Оптимизация и регуляризация вероятностных тематических моделей (лекция №1)

Воронцов Константин Вячеславович

(ФИЦ ИУ РАН • МФТИ • МГУ • ВШЭ • Яндекс • FORECSYS • Aithea)

• Традиционная молодёжная летняя школа •

14–20 июня 2017

Содержание

Наши мотивации

Мотивации исследований по тематическим моделям:

- Создавать сервисы, делающие людей информированнее
- Автоматизировать научный поиск
- Создать оружие пропаганды
- Создать защиту против оружия пропаганды
- Научиться выделять темы в любых текстовых коллекциях и систематизировать информацию в каждой теме

Мотивации этой пары лекций:

- Больше рассказать про то, как ставить прикладные задачи оптимизации, и немного про то, как их решать
- Обозначить несколько открытых проблем

Пусть

- W — конечное множество слов (терминов, токенов)
- D — конечное множество текстовых документов
- T — конечное множество тем
- каждое слово w в документе d связано с некоторой темой t
- $D \times W \times T$ — дискретное вероятностное пространство
- порядок слов в документе не важен (bag of words)
- порядок документов в коллекции не важен (bag of docs)
- 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: $p(w|d, t) = p(w|t)$

Тематическая модель, по формуле полной вероятности:

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

Постановка задачи тематического моделирования

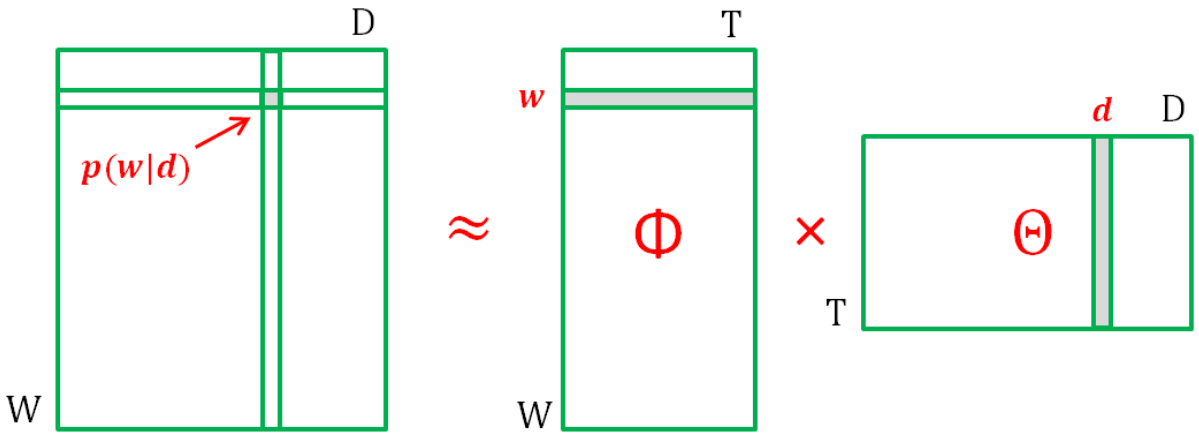
Дано: коллекция текстовых документов

- n_{dw} — частоты терминов в документах, $\hat{p}(w|d) = \frac{n_{dw}}{n_d}$

Найти: параметры тематической модели $p(w|d) = \sum_{t \in T} \phi_{wt}\theta_{td}$

- $\phi_{wt} = p(w|t)$ — вероятности терминов w в каждой теме t
- $\theta_{td} = p(t|d)$ — вероятности тем t в каждом документе d

Это задача стохастического матричного разложения:



Как проверить адекватность модели?

Тесты согласия эмпирических распределений с модельными:

- документ d порождён тематической моделью:

$$\hat{p}(w|d) = \frac{n_{dw}}{n_d} \quad \sim \quad p(w|d) = \sum_{t \in T} \phi_{wt}\theta_{td}$$

- документ d в теме t порождён тематической моделью:

$$\hat{p}(w|d,t) = \frac{n_{tdw}}{n_{td}} \quad \sim \quad p(w|t) = \phi_{wt}$$

Проблемы:

- На практике никто таких проверок не делает
- Асимптотики χ^2 не верны для разреженных распределений
- Перестановочные тесты увеличивают объём вычислений

Принцип максимума правдоподобия

Правдоподобие — плотность распределения выборки $(d_i, w_i)_{i=1}^n$:

$$\prod_{i=1}^{n} p(d_i, w_i) = \prod_{d \in D} \prod_{w \in d} p(d, w)^{n_{dw}}$$

Максимизация логарифма правдоподобия

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \ln p(w|d) p(d) \to \max_{\Phi, \Theta}$$

эквивалентна максимизации функционала

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

при ограничениях неотрицательности и нормировки

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

Задачи, некорректно поставленные по Адамару

Задача *корректно поставлена*, если её решение

- существует,
- единственно,
- устойчиво.

Жак Саломон Адамар
(1865–1963)

Наша задача матричного разложения *некорректно поставлена*:
если Φ, Θ — решение, то стохастические Φ', Θ' — тоже решения

- $\Phi'\Theta' = (\Phi S)(S^{-1}\Theta)$, $\operatorname{rank} S = |T|$
- $\mathscr{L}(\Phi', \Theta') = \mathscr{L}(\Phi, \Theta)$
- $\mathscr{L}(\Phi', \Theta') \leqslant \mathscr{L}(\Phi, \Theta) + \varepsilon$ — приближённые решения

Регуляризация — стандартный приём доопределения решения с помощью дополнительных критериев.

ARTM: аддитивная регуляризация тематических моделей

Максимизация логарифма правдоподобия с регуляризатором:

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

EM-алгоритм: метод простой итерации для системы уравнений

$$\text{E-шаг:} \quad p_{tdw} \equiv p(t|d,w) = \underset{t\in T}{\text{norm}}\left(\phi_{wt}\theta_{td}\right)$$

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

где $\underset{t\in T}{\text{norm}}(x_t) = \dfrac{\max\{x_t,0\}}{\sum\limits_{s\in T}\max\{x_s,0\}}$ — операция нормировки вектора.

Элементарная интерпретация EM-алгоритма

EM-алгоритм — это чередование E и M шагов до сходимости.

E-шаг: условные вероятности тем $p(t|d,w)$ для всех t,d,w вычисляются через ϕ_{wt}, θ_{td} по формуле Байеса:

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

M-шаг: при $R=0$ частотные оценки условных вероятностей вычисляются суммированием счётчика $n_{tdw} = n_{dw}p(t|d,w)$:

$$\phi_{wt} = \frac{n_{wt}}{n_t}, \qquad n_{wt} = \sum_{d\in D} n_{tdw}, \qquad n_t = \sum_{w\in W} n_{wt};$$

$$\theta_{td} = \frac{n_{td}}{n_d}, \qquad n_{td} = \sum_{w\in d} n_{tdw}, \qquad n_d = \sum_{t\in T} n_{td}.$$

Условия вырожденности модели для тем и документов

Решение может быть вырожденным для некоторых тем (столбцов матриц Φ) и документов (столбцов матрицы Θ).

Тема t вырождена, если для всех терминов $w \in W$

$$n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \leqslant 0.$$

Если тема t вырождена, то $p(w|t) = \phi_{wt} \equiv 0$; это означает, что тема исключается из модели (происходит отбор тем).

Документ d вырожден, если для всех тем $t \in T$

$$n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}} \leqslant 0.$$

Если документ d вырожден, то $p(t|d) = \theta_{td} \equiv 0$; это означает, что модель не в состоянии описать данный документ.

Напоминания. Условия Каруша–Куна–Таккера

Задача математического программирования:

$$\begin{cases} f(x) \to \min\limits_x; \\ g_i(x) \leqslant 0, \quad i = 1, \ldots, m; \\ h_j(x) = 0, \quad j = 1, \ldots, k. \end{cases}$$

Необходимые условия. Если x — точка локального минимума, то существуют множители μ_i, $i = 1, \ldots, m$, λ_j, $j = 1, \ldots, k$:

$$\begin{cases} \dfrac{\partial \mathscr{L}}{\partial x} = 0, \quad \mathscr{L}(x; \mu, \lambda) = f(x) + \sum\limits_{i=1}^{m} \mu_i g_i(x) + \sum\limits_{j=1}^{k} \lambda_j h_j(x); \\ g_i(x) \leqslant 0; \ h_j(x) = 0; \ \text{(исходные ограничения)} \\ \mu_i \geqslant 0; \ \text{(двойственные ограничения)} \\ \mu_i g_i(x) = 0; \ \text{(условие дополняющей нежёсткости)} \end{cases}$$

Вывод системы уравнений из условий Каруша–Куна–Таккера

1. Условия ККТ для ϕ_{wt} (для θ_{td} всё аналогично):

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

2. Умножим обе части равенства на ϕ_{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 $\lambda_t \leqslant 0$, 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 λ_t из (5) в (4), получим требуемое. ■

Рациональный EM-алгоритм

Идея: E-шаг встраивается внутрь M-шага,
чтобы не хранить трёхмерный массив значений n_{dwt}.

Вход: коллекция D, число тем $|T|$, число итераций $i_{\max}$;
Выход: матрицы терминов тем Θ и тем документов Φ;

инициализация ϕ_{wt}, θ_{td} для всех $d \in D$, $w \in W$, $t \in T$;
для всех итераций $i = 1, \ldots, i_{\max}$

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

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

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

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

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

Онлайновый EM-алгоритм (реализован в BigARTM)

Вход: коллекция D, число тем $|T|$, параметры $i_{\max}$, $j_{\max}$, γ;
Выход: матрицы терминов тем Θ и тем документов Φ;

инициализировать $n_{wt} := 0$ и ϕ_{wt};
для всех $i = 1, \ldots, i_{\max}$ (для больших коллекций $i_{\max} = 1$)
  для всех документов $d \in D$
    инициализировать $\theta_{td} := \frac{1}{|T|}$;
    для всех $j = 1, \ldots, j_{\max}$ (итерации по документу)
      $n_{tdw} := n_{dw} \operatorname*{norm}_{t \in T}\left(\phi_{wt}\theta_{td}\right)$ для всех $w \in d$;
      $\theta_{td} := \operatorname*{norm}_{t \in T}\left(\sum_w n_{tdw} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right)$;
    $n_{wt} := \gamma n_{wt} + n_{tdw}$;
    если пора обновить матрицу Φ **то**
      $\phi_{wt} := \operatorname*{norm}_{w \in W}\left(n_{wt} + \phi_{wt}\frac{\partial R}{\partial \phi_{wt}}\right)$;

Обобщение на произвольные функции потерь

Максимизация логарифма правдоподобия с регуляризатором:

$$\sum_{d,w} n_{dw} \ell\left(\sum_t \phi_{wt}\theta_{td}\right) + R(\Phi,\Theta) \to \max_{\Phi,\Theta}$$

EM-алгоритм: метод простой итерации для системы уравнений

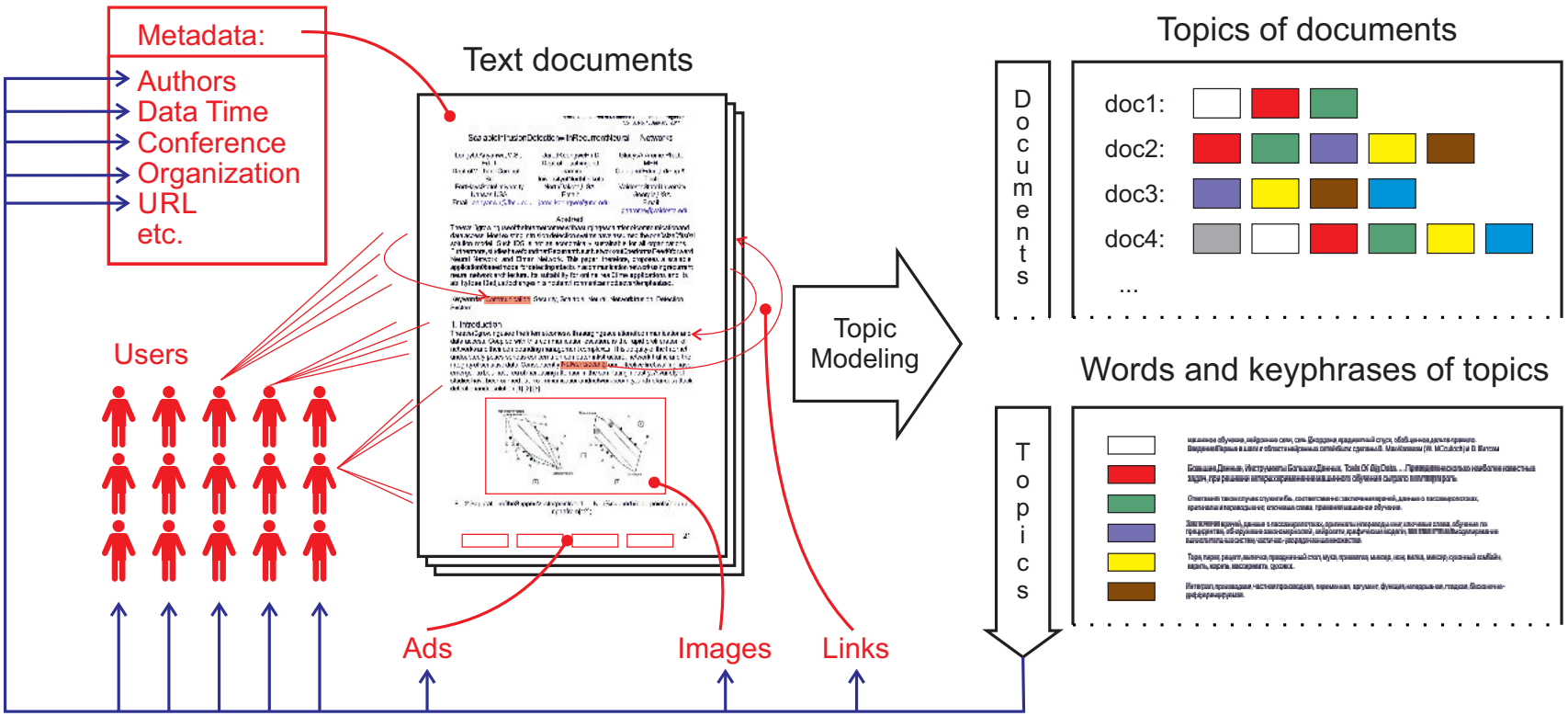
$$\text{E-шаг:} \quad p_{tdw} = \phi_{wt}\theta_{td}\,\ell'\left(\sum_s \phi_{ws}\theta_{sd}\right)$$

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

Вероятностная интерпретация p_{tdw} — только при $\ell = \ln$.

Задачи мультимодального тематического моделирования

Темы определяют распределения не только терминов $p(w|t)$, но и других *модальностей*: $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{ссылка}|t)$, $p(\text{баннер}|t)$, $p(\text{элемент\_изображения}|t)$, $p(\text{пользователь}|\text{t}), \ldots$



Мультимодальная ARTM

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

Максимизация суммы log правдоподобий с регуляризацией:

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

EM-алгоритм: метод простой итерации для системы уравнений

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

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

BigARTM: библиотека тематического моделирования

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 (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 упрощает разработку тематических моделей

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

Этапы моделирования	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	

-- нестандартизуемые этапы, уникальная разработка для каждой задачи

-- стандартизуемые этапы

Классические модели PLSA и LDA

PLSA: probabilistic latent semantic analysis [Hofmann, 1999] (вероятностный латентный семантический анализ):

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

M-шаг — частотные оценки условных вероятностей:

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

LDA: latent Dirichlet allocation (латентное размещение Дирихле):

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

M-шаг — сглаженные частотные оценки с параметрами β_w, α_t:

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

Hofmann T. Probabilistic latent semantic indexing. SIGIR 1999.
Blei D., Ng A., Jordan M. Latent Dirichlet allocation. 2003.

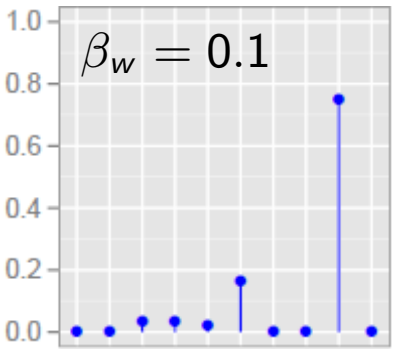
Вероятностная байесовская интерпретация LDA [Blei, 2003]

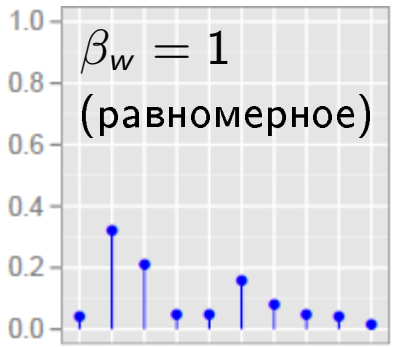
Гипотеза. Вектор-столбцы $\phi_t = (\phi_{wt})_{w \in W}$ и $\theta_d = (\theta_{td})_{t \in T}$ порождаются распределениями Дирихле, $\alpha \in \mathbb{R}^{|T|}$, $\beta \in \mathbb{R}^{|W|}$:

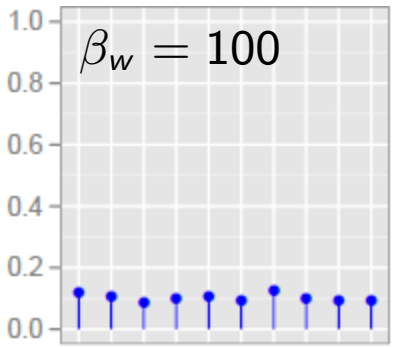
$$\mathrm{Dir}(\phi_t|\beta) = \frac{\Gamma(\beta_0)}{\prod_w \Gamma(\beta_w)} \prod_w \phi_{wt}^{\beta_w - 1}, \quad \phi_{wt} > 0; \quad \beta_0 = \sum_w \beta_w, \ \beta_t > 0;$$

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

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







Максимизация апостериорной вероятности для модели LDA

Совместное правдоподобие данных и модели:

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

Регуляризатор — логарифм априорного распределения:

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

M-шаг — сглаженные или слабо разреженные оценки:

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

при $\beta_w > 1$, $\alpha_t > 1$ — сглаживание,
при $0 < \beta_w < 1$, $0 < \alpha_t < 0$ — слабое разреживание,
при $\beta_w = 1$, $\alpha_t = 1$ априорное распределение равномерно, PLSA.

Почему именно распределение Дирихле?

Плюсы:

- удобно для байесовского вывода, т. к. является сопряжённым к мультиномиальному распределению
- описывает широкий класс распределений на симплексе
- позволяет управлять разреженностью ϕ_{wt} и θ_{td}
- при малых n_{wt}, n_{td} уменьшает переобучение

Минусы:

- не имеет лингвистических обоснований
- не даёт выигрыша против PLSA на больших коллекциях
- слабый разреживатель: запрещены $\beta_w \leqslant 0$, $\alpha_t \leqslant 0$
- слабый регуляризатор: проблема неединственности остаётся

Обобщённая не-байесовская интерпретация LDA

Сглаживание распределений по KL-дивергенции:
приблизить $\phi_{wt} \equiv p(w|t)$ к заданным распределениям $\beta_t(w)$,
приблизить $\theta_{td} \equiv p(t|d)$ к заданным распределениям $\alpha_d(t)$:

$$\sum_{t\in T} \tau_t \mathrm{KL}\big(\beta_t(w)\|\phi_{wt}\big) \to \min_{\Phi}; \qquad \sum_{d\in D} \tau_d \mathrm{KL}\big(\alpha_d(t)\|\theta_{td}\big) \to \min_{\Theta}.$$

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:

$$\phi_{wt} = \underset{w}{\mathrm{norm}}\big(n_{wt} + \underbrace{\tau_t\beta_t(w)}_{\beta_{wt}}\big), \qquad \theta_{td} = \underset{t}{\mathrm{norm}}\big(n_{td} + \underbrace{\tau_d\alpha_d(t)}_{\alpha_{td}}\big).$$

Сглаживание, разреживание и частичное обучение тем

Формулы M-шага:

$$\phi_{wt} = \underset{w}{\text{norm}}\big(n_{wt} + \beta_{wt}\big), \qquad \theta_{td} = \underset{t}{\text{norm}}\big(n_{td} + \alpha_{td}\big).$$

Разреживание и сглаживание описывается общей формулой:

- разреживание — максимизация KL, $\beta_{wt} < 0$, $\alpha_{td} < 0$
- сглаживание — минимизация KL, $\beta_{wt} > 0$, $\alpha_{td} > 0$

Частичное обучение темы t:

- $\beta_{wt} = +\tau_{\text{бт}}[w \in W_t]$ — «белый список» терминов
- $\beta_{wt} = -\tau_{\text{чт}}[w \in W_t]$ — «чёрный список» терминов
- $\alpha_{td} = +\tau_{\text{бд}}[d \in D_t]$ — «белый список» документов
- $\alpha_{td} = -\tau_{\text{чд}}[d \in D_t]$ — «чёрный список» документов

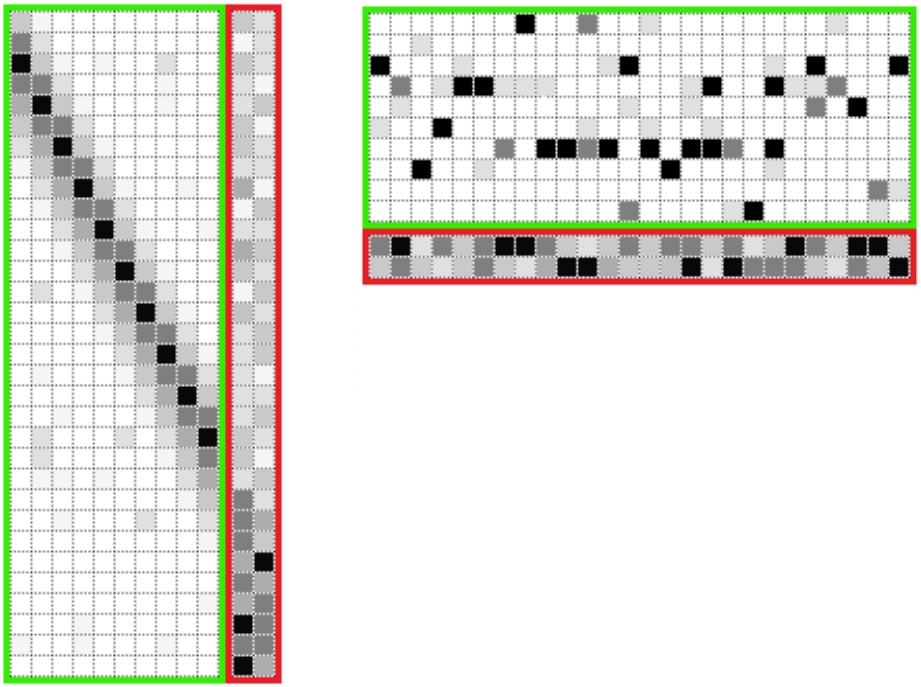
Разделение тем на предметные и фоновые

$T = S \sqcup B$ — множество всех тем

S — разреженные *предметные* темы, специальная лексика

B — сглаженные *фоновые* темы, общая лексика языка

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



Регуляризатор декоррелирования тем

Цель — выделить *лексическое ядро* каждой темы, набор терминов, отличающий её от других тем.

Минимизируем ковариации между вектор-столбцами ϕ_t, ϕ_s:

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

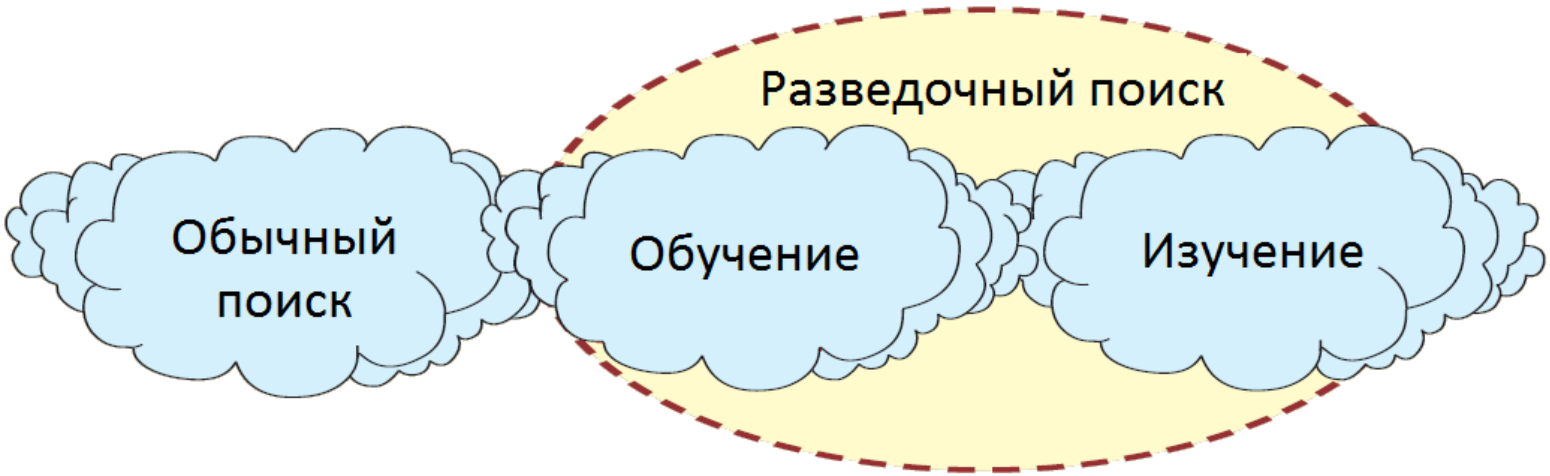
Подставляем, получаем ещё один вариант разреживания — постепенное контрастирование строк матрицы Φ:

$$\phi_{wt} = \underset{w \in W}{\operatorname{norm}} \left(n_{wt} - \tau \phi_{wt} \sum_{s \in T \setminus t} \phi_{ws} \right).$$

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

Концепция разведочного поиска (exploratory search)

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



навигация в сети,	самообразование,	исследование,
поиск фактов,	тематический поиск	экспертиза,
упоминаний,	систематизация	реферирование,
конкретных ответов	знаний	мониторинг тем

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

Разведочный тематический поиск

$q = (w_1, \ldots, w_{n_q})$ — 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:

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

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

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 pymorphy2

Методика оценивания качества разведочного поиска

Поисковый запрос

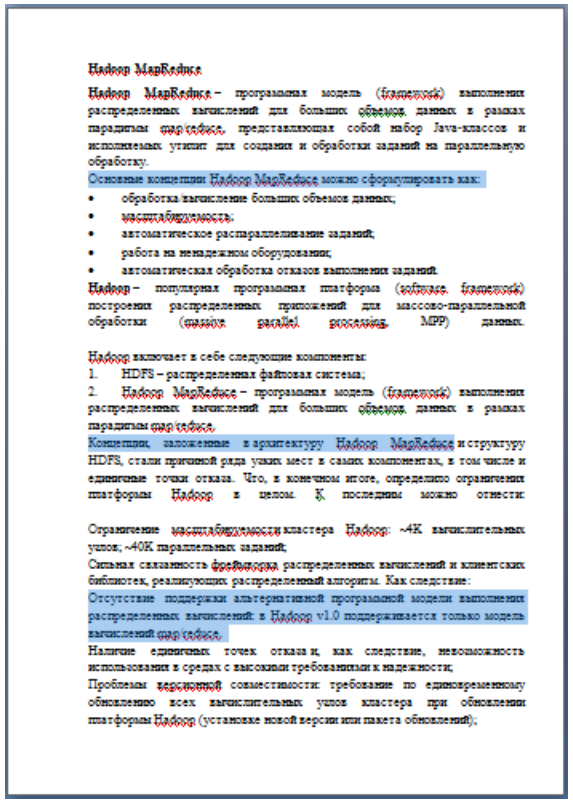
набор ключевых слов или фрагментов текста, около одной страницы A4

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



Hadoop MapReduce

Hadoop MapReduce – программная модель (framework) выполнения распределенных вычислений для больших объемов данных в рамках парадигмы map/reduce, представляющая собой набор Java-классов и исполнимых утилит для создания и обработки заданий на параллельную обработку.

Основные концепции Hadoop MapReduce можно сформулировать как:

- обработка/вычисление больших объемов данных;
- масштабируемость;
- автоматическое распараллеливание заданий;
- работа на ненадежном оборудовании;
- автоматическая обработка отказов выполнения заданий.

Hadoop – популярная программная платформа (software framework) построения распределенных приложений для массово-параллельной обработки (massive parallel processing, MPP) данных.

Hadoop включает в себе следующие компоненты:

1. HDFS – распределенная файловая система;
2. Hadoop MapReduce – программная модель (framework) выполнения распределенных вычислений для больших объемов данных в рамках парадигмы map/reduce.

Концепции, заложенные в архитектуру Hadoop MapReduce и структуру HDFS, стали причиной ряда узких мест в самих компонентах, в том числе и единичные точки отказа. Что, в конечном итоге, определило ограничения платформы Hadoop в целом. К последним можно отнести:

Ограничение масштабируемости кластера Hadoop: ~4K вычислительных узлов; ~40K параллельных заданий;

Сильная связанность фреймворка распределенных вычислений и клиентских библиотек, реализующих распределенный алгоритм. Как следствие:

Отсутствие поддержки альтернативной программной модели выполнения распределенных вычислений: в Hadoop v1.0 поддерживается только модель вычислений map/reduce.

Наличие единичных точек отказа и, как следствие, невозможность использования в средах с высокими требованиями к надежности;

Проблемы версионной совместимости: требование по единовременному обновлению всех вычислительных узлов кластера при обновлении платформы Hadoop (установке 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.

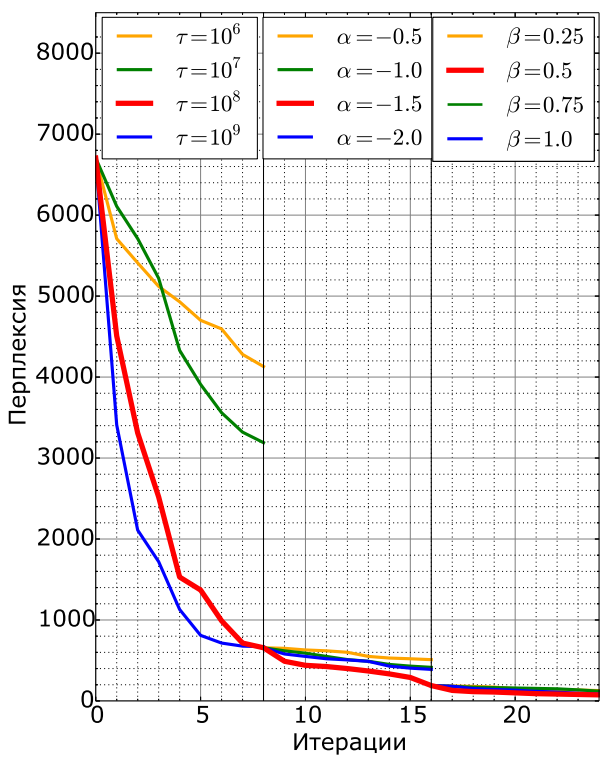
Тематика запросов разведочного поиска

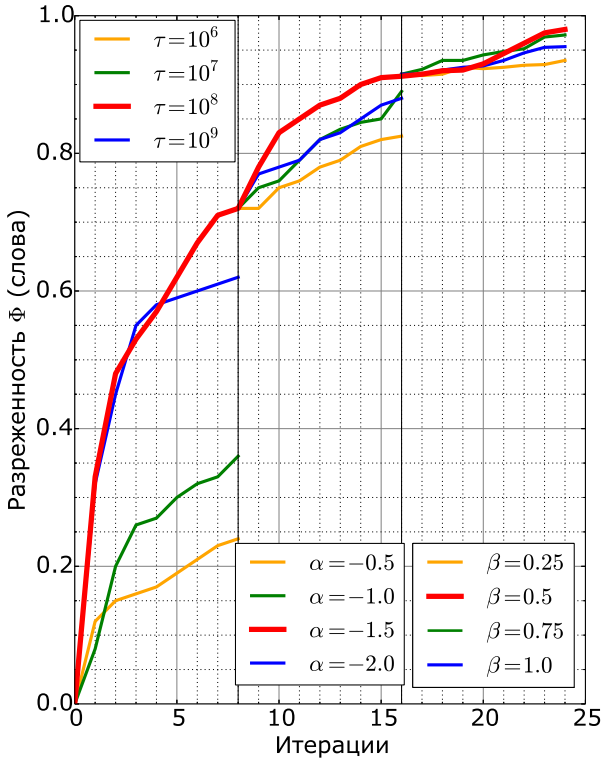
Примеры заголовков разведочных запросов к Хабру (объём каждого запроса — около одной страницы A4):

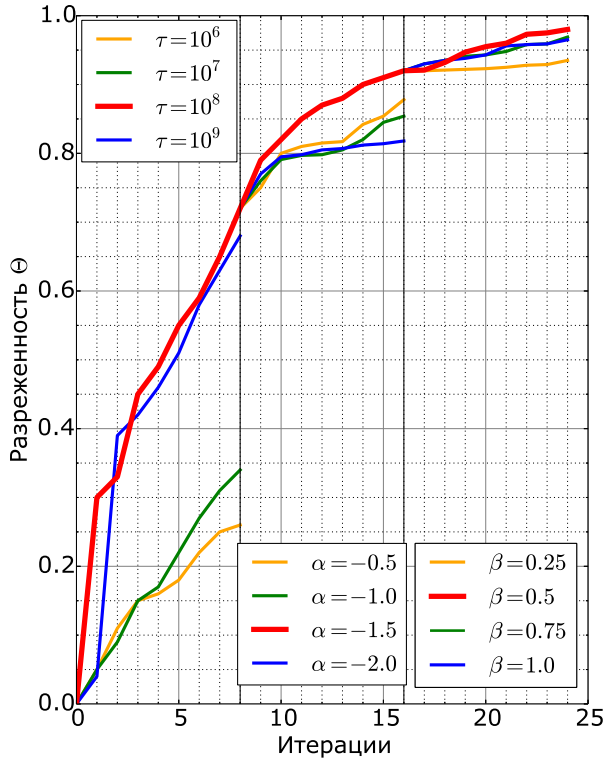
Алгоритмы раскраски графов	Система IBM Watson
Рекомендательная система Netflix	3D-принтеры
Методики быстрого набора текста	CERN-кластер
Космические проекты Илона Маска	AB-тестирование
Технологии Hadoop MapReduce	Облачные сервисы
Беспилотный автомобиль Google car	Контекстная реклама
Криптосистемы с открытым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

Последовательный подбор коэффициентов регуляризации

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







Оценки качества поиска

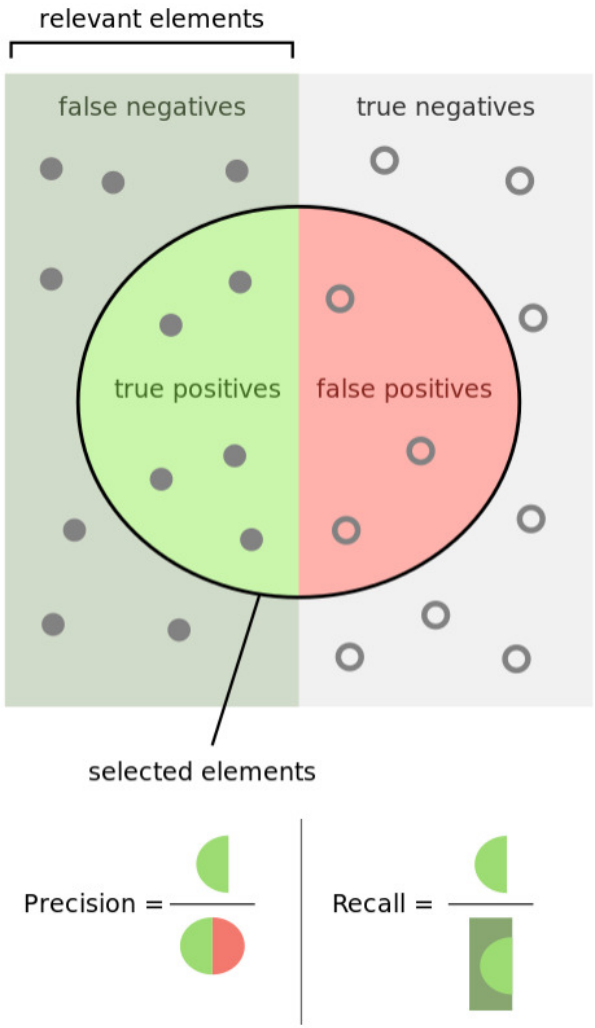
Precision — доля релевантных среди найденных
Recall — доля найденных среди релевантных

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

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

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

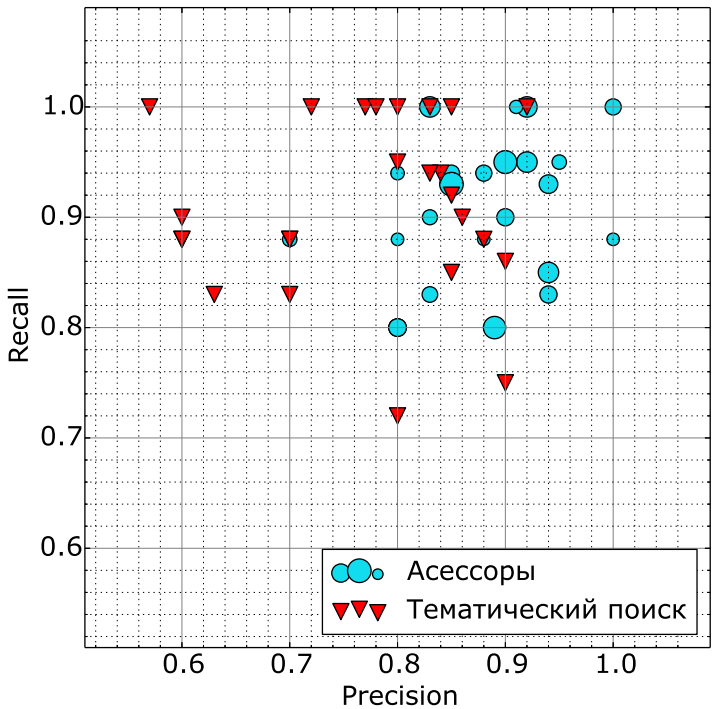
TP (true positive) — найденные релевантные
FP (false positive) — найденные нерелевантные
FN (false negative) — ненайденные релевантные



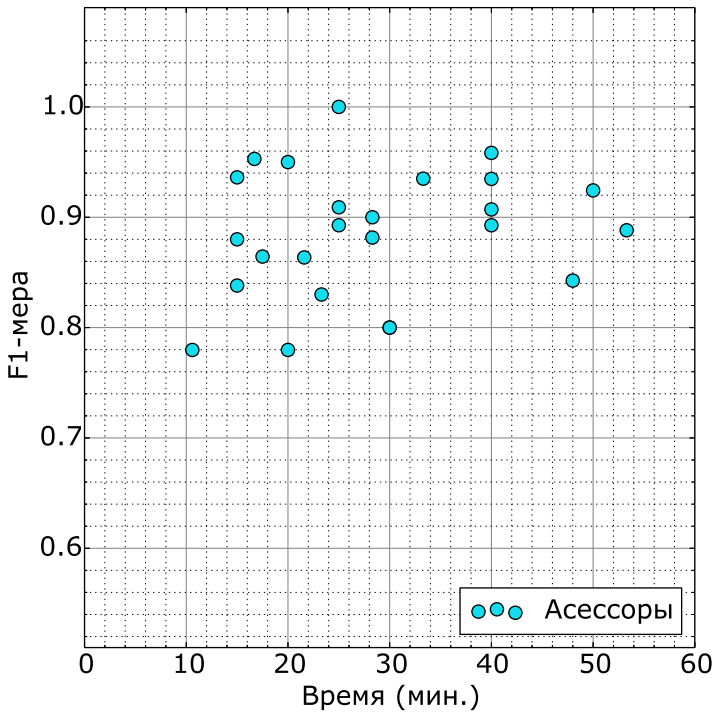
Результаты измерения точности и полноты по запросам

25 запросов, 3 асессора на запрос

точность и полнота поиска



время и F_1-мера (асессоры)



- среднее время обработки запроса асессором — 30 минут
- точность выше у асессоров, полнота — у поисковика

Выбор модальностей по критериям точности и полноты

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

	acессоры	C	K	TX	CT	CX	CTX	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 — слова и теги

Выбор модальностей по критериям точности и полноты

TechCrunch. Число тем $|T| = 450$.
Модальности: Слова, Биграммы, Категории, Авторы.

	асессоры	С	К	СБ	СБК	СБКА
Precision@5	0.83	0.71	0.55	0.77	0.80	**0.80**
Precision@10	0.88	0.72	0.58	0.78	0.81	**0.81**
Precision@15	0.87	0.73	0.59	0.79	0.83	**0.83**
Precision@20	0.86	0.72	0.56	0.77	0.82	**0.82**
Recall@5	0.81	0.75	0.65	0.77	0.82	**0.83**
Recall@10	0.85	0.77	0.66	0.80	0.85	**0.86**
Recall@15	0.89	0.78	0.68	0.82	0.87	**0.91**
Recall@20	0.90	0.82	0.69	0.83	0.89	**0.93**

- Наилучшее качество поиска — по всем модальностям
- Наиболее полезные модальности — слова и категории

Выбор числа тем по критериям точности и полноты

Habrahabr. Используем все 5 модальностей, меняем $|T|$

	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

Выбор числа тем по критериям точности и полноты

TechCrunch. Используем все 4 модальности, меняем $|T|$

	асессоры	350	400	450	475	500
Precision@5	0.83	0.65	0.72	0.75	**0.80**	0.68
Precision@10	0.88	0.66	0.73	0.76	**0.81**	0.69
Precision@15	0.87	0.68	0.74	0.78	**0.82**	0.68
Precision@20	0.86	0.65	0.74	0.77	**0.81**	0.67
Recall@5	0.81	0.65	0.75	0.78	**0.83**	0.79
Recall@10	0.85	0.66	0.78	0.79	**0.86**	0.80
Recall@15	0.89	0.68	0.79	0.79	**0.91**	0.83
Recall@20	0.90	0.69	0.79	0.80	**0.93**	0.85

- Наилучшее качество поиска — при 475 темах
- Тематический поиск превосходит асессоров по полноте

Выбор меры близости документа и запроса

Меры близости распределений:
Euclidean, Cosine, Manhattan, Kullback-Leibler

	Habrahabr, $\|T\| = 200$				*TechCrunch*, $\|T\| = 450$			
	E	C	M	KL	E	C	M	KL
Precision@5	0.61	**0.74**	0.68	0.72	0.63	**0.80**	0.67	0.71
Precision@10	0.65	**0.77**	0.69	0.75	0.66	**0.81**	0.68	0.73
Precision@15	0.62	**0.68**	0.63	0.70	0.64	**0.82**	0.64	0.72
Precision@20	0.62	**0.68**	0.62	0.70	0.64	**0.81**	0.63	0.71
Recall@5	0.67	**0.82**	0.69	0.80	0.66	**0.83**	0.67	0.77
Recall@10	0.68	**0.88**	0.70	0.85	0.67	**0.86**	0.68	0.78
Recall@15	0.70	**0.90**	0.72	0.87	0.71	**0.91**	0.70	0.80
Recall@20	0.70	**0.91**	0.73	0.88	0.71	**0.93**	0.71	0.81

- Наилучшее качество поиска — при косинусной мере

Все ли регуляризаторы были нужны?

Декоррелирование, Разреживание

	Habrahabr					*TechCrunch*				
	асессоры	все	ДР	Д	нет	асессоры	все	ДР	Д	нет
Precision@5	0.82	**0.74**	0.69	0.58	0.52	0.83	**0.80**	0.71	0.57	0.54
Precision@10	0.87	**0.77**	0.70	0.59	0.55	0.88	**0.81**	0.72	0.59	0.55
Precision@15	0.86	**0.68**	0.65	0.56	0.53	0.87	**0.82**	0.68	0.58	0.54
Precision@20	0.85	**0.68**	0.65	0.55	0.52	0.86	**0.81**	0.68	0.58	0.54
Recall@5	0.78	**0.82**	0.75	0.63	0.59	0.81	**0.81**	0.76	0.65	0.60
Recall@10	0.84	**0.88**	0.76	0.65	0.60	0.85	**0.86**	0.78	0.66	0.62
Recall@15	0.88	**0.90**	0.77	0.66	0.61	0.89	**0.89**	0.81	0.64	0.63
Recall@20	0.88	**0.91**	0.77	0.66	0.61	0.90	**0.92**	0.82	0.64	0.63

- Все регуляризаторы необходимы

Сравнение с поиском по векторам TF-IDF слов

Поиск по векторам TF-IDF$(w|d) = \frac{n_{dw}}{\ln N_w}$

	Habrahabr			*TechCrunch*		
	аксессоры	topic	tf-idf	assessors	topic	tf-idf
Precision@5	0.82	**0.74**	0.76	0.83	**0.80**	0.78
Precision@10	0.87	**0.77**	0.77	0.88	**0.81**	0.79
Precision@15	0.86	**0.68**	0.72	0.87	**0.82**	0.76
Precision@20	0.85	**0.68**	0.71	0.86	**0.81**	0.75
Recall@5	0.78	**0.82**	0.76	0.81	**0.81**	0.77
Recall@10	0.84	**0.88**	0.77	0.85	**0.86**	0.78
Recall@15	0.88	**0.90**	0.80	0.89	**0.89**	0.80
Recall@20	0.88	**0.91**	0.81	0.90	**0.92**	0.83

- Тематический поиск немного лучше TF-IDF
- При этом поисковый индекс на 2–3 порядка компактнее

Янина А. О., Воронцов К. В. Мультимодальные тематические модели для разведочного поиска в коллективном блоге. JMLDA, 2016.

Открытые проблемы для исследований

1. Вычислительно эффективные тесты адекватности TM
2. Понимание структуры множества решений Φ, Θ
3. Сходимость EM-ARTM с функциями потерь $\ell \neq \ln$
4. EM-ARTM при распределённом хранении коллекции
5. Стратегии и адаптивные траектории регуляризации
6. Симбиозы EM-алгоритма с градиентными методами
7. Почему эти регуляризаторы улучшают качество поиска?

Резюме

- 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
- В следующей лекции: регуляризаторы для всего-всего