

Вероятностные тематические модели

Лекция 2. Регуляризаторы и разведочный информационный поиск

К. В. Воронцов
vokov@forecsys.ru

Этот курс доступен на странице вики-ресурса
<http://www.MachineLearning.ru/wiki>
«Вероятностные тематические модели (курс лекций, К.В.Воронцов)»

МФТИ – ФИЦ ИУ РАН • 11 сентября 2019

- 1 Разведочный информационный поиск**
 - Концепция разведочного поиска
 - Особенности разведочного поиска
 - Разведочный поиск как рекомендательная система
- 2 Регуляризаторы, модальности, иерархии**
 - Разреживание, сглаживание, декоррелирование
 - Модальности
 - Иерархические тематические модели
- 3 Эксперименты с тематическим поиском**
 - Методика измерения качества поиска
 - Тематическая модель для документного поиска
 - Оптимизация гиперпараметров

Напоминания. Задача тематического моделирования

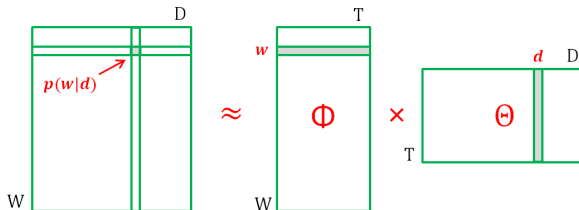
Дано: коллекция текстовых документов, $p(w|d) = \frac{n_{dw}}{n_d}$

Вероятностная тематическая модель:

$$p(w|d) = \sum_{t \in T} p(w|t)p(t|d) = \sum_{t \in T} \phi_{wt}\theta_{td}$$

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

Это задача стохастического матричного разложения:



Максимизация \log правдоподобия с регуляризатором R :

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

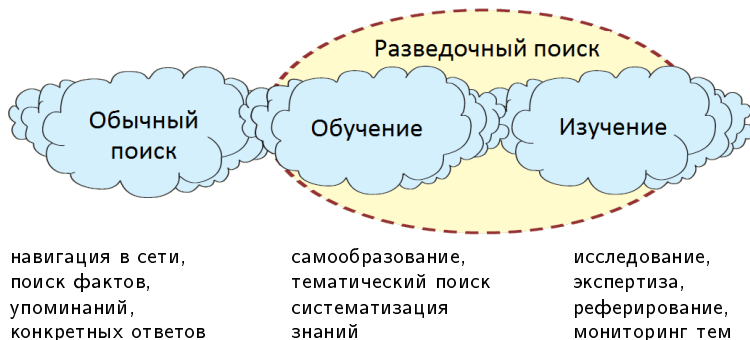
EM-алгоритм: метод простой итерации для системы уравнений

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

где $\mathop{\text{norm}}_{t \in T}(x_t) = \frac{\max\{x_t, 0\}}{\sum_{s \in T} \max\{x_s, 0\}}$ — операция нормирования вектора.

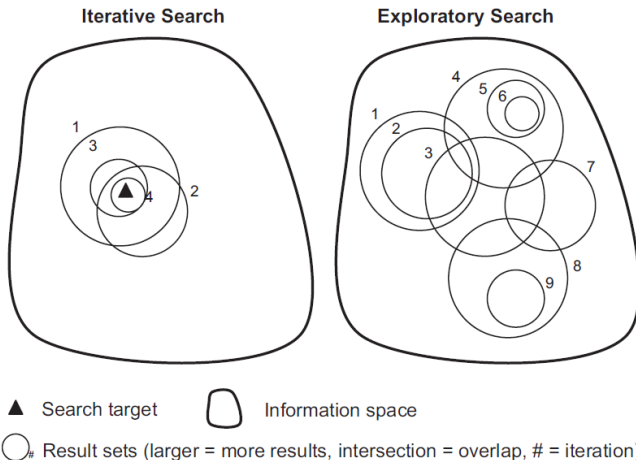
Концепция разведочного поиска (exploratory search)

- пользователь может не знать ключевых терминов
- запросом может быть текст произвольной длины
- информационная потребность — систематизация знаний



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

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



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

Концепция «мастерской знаний»

Огромное и все возрастающее богатство знаний разбросано сегодня по всему миру. Этих знаний, вероятно, было бы достаточно для решения всего громадного количества трудностей наших дней, но они рассеяны и неорганизованы. Нам необходима очистка мышления в *своеобразной мастерской*, где можно **получать, сортировать, суммировать, усваивать, разъяснять и сравнивать** знания и идеи.

— Герберт Уэллс, 1940

Разведочный поиск — это процесс

- **An evolving search process**
разведочный поиск – это многошаговый процесс
каждый шаг – переформулировка или дополнение запроса
- **An anomalous state of knowledge**
в начале поиска у пользователя есть лишь мотивации,
но нет знаний и нет определённого плана, как их получать
- **Multiple targets / goals of search**
нет конкретной, точно определённой цели поиска
есть лишь общий интерес и эволюционирующие подцели

Неопределённости в процессе поиска

- **Multiple possible answers**
возможных правильных ответов может быть много
- **Not an expected exact answer**
не существует единственного правильного ответа
- **A serendipitous attitude**
любой шаг может давать неожиданные новые знания
- **An evolving information need**
на любом шаге цели и стратегии поиска могут измениться
- **Uncertainty is fluctuating**
в процессе поиска неопределённость уменьшается,
но изменение цели может снова её увеличить

E.Palagi et al. A Survey of Definitions and Models of Exploratory Search. 2017.

Разветвлённость процесса поиска

- **Multifaceted search**
при поиске используются различные фильтры (фасеты)
примеры: по авторам, тематике, свежести, сложности
- **Several one-off pinpoint searches**
многократные точечные одноразовые ответвления поиска
примеры: найти определение понятия, первоисточник
- **An open-ended search activity which can occur over time**
процесс поиска никогда не заканчивается
пользователь может вернуться после долгого перерыва

Поисковая система arXiv.AITHEA.com

Пользователь собирает тематические подборки документов

The screenshot shows a web browser window displaying the arXiv.AITHEA.com website. The URL in the address bar is <https://arxiv.aithea.com/collections/Q29sbGVjJdGlvbjozUFVTUEFxaHBH>. The website has a navigation bar with 'FEEDS', 'SEARCH', and 'COLLECTIONS' tabs. The 'COLLECTIONS' tab is active, and the page title is 'Topic Modeling for Opinion Mining'. Below the title, there are two tabs: 'PAPERS' (selected) and 'RECOMMENDED'. The page displays a list of papers. The first paper is titled 'Comparative Opinion Mining: A Review' by Kasturi Dewi Varathan, Anastasia Giachanou, and Fabio Crestani, dated 24 DEC 2017. It has 6 citations and includes icons for bookmarking, liking, and sharing. The second paper is titled 'The survey of sentiment and opinion mining for behavior analysis of social media' by Saqib Iqbal, Ali Zulqurnain, Yaqoob Wani, and Khalid Hussain, dated 7 NOV 2015. It has 0 citations and also includes icons for bookmarking, liking, and sharing.

Поиск тематически близких документов

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

$\theta_{td} = p(t|d)$ — тематические векторы 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: *векторный индекс* для быстрого поиска документов d по каждой из тем t запроса

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

Дивергенция Кульбака–Лейблера и её свойства

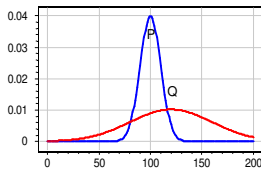
Функция расстояния между распределениями $P = (p_i)_{i=1}^n$ и $Q = (q_i)_{i=1}^n$:

$$KL(P\|Q) \equiv KL_i(p_i\|q_i) = \sum_{i=1}^n p_i \ln \frac{p_i}{q_i}.$$

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

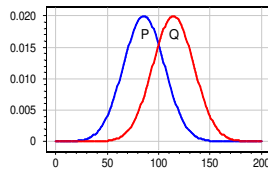
$$KL(P\|Q(\alpha)) = \sum_{i=1}^n p_i \ln \frac{p_i}{q_i(\alpha)} \rightarrow \min_{\alpha} \iff \sum_{i=1}^n p_i \ln q_i(\alpha) \rightarrow \max_{\alpha}.$$

3. Если $KL(P\|Q) < KL(Q\|P)$, то P сильнее вложено в Q , чем Q в P :



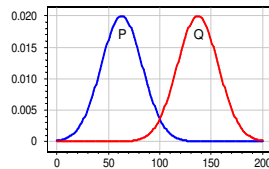
$$KL(P\|Q) = 0.44$$

$$KL(Q\|P) = 2.97$$



$$KL(P\|Q) = 0.44$$

$$KL(Q\|P) = 0.44$$



$$KL(P\|Q) = 2.97$$

$$KL(Q\|P) = 0.44$$

Регуляризатор сглаживания

Гипотеза сглаженности:

распределения ϕ_{wt} близки к заданному распределению β_w ;
распределения θ_{td} близки к заданному распределению α_t .

$$\sum_{t \in T} \text{KL}(\beta_w \| \phi_{wt}) \rightarrow \min_{\Phi}; \quad \sum_{d \in D} \text{KL}(\alpha_t \| \theta_{td}) \rightarrow \min_{\Theta}.$$

Максимизируем сумму регуляризаторов:

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

Подставляем, получаем формулы M-шага, похожие на LDA:

$$\phi_{wt} = \text{norm}_{w \in W}(n_{wt} + \beta_0 \beta_w), \quad \theta_{td} = \text{norm}_{t \in T}(n_{td} + \alpha_0 \alpha_t).$$

Blei D., Ng A., Jordan M. Latent Dirichlet Allocation. JMLR, 2003.

Регуляризатор разреживания

Гипотеза разреженности: среди ϕ_{wt} , θ_{td} много нулей;
распределения ϕ_{wt} **далеки** от заданного распределения β_w ;
распределения θ_{td} **далеки** от заданного распределения α_t .

$$\sum_{t \in T} \text{KL}(\beta_w \| \phi_{wt}) \rightarrow \max_{\Phi}; \quad \sum_{d \in D} \text{KL}(\alpha_t \| \theta_{td}) \rightarrow \max_{\Theta}.$$

Максимизируем сумму регуляризаторов:

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

Получаем «анти-LDA» (в LDA все $\alpha_0, \alpha_t, \beta_0, \beta_t$ положительны):

$$\phi_{wt} = \text{norm}_{w \in W}(n_{wt} - \beta_0 \beta_w), \quad \theta_{td} = \text{norm}_{t \in T}(n_{td} - \alpha_0 \alpha_t).$$

Varadarajan J., Emonet R., Odobez J.-M. A sparsity constraint for topic models — application to temporal activity mining. NIPS-2010.

Объединение сглаживания и разреживания

Общий вид регуляризаторов сглаживания и разреживания:

$$R(\Phi, \Theta) = \beta_0 \sum_{t \in T} \sum_{w \in W} \beta_{wt} \ln \phi_{wt} + \alpha_0 \sum_{d \in D} \sum_{t \in T} \alpha_{td} \ln \theta_{td} \rightarrow \max,$$

где $\beta_0 > 0$, $\alpha_0 > 0$ — коэффициенты регуляризации,

β_{wt} , α_{td} — параметры, задаваемые пользователем:

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

Возможные применения сглаживания и разреживания:

- задать фоновые темы с общей лексикой языка
- задать шумовую тему для нетематичных термов
- задать псевдо-документ с ключевыми термами темы
- скорректировать состав термов и документов темы

Частичное обучение (semi-supervised learning)

Общий вид регуляризаторов сглаживания и разреживания:

$$R(\Phi, \Theta) = \beta_0 \sum_{t \in T} \sum_{w \in W} \beta_{wt} \ln \phi_{wt} + \alpha_0 \sum_{d \in D} \sum_{t \in T} \alpha_{td} \ln \theta_{td} \rightarrow \max,$$

Идея: в построенной модели можно скорректировать темы, добавляя и удаляя в них термы и документы.

Разреживание по «чёрным спискам»:

- $\beta_{wt} = -\frac{1}{|W_t|} [w \in W_t]$ — термов из W_t не должно быть в t
- $\alpha_{td} = -\frac{1}{|T_d|} [t \in T_d]$ — тем из T_d не должно быть в d

Сглаживание по «белым спискам»:

- $\beta_{wt} = \frac{1}{|W_t|} [w \in W_t]$ — термы из W_t должны быть в t
- $\alpha_{td} = \frac{1}{|T_d|} [t \in T_d]$ — темы из T_d должны быть в d

Проблема $\ln 0$ в дивергенции Кульбака–Лейблера

Почему в регуляризаторе сглаживания/разреживания

$$R(\Phi) = \beta_0 \sum_{t \in S} \sum_{w \in W} \beta_w \ln \phi_{wt} \rightarrow \max$$

не возникает проблем с $\ln \phi_{wt}$ при $\phi_{wt} \rightarrow 0$?

Подправим регуляризатор, при сколь угодно малом ε :

$$R(\Phi) = \beta_0 \sum_{t \in S} \sum_{w \in W} \beta_w \ln(\phi_{wt} + \varepsilon) \rightarrow \max.$$

Подставив в формулу M-шага, получим для всех $t \in S$:

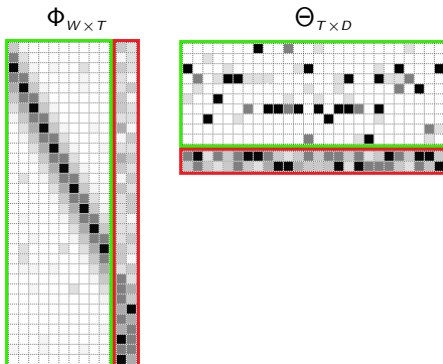
$$\phi_{wt} = \operatorname{norm}_{w \in W} \left(n_{wt} + \beta_0 \beta_w \frac{\phi_{wt}}{\phi_{wt} + \varepsilon} \right).$$

Если $\phi_{wt} = 0$, то разреживания не будет, но оно и не нужно.

Разделение тем на предметные и фоновые

Предметные темы S содержат термины предметной области,
 $p(w|t)$, $p(t|d)$, $t \in S$ — разреженные, существенно различные

Фоновые темы B содержат слова общей лексики,
 $p(w|t)$, $p(t|d)$, $t \in B$ — существенно отличные от нуля



Регуляризатор декоррелирования тем

Цель: сделать темы как можно более различными, выделить для каждой темы *лексическое ядро* — набор термов, отличающий её от других тем.

Минимизируем ковариации между вектор-столбцами ϕ_t :

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

Подставляем, получаем ещё один вариант разреживания — постепенное контрастирование строк матрицы Φ (малые вероятности ϕ_{wt} в строке становятся ещё меньше):

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

Tan Y., Ou Z. Topic-weak-correlated latent Dirichlet allocation. 2010.

Напоминание. Мультимодальная ARTM

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

Максимизация суммы log-правдоподобий с регуляризацией:

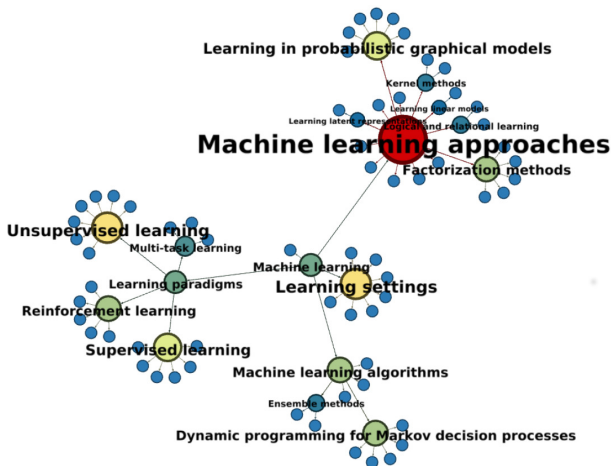
$$\sum_{m \in M} \tau_m \sum_{d \in D} \sum_{w \in W^m} n_{dw} \ln \sum_{t \in T} \phi_{wt} \theta_{td} + R(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta}$$

EM-алгоритм: метод простой итерации для системы уравнений

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

K.Vorontsov, O.Frei, M.Apishev et al. Non-bayesian additive regularization for multimodal topic modeling of large collections. CIKM TM workshop, 2015.

Пример древовидной тематической иерархии



Georgeta Bordea. Domain adaptive extraction of topical hierarchies for Expertise Mining. 2013.

Пример тематической иерархии

Тексты научно-просветительского ресурса Postnauka.ru:
2976 документов, 43196 слов, 1799 тэгов



Chirkova N.A., Vorontsov K.V. Additive regularization for hierarchical multimodal topic modeling. JMLDA, 2016.

Belyy A.V., Seleznova M.S., Sholokhov A.K., Vorontsov K.V. Quality Evaluation and Improvement for Hierarchical Topic Modeling. Dialogue 2018.

Иерархические тематические модели

- структура иерархии: дерево / **многодольный граф**
- направление: снизу вверх / **сверху вниз** / одновременно
- наращивание: попершинное / **последное**

Открытые проблемы:

- “Despite recent activity in the field of HPTMs, determining the hierarchical model that best fits a given data set, in terms of the structure and size of the learned hierarchy, still remains a challenging task and an open issue.”
- “The evaluation of hierarchical PTMs is also an open issue.”

Zavitsanos E., Paliouras G., Vouros G. A. Non-Parametric Estimation of Topic Hierarchies from Texts with Hierarchical Dirichlet Processes. 2011.

Регуляризатор Φ : родительские темы как псевдо-документы

Шаг 1. Строим модель с небольшим числом тем.

Шаг k . Пусть модель с множеством тем T уже построена.
 Строим множество дочерних тем S (subtopics), $|S| > |T|$.

Родительские темы приближаются смесями дочерних тем:

$$\sum_{t \in T} n_t \text{KL}_w \left(p(w|t) \parallel \sum_{s \in S} p(w|s)p(s|t) \right) \rightarrow \min_{\Phi, \tilde{\Psi}}$$

где $\tilde{\Psi} = (\tilde{\psi}_{st})_{S \times T}$ — матрица связей, $\tilde{\psi}_{st} = p(s|t)$.

Родительская $\Phi^P \approx \Phi \tilde{\Psi}$, отсюда регуляризатор матрицы Φ :

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

Родительские темы t — «документы» с частотами термов n_{wt} .

Регуляризатор Θ : родительские темы как модальность

Шаг 1. Строим модель с небольшим числом тем.

Шаг k . Пусть модель с множеством тем T уже построена.
 Строим множество дочерних тем S (subtopics), $|S| > |T|$.

Родительские темы приближаются смесями дочерних тем:

$$\sum_{d \in D} n_d \text{KL}_t \left(p(t|d) \parallel \sum_{s \in S} p(t|s)p(s|d) \right) \rightarrow \min_{\Theta, \Psi}$$

где $\Psi = (\psi_{ts})_{T \times S}$ — матрица связей, $\psi_{ts} = p(t|s)$.

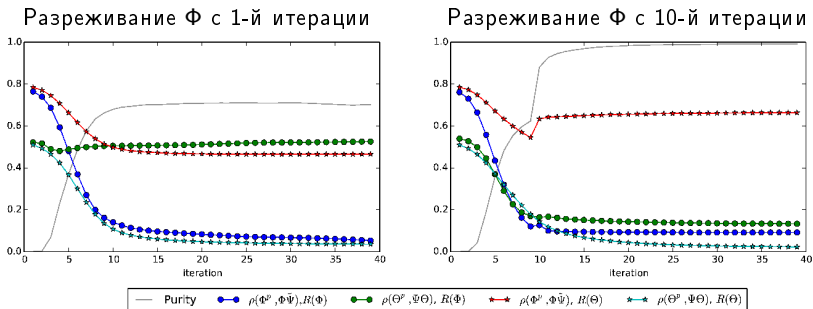
Родительская $\Theta^p \approx \Psi \Theta$, отсюда регуляризатор матрицы Θ :

$$R(\Theta, \Psi) = \tau \sum_{d \in D} \sum_{t \in T} n_{td} \ln \sum_{s \in S} \psi_{ts} \theta_{sd} \rightarrow \max.$$

Родительские темы t — модальность с частотами термов n_{td} .

Эксперимент на коллекции ММРО-ИОИ

Среднее расстояние Хеллингера $\rho(\Phi^P, \Phi\tilde{\Psi})$ и $\rho(\Theta^P, \Psi\Theta)$ для регуляризаторов Φ и Θ при переходе между уровнями $1 \rightarrow 2$:

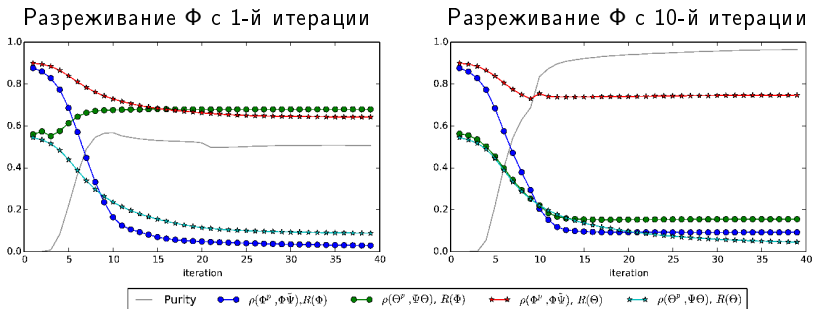


Вывод. Регуляризатор Θ плохо приближает Φ^P .

Chirkova N. A., Vorontsov K. V. Additive regularization for hierarchical multimodal topic modeling. JMLDA, 2016.

Эксперимент на коллекции ММРО-ИОИ

Среднее расстояние Хеллингера $\rho(\Phi^P, \Phi\tilde{\Psi})$ и $\rho(\Theta^P, \Psi\Theta)$ для регуляризаторов Φ и Θ при переходе между уровнями $2 \rightarrow 3$:



Вывод. Регуляризатор Θ плохо приближает Φ^P .

Chirkova N. A., Vorontsov K. V. Additive regularization for hierarchical multimodal topic modeling. JMLDA, 2016.

Выводы

- Регуляризатор Φ приближает $\Phi^P \approx \Phi\tilde{\Psi}$ и $\Theta^P \approx \Psi\Theta$
- Регуляризатор Θ приближает только $\Theta^P \approx \Psi\Theta$
- Сильное разреживание $\psi_{ts} \in \{0, 1\}$ даёт иерархию-дерево
- Нельзя допускать вырождения $\psi_{ts} = p(t|s) \equiv 0$

Дальнейшие задачи:

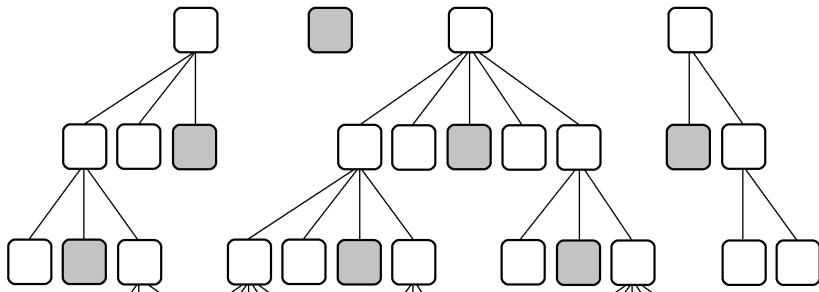
- Согласованная регуляризация: $\tilde{\psi}_{st}p(t) = \psi_{ts}p(s)$

$$\tau_1 \sum_{t,w} n_{wt} \ln \sum_s \phi_{ws} \psi_{ts} \frac{n_s}{n_t} + \tau_2 \sum_{d,t} n_{td} \ln \sum_s \psi_{ts} \theta_{sd} \rightarrow \max_{\Phi, \Psi, \Theta}$$

- Иерархии с темами различной глубины:
 - наращивание уровня для подмножества $T' \subseteq T$
 - критерий неоднородности темы для включения её в T'

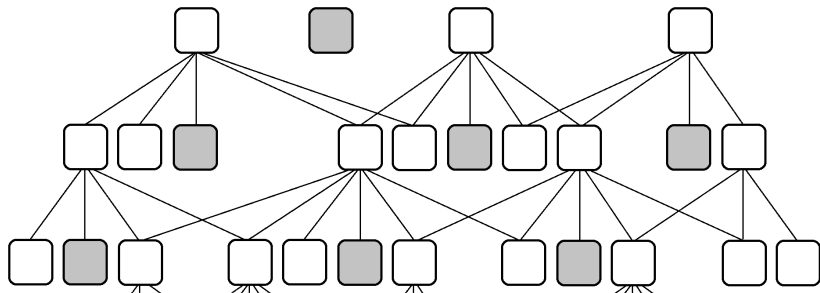
Иерархии с темами различной глубины

- На каждом уровне расщепляются не все темы (допускается вырожденность: $p(s|t) \equiv 0$ для некоторых t)
- Расщепляемая тема может иметь дочернюю фоновую, в которой собирается общая лексика родительской темы
- При максимальном разреживании $p(t|s) \in \{0, 1\}$ иерархия является деревом (корень не показан)



Иерархии с темами различной глубины

- На каждом уровне расщепляются не все темы (допускается вырожденность: $p(s|t) \equiv 0$ для некоторых t)
- Расщепляемая тема может иметь дочернюю фоновую, в которой собирается общая лексика родительской темы
- При умеренном разреживании $p(t|s)$ у вершины может быть несколько родителей (корень не показан)



Иерархии с темами различной глубины

След документа в тематической иерархии определяет степень его специализации, назначение, аудиторию



узко специализированный,
для профессионалов



междисциплинарное исследование,
для профессионалов



обзорный,
для ознакомления с предметной областью



популярный или энциклопедический,
для расширения кругозора

Способы оценивания качества тематических иерархий

- *Перплексия* или правдоподобие: приводит ли постепенное дробление тем к более точному разложению
- *Полезность*: сколько шагов делает пользователь, чтобы найти документ по иерархии
- *Когерентность*: как часто слова темы и её подтемы совместно встречаются рядом в тексте
- *Метод интрузий*: правильно ли ассессоры определяют чужую тему, внедрённую в список дочерних тем
- *Сравнение с «золотым стандартом»*: насколько иерархия похожа на имеющуюся категоризацию документов

Две коллекции новостей про технологии

Habrhabr.ru

175 143 статей на русском
10 552 слов (униграмм)
742 000 биграмм
524 авторов статей
10 000 авторов комментариев
2546 тегов
123 хаба (категории)

TechCrunch.com

759 324 статей на английском
11 523 слов (униграмм)
1.2 млн. биграмм
605 авторов
184 категорий

Предобработка текстов

- отброшены 5% наиболее частотных слов (общая лексика)
- удаление пунктуации
- нижний регистр, ё→е
- лемматизация r morphology2

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 ассессорам

- 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**) вычислений распределенных вычислений для больших объемов данных в рамках параллельных шардов, представляющих собой набор Java-классов и исполняемых единиц для создания и обработки данных по параллельной обработке.

Основные компоненты Поиска MapReduce можно сформулировать как:

- обработка вычислений больших объемов данных;
- масштабируемость;
- автоматическое распределение заданий;
- работа по минимальным обработкам;
- автоматическая обработка отказов вычислений заданий.

Поиск – популярная программная платформа (**библиотека библиотек**) построена распределенных приложений для массово-параллельной обработки (**разделов разбитых документов**, МР) данных.

Поиск включает в себе следующие компоненты:

1. HDFS – распределенная файловая система;

2. **Поиск MapReduce** – программная платформа (**библиотека библиотек**) вычислений распределенных вычислений для больших объемов данных в рамках параллельных шардов.

Ключевые особенности в архитектуре **Поиска MapReduce** и структуре HDFS, стали прорывной ролью умови в своем направлении, в том числе и в отношении точки отказа. Это, в конечном итоге, определило ограничение платформ **Поиск** в целом. К последним можно отнести:

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

Сильная связность **Фреймворка** распределенных вычислений и элементных вычислений распределенных алгоритмов. Как следствие:

Отсутствие поддержки альтернативной программы вычисления распределенных вычислений в **Поиск v1.0** поддерживается только модель вычислений шардов.

Модель вычислений точки отказа и как следствие, необходимость использования в среде с высокими требованиями к надежности;

Проблема **взаимосвязи** совместности требований по единичному объекту и всем вычислительным узлам кластера при обновлении 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

Оценивание качества поиска

Precision — доля релевантных среди найденных

Recall — доля найденных среди релевантных

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

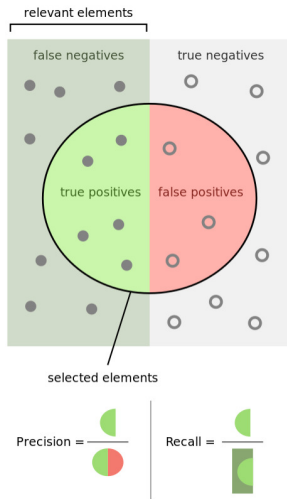
$$R = \frac{TP}{TP + FN} \text{ — полнота, (recall)}$$

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

TP (true positive) — найденные релевантные

FP (false positive) — найденные нерелевантные

FN (false negative) — ненайденные релевантные



Какие модели поиска сравнивались

- **assessors**: результаты поиска, выполненного ассессорами
- **TF-IDF, BM25**: сравнение документов по частотам слов
- **word2vec**: нетематические векторные представления слов
- **PLSA**: Probabilistic Latent Semantic Analysis (1999)
- **LDA**: Latent Dirichlet Allocation (2003)
- **ARTM**: тематическая модель с тремя регуляризаторами
- **hARTM**: двухуровневая иерархическая модель ARTM

Задачи регуляризаторов в ARTM и hARTM:

- сделать темы как можно более различными
- сделать векторы $p(t|d)$ как можно более разреженными
- не допустить вырожденности распределений $p(w|t)$

Стратегия регуляризации

Последовательное применение трёх регуляризаторов

- 1 декоррелирование тем:

$$R(\Phi) = -\tau \sum_{s,t \in T} \sum_{w \in W} \phi_{wt} \phi_{ws}$$

- 2 разреживание распределений $p(t|d)$:

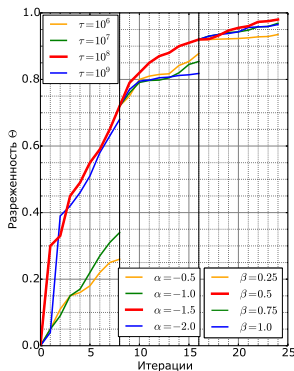
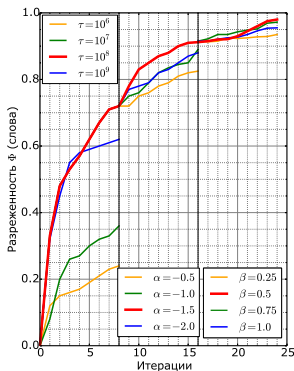
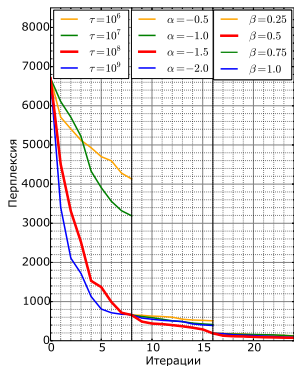
$$R(\Theta) = -\alpha \sum_{d,t} \ln \theta_{td}$$

- 3 сглаживание распределений $p(w|t)$:

$$R(\Phi) = \beta \sum_{t,w} \ln \phi_{wt}$$

Последовательный подбор коэффициентов регуляризации

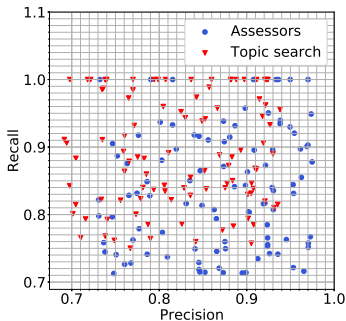
- декоррелирование распределений термов в темах (τ),
- разреживание распределений тем в документах (α),
- сглаживание распределений термов в темах (β).



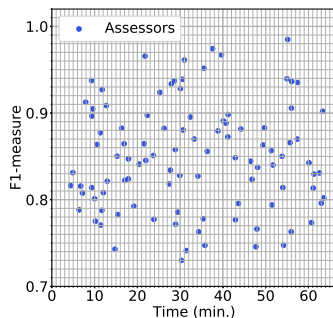
Результаты измерения точности и полноты по запросам

100 запросов, 3 ассессора на запрос

точность и полнота поиска



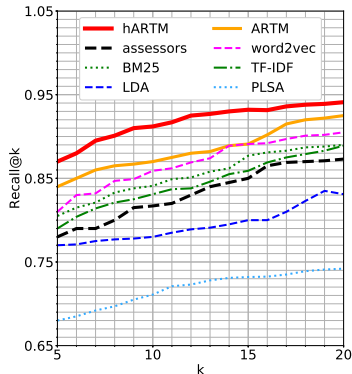
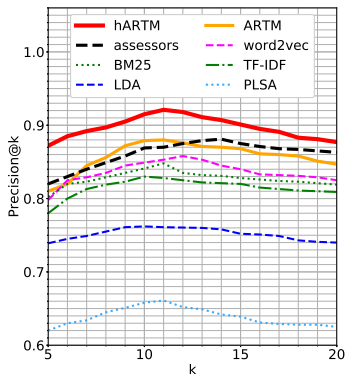
время и F_1 -мера (ассессоры)



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

Сравнение с ассессорами по качеству поиска

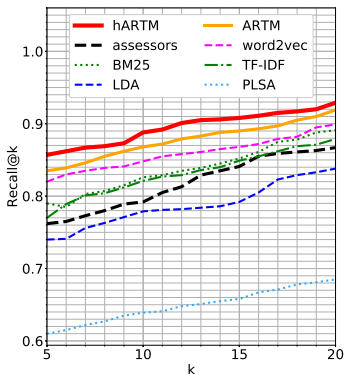
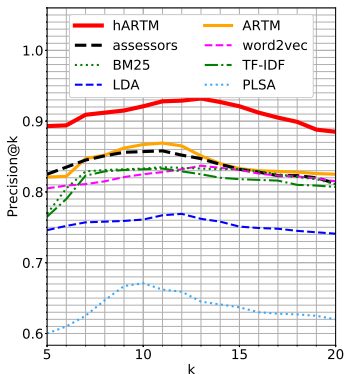
Точность и полнота по первым k позициям поисковой выдачи (коллекция Habrahabr.ru)



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

Сравнение с ассессорами по качеству поиска

Точность и полнота по первым k позициям поисковой выдачи (коллекция TechCrunch.com)



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

Влияние числа тем на качество поиска

Все регуляризаторы и модальности, **плоская модель**

	Habrahabr						TechCrunch					
	асесс	100	150	200	250	400	асесс	350	400	450	475	500
Pr@5	0.821	0.662	0.721	0.810	0.761	0.693	0.822	0.653	0.725	0.752	0.819	0.777
Pr@10	0.869	0.761	0.812	0.879	0.825	0.673	0.851	0.663	0.732	0.762	0.867	0.811
Pr@15	0.875	0.733	0.795	0.868	0.791	0.651	0.835	0.682	0.743	0.787	0.833	0.793
Pr@20	0.863	0.724	0.795	0.847	0.792	0.642	0.813	0.650	0.743	0.773	0.825	0.793
R@5	0.780	0.732	0.807	0.840	0.821	0.721	0.762	0.731	0.762	0.793	0.835	0.817
R@10	0.817	0.771	0.843	0.870	0.851	0.751	0.792	0.763	0.793	0.812	0.868	0.855
R@15	0.850	0.824	0.895	0.891	0.871	0.773	0.835	0.782	0.807	0.855	0.890	0.882
R@20	0.873	0.857	0.905	0.925	0.892	0.771	0.867	0.792	0.823	0.862	0.919	0.903

- существует оптимальное число тем
- чем больше коллекция, тем больше оптимум числа тем

Влияние числа тем на качество поиска

Nabrahabr. Все регуляризаторы и модальности, **три уровня**

$ T_1 $	20		25						30		
$ T_2 $	150	200	250		275		300		400	450	
$ T_3 $	750	800	1200	1300	1300	1400	1500	1500	1600	3000	3500
Pr@5	0.625	0.743	0.840	0.852	0.869	0.872	0.870	0.805	0.771	0.705	0.672
Pr@10	0.648	0.754	0.851	0.867	0.882	0.915	0.901	0.811	0.799	0.722	0.694
Pr@15	0.632	0.752	0.850	0.872	0.878	0.895	0.889	0.809	0.785	0.729	0.703
Pr@20	0.629	0.745	0.845	0.861	0.871	0.877	0.882	0.803	0.778	0.710	0.681
R@5	0.632	0.780	0.845	0.869	0.883	0.889	0.872	0.851	0.841	0.721	0.695
R@10	0.654	0.792	0.859	0.873	0.905	0.922	0.881	0.873	0.850	0.749	0.703
R@15	0.675	0.805	0.874	0.892	0.932	0.942	0.905	0.889	0.863	0.787	0.725
R@20	0.684	0.824	0.889	0.901	0.958	0.961	0.912	0.904	0.878	0.805	0.734

- существует оптимальное число тем на каждом уровне
- три уровня слегка лучше, чем два;
 оптимальное число тем увеличивается

Влияние числа тем на качество поиска

Nabrahabr. Все регуляризаторы и модальности, **два уровня**

$ T_1 $	20		25			30					
$ T_2 $	150	200	250	275	300	400	450				
Pr@5	0.621	0.742	0.839	0.850	0.865	0.869	0.869	0.803	0.769	0.701	0.670
Pr@10	0.645	0.749	0.850	0.861	0.879	0.911	0.895	0.809	0.796	0.719	0.689
Pr@15	0.635	0.751	0.848	0.869	0.873	0.893	0.887	0.807	0.781	0.721	0.701
Pr@20	0.630	0.745	0.841	0.855	0.864	0.874	0.875	0.800	0.775	0.709	0.675
R@5	0.628	0.773	0.843	0.865	0.881	0.881	0.868	0.849	0.839	0.715	0.691
R@10	0.652	0.782	0.855	0.871	0.902	0.918	0.877	0.871	0.845	0.745	0.699
R@15	0.671	0.801	0.870	0.889	0.929	0.939	0.901	0.883	0.861	0.781	0.722
R@20	0.680	0.819	0.886	0.892	0.955	0.955	0.907	0.901	0.872	0.801	0.729

- существует оптимальное число тем на каждом уровне
- два уровня лучше, чем один;
 при этом оптимальное число тем увеличивается

Влияние числа тем на качество поиска

TechCrunch. Все регуляризаторы и модальности, **три уровня**

$ T_1 $	80		100						120		
$ T_2 $	300	350	500		550		600		700	750	
$ T_3 $	1500	1700	2500	2600	2600	2800	3000	3000	3200	4500	4700
Pr@5	0.655	0.707	0.751	0.792	0.887	0.893	0.890	0.789	0.722	0.703	0.678
Pr@10	0.678	0.712	0.773	0.823	0.895	0.922	0.905	0.805	0.741	0.722	0.692
Pr@15	0.692	0.715	0.775	0.831	0.902	0.921	0.907	0.821	0.743	0.725	0.703
Pr@20	0.687	0.709	0.761	0.819	0.889	0.885	0.898	0.809	0.736	0.719	0.683
R@5	0.751	0.795	0.802	0.856	0.871	0.877	0.863	0.852	0.831	0.738	0.705
R@10	0.767	0.812	0.825	0.875	0.892	0.908	0.879	0.871	0.842	0.751	0.711
R@15	0.772	0.824	0.841	0.887	0.912	0.927	0.901	0.893	0.854	0.772	0.721
R@20	0.783	0.830	0.854	0.892	0.931	0.949	0.935	0.905	0.871	0.790	0.732

- существует оптимальное число тем на каждом уровне
- три уровня слегка лучше, чем два;
 оптимальное число тем увеличивается

Влияние числа тем на качество поиска

TechCrunch. Все регуляризаторы и модальности, **два уровня**

$ T_1 $	80		100			120					
$ T_2 $	300	350	500	550	600	700	750				
Pr@5	0.651	0.701	0.749	0.789	0.883	0.889	0.889	0.785	0.721	0.701	0.675
Pr@10	0.675	0.709	0.771	0.821	0.891	0.918	0.902	0.803	0.738	0.718	0.691
Pr@15	0.687	0.712	0.773	0.827	0.899	0.919	0.905	0.817	0.741	0.721	0.701
Pr@20	0.683	0.707	0.759	0.815	0.885	0.888	0.895	0.805	0.732	0.716	0.679
R@5	0.749	0.791	0.801	0.854	0.868	0.875	0.861	0.849	0.829	0.731	0.701
R@10	0.765	0.809	0.823	0.873	0.890	0.904	0.875	0.867	0.835	0.745	0.708
R@15	0.771	0.820	0.841	0.882	0.909	0.921	0.895	0.890	0.848	0.769	0.717
R@20	0.778	0.825	0.851	0.887	0.928	0.942	0.929	0.901	0.869	0.785	0.728

- существует оптимальное число тем на каждом уровне
- два уровня лучше, чем один;
 при этом оптимальное число тем увеличивается

Влияние модальностей на качество поиска

Все регуляризаторы и модальности, 3 уровня, оптимальное $|T|$
 Модальности: Words, Bigrams, Authors, Comments, Tags, Hubs, Categories

	Habrahabr						TechCrunch					
	асесс	W	Com	WB	WBTH	All	асесс	W	C	WB	WBC	All
Pr@5	0.821	0.621	0.558	0.673	0.871	0.872	0.822	0.718	0.569	0.795	0.891	0.893
Pr@10	0.869	0.645	0.567	0.712	0.911	0.915	0.851	0.729	0.592	0.807	0.919	0.922
Pr@15	0.875	0.631	0.532	0.693	0.894	0.895	0.835	0.737	0.603	0.803	0.920	0.921
Pr@20	0.863	0.628	0.531	0.688	0.877	0.877	0.813	0.729	0.594	0.792	0.883	0.885
R@5	0.780	0.725	0.645	0.797	0.888	0.889	0.762	0.754	0.659	0.775	0.874	0.877
R@10	0.817	0.748	0.652	0.812	0.921	0.922	0.792	0.778	0.671	0.808	0.908	0.908
R@15	0.850	0.782	0.679	0.842	0.941	0.942	0.835	0.783	0.679	0.825	0.927	0.927
R@20	0.873	0.789	0.672	0.852	0.960	0.961	0.867	0.785	0.711	0.837	0.949	0.949

- лучше использовать все модальности
- биграммы и категории выигрывают у ассессоров
- авторы и комментаторы наименее важны

Влияние регуляризаторов на качество поиска

Все регуляризаторы и модальности, 3 уровня, оптимальное | T |
 Регуляризаторы: \underline{D} ecorrelation, $\underline{\Theta}$ -sparsing, $\underline{\Phi}$ -smoothing, \underline{H} ierarchy

	Habrahabr					TechCrunch				
	нет	D	D Θ	D Φ	D $\Theta\Phi$	нет	D	D Θ	D Φ	D $\Theta\Phi$
Pr@5	0.628	0.772	0.771	0.865	0.872	0.652	0.777	0.779	0.879	0.893
Pr@10	0.653	0.781	0.812	0.883	0.915	0.679	0.788	0.819	0.895	0.922
Pr@15	0.642	0.785	0.792	0.891	0.895	0.669	0.791	0.798	0.901	0.921
Pr@20	0.643	0.771	0.783	0.875	0.877	0.673	0.775	0.792	0.892	0.885
R@5	0.692	0.820	0.805	0.875	0.889	0.673	0.825	0.812	0.869	0.877
R@10	0.714	0.831	0.834	0.905	0.922	0.685	0.856	0.845	0.881	0.908
R@15	0.725	0.847	0.867	0.921	0.942	0.712	0.877	0.869	0.912	0.927
R@20	0.735	0.873	0.891	0.943	0.961	0.723	0.892	0.895	0.934	0.949

- Лучше использовать все регуляризаторы
- Модели со слабой регуляризацией (PLSA, LDA) слабы

Влияние функции близости эмбедингов на качество поиска

Все регуляризаторы и модальности, 3 уровня, оптимальное |T|
Функции близости: Euclidean, Cosine, Manhattan, Hellinger, KL-div

	Habrahabr					TechCrunch				
	Eu	cos	Ma	He	KL	Eu	cos	Ma	He	KL
Pr@5	0.652	0.872	0.772	0.725	0.741	0.647	0.893	0.752	0.742	0.735
Pr@10	0.693	0.915	0.798	0.749	0.772	0.658	0.922	0.794	0.758	0.751
Pr@15	0.695	0.895	0.803	0.737	0.751	0.672	0.921	0.801	0.745	0.742
Pr@20	0.671	0.877	0.789	0.731	0.738	0.652	0.885	0.793	0.739	0.738
R@5	0.693	0.889	0.721	0.742	0.833	0.688	0.877	0.708	0.733	0.858
R@10	0.715	0.922	0.732	0.775	0.868	0.692	0.908	0.715	0.753	0.872
R@15	0.732	0.942	0.739	0.791	0.892	0.724	0.927	0.719	0.785	0.895
R@20	0.741	0.961	0.721	0.812	0.902	0.732	0.949	0.711	0.808	0.901

- косинусная функция близости уверенно лидирует

Выводы по результатам экспериментов

- Ассессорские данные относятся не к темам, а к коллекции; поэтому с их помощью можно оценивать новые модели
- Небольших ассессорских данных хватает для оценивания тематических моделей, которые обучаются *без учителя*
- Регуляризаторы, улучшающие интерпретируемость модели, повышают также и качество поиска
- Иерархия улучшает качество поиска (в основном точность) благодаря сужению области поиска
- Подбор траектории регуляризации и оптимизация коэффициентов регуляризации влияет на качество поиска
- При тщательной оптимизации тематический поиск превосходит как ассессоров, так и конкурирующие модели

Резюме

Разведочный информационный поиск (exploratory search):

- это поиск по смыслу, а не по ключевым словам
- может быть построен на тематическом моделировании
- требует многофункциональности от тематических моделей
- является одной из главных мотиваций для ARTM
- и, в частности, для иерархических моделей

Открытые проблемы

- построение разноуровневых иерархий в ARTM
- оценивание качества тематических иерархий
- оптимизация числа тем на каждом уровне иерархии