Темпоральная тематическая модель коллекции научных статей портала arxive.org

М.Д.Филин, К.В.Воронцов

maxapple@yandex.ru, vokov@forecsys.ru

Аннотация

Данная работа описывает процесс построения темпоральной тематической модели для большой текстовой коллекции научных статей. В работе используется подход аддитивной регуляризации и строится тематическая модель, учитывающая метки времени документов в виде отдельной модальности времени. Модель обучена на наборе из более чем 100000 научных статей за 2017 год.

Ключевые слова: *тематическая модель, аддитивная регуляризация, модальности, bigartm.*

1. Введение

Данная работа посвящена методам анализа тематической структуры большой текстовой коллекции и её динамики во времени.

Тематическая модель коллекции текстовых документов разбивает коллекцию на некоторое количество тем и определяет, к каким темам относятся документы, а также какие слова образуют каждую тему. Эта задача решается с помощью *вероятностного тематического моделирования* – пользователем фиксируется число тем, после чего модель находит распределения $\phi_{wt} = p(w|t)$ слов по темам и $\theta_{td} = p(t|d)$ тем по документам.

Классический подход к решению задачи тематического моделирования – это латентное размещение Дирихле (LDA), описанный в работе [6]. Этот метод предполагает, что плотности ϕ_{wt} и θ_{td} имеют распределение Дирихле, которое является в данном случае байесовским регуляризатором модели, который предотвращает переобучение. Но эта модель не даёт возможности для добавления требований различности тем или разреженности распределений ϕ_{wt} и θ_{td} и внесения других ограничений на модель.

Другим подходом, устраняющим эти ограничения, является аддитивная регуляризация тематических моделей (ARTM, [2]). Он позволяет записать любое количество дополнительных требований к тематической модели в виде взвешенной суммы критериев, добавляемых к основному функционалу логарифмированного правдоподобия. В [2] показано, что функционалы правдоподобия многих известных тематических моделей, таких как LDA и PLSA, допускают такое представление, то есть фактически являются частными случаями регуляризации. При этом, в отличие от стандартных задач машинного обучения, таких как классификация и регрессия, в тематическом моделировании возникает огромное разнообразие регуляризаторов, направленных на учёт различной дополнительной информации о текстовой коллекции.

Темпоральные тематические модели учитывают дополнительно метки времени y_d, привязанные к каждому документу d. Помимо распределений ϕ_{wt} и θ_{td}, вводится распределение каждой темы во времени $\xi_{yt} = p(y|t)$, что позволяет рассмотреть динамику изменения тем во времени.

Одним способом [7], [8] анализа тем во времени является разбиение исходной коллекции документов на пачки относящихся к одному временному интервалу и построение отдельной тематической модели для каждой пачки, с последующим анализом тем.

Способы явного включения времени в вероятностную модель чаще всего основаны на байесовском подходе: желаемые особенности модели добавляются с помощью указания априорных распределений на параметры. Можно выделить два направления: использование непрерывного априорного распределения $p(y|t)$ времени для каждой темы и модели с дискретным временем, основанные на Марковском свойстве. Относящаяся к первому классу модель TOT (Topics Over Time), [5] расширяет модель LDA, задавая априорное бета-распределение времени для каждой темы. Минусом этой модели является то, что в реальной жизни темы могут быть распределены совсем иначе и хочется найти их истиное распределение.

В данной работе с помощью подхода ARTM предлагается темпоральная тематическая модель, обученная на выборке из более чем 100000 научных статей портала arxiv.org, описывается использование различных модальностей для улучшения модели, проводится анализ получившейся модели.

2. Цель работы

Для коллекции научных статей построить темпоральную тематическую модель, учитывающую временную принадлежность каждого документа. Предполагается использование подхода аддитивной регуляризации. На основании заданных метрик качества подобрать оптимальные параметры модели. Сравнить влияние регуляризации и использования модальностей на качество моделей.

3. Постановка задачи

Пусть D – конечный набор текстов, называемый коллекцией, а W – набор слов, из которых состоят тексты (словарь). Для каждого слова w известно, сколько раз оно встречается в данном документе d, обозначим эту частоту как n_{dw}. Длину документа обозначим n_d. Предположим, что появление каждого слова в каждом документе связано с некоторой латентной переменной из некоторого множества тем T.

На множестве $D \times W \times T$ введём вероятностное пространство с плотностью $p(d, w, t)$. Примем *гипотезу условной независимости* – будем считать, что вероятность появления слова w, относящегося к теме t в документе d не зависит от документа и описывается общим для всей коллекции распределением:

$$p(w|d,t) \;=\; p(w|t). \tag{1}$$

Используя формулу полной вероятности и данную гипотезу, получаем:

$$p(w|d) = \sum_{t \in T} p(t|d)p(w|t). \tag{2}$$

Параметрами модели являются условные вероятности $\phi_{wt} \equiv p(w|t)$ и $\theta_{td} \equiv p(t|d)$. Известными данными в данной задаче является матрица $F = (\hat{p}(w|d))_{W \times D}$ частотных оценок вероятностей $p(w|d)$:

$$\hat{p}(w|d) = \frac{n_{dw}}{n_d}.$$

Построить *тематическую модель* коллекции D значит найти множество тем T и стохастические матрицы $\Phi = (\phi_{wt})_{W \times T}$ и $\Theta = (\theta_{td})_{T \times D}$, столбцы которых — распределения слов по темам и тем по документам. Поиск

этих матриц производится методом максимизации логарифма правдоподобия коллекции:

$$\log L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in W} n_{dw} \log \sum_{t \in T} \phi_{wt} \theta_{td} \longrightarrow \max_{\Phi, \Theta}. \tag{3}$$

При добавлении модальности времени вводится аналогичная гипотеза условной независимости:

$$p(y|d,t) \;=\; p(y|t), \tag{4}$$

где y – момент времени. По формуле полной вероятности выражаем распределение моментов времен по документам через смесь распределений:

$$p(y|d) = \sum_{t \in T} p(t|d) p(y|t). \tag{5}$$

Так как в задаче каждому документу d приписана метка времени $y_d \in Y$, есть эмпирическое распределение:

$$\hat{p}(y|d) = [y = y_d].$$

Аналогично ставя задачу оптимизации для матриц

$$\Xi = (\xi_{yt})_{Y \times T} \text{ и } \Theta = (\theta_{td})_{T \times D}$$

и взвешенно суммируя две функции правдоподобия, получаем общую задачу оптимизации для темпоральной тематической модели:

$$L(\Phi, \Theta, \Xi) = L_1(\Phi, \Theta) + \tau L_2(\Theta, \Xi), \tag{6}$$

$$\log L(\Phi, \Theta, \Xi) \longrightarrow \max_{\Phi, \Theta, \Xi} \tag{7}$$

Задача максимизации правдоподобия имеет бесконечно много локальных максимумов, что влечёт за собой неустойчивость модели.

В подходе ARTM [2] авторы предлагают в оптимизационной задаче (3) добавить к логарифму правдоподобия еще r функционалов: $R_i(\Phi, \Theta)$, $i = 1, \ldots, r$ называемых *регуляризаторами*, каждый со своим неотрицательным весом τ_i:

$$\begin{cases} R(\Phi, \Theta) = \sum_i \tau_i R_i(\Phi, \Theta), \qquad \log L(\Phi, \Theta) + R(\Phi, \Theta) \longrightarrow \max\limits_{\Phi, \Theta}, \\ \\ \phi_{wt} \geqslant 0, \quad \theta_{td} \geqslant 0, \quad \sum\limits_w \phi_{wt} = 1, \quad \sum\limits_t \theta_{td} = 1. \end{cases} \tag{8}$$

4. Метрики для оценки модели

От распределений ϕ_{wt} и θ_{td}, полученных в ходе построения тематической модели, требуется обладание многими полезными свойствами: разреженностью — большим числом нулей, отсутствием фоновых слов в предметных темах, различностью предметных тем друг от друга, плавностью изменения тем во времени, а главное — интерпретируемостью.

Будем следить за набором дополнительных метрик, позволяющих наблюдать за процессом сходимости и определять, обладают ли искомые распределения ϕ_{wt} и θ_{td} перечисленными свойствами.

1) **Перплексия** — величина, выражающаяся через правдоподобие выборки и позволяющая отслеживать сходимость метода оптимизации:

$$\text{Perplexity}(\Phi, \Theta) \;=\; \exp\Big(-\frac{1}{n}\sum_{d\in D}\sum_{w\in W} n_{dw}\log p(w|d)\Big),$$

$$p(w|d) \;=\; \sum_{t\in T}\phi_{wt}\theta_{td}, \quad n \;\equiv\; \sum_{d\in D}\sum_{w\in W} n_{dw}.$$

   Численное значение перплексии не имеет интерпретации и позволяет лишь сравнивать алгоритмы между собой. Значения чем меньше, тем лучше.

2) **Разреженность матриц Φ и Θ** — доля нулевых элементов. В предметных темах разреженность достигает $90-95\%$, поэтому, для хорошей тематической модели разреженность необходима.

Лексическим *ядром темы* будем называть множество слов, отличающих данную тему от остальных:

$$W_t \;=\; \{w \in W \mid p(t|w) > \delta\}.$$

$$p(t|w) = \phi_{wt}\frac{n_t}{n_w},$$

$$n_t = \sum_{d\in D}\sum_{w\in d} n_{tdw}$$

$$n_w = \sum_{d\in D} n_{dw}$$

На основе ядра темы строятся следующие две оценки:

3) **Чистота** — суммарная вероятность слов ядра:

$$\text{Purity}(t) \;=\; \sum_{w \in W_t} p(w|t) \;=\; \sum_{w \in W_t} \phi_{wt}$$

показывает насколько хорошо тема описывается своим ядром. Чем выше, тем лучше.

4) **Контрастность** — средняя вероятность встретить слова ядра в конкретной теме:

$$\text{Contrast}(t) \;=\; \frac{1}{|W_t|} \sum_{w \in W_t} p(t|w)$$

При большой контрастности тема однозначно угадывается по своему ядру, при малой – тема размывается, становится нечеткой.

5. Эксперимент

5.1. Исходные данные. В качестве исходных данных для создания тематической модели был использован корпус научных статей портала **archive.org**. Портал предоставляет бесплатный доступ к различным материалам, в частности — к научным статьям. Все данные хранятся в облачном хранилище **Amazon S3**.

Нас будут интересовать научные статьи, относящиеся к шести большим разделам науки, представленным на портале:

- физика
- математика
- компьютерные науки
- финансы
- биология
- статистика

Все данные архива хранятся в облачном хранилище Amazon S3. Нужные нам статьи содержатся в формате pdf. Они заархивированы и распределены по пакетам, каждый из которых весит порядка 500 мегабайт. Для формирования корпуса текстовых документов был автоматизирован процесс загрузки и обработки данных, а также создан модуль, собирающий метаинформацию для обработанных документов, включающую сведения об авторах, времени публикации, разделе науки, к которому относится документ и т.д. В качестве осоновы использовались результаты работы [3]. Для извлечения текста из pdf файлов был использован инструмент **XpdfReader [4]**, способный также доставать дополнительную информацию о файлах и все изображения (может быть использовано в дальнейшем).

5.2. Подготовка данных. Для того, чтобы можно было приступить к анализу текстового корпуса необходимо провести обработку данных. Исходные текстовые файлы содержат очень «сырой» текст, содержащий некорректно конвертированные из pdf в txt слова, опечатки, сноски, пометки. Необходимо очистить текст от всего лишнего и в дальнейшем работать уже с подготовленными данными. Для каждого текстового документа выполняются следующие действия:

1) Все слова приводятся к нижнему регистру.

2) Удаляются ненужные символы окончания строк и обрабатываются переносы слов.
3) Текст разделяется на слова с помощью регулярных выражений python — специальных шаблонов для поиска подстроки в тексте. Таким образом, мы избавляемся от всех формул, гиперссылок, обозначений параграфов, чисел и других объектов, не представляющих смысловой нагрузки.
4) Удаляются слова, длина которых меньше трех или больше двадцати символов, поскольку короткие слова, такие как предлоги и союзы, не будут отражать общей картины при анализе текстов, а очень длинные слова вероятнее всего появились в результате склеивания при неправильном конвертировании.
5) Удаляются слова, в которых встречаются три и более подряд идущих одинаковых символов.
6) Удаляются стоп-слова — слова, встречающиеся почти в каждом документе. Для получения списка стоп-слов используется библиотека **ntlk**.
7) Проводится стемминг оставшихся слов. Стемминг — процесс нахождения основы слова (неизменяемой части, которая необязательно совпадает с морфологическим корнем) для заданного исходного слова. Для нормализации текста был выбран стеммер из библиотеки **Snowball**.

Для лучшей интерпретируемости тем наряду с обычными токенами (униграммами) было решено использовать биграммы и триграммы. Для решения данной задачи применяется алгоритм **TopMine** (рис. 1).

В итоге для каждого документа были получены словари униграм, биграм и триграм. Каждый вид терма был соотнесен с отдельной модальностью. Общее количество обработанных документов — 111924. Размер получившегося словаря токенов всех модальностей —100892.

5.3. Базовая модель. Заданное количество тем для эксперимента - 200. Для выделения фона и слов общей лексики было отведено 5 тем (backgroung topics). В качестве базовой модели будем строить тематическую модель используя только униграммы и без регуляризатров. Для оценки качества модели рассматриваем слежующие метрики: разреженность матриц Φ и Θ, перплексия, контрастность и чистота ядер тем.

Вход: коллекция D, пороги ε_k;
Выход: хэш-таблица частот $C(a_1, \ldots, a_k)$, $k = 1, \ldots, k_{\max}$;

$A_{d,1} := \{1, \ldots, n_d\}$;
$C(w) := n_w$ для всех $w \in W$ таких, что $n_w \geqslant \varepsilon_1$;
для $k := 2, \ldots, k_{\max}$ **пока** $D \neq \varnothing$
  для всех $d \in D$
    $A_{d,k} := \{i \in A_{d,k-1} \mid C(w_{d,i}, \ldots, w_{d,i+k-2}) \geqslant \varepsilon_k\}$;
    если $A_{d,k} = \varnothing$ **то** $D := D \backslash \{d\}$;
    для всех $i \in A_{d,k}$
      если $i + 1 \in A_{d,k}$ **то** $++C(w_{d,i}, \ldots, w_{d,i+k-1})$;
  оставить только частые k-граммы: $C(a_1, \ldots, a_k) \geqslant \varepsilon_k$;

Рис. 1

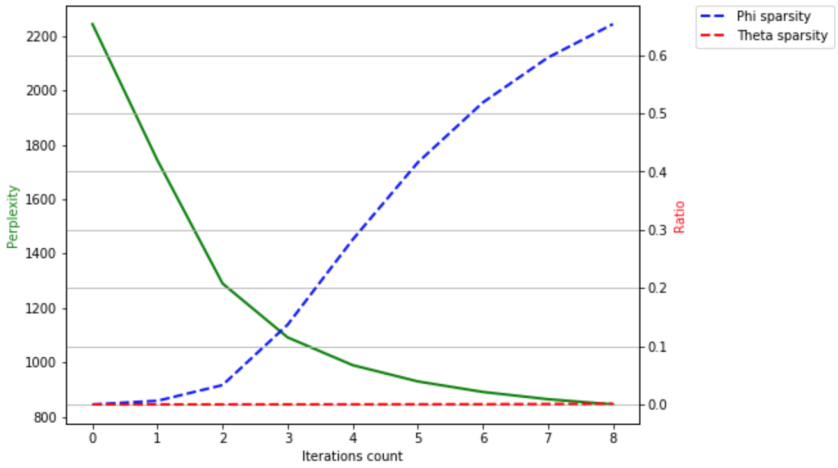
После 10 эпох EM-алгоритма получаем следующие результаты (рис. 2).

5.4. Использование регуляризаторов. Добавим набор регуляризаторов для улучшения модели. Испоьзуем подход аддитивной регуляризации.

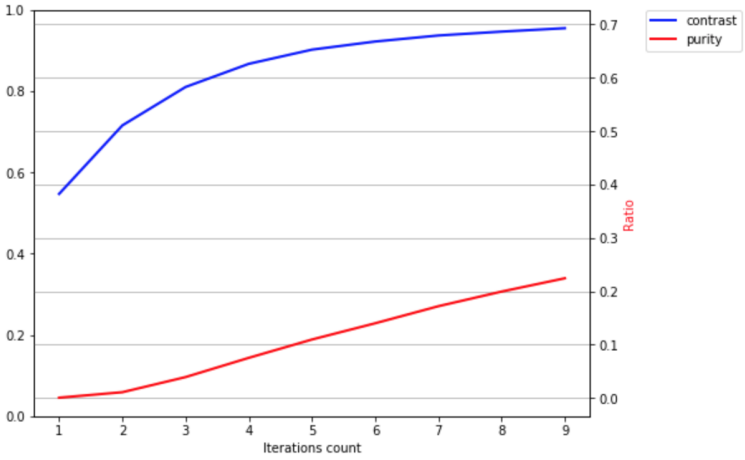
Тактика использования регуляризаторов:

1) Декореллирующий регуляризатор предметных тем матрицы Φ (для различности предметных тем).
2) Сглаживающий регуляризатор фоновых тем матрицы Φ
3) Разреживающий регуляризатор предметных тем матрицы Φ (для выделения более четких ядер тем)
4) Сглаживающий регуляризатор фоновых тем матрицы Θ
5) Разреживающий регуляризатор предметных тем матрицы Θ

В результате мы получили хорошие значения разреженности матриц Φ и Θ, а также улучшили значения контрастности и чистоты тем (рис. 3).



(a) График перплексии и разреженности матриц Φ и Θ



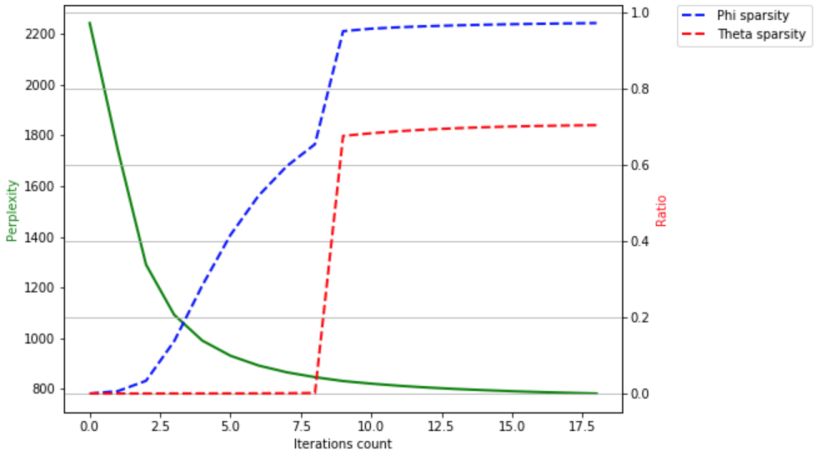
(b) График чистоты и контрастности тем

Рис. 2. Графики для униграммной тематической модели без регуляризаторов

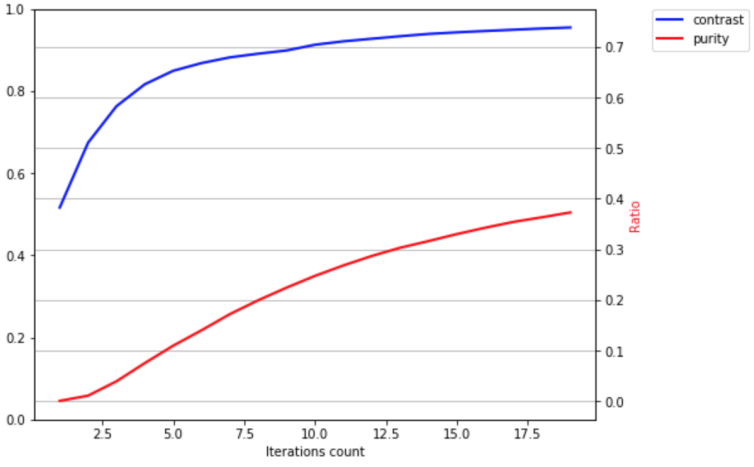
5.5. Добавление времени. В рамках основного эксперимента к предыдущей модели были добавлены метки времени документов. Меткой времени в данном случае являлся месяц публикации статьи. Наряду с описанными выше регуляризаторами для модальности времени будем использовать регуляризатор разреживания предметных тем и регуляризатор сглажи-

вания фоновых тем. Спустя 20 эпох имеем следующие результаты (рис. 4).

Используя ngramm-ы в качестве отдельных модальностей можно добиться достаточно хорошо интерпретируемых тем (рис. 5).

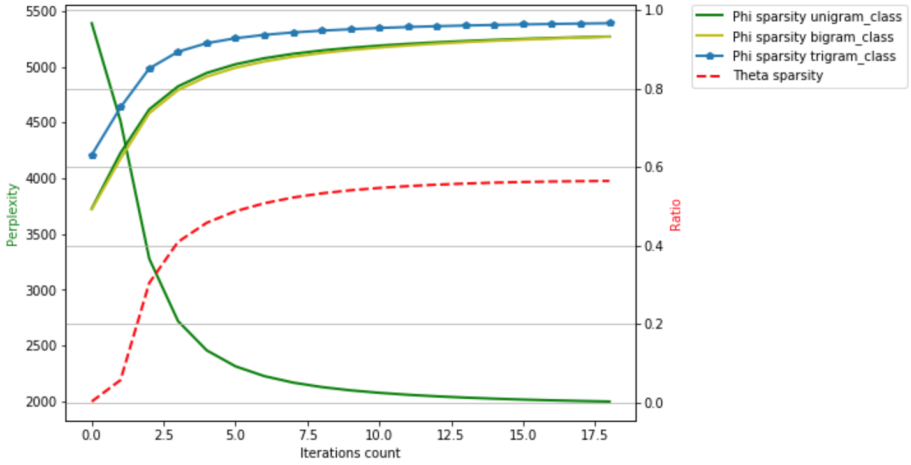


(a) График перплексии и разреженности матриц Φ и Θ

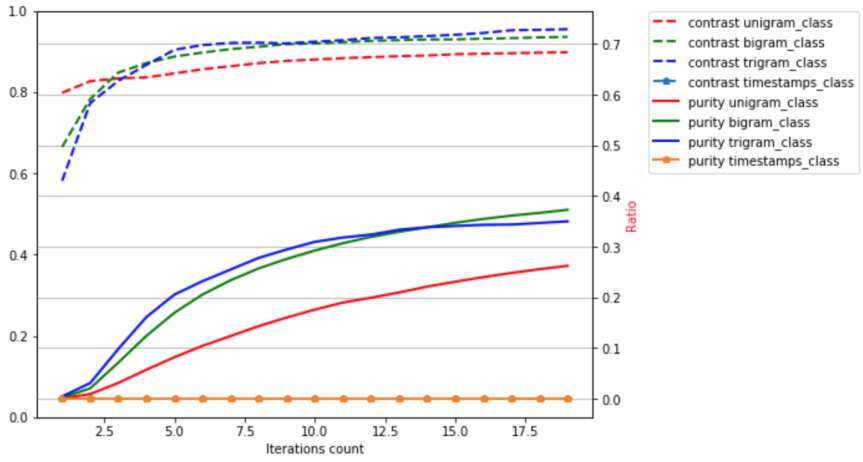


(b) График чистоты и контрастности тем

Рис. 3. Графики для униграммной тематической с использованием регуляризаторов



(a) График перплексии и разреженности матриц Φ и Θ



(b) График чистоты и контрастности тем

Рис. 4. Графики для тематической модели с 4-мя модальностями (униграммы, биграммы, триграммы, временные метки) с использованием регуляризаторов

Получив распределния тем по временным отрезкам можно выделить событийные темы — темы, которые имеют явные всполохи лишь в небольшом количестве временных отрезков. Если упорядочить матрицу timestamsp-topics, то при визуализации видно, что темы в перых столбцах сконцентрированы в определенных временных отрезках (рис. 6).

topic_d2:

scheme, discret, method, mesh, problem, numer, comput, element, approxim, solut
finit_element, basi_function, time_step, numer_method, numer_solut, high_order, error_estim, second_order, finit_diff
er, numer_experi
finit_element_method, partial_differenti_equat, discontinu_galerkin_method, posteriori_error_estim, comput_method_app
li, finit_differ_method, reduc_order_model, finit_differ_scheme, finit_element_space, finit_element_approxim

topic_d3:

particl, collis, hydrodynam, veloc, relativist, momentum, fluid, equilibrium, effect, mass
distribut_function, boltzmann_equat, energi_densiti, kinet_theori, equat_state, singl_particl, initi_condit, transpor
t_coeffici, shear_viscos, particl_move
phys_rev_lett, mean_free_path, energi_momentum_tensor, heavi_ion_collis, quark_gluon_plasma, phys_rev_arxiv, phase_sp
ace_distribut, per_unit_volum, intern_degre_freedom, relativist_heavi_ion

topic_d4:

sensor, vehicl, speech, audio, music, speaker, sound, acoust, base, grasp
speech_recognit, signal_process, acoust_speech, ieee_transact, sensor_data, sensor_network, time_frequenc, error_rate
, sourc_separ, activ_recognit
recurr_neural_network, ieee_intern_confer, signal_process_icassp, hidden_markov_model, deep_neural_network, ieee_sign
al_process, speech_signal_process, short_term_memori, automat_speech_recognit, onlin_avail_http

Рис. 5. Примеры выделенных тем

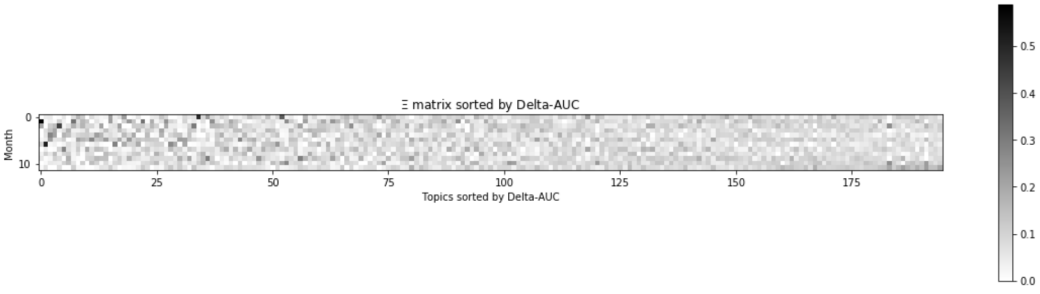


Рис. 6

ЛИТЕРАТУРА

1. Воронцов К. В. Вероятностное тематическое моделирование. Москва, 2009. — 2013. – URL: http://www.machinelearning.ru/wiki/images/2/22/Voron-2013-ptm.pdf
2. Воронцов К. В. Аддитивная Регуляризация Тематических Моделей Коллекций Текстовых Документов. — Доклады РАН, Т.455, №3. С.268-271, 2014
3. Филин. М. Д. Система информационного поиска на основе тематического моделирования. Сборник научных XIX Международной конференция DAMDID/RCDL'2017. С.306-309, 2017.
4. XpdfReader — a free PDF toolkit. URL:http://xpdfreader.com
5. Xuerui Wang, Andrew McCallum. Topics over Time: A Non-Markov Continuous-Time Model of Topical Trends — URL: https://people.cs.umass.edu/ mccallum/papers/tot-kdd06.pdf
6. Blei, David M and Ng, Andrew Y and Jordan, Michael I. Latent dirichlet allocation — the Journal of machine Learning research, 2003
7. Griffiths T. L., Steyvers M. Finding scientific topics. Proceedings of the National Academy of Sciences. 2004
8. Hall D., Jurafsky D., Manning C. D. Studying the history of ideas using topic models. Proceedings of the conference on empirical methods in natural language processing / Association for Computational Linguistics. 2008