Вероятностные тематические модели
Лекция 5. Модальности, иерархии и тематический поиск

К. В. Воронцов
k.vorontsov@iai.msu.ru

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

ВМК МГУ • 24 марта 2025

Содержание

Напоминание. Задача тематического моделирования

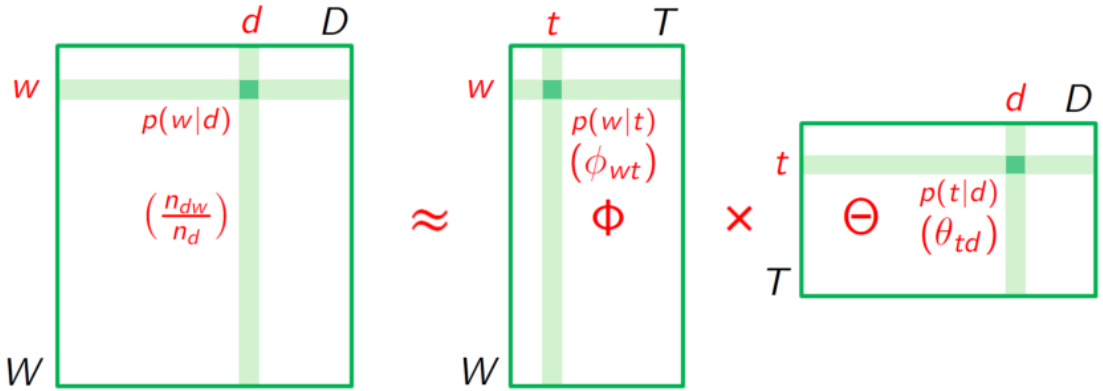
Дано: коллекция текстовых документов, $p(w|d) = \frac{n_{dw}}{n_d}$

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:



Hofmann T. Probabilistic Latent Semantic Indexing. ACM SIGIR, 1999.
Blei D., Ng A., Jordan M. Latent Dirichlet Allocation. NIPS-2001. JMLR 2003.

Напоминание. ARTM — аддитивная регуляризация

Максимизация log правдоподобия с регуляризатором R:

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

EM-алгоритм: метод простой итерации для системы уравнений

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

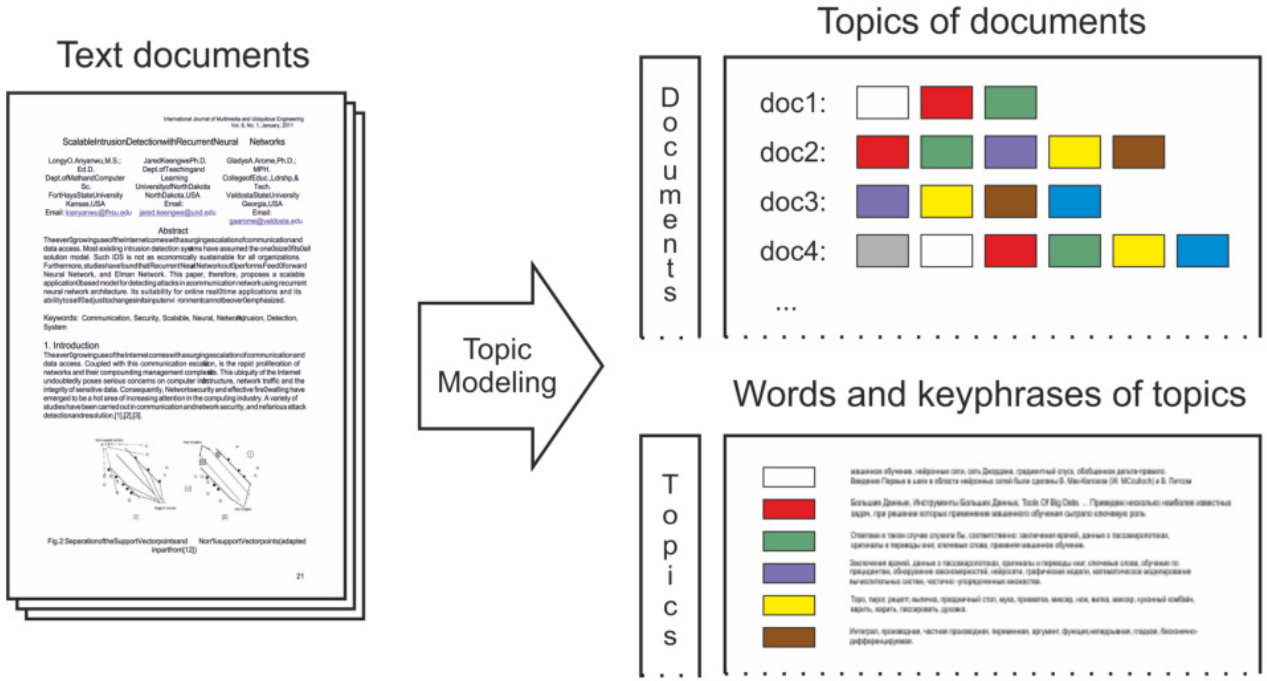
$$\text{M-шаг:} \quad \begin{cases} \phi_{wt} = \underset{w\in W}{\text{norm}}\left(n_{wt} + \phi_{wt}\dfrac{\partial R}{\partial \phi_{wt}}\right), & n_{wt} = \sum\limits_{d\in D} n_{dw}p_{tdw} \\ \theta_{td} = \underset{t\in T}{\text{norm}}\left(n_{td} + \theta_{td}\dfrac{\partial R}{\partial \theta_{td}}\right),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

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

Воронцов К. В. Аддитивная регуляризация тематических моделей коллекций текстовых документов. Доклады РАН, 2014.

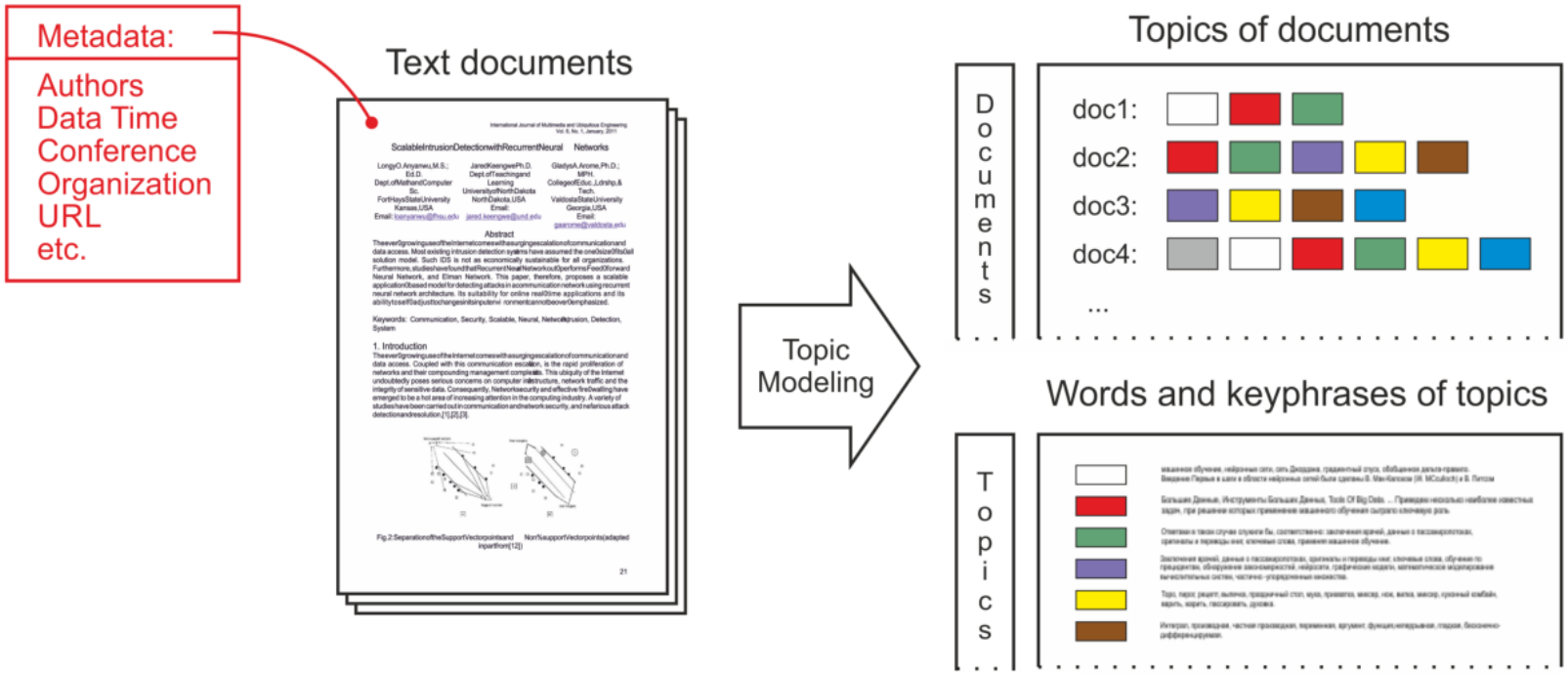
Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$,



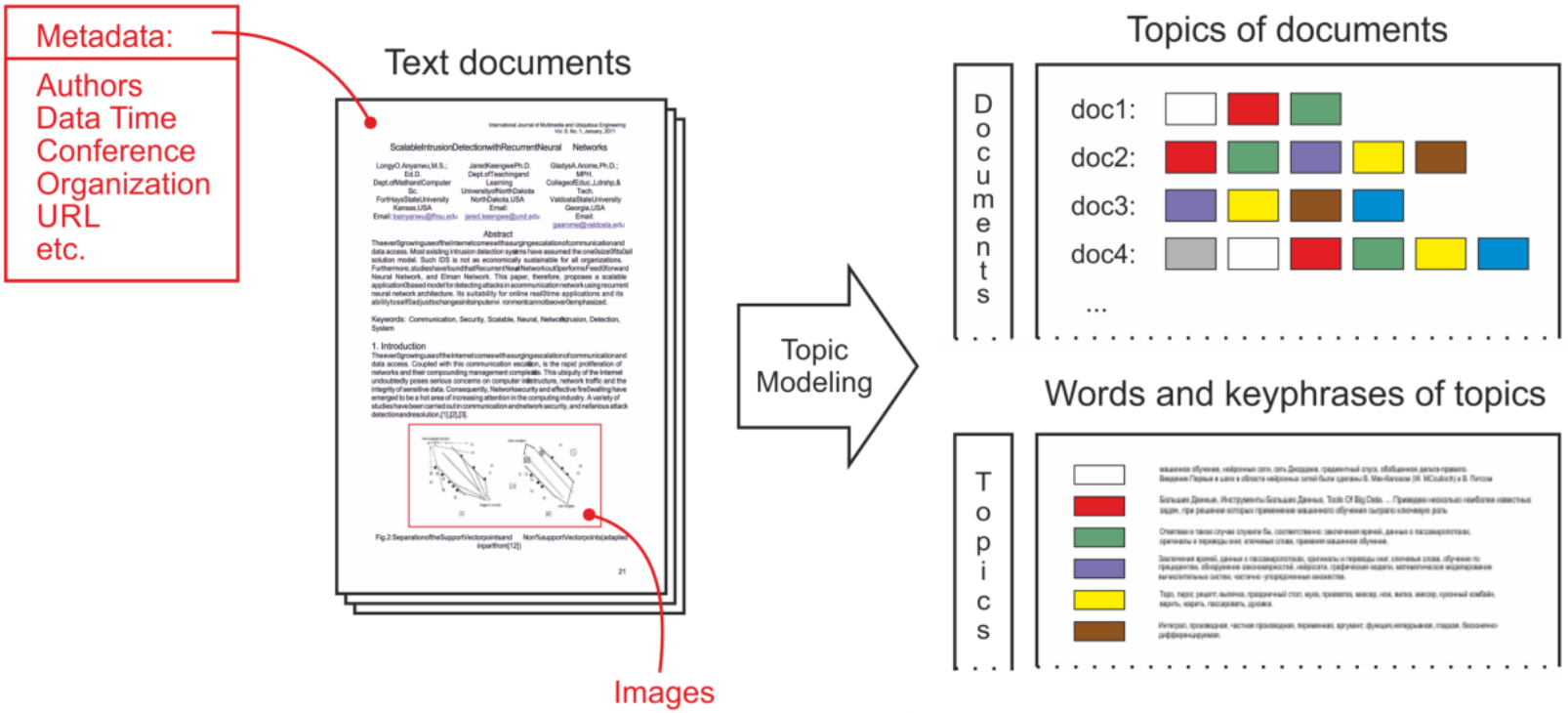
Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$, $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{источник}|t)$,



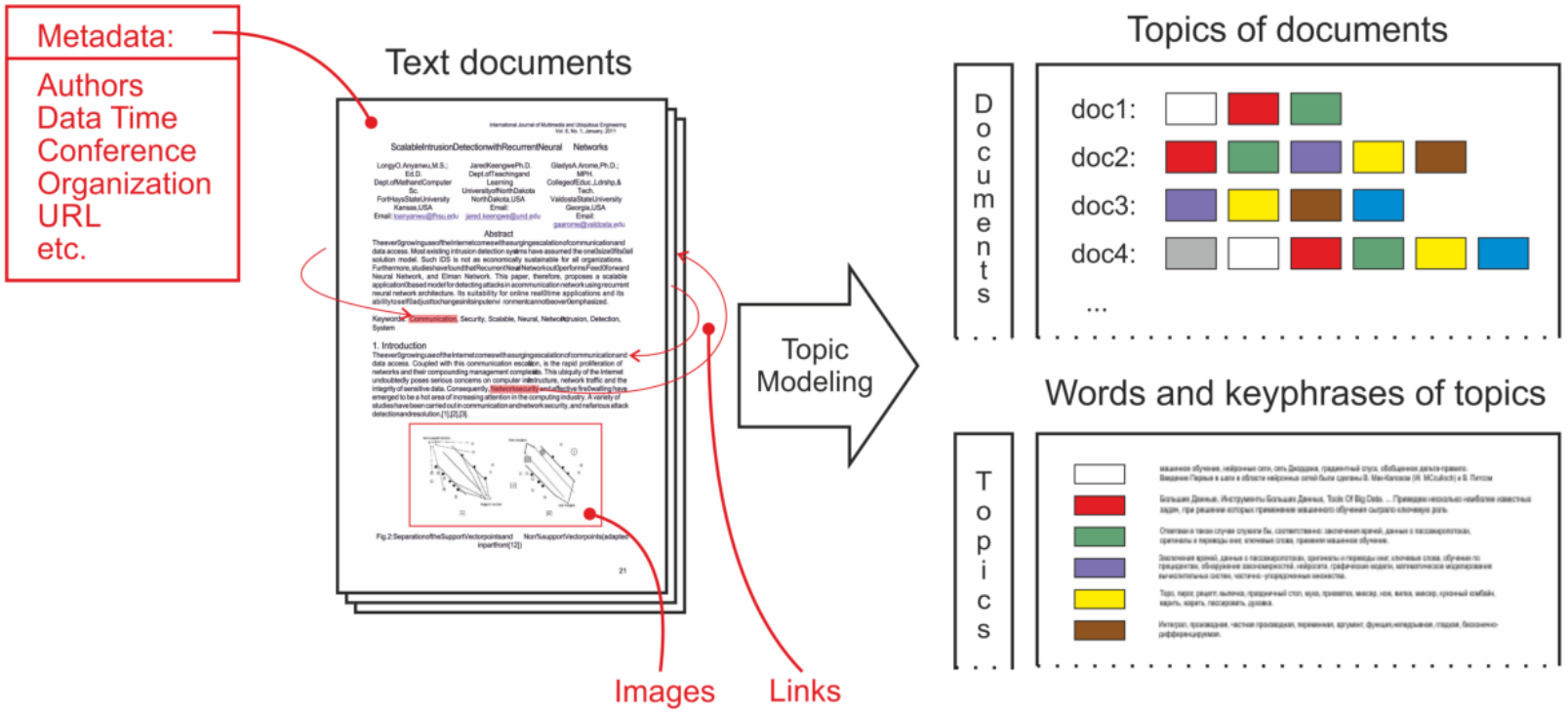
Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$, $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{источник}|t)$,
$p(\text{объект}|t)$,



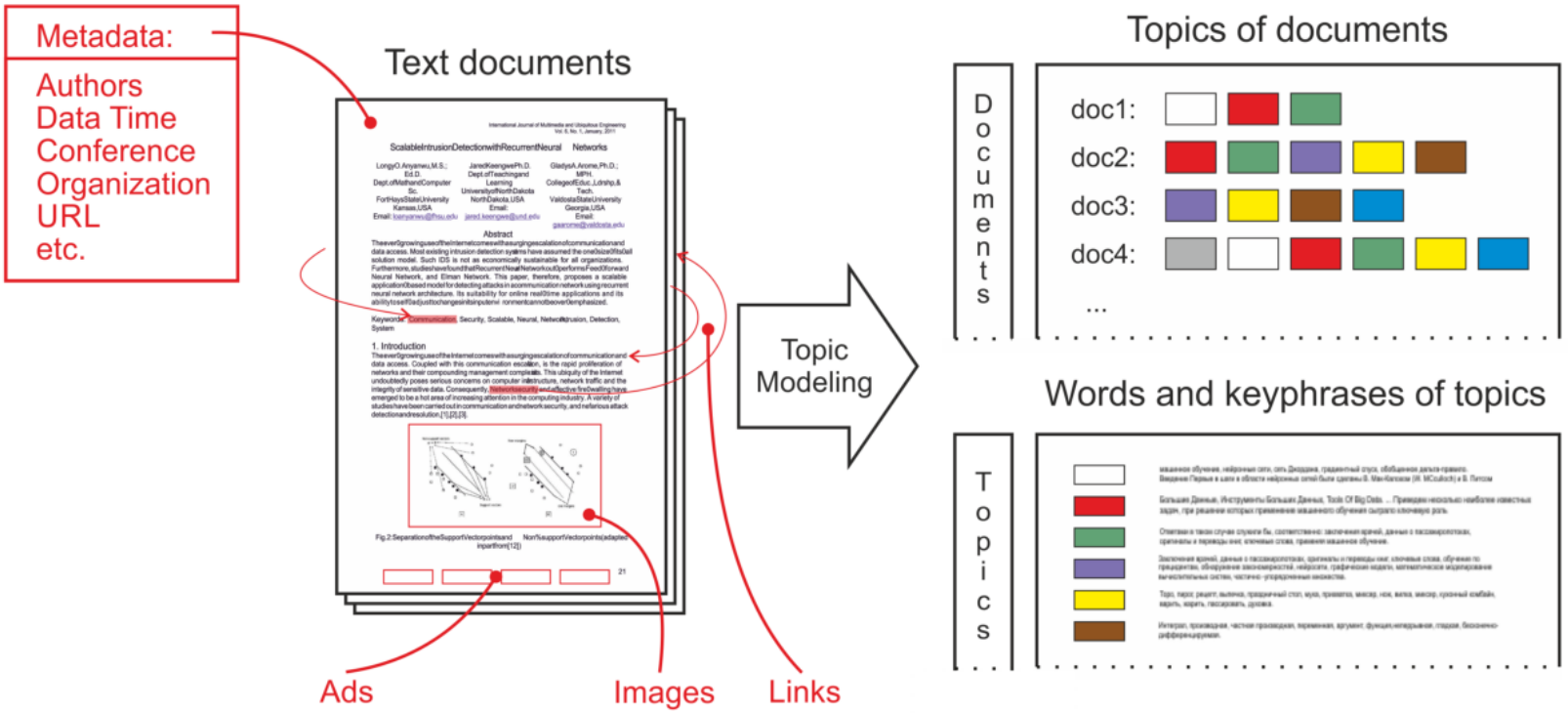
Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$, $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{источник}|t)$,
$p(\text{объект}|t)$, $p(\text{ссылка}|t)$,



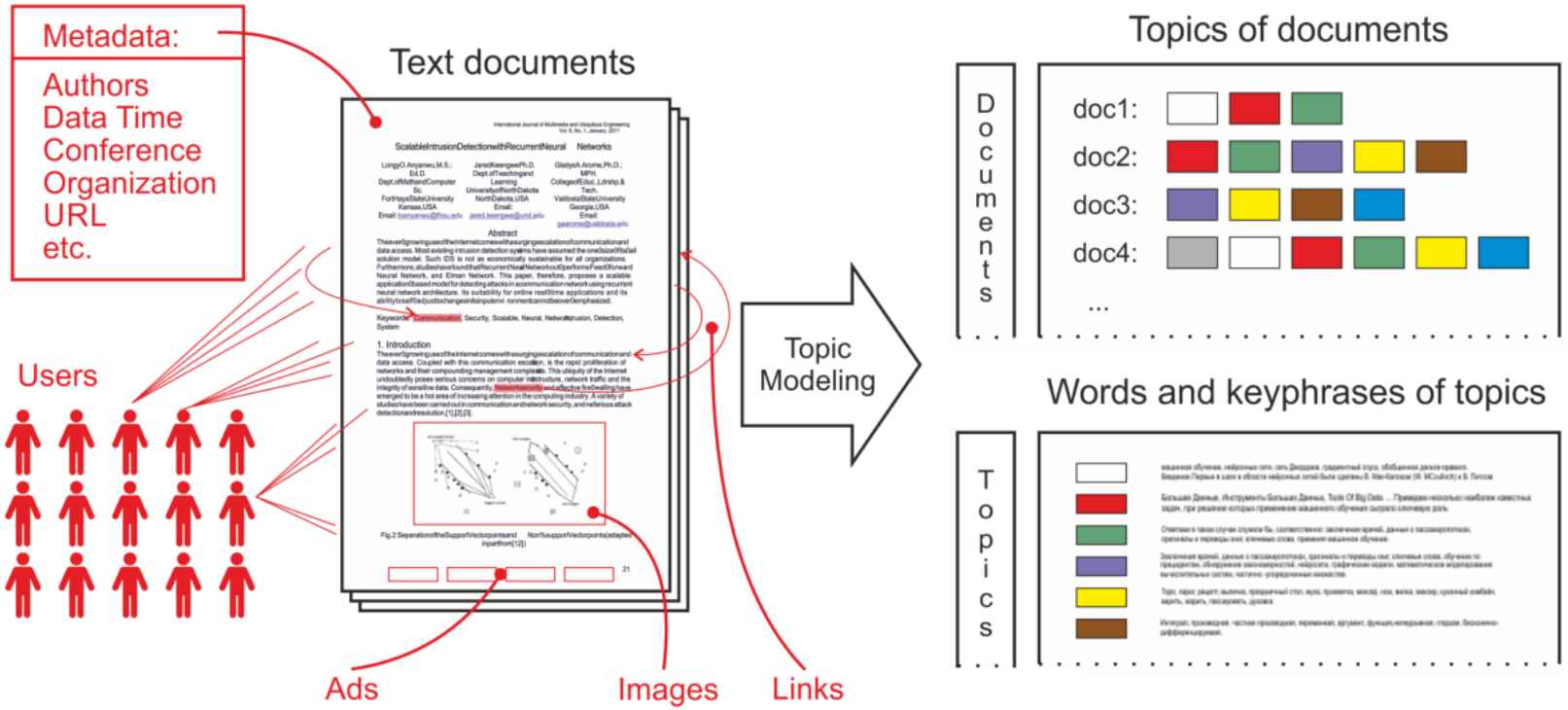
Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$, $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{источник}|t)$,
$p(\text{объект}|t)$, $p(\text{ссылка}|t)$, $p(\text{баннер}|t)$,



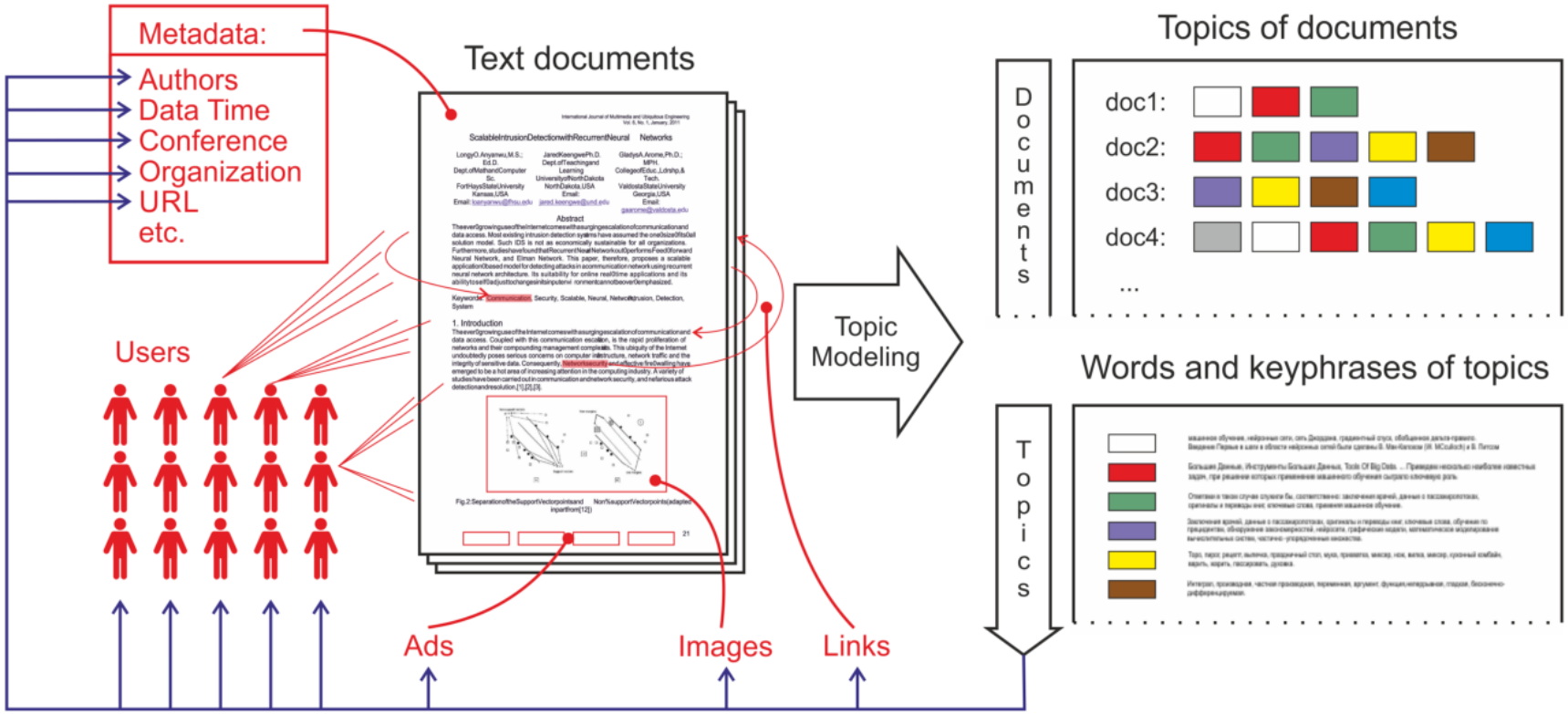
Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$, $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{источник}|t)$,
$p(\text{объект}|t)$, $p(\text{ссылка}|t)$, $p(\text{баннер}|t)$, $p(\text{пользователь}|t)$



Мультимодальная тематическая модель

Тема может порождать термы различных *модальностей*:
$p(\text{слово}|t)$, $p(n\text{-грамма}|t)$, $p(\text{автор}|t)$, $p(\text{время}|t)$, $p(\text{источник}|t)$,
$p(\text{объект}|t)$, $p(\text{ссылка}|t)$, $p(\text{баннер}|t)$, $p(\text{пользователь}|t)$



EM-алгоритм для мультимодальной ARTM

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

Максимизация суммы log-правдоподобий с регуляризацией:

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

EM-алгоритм: метод простой итерации для системы уравнений

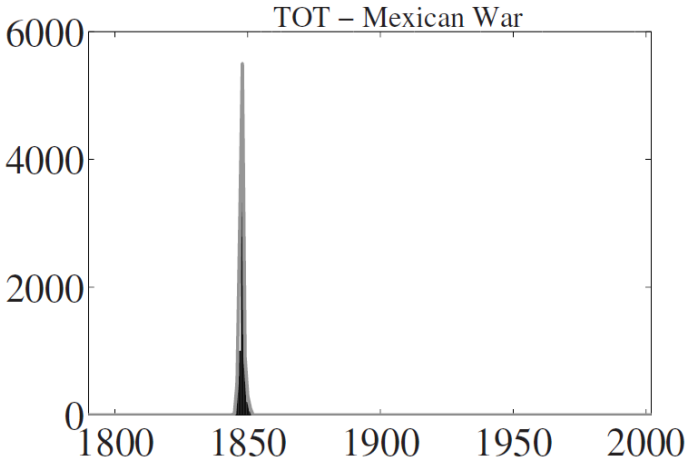
$$\text{E-шаг:} \quad p_{tdw} = \underset{t \in T}{\text{norm}}\left(\phi_{wt} \theta_{td}\right)$$

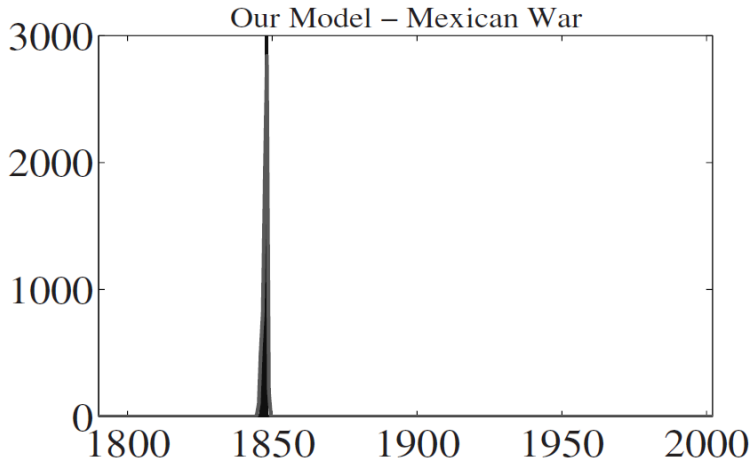
$$\text{M-шаг:} \quad \begin{cases} \phi_{wt} = \underset{w \in W^m}{\text{norm}}\left(\sum_{d \in D} \tau_{m(w)} n_{dw} p_{tdw} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}} \right) \\ \theta_{td} = \underset{t \in T}{\text{norm}}\left(\sum_{w \in d} \tau_{m(w)} n_{dw} p_{tdw} + \theta_{td} \frac{\partial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

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

Пример. Использование модальностей времени и *n*-грамм

По коллекции выступлений президентов США





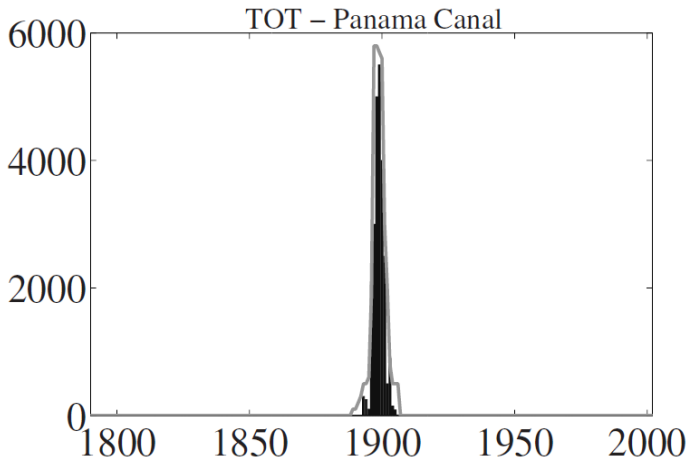
1. mexico	8. territory
2. texas	9. army
3. war	10. peace
4. mexican	11. act
5. united	12. policy
6. country	13. foreign
7. government	14. citizens

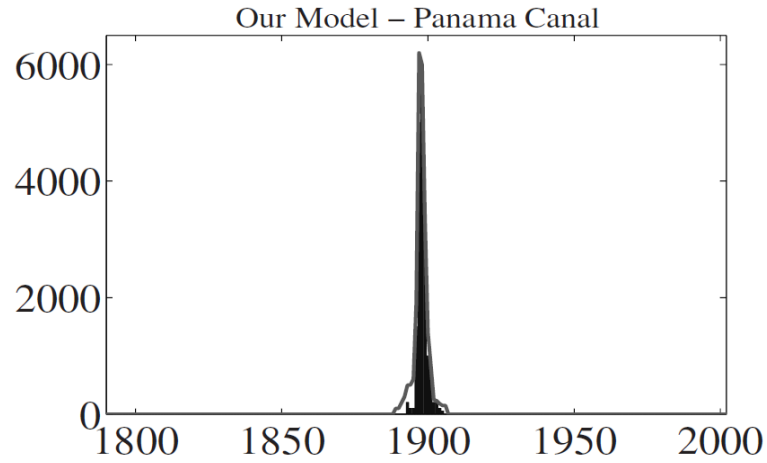
1. east bank	8. military
2. american coins	9. general herrera
3. mexican flag	10. foreign coin
4. separate independent	11. military usurper
5. american commonwealth	12. mexican treasury
6. mexican population	13. invaded texas
7. texan troops	14. veteran troops

Shoaib Jameel, Wai Lam. An *n*-gram topic model for time-stamped documents. 2013

Пример. Использование модальностей времени и *n*-грамм

По коллекции выступлений президентов США



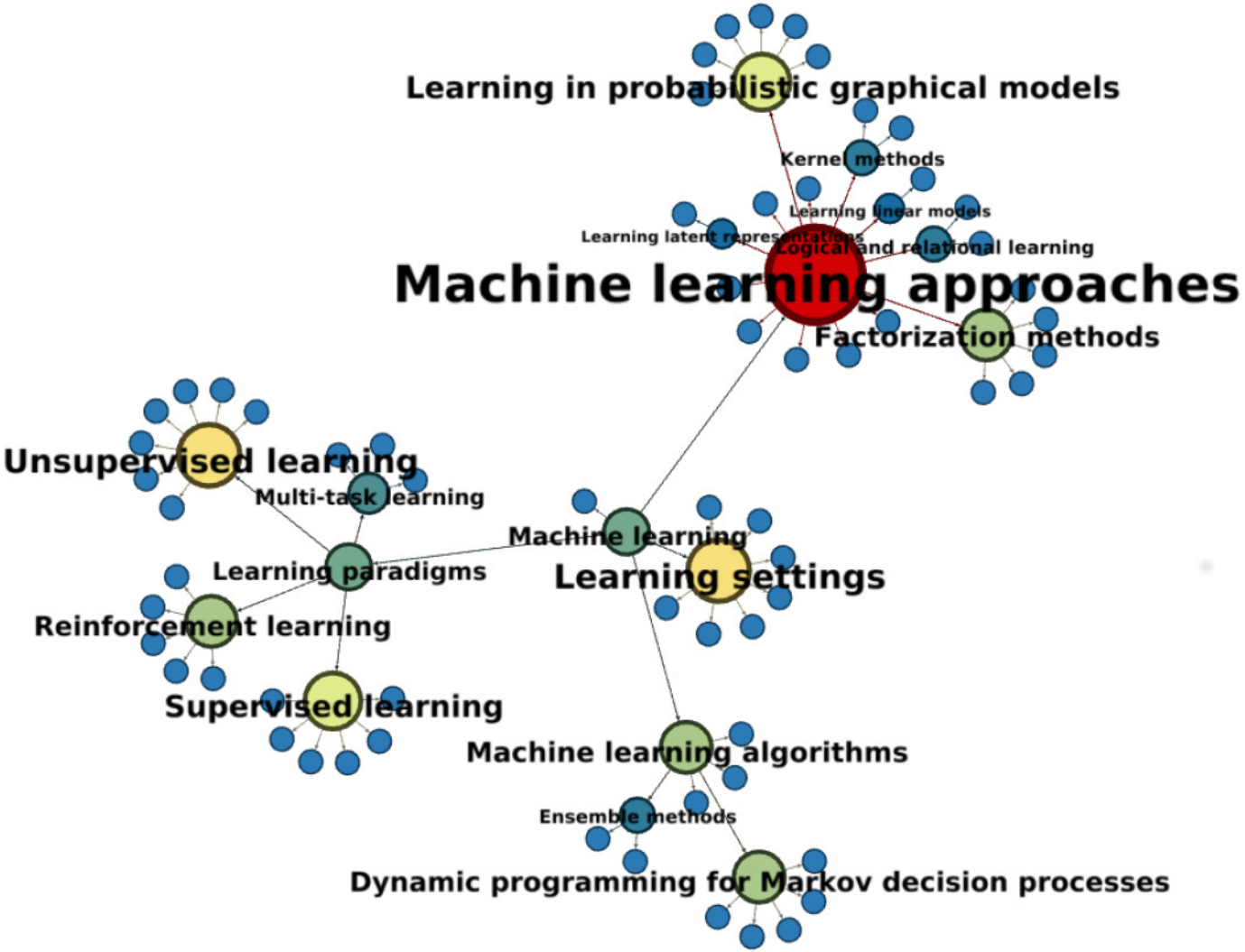


1. government	8. spanish
2. cuba	9. island
3. islands	10. act
4. international	11. commission
5. powers	12. officers
6. gold	13. spain
7. action	14. rico

1. panama canal	8. united states senate
2. isthmian canal	9. french canal company
3. isthmus panama	10. caribbean sea
4. republic panama	11. panama canal bonds
5. united states government	12. panama
6. united states	13. american control
7. state panama	14. canal

Shoaib Jameel, Wai Lam. An *n*-gram topic model for time-stamped documents. 2013

Пример древовидной тематической иерархии



G.Bordea. Domain adaptive extraction of topical hierarchies for expertise mining. 2013.

Стратегии иерархического разделения тем на подтемы

Процесс построения иерархии тем:

- структура: дерево / многодольный граф
- направление: снизу вверх / сверху вниз / одновременно
- наращивание: повершинное / послойное
- обучение: без учителя / по готовым рубрикаторам

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 Ф: 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:

$$\sum_{t\in T} n_t \,\mathrm{KL}_w\Big(p(w|t) \,\Big\|\, \sum_{s\in S} p(w|s)p(s|t)\Big) \;\to\; \min_{\Phi,\Psi},$$

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

Родительская $\Phi^p \approx \Phi\Psi$, отсюда регуляризатор матрицы Φ:

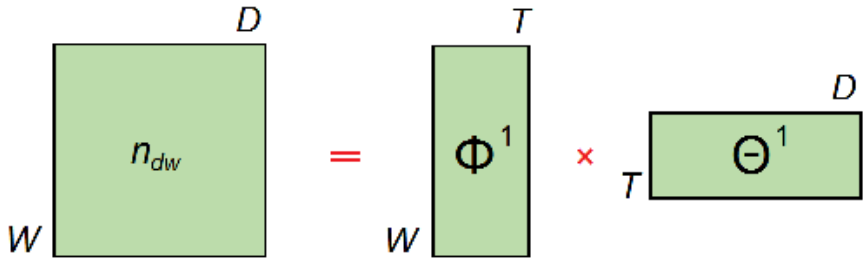
$$R(\Phi,\Psi) = \tau \sum_{t\in T}\sum_{w\in W} n_{wt} \ln \sum_{s\in S} \phi_{ws}\psi_{st} \;\to\; \max$$

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

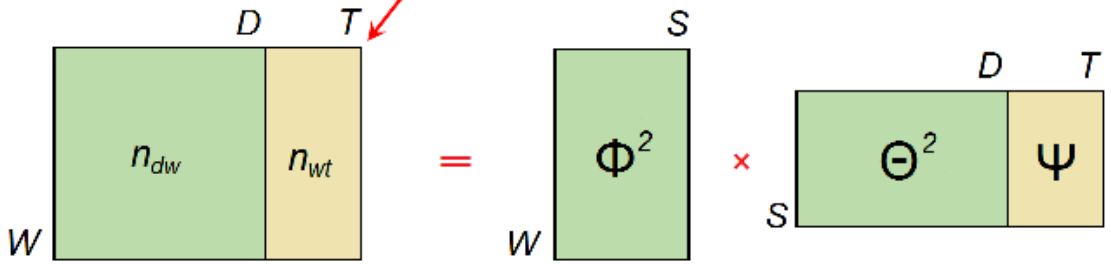
Регуляризатор Φ: построение второго уровня с подтемами S

Добавим в коллекцию $|T|$ псевдо-документов родительских тем
с частотами термов $n_{wt} = \tau n_t \phi_{wt}$, $t \in T$

родительский уровень 1



дочерний уровень 2



Матрица связей тем с подтемами $\Psi = \big(p(s|t)\big)$ образуется
в столбцах матрицы Θ, соответствующих псевдо-документам.

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

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:

$$\sum_{d\in D} n_d \,\mathrm{KL}_t\Big(p(t|d) \,\Big\|\, \sum_{s\in S} p(t|s)p(s|d)\Big) \;\to\; \min_{\Theta,\Psi},$$

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

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

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

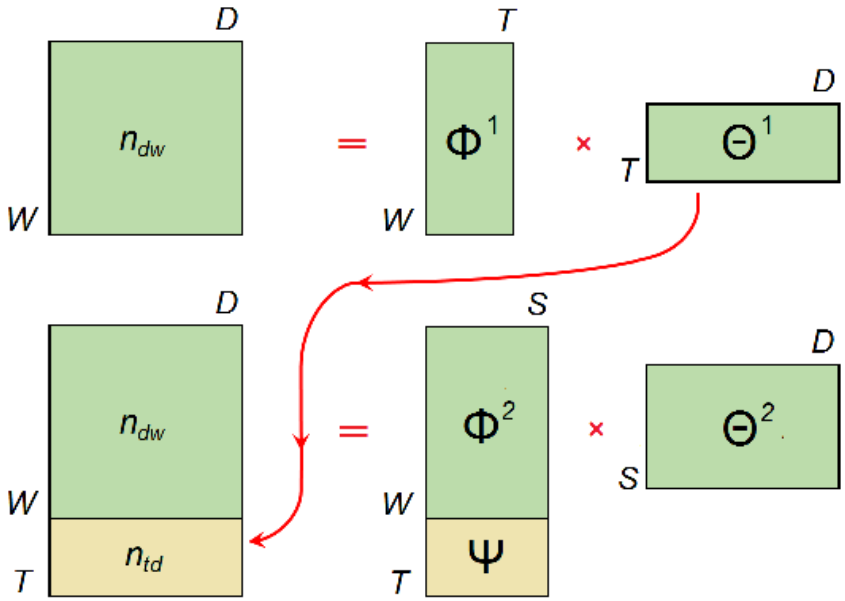
Родительские темы t — модальность с частотами термов n_{td}

Регуляризатор Θ: построение второго уровня с подтемами S

Добавим в каждый документ модальность родительских тем
с частотами термов $n_{td} = \tau n_d \theta_{td}$, $t \in T$

родительский уровень 1

дочерний уровень 2



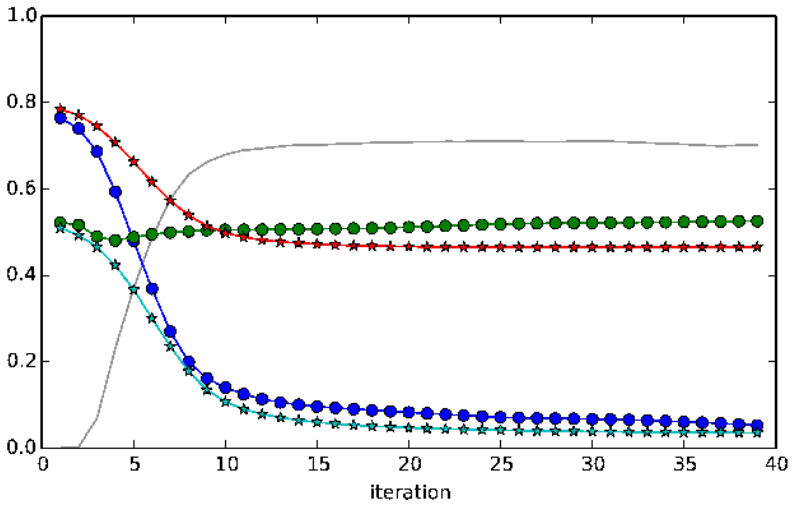
Матрица связей тем с подтемами $\Psi = \big(p(t|s)\big)$ образуется
в строках матрицы Φ, соответствующих родительским темам.

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

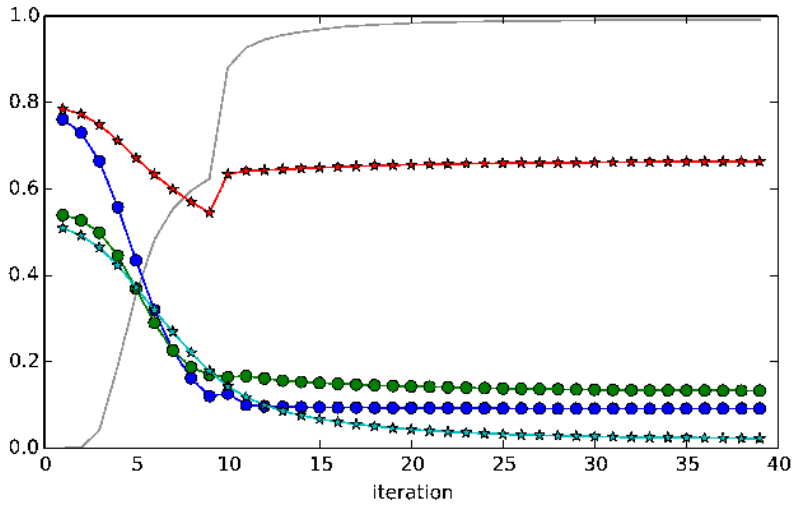
Эксперимент на коллекции ММРО-ИОИ

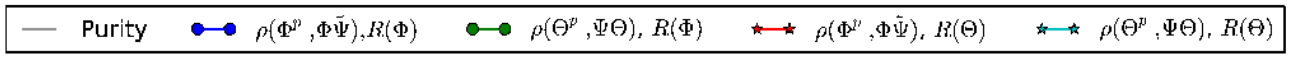
Среднее расстояние Хеллингера $\rho(\Phi^p, \Phi\tilde{\Psi})$ и $\rho(\Theta^p, \Psi\Theta)$ для регуляризаторов $R(\Phi)$ и $R(\Theta)$ при переходе с уровня 1 на 2:

Разреживание Φ с 1-й итерации



Разреживание Φ с 10-й итерации





Выводы. $R(\Theta)$ плохо приближает Φ^p. При разреживании Φ с 10-й итерации $R(\Phi)$ хорошо приближает Φ^p и Θ^p

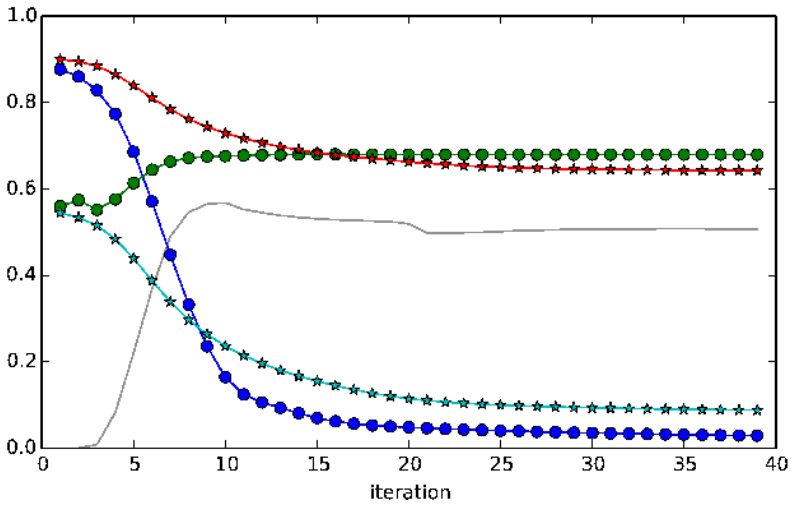
---

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

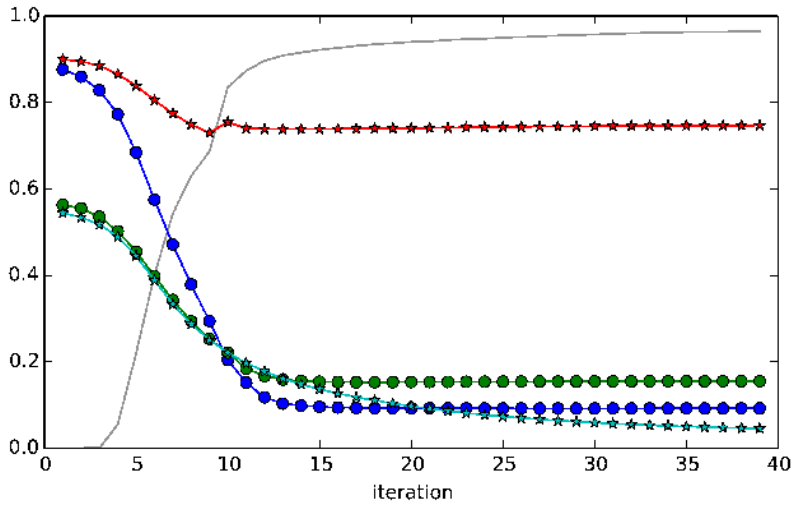
Эксперимент на коллекции ММРО-ИОИ

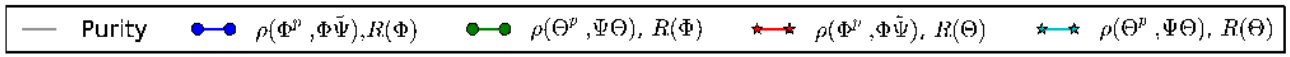
Среднее расстояние Хеллингера $\rho(\Phi^p, \Phi\tilde{\Psi})$ и $\rho(\Theta^p, \Psi\Theta)$ для регуляризаторов $R(\Phi)$ и $R(\Theta)$ при переходе с уровня 2 на 3:

Разреживание Φ с 1-й итерации



Разреживание Φ с 10-й итерации





Выводы. $R(\Theta)$ плохо приближает Φ^p. При разреживании Φ с 10-й итерации $R(\Phi)$ хорошо приближает Φ^p и Θ^p

---

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

Выводы

- $R(\Phi)$ лучше $R(\Theta)$, т.к. добавлять псевдо-документы удобнее, чем вставлять модальности в каждый документ
- $R(\Phi)$ хорошо приближает $\Phi^p \approx \Phi\tilde{\Psi}$ и $\Theta^p \approx \Psi\Theta$ при осторожном (с 10-й итерации) разреживании Φ
- $R(\Theta)$ приближает только $\Theta^p \approx \Psi\Theta$
- сильное разреживание $\psi_{ts} \in \{0, 1\}$ даёт иерархию–дерево
- нельзя допускать вырождения $\psi_{ts} = p(t|s) \equiv 0$

Трудные и/или открытые проблемы:

- тематические иерархии с ветвлением различной глубины
- автоматическое оценивание качества иерархии
- автоматическое именование подтем с учётом родительской
- определение типа документа по его следу в иерархии

Визуализация тематической иерархии

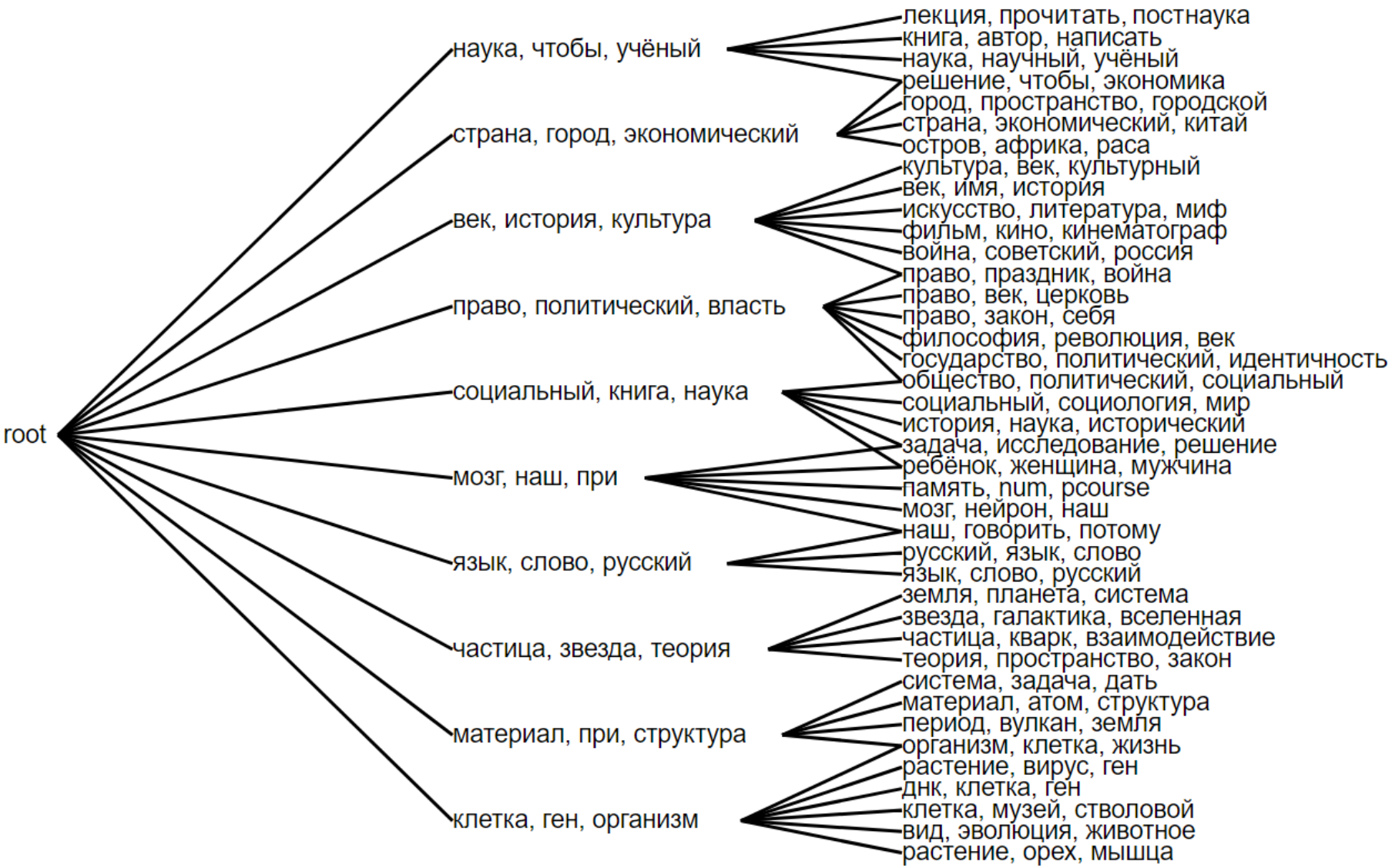
Тексты научно-просветительского ресурса Postnauka.ru:
2976 документов, 43196 слов, 1799 тегов

Для именования темы используются три топовых слова темы

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.

Иерархический спектр тем (коллекция postnauka.ru)



Построение спектра тем. Постановка задачи

Тематический спектр — такая перестановка тем $t_1, \ldots, t_{|T|}$, что сумма расстояний между соседними темами минимальна:

$$\sum_{i=2}^{|T|} \rho(t_i, t_{i-1}) \to \min$$

Функция расстояния $\rho(t, t')$ между темами, примеры:

- Манхэттенское: $\rho(t, t') = \sum_{w \in W} \left| \phi_{wt} - \phi_{wt'} \right|$
- Хеллингера: $\rho^2(t, t') = \frac{1}{2} \sum_{w \in W} \left(\sqrt{\phi_{wt}} - \sqrt{\phi_{wt'}} \right)^2$
- Жаккара: $\rho(t, t') = 1 - \frac{|W_t \cap W_{t'}|}{|W_t \cup W_{t'}|}, \quad W_t = \left\{ w : \phi_{wt} > \frac{1}{|W|} \right\}$

Построение спектра тем — это задача коммивояжёра

Задача TSP (traveling salesman problem)

Найти путь минимальной суммарной стоимости, соединяющий T городов так, чтобы в каждом городе побывать один раз.

Алгоритм Лина–Кернигана в реализации Хельсгауна
— лучший для решения задачи TSP
(по данным *Encyclopedia of operations research* на 2013 год)

Вычислительная сложность алгоритм — $O(T^{2.2})$.

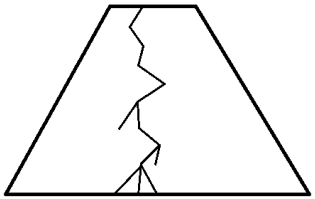
Другие алгоритмы оказались не только медленнее,
но и хуже по качеству тематических спектров.

Keld Helsgaun. An effective implementation of the Lin–Kernighan traveling salesman heuristic. EJOR, 2000.

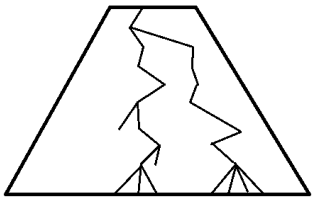
Дмитрий Федоряка. Технология интерактивной визуализации тематических моделей. Бакалаврская диссертация. МФТИ, 2017.

Иерархическая тематизация коллекции научных публикаций

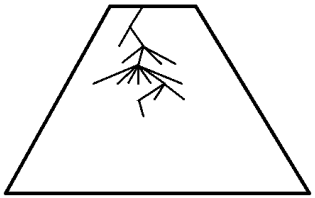
След документа в глубокой тематической иерархии определяет его тип — степень специализации, назначение, аудиторию:



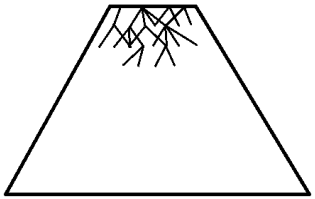
узко специализированный,
для профессионалов



междисциплинарное исследование,
для профессионалов



обзорный,
для ознакомления с предметной областью



популярный или энциклопедический,
для самообразования, расширения кругозора

Две коллекции новостей про технологии

Habrahabr.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)
- удалена пунктуация, ё→е, лемматизация pymorphy2

Анастасия Янина. Тематические и нейросетевые модели языка для разведочного информационного поиска. Диссертация к.ф.-м.н., МФТИ. 2022.

Методика оценивания качества разведочного поиска

Поисковый запрос

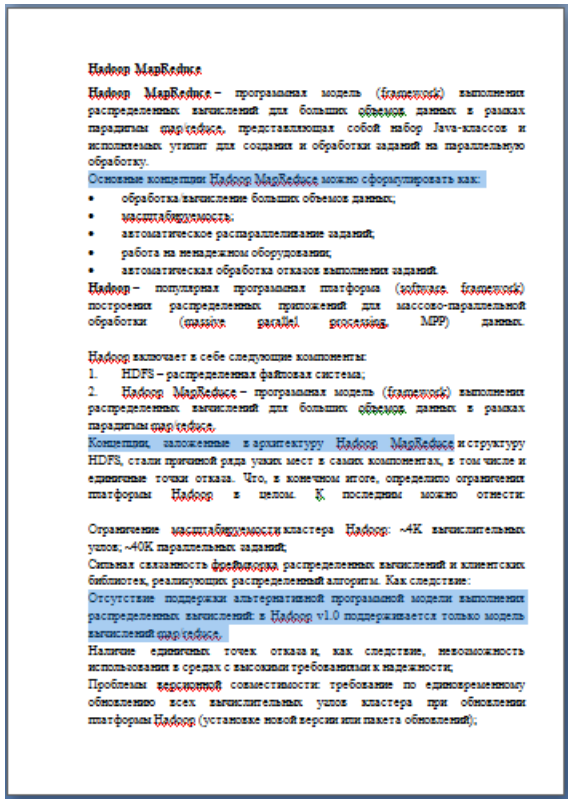
набор ключевых слов или фрагментов текста, около одной страницы A4

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 исполняемых 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

Векторный 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:

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

Ранжируем документы коллекции $d \in D$ по убыванию $\text{sim}(q,d)$
Выдача тематического поиска — k первых документов.

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

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

A.Ianina, K.Vorontsov. Regularized multimodal hierarchical topic model for document-by-document exploratory search. FRUCT–ISMW, 2019.

Оценивание качества поиска

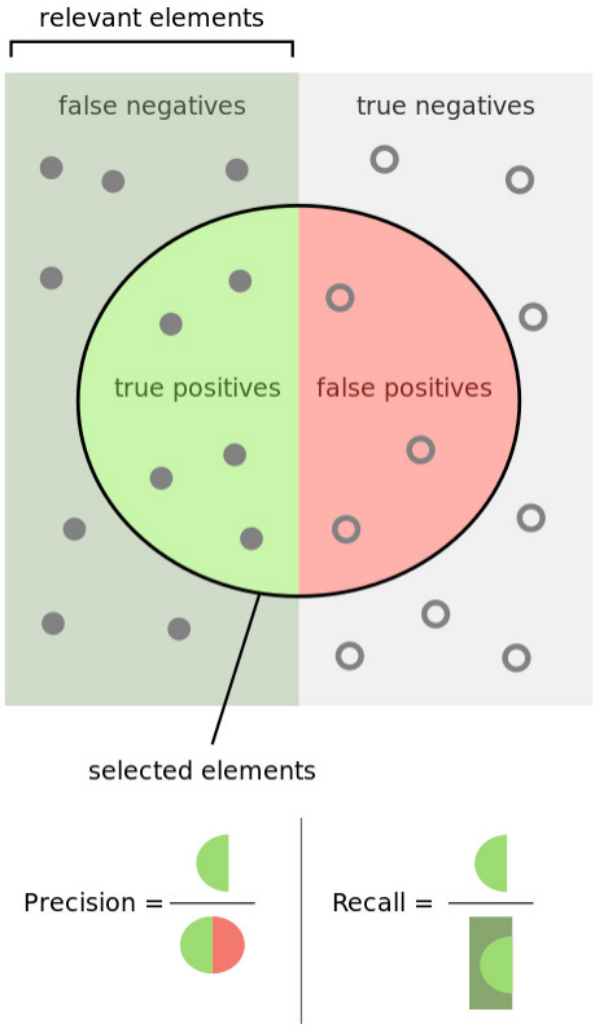
Precision — доля релевантных среди найденных
Recall — доля найденных среди релевантных

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

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

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

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



Какие модели поиска сравнивались

- **assessors:** результаты поиска, выполненного асессорами
- **TF-IDF, BM25:** сравнение документов по частотам слов
- **word2vec:** нетематические векторные представления слов
- **PLSA:** Probabilistic Latent Semantic Analysis (1999)
- **LDA:** Latent Dirichlet Allocation (2001)
- **ARTM:** тематическая модель с тремя регуляризаторами
- **hARTM:** иерархические модели ARTM 2x и 3x уровней

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 $p(t|d)$:

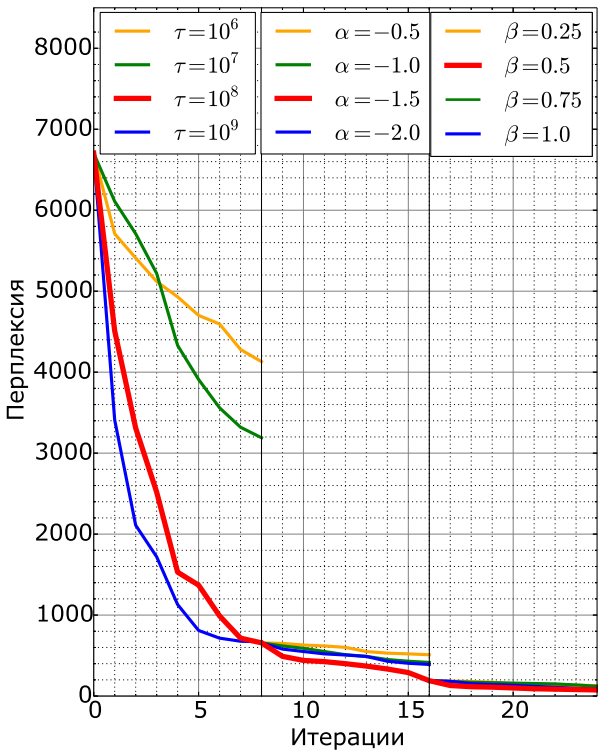
$$R(\Theta) = -\alpha \sum_{d,t} \ln \theta_{td}$$

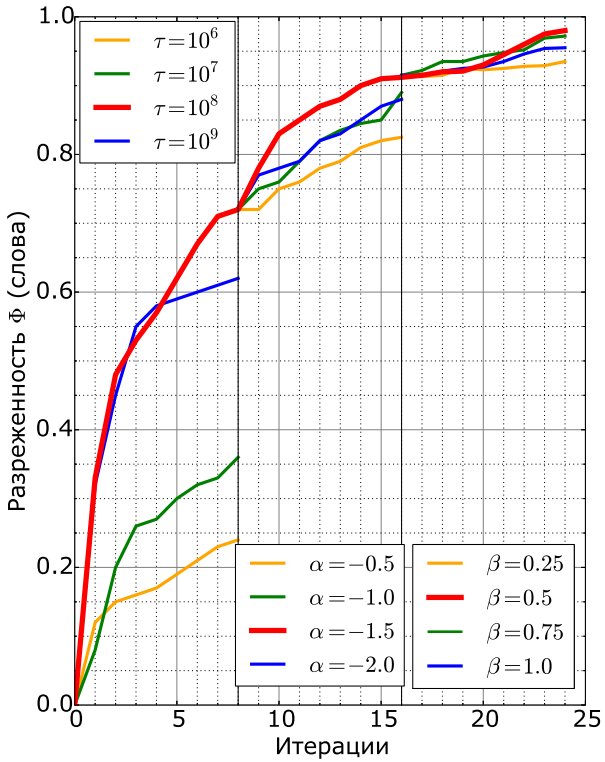
3. сглаживание распределений $p(w|t)$:

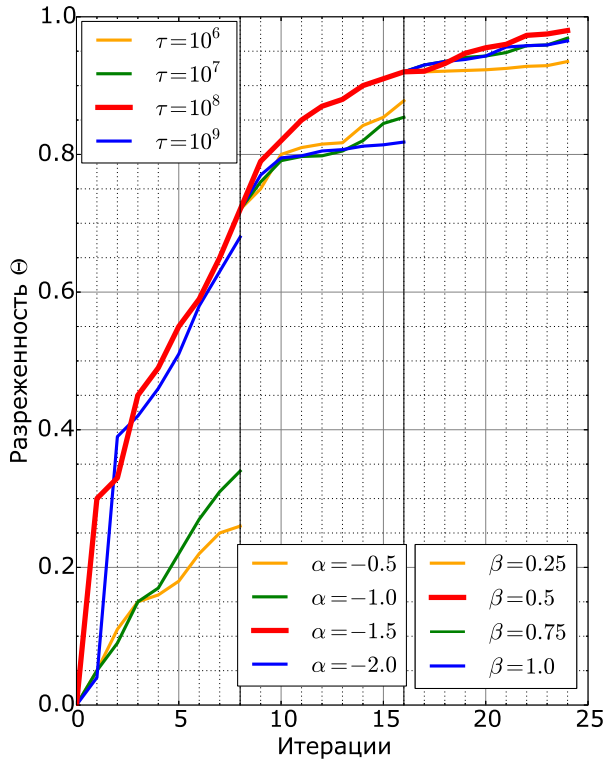
$$R(\Phi) = \beta \sum_{t,w} \ln \phi_{wt}$$

Последовательный подбор коэффициентов регуляризации

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



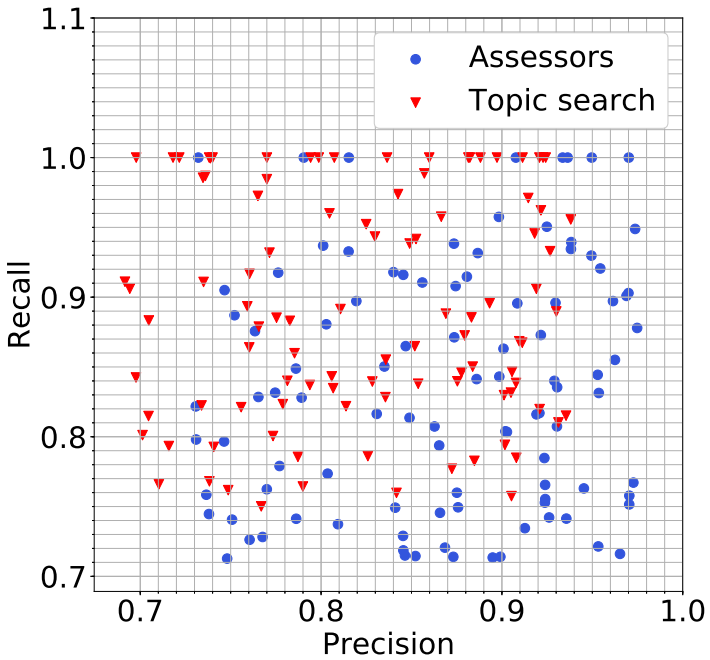




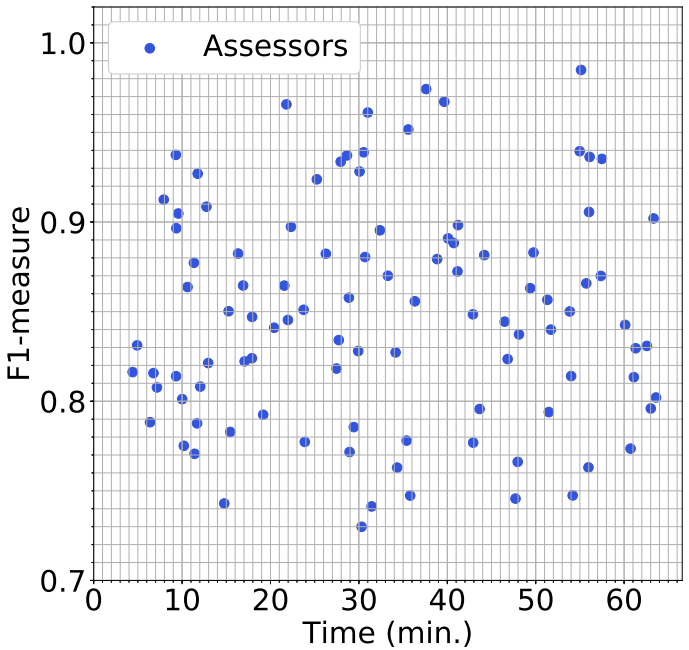
Результаты измерения точности и полноты по запросам

100 запросов, 3 асессора на запрос

точность и полнота поиска



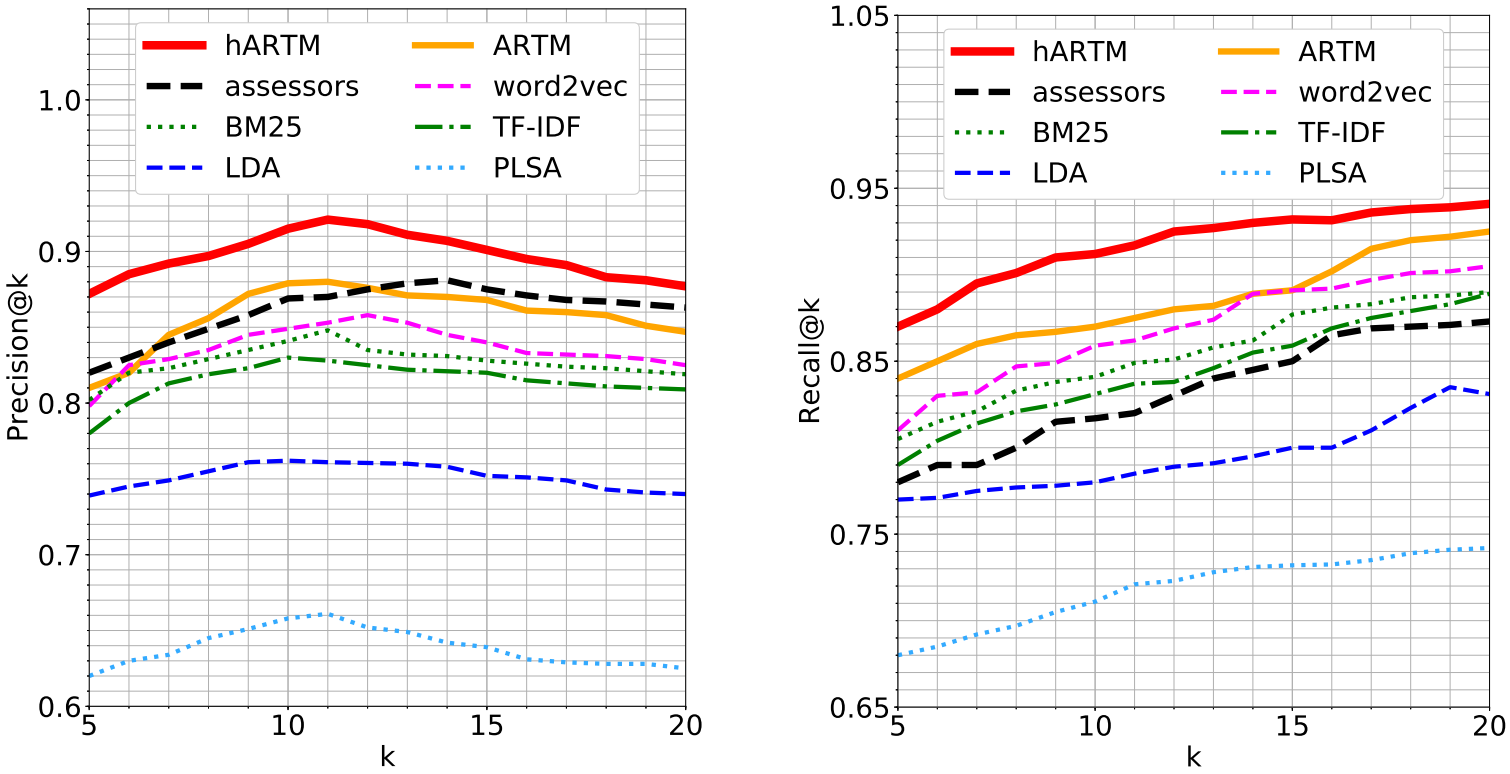
время и F_1-мера (асессоры)



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

Сравнение с асессорами по качеству поиска

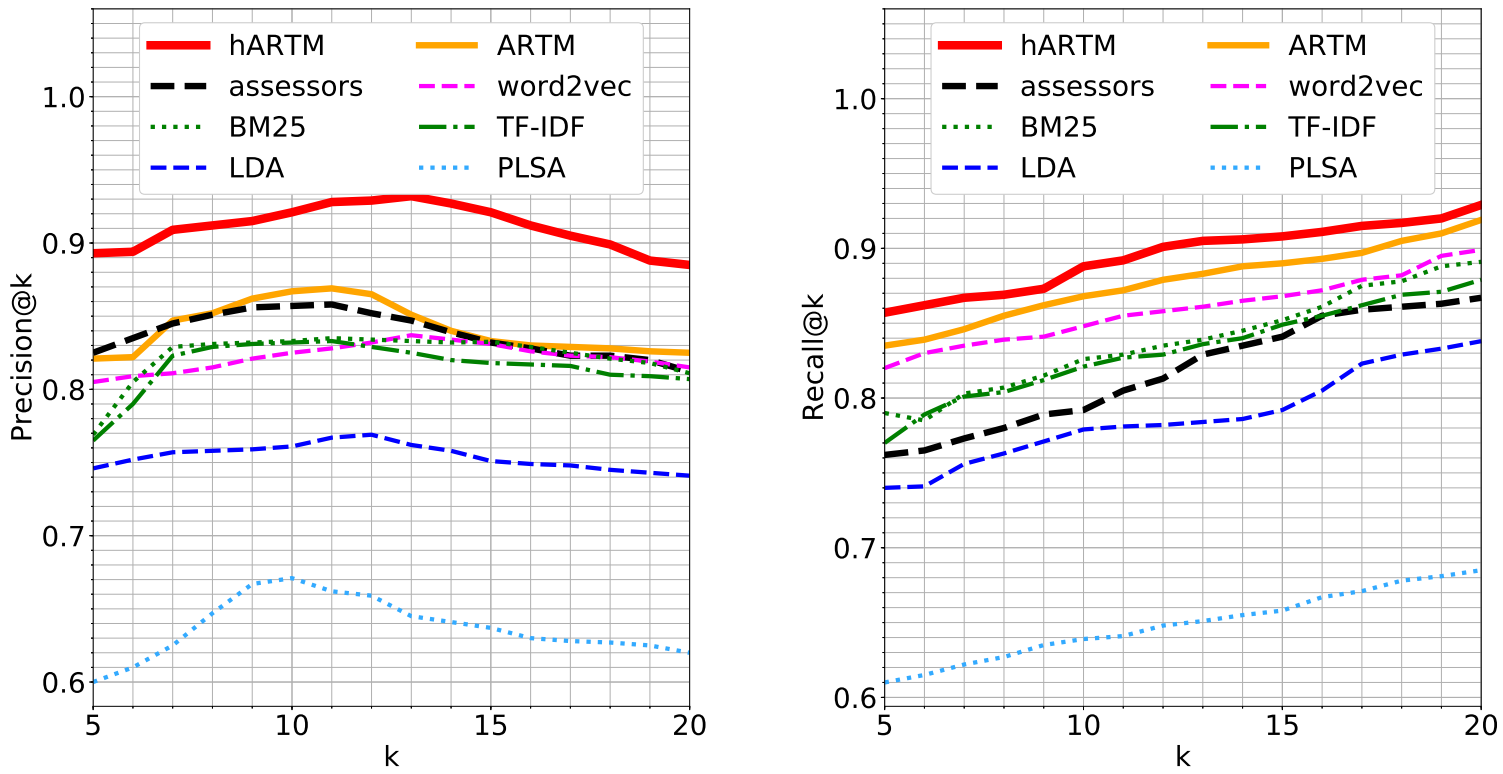
Точность и полнота по первым k позициям поисковой выдачи (коллекция Habrahabr.ru)



A.Ianina, K.Vorontsov. Regularized multimodal hierarchical topic model for document-by-document exploratory search. 2019.

Сравнение с асессорами по качеству поиска

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



A.Ianina, K.Vorontsov. Regularized multimodal hierarchical topic model for document-by-document exploratory search. 2019.

Влияние числа тем на качество поиска

Все регуляризаторы и модальности, плоская модель

	Habrahabr						TechCrunch					
	acecc	100	150	**200**	250	400	acecc	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

Habrahabr. 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
- увеличивается оптимальное число тем на нижнем уровне

Влияние числа тем на качество поиска

Habrahabr. 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 лучше, чем один или два
- увеличивается оптимальное число тем на нижнем уровне

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
- увеличивается оптимальное число тем на нижнем уровне

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 лучше, чем один или два
- увеличивается оптимальное число тем на нижнем уровне

Влияние модальностей на качество поиска

Все регуляризаторы и модальности, 3 уровня, оптимальное $|T|$

Модальности: Words, Bigrams, Authors, Comments, Tags, Hubs, Categories

	Habrahabr						TechCrunch					
	acecc	W	Com	WB	WBTH	**All**	acecc	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 асессоров
- авторы и комментаторы наименее важны

Влияние регуляризаторов на качество поиска

Все регуляризаторы и модальности, 3 уровня, оптимальное $|T|$

Регуляризаторы: Decorrelation, Θ-sparsing, Φ-smoothing, Hierarchy

	Habrahabr					TechCrunch				
	нет	D	DΘ	DΘΦ	DΘΦH	нет	D	DΘ	DΘΦ	DΘΦH
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 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

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

A.Ianina, K.Vorontsov. Regularized multimodal hierarchical topic model for document-by-document exploratory search. 2019.

Тематическая модель для научного поиска должна быть...

1. **Интерпретируемая**: объяснять смысл каждой темы
2. **Иерархическая**: разделять тем на подтемы
3. **Хронологическая**: прослеживать темы во времени
4. **Мультимодальная**: слова, авторы, категории, связи, теги,...
5. **Мультиграммная**: слова, термины-словосочетания
6. **Мультиязычная** для кросс- и много-языкового поиска
7. **Сегментирующая** документ на тематические блоки
8. **Обучаемая** по обратной связи с пользователями
9. **Определяющая число тем** автоматически
10. **Создающая и именующая новые темы** автоматически
11. **Онлайновая**: обрабатывать поток документов
12. **Параллельная, распределённая** при больших данных

Резюме

Разведочный информационный поиск (exploratory search):

- это поиск по смыслу, а не по ключевым словам
- строится на векторных представлениях текста (тематических или нейросетевых эмбедингах текста)
- требует от тематических моделей многофункциональности
- является одной из главных мотиваций для ARTM,
- в том числе для мультимодальных и иерархических ARTM

Открытые проблемы:

- тематизация подборок с дисбалансом тем
- автоматическое именование и суммаризация тем
- эффективные методы визуализации (картирования)

Напоминание. Какими будут задания по курсу

Задача-минимум: научиться решать задачи NLP
с использованием тематического моделирования в BigARTM

Задача-максимум: сделать полезное мини-исследование

виды деятельности	оценка
теоретические задания	$\sum_i X_i$
решение прикладной задачи	$5X$
обзор по NeuralTM	$5X$
интеграция ARTM в pyTorch	$5X$
участие в одном из проектов	$10X$
работа над открытой проблемой	$10X$

где X — оценка за вид деятельности по 5-балльной шкале.

Итоговая оценка: $\min\big(10, \lfloor \text{score}/5 \rfloor\big)$ по 10-балльной шкале.

Вариант задания: участие в проекте «Мастерская знаний»

Дано:

- подборки, сгенерированные SciRus по одной статье
- асессорская разметка статей подборки по релевантности
- несколько вариантов токенизации
  — в том числе с автоматическим выделением терминов

Найти:

- тематическую модель
- модель ранжирования подборки по релевантности
- оптимальные: токенизацию, число тем, регуляризаторы
- распределение терминов по тематичности

Критерий:

- качество ранжирования
- (визуально) интерпретируемость тем
  — в том числе автоматического именования тем

Теоретическое задание к лекции 2

Упражнения на принцип максимума правдоподобия:

1. Униграммная модель документов: $p(w|d) = \xi_{dw}$
Найти параметры модели ξ_{dw}.

2. Униграммная модель коллекции: $p(w|d) = \xi_w$ для всех d
Найти параметры модели ξ_w.

Подсказка: применить условия ККТ или основную лемму.

3. (более творческое задание)
Предложите модель, определяющую роли слов в текстах:
— тематические слова
— специфичные слова документа (шум)
— слова общей лексики (фон)

Подсказка 1: искать распределение ролей слов $p(r|w)$, $r \in \{\text{т}, \text{ш}, \text{ф}\}$.
Подсказка 2: можно разреживать $p(r|w)$ для жёсткого определения ролей.
Подсказка 3: можно использовать документную частоту слов.

Теоретическое задание к лекции 3

4. Заменим log другой монотонно возрастающей функцией μ:

$$\sum_{d\in D}\sum_{w\in d} n_{dw}\mu\left(\sum_{t\in T}\phi_{wt}\theta_{td}\right) + R(\Phi,\Theta) \to \max_{\Phi,\Theta}$$

Как изменится EM-алгоритм? Возможно ли подобрать функцию μ так, чтобы сократился объём вычислений?

5. Заменим log монотонно возрастающей функцией μ в регуляризаторе сглаживания–разреживания (модель LDA):

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

Как изменится M-шаг и воздействие регуляризатора на модель?

6. Какому регуляризатору соответствует формула M-шага

$$\phi_{wt} = \operatorname*{norm}_{w}\left(n_{wt}[n_{wt} > \gamma n_t]\right)$$

Теоретическое задание к лекции 3

Аналитик построил тематическую модель Φ^0, Θ^0
и отметил среди столбцов матрицы Φ^0 темы двух типов:
удачные $T_+ \subset T$ и неудачные $T_- \subset T$.
Теперь он хочет построить модель ещё раз так, чтобы

- удачные темы остались в матрице Φ;
- остальные темы построились по-другому и были не похожи на каждую из неудачных тем $t \in T_-$.

7. Предложите регуляризаторы для этого.

8. Не получится ли так, что новые темы будут отдаляться от суммы неудачных тем $\sum_{t \in T_-} \phi^0_{wt}$ вместо того, чтобы отдаляться от каждой из неудачных тем по отдельности? Почему это плохо и как этого избежать?

9. Предложите способ инициализации Φ для новой модели.

Теоретическое задание к лекции 5

10. Для иерархической тематической модели с рег. $R(\Phi, \Psi)$ предложите способ разреживания матрицы связей $\Psi = \big(p(s|t)\big)$, гарантирующий, что
1) у каждой родительской темы будет хотя бы одна дочерняя;
2) у каждой дочерней темы будет хотя бы одна родительская.

Подсказка: можно придумывать критерий регуляризации, а можно — формулу М-шага для матрицы Ψ.

11. Предложите способ гарантировать, что если родительская тема t получает только одну дочернюю s, то она переходит в неё целиком и как распределение: $p(w|s) = p(w|t)$.

12. Предложите способ согласования вероятностных смесей $p(w|t) \approx \sum_{s \in S} p(w|s)p(s|t)$ и $p(t|d) \approx \sum_{s \in S} p(t|s)p(s|d)$ с учётом тождества $p(s|t)p(t) = p(t|s)p(s)$.