Parallel Non-blocking Deterministic Algorithm for Online Topic Modeling

Murat Apishev
great-mel@yandex.ru
Oleksandr Frei
oleksandr.frei@gmail.com

HSE, MSU, MIPT

April 8, 2016

Contents

Topic modeling

Topic modeling — an application of machine learning to statistical text analysis.

Topic — a specific terminology of the subject area, the set of terms (unigrams or $n-$grams) frequently appearing together in documents.

Topic model uncovers latent semantic structure of a text collection:

- *topic t* is a probability distribution $p(w|t)$ over terms w
- *document d* is a probability distribution $p(t|d)$ over topics t

Applications — information retrieval for long-text queries, classification, categorization, summarization of texts.

Topic modeling task

Given: W — set (vocabulary) of terms (unigrams or $n-$grams),
D — set (collection) of text documents $d \subset W$,
n_{dw} — how many times term w appears in document d.

Find: model $p(w|d) = \sum\limits_{t \in T} \phi_{wt}\theta_{td}$ with parameters $\underset{W \times T}{\Phi}$ и $\underset{T \times D}{\Theta}$:
$\phi_{wt} = p(w|t)$ — term probabilities w in each topic t,
$\theta_{td} = p(t|d)$ — topic probabilities t in each document d.

Criteria log-likelihood maximization:

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \ln \sum_{t \in T} \phi_{wt}\theta_{td} \rightarrow \max_{\phi,\theta};$$

$$\phi_{wt} \geqslant 0; \quad \textstyle\sum_w \phi_{wt} = 1; \qquad \theta_{td} \geqslant 0; \quad \textstyle\sum_t \theta_{td} = 1.$$

Issue: the problem of stochastic matrix factorization is *ill-posed*:
$\Phi\Theta = (\Phi S)(S^{-1}\Theta) = \Phi'\Theta'$.

PLSA and EM-algorithm

Log-likelihood maximization:

$$\sum_{d \in D} \sum_{w \in W} n_{dw} \ln \sum_{t} \phi_{wt} \theta_{td} \rightarrow \max_{\Phi, \Theta}$$

EM-algorithm: the simple iteration method for the set of equations

$$\begin{array}{rl} \text{E-шаг:} & \left\{ \begin{array}{l} p_{tdw} = \operatorname*{norm}_{t \in T} \left(\phi_{wt} \theta_{td} \right) \\ \phi_{wt} = \operatorname*{norm}_{w \in W} \left(n_{wt} \right), \quad n_{wt} = \sum\limits_{d \in D} n_{dw} p_{tdw} \\ \theta_{td} = \operatorname*{norm}_{t \in T} \left(n_{td} \right), \quad n_{td} = \sum\limits_{w \in d} n_{dw} p_{tdw} \end{array} \right. \\ \text{M-шаг:} & \end{array}$$

where $\operatorname*{norm}_{i \in I} x_i = \frac{\max\{x_i, 0\}}{\sum\limits_{j \in I} \max\{x_j, 0\}}$

ARTM and regularized EM-algorithm

Log-likelihood maximization with additive regularization criterion R:

$$\sum_{d \in D} \sum_{w \in W} n_{dw} \ln \sum_{t} \phi_{wt} \theta_{td} + R(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta}$$

EM-algorithm: the simple iteration method for the set of equations

$$\begin{array}{ll} \text{E-шаг:} & \left\{ \begin{array}{l} p_{tdw} = \underset{t \in T}{\operatorname{norm}} \big(\phi_{wt} \theta_{td} \big) \\ \phi_{wt} = \underset{w \in W}{\operatorname{norm}} \Big(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \Big), \quad n_{wt} = \sum\limits_{d \in D} n_{dw} p_{tdw} \\ \theta_{td} = \underset{t \in T}{\operatorname{norm}} \Big(n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \Big), \quad n_{td} = \sum\limits_{w \in d} n_{dw} p_{tdw} \end{array} \right. \\ \text{M-шаг:} & \end{array}$$

Examples of regularizers

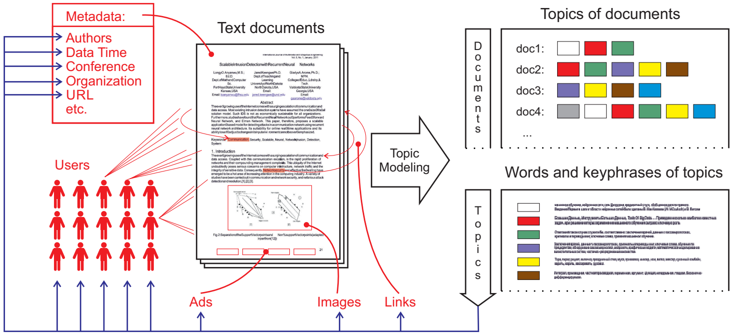
Many Bayesian models can be reinterpreted as regularizers in ARTM.

Some examples of regularizes:

1. Smoothing Φ / Θ (leads to popular LDA model)
2. Sparsing Φ / Θ
3. Decorrelation of topics in Φ
4. Semi-supervised learning
5. Topic coherence maximization
6. Topic selection
7. ...

Multimodal Topic Model

Multimodal Topic Model finds topical distributions for terms $p(w|t)$, authors $p(a|t)$, time $p(y|t)$, objects of 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.



M-ARTM and multimodal regularized EM-algorithm

W^m is a vocabulary of terms of m-th modality, $m \in M$,
$W = W^1 \sqcup W^m$ as a joint vocabulary of all modalities

Multimodal log-likelihood maximization with additive regularization criterion R:

$$\sum_{m \in M} \lambda_m \sum_{d \in D} \sum_{w \in W^m} n_{dw} \ln \sum_t \phi_{wt} \theta_{td} + R(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta}$$

EM-algorithm: the simple iteration method for the set of equations

$$\begin{array}{ll} \text{E-шаг:} & \left\{ \begin{array}{l} p_{tdw} = \underset{t \in T}{\operatorname{norm}} \big(\phi_{wt} \theta_{td} \big) \\ \\ \phi_{wt} = \underset{w \in W^m}{\operatorname{norm}} \left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right), \quad n_{wt} = \sum_{d \in D} \lambda_{m(w)} n_{dw} p_{tdw} \\ \\ \theta_{td} = \underset{t \in T}{\operatorname{norm}} \left(n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \right), \quad n_{td} = \sum_{w \in d} \lambda_{m(w)} n_{dw} p_{tdw} \end{array} \right. \\ \text{M-шаг:} & \end{array}$$

BigARTM project

BigARTM features:

- Fast[1] parallel and online processing of Big Data;
- Multimodal and regularized topic modeling;
- Built-in library of regularizers and quality measures;

BigARTM community:

- Open-source `https://github.com/bigartm`
- 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++, Python

[1] Vorontsov K., Frei O., Apishev M., Romov P., Dudarenko M. BigARTM: Open Source Library for Regularized Multimodal Topic Modeling of Large Collections Analysis of Images, Social Networks and Texts. 2015

BigARTM vs. Gensim vs. Vowpal Wabbit LDA

- 3.7M articles from Wikipedia, 100K unique words

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

- *procs* = number of parallel threads
- *inference* = time to infer θ_d for 100K held-out documents
- *perplexity* $\mathscr{P}$ is calculated on held-out documents

$$\mathscr{P}(D) = \exp\left(-\frac{1}{n}\sum_{d\in D}\sum_{w\in d} n_{dw} \ln \sum_{t\in T} \phi_{wt}\theta_{td}\right), \quad n = \sum_{d} n_d.$$

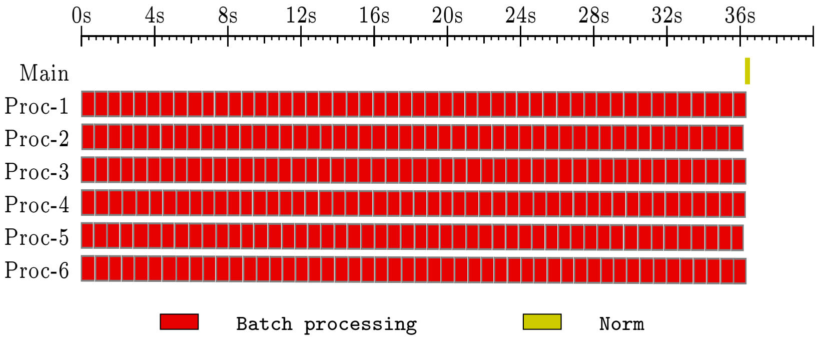
Offline algorithm

- The collection is split into *batches*.
- Offline algorithm performs scans over the collection.
- Each thread process one batch at a time, inferring n_{wt} and θ_{td} (using Θ regularization).
- After each scan algorithm recalculates Φ matrix and apply Φ regularizers according to the equation

$$\phi_{wt} = \operatorname*{norm}_{w \in W} \left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right).$$

- The implementation never stores the entire Θ matrix at any given time.

Offline algorithm: Gantt chart

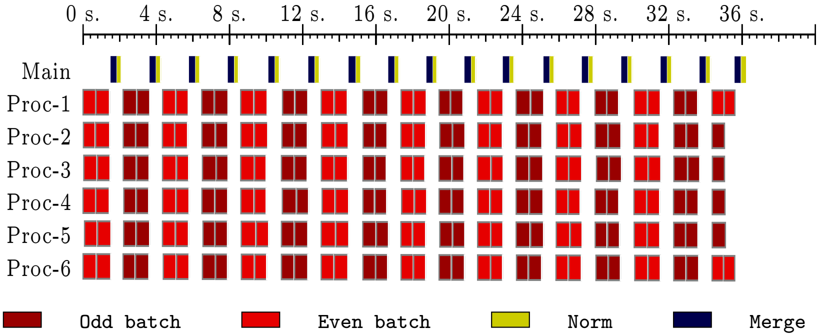


- This and further Gantt charts were created using the NYTimes dataset: `https://archive.ics.uci.edu/ml/datasets/Bag+of+Words`.
- Size of dataset is $\approx$ 300k documents, but each algorithm was run on some subset (from 70% to 100%) to archive the $\approx$ 36 sec. working time.

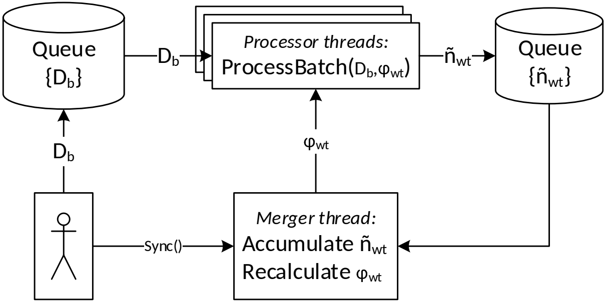
Online algorithm

- The algorithm is a generalization of Online variational Bayes algorithm for LDA model.
- Online ARTM improves the convergence rate of the Offline ARTM by re-calculating matrix Φ after every η batches.
- Better suited for large and heterogeneous text collections.
- Weighted sum of n_{wt} from previous and current η batches to control the importance of new information.
- **Issue**: all threads has no useful work to do during the update of Φ matrix.

Online algorithm: Gantt chart

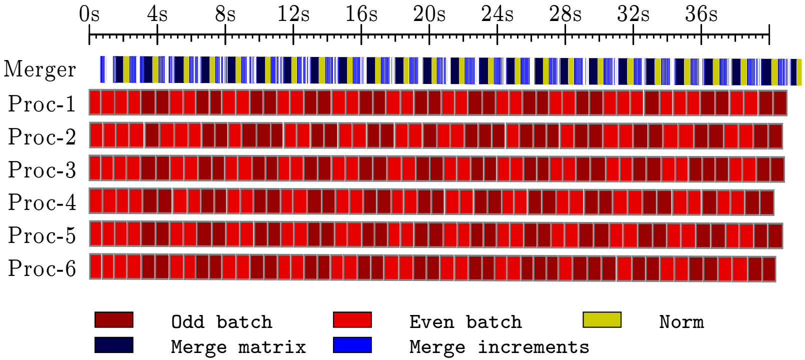


Async: Asynchronous online algorithm

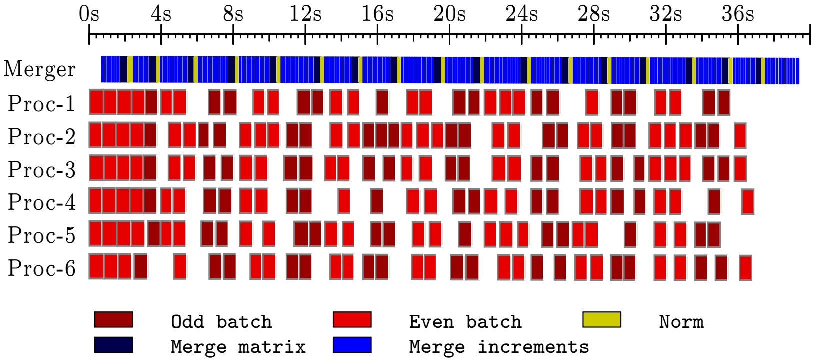


- Faster asynchronous implementation (it was compared with Gensim and VW LDA)
- **Issue**: `Merger` and `DataLoader` can become a bottleneck.
- **Issue**: the result of such algorithm is *non-deterministic*.

Async: Gantt chart in normal case



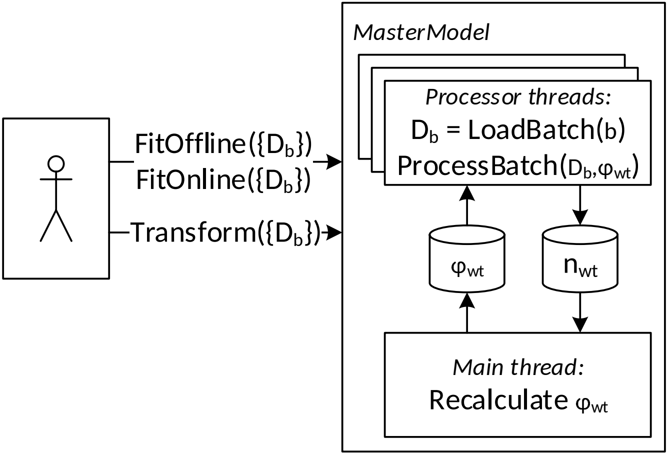
Async: Gantt chart in bad case



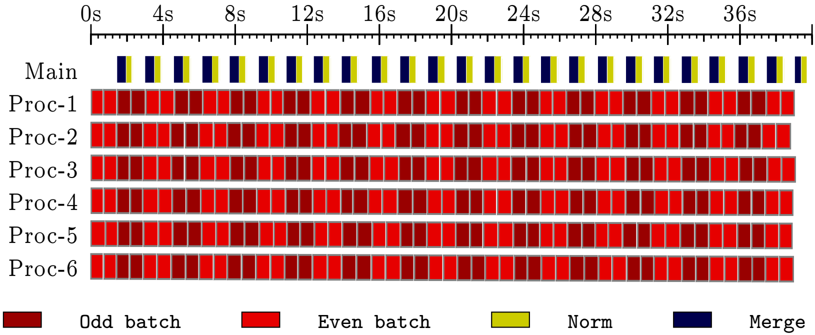
DetAsync: Deterministic asynchronous online algorithm

- To avoid the indeterministic behavior lets replace the update after *first* η batches with update after *given* η batches.
- Remove `Merger` and `DataLoader` threads. Each `Processor` thread reads batches and writes results into n_{wt} matrix by itself.
- `Processor` threads get a set of batches to process, start processing and immediately return a *future* object to main thread.
- The main thread can process the updates of Φ matrix while `Processor` threads work, and then get the result by passing received *future* object to `Await` function.

DetAsync: schema



DetAsync: Gantt chart



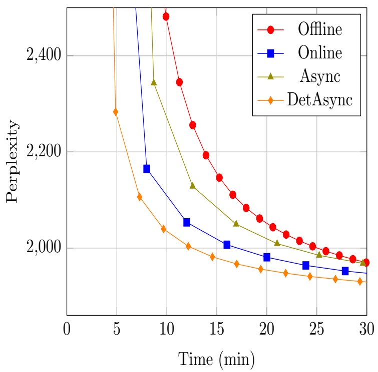
Experiments

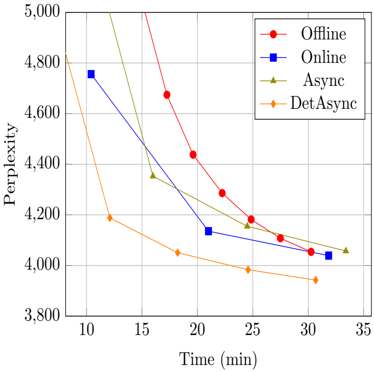
- Datasets: *Wikipedia* ($|D| = 3.7$M articles, $|W| = 100$K words), *Pubmed* ($|D| = 8.2$M abstracts, $|W| = 141$K words).
- Node: Intel Xeon CPU E5-2650 v2 system with 2 processors, 16 physical cores in total (32 with hyper-threading).
- Metric: perplexity $\mathscr{P}$ value achieved in the allotted time.
- Time: each algorithm was time-boxed to run for a 30 minutes.

Peak memory usage (Gb):

	$\lvert T\rvert$	Offline	Online	DetAsync	Async (v0.6)
Pubmed	1000	5.17	4.68	8.18	13.4
Pubmed	100	1.86	1.62	2.17	3.71
Wiki	1000	1.74	2.44	3.93	7.9
Wiki	100	0.54	0.53	0.83	1.28

Reached perplexity value





Wikipedia (left), Pubmed (right).

DetAsync achives best perplexity in given time-box.

Mining ethnic-related content from blogosphere

Development of concept and methodology for multi-level monitoring of the state of inter-ethnic relations with the data from social media.

The objectives of Topic Modeling in this project:

1. Identify ethnic topics in social media big data
2. Identify event and permanent ethnic topics
3. Identify spatio-temporal patterns of the ethnic discourse
4. Estimate the sentiment of the ethnic discourse
5. Develop the monitoring system of inter-ethnic discourse

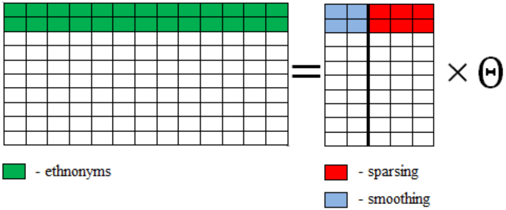
The Russian Science Foundation grant 15-18-00091 (2015–2017)
(Higher School of Economics, St. Petersburg School of Social Sciences and Humanities, Internet Studies Laboratory LINIS)

Example ethnonyms for semi-supervised topic modeling

османский	русич
восточноевропейский	сингапурец
эвенк	перуанский
швейцарская	словенский
аланский	вепсский
саамский	ниггер
латыш	адыги
литовец	сомалиец
цыганка	абхаз
ханты-мансийский	темнокожий
карачаевский	нигериец
кубинка	лягушатник
гагаузский	камбоджиец

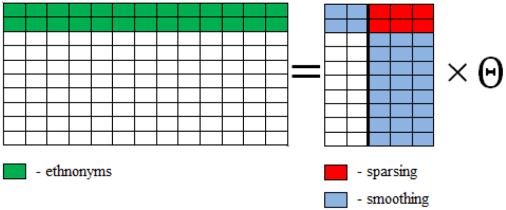
Regularization for finding ethnic topics

- smoothing ethnonyms in ethnic topics
- sparsing ethnonyms in background topics
-
-
-



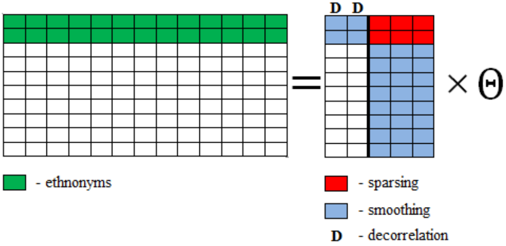
Regularization for finding ethnic topics

- smoothing ethnonyms in ethnic topics
- sparsing ethnonyms in background topics
- smoothing non-ethnonyms for background topics
-
-



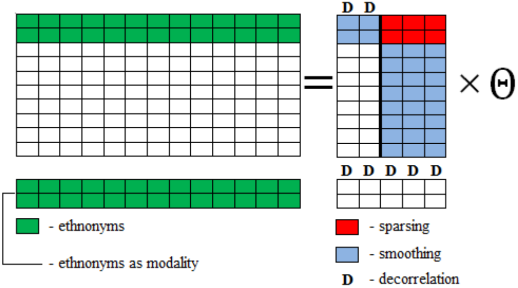
Regularization for finding ethnic topics

- smoothing ethnonyms in ethnic topics
- sparsing ethnonyms in background topics
- smoothing non-ethnonyms in background topics
- decorrelating ethnic topics
-



Regularization for finding ethnic topics

- smoothing ethnonyms in ethnic topics
- sparsing ethnonyms in background topics
- smoothing non-ethnonyms in background topics
- decorrelating ethnic topics
- adding ethnonyms modality and decorrelating their topics



Experiment

- LiveJournal collection: 1.58M of documents
- 860K of words in the raw vocabulary after lemmatization
- 90K of words after filtering out
  - short words with length $\leqslant 2$,
  - rare words with $n_w < 20$ including:
  - non-Russian words
- 250 ethnonyms

Semi-supervised ARTM for ethnic topic modeling

The number of ethnic topics found by the model:

model	ethnic $\lvert S\rvert$	background $\lvert B\rvert$	$++$	$+-$	$-+$	coh_{20} [2]	$tfidf_{20}$
PLSA		400	12	15	17	-1447	-1012
LDA		400	12	15	17	-1540	-1121
ARTM-4	250	150	21	27	20	-1651	-1296
ARTM-5	250	150	38	42	30	-1342	-908

- ARTM-4:
  - ethnic topics: sparsing and decorrelating, ethnonyms smoothing
  - background topics: smoothing, ethnonyms sparsing
- ARTM-5:
  - ARTM-4 + ethnonyms as additional modality

[2] Coherence and TF-IDF coherence are metrics that match the human judgment of topic quality. The topic is better if it has higher coherence value.

Ethnic topics examples

(русские): русский, князь, россия, татарин, великий, царить, царь, иван, император, империя, грозить, государь, век, московская, екатерина, москва,
(русские): акция, организация, митинг, движение, активный, мероприятие, совет, русский, участник, москва, оппозиция, россия, пикет, протест, проведение, националист, поддержка, общественный, проводить, участие,
(славяне, византийцы): славянский, святослав, жрец, древние, письменность, рюрик, летопись, византия, мефодий, хазарский, русский, азбука,
(сирийцы): сирийский, асад, боевик, район, террорист, уничтожать, группировка, дамаск, оружие, алесио, оппозиция, операция, селение, сша, нусра, турция,
(турки): турция, турецкий, курдский, эрдоган, стамбул, страна, кавказ, горин, полиция, премьер-министр, регион, курдистан, ататюрк, партия,
(иранцы): иран, иранский, сша, россия, ядерный, президент, тегеран, сирия, оон, израиль, переговоры, обама, санкция, исламский,
(палестинцы): террорист, израиль, терять, палестинский, палестинец, террористический, палестина, взрыв, территория, страна, государство, безопасность, арабский, организация, иерусалим, военный, полиция, газ,
(ливанцы): ливанский, боевик, район, ливан, армия, террорист, али, военный, хизбалла, раненый, уничтожать, сирия, подразделение, квартал, армейский,
(ливийцы): ливан, демократия, страна, ливийский, каддафи, государство, алжир, война, правительство, сша, арабский, али, муаммар, сирия,
(евреи): израиль, израильский, страна, израил, война, нетаньяху, тель-авив, время, сша, сирия, египет, случай, самолет, еврейский, военный, ближний,

Conclusions

- BigARTM is an open-source library supporting multimodal ARTM theory.
- Fast implementation of the underlying online EM-algorithm was even more improved. Memory usage was reduced.
- Combination of 8 regularizers in the task of ethnic topics extraction showed the supirity of ARTM approach.
- BigARTM is using to process more than 20 collections in several different projects.

Join our comunity!
Contacts: bigartm.org, great-mel@yandex.ru