

# Тематическое моделирование.

Мурат Апишев

НИУ ВШЭ, МГУ им. Ломоносова, Яндекс, ШАД

17 октября, 2017

# Тематическое моделирование

**Тематическое моделирование** (*Topic Modeling*) — приложение машинного обучения к статистическому анализу текстов.

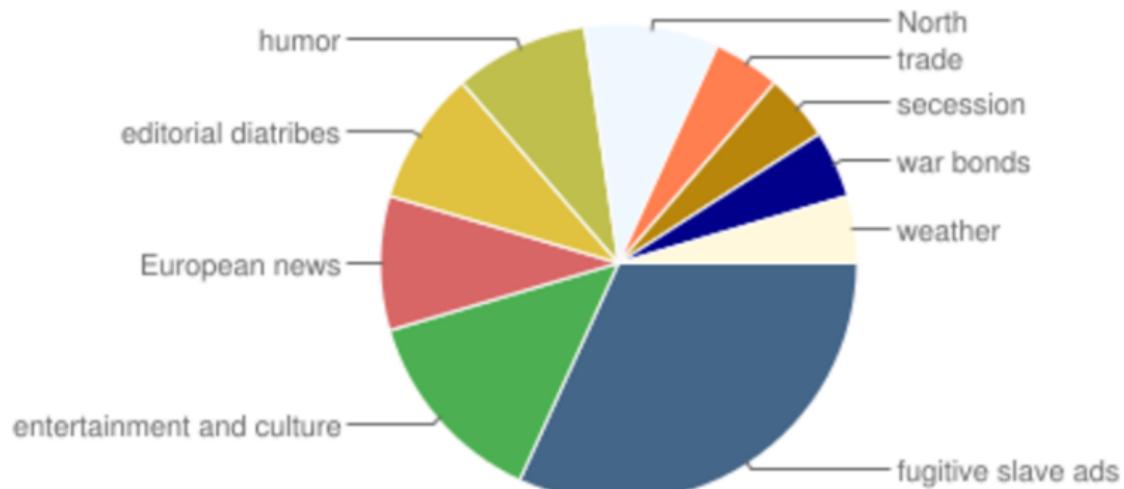
**Тема** — терминология предметной области, набор терминов (униграм или  $n$ -грамм) часто встречающихся вместе в документах.

Тематическая модель исследует скрытую тематическую структуру коллекции текстов:

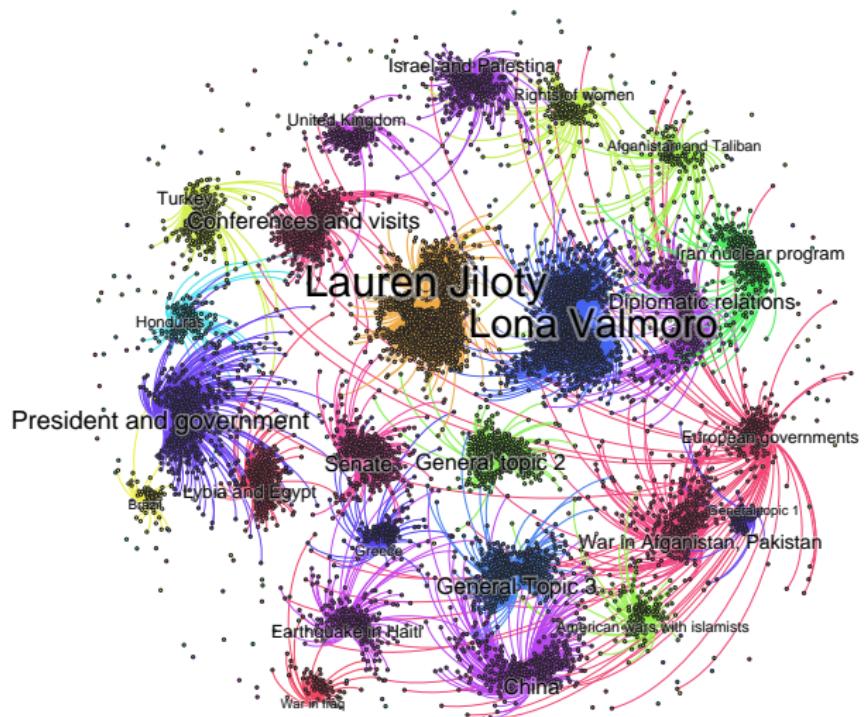
- ▶ тема  $t$  — это вероятностное распределение  $p(w|t)$  над терминами  $w$
- ▶ документ  $d$  — это вероятностное распределение  $p(t|d)$  над темами  $t$

Нестрого говоря, тема — это набор слов, глядя на которые можно сказать, какую предметную область они описывают.

# Определение тем и их соотношений



# Кластеризация и классификация документов



# Приложения тематического моделирования

разведочный поиск в электронных библиотеках



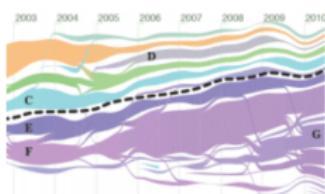
персонализированный поиск в соцсетях



мультимодальный поиск текстов и изображений



детектирование и трекинг новостных сюжетов



навигация по большим текстовым коллекциям



управлением диалогом в разговорном интеллекте



# Приложения тематического моделирования

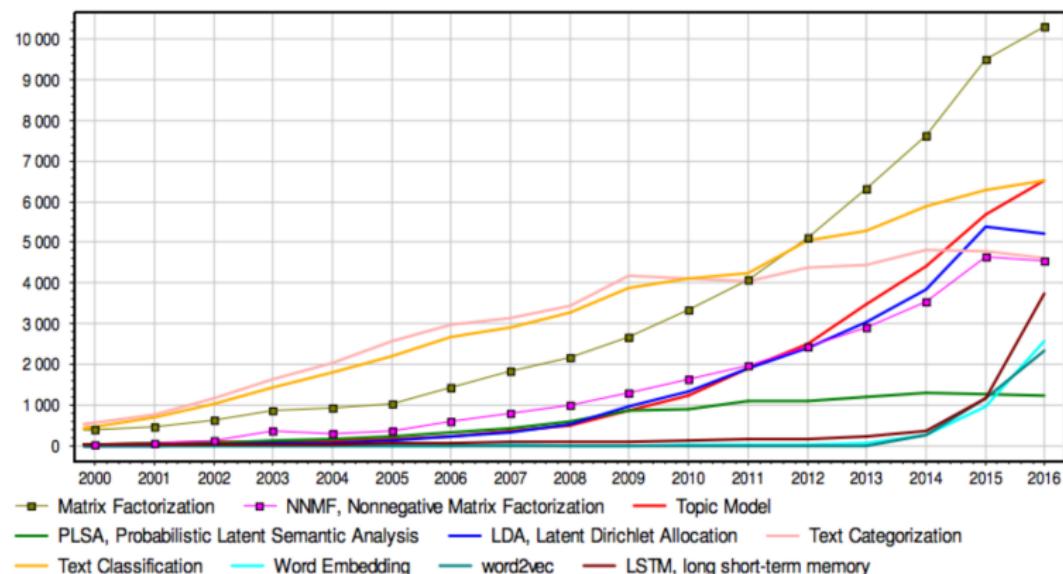
- ▶ Тематический поиск документов по тексту любой длины, или по любому объекту
- ▶ Поиск научных статей, экспертов, рецензентов, проектов
- ▶ Выявление трендов и фронтов исследования
- ▶ Суммаризация и аннотирование текстовых документов
- ▶ Анализ и агрегирование новостных потоков
- ▶ Рубрикация документов, видео, музыки
- ▶ Различные задачи биоинформатики

## Требования к тематической модели

- ▶ Интерпретируемость выделяемых тем
- ▶ Обработка больших объёмов данных

# ТМ и смежные области исследований

Динамика цитирования, по данным Google Scholar:



## Мешок слов

**Мешок слов** (Bag-Of-Words) — представление текстовых данных, в котором учитывается только частота встречаемости слов в документах. Порядок слов игнорируется.

**Исходное предложение:** I can drink a milk can

**Его мешок слов:**

I: 1

can: 2

drink: 1

a: 1

milk: 1

Проще, но теряется много полезной информации.

# Основные предположения

- ▶ Порядок терминов в документе не важен (bag of words)
- ▶ Порядок документов в коллекции не важен (bag of docs)
- ▶ Каждый термин в документе связан с некоторой темой  $t \in T$
- ▶  $D \times W \times T$  — дискретное вероятностное пространство
- ▶ Коллекция — это i.i.d. выборка  $(d_i, w_i, t_i)_{i=1}^n \sim p(d, w, t)$
- ▶  $d_i, w_i$  — наблюдаемые, темы  $t_i$  — скрытые
- ▶ гипотеза условной независимости:  $p(w|d, t) = p(w|t)$

## Предварительная обработка текста:

- ▶ Лемматизация (русский) или стемминг (английский)
- ▶ Выделение терминов (term extraction)
- ▶ Удаление стоп-слов и слишком редких слов

# PLSA

## Probabilistic Latent Semantic Analysis:

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

	doc_1	doc_2	doc_3	doc_4	doc_5
word_1					
word_2					
word_3					
word_4					
word_5					
word_6					
word_7					
word_8					

=

	topic_1	topic_2	topic_3
word_1			
word_2			
word_3			
word_4			
word_5			
word_6			
word_7			
word_8			

X

	doc_1	doc_2	doc_3	doc_4	doc_5
topic_1					
topic_2					
topic_3					

$$\theta = p(t|d)$$

$$F = p(w|d)$$

$$\Phi = p(w|t)$$

# PLSA

**Дано:**  $W$  — словарь терминов (униграм или  $n$ -биграм),  
 $D$  — коллекция текстовых документов  $d \subset W$ ,  
 $n_{dw}$  — счётчик частоты появления слова  $w$  в документе  $d$ .

**Найти:** модель  $p(w|d) = \sum_{t \in T} \phi_{wt} \theta_{td}$  с параметрами  $\Phi_{w \times T}$  и  $\Theta_{T \times D}$ :  
 $\phi_{wt} = p(w|t)$  — вероятности терминов  $w$  в каждой теме  $t$ ,  
 $\theta_{td} = p(t|d)$  — вероятности тем  $t$  в каждом документе  $d$ .

**Критерий максимизация логарифма правдоподобия:**

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

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

# Интуиция EM-алгоритма

Есть текст

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

**Хотим оценить вероятности слов в темах  $p(w|t)$  и тем в документах  $p(t|d)$ .**

# Интуиция EM-алгоритма

Если бы у нас были присваивания слов темам...

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

...мы могли бы просто посчитать:

$$p(w = \text{sky} | t) = \frac{n_{w|t}}{\sum_w n_{w|t}} = \frac{1}{4} \quad p(t = \text{sky} | d) = \frac{n_{t|d}}{\sum_t n_{t|d}} = \frac{4}{54}$$

## Но у нас есть только текст

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

## Но у нас есть только текст

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

**Идея: попробуем оценить присваивания тем:**

$$\begin{aligned} p(t|d, w) &= \{\text{cond.rule}\} = \\ &= \frac{p(w, t|d)}{p(w|d)} = \{\text{indep. + prod.rule}\} = \\ &= \frac{p(w|t)p(t|d)}{p(w|d)} \end{aligned}$$

# Собираем всё вместе: PLSA

Максимизация логарифма правдоподобия:

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

**EM-алгоритм:** метод простых итераций для решения системы уравнений

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

$$\text{где } \text{norm}_{i \in I} x_i = \frac{\max\{x_i, 0\}}{\sum_{j \in I} \max\{x_j, 0\}}$$

# PLSA и перплексия

Величина, характеризующая степень сходимости модели с заданным словарём  $W$  — *перплексия*:

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

Она построена на основе логарифма правдоподобия и характеризует степень качества описания коллекции моделью. Чем ниже — тем лучше. Алгоритм оптимизирует именно её.

**Но!** Матричное разложение  $F \approx \Phi \times \Theta$  имеет бесконечное множество решений:  $\Phi\Theta = (\Phi S)(S^{-1}\Theta) = \Phi'\Theta' \Rightarrow$  можно подобрать  $\Phi$  и  $\Theta$  подходящего вида.

Существуют различные прикладные метрики качества. Они не оптимизируются моделью напрямую, но их значения хочется повышать. **Выход — регуляризация!**

# Пример регуляризации

Логичные предположения:

- ▶ темы должны состоять из небольшого числа слов, и эти множества слов не должны сильно пересекаться;
- ▶ каждый документ должен относиться к небольшому числу тем.

$$\begin{array}{c} \text{doc\_1} \\ \text{doc\_2} \\ \text{doc\_3} \\ \text{doc\_4} \\ \text{doc\_5} \end{array} \quad \begin{array}{c} \text{word\_1} \\ \text{word\_2} \\ \text{word\_3} \\ \text{word\_4} \\ \text{word\_5} \\ \text{word\_6} \\ \text{word\_7} \\ \text{word\_8} \end{array} = \begin{array}{c} \text{topic\_1} \\ \text{topic\_2} \\ \text{topic\_3} \end{array} \quad \begin{array}{c} \text{word\_1} \\ \text{word\_2} \\ \text{word\_3} \\ \text{word\_4} \\ \text{word\_5} \\ \text{word\_6} \\ \text{word\_7} \\ \text{word\_8} \end{array} \quad \begin{array}{c} \text{topic\_1} \\ \text{topic\_2} \\ \text{topic\_3} \end{array} \quad \begin{array}{c} \text{doc\_1} \\ \text{doc\_2} \\ \text{doc\_3} \\ \text{doc\_4} \\ \text{doc\_5} \end{array}$$

$\times$

$$F = p(w|d) \qquad \Phi = p(w|t) \qquad \theta = p(t|d)$$

# Пример регуляризации

Извлечение специфичной тематики по ключевым словам:

- ▶ хотим собрать темы около интересующих слов, а документы — около интересующих тем;
- ▶ прочие темы хотим сглаживать по неважным словам, чтобы собрать «мусор».

$$\begin{array}{c} \text{doc\_1} \\ \text{doc\_2} \\ \text{doc\_3} \\ \text{doc\_4} \\ \text{doc\_5} \\ \hline \text{word\_1} \\ \text{word\_2} \\ \text{word\_3} \\ \text{word\_4} \\ \text{word\_5} \\ \text{word\_6} \\ \text{word\_7} \\ \text{word\_8} \end{array} = \begin{array}{c} \text{topic\_1} \\ \text{topic\_2} \\ \text{topic\_3} \\ \hline \text{word\_1} \\ \text{word\_2} \\ \text{word\_3} \\ \text{word\_4} \\ \text{word\_5} \\ \text{word\_6} \\ \text{word\_7} \\ \text{word\_8} \end{array} \times \begin{array}{c} \text{topic\_1} \\ \text{topic\_2} \\ \text{topic\_3} \\ \hline \text{doc\_1} \\ \text{doc\_2} \\ \text{doc\_3} \\ \text{doc\_4} \\ \text{doc\_5} \end{array}$$
$$F = p(w|d) \quad \Phi = p(w|t) \quad \Theta = p(t|d)$$

# ARTM

## Additive Regularization of Topic Models:

Максимизация логарифма правдоподобия с дополнительными аддитивными регуляризаторами  $R$ :

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

**EM-алгоритм:** метод простых итераций для системы уравнений

E-шаг:  $p_{tdw} = \text{norm}_{t \in T}(\phi_{wt} \theta_{td})$

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

# Условия Каруша–Куна–Таккера

Задача математического программирования:

$$\begin{cases} f(x) \rightarrow \min_x; \\ g_i(x) \leq 0, \quad i = 1, \dots, m; \\ h_j(x) = 0, \quad j = 1, \dots, k. \end{cases}$$

Необходимые условия. Если  $x$  — точка локального минимума, то существуют множители  $\mu_i$ ,  $i = 1, \dots, m$ ,  $\lambda_j$ ,  $j = 1, \dots, k$ :

$$\begin{cases} \frac{\partial}{\partial x} = 0, \quad (x; \mu, \lambda) = f(x) + \sum_{i=1}^m \mu_i g_i(x) + \sum_{j=1}^k \lambda_j h_j(x); \\ g_i(x) \leq 0; \quad h_j(x) = 0; \quad (\text{исходные ограничения}) \\ \mu_i \geq 0; \quad (\text{двойственные ограничения}) \\ \mu_i g_i(x) = 0; \quad (\text{условие дополняющей нежёсткости}) \end{cases}$$

# Вывод системы уравнений из условий Каруша–Куна–Таккера

1. Условия ККТ для  $\phi_{wt}$  (для  $\theta_{td}$  всё аналогично):

$$\sum_d n_{dw} \frac{\theta_{td}}{p(w|d)} + \frac{\partial R}{\partial \phi_{wt}} = \lambda_t - \mu_{wt}; \quad \mu_{wt} \geq 0; \quad \mu_{wt} \phi_{wt} = 0.$$

2. Умножим обе части равенства на  $\phi_{wt}$  и выделим  $p_{tdw}$ :

$$\phi_{wt} \lambda_t = \sum_d n_{dw} \frac{\phi_{wt} \theta_{td}}{p(w|d)} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} = n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}}.$$

3. Если  $\lambda_t \leq 0$ , то тема  $t$  вырождена,  $\phi_{wt} \equiv 0$  для всех  $w$ .
4. Если  $\lambda_t > 0$ , то либо  $\phi_{wt} = 0$ , либо  $n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} > 0$ :

$$\phi_{wt} \lambda_t = \left( n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right)_+.$$

5. Суммируем обе части равенства по  $w \in W$ :

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

6. Подставим  $\lambda_t$  из (5) в (4), получим требуемое.

# Рациональный EM-алгоритм

**Идея:** Е-шаг встраивается внутрь М-шага,  
чтобы не хранить трёхмерный массив значений  $n_{dwt}$ .

**Вход:** коллекция  $D$ , число тем  $|T|$ , число итераций  $i_{\max}$ ;

**Выход:** матрицы терминов тем  $\Theta$  и тем документов  $\Phi$ ;

инициализация  $\phi_{wt}, \theta_{td}$  для всех  $d \in D, w \in W, t \in T$ ;

**для** всех итераций  $i = 1, \dots, i_{\max}$

$n_{wt}, n_{td} := 0$  для всех  $d \in D, w \in W, t \in T$ ;

**для** всех документов  $d \in D$  и всех слов  $w \in d$

$n_{tdw} := n_{dw} \operatorname{norm}_{t \in T}(\phi_{wt} \theta_{td})$  для всех  $t \in T$ ;

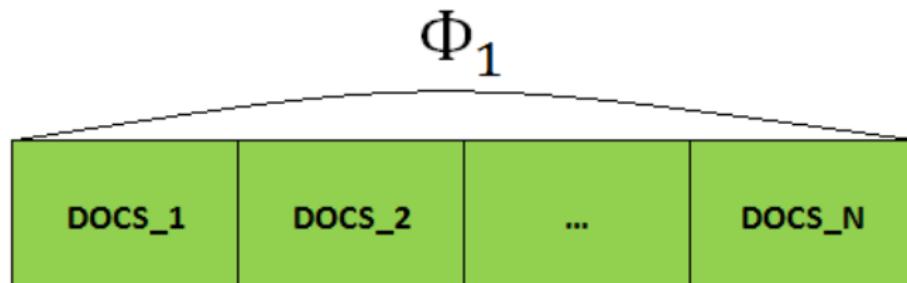
$n_{wt} += n_{tdw}; n_{td} += n_{tdw}$  для всех  $t \in T$ ;

$\phi_{wt} := \operatorname{norm}_{w \in W} \left( n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right)$  для всех  $w \in W, t \in T$ ;

$\theta_{td} := \operatorname{norm}_{t \in T} \left( n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \right)$  для всех  $d \in D, t \in T$ ;

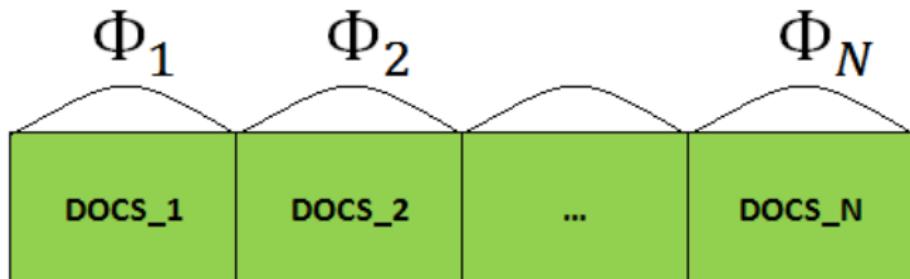
# Оффлайн EM-алгоритм

1. Многократное итерирование по коллекции.
2. Однократный проход по документу.
3. Необходимость хранить матрицу  $\Theta$ .
4.  $\Phi$  обновляется в конце каждого прохода по коллекции.
5. Применяется при обработке небольших коллекций.



# Онлайн EM-алгоритм

1. Однократный проход по коллекции.
2. Многократное итерирование по документу.
3. Нет необходимости хранить матрицу  $\Theta$ .
4.  $\Phi$  обновляется через определённое число обработанных документов.
5. Применяется при обработке больших коллекций в потоковом режиме.



# Онлайновый EM-алгоритм (BigARTM)

**Вход:** коллекция  $D$ , число тем  $|T|$ , параметры  $i_{\max}, j_{\max}, \gamma$ ;

**Выход:** матрицы терминов тем  $\Theta$  и тем документов  $\Phi$ ;

инициализировать  $n_{wt} := 0$  и  $\phi_{wt}$ ;

для всех  $i = 1, \dots, i_{\max}$  (для больших коллекций  $i_{\max} = 1$ )

для всех документов  $d \in D$

инициализировать  $\theta_{td} := \frac{1}{|T|}$ ;

для всех  $j = 1, \dots, j_{\max}$  (итерации по документу)

$n_{tdw} := \text{norm}_{t \in T}(\phi_{wt} \theta_{td})$  для всех  $w \in d$ ;

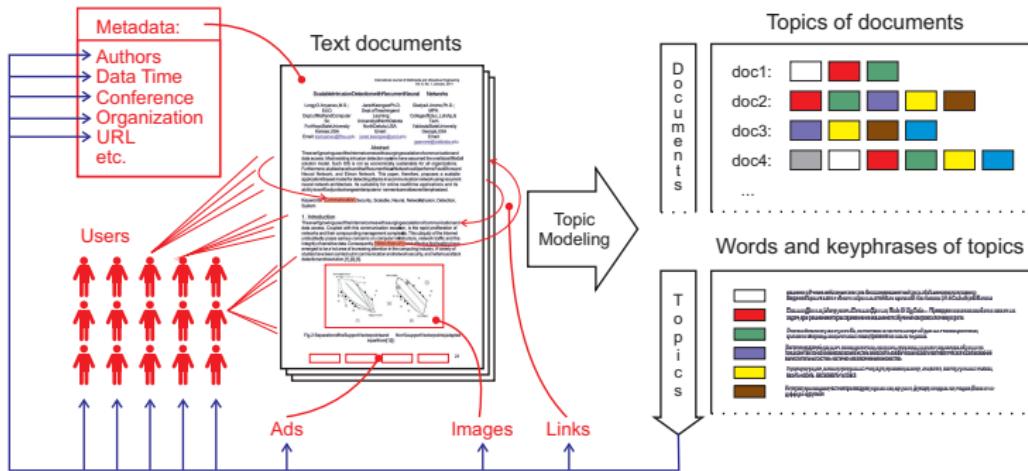
$\theta_{td} := \text{norm}_{t \in T}\left(\sum_w n_{tdw} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\right)$ ;

$n_{wt} := \gamma n_{wt} + n_{tdw}$ ;

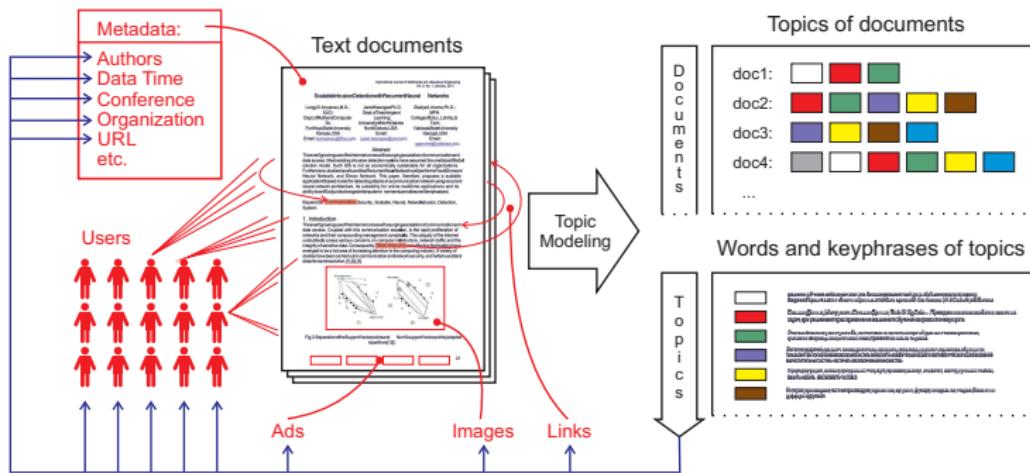
если пора обновить матрицу  $\Phi$  то

$\phi_{wt} := \text{norm}_{w \in W}\left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}}\right)$ ;

# Модальности слов



# Мультимодальная тематическая модель



Мультимодальная ТМ строит распределения тем на терминах  $p(w|t)$ , авторах  $p(a|t)$ , метках времени  $p(y|t)$ , связанных документах  $p(d'|t)$ , рекламных баннерах  $p(b|t)$ , пользователях  $p(u|t)$ , и объединяет все эти модальности в одно тематическую модель.

# Пример

Пусть у нас есть две модальности:

- ▶ обычные слова;
- ▶ слова-имена авторов (а можно, например, метки классов).

word\_1 doc\_1  
word\_1 doc\_2  
word\_1 doc\_3  
word\_1 doc\_4  
word\_1 doc\_5

...

word\_n doc\_1  
word\_n doc\_2  
word\_n doc\_3  
word\_n doc\_4  
word\_n doc\_5

$$F_w = p(w|d)$$

name\_1 doc\_1  
name\_1 doc\_2  
name\_1 doc\_3  
name\_1 doc\_4  
name\_1 doc\_5

...

name\_m doc\_1  
name\_m doc\_2  
name\_m doc\_3  
name\_m doc\_4  
name\_m doc\_5

$$F_n = p(n|d)$$

$$\Phi_w = p(w|t) \quad \times \quad \Phi_n = p(n|t)$$

topic\_1 doc\_1  
topic\_1 doc\_2  
topic\_1 doc\_3  
topic\_1 doc\_4  
topic\_1 doc\_5

topic\_2 doc\_1  
topic\_2 doc\_2  
topic\_2 doc\_3  
topic\_2 doc\_4  
topic\_2 doc\_5

topic\_3 doc\_1  
topic\_3 doc\_2  
topic\_3 doc\_3  
topic\_3 doc\_4  
topic\_3 doc\_5

$$\Theta = p(t|d)$$

# M-ARTM и EM-алгоритм

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

$W = W^1 \sqcup W^m$  как объединение словарей всех модальностей.

Максимизация логарифма **мультимодального** правдоподобия с аддитивными регуляризаторами  $R$ :

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

**EM-алгоритм:** метод простых итераций для системы уравнений

Е-шаг:  $p_{tdw} = \text{norm}_{t \in T}(\phi_{wt} \theta_{td})$

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

# BigARTM – библиотека ТМ

## Ключевые возможности

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

## Сообщество

- ▶ Открытый код <https://github.com/bigartm>
- ▶ Документация <http://bigartm.org/>



## Лицензия и среда разработки

- ▶ Freely available for commercial usage (BSD 3-Clause license)
- ▶ Cross-platform: Window, Linux, Mac OS X (x32, x64)
- ▶ Programming APIs: CLI, C++, Python (2.7, 3)

# BigARTM сильно упрощает моделирование

Для построения сложных моделей в BigARTM не нужны ни математические выкладки, ни программирование «с нуля».

## Этапы моделирования

Формализация:

Алгоритмизация:

Реализация:

Оценивание:

## Bayesian TM

Анализ требований
Вероятностная порождающая модель данных
Байесовский вывод для данной порождающей модели (VI, GS, EP)
Исследовательский код (Matlab, Python, R)
Исследовательские метрики, исследовательский код
Внедрение

## ARTM

Анализ требований	
Стандартные критерии	Свои критерии
Общий регуляризованный EM-алгоритм для любых моделей	
Промышленный код BigARTM (C++, Python API)	
Стандартные метрики	Свои метрики
Внедрение	

-- нестандартизуемые этапы, уникальная разработка для каждой задачи

-- стандартизуемые этапы

# Классические модели PLSA и LDA

**PLSA:** probabilistic latent semantic analysis<sup>2</sup> (вероятностный латентный семантический анализ):

$$R(\Phi, \Theta) = 0.$$

М-шаг – частотные оценки условных вероятностей:

$$\phi_{wt} = \underset{w}{\text{norm}}(n_{wt}), \quad \theta_{td} = \underset{t}{\text{norm}}(n_{td}).$$

**LDA:** latent dirichlet allocation<sup>3</sup> (латентное размещение Дирихле):

$$R(\Phi, \Theta) = \sum_{t,w} (\beta_w - 1) \ln \phi_{wt} + \sum_{d,t} (\alpha_t - 1) \ln \theta_{td}.$$

М-шаг – сглаженные частотные оценки с параметрами:

$$\phi_{wt} = \underset{w}{\text{norm}}(n_{wt} + \beta_w - 1), \quad \theta_{td} = \underset{t}{\text{norm}}(n_{td} + \alpha_t - 1).$$

---

<sup>2</sup>Hoffman T. Probabilistic latent semantic indexing. SIGIR 1999.

<sup>3</sup>Blei D., Ng A., Jordan M. Latent Dirichlet Allocation. 2003.

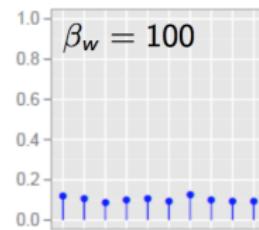
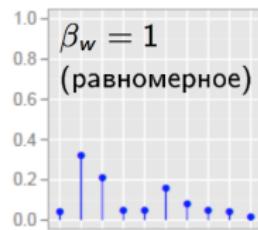
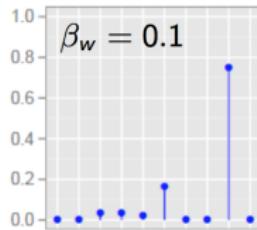
# Байесовская интерпретация LDA

**Гипотеза.** Вектор-столбцы  $\phi_t = (\phi_{wt})_{w \in W}$  и  $\theta_d = (\theta_{td})_{t \in T}$  порождаются распределениями Дирихле,  $\alpha \in \mathbb{R}^{|T|}$ ,  $\beta \in \mathbb{R}^{|W|}$ :

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

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

**Пример.** Распределение  $\phi \sim \text{Dir}(\beta)$  при  $|W| = 10$ ,  $\phi, \beta \in \mathbb{R}^{10}$ :



# MAP для LDA

Совместное правдоподобие данных и модели:

$$\ln \prod_{d \in D} \prod_{w \in d} p(d, w | \Phi, \Theta)^{n_{dw}} \prod_{t \in T} \text{Dir}(\phi_t | \beta) \prod_{d \in D} \text{Dir}(\theta_d | \alpha) \rightarrow \max_{\Phi, \Theta}$$

Регуляризатор – логарифм априорного распределения:

$$R(\Phi, \Theta) = \sum_{t,w} (\beta_w - 1) \ln \phi_{wt} + \sum_{d,t} (\alpha_t - 1) \ln \theta_{td}$$

М-шаг – сглаженные или слабо разреженные оценки:

$$\phi_{wt} = \text{norm}_w(n_{wt} + \beta_w - 1), \quad \theta_{td} = \text{norm}_t(n_{td} + \alpha_t - 1).$$

- ▶  $\beta_w > 1, \alpha_t > 1$  – сглаживание,
- ▶  $0 < \beta_w < 1, 0 < \alpha_t < 1$  – слабое разреживание,
- ▶  $\beta_w = 1, \alpha_t = 1$  – равном. априорное распределение, PLSA.

# Почему распределение Дирихле?

## Плюсы:

- ▶ Удобство байесовского вывода.
- ▶ Описывает широкий класс распределений на симплексе.
- ▶ Позволяет управлять разреженностью  $\Phi$  и  $\Theta$ .
- ▶ При малых  $n_{wt}$  и  $n_{td}$  уменьшает переобучение.

## Минусы:

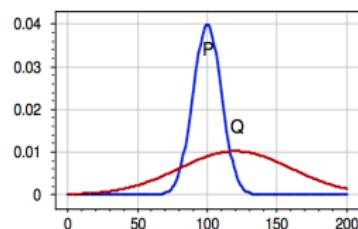
- ▶ Не имеет лингвистических обоснований.
- ▶ Не даёт выигрыша против PLSA на больших коллекциях.
- ▶ Слабый разреживатель: запрещены отрицательные параметры.
- ▶ Слабый регуляризатор: проблема неединственности остаётся.

# Дивергенция Кульбака-Лейблера

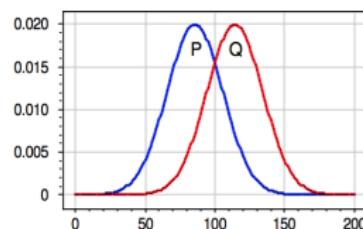
1.  $\text{KL}(P\|Q) \geq 0$ ;  $\text{KL}(P\|Q) = 0 \Leftrightarrow P = Q$ ;
2. Минимизация  $\text{KL}$  эквивалентна максимизации правдоподобия:

$$\text{KL}(P\|Q(\alpha)) = \sum_{i=1}^n p_i \ln \frac{p_i}{q_i(\alpha)} \rightarrow \min_{\alpha} \Leftrightarrow \sum_{i=1}^n p_i \ln q_i(\alpha) \rightarrow \max_{\alpha}$$

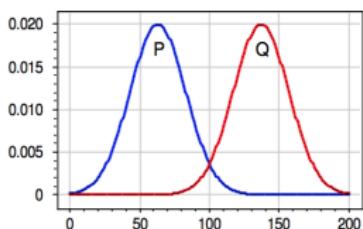
3. Если  $\text{KL}(P\|Q) < \text{KL}(Q\|P)$ , то  $P$  вложено в  $Q$ :



$$\begin{aligned}\text{KL}(P\|Q) &= 0.44 \\ \text{KL}(Q\|P) &= 2.97\end{aligned}$$



$$\begin{aligned}\text{KL}(P\|Q) &= 0.44 \\ \text{KL}(Q\|P) &= 0.44\end{aligned}$$



$$\begin{aligned}\text{KL}(P\|Q) &= 2.97 \\ \text{KL}(Q\|P) &= 2.97\end{aligned}$$

# Обобщённая интерпретация LDA

Сглаживание распределений по KL-дивергенции:

Приблизить  $\phi_{wt} = p(w|t)$  к заданным распределениям  $\beta_t(w)$ ,

Приблизить  $\theta_{td} = p(t|d)$  к заданным распределениям  $\alpha_d(t)$ :

$$\sum_{t \in T} \tau_t \text{KL}(\beta_t(w) || \phi_{wt}) \rightarrow \min_{\Phi}, \quad \sum_{d \in D} \tau_d \text{KL}(\alpha_d(t) || \theta_{td}) \rightarrow \min_{\Theta}$$

Взвешенная сумма регуляризаторов:

$$R(\Phi, \Theta) = \sum_{t \in T} \tau_t \sum_{w \in W} \beta_t(w) \ln \phi_{wt} + \sum_{d \in D} \tau_d \sum_{t \in T} \alpha_d(t) \ln \theta_{td}.$$

Формулы M-шага:

$$\phi_{wt} = \underset{w}{\text{norm}} \left( n_{wt} + \underbrace{\tau_t \beta_t(w)}_{\beta_{wt}} \right), \quad \theta_{td} = \underset{t}{\text{norm}} \left( n_{td} + \underbrace{\tau_t \alpha_d(t)}_{\alpha_{td}} \right)$$

# Сглаживание, разреживание и част. обучение

Формулы М-шага:

$$\phi_{wt} = \text{norm}_w(n_{wt} + \beta_{wt}), \quad \theta_{td} = \text{norm}_t(n_{td} + \alpha_{td})$$

Разреживание и сглаживание описывается общей формулой:

- ▶ разреживание –  $\beta_{wt} < 0, \alpha_{td} < 0$
- ▶ сглаживание –  $\beta_{wt} > 0, \alpha_{td} > 0$

Частичное обучение темы  $t$ :

- ▶  $\beta_{wt} = +\tau_{white}[w \in W_t]$  – «белый список» терминов
- ▶  $\beta_{wt} = +\tau_{black}[w \in W_t]$  – «чёрный список» терминов
- ▶  $\alpha_{td} = +\tau_{white}[d \in D_t]$  – «белый список» документов
- ▶  $\alpha_{td} = +\tau_{black}[d \in D_t]$  – «чёрный список» документов

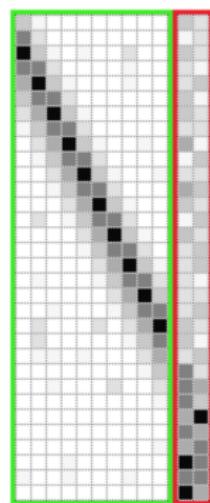
# Предметные и фоновые темы

$T = S \sqcup B$  – множество всех тем

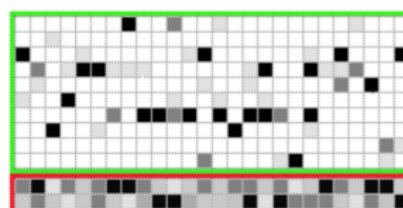
$S$  – разреженные *предметные* темы, специальная лексика

$B$  – сглаженные *фоновые* темы, общая лексика языка

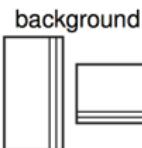
$$\Phi_{W \times T}$$



$$\Theta_{T \times D}$$

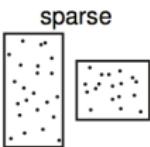


# Регуляризаторы улучшения интерпретируемости



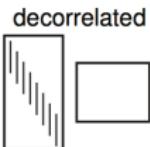
Сглаживание фоновых тем  $B \subset T$ :

$$R(\Phi, \Theta) = \beta_0 \sum_{t \in B} \sum_w \beta_w \ln \phi_{wt} + \alpha_0 \sum_d \sum_{t \in B} \alpha_t \ln \theta_{td}$$



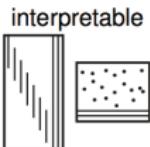
Разреживание предметных тем  $S = T \setminus B$ :

$$R(\Phi, \Theta) = -\beta_0 \sum_{t \in S} \sum_w \beta_w \ln \phi_{wt} - \alpha_0 \sum_d \sum_{t \in S} \alpha_t \ln \theta_{td}$$



Декорелирование для повышения различности тем:

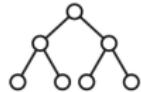
$$R(\Phi) = -\frac{\tau}{2} \sum_{t,s} \sum_w \phi_{wt} \phi_{ws}$$



Сглаживание + разреживание + декорелирование  
для улучшения интерпретируемости тем

# Иерархические, темпоральные, регрессионные модели

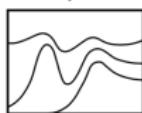
hierarchy



Связь родительских тем  $t$  с дочерними подтемами  $s$ :

$$R(\Phi, \Psi) = \tau \sum_{t \in T} \sum_{w \in W} n_{wt} \ln \sum_{s \in S} \phi_{ws} \psi_{st}.$$

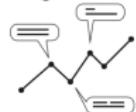
temporal



Темпоральные модели с модальностью времени  $i$ :

$$R(\Phi) = -\tau \sum_{i \in I} \sum_{t \in T} |\phi_{it} - \phi_{i-1,t}|.$$

regression



Линейная модель регрессии  $\hat{y}_d = \langle v, \theta_d \rangle$  документов:

$$R(\Theta, v) = -\tau \sum_{d \in D} \left( y_d - \sum_{t \in T} v_t \theta_{td} \right)^2.$$

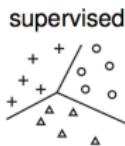
n of topics



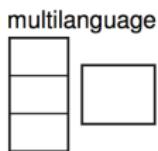
Разреживание  $p(t)$  для отбора тем:

$$R(\Theta) = -\tau \sum_{t \in T} \frac{1}{|T|} \ln p(t), \quad p(t) = \sum_{d \in D} p(d) \theta_{td}.$$

# Спец. мультимодальные модели



Модальности меток классов или категорий для задач классификации и категоризации текстов.



Модальность языков и регуляризация со словарём  
 $\pi_{uwt} = p(u|w, t)$  переводов с языка  $k$  на  $\ell$ :

$$R(\Phi, \Pi) = \tau \sum_{u \in W^k} \sum_{t \in T} n_{ut} \ln \sum_{w \in W^\ell} \pi_{uwt} \phi_{wt}$$

graph



Модальность вершин графа  $v$ , содержащих  $D_v$ :

$$R(\Phi) = -\frac{\tau}{2} \sum_{(u,v) \in E} S_{uv} \sum_{t \in T} n_t^2 \left( \frac{\phi_{vt}}{|D_v|} - \frac{\phi_{ut}}{|D_u|} \right)^2.$$

geospatial



Модальность геолокаций  $g$  с близостью  $S_{gg'}$ :

$$R(\Phi) = -\frac{\tau}{2} \sum_{g,g' \in G} S_{gg'} \sum_{t \in T} n_t^2 \left( \frac{\phi_{gt}}{n_g} - \frac{\phi_{g't}}{n_{g'}} \right)^2$$

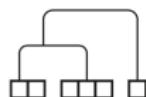
# Обход «мешка слов»

n-gram



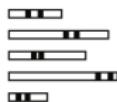
Модальности  $n$ -грамм, коллокаций,  
именованных сущностей

syntax



Модальность  $n$ -грамм после применения SyntaxNet

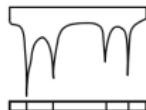
coherence



Совстречаемость слов  $n_{uv}$  в битермах ( $u, v$ ):

$$R(\Phi) = \tau \sum_{u,v} n_{uv} \ln \sum_t n_t \phi_{ut} \phi_{vt}$$

segmentation



Регуляризация  $E$ -шага, постобработка распределений  
 $p(t|d, w)$  для тематической сегментации

# Поиск этнодискурса

**Задача:** найти все этно-релевантные темы для мониторинга межнациональных отношений.

Используем словарь из 300 этнонимов для обучения тем.

Мешок регуляризаторов:

$$\mathcal{L} \left( \begin{array}{c} \text{PLSA} \\ \Phi \quad \Theta \end{array} \right) + R \left( \begin{array}{c} \text{interpretable} \\ \text{vertical bars} \quad \text{grid} \end{array} \right) + R \left( \begin{array}{c} \text{multimodal} \\ \text{vertical bars} \quad \text{square} \end{array} \right) \\ + R \left( \begin{array}{c} \text{temporal} \\ \text{wavy line} \end{array} \right) + R \left( \begin{array}{c} \text{geospatial} \\ \text{map} \end{array} \right) + R \left( \begin{array}{c} \text{sentiment} \\ \text{multiple scales} \end{array} \right) \rightarrow \max$$

**Результаты:** число релевантных тем выросло с 45 для LDA до 83 для ARTM.

---

*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.*

# Разведочный поиск в блогах

**Задача:** поиск документов по длинному запросу.

Мешок регуляризаторов:

$$\mathcal{L}\left(\begin{array}{c|c} \text{PLSA} \\ \Phi & \Theta \end{array}\right) + R\left(\begin{array}{c|c} \text{interpretable} \\ \text{vertical bars} & \text{grid} \end{array}\right) + R\left(\begin{array}{c|c} \text{multimodal} \\ \text{horizontal bars} & \square \end{array}\right) + R\left(\begin{array}{c|c} \text{n-gram} \\ \text{grid} & \text{grid} \end{array}\right) \rightarrow \max$$

**Результаты:**

- Точность и полнота увеличились с (65%, 73%) для LDA до (85%, 92%) для ARTM на данных Habrahabr.ru и TechCrunch.com.
- Точность и полнота сравнимы с результатами асессоров.
- Тематический поиск даёт результат мгновенно, асессоры тратят на эту же работу в среднем 30 минут.

---

A.Ianina, K.Vorontsov. Multi-objective topic modeling for exploratory search in tech news. AINL, 2017.

# Модель новостных потоков

## Задачи:

- наращивать 3х-уровневую иерархию динамически
- обеспечить интерпретируемость и именование всех тем
- управлять медиакомпаниями и творческими заданиями

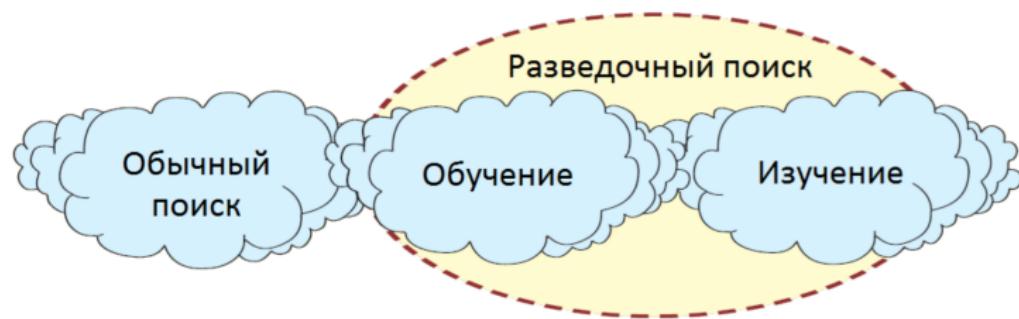
Мешок регуляризаторов:

$$\begin{aligned} \mathcal{L}\left(\Phi \begin{array}{|c|c|} \hline \text{PLSA} & \\ \hline \Theta & \\ \hline \end{array}\right) + R\left(\begin{array}{|c|c|} \hline \text{interpretable} & \\ \hline \text{grid} & \\ \hline \end{array}\right) + R\left(\begin{array}{|c|c|} \hline \text{hierarchy} & \\ \hline \text{tree} & \\ \hline \end{array}\right) + R\left(\begin{array}{|c|c|} \hline \text{temporal} & \\ \hline \text{waveform} & \\ \hline \end{array}\right) \\ + R\left(\begin{array}{|c|c|} \hline \text{multimodal} & \\ \hline \text{grid} & \\ \hline \end{array}\right) + R\left(\begin{array}{|c|c|} \hline \text{n-gram} & \\ \hline \text{matrix} & \\ \hline \end{array}\right) + R\left(\begin{array}{|c|c|} \hline \text{multilanguage} & \\ \hline \text{grid} & \\ \hline \end{array}\right) + R\left(\begin{array}{|c|c|} \hline \text{sentiment} & \\ \hline \text{grid} & \\ \hline \end{array}\right) \rightarrow \max \end{aligned}$$

Результат: ... (исследование продолжается)

# Разведочный поиск (exploration search)

- ▶ пользователь может не знать ключевых терминов,
- ▶ запросом может быть текст произвольной длины,
- ▶ информационной потребностью – систематизация знаний



навигация в сети,  
поиск фактов,  
упоминаний,  
конкретных ответов

самообразование,  
тематический поиск,  
систематизация  
знаний

исследование,  
экспертиза,  
реферирование,  
мониторинг тем

## Разведочный тематический поиск

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

$\theta_{tq}$  – тематический профиль запроса  $q$ .

$\theta_{td}$  – тематический профили документов  $d \in D$

Косинусная мера близости документа  $d$  и запроса  $q$ :

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

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

Выдача тематического поиска – первые  $k$  документов

# Данные блога Habrahabr.ru

## Данные

- ▶ ≈ 175K статей.
- ▶ Модальности:
  - ▶ 10.5K униграмм, 742K биграмм
  - ▶ 524 автора статей
  - ▶ 10К комментаторов
  - ▶ 2546 тегов
  - ▶ 123 хаба (категории)

## Предобработка текстов

- ▶ лемматизация pymorphy2
- ▶ Отброшены 5% наиболее частотных слов
- ▶ удаление пунктуации
- ▶ нижний регистр, ё → е

# Методика оценивания качества поиска

## Поисковый запрос

Набор ключевых слов или фрагментов текста около одной страницы А4

## Поисковый выдача

Документы  $dc$  распределены с распределением  $p(t|d)$ , близким к распределению запроса  $p(t|q)$  запроса.

## Два задания асессорам

1. Найти как можно больше статей, пользуясь любыми средствами (засечь время).
2. Оценить релевантность поисковой выдачи на том же запросе.

**Набор МарКейблс**

Набор МарКейблс – программная модель (Базисной) выполнения распределения запросов для больших объемов данных в рамках параллелизма параллельных, представляющая собой набор Java-классов и исполнительных утилит для создания и обработки задач на параллельную обработку.

Основные компоненты Набора МарКейблс можно сформулировать так:

- обработка множества больших объемов данных;
- высокая производительность;
- автоматическое распараллеливание задач;
- работа на недорогом оборудовании;
- автоматическая обработка отложенных выполнения задач.

Набор – типичная программа для платформы (Java-платформа) параллельного распределенного приложения для высокопараллельных областей (Большие данные, распределенные системы, HPC) задач.

Набор включает в себя следующие компоненты:

1. НСРР – распределенная файловая система;
2. Набор МарКейблс – программная модель (Базисной) выполнения распределения запросов для больших объемов данных в рамках параллелизма параллельных.

Компоненты, используемые в качестве ядра Набора МарКейблс в структуре НСРР, стали причиной ряда успехов мест в соревнованиях, в том числе и в международных турнирах. Чем в конечном итоге, определило привлечения платформы Набор в цепочку К победителям можно отнести:

Со временем масштабируемость ядра Набор – «ЯК» значительно улучшилась, «ЯК» параллелизации задач.

Система склонности «Федоровка» распределения запросов и клиентской библиотеки, реализующих распределенный алгоритм. Как следствие:

Отсутствие поддержки контекстной программы модели выполнения распределенных запросов в ядре НСРР V.0 подтверждается теми же фактами.

Изменение единичной точки отказа и, как следствие, невозможность использования в средах с высокими требованиями к надежности;

Проблемы версийской совместимости требований по единому времени обновления всех компонентов ядра ядра при обновлении платформы Набор (установка новой версии или пакета обновлений);

Рис.: Пример запроса для разведочного поиска

## Пример: фрагмент запроса

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 целевые 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.

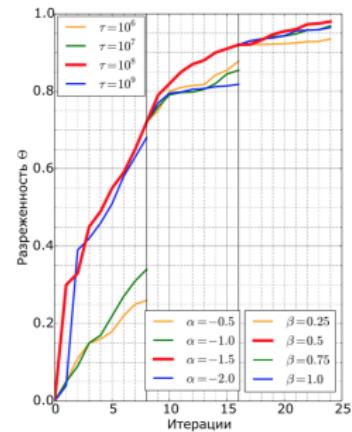
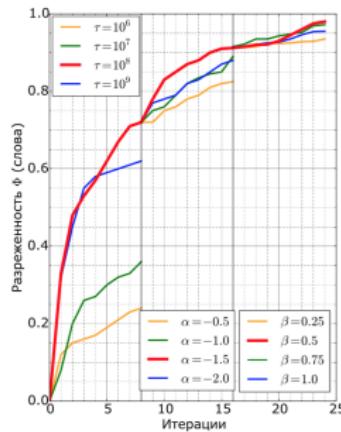
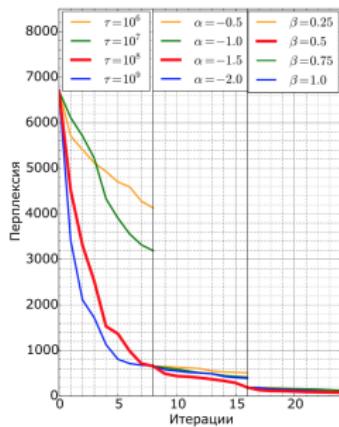
# Тематика запросов разведочного поиска

Примеры заголовков разведочных запросов к Хабру  
(объём каждого запроса — около одной страницы А4):

Алгоритмы раскраски графов	Система IBM Watson
Рекомендательная система Netflix	3D-принтеры
Методики быстрого набора текста	CERN-кластер
Космические проекты Илона Маска	AB-тестирование
Технологии Hadoop MapReduce	Облачные сервисы
Беспилотный автомобиль Google car	Контекстная реклама
Крипtosистемы 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

# Подбор коэффициентов регуляризации

- декореллирование распределений терминов в темах ( $\tau$ ),
- разреживание распределений тем в документах ( $\alpha$ ),
- сглаживание распределений терминов в темах ( $\beta$ ).



# Оценки качества поиска

Precision — доля релевантных среди найденных

Recall — доля найденных среди релевантных

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

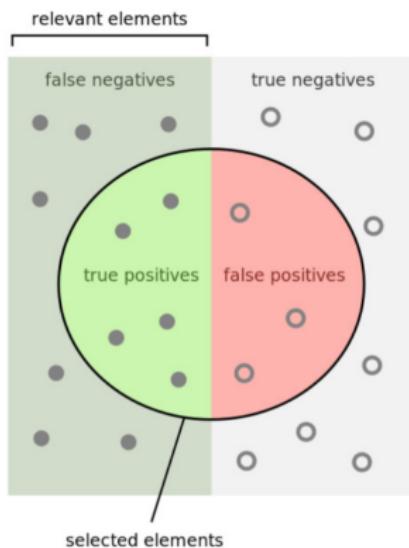
$$R = \frac{TP}{TP + FN} \text{ — полнота, (recall)}$$

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

TP (true positive) — найденные релевантные

FP (false positive) — найденные нерелевантные

FN (false negative) — ненайденные релевантные

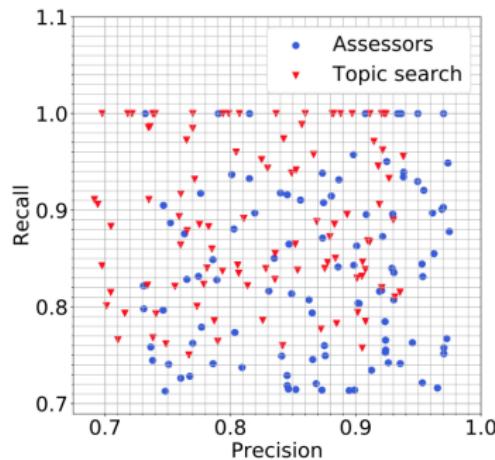


$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$
 | 
$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

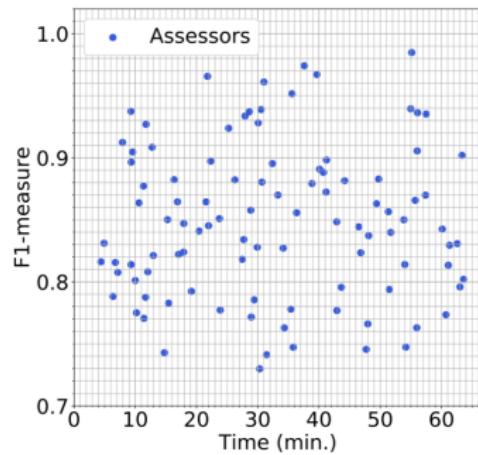
# Результаты измерений

100 запросов, 3 асессора на запрос

точность и полнота поиска



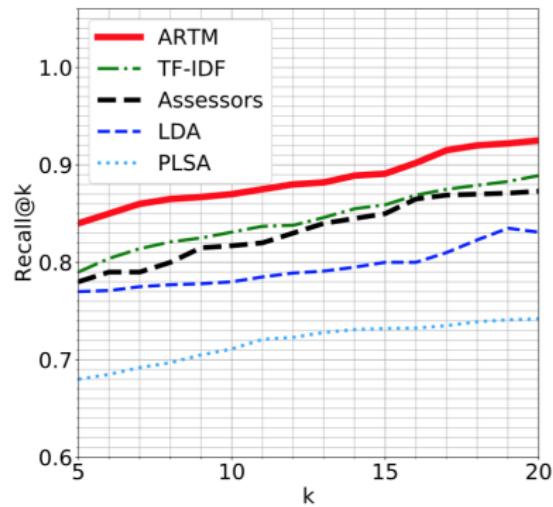
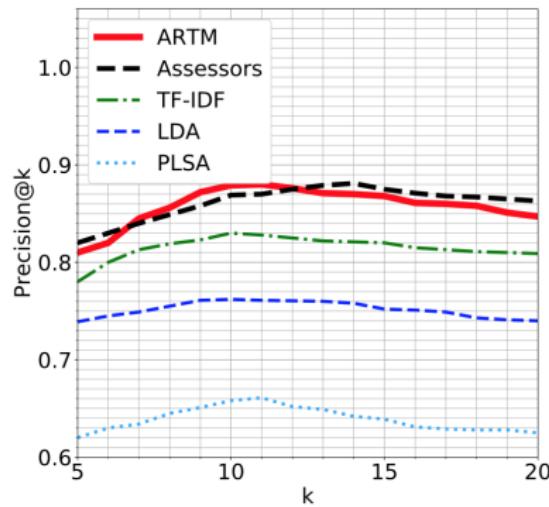
время и  $F_1$ -мера (асессоры)



- ▶ среднее время обработки запроса асессором – 30 мин.
- ▶ точность выше у асессоров, полнота – у поисковика.

# Сравнение с ассессорами

Точность и полнота по первым  $k$  позициям поисковой выдачи  
(коллекция Habrahabr.ru)

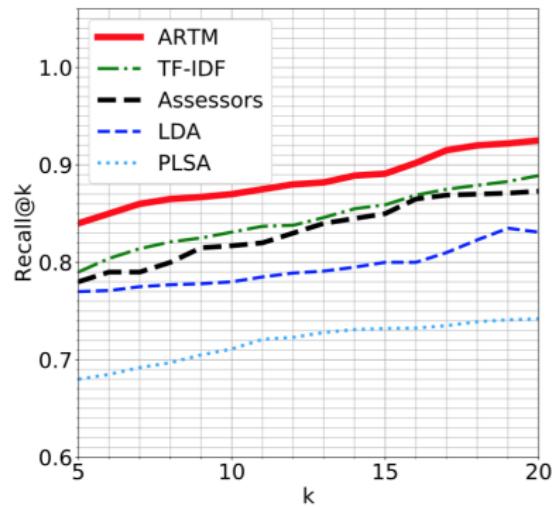
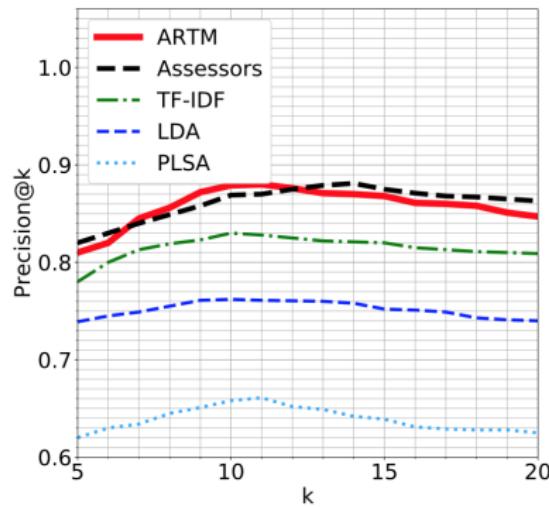


---

A.Ianina, K.Vorontsov. Multi-objective topic modeling for exploratory search in tech news. AINL, 2017.

# Сравнение с ассессорами

Точность и полнота по первым  $k$  позициям поисковой выдачи  
(коллекция Habrahabr.ru)



---

A.Ianina, K.Vorontsov. Multi-objective topic modeling for exploratory search in tech news. AINL, 2017.

# Влияние регуляризаторов на качество

Декоррелирование,  $\Theta$ -разреживание,  $\Phi$ -сглаживание

	Habrahabr				TechCrunch			
	$R = 0$	Д	Д $\Theta$	Д $\Theta\Phi$	$R = 0$	Д	Д $\Theta$	Д $\Theta\Phi$
Prec@5	0.628	0.748	0.771	<b>0.810</b>	0.652	0.775	0.779	<b>0.819</b>
Prec@10	0.653	0.776	0.812	<b>0.879</b>	0.679	0.787	0.819	<b>0.867</b>
Prec@15	0.642	0.765	0.792	<b>0.868</b>	0.669	0.773	0.798	<b>0.833</b>
Prec@20	0.643	0.759	0.783	<b>0.847</b>	0.673	0.777	0.792	<b>0.825</b>
Recall@5	0.692	0.784	0.805	<b>0.840</b>	0.673	0.812	0.812	<b>0.835</b>
Recall@10	0.714	0.814	0.834	<b>0.870</b>	0.685	0.821	0.845	<b>0.868</b>
Recall@15	0.725	0.835	0.867	<b>0.891</b>	0.712	0.859	0.869	<b>0.890</b>
Recall@20	0.735	0.862	0.891	<b>0.925</b>	0.723	0.882	0.895	<b>0.919</b>

- ▶ комбинирование регуляризаторов улучшает качество поиска
- ▶ хотя исходно все регуляризаторы были нацелены на улучшение интерпретируемости и не оптимизируют поиск явно

# Итоги занятия

- ▶ Тематическое моделирование — это восстановление латентных тем по коллекции текстовых документов.
- ▶ Задача сводится к стохастическому матричному разложению.
- ▶ Задача является некорректно поставленной, так как множество её решений в общем случае бесконечно.
- ▶ Стандартные методы PLSA и LDA не решают эту проблему.
- ▶ Аддитивная регуляризация (ARTM) доопределяет задачу и позволяет строить модели с заданными свойствами.
- ▶ Онлайновый EM-алгоритм хорошо распараллеливается и тематизирует большие коллекции за один проход.
- ▶ Разведочный информационный поиск — одно из основных перспективных приложений тематического моделирования.