{d,w} n_{dw} \ln \sum_{t \in T} \phi_{wt} \theta_{td}}_{\text{log-likelihood } \mathscr{L}(\Phi,\Theta)} + \underbrace{\sum_{i=1}^{n} \tau_i R_i(\Phi, \Theta)}_{R(\Phi,\Theta)} \to \max_{\Phi,\Theta},$$

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

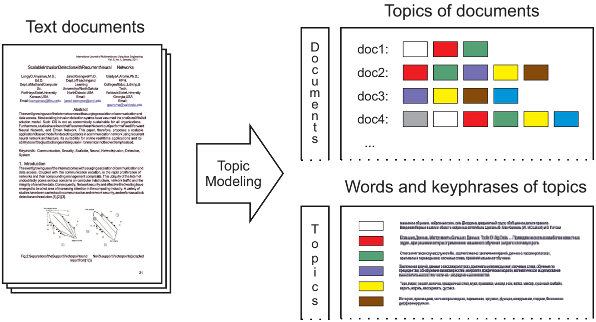
where $\tau_i > 0$ are *regularization coefficients*.

PLSA: $R(\Phi, \Theta) = 0$

LDA: $R(\Phi, \Theta) = \sum_{t,w} \beta_w \ln \phi_{wt} + \sum_{d,t} \alpha_t \ln \theta_{td}$

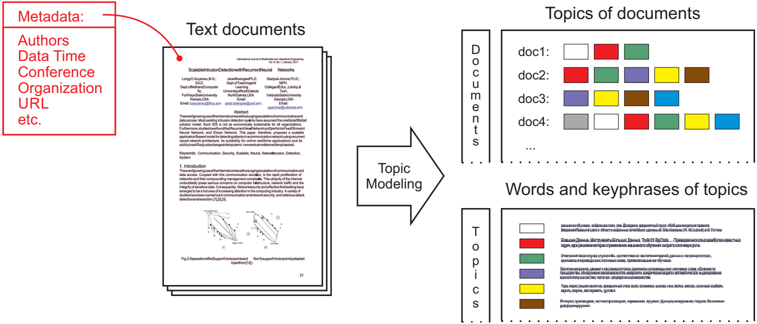
Multimodal Probabilistic Topic Modeling

Given a text document collection *Probabilistic Topic Model* finds:
$p(t|d)$ — topic distribution for each document d,
$p(w|t)$ — term distribution for each topic t.



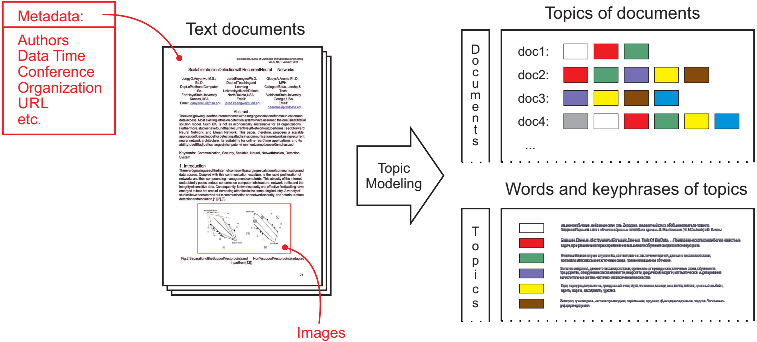
Multimodal Probabilistic Topic Modeling

Multimodal Topic Model finds topical distribution for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$,



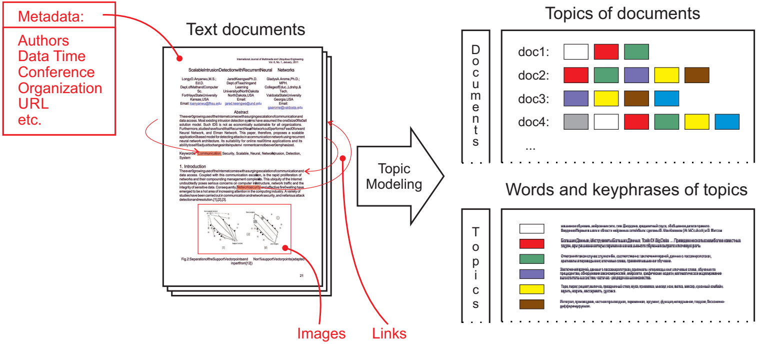
Multimodal Probabilistic Topic Modeling

Multimodal Topic Model finds topical distribution for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$, objects on images $p(o|t)$,



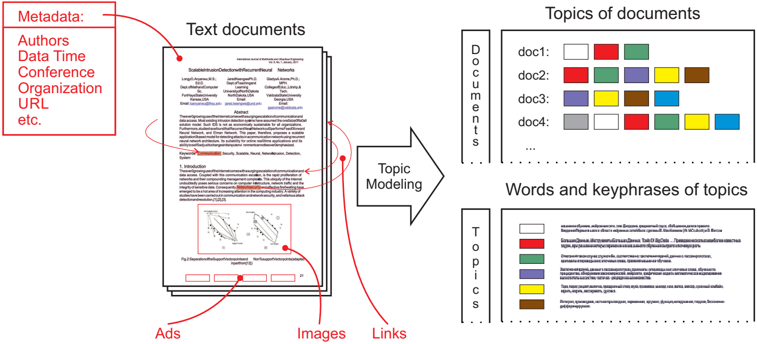
Multimodal Probabilistic Topic Modeling

Multimodal Topic Model finds topical distribution for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$, objects on images $p(o|t)$, linked documents $p(d'|t)$,



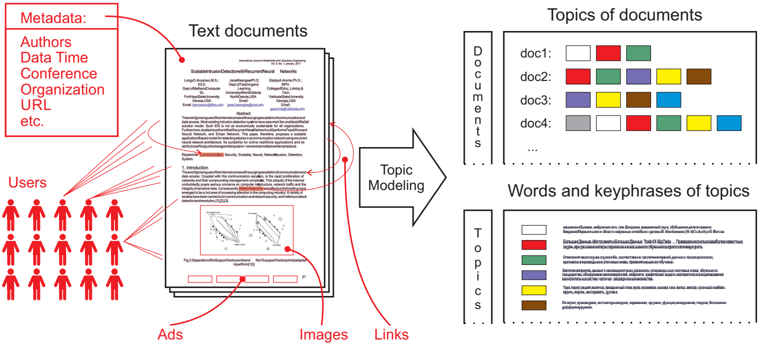
Multimodal Probabilistic Topic Modeling

Multimodal Topic Model finds topical distribution for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$, objects on images $p(o|t)$, linked documents $p(d'|t)$, advertising banners $p(b|t)$,



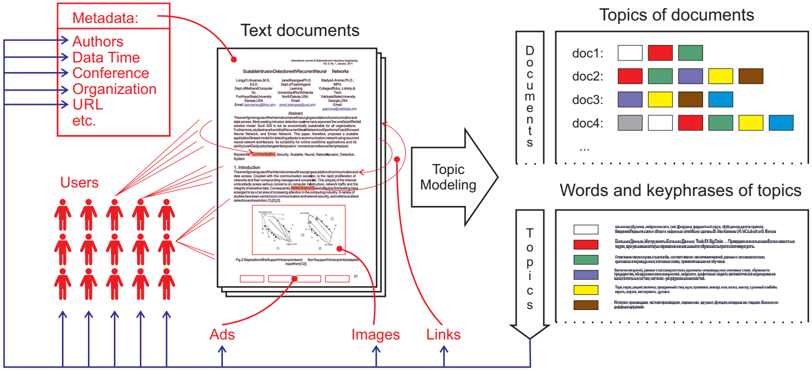
Multimodal Probabilistic Topic Modeling

Multimodal Topic Model finds topical distribution for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$, objects on images $p(o|t)$, linked documents $p(d'|t)$, advertising banners $p(b|t)$, users $p(u|t)$,



Multimodal Probabilistic Topic Modeling

Multimodal Topic Model finds topical distribution for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$, objects on images $p(o|t)$, linked documents $p(d'|t)$, advertising banners $p(b|t)$, users $p(u|t)$, and binds all these modalities into a single topic model.



Multimodal ARTM: combining multimodality and regularization

M is the set of modalities
W^m is a vocabulary of tokens of m-th modality, $m \in M$
$W = W^1 \sqcup \dots \sqcup W^M$ is a joint vocabulary of all modalities

The problem of multimodal regularized log-likelihood maximization under non-negativeness and normalization constraints:

$$\sum_{m \in M} \lambda_m \underbrace{\sum_{d \in D} \sum_{w \in W^m} n_{dw} \ln \sum_{t \in T} \phi_{wt} \theta_{td}}_{\text{modality log-likelihood } \mathscr{L}_m(\Phi,\Theta)} + \underbrace{\sum_{i=1}^{n} \tau_i R_i(\Phi, \Theta)}_{R(\Phi,\Theta)} \to \max_{\Phi,\Theta},$$

$$\phi_{wt} \geqslant 0, \quad \sum_{w \in W^m} \phi_{wt} = 1, \; m \in M; \qquad \theta_{td} \geqslant 0, \quad \sum_{t \in T} \theta_{td} = 1.$$

where $\lambda_m > 0$, $\tau_i > 0$ are *regularization coefficients*.

EM-algorithm for multimodal ARTM

EM-algorithm is a simple-iteration method for a system of equations

Theorem. The local maximum (Φ, Θ) satisfies the following system of equations with auxiliary variables $p_{tdw} = p(t|d, w)$:

$$p_{tdw} = \underset{t \in T}{\operatorname{norm}} \bigl(\phi_{wt} \theta_{td} \bigr);$$

$$\phi_{wt} = \underset{w \in W^m}{\operatorname{norm}} \left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right); \quad n_{wt} = \sum_{d \in D} \lambda_{m(w)} n_{dw} p_{tdw};$$

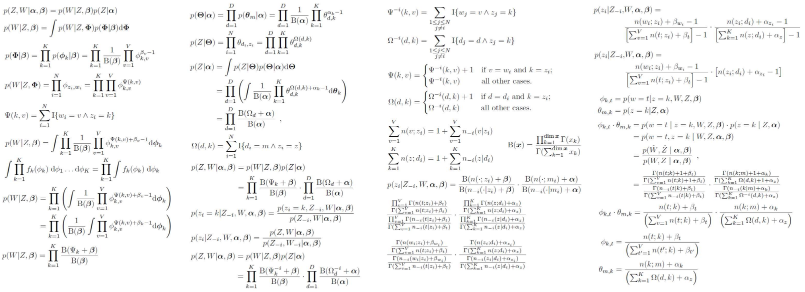
$$\theta_{td} = \underset{t \in T}{\operatorname{norm}} \left(n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \right); \quad n_{td} = \sum_{w \in d} \lambda_{m(w)} n_{dw} p_{tdw};$$

where $\underset{t \in T}{\operatorname{norm}}\, x_t = \frac{\max\{x_t, 0\}}{\sum\limits_{s \in T} \max\{x_s, 0\}}$ is nonnegative normalization;

$m(w)$ is the modality of the term w, so that $w \in W^{m(w)}$.

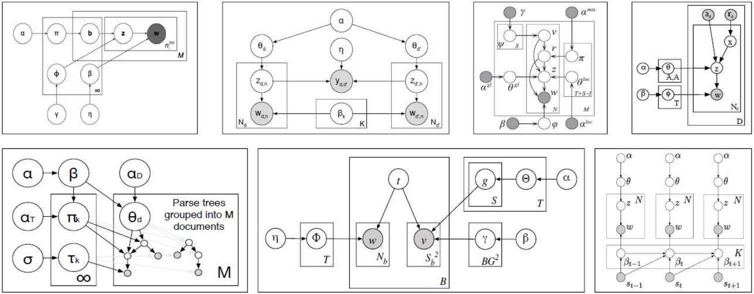
ARTM vs. Graphical Models and Bayesian Inference

In Bayesian approach, a lot of calculus to be done *for each model* to go from the problem statement to the solution algorithm:



ARTM vs. Graphical Models and Bayesian Inference

In Bayesian approach, Graphical Models are used to make model representation clear:



ARTM vs. Graphical Models and Bayesian Inference

In ARTM, a general system of equations holds *for all the models*, each model represented by its own regularizer $R(\Phi, \Theta)$:

$$\begin{cases} p_{tdw} = \operatorname*{norm}_{t}\big(\phi_{wt}\theta_{td}\big) \\ \phi_{wt} = \operatorname*{norm}_{w}\left(\sum_{d} n_{dw} p_{tdw} + \phi_{wt}\frac{\partial R}{\partial \phi_{wt}}\right) \\ \theta_{td} = \operatorname*{norm}_{t}\left(\sum_{w} n_{dw} p_{tdw} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right) \end{cases}$$

ARTM vs. Graphical Models and Bayesian Inference

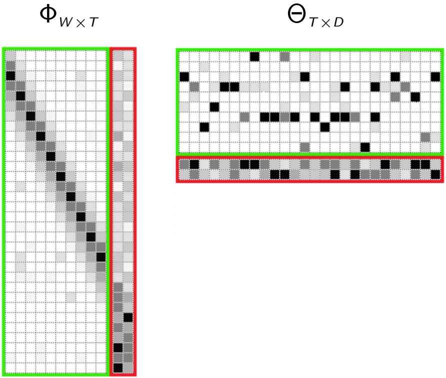
In ARTM, we can combine models, simply making a weighted sum of their regularizers $\tau_1 R_1 + \dots + \tau_k R_k$:

$$\begin{cases} p_{tdw} = \operatorname*{norm}_{t}\bigl(\phi_{wt}\theta_{td}\bigr) \\ \phi_{wt} = \operatorname*{norm}_{w}\left(\sum_{d} n_{dw} p_{tdw} + \phi_{wt} \sum_{i} \tau_i \frac{\partial R_i}{\partial \phi_{wt}}\right) \\ \theta_{td} = \operatorname*{norm}_{t}\left(\sum_{w} n_{dw} p_{tdw} + \theta_{td} \sum_{i} \tau_i \frac{\partial R_i}{\partial \theta_{td}}\right) \end{cases}$$

Assumptions: what topics would be well-interpretable?

Topics $S \subset T$ contain domain-specific terms
$p(w|t)$, $t \in S$ are sparse and different (weakly correlated)

Topics $B \subset T$ contain background terms
$p(w|t)$, $t \in B$ are dense and contain common lexis words



Smoothing regularization (rethinking LDA)

The non-sparsity assumption for background topics $t \in B$:
ϕ_{wt} are similar to a given distribution β_w;
θ_{td} are similar to a given distribution α_t.

$$\sum_{t \in B} \mathrm{KL}_w(\beta_w \| \phi_{wt}) \to \min_{\Phi}; \qquad \sum_{d \in D} \mathrm{KL}_t(\alpha_t \| \theta_{td}) \to \min_{\Theta}.$$

We minimize the sum of these KL-divergences to get a regularizer:

$$R(\Phi, \Theta) = \beta_0 \sum_{t \in B} \sum_{w \in W} \beta_w \ln \phi_{wt} + \alpha_0 \sum_{d \in D} \sum_{t \in B} \alpha_t \ln \theta_{td} \to \max.$$

The regularized M-step applied for all $t \in B$ coincides with LDA:

$$\phi_{wt} \propto n_{wt} + \beta_0 \beta_w, \qquad \theta_{td} \propto n_{td} + \alpha_0 \alpha_t,$$

which is new non-Bayesian interpretation of LDA [Blei 2003].

Sparsing regularizer (further rethinking LDA)

The sparsity assumption for domain-specific topics $t \in S$: distributions ϕ_{wt}, θ_{td} contain many zero probabilities.

We maximize the sum of KL-divergences $\text{KL}(\beta \| \phi_t)$ and $\text{KL}(\alpha \| \theta_d)$:

$$R(\Phi, \Theta) = -\beta_0 \sum_{t \in S} \sum_{w \in W} \beta_w \ln \phi_{wt} - \alpha_0 \sum_{d \in D} \sum_{t \in S} \alpha_t \ln \theta_{td} \to \max.$$

The regularized M-step gives “anti-LDA”, for all $t \in S$:

$$\phi_{wt} \propto \big(n_{wt} - \beta_0 \beta_w\big)_+, \qquad \theta_{td} \propto \big(n_{td} - \alpha_0 \alpha_t\big)_+.$$

Varadarajan J., Emonet R., Odobez J.-M. A sparsity constraint for topic models — application to temporal activity mining // NIPS-2010 Workshop on Practical Applications of Sparse Modeling: Open Issues and New Directions.

Regularization for topics decorrelation

The dissimilarity assumption for domain-specific topics $t \in S$: if topics are interpretable then they must differ significantly.

We maximize covariances between column vectors ϕ_t:

$$R(\Phi) = -\frac{\tau}{2} \sum_{t \in S} \sum_{s \in S \backslash t} \sum_{w \in W} \phi_{wt} \phi_{ws} \to \max .$$

The regularized M-step makes columns of Φ more distant:

$$\phi_{wt} \propto \Big(n_{wt} - \tau \phi_{wt} \sum_{s \in S \backslash t} \phi_{ws} \Big)_{+} .$$

Tan Y., Ou Z. Topic-weak-correlated latent Dirichlet allocation // 7th Int'l Symp. Chinese Spoken Language Processing (ISCSLP), 2010. — Pp. 224–228.

ARTM: available regularizers

- topic smoothing ($\Leftrightarrow$ Latent Dirichlet Allocation)
- topic sparsing
- topic decorrelation
- topic selection via entropy sparsing
- topic coherence maximization
- supervised learning for classification and regression
- semi-supervised learning
- using documents citation and links
- modeling temporal topic dynamics
- using vocabularies in multilingual topic models
- etc.

Vorontsov K. V., Potapenko A. A. Additive Regularization of Topic Models. Machine Learning Journal. Springer, 2014.

BigARTM project

BigARTM features:

- Parallel + Online + Multimodal + Regularized Topic Modeling
- Out-of-core processing of Big Data
- Built-in library of regularizers and quality measures

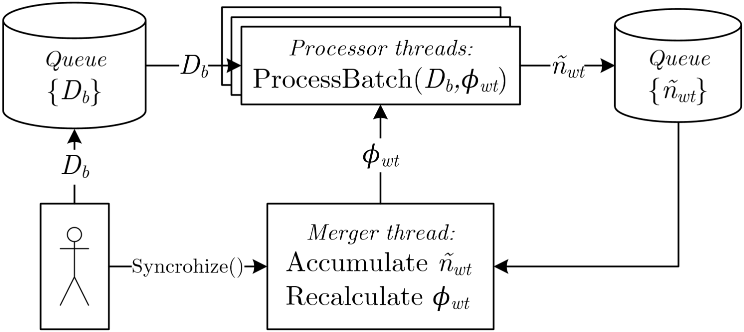
BigARTM community:

- Open-source https://github.com/bigartm
  (discussion group, issue tracker, pull requests)
- Documentation http://bigartm.org

BigARTM license and programming environment:

- Freely available for commercial usage (BSD 3-Clause license)
- Cross-platform — Windows, Linux, Mac OS X (32 bit, 64 bit)
- Programming APIs: command-line, C++, and Python

The BigARTM project: parallel architecture



- Concurrent processing of batches $D = D_1 \sqcup \cdots \sqcup D_B$
- Simple single-threaded code for *ProcessBatch*
- User controls when to update the model in online algorithm
- Deterministic (reproducible) results from run to run

Fast online EM-algorithm for regularized multimodal PTMs

Input: collection D split into batches D_b, $b = 1, \dots, B$;
Output: matrix Φ;

1 initialize ϕ_{wt} for all $w \in W$, $t \in T$;
2 $n_{wt} := 0$, $\tilde{n}_{wt} := 0$ for all $w \in W$, $t \in T$;
3 **for all** batches D_b, $b = 1, \dots, B$
4 iterate each document $d \in D_b$ at a constant matrix Φ:
$(\tilde{n}_{wt}) := (\tilde{n}_{wt}) + \textbf{ProcessBatch}(D_b, \Phi)$;
5 **if** (synchronize) **then**
6 $n_{wt} := n_{wt} + \tilde{n}_{dw}$ for all $w \in W$, $t \in T$;
7 $\phi_{wt} := \underset{w \in W^m}{\text{norm}} \left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right)$ for all $w \in W^m$, $m \in M$, $t \in T$;
8 $\tilde{n}_{wt} := 0$ for all $w \in W$, $t \in T$;

Fast online EM-algorithm for Multi-ARTM

ProcessBatch iterates documents $d \in D_b$ at a constant matrix Φ.

matrix $(\tilde{n}_{wt})$:= **ProcessBatch** (set of documents D_b, matrix Φ)

$\tilde{n}_{wt} := 0$ for all $w \in W$, $t \in T$;

for all $d \in D_b$

  initialize $\theta_{td} := \frac{1}{|T|}$ for all $t \in T$;

  repeat

    $p_{tdw} := \underset{t \in T}{\operatorname{norm}} \big(\phi_{wt} \theta_{td}\big)$ for all $w \in d$, $t \in T$;

    $n_{td} := \sum\limits_{w \in d} \lambda_{m(w)} n_{dw} p_{tdw}$ for all $t \in T$;

    $\theta_{td} := \underset{t \in T}{\operatorname{norm}} \big(n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\big)$ for all $t \in T$;

  until θ_d converges;

  $\tilde{n}_{wt} := \tilde{n}_{wt} + \lambda_{m(w)} n_{dw} p_{tdw}$ for all $w \in d$, $t \in T$;

Summary of ARTM approach

EM-algorithm is computationally effective:

- It has linear time complexity $O(n \cdot |T| \cdot \text{nIter})$
- Its online version makes only one pass through big collection
- Parallelism is possible for both multi-core CPUs and clusters

ARTM reduces barriers to entry into PTM research field:

- General EM-algorithm for many models and their combinations
- PLSA, LDA, and 100s of PTMs are covered by ARTM
- Combining multiple modalities and regularizers is easy
- No complicated Bayesian inference and graphical models

Open problem / Under development:

- Adaptive optimization of regularization coefficients τ_i, λ_m

BigARTM vs Gensim vs Vowpal Wabbit

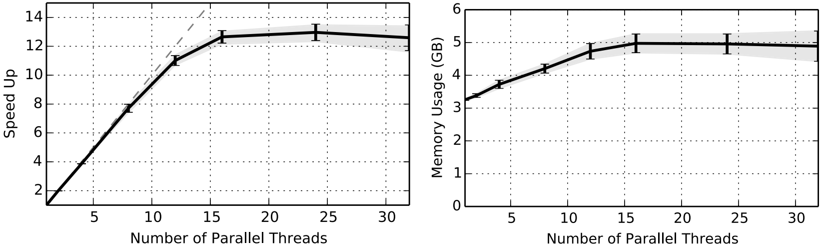
- 3.7M articles from Wikipedia, 100K unique words

	procs	train	inference	perplexity
BigARTM	1	35 min	72 sec	4000
Gensim.LdaModel	1	369 min	395 sec	4161
VowpalWabbit.LDA	1	73 min	120 sec	4108
BigARTM	4	9 min	20 sec	4061
Gensim.LdaMulticore	4	60 min	222 sec	4111
BigARTM	8	4.5 min	14 sec	4304
Gensim.LdaMulticore	8	57 min	224 sec	4455

- *procs* = number of parallel threads
- *inference* = time to infer θ_d for 100K held-out documents
- *perplexity* is calculated on held-out documents.

Running BigARTM in parallel

- 3.7M articles from Wikipedia, 100K unique words

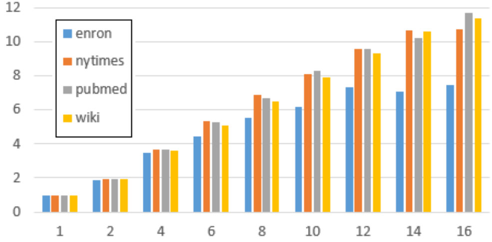


- Amazon EC2 c3.8xlarge (16 physical cores + hyperthreading)
- No extra memory cost for adding more threads

Running BigARTM on large collections

collection	$\lvert W \rvert$, 10^3	$\lvert D \rvert$, 10^6	n, 10^6	size, GB
enron	28	0.04	6.4	0.07
nytimes	103	0.3	100	0.13
pubmed	141	8.2	738	1.0
wiki	100	3.7	1009	1.2

speedup

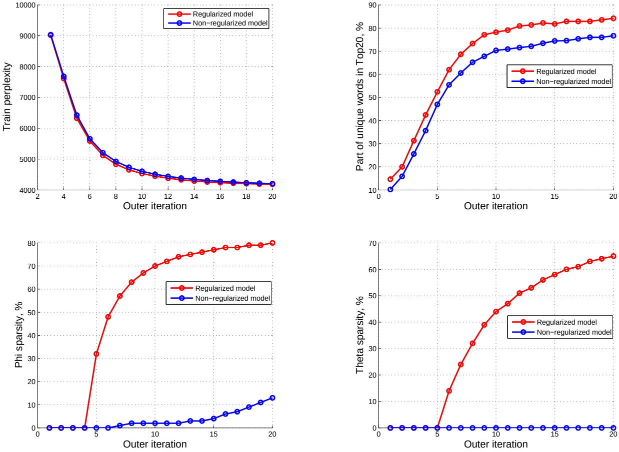


processors

Amazon EC2 cc2.8xlarge instance:
16 cores + hyperthreading, Intel® Xeon® CPU E5-2670 2.6GHz.

Running BigARTM with multiple regularizers

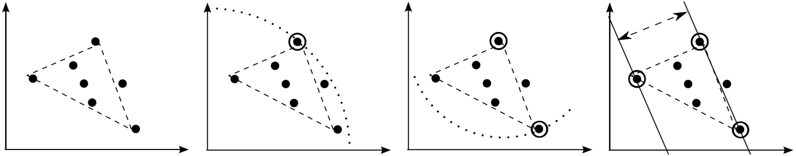
ARTM combines regularizers to improve sparsity and number of topical words without a loss of the perplexity.



Arora's algorithm based on anchor words recovery

Def. The word w is an *anchor word* of the topic t if $p(w|t) = 1$.

Arora's algorithm finds Φ with the identity submatrix of anchors

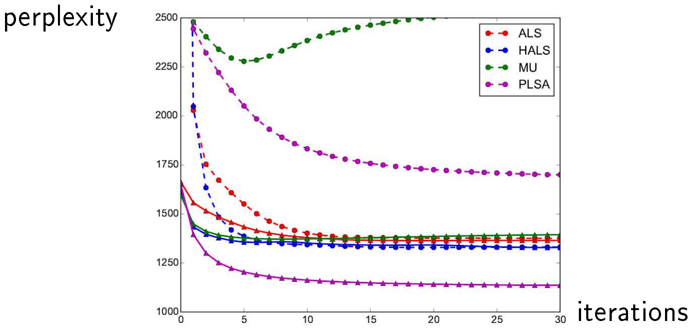


- ⊕ The fastest algorithm for Topic Modeling
- ⊕ Theoretical guarantees for polynomial time and global optimum
- ⊖ The hypothesis that $\forall t$ the anchor word exists is restrictive
- ⊖ The algorithm is not so fast for big vocabularies $|W|$

Sanjeev Arora et al. A Practical Algorithm for Topic Modeling with Provable Guarantees. ICML 2013.

Arora's algorithm for EM-algorithms initialization

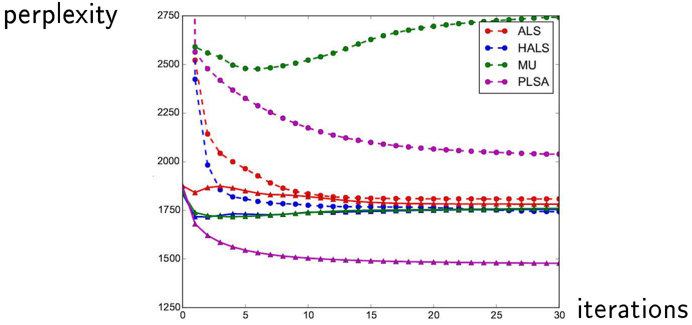
ALS, HALS, MU — methods for nonnegative matrix factorization
NIPS collection: $|D| = 1500$, $|W| = 12419$, $|T| = 25$.
Perplexity $= \exp\left(-\frac{1}{n}\mathscr{L}(\Phi, \Theta)\right)$



solid lines — initialization by Arora's algorithm
dotted lines — random initialization

Arora's algorithm for EM-algorithms initialization

ALS, HALS, MU — methods for nonnegative matrix factorization
Daily Kos collection: $|D| = 3430$, $|W| = 6906$, $|T| = 25$.
Perplexity $= \exp\left(-\frac{1}{n}\mathscr{L}(\Phi, \Theta)\right)$



solid lines — initialization by Arora's algorithm
dotted lines — random initialization

Conclusions about Arora's initialization

- Arora's initialization greatly improves PLSA (PLSA — Probabilistic Latent Semantic Analysis is equivalent to ARTM without regularization)
- Arora's initialization does not improve quadratic loss minimizers
- PLSA is capable of improving the Arora's initialization, perhaps, because of restrictive assumptions of Arora's algorithm do not hold in the real data

Regularization for topic selection

Let us maximize KL-divergence: $\mathrm{KL}\big(\frac{1}{|T|} \,\|\, p(t)\big) \to \max$
to make distribution over topics $p(t)$ sparse:

$$R(\Theta) = -\tau n \sum_{t \in S} \frac{1}{|T|} \ln \underbrace{\sum_{d \in D} p(d)\theta_{td}}_{p(t)} \to \max .$$

The regularized M-step formula results in Θ row sparsing:

$$\theta_{td} = \operatorname*{norm}_{t \in T} \left(n_{td} \left(1 - \tau \frac{n}{n_t |T|} \right) \right).$$

The row sparsing effect:
if $n_t < \tau \frac{n}{|T|}$ then all values in the t-th row turn into zeros.

The experiments with topic selection

Real dataset: NIPS (Neural Information Processing System)

- $|D| = 1566$ preprocessed papers from NIPS conference;
- vocabulary: $|W| \approx 1.3 \cdot 10^4$; hold-out set: $|D'| = 174$.

Synthetic dataset:

- 500 EM iterations for PLSA with $|T_0| = 50$ topics on NIPS
- generate synthetic dataset (n^0_{dw}) using obtained Φ and Θ:

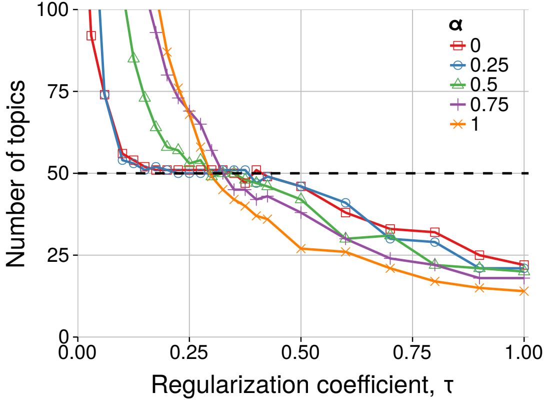
$$n^0_{dw} = n_d \sum_{t \in T} \phi_{wt} \theta_{td}$$

Parametric family of semi-real datesets:

- (n^α_{dw}) is a mixture of synthetic (n^0_{dw}) and real (n_{dw}) datasets:

$$n^\alpha_{dw} = \alpha n_{dw} + (1 - \alpha) n^0_{dw}$$

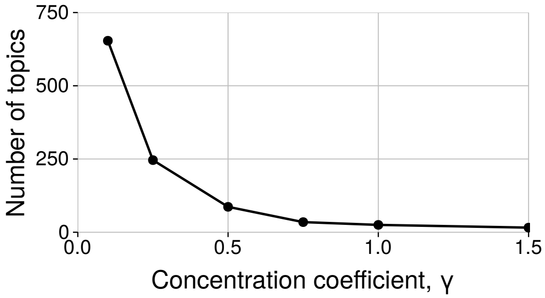
Number of topics determination



- For synthetic dataset ARTM reliably finds the truth: $|T| = 50$.
- The range of τ values leading to the correct number is wide.
- For real data the number of topics is not clear.

Comparison to HDP topic model

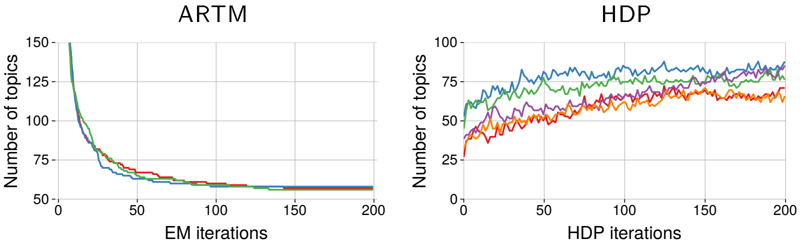
HDP (Hierarchical Dirichlet Process, Tech et. al, 2006)
is the state-of-art approach for a number of topics optimization.



- The choice of the concentration coefficient γ of Dirichlet process may lead to nearly any number of topics.

Stability of ARTM vs HDP

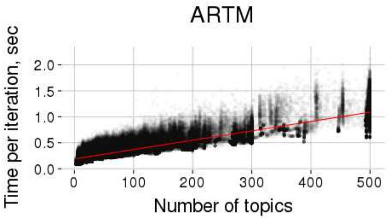
Starting ARTM and HDP many times from different initializations:

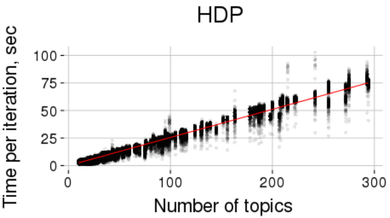


1. HDP is less stable in two ways:
   1. The number of topics fluctuates from iteration to iteration
   2. The results for several random starts significantly differ
2. The “recommended” parameters γ for HDP and τ for ARTM give the similar number of topics ≈ 60

Running time of ARTM vs HDP

Comparing the running time per iteration (sec) of ARTM vs HDP (one iteration = one pass through the collection)

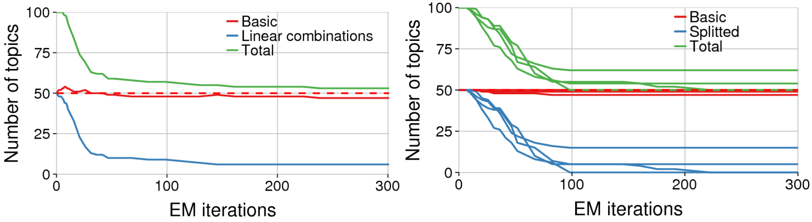




- Our method is 100 times faster!

Elimination of linearly dependent topics and nested subtopics

- Add 50 linear combinations of topics in synthetic dataset.
- Add 50 nested subtopics in synthetic dataset.



- Our regularizer effectively eliminates both linearly dependent topics and nested subtopics from the model
- More diverse topics of the original model remain.

Conclusions about number of topics

- It seems that the “true number of topics” does not exist in real text collections.
- ARTM has a special regularizer for topic selection, which eliminates small, linearly dependent, and nested topics.
- It is faster and more stable than state-of-the-art HDP.

Conclusions

- Topic Modeling is an applied area of optimization and matrix factorization in text analysis
- ARTM (Additive Regularization) is a semi-probabilistic non-Bayesian multicriteria view on Topic Modeling
- BigARTM is open source project for parallel online multimodal regularized Topic Modeling of large collections

Contacts:

Konstantin Vorontsov: voron@forecsys.ru

Wiki www.MachineLearning.ru User:Vokov (in Russian)