Анализ текстов и вероятностные тематические модели

К. В. Воронцов

22 марта 2013 г.

Содержание

1 Вероятностные тематические модели

Тематическое моделирование (topic modeling) — одно из современных приложений машинного обучения к анализу текстов, активно развивающееся с конца 90-х годов. *Тематическая модель* (topic model) коллекции текстовых документов определяет, к каким темам относится каждый документ и какие слова (термины) образуют каждую тему.

Вероятностная тематическая модель (BTM) описывает каждую тему дискретным распределением на множестве терминов, каждый документ — дискретным распределением на множестве тем. Предполагается, что коллекция документов — это последовательность терминов, выбранных случайно и независимо из смеси таких распределений, и ставится задача восстановления компонент смеси по выборке. В большинстве приложений требуется определить также и число тем.

Поскольку документ или термин может относиться одновременно ко многим темам с различными вероятностями, говорят, что BTM осуществляет «мягкую» кластеризацию документов и терминов по кластерам-темам. Тем самым решаются проблемы синонимии и полисемии (многозначности) терминов, возникающие при обычной «жёсткой» кластеризации. Синонимы, часто употребляющиеся в схожих контекстах, с большой вероятностью попадают в одну тему. Многозначные слова, употребляющиеся в разных контекстах, распределяются между несколькими темами соответственно частоте их употребления.

Тематические модели применяются для выявления трендов в научных публикациях или новостных потоках [39, 28], для классификации и категоризации документов [23] и изображений [34, 14], для семантического информационного поиска [36], в том числе многоязычного [29], для тегирования веб-страниц [19], для обнаружения текстового спама [4], в рекомендательных системах [35] и других приложениях. Для конкретности будем рассматривать BTM, применяемые для тематического поиска по коллекциям научных публикаций.

Для этого надо будет ввести функционал качества поиска ToDo[1]

В информационном поиске документы принято представлять векторами, координаты которых соответствуют словам, а значения — статистическим характеристикам слов, например частотам или tf-idf. Поиск документов по коротким запросам реализуется путём поиска векторов, в которых часто встречаются слова запроса [3]. Тематическая модель позволяет использовать тот же механизм для поиска документов схожей тематики по целому документу или по длинному фрагменту текста. При этом документы представляются векторами тем, а не векторами слов. Векторами тем представляются также связанные с документами объекты: термины, рисунки, авторы, научные группы, организации, конференции, журналы, сайты и т. д., что позволяет задавать в качестве запроса любой объект или совокупность объектов и искать по ним объекты того же или другого типа, имеющие схожую тематику.

Вероятностные тематические модели могут учитывать марковские зависимости в последовательностях терминов, связи между документами через авторство или ссылки, плавные изменения тематики во времени, иерархические отношения между темами и другие особенности текстовых коллекций. Многочисленные разновидности вероятностных тематических моделей описаны в обзоре [11].

§1.1 Вероятностная модель коллекции текстовых документов

Пусть D — множество (коллекция) текстовых документов, W — множество (словарь) всех употребляемых в них терминов (слов или словосочетаний). Каждый документ $d \in D$ представляет собой последовательность n_d терминов $(w_1, \ldots, w_{n_d})$ из словаря W. Термин может повторяться в документе много раз.

Вероятностное пространство и гипотеза независимости. Предполагается, что существует конечное множество тем T, и каждое употребление термина w в каждом документе d связано с некоторой темой $t \in T$, которая не известна. Коллекция документов рассматривается как множество троек (d, w, t), выбранных *случайно и независимо* из дискретного распределения $p(d, w, t)$, заданного на конечном множестве $D \times W \times T$. Документы $d \in D$ и термины $w \in W$ являются наблюдаемыми переменными, тема $t \in T$ является *латентной* (скрытой) переменной.

Гипотеза о независимости элементов выборки эквивалентна предположению, что порядок терминов в документах не важен для выявления тематики, то есть тематику документа можно узнать даже после произвольной перестановки терминов, хотя для человека такой текст теряет смысл. Это предположение называют гипотезой «мешка слов» (bag of words). Порядок документов в коллекции также не имеет значения; это предположение называют гипотезой «мешка документов».

Приняв гипотезу «мешка слов», можно перейти к более компактному представлению документа как подмножества $d \subset W$, в котором каждому элементу $w \in d$ поставлено в соответствие число n_{dw} вхождений термина w в документ d.

Постановка задачи тематического моделирования. Построить *тематическую модель* коллекции документов D — значит найти множество тем T, распределения $p(w \mid t)$ для всех тем $t \in T$ и распределения $p(t \mid d)$ для всех документов $d \in D$. Можно также говорить о задаче совместной мягкой кластеризации множества документов и множества слов по одному и тому же множеству кластеров-тем.

Построенные распределения используются затем для решения прикладных задач. В частности, распределение $p(t \mid d)$ является удобным признаковым описанием документов для решения задач тематического поиска, классификации и категоризации документов.

Гипотеза условной независимости. Будем полагать, что распределения вероятностей терминов в теме t одинаковы во всех документах $d \in D$. Это предположение, называемое гипотезой условной независимости, допускает три эквивалентных представления:

$$p(w \mid d, t) = p(w \mid t); \qquad p(d \mid w, t) = p(d \mid t); \qquad p(d, w \mid t) = p(d \mid t)p(w \mid t). \tag{1.1}$$

Вероятностная модель порождения данных. Согласно определению условной вероятности, формуле полной вероятности и гипотезе условной независимости

$$p(w \mid d) = \sum_{t \in T} p(t \mid d)\, p(w \mid t), \tag{1.2}$$

Алгоритм 1.1. Порождение коллекции текстов с помощью вероятностной модели.

Вход: распределения $p(w \mid t)$, $p(t \mid d)$;
Выход: выборка пар (d_i, w_i), $i = 1, \ldots, n$;

1: **для всех** $d \in D$
2: задать длину n_d документа d;
3: **для всех** $i = 1, \ldots, n_d$
4: выбрать случайную тему t из распределения $p(t \mid d)$;
5: выбрать случайный термин w из распределения $p(w \mid t)$;
6: добавить в выборку пару (d, w), при этом тема t «забывается»;

где $p(t \mid d)$ и $p(w \mid t)$ — искомые распределения. Согласно модели порождения данных (1.2), коллекция D — это выборка наблюдений (d, w), генерируемых Алгоритмом 1.1.

Гипотеза разреженности. Каждый документ d и каждый термин w связан, как правило, с небольшим числом тем t. Поэтому значительная доля вероятностей $p(t \mid d)$ и $p(w \mid t)$ обращается в нуль.

Если документ относится к большому числу тем (например, энциклопедия, журнал, сборник статей), то в задачах тематического поиска или классификации документов его имеет смысл разбивать на части, более однородные по тематике. Если термин относится к большому числу тем, то, скорее всего, это общеупотребительное слово, бесполезное для определения тематики.

Частотные (выборочные) оценки вероятностей. Вероятности, связанные с наблюдаемыми переменными d и w, можно оценивать по выборке как частоты (здесь и далее выборочные оценки вероятностей p будем обозначать через $\hat{p}$):

$$\hat{p}(d, w) = \frac{n_{dw}}{n}, \qquad \hat{p}(d) = \frac{n_d}{n}, \qquad \hat{p}(w) = \frac{n_w}{n}, \qquad \hat{p}(w \mid d) = \frac{n_{dw}}{n_d}, \tag{1.3}$$

n_{dw} — число вхождений термина w в документ d;

$n_d = \sum\limits_{w \in W} n_{dw}$ — длина документа d в терминах;

$n_w = \sum\limits_{d \in D} n_{dw}$ — число вхождений термина w во все документы коллекции;

$n = \sum\limits_{d \in D} \sum\limits_{w \in d} n_{dw}$ — длина коллекции в терминах.

Вероятности, связанные со скрытой переменной t, также можно оценивать как частоты, если рассматривать коллекцию документов как выборку троек (d, w, t):

$$\hat{p}(t) = \frac{n_t}{n}, \qquad \hat{p}(w \mid t) = \frac{n_{wt}}{n_t}, \qquad \hat{p}(t \mid d) = \frac{n_{dt}}{n_d}, \qquad \hat{p}(t \mid d, w) = \frac{n_{dwt}}{n_{dw}}, \tag{1.4}$$

n_{dwt} — число троек, в которых термин w документа d связан с темой t;

$n_{dt} = \sum\limits_{w \in W} n_{dwt}$ — число троек, в которых термин документа d связан с темой t;

$n_{wt} = \sum\limits_{d \in D} n_{dwt}$ — число троек, в которых термин w связан с темой t;

$n_t = \sum\limits_{d \in D} \sum\limits_{w \in d} n_{dwt}$ — число троек, связанных с темой t.

В пределе $n \to \infty$ частотные оценки $\hat{p}(\cdot)$, определяемые формулами (1.3)–(1.4), стремятся к соответствующим вероятностям $p(\cdot)$.

Связь с задачами неотрицательного матричного разложения. Если число тем $|T|$ много меньше числа документов $|D|$ и числа терминов $|W|$, то равенство (1.2) можно понимать как задачу приближённого представления заданной матрицы частот $F = \big(\hat{p}_{wd}\big)_{W \times D}$, где $\hat{p}_{wd} = \hat{p}(w \mid d) = n_{dw}/n_d$, в виде произведения $F \approx \Phi\Theta$ двух неизвестных матриц меньшего размера — *матрицы тем* $\Phi = (\varphi_{wt})_{W \times T}$, $\varphi_{wt} = p(w \mid t)$ и *матрицы документов* $\Theta = (\theta_{td})_{T \times D}$, $\theta_{td} = p(t \mid d)$.

Одно из таких представлений строится из $|T|$ главных компонент сингулярного разложения матрицы F и является решением задачи наименьших квадратов

$$\sum_{d \in D} \sum_{w \in W} \big(\hat{p}_{wd} - p(w \mid d)\big)^2 = \sum_{d \in D} \sum_{w \in W} \Big(\hat{p}_{wd} - \sum_{t \in T} \varphi_{wt}\theta_{td}\Big)^2 = \|F - \Phi\Theta\|^2 \to \min_{\Theta, \Phi}. \tag{1.5}$$

Хотя сингулярное разложение имеет массу приложений в анализе данных, оно по ряду причин плохо подходит для приближения вероятностных распределений. Во-первых, столбцы получаемых матриц Θ и Φ не удовлетворяют условиям неотрицательности и нормировки, поэтому их нельзя интерпретировать как распределения. Во-вторых, квадратичная функция потерь не чувствительна к малым различиям «хвостов» распределений, из-за которых их статистические свойства могут различаться существенно.

В вероятностном тематическом моделировании вместо принципа наименьших квадратов используется принцип максимума правдоподобия. Он также приводит к задаче матричного разложения вида (1.5), только вместо евклидовой нормы используется взвешенная дивергенция Кульбака–Лейблера.

1.1.1 Метод максимума правдоподобия

Для оценивания параметров Θ, Φ тематической модели по коллекции документов D будем максимизировать правдоподобие (плотность распределения) выборки:

$$p(D; \Theta, \Phi) = C \prod_{d \in D} \prod_{w \in d} p(d, w)^{n_{dw}} = \prod_{d \in D} \prod_{w \in d} p(w \mid d)^{n_{dw}} \underbrace{C p(d)^{n_{dw}}}_{\text{const}} \to \max_{\Theta, \Phi},$$

где C — нормировочный множитель мультиномиального распределения, зависящий только от чисел n_{dw}. Отбросим множители C и $p(d)$, не влияющие на положение точки

максимума, подставим выражение для $p(w\,|\,d)$ из (1.2) и воспользуемся обозначениями $\theta_{td} = p(t\,|\,d)$, $\varphi_{wt} = p(w\,|\,t)$. Прологарифмировав правдоподобие, получим задачу максимизации

$$L(D;\Theta,\Phi) = \sum_{d\in D}\sum_{w\in d} n_{dw} \ln \sum_{t\in T} \varphi_{wt}\theta_{td} \;\to\; \max_{\Theta,\Phi} \tag{1.6}$$

при ограничениях неотрицательности $\theta_{td} \geqslant 0$, $\varphi_{wt} \geqslant 0$ и нормировки

$$\sum_{w\in W} \varphi_{wt} = 1, \quad \sum_{t\in T} \theta_{td} = 1.$$

Заметим, что максимизация (1.6) эквивалентна минимизации взвешенной суммы расстояний Кульбака–Лейблера $\mathrm{KL}(\hat{p}\|p) = \sum_w \hat{p}_{wd} \ln \frac{\hat{p}_{wd}}{p(w\,|\,d)}$ между эмпирическими распределениями $\hat{p}_{wd}$ и модельными $p(w\,|\,d)$ по всем документам d из D:

$$\sum_{d\in D} n_d \,\mathrm{KL}\Big(\frac{n_{dw}}{n_d} \,\Big\|\, \sum_{t\in T} \varphi_{wt}\theta_{td}\Big) \;\to\; \min_{\Theta,\Phi},$$

где весом документа d является его длина n_d. Если веса n_d убрать, то все документы будут искусственно приведены к одинаковой длине. Такая модификация функционала качества полезна при моделировании коллекций, содержащих документы одинаковой важности, но существенно разной длины.

1.1.2 Предварительная обработка данных

Понятие «термина» может изменяться в зависимости от целей построения тематической модели и таких особенностей задачи, как язык документов, средняя длина документов, тематика коллекции.

Лемматизация и стемминг. При построении тематической модели нет смысла различать формы (склонения, спряжения) одного и того же слова. Это приведёт к неоправданному разрастанию словаря, дроблению статистики, увеличению ресурсоёмкости и снижению качества модели.

Лемматизация — это приведение каждого слова в документе к его нормальной форме. В русском языке нормальными формами считаются: для существительных — именительный падеж, единственное число; для прилагательных — именительный падеж, единственное число, мужской род; для глаголов, причастий, деепричастий — глагол в инфинитиве. Существуют специальные программы — *лемматизаторы* (lemmatizer), обычно основанные на явном хранении грамматического словаря со всеми формами слов. Недостатком лемматизации является трудоёмкость составления словарей, и, как следствие, их неполнота, особенно по части специальной терминологии и неологизмов, которые во многих приложениях как раз и представляют наибольший интерес.

ссылка на рекомендуемые русский и английский лемматизаторы ToDo[2]

Стемминг — это более простая технология, которая состоит в отбрасывании изменяемых частей слов, главным образом, окончаний. Она не требует хранения словаря всех слов и основана на правилах морфологии языка. Недостатком стемминга является большее число ошибок. Стемминг хорошо подходит для английского языка, но хуже подходит для русского.

ссылка на рекомендуемые русский и английский стеммеры ToDo[3]

Отбрасывание стоп-слов. Слова, встречающиеся во многих текстах различной тематики, бесполезны для тематического моделирования, и могут быть отброшены. К ним относятся предлоги, союзы, числительные, местоимения, некоторые глаголы, прилагательные и наречия. Число таких слов обычно варьируется в пределах нескольких сотен. Их отбрасывание почти не влияет на длину словаря, но может приводить к заметному сокращению длины некоторых текстов.

Отбрасывание редких слов. Слова, встречающиеся в длинном документе слишком редко, например, только один раз, также можно отбрасывать, полагая, что данное слово не характеризует тематику данного документа. При обработке коллекций коротких новостных сообщений этот приём лучше не использовать.

Выделение ключевых фраз. При обработке коллекций научных, юридических или других специальных текстов вместо отдельных слов выделяют *ключевые фразы* — словосочетания, являющиеся устойчивыми оборотами или терминами в данной предметной области. Это отдельная довольно сложная задача, для решения которой используются тезаурусы, составленные экспертами [2], либо методы машинного обучения [21, 40], при этом для формирования обучающих выборок всё равно приходится привлекать экспертов.

Далее будем полагать, что словарь W получен в результате предварительной обработки всех документов коллекции D и может содержать как отдельные слова, так и ключевые фразы. Элементы словаря $w \in W$ будем называть «терминами».

1.1.3 Оценивание качества тематических моделей

Оценивание качества тематических моделей является нетривиальной проблемой. В отличие от задач классификации или регрессии здесь нет чёткого понятия «ошибки» или «потери». Стандартные критерии качества кластеризации типа средних внутрикластерных или межкластерных расстояний или их отношений плохо подходят для оценивания «мягкой» совместной кластеризации документов и терминов.

Наиболее распространённым критерием является *перплексия* (perplexity), используемая для оценивания моделей языка в компьютерной лингвистике. Это мера несоответствия или «удивлённости» модели $p(w \mid d)$ терминам w, наблюдаемым в документах d коллекции D, определяемая через логарифм правдоподобия (1.6):

$$\mathscr{P}(D; p) = \exp\left(-\frac{1}{n}\sum_{d \in D}\sum_{w \in d} n_{dw} \ln p(w \mid d)\right). \tag{1.7}$$

Чем меньше эта величина, тем лучше модель p предсказывает появление терминов w в документах d коллекции D.

Интерпретация перплексии. Если термины w порождаются из равномерного распределения $p(w) = 1/V$ на словаре мощности V, то перплексия модели p на таком тексте сходится к V с ростом его длины. Чем сильнее распределение p отличается от равномерного, тем меньше перплексия. Чем сильнее модель p отличается от генерирующего распределения, тем больше перплексия. В нашем случае в (1.7) используются условные вероятности терминов $p(w \mid d)$, и интерпретация немного другая: если

каждый документ генерируется из V равновероятных терминов (возможно, различных в разных документах), то перплексия сходится к V. Опять-таки, чем сильнее распределение отличается от равномерного, тем меньше перплексия.

Чтобы сравнение перплексии двух коллекций было корректным, необходимо, чтобы они имели один и тот же словарь. ToDo[4]

Чтобы перплексия была характеристикой только качества модели, необходимо вводить нормировки, чтобы длины документов и эффективная мощность словаря не влияли на перплексию. ToDo[5]

Перплексия контрольной выборки. Обозначим через $p_D(w \mid d)$ модель, построенную по обучающей коллекции документов D. Перплексия обучающей выборки $\mathscr{P}(D; p_D)$ является оптимистично смещённой (заниженной) характеристикой качества модели из-за эффекта переобучения. Обобщающую способность модели принято оценивать *перплексией контрольной выборки* (hold-out perplexity) $\mathscr{P}(D'; p_D)$.

Вопрос о том, как разделить исходную коллекцию на обучение D и контроль D', не тривиален. К сожалению, детали этой процедуры во многих статьях опускаются. В [6] предлагается разделять все документы на обучающие и контрольные случайным образом в пропорции $9:1$. Однако в силу гипотез «мешка слов» и «мешка документов» более корректным было бы случайное разбиение каждого документа на обучающую и контрольную части. С другой стороны, во многих приложениях важно проверить способность тематической модели хорошо описывать новые документы.

Новые документы порождают две проблемы: во-первых, для них необходимо оценивать θ_{td}; во-вторых, они могут содержать новые термины w, для которых придётся оценивать также φ_{wt}, увеличивать размерность векторов $\varphi_t = (\varphi_{wt})_{w \in W}$ и перенормировать их. Такая процедура оценивания модели частично включает в себя процедуру обучения, в результате чего оценка качества снова может оказаться оптимистично смещённой.

Частичное решение этой проблемы предлагается в [5]. После обучения модели p_D векторы φ_t фиксируются, векторы θ_d контрольных документов $d \in D'$ оцениваются по первой половине каждого документа, по вторым половинам вычисляется контрольная перплексия. Что такое «половина», не уточняется. Простое разрезание текста на две части может приводить к смещённым оценкам. Например, научные статьи обычно начинаются с введения и обзора, использующих общую терминологию, затем идёт изложение частных результатов. Если в коллекции много таких текстов, то оценка окажется пессимистично смещённой. Противоположный пример неслучайного разбиения текста — когда число вхождений каждого термина n_{dw} делится ровно пополам между обучающей и контрольной выборками. В таком случае обучающая и контрольная половины документа будут неразличимы для тематической модели, и оценка окажется оптимистично смещённой.

В наших экспериментах последовательность терминов $\{w_1, \ldots, w_{n_d}\}$ каждого контрольного документа $d \in D'$ после случайной перестановки разбивается на две части равной длины. Новые слова, попадающие во вторую часть, игнорируются.

Ещё один выход — робастные модели. Новые редкие слова считаются шумом, описываются униграммной моделью и почти не дают вклада в контрольную перплексию. Робастную модель трудно удивить новыми словами, т.к. она трактует их как шум. ToDo[6]

Более сложные процедуры несмещённого оценивания правдоподобия предложены в [32] и улучшены в [7]. Они имеют трудоёмкость, квадратичную по длине документа, и в процессе оценивания используют ту же тематическую модель, качество которой оценивается. Эти недостатки несколько ограничивают их применимость.

Эксперименты на модельных данных. Алгоритм 1.1 можно использовать для генерации модельных данных по заданным распределениям $p(w\,|\,t)$ и $p(t\,|\,d)$. Это крайне полезно на стадии тестирования методов обучения тематических моделей, решающих задачу (1.6). Хороший метод должен быть способен восстановить по данным ту самую модель, которая эти данные породила. Модельные данные можно генерировать различной длины n; можно добавлять в них шум — случайные пары (d_i, w_i) из распределения, заведомо плохо приближаемого моделью (1.2); можно задавать распределения $p(w\,|\,t)$, $p(t\,|\,d)$ более различными или более похожими, тем самым делая задачу восстановления модели более лёгкой или более трудной; задавать различное число тем $|T|$, а восстанавливать модель при другом числе тем, либо пытаться его определить. Эксперименты с варьированием модели данных позволяют исследовать устойчивость метода и узнать границы его применимости. Только в случае модельных данных известно, какая тема t_i на самом деле связана с каждой парой (d_i, w_i), что позволяет оценивать качество восстановления модели по данным как долю правильно угаданных тем или как расстояние между восстановленными и истинными распределениями $p(w\,|\,t)$, $p(t\,|\,d)$.

Показать эксперименты на модельных данных ToDo[7]

§1.2 Вероятностный латентный семантический анализ

Вероятностный латентный семантический анализ (probabilistic latent semantic analysis, PLSA) был предложен Томасом Хофманном в [17]. Вероятностная модель появления пары «документ–термин» (d, w) записывается тремя эквивалентными способами:

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

где $p(t)$ — распределение тем во всей коллекции. Первое представление называется симметричным, второе и третье — несимметричными. Они приводят к немного разным итерационным процессам обучения тематической модели. Сейчас возьмём за основу второе представление, совпадающее с (1.2).

1.2.1 EM-алгоритм

Для решения задачи (1.6) в PLSA применяется итерационный процесс, в котором каждая итерация состоит из двух шагов — E (expectation) и M (maximization) [12]. Перед первой итерацией выбирается начальное приближение параметров φ_{wt}, θ_{td}.

На E-шаге по текущим значениям параметров φ_{wt}, θ_{td} с помощью формулы Байеса вычисляются условные вероятности $p(t\,|\,d,w)$ всех тем $t \in T$ для каждого термина $w \in d$ в каждом документе d:

$$H_{dwt} = p(t\,|\,d,w) = \frac{p(w\,|\,t)p(t\,|\,d)}{p(w\,|\,d)} = \frac{\varphi_{wt}\theta_{td}}{\sum\limits_{s\in T}\varphi_{ws}\theta_{sd}}. \tag{1.8}$$

На М-шаге, наоборот, по условным вероятностям тем H_{dwt} вычисляется новое приближение параметров φ_{wt}, θ_{td}. Это легко сделать, если заметить, что величина $\hat{n}_{dwt} = n_{dw} H_{dwt}$ оценивает (не обязательно целое) число n_{dwt} вхождений термина w в документ d, связанных с темой t. Просуммировав $\hat{n}_{dwt}$ по документам d и по терминам w, получим оценки $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$, и через них, согласно (1.4), — частотные оценки условных вероятностей φ_{wt}, θ_{td}:

$$\varphi_{wt} = \frac{\hat{n}_{wt}}{\hat{n}_t}, \qquad \hat{n}_t = \sum_{w \in W} \hat{n}_{wt}, \qquad \hat{n}_{wt} = \sum_{d \in D} n_{dw} H_{dwt}. \tag{1.9}$$

$$\theta_{td} = \frac{\hat{n}_{dt}}{\hat{n}_d}, \qquad \hat{n}_d = \sum_{t \in T} \hat{n}_{dt}, \qquad \hat{n}_{dt} = \sum_{w \in d} n_{dw} H_{dwt}. \tag{1.10}$$

Эти простые, но не вполне строгие рассуждения поясняют суть ЕМ-алгоритма. Покажем теперь, что оценки (1.9)–(1.10) действительно являются решением задачи максимизации правдоподобия (1.6) при фиксированных H_{dwt}. Запишем лагранжиан задачи (1.6) при ограничениях нормировки, проигнорировав ограничения неотрицательности (позже убедимся, что решение действительно неотрицательно):

$$\mathscr{L}(\Theta, \Phi) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln \underbrace{\sum_{t \in T} \varphi_{wt} \theta_{td}}_{p(w \,|\, d)} - \sum_{t \in T} \lambda_t \Bigg(\sum_{w \in W} \varphi_{wt} - 1 \Bigg) - \sum_{d \in D} \mu_d \Bigg(\sum_{t \in T} \theta_{td} - 1 \Bigg).$$

Продифференцировав лагранжиан по φ_{wt} и приравняв нулю производную, получим

$$\lambda_t = \sum_{d \in D} n_{dw} \frac{\theta_{td}}{p(w \,|\, d)}. \tag{1.11}$$

Домножим обе части этого равенства на φ_{wt}, просуммируем по всем терминам $w \in W$, применим условие нормировки вероятностей φ_{wt} в левой части и выделим переменную H_{dwt} в правой части. Получим

$$\lambda_t = \sum_{d \in D} \sum_{w \in W} n_{dw} H_{dwt}.$$

Снова домножим обе части (1.11) на φ_{wt}, выделим переменную H_{dwt} в правой части и выразим φ_{wt} из левой части, подставив уже известное выражение для λ_t. Получим

$$\varphi_{wt} = \frac{\sum\limits_{d \in D} n_{dw} H_{dwt}}{\sum\limits_{w' \in W} \sum\limits_{d \in D} n_{dw'} H_{dw't}}.$$

Обозначив числитель через $\hat{n}_{wt}$, получим (1.9). Проделав аналогичные действия с производной лагранжиана по θ_{td}, получим (1.10).

Эффективность ЕМ-алгоритма по времени и по памяти. Число операций растёт линейно по длине коллекции n, числу тем T и числу итераций.

Перебор всех терминов w во всех документах d можно организовать очень эффективно, если хранить каждый документ d в виде последовательности пар (w, n_{dw}).

Алгоритм 1.2. PLSA-EM: рациональный EM-алгоритм для модели PLSA.

Вход: коллекция документов D, число тем $|T|$, начальные приближения Θ и Φ;
Выход: распределения Θ и Φ;

повторять
  обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ для всех $d \in D,\ w \in W,\ t \in T$;
  для всех $d \in D,\ w \in d$
    $Z := \sum_{t \in T} \varphi_{wt}\theta_{td}$;
    для всех $t \in T$ таких, что $\varphi_{wt}\theta_{td} > 0$
      увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на $\delta = n_{dw}\varphi_{wt}\theta_{td}/Z$;
  $\varphi_{wt} := \hat{n}_{wt}/\hat{n}_t$ для всех $w \in W,\ t \in T$;
  $\theta_{td} := \hat{n}_{dt}/n_d$ для всех $d \in D,\ t \in T$;
пока Θ и Φ не стабилизируются.

Рациональный EM-алгоритм. Вычисление переменных $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на M-шаге требует однократного прохода всей коллекции в цикле по всем документам $d \in D$ и всем терминам $w \in d$. Внутри этого цикла переменные H_{dwt} можно вычислять непосредственно в тот момент, когда они понадобятся. От этого результат алгоритма не изменяется, E-шаг встраивается внутрь M-шага без дополнительных вычислительных затрат, отпадает необходимость хранения трёхмерной матрицы H_{dwt}. Заметим также, что переменную $\hat{n}_d$ можно не вычислять, поскольку $\hat{n}_d = n_d$. Этот вариант реализации EM-алгоритма будем называть *рациональным*; он показан в Алгоритме 1.2.

Проблема разреженности. Если начальные приближения θ_{td} и φ_{wt} положительны, то и после каждой итерации они будут оставаться положительными, несмотря на то, что ограничение неотрицательности было проигнорировано в ходе решения. И, наоборот, если $\theta_{td} = 0$ (тема t не представлена в документе d) или если $\varphi_{wt} = 0$ (термин w не относится к теме t), то нулевое значение будет сохраняться на протяжении всех итераций. Таким образом, в PLSA структура разреженности распределений не оптимизируется, а задаётся через начальное приближение. В то же время, использование разреженных матриц для хранения переменных $\hat{n}_{wt}$, $\hat{n}_{dt}$, θ_{td}, φ_{wt} могло бы дать существенную экономию памяти.

Эксперимент: принудительное разреживание портит модель, если его делать с первых итераций. Надо дождаться сходимости, когда правильно определится подмножество малых вероятностей. ToDo[8]

1.2.2 Обобщённый EM-алгоритм

В EM-алгоритме нет необходимости сверхточно решать задачу максимизации правдоподобия на M-шаге. Достаточно ещё немного приблизиться к точке максимума правдоподобия и снова выполнить E-шаг. Это связано с тем, что сам функционал правдоподобия известен не точно — он зависит от приближённых значений H_{dwt}, полученных на E-шаге. EM-алгоритм с сокращённым M-шагом называется *обобщённым EM-алгоритмом* (generalized EM-algorithm, GEM). Для него справедливы те же доказательства сходимости, что и для основного варианта EM-алгоритма [12].

Алгоритм 1.3. PLSA-GEM: обобщённый EM-алгоритм для модели PLSA.

Вход: коллекция документов D, число тем $|T|$, начальные приближения Θ и Φ;
Выход: распределения Θ и Φ;

1: обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$, $\hat{n}_d$, n_{dwt} для всех $d \in D$, $w \in W$, $t \in T$;
2: **повторять**
3: **для всех** $d \in D$, $w \in d$
4: $Z := \sum_{t \in T} \varphi_{wt}\theta_{td}$;
5: **для всех** $t \in T$ таких, что $n_{dwt} > 0$ или $\varphi_{wt}\theta_{td} > 0$
6: $\delta := n_{dw}\varphi_{wt}\theta_{td}/Z$;
7: увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$, $\hat{n}_d$ на $(\delta - n_{dwt})$;
8: $n_{dwt} := \delta$;
9: **если** пора обновить параметры Φ, Θ **то**
10: $\varphi_{wt} := \hat{n}_{wt}/\hat{n}_t$ для всех $w \in W$, $t \in T$ таких, что $\hat{n}_{wt}$ изменился;
11: $\theta_{td} := \hat{n}_{dt}/\hat{n}_d$ для всех $d \in D$, $t \in T$ таких, что $\hat{n}_{dt}$ изменился;
12: **пока** Θ и Φ не стабилизируются.

В случае PLSA сокращение M-шага сводится к более частому обновлению параметров θ_{td} и φ_{wt} по значениям счётчиков $\hat{n}_{wt}$ и $\hat{n}_{dt}$. В Алгоритме 1.2 это происходит после каждого просмотра всей коллекции. Обновления можно делать после обработки каждого документа или после заданного числа обработанных пар (d, w) или даже после каждой пары. На больших коллекциях частые обновления повышают скорость сходимости. В Алгоритме 1.3 выбор условия обновления на шаге 9 оставлен на усмотрение разработчика.

На первой итерации (т. е. при первом проходе коллекции) частые обновления не делаются, чтобы в счётчиках накопилась информация по всей коллекции. В противном случае оценки параметров θ_{td} и φ_{wt} по начальному фрагменту выборки могут оказаться хуже начального приближения. Начиная со второй итерации, для каждой пары (d, w) из счётчиков $\hat{n}_{wt}$ и $\hat{n}_{dt}$ вычитается n_{dwt} — то самое значение δ, которое было к ним прибавлено при обработке пары (d, w) на предыдущей итерации. Таким образом, счётчики $\hat{n}_{wt}$ и $\hat{n}_{dt}$ всегда содержат результат последнего однократного прохода всей коллекции.

Необходимость хранения трёхмерной матрицы n_{dwt} делает Алгоритм 1.3 непригодным для больших коллекций. Этот недостаток устраняется путём реорганизации итераций, либо применением сэмплирования. Рассмотрим оба способа.

Эксперимент: частота обновления влияет на эффективность но не влияет на качество. Лучше всего обновлять после каждого слова. При этом можно вообще отказаться от хранения матриц тета и фи. ToDo[9]

1.2.3 Быстрый EM-алгоритм

На больших коллекциях Алгоритмы 1.2 и 1.3 могут сходиться очень медленно. Причина в том, что на каждой итерации производится однократный проход всей коллекции; за это время оценки распределений терминов в темах $\varphi_{wt} = n_{wt}/n_t$ уточняются огромное число раз и успевают сойтись; тогда как распределения тем в документах θ_d проходят лишь одну итерацию. На начальных итерациях, пока распре-

Алгоритм 1.4. PLSA-FEM: быстрый EM-алгоритм для модели PLSA.

Вход: коллекция документов D, число тем $|T|$;
Выход: распределения Θ и Φ;

1: инициализировать φ_{wt} для всех $w \in W,\ t \in T$;
2: **повторять**
3: $\quad \hat{n}_{wt} := 0;\ \hat{n}_t := 0$ для всех $w \in W,\ t \in T$;
4: $\quad$ **для всех** $d \in D$
5: $\quad\quad$ инициализировать θ_{td} для всех $d \in D,\ t \in T$;
6: $\quad\quad$ **повторять**
7: $\quad\quad\quad H_{wt} := \dfrac{\varphi_{wt}\theta_{td}}{\sum\limits_{s \in T} \varphi_{ws}\theta_{sd}}$ для всех $w \in W,\ t \in T$;
8: $\quad\quad\quad \theta_{td} := \frac{1}{n_d} \sum\limits_{w \in d} n_{dw} H_{wt}$ для всех $t \in T$;
9: $\quad\quad$ **пока** θ_d не сойдётся.
10: $\quad\quad$ увеличить $\hat{n}_{wt}$, $\hat{n}_t$ на $n_{dw}H_{wt}$ для всех $t \in T$;
11: $\quad\quad$ $\varphi_{wt} := \hat{n}_{wt}/\hat{n}_t$ для всех $w \in W,\ t \in T$ таких, что $\hat{n}_{wt}$ изменился;
12: **пока** Θ и Φ не стабилизируются.

деления θ_d далеки от оптимальных, распределения φ_t сходятся к некоторым приближениям, также далёким от оптимальных. В результате масса усилий тратится впустую.

Решение проблемы заключается в реорганизации итерационного процесса. Проход каждого документа $d \in D$ производится не один раз, а много раз подряд. Для каждого термина $w \in d$ выполняется E-шаг и обновляется распределение θ_d. Обновление распределений φ_t может производиться после каждого прохода коллекции, как в Алгоритме 1.2, либо чаще, например, через фиксированное число документов. В результате распределения φ_t и θ_d сходятся более согласованно.

Реорганизация позволяет отказаться хранения трёхмерных массивов. В псевдокоде Алгоритма 1.5 вместо переменных $H_{dwt} = p(t \mid d, w)$ используются H_{wt}, чтобы подчеркнуть, что по окончании обработки документа d эти данные уже не нужны, и двумерный массив H_{wt} можно использовать для обработки следующего документа.

Описанный итерационный процесс используется в алгоритме Online-LDA [**?**], который считается одним из самых быстрых в тематическом моделировании.

1.2.4 Стохастический EM-алгоритм

В Алгоритме 1.3 для каждой пары (d, w) происходит распределение n_{dw} вхождений термина w в документ d между всеми $|T|$ темами пропорционально вероятностям $p(t \mid d, w)$. При этом приходится хранить массив значений n_{dwt} для всех тем $t \in T$. Расход памяти объёма $O(n|T|)$ может оказаться неприемлемым даже при небольшом числе тем. В то же время, согласно гипотезе разреженности, употребление термина w в документе d связано, скорее всего, с небольшим числом тем.

Можно было бы оставлять только несколько наибольших значений n_{dwt} на каждом шаге. Однако эксперименты показывают, что эта эвристика приводит к накоплению систематической ошибки и смещению модели.

Сделать подтверждающий эксперимент по max-разреживанию для PLSA. ToDo[10]

Проблема разреживания условного распределения $p(t\,|\,d,w)$ адекватно решается с помощью стохастического EM-алгоритма (stochastic EM-algorithm, SEM) [8]. Распределение скрытой переменной t, вычисленное на E-шаге, не используется непосредственно на M-шаге. Вместо этого из него сэмплируется искусственная выборка, по этой выборке вычисляется эмпирическое распределение, и оно уже используется в формулах M-шага. Это позволяет упростить задачу M-шага, сохранив свойства несмещённости оценок и сходимости EM-алгоритма. Размер сэмплируемой выборки является параметром метода.

В случае PLSA реализация SEM сводится к следующему: для каждой пары (d,w) сэмплируются s случайных тем t_{dwi}, $i=1,\ldots,s$ из распределения $p(t\,|\,d,w)$, возможно, повторяющихся. В формулах M-шага вместо распределения $p(t\,|\,d,w)$ используется его несмещённая эмпирическая оценка:

$$\hat{p}(t\,|\,d,w) = \frac{1}{s}\sum_{i=1}^{s}\big[t_{dwi}=t\big]. \tag{1.12}$$

Модификация Алгоритма 1.3, трансформирующая его в стохастический обобщённый EM-алгоритм (PLSA-SGEM), состоит из трёх изменений:

1) перед шагом 5 сэмплируется s тем t_{dwi}, $i=1,\ldots,s$ из $p(t\,|\,d,w)$;

2) на шаге 5 цикл по всем $t\in T$ заменяется циклом по $t=t_{dwi}$, $i=1,\ldots,s$;

3) на шаге 6 вычисляется $\delta := n_{dw}/s$.

При $s=n_{dw}$ стохастический EM-алгоритм соответствует *сэмплированию Гиббса* [33], которое считается одним из основных методов обучения вероятностных тематических моделей. Однако эксперименты показывают, что параметр s можно брать намного меньше, от 1 до 5. Эта эвристика, названная *экономным сэмплированием* [1], приводит к разреживанию распределений $p(t\,|\,d,w)$ и существенной экономии вычислительного ресурса и памяти без потери качества тематической модели.

Эксперимент: зависимость контрольной перплексии от параметра разреживания s. ToDo[11]

1.2.5 Формирование начальных приближений

Начальные приближения φ_t и θ_d можно задавать нормированными случайными векторами из равномерного распределения.

Другая распространённая рекомендация — пройти по всей коллекции, выбрать для каждой пары (d,w) случайную тему t, и вычислить частотные оценки (1.4) вероятностей φ_{wt} и θ_{td} для всех $d\in D$, $w\in W$, $t\in T$.

Учёт дополнительной информации о принадлежности некоторых документов или терминов темам позволяет с самого начала фиксировать интерпретации тем.

Инициализация на основе априорной информации. Если известно, что подмножество терминов $W_t\subset W$ относится к теме t, то в качестве начального φ_{wt} можно взять равномерное распределение на этом подмножестве:

$$\varphi_{wt} = \frac{1}{|W_t|}\big[w\in W_t\big].$$

Если известно, что подмножество документов D_t относится к теме t, то можно взять эмпирическое распределение слов в объединённом документе:

$$\varphi_{wt} = \frac{\sum_{d \in D_t} n_{dw}}{\sum_{d \in D_t} n_d}.$$

Если нет никакой априорной информации о связи документов с темами, то последнюю формулу можно применить к случайным подмножествам документов D_t. В [16] предлагается брать один случайный документ.

Если известно, что документ d относится к подмножеству тем $T_d \subset T$, то в качестве начального θ_{td} можно взять равномерное распределение на этом подмножестве:

$$\theta_{td} = \frac{1}{|T_d|}\big[t \in T_d\big].$$

Сглаживание. Если полученное начальное приближение φ^0_{wt} или θ^0_{td} содержит нулевые вероятности, то его можно сгладить, смешав с каким-нибудь неразреженным распределением. Например, φ^0_{wt} смешивается с эмпирическим распределением слов во всей коллекции и со случайным распределением $\rho(w)$, при некоторых значениях параметров смеси τ_1 и τ_2:

$$\varphi_{wt} = (1 - \tau_1 - \tau_2)\varphi^0_{wt} + \tau_1 n_w/n + \tau_2 \rho(w).$$

Инициализация тем документов. Если для всех тем известны оценки распределений $\varphi_{wt} = p(w \,|\, t)$, то инициализация θ_{td} упрощается. Одна итерация EM-алгоритма для документа d при равномерном начальном приближении θ_{td} приводит к формуле усреднения распределений $p(t \,|\, w)$ по словам документа d:

$$\theta_{td} = \frac{1}{n_d}\sum_{w \in d} n_{dw} H_{dwt} = \sum_{w \in d} \frac{n_{dw}}{n_d} \frac{\varphi_{wt}}{\sum_s \varphi_{ws}}. \tag{1.13}$$

Если полученное начальное приближение θ_{td} содержит нулевые вероятности, то его можно сгладить, смешав с каким-нибудь неразреженным распределением.

Начальное приближение путём равномерного сэмплирования тем [Потапенко]. ToDo[12]

Быстрое формирование начального приближения $p(w \,|\, t)$ путём разреживания множества документов и рандомизации порядка документов. ToDo[13]

Эксперименты: сравнить несколько способов задания начального приближения. ToDo[14]

1.2.6 Оптимизация числа тем

Известно много подходов к определению числа тем в коллекции. Наиболее традиционный — построить тематическую модель при различных значениях $|T|$ и выбрать оптимальное число тем по одному из критериев качества, описанных ниже в разделе §1.6. В эффективной реализации этого метода темы добавляются постепенно. При появлении новых тем для них инициализируются распределения φ_t, добавляются и инициализируются новые значения в распределениях θ_d, затем распределения θ_d перенормируются. После этого продолжаются итерации EM-алгоритма.

Поскольку параметры были изменены не сильно, сходимость достигается относительно быстро, за небольшое число проходов всей коллекции. По достижении сходимости вычисляется выбранный критерий качества. Процесс наращивания тем прекращается, когда критерий перестаёт заметно улучшаться.

Про непараметрический байесовский вывод: HDP ToDo[15]

1.2.7 Присоединение нового документа к коллекции

Присоединение (folding-in) нового документа d к коллекции D, для которой уже построена тематическая модель Φ, Θ, можно осуществлять различными способами.

Алгоритм PLSA-EM. В [17] предлагается фиксировать матрицу Φ, найденную по всем предыдущим документам, и определять только вектор θ_d для нового документа. Эта эвристика основана на предположении, что коллекция достаточно велика, и один документ d не может существенно повлиять на оценки распределений φ_t. Оно может не выполняться, если документ d содержит значительное число новых терминов или относится к темам, слабо представленным в коллекции.

Для реализации этой эвристики в Алгоритме 1.2 убирается цикл перебора всех документов и вычисление значений φ_{wt} на шаге 7. Упрощённый таким способом итерационный процесс максимизирует правдоподобие (1.6) при фиксированных матрице Φ и всех столбцах матрицы Θ, кроме θ_d. Таким образом, происходит подмена исходной постановки задачи. Это один из основных недостатков PLSA, мотивировавших переход к более современной тематической модели LDA [6].

Обобщённый алгоритм PLSA-GEM уже не обладает указанным недостатком. Благодаря частым обновлениям параметров θ_{td} и φ_{wt} он естественным образом адаптируется для присоединения новых документов. Достаточно применить Алгоритм 1.3 к коллекции из одного документа $\{d\}$ без инициализации счётчиков на шаге 1, сохранив текущие значения $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$, n_{dwt}. Затем продолжаются итерации на пополненной коллекции $D \cup \{d\}$, также без шага 1.

Возможно пополнять коллекцию D непосредственно в ходе итераций Алгоритма 1.3. Допустим, коллекция уже была просмотрена хотя бы один раз, хотя бы частично, и имеются текущие приближения параметров φ_{wt}. Тогда для присоединения нового документа d достаточно инициализировать θ_d по формуле (1.13) и выполнить несколько итераций только для этого документа до сходимости распределения θ_d. После этого цикл перебора документов возобновляется в обычном порядке.

Если документ d содержит новые термины, которые ранее не встречались в коллекции, то для них должны быть инициализированы ненулевые значения φ_{wt}. Это повлечёт необходимость перенормировки распределений φ_t для всех тем t, на которые повлиял документ d.

Итерационный процесс с постепенным присоединением новых документов можно рассматривать как *бесконечно долгий алгоритм* (any-time algorithm) [41]. Концепция any-time предполагает, что алгоритм может работать над улучшением решения сколь угодно долго, причём в любой момент у него можно запросить текущее приближённое решение, но чем дольше работает алгоритм, тем лучшее решение он выдаёт.

ToDo[16]

Аккуратный folding-in с контролируемой цепной реакцией обновлений параметров (Тимур Исмагилов).

1.2.8 Учёт априорной информации (частичное обучение)

В некоторых задачах классификации и каталогизации текстов бывает известно, что какие-то документы или термины относятся или, наоборот, не относятся к некоторым темам. В таких случаях говорят о задачах с *частичным обучением* (semi-supervised learning). В EM-алгоритме нетрудно учесть априорную информацию данного типа. Рассмотрим Алгоритм 1.3. Модификации коснутся только правила пересчёта параметров θ_{td} и φ_{wt} на шагах 10–11.

Проще всего учесть априорную информацию о том, что документ d или термин w не связан с темой t. Для этого достаточно обнулить параметр θ_{td} или φ_{wt}, обнулив соответствующие счётчики перед шагом 10:

если документ d не связан с темой t **то**
$\hat{n}_d := \hat{n}_d - \hat{n}_{dt}$;
$\hat{n}_{dt} := 0$;

если термин w не связан с темой t **то**
$\hat{n}_t := \hat{n}_t - \hat{n}_{wt}$;
$\hat{n}_{wt} := 0$.

Информация о том, что документ d или термин w связан с темой t, как правило, не исключает наличия у них связей с другими темами. Поэтому обнулять вероятности, соответствующие всем темам, кроме одной или нескольких заданных, было бы неправильно. Будем полагать, что априорную информацию задаёт эксперт, который имеет возможность просматривать список тем по любому документу d. Список ранжирован по убыванию частот $\hat{n}_{dt}$, и эксперт может перенести любую тему на то место в списке, которое он считает наиболее релевантным. Чтобы тема t оказалась на k-м месте в списке тем документа d, достаточно сделать значение $\hat{n}_{dt}$ немного бóльшим k-го значения $\hat{n}_{dt}^{(k)}$ и скорректировать счётчик $\hat{n}_d$:

если тема t для документа d должна быть на k-м месте **то**
$\hat{n}'_{dt} := \frac{1}{2}\big(\hat{n}_{dt}^{(k)} + \hat{n}_{dt}^{(k-1)}\big)[k > 1] + \frac{1}{2}\big(3\hat{n}_{dt}^{(k)} - \hat{n}_{dt}^{(k+1)}\big)[k = 1]$;
$\hat{n}_d := \hat{n}_d - \hat{n}_{dt} + \hat{n}'_{dt}$;
$\hat{n}_{dt} := \hat{n}'_{dt}$;

Пусть имеется также список терминов по любой теме t, ранжированный по убыванию вероятностей φ_{wt}, и эксперт может проделать с ним аналогичную работу. Чтобы термин w оказался на k-м месте в списке терминов темы t, достаточно сделать значение $\hat{n}_{wt}$ немного бóльшим k-го значения $\hat{n}_{wt}^{(k)}$ и скорректировать счётчик $\hat{n}_t$:

если термин w для темы t должен быть на k-м месте **то**
$\hat{n}'_{wt} := \frac{1}{2}\big(\hat{n}_{wt}^{(k)} + \hat{n}_{wt}^{(k-1)}\big)[k > 1] + \frac{1}{2}\big(3\hat{n}_{wt}^{(k)} - \hat{n}_{wt}^{(k+1)}\big)[k = 1]$;
$\hat{n}_t := \hat{n}_t - \hat{n}_{wt} + \hat{n}'_{wt}$;
$\hat{n}_{wt} := \hat{n}'_{wt}$;

Существуют и другие способы задания априорной информации, для которых также можно адаптировать правила пересчёта распределений по счётчикам.

ToDo[17]

Подмножество W_t ключевых терминов темы t можно включить в коллекцию как «виртуальный документ», для которого $\theta_{\tau d} = [\tau = t]$ устанавливается во время инициализации.

Эксперименты с виртуальными документами. ToDo[18]

Стандартные методы SSL и сравнение с предложенным методом. ToDo[19]

§1.3 Латентное размещение Дирихле

Основным недостатком PLSA считается высокая размерность пространства параметров, вызывающая переобучение [6]. В задачах машинного обучения для сокращения размерности обычно используется *регуляризация* — наложение дополнительных ограничений на параметры. *Байесовская регуляризация* основана на введении априорного распределения вероятности в пространстве параметров.

Тематическая модель *латентного размещения Дирихле* (latent Dirichlet allocation, LDA) [6] основана на разложении (1.2) при дополнительном предположении, что векторы документов $\theta_d = (\theta_{td}) \in \mathbb{R}^{|T|}$ и векторы тем $\varphi_t = (\varphi_{wt}) \in \mathbb{R}^{|W|}$ порождаются распределениями Дирихле с параметрами $\alpha \in \mathbb{R}^{|T|}$ и $\beta \in \mathbb{R}^{|W|}$ соответственно:

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

$$\operatorname{Dir}(\varphi_t;\beta) = \frac{\Gamma(\beta_0)}{\prod_w \Gamma(\beta_w)} \prod_w \varphi_{wt}^{\beta_w - 1}, \quad \beta_w > 0, \quad \beta_0 = \sum_w \beta_w, \quad \varphi_{wt} > 0, \quad \sum_w \varphi_{wt} = 1.$$

где $\Gamma(z)$ — гамма-функция. Считается, что распределение Дирихле хорошо подходит в качестве байесовского регуляризатора в задачах тематического моделирования.

Во-первых, это достаточно широкое параметрическое семейство распределений на единичном симплексе, то есть на множестве дискретных распределений. Если $\alpha_t = 1$ для всех t, то распределение Дирихле переходит в равномерное. Математическое ожидание и дисперсия t-й координаты вектора θ_d равны, соответственно,

$$\mathsf{E}\theta_{td} = \int \theta_{td} \operatorname{Dir}(\theta_d;\alpha)\, d\theta_d = \frac{\alpha_t}{\alpha_0}, \qquad \mathsf{D}\theta_{td} = \frac{\alpha_t(\alpha_0 - \alpha_t)}{\alpha_0^2(\alpha_0 + 1)}. \tag{1.14}$$

Параметр α определяет степень разреженности векторов θ_d, порождаемых распределением $\operatorname{Dir}(\theta;\alpha)$. Чем больше α_0, тем сильнее векторы θ_d, концентрируется вокруг вектора математического ожидания $\mathsf{E}\theta_d$. Чем меньше α_t, тем сильнее значения θ_{td} концентрируются вокруг нуля. Чем меньше α_0, тем более разрежен вектор θ_d. Поэтому α_t называют *параметрами контраста*. Они позволяют управлять разреженностью, но не в полной мере: условные вероятности θ_{td} могут принимать сколь угодно близкие к нулю значения, но не могут обращаться в нуль.

Во-вторых, двухуровневая вероятностная модель порождения данных хорошо подходит для описания кластерных структур. На первом уровне распределение Дирихле $\operatorname{Dir}(\varphi;\beta)$ порождает центры кластеров тем — случайные векторы φ_t в пространстве терминов $\mathbb{R}^{|W|}$. На втором уровне эти векторы, будучи дискретными распределениями, порождают выборки терминов, эмпирические распределения которых группируются вокруг своих центров. Каждый документ d формируется как смесь выборок из распределений φ_t, взятых с весами θ_{td}. Векторы весов тем θ_d порождаются распределением Дирихле $\operatorname{Dir}(\theta;\alpha)$ в пространстве тем $\mathbb{R}^{|T|}$, образуя кластеры

документов. Степенью выраженности кластеров управляют параметры α и β. Чем меньше α_0 и β_0, тем более разрежены распределения Дирихле, и тем чётче выражены кластерные структуры.

В-третьих, распределение Дирихле является сопряжённым к мультиномиальному, что упрощает вывод апостериорных оценок вероятностей θ_{td} и φ_{wt}.

Основным недостатком распределения Дирихле является отсутствие убедительных лингвистических обоснований в пользу данной модели порождения центров кластеров.

1.3.1 Байесовский вывод

Рассмотрим процесс порождения документа d как выборки n_d пар тема–термин $X_d = \{(t_1, w_1), \ldots, (t_{n_d}, w_{n_d})\}$. В каждой паре (t_i, w_i) тема t_i выбирается из дискретного распределения $p(t \mid d) = \theta_{td}$. Следовательно, вероятность встретить каждую из тем t ровно n_{td} раз подчиняется мультиномиальному распределению:

$$p(X_d|\theta_d) = \frac{n_d!}{\prod_t n_{td}!} \prod_t \theta_{td}^{n_{td}}.$$

Распределение Дирихле является *сопряжённым* к мультиномиальному. Это означает, что при априорном распределении Дирихле $\theta_d \sim \mathrm{Dir}(\theta; \alpha)$ апостериорное распределение вектора θ_d принадлежит тому же семейству распределений, но с другим значением параметра: $\theta_d|X_d \sim \mathrm{Dir}(\theta; \alpha')$. Действительно, по формуле Байеса

$$p(\theta_d|X_d, \alpha) = \frac{p(X_d|\theta_d)\,\mathrm{Dir}(\theta_d; \alpha)}{p(X_d)} = C \prod_t \theta_{td}^{n_{td}} \theta_{td}^{\alpha_t - 1} = \mathrm{Dir}(\theta_d; \alpha'), \quad \alpha'_t = \alpha_t + n_{td},$$

где C — нормировочная константа, не зависящая от θ_d.

Оценим случайную величину θ_{td} её математическим ожиданием (1.14) по апостериорному распределению:

$$p(t|d, X_d, \alpha) = \int p(t|d) p(\theta_d|X_d, \alpha)\, d\theta_d = \int \theta_{td}\, \mathrm{Dir}(\theta_d, \alpha')\, d\theta_d = \frac{n_{td} + \alpha_t}{n_d + \alpha_0}. \tag{1.15}$$

Заменив величину n_{td} её оценкой $\hat{n}_{td}$, получим сглаженную байесовскую оценку параметра θ_{td} для EM-алгоритма, альтернативную оценке максимума правдоподобия (1.10):

$$\theta_{td} = \frac{\hat{n}_{dt} + \alpha_t}{\hat{n}_d + \alpha_0}. \tag{1.16}$$

Аналогично выводится сглаженная байесовская оценка и для φ_{wt}, альтернативная (1.9):

$$\varphi_{wt} = \frac{\hat{n}_{wt} + \beta_w}{\hat{n}_t + \beta_0}. \tag{1.17}$$

Замена в обобщённом EM-алгоритме частотных оценок условных вероятностей (1.9) и (1.10) сглаженными оценками (1.17) и (1.16) трансформирует PLSA в LDA. Её строгое обоснование приводится в [25, 33] для метода сэмплирования Гиббса и в [27]

Алгоритм 1.5. LDA-GS: сэмплирование Гиббса для тематической модели LDA.

Вход: коллекция D, число тем $|T|$, начальные Θ, Φ, векторы гиперпараметров α, β;
Выход: распределения Θ и Φ;

обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ для всех $d \in D,\ w \in W,\ t \in T$;
повторять
  для всех $d \in D,\ w \in d,\ i = 1, \ldots, n_{dw}$
    если не первая итерация **то**
      $t := t_{dwi}$; уменьшить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на 1;
    сэмплировать тему t_{dwi} из $p(t \mid d, w) \propto (\hat{n}_{dt} + \alpha_t)(\hat{n}_{wt} + \beta_w)/(\hat{n}_t + \beta_0)$;
    $t := t_{dwi}$; увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на 1;
пока Θ и Φ не стабилизируются.
$\varphi_{wt} = (\hat{n}_{wt} + \beta_w)/(\hat{n}_t + \beta_0)$ для всех $t \in T,\ w \in W$;
$\theta_{td} := (\hat{n}_{dt} + \alpha_t)/(n_d + \alpha_0)$ для всех $d \in D,\ t \in T$;

для метода вариационной байесовской аппроксимации. В [15] показано, что PLSA является частным случаем LDA при $\alpha = 0$ и $\beta = 0$. В [5] показано, что основные известные алгоритмы обучения LDA являются вариантами обобщённого EM-алгоритма и отличаются, главным образом, формулой сглаживания частотных оценок вероятностей. Другие различия слабо влияют на качество тематической модели и являются скорее техническими. Оптимизация гиперпараметров α и β, предложенная в [30, 31], ещё сильнее нивелирует различия между моделями. Согласно экспериментам на 7 текстовых коллекциях [5], более эффективным по качеству и по времени является алгоритм *свёрнутой вариационной байесовской аппроксимации* CVB0 (collapsed variational Bayes). В нашей нотации ему наиболее близок LDA-GEM.

Сэмплирование Гиббса является является частным случаем LDA-SGEM.

PLSA является частным случаем LDA при $\alpha = 0$ и $\beta = 0$, см. также [15].

1.3.2 Сэмплирование Гиббса

В задачах статистического оценивания часто возникает ситуация, когда вычисление или хранение некоторой функции распределении слишком ресурсоёмко, в то же время, генерация случайной выборки из этого распределения не вызывает затруднений. Сэмплированием Гиббса (Gibbs sampling, GS) называется общий метод решения таких задач, основанный на замене исходного распределения эмпирическим, вычисленным по выборке, сэмплированной из данного распределения.

Применение GS к тематической модели LDA предложено в [25]. Строгий вывод формул LDA-GS приводится в отчёте [33]. LDA-GS (Алгоритм 1.6) имеет несколько отличий от PLSA-SGEM — стохастического варианта Алгоритма 1.3, но только одно из них оказывается существенным с точки зрения качества модели.

1. В LDA-GS жёстко фиксируется число сэмплирований тем $s = n_{dw}$ для каждой пары (d, w). Однако *гипотеза разреженности* предполагает, что термин w в документе d связан с небольшим числом тем. В наших экспериментах $s = 5$ тем оказалось достаточно. В некоторых задачах достаточно и одной темы, в других одной темы мало, см. рис. 2. Эвристика *экономного сэмплирования* повышает эффективность алгоритма как по скорости, так и по памяти, не ухудшая качество модели.

Вставить график

2. В LDA-GS параметры φ_{wt} и θ_{td} обновляются предельно часто — после обработки каждого вхождения термина w в документ d. Эксперименты показывают, что достаточно делать обновления после каждой пары (d, w), это не влияет на качество модели.

Вставить график

ToDo[21]

3. В LDA-GS перед сэмплированием счётчики уменьшаются на единицу (шаг 5). Тем самым в оценке распределений не учитывается i-е вхождение термина w в документ d, для которого сэмплируется тема t_{dwi}. Эта особенность алгоритма следует из теории [33]. Однако эксперименты показывают, что она не влияет на качество модели. Можно одновременно уменьшать счётчики для старой темы и увеличивать для новой, как в Алгоритме 1.3.

4. Единственным существенным различием, влияющим на качество модели, является применение в LDA байесовской регуляризации, приводящей к сглаживанию частотных оценок условных вероятностей.

Таким образом, LDA-GS существенно отличается от PLSA-GEM только тремя эвристиками: частотой обновления параметров, сэмплированием и сглаживанием. Эти эвристики не связаны друг с другом и могут применяться в любых сочетаниях.

1.3.3 Оптимизация гиперпараметров

В первых работах по LDA [6] и сэмплированию Гиббса [25], а также в последовавших за ними исследованиях использовались симметричные распределения Дирихле с гиперпараметрами $\alpha = (a, \ldots, a)$ и $\beta = (b, \ldots, b)$. Скалярные гиперпараметры a и b либо фиксировались, либо настраивались путём перебора по сетке значений. В более поздних работах были предложены эффективные численные методы оптимизации гиперпараметров, их обзор и сравнение приводится в диссертации [30].

Эксперименты показали, что оптимизация гиперпараметров существенно улучшает качество тематической модели [31]. Оказалось, что априорное распределение $\mathrm{Dir}(\theta; \alpha)$ лучше брать несимметричным и оптимизировать вектор гиперпараметров $\alpha = (\alpha_1, \ldots, \alpha_{|T|})$, а распределение $\mathrm{Dir}(\varphi; \beta)$ лучше брать симметричным и оптимизировать скалярный гиперпараметр b.

1.3.4 Действительно ли сглаживание уменьшает переобучение

Эксперименты [6] показали, что LDA обеспечивает существенно меньшие значения контрольной перплексии, чем PLSA. По аналогии с задачами классификации и регрессии отсюда был сделан стандартный вывод, что модель PLSA имеет слишком много параметров θ_{td}, φ_{wt}, и при отсутствии ограничений на них возникает переобучение. Байесовская регуляризация должна сокращать эффективную размерность и уменьшать переобучение.

Однако возможна и другая интерпретация этих экспериментов. Сложность модели здесь не при чём — PLSA и LDA оценивают одни и те же матрицы параметров Φ и Θ. Оптимальные значения гиперпараметров α и β в LDA обычно близки к нулю. Поэтому оценки параметров φ_{wt} и θ_{td} в PLSA и в LDA могут заметно отличаться только для тем, редких в документе, и терминов, редких в теме. С одной стороны,

они не несут статистически значимой информации о тематике коллекции. С другой стороны, именно для редких терминов w тематическая модель предсказывает близкую к нулю вероятность $p(w \mid d)$. При появлении этих терминов в документах контрольная перплексия резко увеличивается. В PLSA оценки вероятности $p(w \mid d)$ редких терминов с итерациями могут стремиться к нулю, что выглядит как переобучение, хотя по сути им не является. В LDA вероятности редких терминов никогда не стремятся к нулю благодаря сглаженным байесовским оценкам φ_{wt} и θ_{td}.

Возникает гипотеза, что контрольная перплексия у PLSA хуже, чем у LDA только из-за редких терминов, практически бесполезных для выявления тематики. Другими словами, кажущееся переобучение является побочным следствием гиперчувствительности перплексии к малым вероятностям.

Эксперименты [1] показали, что если из контрольных документов убрать новые термины, то сглаживание не даёт никакого выигрыша, и перплексии PLSA и LDA практически совпадают, см. рис. 1. Этот результат согласуется с рядом недавних исследований [?, ?, ?], также подтверждающих, что для больших коллекций нет существенных различий в качестве моделей PLSA и LDA.

В то же время, сглаживание создаёт ряд проблем: необходимо оптимизировать гиперпараметры, инициализировать β_w для новых терминов w и обеспечивать разреженность при том, что распределение Дирихле не позволяет обнулять вероятности θ_{td} и φ_{wt}.

Идея автоматического выделения терминов, бесполезных для тематической модели, приводит к робастным моделям, которые легко поддаются разреживанию и могут обходиться без байесовской регуляризации и сглаживания [1].

§1.4 Робастные и разреженные тематические модели

Согласно вероятностной модели (1.2), каждый термин w в каждом документе d порождается некоторой темой t. Однако появление отдельных терминов может объясняться не только тематикой документа. Возможны, как минимум, ещё два альтернативных объяснения, условно называемых шумом и фоном.

Шум — это термины, специфичные для конкретного документа, либо редкие термины, относящиеся к темам, слабо представленным в данной коллекции. Тематическая модель даёт слишком низкие значения вероятности $p(w \mid d)$ для таких терминов, то есть не способна объяснить их появление в документах коллекции. Шумовые термины увеличивают перплексию и искажают тематическую модель.

Фон — это общеупотребительные слова, в частности, стоп-слова, не отброшенные на стадии предварительной обработки. Фоновые слова имеют значимые вероятности во многих темах, снижая релевантность тематического поиска.

1.4.1 Робастная тематическая модель с шумом и фоном

Робастная вероятностная тематическая модель SWB (special words with background) представляет собой вероятностную смесь трёх компонент — тематической, шумовой и фоновой [9]:

$$p(w \mid d) = \frac{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}{1 + \gamma + \varepsilon}; \qquad Z_{dw} = \sum_{t \in T} \varphi_{wt}\theta_{td}. \tag{1.18}$$

где *шумовая компонента* $\pi_{dw} \equiv p_{\text{ш}}(w \,|\, d)$ — неизвестное распределение терминов в документе d, *фоновая компонента* $\pi_w \equiv p_{\text{ф}}(w)$ — неизвестное распределение терминов во всей коллекции. Априорные вероятности тематической, шумовой и фоновой компонент модели обозначим, соответственно, $q_{\text{т}} = \frac{1}{1+\gamma+\varepsilon}$, $q_{\text{ш}} = \frac{\gamma}{1+\gamma+\varepsilon}$, $q_{\text{ф}} = \frac{\varepsilon}{1+\gamma+\varepsilon}$, где γ и ε — неотрицательные параметры.

Суть робастной модели в том, что если тематическая компонента Z_{dw} плохо объясняет избыточную частоту n_{dw} некоторого термина w в некотором документе d, то она может быть объяснена альтернативным образом либо шумовой компонентной π_{dw}, либо фоновой π_w.

Требуется найти значения вероятностей φ_{wt}, θ_{td}, π_{dw}, π_w, при которых логарифм правдоподобия достигает максимума:

$$L(D;\Theta,\Phi,\Pi) = \sum_{d\in D}\sum_{w\in d} n_{dw} \ln \frac{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}{1+\gamma+\varepsilon} \;\to \max_{\Theta,\Phi,\Pi}, \tag{1.19}$$

при ограничениях неотрицательности $\pi_{dw} \geqslant 0$, $\pi_w \geqslant 0$ и нормировки

$$\sum_{w\in W} \varphi_{wt} = 1, \quad \sum_{t\in T} \theta_{td} = 1, \quad \sum_{w\in d} \pi_{dw} = 1, \quad \sum_{w\in W} \pi_w = 1.$$

Чтобы получить приближённое решение М-шага, запишем лагранжиан данной задачи при ограничениях нормировки и неотрицательности π_{dw}, π_w, проигнорировав ограничения неотрицательности θ_{td} и φ_{wt}, которые будут выполнены автоматически.

$$\mathscr{L}(D;\Theta,\Phi,\Pi) = \sum_{d\in D}\sum_{w\in d} n_{dw} \ln \frac{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}{1+\gamma+\varepsilon} + \sum_{d\in D}\sum_{w\in d} \kappa_{dw}\pi_{dw} + \sum_{w\in W} \kappa'_w\pi_w -$$
$$-\sum_{t\in T} \lambda_t \Bigg(\sum_{w\in W} \varphi_{wt} - 1\Bigg) - \sum_{d\in D} \mu_d \Bigg(\sum_{t\in T} \theta_{td} - 1\Bigg) - \sum_{d\in D} \nu_d \Bigg(\sum_{w\in d} \pi_{dw} - 1\Bigg) - \nu' \Bigg(\sum_{w\in W} \pi_w - 1\Bigg).$$

Двойственные переменные κ_{dw}, соответствующие ограничениям $\pi_{dw} \geqslant 0$, должны быть неотрицательны и удовлетворять условиям дополняющей нежёсткости

$$\kappa_{dw}\pi_{dw} = 0, \quad d \in D,\ w \in d.$$

Аналогично, для двойственных переменных κ'_w, соответствующих $\pi_w \geqslant 0$:

$$\kappa'_w\pi_w = 0, \quad w \in W.$$

По аналогии со стандартным EM-алгоритмом, на Е-шаге для каждой пары (d,w) вычисляются по формуле Байеса условные вероятности тем $H_{dwt} = p(t \,|\, d,w)$:

$$H_{dwt} = \frac{\varphi_{wt}\theta_{td}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}, \quad t \in T, \tag{1.20}$$

а также условные вероятности того, что термин w является шумом H_{dw} и фоном H'_{dw}:

$$H_{dw} = \frac{\gamma\pi_{dw}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}; \qquad H'_{dw} = \frac{\varepsilon\pi_w}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}. \tag{1.21}$$

Продифференцировав лагранжиан по переменным θ_{td} и φ_{wt} и приравняв нулю производные, получим прежние формулы для φ_{wt} (1.9) и θ_{td} (1.10), с единственным отличием, что теперь H_{dwt} вычисляются по новой формуле (1.20).

Продифференцируем лагранжиан по π_{dw} и приравняем нулю производную:

$$\nu_d = \frac{n_{dw}\gamma}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w} + \kappa_{dw}. \tag{1.22}$$

Домножим обе части этого равенства на π_{dw}, просуммируем по всем терминам $w \in W$, применим условие нормировки вероятностей π_{dw} в левой части и условие дополняющей нежёсткости в правой части. Получим выражение двойственной переменной ν_d через все основные переменные:

$$\nu_d = \sum_{w\in d} n_{dw} \frac{\gamma\pi_{dw}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w} = \sum_{w\in d} n_{dw} H_{dw}. \tag{1.23}$$

Поскольку H_{dw} есть апостериорная вероятность того, что термин w в документе d является шумом, величина ν_d интерпретируется как оценка числа шумовых терминов в документе d.

Проделав аналогичные действия для фоновой компоненты, получим

$$\nu' = \sum_{d\in D} n_{dw} \frac{\varepsilon}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w} + \kappa'_w,$$
$$\nu' = \sum_{d\in D}\sum_{w\in d} n_{dw} \frac{\varepsilon\pi_w}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w} = \sum_{d\in D}\sum_{w\in d} n_{dw} H'_{dw},$$

где ν' интерпретируется как оценка числа фоновых терминов во всей коллекции.

Мультипликативный М-шаг. Домножим обе части (1.22) на π_{dw}, но не будем суммировать по w. Получим формулу М-шага для шумовой компоненты:

$$\pi_{dw} = \frac{1}{\nu_d} n_{dw} \frac{\gamma\pi_{dw}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w} = \frac{n_{dw} H_{dw}}{\sum\limits_{w'\in d} n_{dw'} H_{dw'}}.$$

Аналогично получается формула М-шага для фоновой компоненты:

$$\pi_w = \frac{1}{\nu'} n_{dw} \frac{\varepsilon\pi_w}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w} = \frac{\sum\limits_{d\in D} n_{dw} H'_{dw}}{\sum\limits_{d\in D}\sum\limits_{w'\in d} n_{dw'} H'_{dw'}}.$$

Неотрицательность решения π_{dw}, π_w гарантируется, коль скоро начальные приближения π_{dw}, π_w неотрицательны. Мультипликативный М-шаг приводит к аналогичной проблеме разреженности для переменных π_{dw} и π_w, что и для переменных φ_{wt} и θ_{td}. Если в начальном приближении значение π_{dw} или π_w равно нулю, то оно сохранится и далее на протяжении итераций. Если в начальном приближении π_{dw} или π_w не равно нулю, то оно так и останется ненулевым.

Аддитивный М-шаг решает проблему разреживания шумовой компоненты [1]. Перепишем (1.22) в другом виде:

$$n_{dw}\gamma = (\nu_d - \kappa_{dw})(Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w).$$

Согласно условиям дополняющей нежёсткости, хотя бы одна из двух неотрицательных переменных κ_{dw}, π_{dw} должна быть равна нулю. Отсюда следует, что если $n_{dw}\gamma < \nu_d(Z_{dw} + \varepsilon\pi_w)$, то $\pi_{dw} = 0$ и $\kappa_{dw} > 0$. Если же имеет место противоположное неравенство, то $\kappa_{dw} = 0$ и π_{dw} находится из уравнения $n_{dw}\gamma = \nu_d(Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w)$. Объединяя оба эти случая, получаем итоговое выражение для π_{dw}:

$$\pi_{dw} = \left(\frac{n_{dw}}{\nu_d} - \frac{Z_{dw} + \varepsilon\pi_w}{\gamma}\right)_+. \tag{1.24}$$

Таким образом, если термин w в документе d встречается существенно чаще, чем предсказывают тематическая и фоновая компоненты модели, то его появление объясняется особенностями данного документа, и тогда $\pi_{dw} > 0$.

Аддитивный М-шаг, в отличие от мультипликативного, приводит к автоматическому выбору структуры разреженности матрицы $\Pi = (\pi_{dw})_{D\times W}$.

Эксперименты также показывают, что аддитивный М-шаг предпочтительнее для всех тематических моделей. ToDo[22]

Робастная модификация PLSA-RGEM итерационного процесса PLSA-GEM показана в Алгоритме 1.7. Главное отличие от обычного PLSA в том, что теперь n_{dw} вхождений термина w в документ d распределяются не только между темами $t \in T$, но также между шумовой и фоновой компонентами, пропорционально вероятностям

$$\tilde{H}_{dw} = \left(\tfrac{1}{Z}\varphi_{wt}\theta_{td},\ t \in T;\ \ \tfrac{1}{Z}\gamma\pi_{dw};\ \ \tfrac{1}{Z}\varepsilon\pi_w\right),$$

где Z — нормирующий множитель.

Возможны различные варианты алгоритма PLSA-RGEM: только с шумовой компонентой ($\varepsilon = 0$), только с фоновой компонентой ($\gamma = 0$), с аддитивным и мультипликативным М-шагом. Для простоты в Алгоритме 1.7 показан вариант с шумом и фоном, обновлением параметров по каждой паре (d, w), без сэмплирования, без сглаживания.

Сглаживание вводится в Алгоритм 1.7 заменой частотных оценок (1.9)–(1.10) параметров φ_{wt}, θ_{td} на шагах 6, 7 байесовскими оценками (1.16)–(1.17).

Сэмплирование вводится заменой распределения $\tilde{H}_{dw}$ его эмпирической оценкой, аналогичной (1.12), при вычислении переменных δ.

Результаты экспериментов: робастная модель менее чувствительна к выбору параметра экономного сэмплирования s. ToDo[23]

Два варианта сэмплирования для каждого (d, w): ToDo[24]
(1) пропорциональное распределение вероятности темы–шум–фон, только темы сэмплируются;
(2) сэмплирование из всего распределения $\tilde{H}_{dw}$.

О невозможности оптимизации априорных вероятностей шума и фона. Приравняв нулю производные лагранжиана по γ и ε, нетрудно получить формулы для обновления γ и ε. Однако эксперименты показывают, что с итерациями $\gamma \to \infty$, $\varepsilon \to 0$, что приводит к полному вырождению тематической модели в простейшую униграммную модель. Поэтому параметры γ и ε необходимо фиксировать.

Возможно ли их оптимизация с помощью непараметрического байесовского вывода? ToDo[25]

Алгоритм 1.6. PLSA-RGEM: робастный EM-алгоритм для модели PLSA.

Вход: коллекция D, число тем $|T|$, начальные приближения Θ, Φ, параметры γ, ε;
Выход: распределения: матрицы (φ_{wt}), (θ_{td}), (π_{dw}) и вектор (π_w);

обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$, $\hat{n}_d$, n_{dwt}, ν_{dw}, ν_d, ν, ν'_{dw}, ν'_w, ν' для всех $d \in D$, $w \in W$, $t \in T$;
инициализировать $\pi_{dw} := n_{dw}/n_d$; $\pi_w := n_w/n$; для всех $d \in D$, $w \in d$;
повторять
  для всех $d \in D$, $w \in d$
    если не первый проход коллекции **то**
      $\varphi_{wt} := \hat{n}_{wt}/\hat{n}_t$ для всех $t \in T$;
      $\theta_{td} := \hat{n}_{dt}/\hat{n}_d$ для всех $t \in T$;
      $\pi_w := \nu'_w/\nu'$;
      $\pi_{dw} := \begin{cases} (n_{dw}/\nu_d - Z_{dw}/\gamma - \varepsilon\pi_w/\gamma)_+, & \text{при аддитивном М-шаге;} \\ \nu_{dw}/\nu_d, & \text{при мультипликативном М-шаге;} \end{cases}$
    если не последний проход коллекции **то**
      $Z := \gamma\pi_{dw} + \varepsilon\pi_w + Z_{dw}$;
      для всех $t \in T$ таких, что $n_{dwt} > 0$ или $\varphi_{wt}\theta_{td} > 0$
        $\delta := n_{dw}\varphi_{wt}\theta_{td}/Z$; увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$, $\hat{n}_d$ на $(\delta - n_{dwt})$; $n_{dwt} := \delta$;
      если $\nu_{dw} > 0$ или $\pi_{dw} > 0$ **то**
        $\delta := n_{dw}\gamma\pi_{dw}/Z$; увеличить ν_d, ν на $(\delta - \nu_{dw})$; $\nu_{dw} := \delta$;
      если $\nu'_{dw} > 0$ или $\pi_w > 0$ **то**
        $\delta := n_{dw}\varepsilon\pi_w/Z$; увеличить ν'_w, ν' на $(\delta - \nu'_{dw})$; $\nu'_{dw} := \delta$;
пока распределения (φ_{wt}), (θ_{td}), (π_{dw}), (π_w) не стабилизируются.

§1.5 Принудительное разреживание

Гипотеза разреженности предполагает, что в дискретных распределениях $p(w\,|\,t) = \varphi_{wt}$, $p(t\,|\,d) = \theta_{td}$, $p(t\,|\,d,w) = H_{dwt}$ подавляющее большинство вероятностей равны нулю или очень близки к нулю. Алгоритмы, в которых нулевые значения не хранятся, намного эффективнее по памяти и по скорости. Поэтому для больших коллекций разреженность должна учитываться обязательно.

Модель PLSA не оптимизирует структуру разреженности распределений и требует задавать её через начальное приближение. Отдельные значения θ_{td} и φ_{wt} могут в ходе итераций стремиться к нулю, но, как правило, их доля недостаточна для получения выигрыша в производительности.

Модель LDA также не является разреженной — сглаживание частотных оценок вероятностей приводит к тому, что матрицы Φ и Θ не содержат нулевых значений. Эта проблема имеет много решений, например в [13] предлагается хранить не сами значения θ_{td} и φ_{wt}, а только их разности с фоновыми распределениями.

Принудительное разреживание вводится в любой из описанных выше EM-подобных алгоритмов. По окончании цикла по всем документам $d \in D$ в каждом из $|T|$ распределений $\varphi_{wt} = p(w\,|\,t)$ обнуляется небольшая доля r_φ наименьших значений вероятностей. Аналогично, в каждом из $|D|$ распределений $\theta_{td} = p(t\,|\,d)$ обнуляется небольшая доля r_θ наименьших значений. После обнуления производится перенормировка распределений. Разреживания начинаются с некоторой итерации i_0, чтобы к этому моменту в распределениях правильно выделились малые вероятности. Кроме

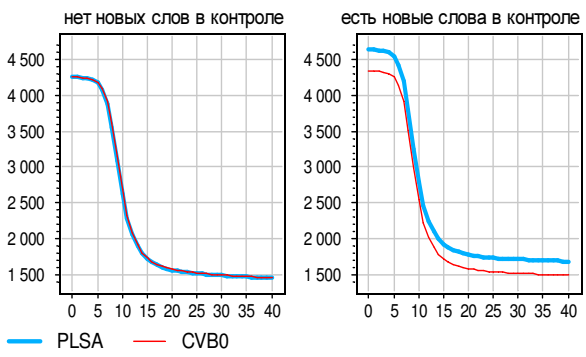

Рис. 1. Сглаживание даёт преимущество только когда в контроле есть новые термины (метод CVB0 — это PLSA-GEM со сглаживанием но без сэмплирования).

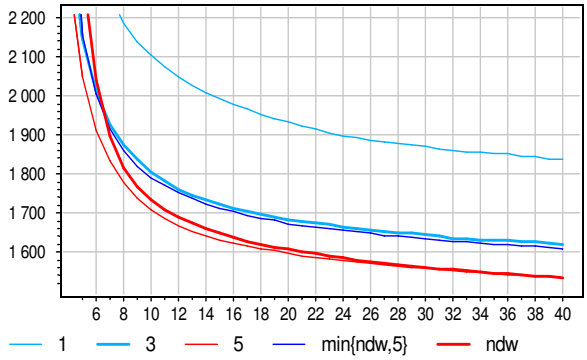

Рис. 2. При экономном сэмплировании пяти тем для каждой пары (d, w) перплексия не хуже, чем при сэмплировании n_{dw} тем. Но одной или трёх тем не достаточно.

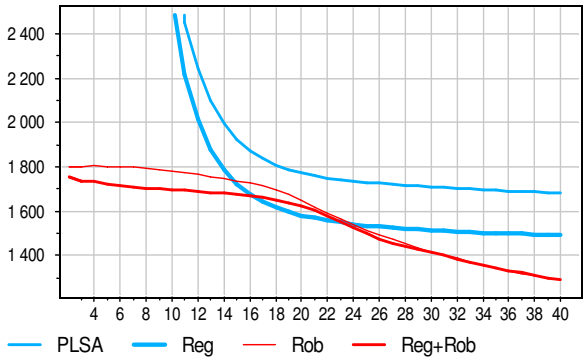

Рис. 3. Робастность сильнее уменьшает перплексию PLSA, чем сглаживание. Сглаживание не улучает робастную модель.

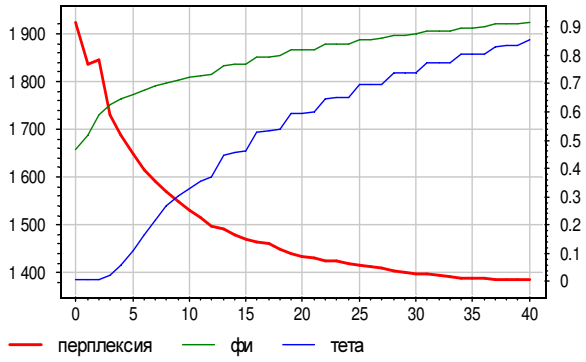

Рис. 4. В процессе разреживания доля нулевых φ_{wt} и θ_{td} (отложена по правой оси) увеличивается при монотонном уменьшении перплексии.

того, разреживания имеет смысл делать не на каждой итерации, а хотя бы через одну, чтобы восстановить адекватность модели. В наших экспериментах разреживание выполнялось на итерациях с номерами вида $i = i_0 + kd,\ \ k = 1, 2, \ldots$ при $r_\varphi = 0.05$, $r_\theta = 0.05$, $i_0 = 10$, $d = 2$.

Принудительное разреживание может увеличивать перплексию моделей PLSA и LDA. Кроме того, при сильном разреживании в PLSA может происходить обнуление модели $p(w \mid d) = 0$ при $n_{dw} > 0$, и тогда перплексия уходит в бесконечность.

Робастные модели допускают принудительное разреживание без ухудшения перплексии и одновременно исключают ситуацию бесконечной перплексии, так как нулевое значение Z_{dw} компенсируется ненулевым значением шума π_{dw} или фона π_w. Чем больше γ и ε, тем более разреженной может быть тематическая компонента модели. В экспериментах разреженность матриц Θ и Φ в робастном PLSA достигала более 90% без потери качества модели или даже с незначительным улучшением, рис. 4.

Эксперименты: сравнение разреживания для PLSA, LDA и робастных. ToDo[26]

Эксперименты с выбором стратегии разреживания. ToDo[27]

Эксперименты на реальных данных Эксперименты производились на двух коллекциях.

Коллекция RuDis содержала $|D| = 2000$ авторефератов диссертаций на русском языке; суммарная длина $n \approx 8.7 \cdot 10^6$, объём словаря $|W| \approx 3 \cdot 10^4$. Контрольная коллекция D' состояла из 200 авторефератов. Предварительно производилась лемматизация и отбрасывались стоп-слова.

Коллекция NIPS содержала $|D| = 1566$ текстов статей научной конференции Neural Information Processing Systems на английском языке; суммарная длина $n \approx 2.3 \cdot 10^6$, объём словаря $|W| \approx 1.3 \cdot 10^4$. Контрольная коллекция D' состояла из 174 документов. Предварительно производился стемминг и отбрасывались стоп-слова.

Качество модели оценивалось *перплексией* контрольной коллекции D' документов, не включённых в обучающую коллекцию. Каждый контрольный документ d случайным образом разделялся на две половины, d' и d''. Параметры θ_{td} и ν_d оценивались по d'. Параметры φ_{wt} и π_w оценивались по обучающей выборке D. Параметры π_{dw} оценивались для каждой пары (d, w) согласно (1.24). Перплексия вычислялась по вторым половинам d'' контрольных документов.

На рис. 1–4 показаны зависимости перплексии от числа итераций (одна итерация — один проход по коллекции). Число итераций 40; число тем $|T| = 100$; параметры сглаживания $\alpha_t = 0.5$, $\beta_w = 0.01$; параметры робастности $\gamma = 0.3$, $\varepsilon = 0.1$.

- Проверить, действительно ли $\pi_w > 0$ для стоп-слов; $\pi_{dw} > 0$ для слов, не типичных ни для одной из тем. Просмотреть эти слова и сравнить их со словами из тем. Показать одни и те же темы как ранжированные списки терминов в неробастной модели, робастной модели, робастной модели после разреживания. ToDo[28]

- Если добавить фоновую компоненту, то приведёт ли это к улучшению HOP и степени разреженности? Позволяет ли фоновая компонента избавиться от предварительного отбрасывания стоп-слов?

Позволяет ли фоновая компонента избавиться от предварительной фильтрации стоп-слов? ToDo[29]

1.5.1 Принудительное разреживание

Гипотеза разреженности предполагает, что в дискретных распределениях $p(w \mid t) = \varphi_{wt}$, $p(t \mid d) = \theta_{td}$, $p(t \mid d, w) = H_{dwt}$ подавляющее большинство вероятностей равны нулю или очень близки к нулю. Алгоритмы, в которых нулевые значения не хранятся, намного эффективнее по памяти и по скорости. Поэтому для больших коллекций разреженность должна учитываться обязательно.

Модель PLSA не оптимизирует структуру разреженности распределений и требует задавать её через начальное приближение. Отдельные значения θ_{td} и φ_{wt} могут в ходе итераций стремиться к нулю, но, как правило, их доля недостаточна для получения выигрыша в производительности.

Модель LDA также не является разреженной — сглаживание частотных оценок вероятностей приводит к тому, что матрицы Φ и Θ не содержат нулевых значений. Эта проблема имеет много решений, например в [13] предлагается хранить не сами значения θ_{td} и φ_{wt}, а только их разности с фоновыми распределениями.

Принудительное разреживание путём обнуления малых значений θ_{td} и φ_{wt} с последующей перенормировкой может приводить к ухудшению качества моделей PLSA и LDA, особенно на первых итерациях. Кроме того, не исключается возможность ситуации, когда $p(w\,|\,d) = 0$, $n_{dw} > 0$ и перплексия уходит в бесконечность.

Робастные модели допускают принудительное разреживание и одновременно исключают ситуацию бесконечной перплексии, так как нулевое значение Z_{dw} компенсируется ненулевым значением шумовой компоненты $p_{\text{ш}}(w\,|\,d)$. Чем больше γ, тем более разреженной может быть тематическая компонента модели.

Эвристика принудительного разреживания. В эксперименте с робастным PLSA на каждой итерации принудительно обнулялись 5% наименьших значений θ_{td} и φ_{wt}. При этом разреженность матриц Θ и Φ достигала порядка 90% без существенной потери качества модели (рис. 4).

Сравнить с разреживанием в неробастных моделях ToDo[30]

§1.6 Критерии качества вероятностных тематических моделей

1.6.1 Критерии, проверяющие гипотезу условной независимости

Гипотеза условной независимости $p(w\,|\,d,t) = p(w\,|\,t)$ чрезвычайно важна для вероятностных тематических моделей. Именно она обеспечивает переход к компактному представлению данных $F \approx \Phi\Theta$. Для её проверки не требуется выделять контрольную выборку, что является преимуществом данного типа критериев.

Оба распределения легко оцениваются в ЕМ-алгоритме:

$$\hat{p}(w\,|\,d,t) = \frac{n_{dwt}}{\hat{n}_{dt}}, \quad t \in T,\ d \in D;$$

$$\hat{p}(w\,|\,t) = \frac{\hat{n}_{wt}}{\hat{n}_{t}}, \quad t \in T.$$

Рассмотрим статистические тесты, проверяющие нулевую гипотезу о том, что различия между этими распределениями незначимы, точнее, что выборка с эмпирическим распределением $\hat{p}(w\,|\,d,t)$ могла быть получена из генеральной совокупности с распределением $\hat{p}(w\,|\,t)$.

Число $\hat{n}_{dt}$ интерпретируется как длина части документа d, связанной с темой t, а число n_{dwt} — как число вхождений термина w в документ d, связанных с темой t. Введём ожидаемое число вхождений термина w в документ d, связанных с темой t:

$$E_{dwt} = \hat{n}_{dt}\hat{p}(w\,|\,t).$$

Критерий X^2 Пирсона основан на вычислении статистики хи-квадрат, которая является естественной мерой различия двух распределений:

$$X^2_{dt} = \sum_{w \in W_{dt}} \frac{(E_{dwt} - n_{dwt})^2}{E_{dwt}} = \hat{n}_{dt} \sum_{w \in W_{dt}} \frac{\big(\hat{p}(w\,|\,t) - \hat{p}(w\,|\,d,t)\big)^2}{\hat{p}(w\,|\,t)},$$

где $W_{dt} = \big\{w \in W\colon E_{dwt} > 0\big\}$. Если значение X^2_{dt} превышает $(1-\alpha)$-квантиль распределения хи-квадрат $\chi^2_{k,1-\alpha}$ с числом степеней свободы $k = |W_{dt}| - 1$, то нулевая гипотеза отвергается.

Условием применимости асимптотики χ^2_k считается наличие достаточного числа наблюдений во всей выборке, $\hat{n}_{dt} \geqslant 50$, а также достаточного ожидаемого числа наблюдений каждого термина, $E_{dwt} \geqslant 5$. Второе требование в типичном случае не выполняется для большинства терминов w, так как распределение $\hat{p}(w \mid t)$, как правило, разрежено, более того, мощность словаря W_{dt} может превышать длину документа $\hat{n}_{dt}$. Таким образом, в нашем случае критерий Пирсона применять нельзя. Для случая разреженных распределений больше подходят статистики G^2 и D^2.

Статистика G^2 определяется через дивергенцию Кульбака–Лейблера, и для неё также справедливо асимптотическое распределение χ^2_k с тем же числом степеней свободы, но при менее жёстких требованиях к числу наблюдений:

$$G^2_{dt} = 2 \sum_{w \in W_{dt}} n_{dwt} \ln \frac{n_{dwt}}{E_{dwt}}.$$

Статистика D^2 — это поправка к статистике X^2, предложенная Зельтерманом в [38] специально для случая разреженных распределений:

$$D^2_{dt} = \sum_{w \in W_{dt}} \frac{(E_{dwt} - n_{dwt})^2 - n_{dwt}}{E_{dwt}}.$$

Эта статистика имеет асимптотически нормальное распределение. Особенности её применения обсуждаются в [26, 18].

Семейство функций расстояния Кресси–Рида. Для сравнения эмпирической функции вероятности $\hat{p}(w)$, оцененной по выборке длины n, с истинной функцией вероятности $p(w)$ принято использовать функции расстояния, придающие больший вес малым вероятностям:

$$\mathrm{KL}(\hat{p} \,\|\, p) = \sum_w \hat{p}(w) \ln \frac{\hat{p}(w)}{p(w)} \text{ — дивергенция Кульбака–Лейблера;} \tag{1.25}$$

$$X^2(\hat{p}, p) = \sum_w \frac{\big(p(w) - \hat{p}(w)\big)^2}{p(w)} \text{ — ненормированная } \chi^2\text{-статистика;} \tag{1.26}$$

$$H^2(\hat{p}, p) = \sum_w \Big(\sqrt{p(w)} - \sqrt{\hat{p}(w)}\Big)^2 \text{ — расстояние Хеллингера.} \tag{1.27}$$

Эти и другие «разумные» функции расстояния обобщаются (с точностью до константного множителя) параметрическим семейством дивергенций Кресси–Рида [10, 22]:

$$\mathrm{CR}_\lambda(\hat{p} : p) = \frac{2}{\lambda(\lambda+1)} \sum_w \hat{p}(w) \left(\left(\frac{\hat{p}(w)}{p(w)} \right)^\lambda - 1 \right).$$

Перестановочный тест основан на использовании эмпирического распределения статистики, полученного путём сэмплирования большого числа выборок в условиях истинности нулевой гипотезы. Перестановочные тесты применяются в тех случаях, когда функция распределения статистики неизвестна или имеет слишком сложный вид или её известные асимптотики не достаточно точны.

Пусть S — одна из статистик X^2, G^2, D^2. Зафиксируем тему t. Сгенерируем N независимых выборок терминов из распределения $\hat{p}(w\,|\,t)$. Для каждой из них вычислим эмпирическое распределение $\hat{p}(w)$ и значение статистики S. По выборке значений статистики $\{S_1, \ldots, S_N\}$ построим эмпирическое распределение $\hat{F}_t(S)$ и найдём его $(1-\alpha)$-квантиль $\hat{F}_{t,1-\alpha}$. Число N должно быть порядка 10^3 при $\alpha = 0.05$.

Обозначим через S_{dt} значение статистики S, вычисленное по распределению $\hat{p}(w\,|\,d,t)$ для заданных $t \in T$ и $d \in D$. Поскольку распределение $\hat{F}_t(S)$ построено в условиях истинности нулевой гипотезы, неравенство $S_{dt} > \hat{F}_{t,1-\alpha}$ является критерием отклонения нулевой гипотезы для документа d на уровне значимости α.

Заметим, что квантиль $\hat{F}_{t,1-\alpha}$ достаточно вычислить один раз для каждой темы t и использовать для всех документов $d \in D$, что даёт значительную экономию времени. Однако при изменении распределения $\hat{p}(w\,|\,t)$ распределение $\hat{F}_t(S)$ и его квантиль придётся пересчитать заново.

Оценки средней несогласованности для документов и тем. Введём индикатор события «*тема t не согласована в документе d* при уровне значимости α»:

$$B_{dt}(\alpha) = \big[S_{dt} > \hat{F}_{t,1-\alpha}\big];$$

Определим *среднюю несогласованность* темы, документа и тематической модели в целом при уровне значимости α:

$$B_t(\alpha) = \sum_{d \in D} \frac{\hat{n}_{dt}}{\hat{n}_t} B_{dt}(\alpha) \quad \text{— средняя несогласованность темы } t;$$

$$B_d(\alpha) = \sum_{t \in T} \frac{\hat{n}_{dt}}{n_d} B_{dt}(\alpha) \quad \text{— средняя несогласованность документа } d;$$

$$B(\alpha) = \sum_{d \in D} \sum_{t \in T} \frac{\hat{n}_{dt}}{n} B_{dt}(\alpha) \quad \text{— средняя несогласованность модели.}$$

Это нормированные величины, принимающие значения из отрезка $[0, 1]$. Чем меньше средняя несогласованность, тем лучше модель описывает соответствующую тему t, документ d или всю коллекцию в целом.

Критерий условной независимости. В [20] предлагается ещё один критерий, оценивающий степень несоответствия темы $t \in T$ гипотезе условной независимости. Он основан на дивергенции Кульбака–Лейблера и может быть легко вычислен в EM-алгоритме на каждом проходе коллекции:

$$\mathrm{KL}_t = \mathrm{KL}\Big(\hat{p}(d,w\,|\,t) \;\Big\|\; \hat{p}(d\,|\,t)\,\hat{p}(w\,|\,t)\Big) = \sum_{d,w} \frac{n_{dwt}}{\hat{n}_t} \ln \frac{n_{dwt}}{E_{dwt}}.$$

Статистика $G_t^2 = \sum\limits_{d \in D} G_{dt}^2 = 2\hat{n}_t \, \mathrm{KL}_t$ имеет асимптотически распределение χ_k^2 с числом степеней свободы $k = \sum\limits_{d \in D} |W_{dt}| - |W| - |D| + 1$. В силу разреженности распределения $\hat{p}(d, w \,|\, t)$ вместо критерия хи-квадрат лучше применять перестановочный тест. Гипотеза условной независимости принимается для темы t, когда значение статистики G_t^2 меньше критического.

Выделение несогласованных тем. Статистические критерии позволяют находить «неудачные» темы, которые целесообразно разбивать на подтемы, непосредственно во время итераций EM-алгоритма. Темы можно ранжировать и сравнивать по значениям средней согласованности $B_t(\alpha)$ или статистики G_t^2. Заметим, что сравнивать темы по значению дивергенции KL_t некорректно, так как только после умножения на «длину темы» $\hat{n}_t$ получается величина $G_t^2 = 2\hat{n}_t \, \mathrm{KL}_t$, имеющая (асимптотически) одинаковое распределение для всех тем.

Эксперименты Влады Целых ToDo[31]

1.6.2 Критерии качества классификации документов

Оценивание качества тематической модели упрощается в тех случаях, когда она строится с целью классификации или поиска документов. Каждый документ описывается $|T|$-мерным вектором тем $\theta_d = \big(p(t \,|\, d)\big)_{t \in T}$. Качество модели определяется тем, насколько хорошо классифицируются документы, представленные этими векторами.

Пусть каждый документ $d \in D$ относится к классу $y_d \in Y$, алгоритм классификации $a \colon \mathbb{R}^{|T|} \to Y$ относит документ d к классу $a_d = a(\theta_d)$. В задачах информационного поиска и категоризации текстов качество классификации принято измерять в терминах точности и полноты [24].

Точность (precision) относительно класса $y \in Y$ определяется как доля правильно классифицированных документов среди всех документов, отнесённых алгоритмом a к классу y:

$$P_y(a) = \frac{\#\big\{d \in D \colon a_d = y_d = y\big\}}{\#\big\{d \in D \colon a_d = y\big\}}.$$

Полнота (recall) относительно класса $y \in Y$ определяется как доля правильно классифицированных документов среди всех документов класса y:

$$R_y(a) = \frac{\#\big\{d \in D \colon a_d = y_d = y\big\}}{\#\big\{d \in D \colon y_d = y\big\}}.$$

Чем больше значения точности и полноты, тем выше качество классификации.

В задачах информационного поиска обычно рассматривают два класса — документ либо «релевантен», либо «нерелевантен»; точность и полноту определяют только относительно класса релевантных документов.

Задачи категоризации, как правило, являются *многоклассовыми*, $|Y| \gg 2$. В таких случаях точность и полноту усредняют по всем классам.

В качестве агрегированного показателя, объединяющего точность P и полноту R, принято использовать F_1-меру:

$$F_1 = \frac{2PR}{P+R}.$$

1.6.3 Критерии качества тематического поиска

Описать идею разбиения каждого документа на части и поиска одних частей по другим. Качество поиска может измеряться с помощью Mean Average Precision. ToDo[32]

§1.7 Иерархические тематические модели

Для больших коллекций текстовых документов естественно строить иерархии вложенных друг в друга тем (называемых также категориями, каталогами или рубриками), чтобы упростить поиск документов. Люди привыкли использовать такие иерархии, интуитивно разделяя каждую тему на более узкие подтемы. Это разделение, как правило, субъективно, неоднозначно и вызывает споры, однако специалисты рано или поздно договариваются о внутренней структуре своей предметной области. По всей видимости, отношение «тема–подтема» объективно существует. Нашей ближайшей целью будет вероятностная формализация этого отношения. Затем мы рассмотрим задачу восстановления иерархической тематической структуры по коллекции текстовых документов.

В статье [37] приводится обзор иерархических тематических моделей и отмечается, что проблемы оптимизации структуры и оценивания качества тематических иерархий по коллекции текстовых документов всё ещё остаются открытыми. Многие иерархические модели имеют те или иные неестественные ограничения: либо фиксируется число уровней, либо фиксируется число подтем в каждой теме или на каждом уровне, либо документ не может относиться к темам из различных ветвей дерева, либо темы не могут иметь общую подтему, либо темам во внутренних узлах не сопоставляется распределение на множестве терминов.

Мы рассмотрим тематические иерархии двух типов: деревья и сети. Сеть — это направленный ациклический граф. В отличие от деревьев, сети допускают наличие тем, имеющих общие подтемы.

1.7.1 Требования к иерархической тематической модели

Требования к тематическим иерархиям могут быть продиктованы различными приложениями, поэтому в литературе можно найти прямо противоположные критерии и рекомендации. Мы будем исходить из того, что иерархия необходима для организации тематического поиска и навигации по большой коллекции научных документов; типичный документ — статья объёмом около 10 страниц; типичный объём коллекции — от нескольких тысяч до десятков миллионов документов.

Сформулированные ниже требование в дальнейшем будут уточняться и переводиться на формальный язык вероятностных тематических моделей.

1. *Интерпретируемость* отношения тема–подтема. В подтеме используются те же термины, что и в родительской теме. Подтема отличается от родительской темы более частым употреблением специфичного подмножества терминов.

Если документ относится к подтеме, то он относится и к родительской теме.

Если документ не относится к теме, то он не относится и к её подтемам.

2. *Связность.* Иерархия должна представляться направленным ациклическим графом. В этом графе должен существовать путь от корневой вершины до любой другой вершины. Некорневая вершина (тема) может иметь несколько родительских вершин (надтем), что необходимо для представления тем, находящихся на стыке научных направлений.

3. *Сбалансированность.* Тематическая иерархия не должна иметь ограничений по глубине и ширине. Терминальные темы не должны различаться по числу документов более чем на порядок. Если к теме относится слишком много документов, она должна дробиться на подтемы.

4. *Гранулированность.* Списки документов и подтем для каждой темы должны быть небольшими, для удобства навигации по тематической иерархии. У каждой темы должно быть ориентировочно до 50 документов, до 20 подтем, до 10 надтем.

5. *Разреженность.* С каждым документом и с каждым термином может быть связано ориентировочно до 20 тем. Предполагается, что модель, не перегруженная лишними связями, будет лучше интерпретироваться, быстрее строиться, лучше масштабироваться на большие коллекции и даёт более релевантные результаты при тематическом поиске.

6. *Адаптивность.* Тематическая иерархия должна перестраиваться по мере роста коллекции документов. В то же время, полностью автоматическое построение иерархии нежелательно, так как окончательное решение о разбиении тем на подтемы должно оставаться за экспертами. Возможный компромисс заключается в том, чтобы обновление иерархии происходило в *полуавтоматическом* режиме. При построении модели генерируется список рекомендованных операций по улучшению иерархии. Этот список может исполняться автоматически или модерироваться экспертами.

7. *Устойчивость.* При различных начальных приближениях должна строиться одна и та же сеть. Неоднозначность неотрицательного матричного разложения должна разрешаться в пользу наиболее интерпретируемой модели.

8. *Робастность.* Модель должна быть устойчива и к «шуму» — появлению редких слов, специфичных для отдельного документа, и к «фону» — появлению частых слов, не специфичных ни для одной темы.

9. *Обобщающая способность.* Модель должна как можно точнее предсказывать тематику новых документов. Качество модели должно оцениваться по стандартным критериям перплексии контрольных данных, независимости тем, полноты и точности категоризации размеченной выборки документов.

10. *Согласованность с внешними рубрикаторами.* Тематическая иерархия должна учитывать сложившиеся представления о структуре областей знания, задаваемые одним или несколькими «внешними» рубрикаторами. Рубрикаторы могут содержать важную, но неполную и неточную информацию. Они могут иметь относительно хорошо проработанные верхние уровни, но с глубиной их качество может ухудшаться. Степень проработки может отличаться в разных темах. Несколько рубрикаторов могут строиться на различных принципах и противоречить друг другу.

11. *Согласованность с экспертными оценками.* Эксперты могут отмечать найденные моделью связи тема–подтема, термин–тема и документ–тема как нерелевантные; переранжировать эти связи; предлагать свои релевантные связи; разделять тему на насколько дочерних подтем или тем того же уровня; сливать нескольких подтем

в одну; удалять все дочерние подтемы у данной темы. Тематическая модель должна корректно обновляться после таких операций.

12. *Мультиязычность.* Документы на разных языках по одной теме должны относиться к одной вершине. У каждой темы должны быть названия на всех тех языках, на которых в данной теме или её дочерних имеется хотя бы один документ.

Создание иерархической тематической модели, удовлетворяющей всей совокупности перечисленных требований, является сложной задачей и считается открытой научной проблемой.

1.7.2 Определение тематического дерева

Гипотеза о существовании тематического дерева. Рассмотрим дерево с множеством вершин V и корнем $t_0 \in V$. Вершины дерева соответствуют темам. Каждой теме $t \in V$ соответствует множество её подтем — дочерних вершин в дереве $S_t \subset V$. Каждое ребро дерева соответствует паре «тема–подтема» (t, s), $s \in S_t$. Если $S_t = \varnothing$, то тема t называется *терминальной* или *листом* тематического дерева. Для каждой вершины t в дереве V существует только одна родительская вершина, следовательно, только один путь $(t_0, \ldots, t)$ от корня дерева t_0 до темы t.

Ранее мы предполагали, что каждое вхождение термина w в документ d связано только с одной темой t. Теперь примем за аксиому другие предположения:

1) если пара (d, w) связана с темой t, то она связана и со всеми темами выше вершины t на пути до корня t_0;

2) если пара (d, w) не связана с темой t, то она не связана и со всеми подтемами в поддереве ниже вершины t.

Этих двух предположений уже достаточно, чтобы построить иерархическую вероятностную тематическую модель.

Вероятностная интерпретация отношения «тема–подтема». Каждому ребру тематического дерева (t, s) соответствует условная вероятность $p(s \mid t)$ того, что термин документа, связанный с темой t, связан также с подтемой $s \in S_t$:

$$p(s \mid t) = \frac{p(t, s)}{p(t)} = \frac{p(s)}{p(t)}. \tag{1.28}$$

Если рассматривать коллекцию документов как выборку троек (d, w, t), то частотной оценкой этой условной вероятности будет $\hat{p}(s \mid t) = n_s / n_t$ — доля троек, связанных с подтемой s, среди всех троек, связанных с темой t.

Условные вероятности подтем удовлетворяют ограничениям нормировки, которые, в силу (1.28), допускают две эквивалентные записи:

$$\sum_{s \in S_t} p(s \mid t) = 1, \quad \sum_{s \in S_t} p(s) = p(t), \quad t \in V. \tag{1.29}$$

Обозначим через T множество тем, соответствующих терминальным вершинам дерева V. Условие нормировки

$$\sum_{t \in T} p(t) = 1. \tag{1.30}$$

Алгоритм 1.7. Порождение тематического дерева.

Выход: дерево V;
вероятности $p(s\,|\,t)$ для всех рёбер (t,s) дерева V;
вероятности $p(w\,|\,t)$, $p(t\,|\,d)$ для всех вершин t дерева V;

создать корень дерева t_0; положить $V := \{t_0\}$; $p(t_0) := 1$;
задать распределения $p(w\,|\,t_0) = p(w)$ и $p(d\,|\,t_0) = p(d)$;
пока есть терминальная тема $t \in T$, которую нужно расщепить на подтемы
    создать множество подтем S_t, добавить их в T, удалить t из T;
    задать распределение $p(s\,|\,t)$;
    задать распределения $p(w\,|\,s)$ для всех $s \in S_t$;
    задать вероятности $p(s\,|\,d)$ для всех $s \in S_t$, $d \in D$;

выполняется именно для этого множества, а не для всего множества тем в дереве V. Из (1.29) следует, что условие нормировки останется справедливым, если заменить любое из множеств S_t его родительской темой t, а также если делать такие замены многократно в произвольном порядке. В частности, для корневой темы $p(t_0) = 1$.

При разделении темы t на подтемы $s \in S_t$ условные распределения для подтем $p(w\,|\,s)$ и $p(s\,|\,d)$ должны удовлетворять требованиям нормировки

$$\sum_{w\in W} p(w\,|\,s) = 1, \quad s \in S_t; \qquad\qquad \sum_{s\in S_t} p(s\,|\,d) = p(t\,|\,d), \quad d \in D. \tag{1.31}$$

Распределения $p(s\,|\,w) = p(w\,|\,s)\frac{p(s)}{p(w)}$ и $p(d\,|\,s) = p(s\,|\,d)\frac{p(d)}{p(s)}$ также должны быть нормированы, откуда следуют ещё две серии тождеств:

$$\sum_{s\in S_t} p(w\,|\,s)\,p(s) = p(w\,|\,t)\,p(t), \quad w \in W; \qquad \sum_{d\in D} p(s\,|\,d)\,p(d) = p(s), \quad s \in S_t. \tag{1.32}$$

Документы во внутренних вершинах. В некоторых приложениях важно, чтобы документы и термины могли относиться не только к терминальным вершинам, но и к любым внутренним вершинам тематического дерева. В частности, это могут быть документы, относящиеся сразу к нескольким подтемам, к возникающим новым подтемам или подтемам, слабо представленным в данной коллекции.

Для каждой внутренней вершины $t \in V\backslash T$ создаётся выделенная терминальная вершина — подтема $\sigma_t \in S_t$. Если документ или термин попадает в σ_t, то считается, что он остался в теме t. В терминах кластеризации выделенная подтема σ_t — это специальный «фоновый» кластер, к которому относится всё, что не удалось с уверенностью отнести к другим кластерам.

К выделенной подтеме σ_t естественно предъявлять требование минимизации числа документов и описывать её тем же распределением, что и родительскую тему t, либо вообще не накладывать на распределение никаких ограничений.

Процесс порождения тематического дерева описывается Алгоритмом 1.8. Его можно непосредственно применять для генерации модельных данных. В этом алгоритме не накладывается никаких ограничений на структуру дерева — глубина, ширина, число ветвлений в каждой вершине могут быть выбраны произвольным образом.

Не вводятся ограничения и на вид распределений $p(s\,|\,t)$, $p(w\,|\,s)$, $p(s\,|\,d)$, кроме ограничений нормировки (1.29), (1.31), (1.32). В частности, ничего не предполагается об их разреженности или принадлежности тому или иному параметрическому семейству. Ограничения на структуру дерева и вид распределений при необходимости могут вводиться дополнительно.

1.7.3 Восходящее построение тематического дерева

В [37] предложен восходящий метод, в котором тематическая сеть строится по слоям снизу вверх. Сначала строится обычная «плоская» тематическая модель. Затем полученные темы $p(w\,|\,t)$ рассматриваются как виртуальные документы, для которых строится следующая тематическая модель из существенно меньшего числа тем, и так далее, пока не будет получена единственная корневая тема. Исследованы две разновидности модели: в первой (hvHDP) каждой внутренней вершине соответствует некоторая тема t и распределения $p(w\,|\,t)$ и $p(d\,|\,t)$; во второй (htHDP) только терминальные вершины являются темами, а сеть описывает иерархическую кластерную структуру на множестве тем.

Недостатки этого подхода следуют из того, что все терминальные вершины располагаются на одном уровне, и документы относятся только к терминальным вершинам. Такая сеть не соответствует некоторым из наших требований, а именно, сбалансированности, гранулированности, согласованности с внешними рубрикаторами и с экспертными оценками.

Поэтому далее более подробно рассматриваются нисходящие методы, в которых темы могут расщепляться на подтемы по необходимости.

1.7.4 Нисходящее построение тематического дерева

Рассмотрим нисходящий итерационный процесс, в котором тематическое дерево строится слой за слоем, от корня к листьям, см. Алгоритм 1.9. Сначала дерево состоит из единственной корневой вершины t_0, для которой $\varphi_{wt_0} = p(w)$ и $\theta_{t_0 d} = 1$. Затем на каждой итерации происходит расщепление некоторых тем на подтемы.

Пусть в начале итерации имеется тематическое дерево V, и для всех его терминальных вершин $t \in T$ известны распределения $\varphi_{wt} = p(w\,|\,t)$ и $\theta_{td} = p(t\,|\,d)$. Выбирается подмножество расщепляемых терминальных вершин $R \subseteq T$. Для каждой темы $t \in R$ создаётся множество подтем S_t, сначала содержащее только две подтемы. Для каждой подтемы $s \in S_t$ оцениваются параметры φ_{ws}, θ_{sd} и $p(s\,|\,t)$. Число подтем увеличивается до тех пор, пока среди подтем множества S_t не появится двух слишком близких тем. Рассмотрим эти шаги подробнее.

Критерий расщепления темы на подтемы (шаг 2). Если бы на каждой итерации расщеплялись все вершины, $R = T$, то было бы построено идеально сбалансированное дерево, в котором все терминальные вершины находились бы на одинаковом расстоянии от корня. На практике требование сбалансированности является избыточным и расщепления производятся для отдельных тем по необходимости.

Критерием включения темы t в подмножество расщепляемых тем R будем считать одно из следующих условий, либо их сочетание:

1) средняя несогласованность темы $B_t(\alpha)$ превышает порог;

Алгоритм 1.8. Иерархический PLSA-GEM: построение тематического дерева.

Вход: коллекция документов D;
Выход: тематическое дерево, распределения Θ и Φ;

$T := \{t_0\}$; $\theta_{t_0 d} := 1$ для всех $d \in D$;
пока множество расщепляемых терминальных вершин $R \subseteq T$ не пусто
  $D' := \Big\{ d \in D \colon \sum_{t \in R} \theta_{td} > 0 \Big\}$;
  $\sigma_{dw} := \sum_{t \in T \setminus R} \varphi_{wt} \theta_{td}$ для всех $d \in D'$, $w \in d$.
  для каждого $t \in R$ создать множество S_t из двух подтем; $T := T \setminus \{t\} \cup S_t$;
  задать начальные приближения φ_{ws}, θ_{sd} для всех $s \in S_t$, $t \in R$;
  для всех $k := 2, 3, \ldots$ (k — число подтем, мощность множества S_t)
    обнулить $\hat{n}_{ws}$, $\hat{n}_{ds}$, $\hat{n}_s$, $\hat{n}_{dt}$, n_{dws} для всех $d \in D'$, $w \in W$, $t \in R$, $s \in S_t$;
    повторять
      для всех $d \in D'$, $w \in d$
        $Z := \sigma_{dw} + \sum_{s \in S} \varphi_{ws} \theta_{sd}$;
        для всех $t \in R$, $s \in S_t$
          $\delta := n_{dw} \varphi_{ws} \theta_{sd} / Z$;
          увеличить $\hat{n}_{ws}$, $\hat{n}_{ds}$, $\hat{n}_s$, $\hat{n}_{dt}$ на $(\delta - n_{dws})$;
          $n_{dws} := \delta$;
        если пора обновить параметры Φ_S, Θ_S **то**
          $\varphi_{ws} := \hat{n}_{ws} / \hat{n}_s$ для всех $w \in W$, $s \in S$;
          $\theta_{sd} := \theta_{td}\, \hat{n}_{ds} / \hat{n}_{dt}$ для всех $d \in D'$, $t \in R$, $s \in S_t$;
    пока Θ_S и Φ_S не стабилизируются.
    если среди подтем S_t не появилось слишком близких тем **то**
      добавить в S_t подтему s; задать начальные приближения φ_{ws}, θ_{sd};
    иначе
      вернуться к оптимальному S_t; восстановить φ_{ws}, θ_{sd}, $s \in S_t$;

2) статистика G_t^2 в критерии условной независимости превышает порог;
3) число документов, для которых $\hat{n}_{dt} > \delta \hat{n}_d$, превышает порог;
4) соответствующая тема во внешнем рубрикаторе имеет подтемы;
5) пользователи создали в теме t новые подтемы, приписав им релевантные документы или термины.

На шаге 3 выделяется подмножество документов D', связанных хотя бы с одной из расщепляемых тем. Если документ d не связан с темой t, $\theta_{td} = 0$, то он не может быть связан и с подтемами $s \in S_t$, поскольку $\theta_{sd} \leqslant \theta_{td}$ согласно (1.31). Таким образом, чем глубже в дереве находится расщепляемая тема, тем меньше выборка документов, по которой строится тематическая модель для её подтем.

Формирование начального приближения (шаг 6). Вызывается рассмотренный выше Алгоритм **??**, которому подаётся на вход множество документов $D := D'$ и множество тем $T := S_t$, состоящее из двух тем. На выходе алгоритма получаются распределения φ_{ws} для двух тем $s \in S_t$ и по паре значений θ_{sd}, $s \in S_t$, для каждого

документа $d \in D$, нормированных на единицу, $\sum_{s \in S_t} \theta_{sd} = 1$. При замене темы t множеством её подтем S_t должна сохраниться нормировка $\sum_{t \in T} \theta_{td} = 1$ для распределений θ_d. Поэтому вычисленные Алгоритмом **??** значения θ_{sd} необходимо умножить на θ_{td}:

$$\theta_{sd} := \theta_{sd}\theta_{td}, \quad s \in S_t,\ d \in D.$$

Оценивание параметров (шаги 8–19). Возьмём за основу вероятностную модель (1.2) при ограничениях нормировки (1.31). Параметры φ_{wt} и θ_{td} для нерасщепляемых тем $t \notin R$ можно либо фиксировать, либо оптимизировать совместно с новыми параметрами. Рассмотрим вычислительно более эффективный вариант, когда они фиксируются. Фиксируемая часть вероятностной модели вычисляется один раз в начале каждой итерации (шаг 4):

$$\sigma_{dw} = \sum_{t \in T \setminus R} \varphi_{wt}\theta_{td}, \quad d \in D,\ w \in d.$$

Множество всех новых подтем обозначим через $S = \bigcup_{t \in R} S_t$.

Задача максимизации логарифма правдоподобия аналогична задаче (1.6), но оптимизируется только часть параметров $\Phi_S = (\varphi_{ws})_{W \times S}$ и $\Theta_S = (\theta_{sd})_{S \times D}$, связанных с новыми темами из S:

$$\begin{aligned} L(D; \Theta_S, \Phi_S) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln\Big(\sigma_{dw} + \sum_{s \in S} \varphi_{ws}\theta_{sd}\Big) \to \max_{\Phi_S, \Theta_S}; \\ \sum_{w \in W} \varphi_{ws} = 1, \quad s \in S; \\ \sum_{s \in S_t} \theta_{sd} = \theta_{td}, \quad t \in R,\ \ d \in D. \end{aligned}$$

Как обычно, нужно записать лагранжиан, приравнять нулю его производные по переменным φ_{ws} и θ_{sd}, из полученных уравнений исключить двойственные переменные и выразить φ_{ws} и θ_{sd} через H_{dws}:

$$H_{dws} = \frac{\varphi_{ws}\theta_{sd}}{\sigma_{dw} + \sum_{s' \in S} \varphi_{ws'}\theta_{s'd}}, \quad d \in D,\ \ w \in d,\ \ s \in S;$$

$$\varphi_{ws} = \frac{\sum_{d \in D} n_{dw} H_{dws}}{\sum_{w' \in W} \sum_{d \in D} n_{dw'} H_{dw's}}, \quad w \in W,\ \ s \in S;$$

$$\theta_{sd} = \theta_{td} \frac{\sum_{w \in d} n_{dw} H_{dws}}{\sum_{s' \in S_t} \sum_{w' \in d} n_{dw'} H_{dw's'}}, \quad d \in D,\ \ s \in S_t,\ \ t \in R;$$

или, в более компактной записи с использованием счётчиков:

$$\varphi_{ws} = \frac{\hat{n}_{ws}}{\hat{n}_s}, \qquad \hat{n}_s = \sum_{w \in W} \hat{n}_{ws}, \qquad \hat{n}_{ws} = \sum_{d \in D} n_{dw} H_{dws}. \tag{1.33}$$

$$\theta_{sd} = \theta_{td} \frac{\hat{n}_{ds}}{\hat{n}_{dt}}, \qquad \hat{n}_{dt} = \sum_{s \in S_t} \hat{n}_{ds}, \qquad \hat{n}_{ds} = \sum_{w \in d} n_{dw} H_{dws}. \tag{1.34}$$

Таким образом, формулы М-шага и Е-шага для иерархического алгоритма лишь немногим отличаются от обычного PLSA-EM. Шаги 8–19 представляют собой незначительную модификацию Алгоритма 1.3.

Регуляризация Дирихле может быть добавлена в Алгоритм 1.9 обычным образом. В случае байесовской оценки формулы шагов 17, 18 заменяются следующими:

$$\varphi_{ws} = \frac{\beta_w + \hat{n}_{ws}}{\sum\limits_{w' \in W} \left(\beta_{w'} + \hat{n}_{w's}\right)} = \frac{\beta_w + \hat{n}_{ws}}{\beta_0 + \hat{n}_s}. \tag{1.35}$$

$$\theta_{sd} = \theta_{td} \frac{\alpha_s + \hat{n}_{ds}}{\sum\limits_{s' \in S_t} \left(\alpha_{s'} + \hat{n}_{ds'}\right)} = \theta_{td} \frac{\alpha_s + \hat{n}_{ds}}{\alpha_t + \hat{n}_{dt}}. \tag{1.36}$$

Заметим, что при расщеплении темы t на множество подтем S_t расщепляются также и гиперпараметры распределения Дирихле $\mathrm{Dir}(\theta_d; \alpha)$, а их сумма α_0 не меняется. Это следует из условий нормировки (1.31) и свойства (1.14):

$$\sum_{s \in S_t} \theta_{sd} = \theta_{td} \quad \Rightarrow \quad \sum_{s \in S_t} \mathsf{E}\theta_{sd} = \mathsf{E}\theta_{td} \quad \Rightarrow \quad \sum_{s \in S_t} \alpha_s = \alpha_t.$$

Критерий остановки (шаг 20). Вопрос о том, сколько новых подтем должно быть в подмножестве S_t, пожалуй, наиболее спорный во всём процессе восстановления тематического дерева. Все подтемы одной темы должны быть равноправны. Нельзя допустить появления в S_t таких пар подтем (s, s'), что s' является подтемой s. Если это произойдёт, то наращивание множества подтем S_t должно быть остановлено, тема s' удалена, а тема s занесена в множество расщепляемых тем R для следующей итерации. Проблема в том, что в вероятностных моделях отношение «тема–подтема» формализуется неоднозначно.

Наиболее подходящей «мерой вложенности» темы s' в тему s является дивергенция Кульбака–Лейблера между распределениями $\varphi_{ws'} = p(w \,|\, s')$ и $\varphi_{ws} = p(w \,|\, s)$:

$$\mathrm{KL}_{s's} = \mathrm{KL}(\varphi_{s'} \,\|\, \varphi_s) = \sum_{w \in W_{s's}} \varphi_{ws'} \ln \frac{\varphi_{ws'}}{\varphi_{ws}}, \quad W_{s's} = \left\{w \in W : \varphi_{ws'} > 0,\ \varphi_{ws} > 0\right\}.$$

Чем меньше дивергенция, тем сильнее распределение $\varphi_{s'}$ вложено в φ_s. Дивергенция $\mathrm{KL}_{ss'}$, наоборот, характеризует степень вложенности φ_s в $\varphi_{s'}$. Равноправные темы должны иметь близкие значения обеих дивергенций.

Критерием остановки наращивания множества подтем S_t будем считать одно из следующих условий, либо их сочетание:

1) минимальная дивергенция $\min\limits_{s,s' \in S_t} \mathrm{KL}_{s's}$ ниже порога;

2) минимальная дивергенция резко уменьшилась по сравнению с предыдущей итерацией;

3) максимальная относительная разность дивергенций $\max\limits_{s,s' \in S_t} \frac{|\mathrm{KL}_{s's} - \mathrm{KL}_{ss'}|}{\mathrm{KL}_{ss'} + \mathrm{KL}_{s's}}$ выше порога;

4) создано ровно столько подтем, сколько есть у соответствующей темы во внешнем рубрикаторе;

5) создано ровно столько подтем, сколько было задано пользователем.

Список литературы

[1] *Воронцов К. В., Потапенко А. А.* Регуляризация, робастность и разреженность вероятностных тематических моделей // *Компьютерные исследования и моделирование.* — 2012. — Т. 4, № 4. — С. 693–706.

[2] *Лукашевич Н. В.* Тезаурусы в задачах информационного поиска. — Издательство МГУ имени М. В. Ломоносова, 2011.

[3] *Маннинг К. Д., Рагхаван П., Шютце Х.* Введение в информационный поиск. — Вильямс, 2011.

[4] *Павлов А. С., Добров Б. В.* Метод обнаружения массово порожденных неестественных текстов на основе анализа тематической структуры // *Вычислительные методы и программирование: новые вычислительные технологии.* — 2011. — Т. 12. — С. 58–72.

[5] *Asuncion A., Welling M., Smyth P., Teh Y. W.* On smoothing and inference for topic models // Proceedings of the International Conference on Uncertainty in Artificial Intelligence. — 2009.

[6] *Blei D. M., Ng A. Y., Jordan M. I.* Latent Dirichlet allocation // *Journal of Machine Learning Research.* — 2003. — Vol. 3. — Pp. 993–1022.

[7] *Buntine W. L.* Estimating likelihoods for topic models // 1st Asian Conference on Machine Learning: Advances in Machine Learning. — 2009. — Pp. 51–64. http://www.nicta.com.au/__data/assets/pdf_file/0019/20746/sdca-0202.pdf.

[8] *Celeux G., Chauveau D., Diebolt J.* On stochastic versions of the em algorithm: Tech. Rep. RR-2514: INRIA, 1995.

[9] *Chemudugunta C., Smyth P., Steyvers M.* Modeling general and specific aspects of documents with a probabilistic topic model // Advances in Neural Information Processing Systems. — MIT Press, 2006. — Vol. 19. — Pp. 241–248.

[10] *Cressie N., Read T. R. C.* Multinomial goodness-of-fit tests // *Journal of the Royal Statistical Society, Series B.* — 1984. — Vol. 46, no. 3. — Pp. 440–464.

[11] *Daud A., Li J., Zhou L., Muhammad F.* Knowledge discovery through directed probabilistic topic models: a survey // *Frontiers of Computer Science in China.* — 2010. — Vol. 4, no. 2. — Pp. 280–301. http://dx.doi.org/10.1007/s11704-009-0062-y.

[12] *Dempster A. P., Laird N. M., Rubin D. B.* Maximum likelihood from incomplete data via the EM algorithm // *J. of the Royal Statistical Society, Series B.* — 1977. — no. 34. — Pp. 1–38.

[13] *Eisenstein J., Ahmed A., Xing E. P.* Sparse additive generative models of text // ICML'11. — 2011. — Pp. 1041–1048.

[14] *Feng Y., Lapata M.* Topic models for image annotation and text illustration // Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. — Association for Computational Linguistics, 2010. — Pp. 831–839.

[15] *Girolami M., Kabán A.* On an equivalence between PLSI and LDA // SIGIR'03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. — 2003. — Pp. 433–434.

[16] *Grün B., Hornik K.* topicmodels: An r package for fitting topic models // *Journal of Statistical Software*. — 2011. — Vol. 40, no. 13. — Pp. 1–30.

[17] *Hofmann T.* Probabilistic latent semantic indexing // Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. — New York, NY, USA: ACM, 1999. — Pp. 50–57.

[18] *Kim S.-H., Choi H., Lee S.* Estimate-based goodness-of-fit test for large sparse multinomial distributions // *Computational Statistics and Data Analysis*. — 2009. — Vol. 53, no. 4. — Pp. 1122 – 1131.
http://www.sciencedirect.com/science/article/pii/S0167947308004817.

[19] *Krestel R., Fankhauser P., Nejdl W.* Latent dirichlet allocation for tag recommendation // Proceedings of the third ACM conference on Recommender systems. — ACM, 2009. — Pp. 61–68.

[20] *Mimno D., Blei D.* Bayesian checking for topic models // 11th Conference on Empirical Methods in Natural Language Processing. — Association for Computational Linguistics, 2011. — Pp. 227–237.

[21] *Pecina P., Schlesinger P.* Combining association measures for collocation extraction // Proceedings of the COLING/ACL on Main conference poster sessions. — Association for Computational Linguistics, 2006. — Pp. 651–658.
http://http://dl.acm.org/citation.cfm?id=1273073.1273157.

[22] *Read T., Cressie N.* Goodness-of-Fit Statistics for Discrete Mutivariate Data. — Springer, New York, 1988.

[23] *Rubin T. N., Chambers A., Smyth P., Steyvers M.* Statistical topic models for multi-label document classification // *Machine Learning*. — 2012. — Vol. 88, no. 1-2. — Pp. 157–208.

[24] *Sebastiani F.* Machine learning in automated text categorization // *ACM Computing Surveys*. — 2002. — Vol. 34, no. 1. — Pp. 1–47.

[25] *Steyvers M., Griffiths T.* Finding scientific topics // *Proceedings of the National Academy of Sciences*. — 2004. — Vol. 101, no. Suppl. 1. — Pp. 5228–5235.

[26] *Taneichi N., Sekiya Y., Imai H.* Improvements of goodness-of-fit statistics for sparse multinomials based on normalizing transformations // *Annals of the Institute of Statistical Mathematics*. — 2003. — Vol. 55. — Pp. 831–848.
http://dx.doi.org/10.1007/BF02523396.

[27] *Teh Y. W., Newman D., Welling M.* A collapsed variational bayesian inference algorithm for latent dirichlet allocation // NIPS. — 2006. — Pp. 1353–1360.

[28] TextFlow: Towards better understanding of evolving topics in text. / W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, X. Tong // *IEEE transactions on visualization and computer graphics*. — 2011. — Vol. 17, no. 12. — Pp. 2412–2421.

[29] *Vulić I., Smet W., Moens M.-F.* Cross-language information retrieval models based on latent topic models trained with document-aligned comparable corpora // *Information Retrieval*. — 2012. — Pp. 1–38.

[30] *Wallach H.* Structured Topic Models for Language: Ph.D. thesis / Newnham College, University of Cambridge. — 2008.

[31] *Wallach H., Mimno D., McCallum A.* Rethinking LDA: Why priors matter // Advances in Neural Information Processing Systems 22 / Ed. by Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, A. Culotta. — 2009. — Pp. 1973–1981.
http://books.nips.cc/papers/files/nips22/NIPS2009_0929.pdf.

[32] *Wallach H., Murray I., Salakhutdinov R., Mimno D.* Evaluation methods for topic models // 26th International Conference on Machine Learning, Montreal, Canada. — 2009. — Pp. 1105–1112.
http://www.cs.umass.edu/~mimno/papers/wallach09evaluation.pdf.

[33] *Wang Y.* Distributed Gibbs sampling of latent dirichlet allocation: The gritty details. — 2008.

[34] *xu Li X., bo Sun C., Lu P., jie Wang X., xin Zhong Y.* Simultaneous image classification and annotation based on probabilistic model // *The Journal of China Universities of Posts and Telecommunications*. — 2012. — Vol. 19, no. 2. — Pp. 107–115.

[35] *Yeh J.-h., Wu M.-l.* Recommendation based on latent topics and social network analysis // Proceedings of the 2010 Second International Conference on Computer Engineering and Applications - Volume 01. — IEEE Computer Society, 2010. — Pp. 209–213.

[36] *Yi X., Allan J.* A comparative study of utilizing topic models for information retrieval // Advances in Information Retrieval. — Springer Berlin Heidelberg, 2009. — Vol. 5478 of *Lecture Notes in Computer Science*. — Pp. 29–41.

[37] *Zavitsanos E., Paliouras G., Vouros G. A.* Non-parametric estimation of topic hierarchies from texts with hierarchical Dirichlet processes // *Journal of Machine Learning Research*. — 2011. — Vol. 12. — Pp. 2749–2775.
http://dl.acm.org/citation.cfm?id=1953048.2078193.

[38] *Zelterman D.* Goodness-of-fit tests for large sparse multinomial distributions // *Journal of the American Statistical Association*. — 1987. — Vol. 398, no. 82. — Pp. 624–629.

[39] *Zhang J., Song Y., Zhang C., Liu S.* Evolutionary hierarchical Dirichlet processes for multiple correlated time-varying corpora // Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. — 2010. — Pp. 1079–1088.

[40] *Zhang Z., Iria J., Brewster C., Ciravegna F.* A comparative evaluation of term recognition algorithms // Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC08). — 2008.
http://http://www.dcs.shef.ac.uk/~kiffer/papers/Zhang_LREC08.pdf.

[41] *Zilberstein S.* Using anytime algorithms in intelligent systems // *AI Magazine*. — 1996. — Vol. 17, no. 3. — Pp. 73–83.
http://www.aaai.org/ojs/index.php/aimagazine/article/viewArticle/1232.