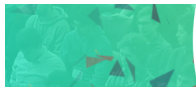


Тематическое моделирование на пути к разведочному информационному поиску

Воронцов Константин Вячеславович
ФИЦ ИУ РАН • МФТИ • МГУ • ВШЭ • Яндекс



Data Fest³



Москва • 10–11 сентября 2016

Технологии информационного поиска сделали научные знания доступнее. Их стало легче найти, но это не означает, что в них стало легче разобраться. Ответ на вопрос «где находится передний край науки по данной теме» по-прежнему требует времени, квалификации и личного общения с экспертами.

Разведочный поиск (exploratory search) — это новая парадигма в информационном поиске, нацеленная на дальнейшее устранение барьеров между Человеком и Знанием.

Разведочный поиск объединяет и автоматизирует процессы поиска, систематизации и усвоения знаний.

В докладе рассматриваются методы *вероятностного тематического моделирования* больших текстовых коллекций и их применение для тематического разведочного поиска.

1 Разведочный информационный поиск

- Разведочный поиск
- Дальнее чтение и визуализация
- Сценарий разведочного поиска

2 Тематическое моделирование

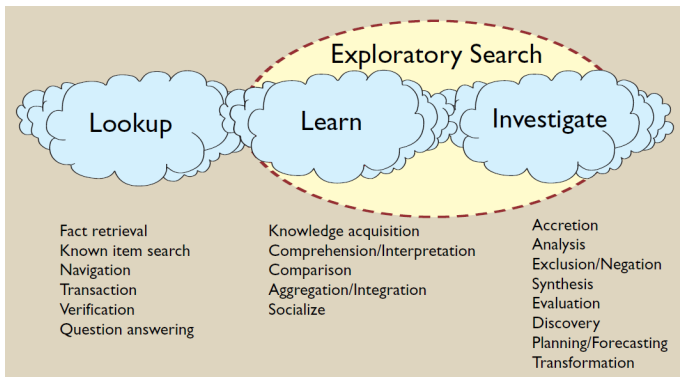
- Вероятностные модели порождения текста
- Теория аддитивной регуляризации (ARTM)
- Проект BigARTM

3 Эксперименты и приложения

- Разведочный поиск для habrahabr.ru
- Тематизация редкого контента
- Интерпретируемость тем

Концепция разведочного поиска (exploratory search)

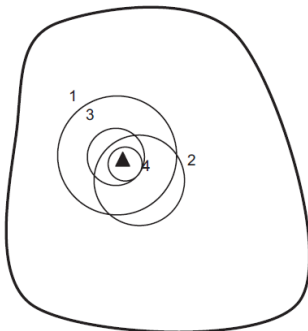
- пользователь может не знать ключевых терминов
- пользователя может интересовать множество ответов



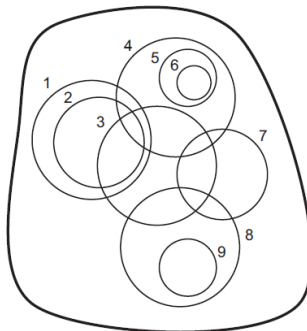
Gary Marchionini. Exploratory Search: from finding to understanding. 2006.

От итераций «query-browse-refine» к разведочному поиску

Iterative Search



Exploratory Search



▲ Search target



Information space

○# Result sets (larger = more results, intersection = overlap, # = iteration)

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

От ближнего чтения (close reading) к дальнему (distant reading)

Information Seeking Mantra [B.Shneiderman, 1996]

«Overview first, **zoom and filter, details on demand**»

Понятие *дальнего чтения* [Franco Moretti, 2005]

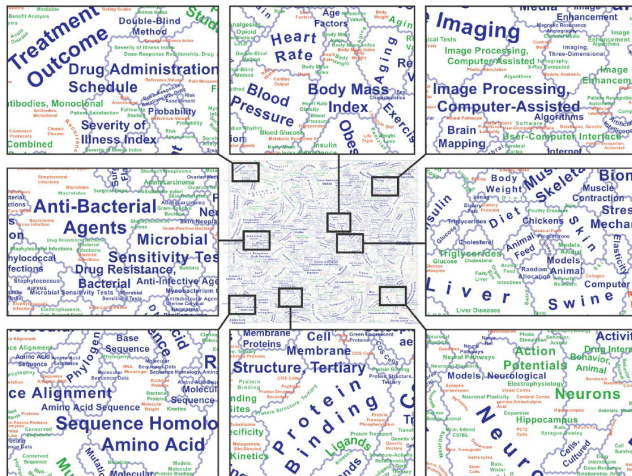
«*Distant reading* is not an obstacle but a specific form of knowledge: fewer elements, hence a sharper sense of their overall interconnection. Shapes, relations, structures. Forms. Models.»

B.Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. Visual Languages, 1996.

F.Moretti. Graphs, Maps, Trees: Abstract Models for a Literary History. 2005.

S.Janicke, G.Franzini, M.F.Cheema, G.Scheuermann. On Close and Distant Reading in Digital Humanities: A Survey and Future Challenges. EuroVis, 2015.

Пример карты медицинских знаний



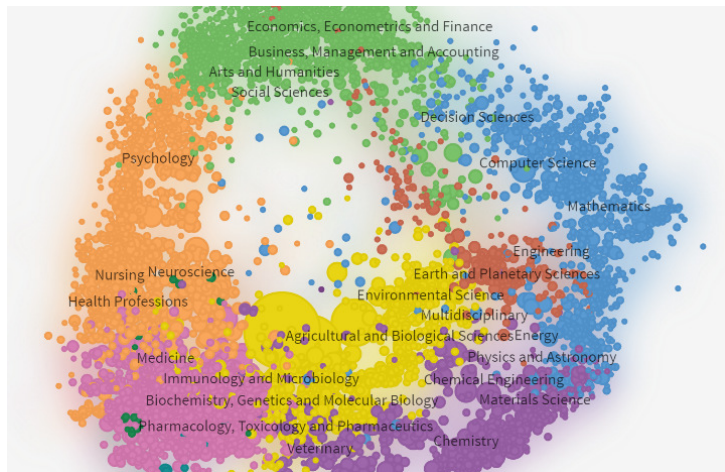
Skupin, Biberstine, Borner. Visualizing the Topical Structure of the Medical Sciences: A Self-Organizing Map Approach. PLoS ONE, 2013.

Пример иерархической карты области *Data Mining*



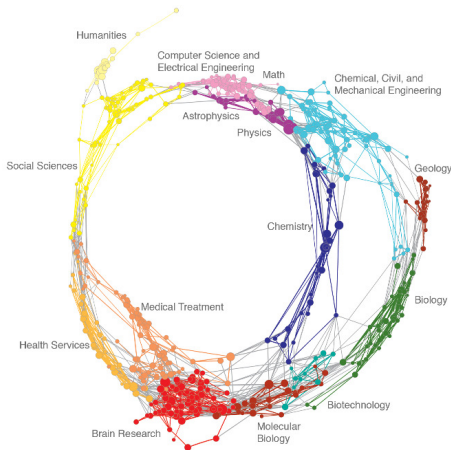
FoamTree: <https://carrotsearch.com/foamtree>

Пример карты науки



<http://onlinelibrary.wiley.com/browse/subjects>

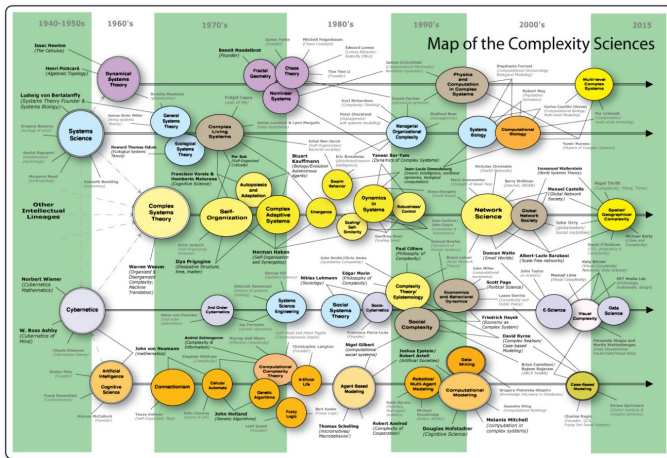
Ещё один пример карты науки



Недостатки: неинтерпретируемость осей, искажение сходства

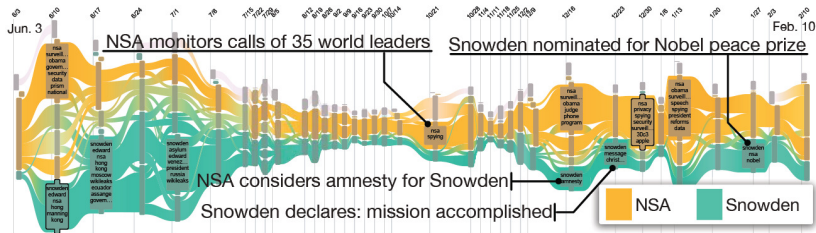
<http://scimaps.org>

Пример карты предметной области, построенной вручную



<http://www.theoryculturesociety.org/brian-castellani-on-the-complexity-sciences>

Динамика тем: эволюция предметной области



Эволюция выбранных тем иерархии. Данные Prism (2013/06/03–2014/02/09)

- эксперт задаёт сечение иерархии (дерева) тем,
- интерактивно выбирает подмножество тем и событий,
- генерирует отчёт.

Weiwei Cui, Shixia Liu, Zhuofeng Wu, Hao Wei. How hierarchical topics evolve in large text corpora. IEEE Trans. Vis. Comput. Graph. 2014.

<http://textvis.lnu.se>

Интерактивный обзор ~~170~~ 330 средств визуализации текстов



Возможный сценарий разведочного поиска

Поисковый запрос:

- документ любой длины или даже коллекция документов

Цели поиска:

- к каким темам относится мой запрос?
- что ещё известно по этим темам?
- какова тематическая структура этой предметной области?
- какие области являются смежными?
- что ещё есть понятного, обзорного, важного, свежего?

Сценарий поиска:

- 1 имея любой текст под рукой, в любом приложении,
- 2 хотим получить картину содержащихся в нём тем-подтем,
- 3 и «дорожную карту» предметной области в целом

Представление результатов поиска (концепт)

- Весь ответ на одной двумерной карте, zoom/filter/details
- Две наиболее важные оси — темы и время
- Ось (кольцо) тем: гуманитарные → естественные → точные
- Темы могут делиться на подтемы иерархически



Технологические элементы разведочного поиска

- 1 Интернет-краулинг
- 2 Фильтрация контента
- 3 Тематическое моделирование
- 4 Инвертированный индекс
- 5 Ранжирование
- 6 Визуализация
- 7 Персонализация
- 8 Монетизация

Что такое «тема»?

- *Тема* — специальная терминология предметной области.
- *Тема* — набор терминов (слов или словосочетаний), совместно часто встречающихся в документах.

Более формально,

- *тема* — условное распределение на множестве терминов, $p(w|t)$ — вероятность термина w в теме t ;
- *тематический профиль* документа — условное распределение $p(t|d)$ — вероятность темы t в документе d .

Когда автор писал термин w в документ d , он думал о теме t , и мы хотели бы выявить, о какой именно.

Тематическая модель выявляет латентные темы по наблюдаемым распределениям слов $p(w|d)$ в документах.

Прямая задача — порождение коллекции по $p(w|t)$ и $p(t|d)$

Вероятностная тематическая модель коллекции документов D описывает появление терминов w в документах d темами t :

$$p(w|d) = \sum_t p(w|t)p(t|d), \quad d \in D$$



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).

Обратная задача — восстановление $p(w|t)$ и $p(t|d)$ по коллекции

Дано: W — словарь терминов

D — коллекция текстовых документов $d = \{w_1 \dots w_{n_d}\}$

n_{dw} = сколько раз термин w встречается в документе d

Найти: параметры модели $p(w|d) = \sum_t \phi_{wt} \theta_{td}$:

$\phi_{wt} = p(w|t)$ — вероятности терминов w в каждой теме t

$\theta_{td} = p(t|d)$ — вероятности тем t в каждом документе d

Задача стохастического матричного разложения является некорректно поставленной — решение не единственно:

$$\Phi \Theta = (\Phi S)(S^{-1} \Theta) = \Phi' \Theta'$$

для невырожденных $S_{T \times T}$ таких, что Φ', Θ' — стохастические.

Модель: разумные дополнительные ограничения на Φ, Θ .

Тематическая модель для разведочного поиска должна быть...

- 1 Темпоральная: отображение динамики развития тем
- 2 Иерархическая: систематизация областей знания
- 3 Интерпретируемая: каждая тема понятна для людей
- 4 Мультиграммная: выделение тематичных словосочетаний
- 5 Разреженная: эффективность инвертированного индекса
- 6 Сегментирующая: выделение тем внутри документа
- 7 Мультимодальная: авторы, связи, тэги, пользователи,...
- 8 Мультиязычная: кросс- и много-языковой поиск
- 9 Обучаемая: учёт обратной связи с пользователями
- 10 Создающая и именующая темы автоматически
- 11 Онлайновая: обрабатывающая коллекцию за 1 проход
- 12 Параллельная, распределённая для больших коллекций

Некоторые тематические модели

- PLSA (1999) решает нерегуляризованную задачу
- LDA (2003) регуляризатор Дирихле, самая известная модель
- ATM (2004) авторы документов
- TOT (2006) метки времени документов
- HDP (2006) определение числа тем
- TNG (2007) группирование слов в мультиграммы
- CTM (2007) корреляции между темами
- NetPLSA (2008) граф связей между документами
- ML-LDA (2009) многоязычные параллельные тексты
- ssLDA (2012) частичное обучение
- Dependency-LDA (2012) классификация
- BitermTM (2013) битермы в коротких документах
- mLDA (2013) метаданные с тремя и более модальностями
- WNTM (2014) локальные контексты слов

Байесовское обучение — доминирующий подход в ВТМ

Основа подхода — байесовский вывод:

$$\text{Prior}(\Phi, \Theta) + \text{Data} \rightarrow \text{Posterior}(\Phi, \Theta).$$

Проблемы:

- Нам нужны лишь значения Φ, Θ , а не распределения
- Удобный $\text{Prior}(\Phi, \Theta)$ лингвистически не обоснован
- Байесовский вывод уникален для каждой модели
- Технически трудно комбинировать модели
- Невозможно реализовать тысячи моделей в одном коде
- Для понимания моделей используют плоскую нотацию, которая неоднозначна и только всё запутывает

Rob Zinkov. Stop using Plate Notation.

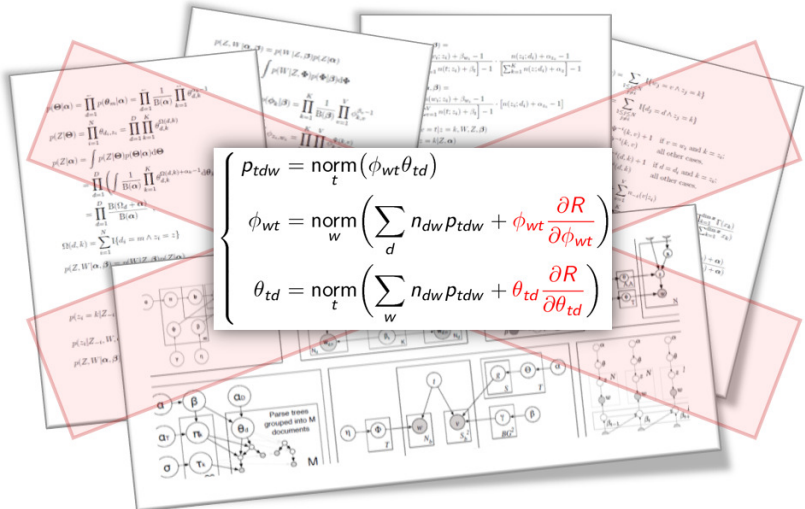
<http://zinkov.com/posts/2013-07-28-stop-using-plates>

Байесовское обучение — доминирующий подход в ВТМ

The collage contains several elements:

- Mathematical Formulas:**
 - Joint probability: $p(\mathbf{Z}, \mathbf{W} | \alpha, \beta) = p(\mathbf{W} | \mathbf{Z}, \beta) p(\mathbf{Z}, \alpha)$
 - Topic distribution: $p(\theta_i | \alpha) = \prod_{d=1}^D p(\theta_{id} | \alpha) = \prod_{d=1}^D \frac{1}{B(\alpha)} \prod_{k=1}^K \alpha_{dk}^{\theta_{id}-1}$
 - Word distribution: $p(\phi_{ij} | \beta) = \prod_{k=1}^K \frac{1}{B(\beta)} \prod_{v=1}^V \phi_{kv}^{\theta_{ij}-1}$
 - Document likelihood: $p(\mathbf{Z} | \alpha) = \int p(\mathbf{Z} | \theta) p(\theta | \alpha) d\theta$
 - Word likelihood: $p(\mathbf{W} | \mathbf{Z}, \alpha, \beta) = \prod_{i=1}^N \prod_{j=1}^M \prod_{k=1}^K \phi_{kj}^{\theta_{ik}}$
 - Dirichlet prior: $B(\alpha, \beta) = \frac{\Gamma(\alpha_1) \dots \Gamma(\alpha_K)}{\Gamma(\alpha_1 + \dots + \alpha_K)}$
 - Gamma prior: $p(\tau_i | \alpha) = \frac{\Gamma(\alpha)}{\Gamma(\alpha)^K} \prod_{k=1}^K \tau_k^{\alpha-1} e^{-\tau_k}$
- Graphical Models:**
 - Plate notation for the generative process.
 - Tree structures for hyperparameters like α and β .
 - Networks showing dependencies between variables like θ , ϕ , \mathbf{Z} , and \mathbf{W} .
 - A diagram showing "Parse trees grouped into M documents" with nodes for grammar rules and document indices.

ARTM — альтернатива байесовскому обучению



ARTM — Аддитивная Регуляризация Тематических Моделей

Максимизация log правдоподобия с регуляризатором R :

$$R(\Phi, \Theta) = \sum_i \tau_i R_i(\Phi, \Theta); \quad \sum_{d,w} n_{dw} \ln \sum_t \phi_{wt} \theta_{td} + R(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta}$$

EM-алгоритм: метод простой итерации для системы уравнений

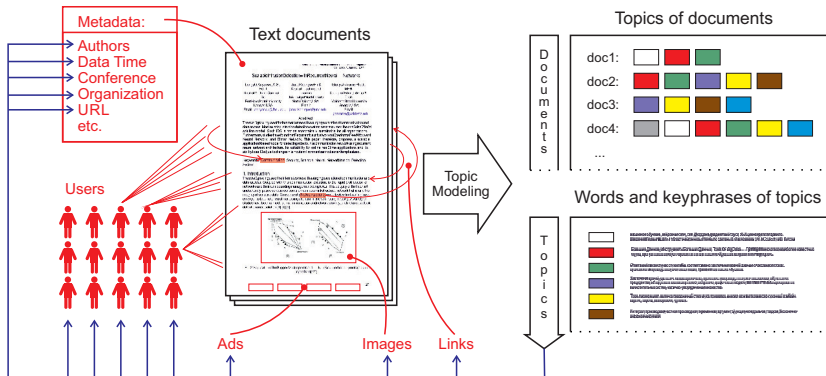
$$\begin{cases} \text{E-шаг:} & p_{tdw} = \operatorname{norm}_{t \in T}(\phi_{wt} \theta_{td}) \\ \text{M-шаг:} & \begin{cases} \phi_{wt} = \operatorname{norm}_{w \in W} \left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right), & n_{wt} = \sum_{d \in D} n_{dw} p_{tdw} \\ \theta_{td} = \operatorname{norm}_{t \in T} \left(n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \right), & n_{td} = \sum_{w \in W} n_{dw} p_{tdw} \end{cases} \end{cases}$$

Модель PLSA: $R(\Phi, \Theta) = 0$

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

ARTM легко обобщается на мультимодальные задачи

Выявление тематики документов $p(t|d)$ и терминов $p(t|w)$, а также модальностей: $p(t|автор)$, $p(t|время)$, $p(t|ссылка)$, $p(t|баннер)$, $p(t|элемент изображения)$, $p(t|пользователь)$,...



Мультимодальная ARTM [Vorontsov et al, 2015]

W^m — словарь токенов m -й модальности, $m \in M$

$W = W^1 \sqcup \dots \sqcup W^M$ — объединённый словарь всех модальностей

Максимизация суммы \log правдоподобий с регуляризацией:

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

EM-алгоритм: метод простой итерации для системы уравнений

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

BigARTM: библиотека тематического моделирования

Ключевые возможности:

- Онлайн-параллельный мультимодальный ARTM
- Большие данные: коллекция не хранится в памяти
- Встроенная библиотека регуляризаторов и мер качества

Сообщество:

- Открытый код <https://github.com/bigartm>
(discussion group, issue tracker, pull requests)
- Документация <http://bigartm.org>



Лицензия и среда разработки:

- 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

BigARTM: унификация разработки тематических моделей

Этапы моделирования

Bayesian TM

ARTM

| | Bayesian TM | ARTM |
|-----------------|--|--|
| | Анализ требований | Анализ требований |
| Формализация: | Вероятностная порождающая модель данных | Стандартные критерии Свои критерии |
| Алгоритмизация: | Байесовский вывод для данной порождающей модели (VI, GS, EP) | Общий регуляризованный EM-алгоритм для любых моделей |
| Реализация: | Исследовательский код (Matlab, Python, R) | Промышленный код BigARTM (C++, Python API) |
| Оценивание: | Исследовательские метрики, исследовательский код | Стандартные метрики Свои метрики |
| | Внедрение | Внедрение |

-- нестандартизируемые этапы, уникальная разработка для каждой задачи

-- стандартизируемые этапы

Разработка тематических моделей в среде IPython Notebook

<http://nbviewer.ipython.org/github/bigartm/bigartm-book/tree/master/>**Коллекция:**

Используем небольшую коллекцию 'kos', доступную в репозитории UCI
<https://archive.ics.uci.edu/ml/machine-learning-databases/bag-of-words/>. Параметры коллекции следующие:

- 3430 документов;
- 6906 слов в словаре;
- 46714 слов в коллекции.

Для начала подключим все необходимые модули (убедитесь, что путь к Python API BigARTM находится в вашей переменной PATH):

```
In [1]: %matplotlib inline
import glob
import matplotlib.pyplot as plt
import artm
```

Прежде всего необходимо подготовить входные данные. BigARTM имеет собственный формат документов для обработки, называемый батчами. В библиотеке присутствуют средства по созданию батчей из файлов Bag-Of-Words в форматах UCI и Wowpal Wabbit (подробности можно найти в <http://docs.bigartm.org/en/latest/formats.html>).

В Python API, по аналогии с алгоритмами из scikit-learn, входные данные представлены одним классом BatchVectorizer. Объект этого класса принимает на вход батчи или файлы с Bag-Of-Words и подается на вход всем методам. В случае, если входные данные не являются батчами, он создаст их и сохранит на диск для последующего быстрого использования.

Итак, создадим объект BatchVectorizer:

```
In [2]: batch_vectorizer = None
if len(glob.glob('kos' + '/*.*.batch')) < 1:
    batch_vectorizer = artm.BatchVectorizer(data_path='', data_format='bow
_uci', collection_name='kos', target_folder='kos')
else:
    batch_vectorizer = artm.BatchVectorizer(data_path='kos', data_format='
batches')
```

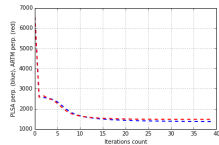
ARTM — это класс, представляющий собой Python API BigARTM, и позволяющий использовать практически все возможности библиотеки в стиле scikit-learn. Создадим две тематические модели для нашего эксперимента. Наиболее важным параметром модели является число тем. Опционально можно указать список регуляризаторов и функционалов качества, которые следует использовать для данной модели. Если этого не сделать, то регуляризаторы и функционалы всегда можно добавить позднее. Обратите внимание, что каждая модель задаёт

Продолжим обучение моделей, инициализировав 25 проходов по коллекции, после чего снова посмотрим на значения функционалов качества:

```
In [11]: model_plsa.fit_offline(batch_vectorizer=batch_vectorizer, num_collection_p
asses=25, num_document_passes=1)
model_artm.fit_offline(batch_vectorizer=batch_vectorizer, num_collection_p
asses=25, num_document_passes=1)
```

```
In [12]: print_measures(model_plsa, model_artm)
```

Sparsity Phi: 0.332 (FLSA) vs. 0.740 (ARTM)
Sparsity Theta: 0.082 (FLSA) vs. 0.602 (ARTM)
Kernel contrast: 0.530 (FLSA) vs. 0.548 (ARTM)
Kernel purity: 0.396 (FLSA) vs. 0.531 (ARTM)
Perplexity: 1362.804 (FLSA) vs. 1475.455 (ARTM)



Кроме того, для наглядности построим графики изменения разреженностей матриц по итерациям:

```
In [13]: plt.plot(xrange(model_plsa.num_phi_updates), model_plsa.score_tracker['Spa
rsityPhiScore'].value, 'b--',
                xrange(model_artm.num_phi_updates), model_artm.score_trac
ker['SparsityPhiScore'].value, 'r--', linewidth=2)
plt.xlabel('Iterations count')
plt.ylabel('FLSA Phi sp. (blue), ARTM Phi sp. (red)')
plt.grid(True)
plt.show()
```

```
plt.plot(xrange(model_plsa.num_phi_updates), model_plsa.score_tracker['Spa
rsityThetaScore'].value, 'b--',
                xrange(model_artm.num_phi_updates), model_artm.score_trac
ker['SparsityThetaScore'].value, 'r--', linewidth=2)
```

Тесты производительности

- 3.7M статей английской Вики, 100K уникальных слов

| | 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* = число параллельных потоков
- *inference* = время тематизации 100K тестовых документов
- *perplexity* вычислена на тестовой выборке документов

Данные коллективного блога Хабрахабр.ру

Данные

- 132 157 статей
- Модальности:
 - 52 354 терминов (слов)
 - 524 авторов статей
 - 10 000 комментаторов (авторов комментариев к статьям)
 - 2546 тегов
 - 123 хаба (категории)

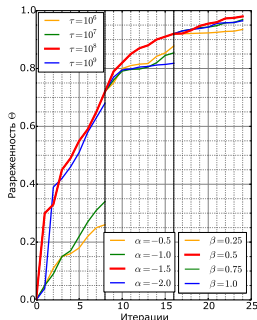
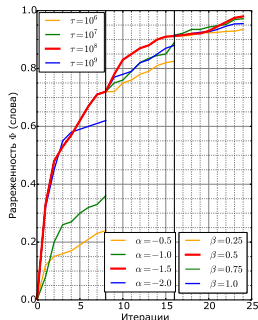
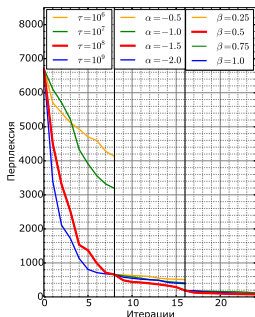
Предобработка текстов

- отброшены 5% наиболее частотных слов (общая лексика)
- удаление пунктуации
- нижний регистр, ё→е
- лемматизация rymorphy2

Подбор коэффициентов регуляризации

Последовательное добавление регуляризаторов:

- декоррелирование распределений терминов в темах (τ),
- разреживание распределений тем в документах (α),
- сглаживание распределений терминов в темах (β).



Разведочный поиск

$q = (w_1, \dots, w_{n_q})$ — текст запроса произвольной длины n_q

$\theta_{tq} = p(t|q)$ — тематический профиль запроса q

$\theta_{td} = p(t|d)$ — тематические профили документов $d \in D$

Косинусная мера близости документа d и запроса q :

$$\text{sim}(q, d) = \frac{\sum_t \theta_{tq} \theta_{td}}{(\sum_t \theta_{tq}^2)^{1/2} (\sum_t \theta_{td}^2)^{1/2}}.$$

Ранжируем документы коллекции $d \in D$ по убыванию $\text{sim}(q, d)$

Выдача тематического поиска — k первых документов.

Реализация: *инвертированный индекс* для быстрого поиска документов d по каждой из тем t запроса

Методика оценивания качества разведочного поиска

Поисковый запрос

набор ключевых слов или фрагментов текста, около одной страницы A4

Поисковая выдача

документы d с распределением $p(t|d)$, близким к распределению $p(t|q)$ запроса

Два задания асессорам

- 1 найти как можно больше статей, пользуясь любыми средствами поиска (и засечь время)
- 2 оценить релевантность поисковой выдачи на том же запросе

Поиск MapReduce

Поиск MapReduce – программа поиска (библиотека) написанная распределенным способом для больших объемов данных в рамках параллельных шардов, представляющих собой набор Java-классов и исполняемых единиц для создания и обработки данных на параллельной обработке.

Основные приложения **Поиск MapReduce** можно сформулировать как:

- обработка написанных большим объемом данных;
- масштабируемость;
- автоматическое распределение данных;
- работа на неопределенном оборудовании;
- автоматическая обработка отлаженных написанных данных.

Поиск – популярная программная платформа (**язык Java, библиотека**) построена распределенными приложениями для массово-параллельной обработки (**разные работы, ресурсы, CPU**) данных.

Поиск включает в себе следующие компоненты:

1. HDFS – распределенная файловая система;
2. **Поиск MapReduce** – программная модель (**библиотека**) написанная распределенным способом для больших объемов данных в рамках параллельных шардов.

Клиенты, приложения в архитектуре **Поиск MapReduce** и структура HDFS, стали привычной реальностью не только в области компьютеров, а в том числе и в бизнесе: точка отсчета "Zero", в конечном итоге, определенно ограниченная платформой **Поиск** в целом. К сожалению, можно отметить:

Ограничение масштабируемости кластера **Поиск** –4K вычислительных узлов, –40K параллельных заданий.

Сильная связность **Фреймворка** распределенных вычислений и клиентских приложений, реализованных распределенным способом. Как следствие:

Отсутствие поддержки альтернативной программной модели написанных распределенных вычислений: в **Поиск v1.0** поддерживается только модель написанных шардов.

Многие «важные» точки отсчета в виде сложности, неопределенности масштабов и средств с высшими требованиями к надежности.

Проблема **взаимосвязи** совместности требований по единичному объекту: все вычислительные узлы кластера при обращении платформ **Поиск** (установка новой версии или пакета обновлений).

Пример запроса для разведочного поиска

Пример: фрагмент запроса «Система IBM Watson»

IBM Watson — суперкомпьютер фирмы IBM, оснащённый вопросно-ответной системой искусственного интеллекта, созданный группой исследователей под руководством Дэвида Феруччи. Его создание — часть проекта DeepQA. Основная задача Уотсона — понимать вопросы, сформулированные на естественном языке, и находить на них ответы в базе данных. Назван в честь основателя IBM Томаса Уотсона.

IBM Watson представляет собой когнитивную систему, которая способна понимать, делать выводы и обучаться. Она также позволяет преобразовывать целые отрасли, различные направления науки и техники. Например, предсказывать появление эпидемий или возникновения очагов природных катастроф в различных регионах, вести мониторинг состояния атмосферы больших городов, оптимизировать бизнес-процессы, узнавать, какие товары будут в тренде в ближайшее время.

... ..

Релевантные тексты: примеры сервисов и приложений, основа которых — когнитивная платформа IBM Watson, используемые в IBM Watson технологии, вопрос-ответные системы, сопоставление IBM Watson с Wolfram-Alpha.

Нерелевантные тексты: общие вопросы искусственного интеллекта, другие коммерческие решения на рынке бизнес-аналитики.

Тематика запросов разведочного поиска

Примеры заголовков разведочных запросов к Хабру
(объём каждого запроса — около одной страницы A4):

| | |
|-----------------------------------|---------------------------|
| Алгоритмы раскраски графов | Система IBM Watson |
| Рекомендательная система Netflix | 3D-принтеры |
| Методики быстрого набора текста | CERN-кластер |
| Космические проекты Илона Маска | АВ-тестирование |
| Технологии Hadoop MapReduce | Облачные сервисы |
| Беспилотный автомобиль Google car | Контекстная реклама |
| Криптосистемы с открытым ключом | Марсоход Curiosity |
| Обзор платформ онлайн-курсов | Видеокарты NVIDIA |
| Data Science Meetups в Москве | Распознавание образов |
| Образовательные проекты mail.ru | Сервисы Google scholar |
| Межпланетная станция New horizons | MIT MediaLab Research |
| Языковая модель word2vec | Платформа Microsoft Azure |

Оценки качества поиска

Precision — доля релевантных среди найденных

Recall — доля найденных среди релевантных

$$P = \frac{TP}{TP + FP} \text{ — точность (precision)}$$

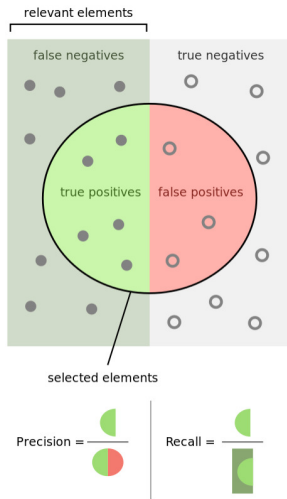
$$R = \frac{TP}{TP + FN} \text{ — полнота, (recall)}$$

$$F_1 = \frac{P + R}{2PR} \text{ — F1-мера}$$

TP (true positive) — найденные релевантные

FP (false positive) — найденные нерелевантные

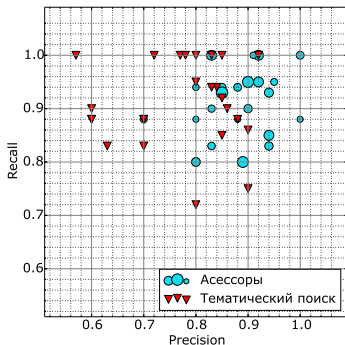
FN (false negative) — ненайденные релевантные



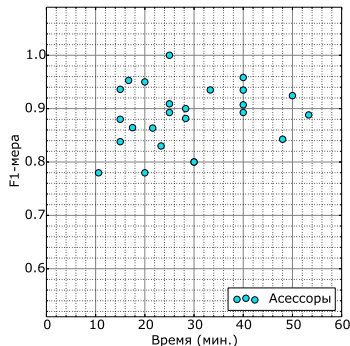
Результаты измерения точности и полноты по запросам

25 запросов, 3 ассессора на запрос

точность и полнота поиска



время и F_1 -мера (ассессоры)



- среднее время обработки запроса ассессором — 30 минут
- точность выше у ассессоров, полнота — у поисковика

Выбор модальностей по критериям точности и полноты

Модальности: Слова, Авторы, Комментаторы, Теги, Хабы.
Число тем $|T| = 200$.

| | ассессоры | С | К | ТХ | СТ | СХ | СТХ | все |
|--------------|-----------|------|------|------|------|------|------|-------------|
| Precision@5 | 0.82 | 0.63 | 0.54 | 0.59 | 0.74 | 0.73 | 0.73 | 0.74 |
| Precision@10 | 0.87 | 0.67 | 0.56 | 0.58 | 0.77 | 0.74 | 0.75 | 0.77 |
| Precision@15 | 0.86 | 0.65 | 0.53 | 0.55 | 0.67 | 0.67 | 0.68 | 0.68 |
| Precision@20 | 0.85 | 0.64 | 0.53 | 0.54 | 0.66 | 0.67 | 0.68 | 0.68 |
| Recall@5 | 0.78 | 0.77 | 0.63 | 0.69 | 0.82 | 0.81 | 0.82 | 0.82 |
| Recall@10 | 0.84 | 0.79 | 0.64 | 0.71 | 0.88 | 0.82 | 0.87 | 0.88 |
| Recall@15 | 0.88 | 0.82 | 0.67 | 0.73 | 0.90 | 0.84 | 0.89 | 0.90 |
| Recall@20 | 0.88 | 0.85 | 0.68 | 0.74 | 0.91 | 0.85 | 0.89 | 0.91 |

- Наилучшее качество поиска — по всем модальностям
- Наиболее полезные модальности — термины и теги

Выбор числа тем по критериям точности и полноты

Теперь используем все 5 модальностей, меняем число тем | T |

| | асессоры | 100 | 200 | 300 | 400 | 500 |
|--------------|----------|------|-------------|------|------|------|
| Precision@5 | 0.82 | 0.61 | 0.74 | 0.71 | 0.69 | 0.59 |
| Precision@10 | 0.87 | 0.65 | 0.77 | 0.72 | 0.67 | 0.61 |
| Precision@15 | 0.86 | 0.67 | 0.68 | 0.67 | 0.65 | 0.62 |
| Precision@20 | 0.85 | 0.64 | 0.68 | 0.67 | 0.64 | 0.60 |
| Recall@5 | 0.78 | 0.62 | 0.82 | 0.80 | 0.72 | 0.63 |
| Recall@10 | 0.84 | 0.63 | 0.88 | 0.81 | 0.75 | 0.64 |
| Recall@15 | 0.88 | 0.67 | 0.90 | 0.82 | 0.77 | 0.67 |
| Recall@20 | 0.88 | 0.69 | 0.91 | 0.85 | 0.77 | 0.68 |

- Наилучшее качество поиска — при 200 темах
- Тематический поиск превосходит асессоров по полноте

Поиск этно-релевантных тем в социальных сетях

Основные задачи проекта:

- Разведочный поиск этнических тем в социальных медиа (сколько различных тем, и что это за темы)
- Мониторинг этих тем во времени и по регионам
- Сентимент-анализ и оценивание конфликтности

Вспомогательные задачи:

- Фильтрация (обогащение) потока данных
- Обеспечение полноты поиска этнических тем
- Выявление тематических сообществ
- Выделение событийных и региональных тем
- Решение проблемы коротких сообщений

Примеры этнонимов

| | |
|---------------------|------------|
| 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 |

Примеры этнических тем

(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, уничтожить, группировка, дамаск, оружие, алесию, 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, уничтожить, сирия, подразделение, квартал, армейский,

(ливийцы): ливан, демократия, страна, ливийский, каддафи, государство, алжир, война, правительство, сша, арабский, али, муаммар, сирия,

Примеры этнических тем

(евреи): израиль, израильский, страна, война, нетаньяху, тель-авив, время, сша, сирия, египет, случай, самолет, еврейский, военный, ближний,

(американцы): американский, американка, война, россия, военный, страна, вашингтон, америка, армия, конгресс, сирия, союзный, российский, обама, войска, русский, оружие, операция,

(немцы): армия, война, войска, советский, военный, дивизия, немец, фронт, немецкий, генерал, борт, операция, оборона, русский, бог, победа,

(немцы): германий, немец, германский, ссср, немецкий, война, старое, советский, россия, береза, русский, правительство, территория, полный, документ, вопрос, сорт, договор, отношение, франция,

(евреи, немцы): еврей, еврейский, холодный, германий, антисемитизм, гетра, немец, синагога, сша, израиль, малиновского, комиссия, нацбол, документ, война, еврейка, миллион, украина,

(украинцы, немцы): украинский, унс, оун, немец, немецкий, ковальков, хохол, волынский, бандера, организация, россиянин, советский, русский, польский, армия, шухевича, ровенский,

(таджики, узбеки): мигрант, страна, россия, миграция, азия, нелегальный, миграционный, таджикистан, гастарбайтер, гражданка, трудовой, рабочий, фмс, коренево, среднее, узбекистан, таджик, проблема, русский, население,

(канадцы): команда, игра, игрок, канадский, сезон, хоккей, сборная, играть, болельщик, победа, кубок, счет, забирать, хоккейный, выигрывать, хоккеист, чемпионат, шайба,

Примеры этнических тем

(японцы): японский, япония, корея, китайский, жилища, авария, фукусиму, цунами, сообщать, океан, станция, хатико, район, правительство, атомный,

(норвежцы): дитя, ребенок, родиться, детский, семья, воспитанный, право, возраст, отец, воспитание, норвежский, родительский, родить, мальчик, взрослый, опека, сын,

(венесуэльцы): куба, кастро, венесуэла, чавес, президент, уго, мадура, боливия, фидель, глава, латинский, венесуэльский, лидер, боливарианской, президентский, альенде, гевару,

(китайцы): китайский, россия, производство, китаи, продукция, страна, предприятие, компания, технология, военный, регион, производить, производственный, промышленность, российский, экономический, кнр,

(азербайджанцы): русский, азербайджан, азербайджанец, россия, азербайджанский, таксист, диаспора, анапа, народ, москва, страна, армянин, слово, рынок,

(грузины): грузинский, спецназ, военный, август, баташева, российский, спецназовец, миротворец, операция, румын, бригада, миротворческий, абхазия, группа, войска, русский, цхинвале,

(осетины): конституция, осетия, аминат, русский, осетинский, южный, северный, россия, война, республика, вопрос, алахай, российский, население, конфликт,

(цыгане): наркотик, цыган, цыганка, хороший, место, страна, деньга, время, работать, жизнь, жить, рука, дом, цыганский, наркоманка,

Результаты: модель ARTM находит намного больше этно-тем

Число этно-релевантных тем, найденных моделью:

| модель | этно-тем | фон.тем | ++ | +- | -+ | всего |
|--------|----------|---------|-----------|-----------|-----------|------------|
| PLSA | 300 | | 9 | 11 | 18 | 38 |
| PLSA | 400 | | 12 | 15 | 17 | 44 |
| ARTM-1 | 200 | 100 | 18 | 33 | 20 | 71 |
| ARTM-1 | 250 | 150 | 21 | 27 | 20 | 68 |
| ARTM-2 | 200 | 100 | 28 | 23 | 23 | 74 |
| ARTM-2 | 250 | 150 | 38 | 42 | 30 | 104 |

Регуляризаторы ARTM-1:

этно темы: разреживание, декоррелирование, сглаживание этнонимов
фоновые темы: сглаживание, разреживание этнонимов

Регуляризаторы ARTM-2:

ARTM-1 + **модальность этнонимов**

Биграммы радикально улучшают интерпретируемость тем

Коллекция 1000 статей конференций ММРО, ИОИ на русском

| распознавание образов в биоинформатике | | теория вычислительной сложности | |
|--|-------------------------|---------------------------------|----------------------|
| unigrams | bigrams | unigrams | bigrams |
| объект | задача распознавания | задача | разделять множества |
| задача | множество мотивов | множество | конечное множество |
| множество | система масок | подмножество | условие задачи |
| мотив | вторичная структура | условие | задача о покрытии |
| разрешимость | структура белка | класс | покрытие множества |
| выборка | распознавание вторичной | решение | сильный смысл |
| маска | состояние объекта | конечный | разделяющий комитет |
| распознавание | обучающая выборка | число | минимальный аффинный |
| информативность | оценка информативности | аффинный | аффинный комитет |
| состояние | множество объектов | случай | аффинный разделяющий |
| закономерность | разрешимость задачи | покрытие | общее положение |
| система | критерий разрешимости | общий | множество точек |
| структура | информативность мотива | пространство | случай задачи |
| значение | первичная структура | схема | общий случай |
| регулярность | тупиковое множество | комитет | задача MASC |

Мультиязычная модель Википедии

216 175 русско-английских пар статей Вики.
 Первые 10 слов и их вероятности $p(w|t)$ в %:

| Topic 68 | | | | Topic 79 | | | |
|-------------|------|--------------|------|----------|------|-----------|------|
| research | 4.56 | институт | 6.03 | goals | 4.48 | матч | 6.02 |
| technology | 3.14 | университет | 3.35 | league | 3.99 | игрок | 5.56 |
| engineering | 2.63 | программа | 3.17 | club | 3.76 | сборная | 4.51 |
| institute | 2.37 | учебный | 2.75 | season | 3.49 | фк | 3.25 |
| science | 1.97 | технический | 2.70 | scored | 2.72 | против | 3.20 |
| program | 1.60 | технология | 2.30 | cup | 2.57 | клуб | 3.14 |
| education | 1.44 | научный | 1.76 | goal | 2.48 | футболист | 2.67 |
| campus | 1.43 | исследование | 1.67 | apps | 1.74 | гол | 2.65 |
| management | 1.38 | наука | 1.64 | debut | 1.69 | забивать | 2.53 |
| programs | 1.36 | образование | 1.47 | match | 1.67 | команда | 2.14 |

Дударенко М. А. Регуляризация многоязычных тематических моделей.
 Вычислительные методы и программирование. 2015. Т. 16. С. 26–36.

Мультиязычная модель Википедии

216 175 русско-английских пар статей Вики.
 Первые 10 слов и их вероятности $p(w|t)$ в %:

| Topic 88 | | | | Topic 251 | | | |
|-------------|------|---------|------|------------|------|--------------|------|
| opera | 7.36 | опера | 7.82 | windows | 8.00 | windows | 6.05 |
| conductor | 1.69 | оперный | 3.13 | microsoft | 4.03 | microsoft | 3.76 |
| orchestra | 1.14 | дирижер | 2.82 | server | 2.93 | версия | 1.86 |
| wagner | 0.97 | певец | 1.65 | software | 1.38 | приложение | 1.86 |
| soprano | 0.78 | певица | 1.51 | user | 1.03 | сервер | 1.63 |
| performance | 0.78 | театр | 1.14 | security | 0.92 | server | 1.54 |
| mozart | 0.74 | партия | 1.05 | mitchell | 0.82 | программный | 1.08 |
| sang | 0.70 | сопрано | 0.97 | oracle | 0.82 | пользователь | 1.04 |
| singing | 0.69 | вагнер | 0.90 | enterprise | 0.78 | обеспечение | 1.02 |
| operas | 0.68 | оркестр | 0.82 | users | 0.78 | система | 0.96 |

Независимый ассессор оценил 396 тем из $|T| = 400$ как хорошо интерпретируемые.

- Для разведочного поиска нужны комбинации моделей
- Мы комбинируем модели, суммируя регуляризаторы
- ARTM: единый EM-алгоритм для любых регуляризаторов
- Библиотека с открытым кодом BigARTM реализует онлайн-параллельный регуляризованный EM-алгоритм
- Комбинация регуляризаторов подбирается в эксперименте



<http://bigartm.org>

-  *K. Vorontsov*. Additive regularization for topic models of text collections. 2014.
-  *K. Vorontsov, A. Potapenko*. Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization. AIST 2014.
-  *K. Vorontsov, A. Potapenko*. Additive regularization of topic models. Machine Learning, 2015.
-  *K. Vorontsov, O. Frei, M. Apishev, P. Romov, M. Suvorova, A. Yanina*. Non-bayesian additive regularization for multimodal topic modeling of large collections. 2015.
-  *K. Vorontsov, A. Potapenko, A. Plavin*. Additive regularization of topic models for topic selection and sparse factorization. SLDS 2015.
-  *O. Frei, M. Apishev*. Parallel non-blocking deterministic algorithm for online topic modeling. AIST 2016. (в печати)
-  *M. Apishev, S. Koltcov, O. Koltsova, S. Nikolenko, K. Vorontsov*. Additive regularization for topic modeling in sociological studies of user-generated text content. MICAI 2016. (в печати)
-  *А.О.Янина, К.В.Воронцов*. Мультимодальные тематические модели для разведочного поиска в коллективном блоге. ИОИ 2016. (в печати)