Вероятностное тематическое моделирование

К. В. Воронцов

30 июля 2013 г.

Содержание

1 Вероятностное тематическое моделирование

Тематическое моделирование (topic modeling) — одно из 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 (ВТМ) 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. Во многих приложениях требуется определить также и число тем.

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

Тематические модели применяются для выявления трендов в научных публикациях или новостных потоках [32, 24], для классификации и категоризации документов [20] и изображений [16, 11], для семантического информационного поиска [31], в том числе многоязычного [25], для тегирования веб-страниц [15], для обнаружения текстового спама [4], в рекомендательных системах [30] и других приложениях. Для конкретности будем рассматривать применение ВТМ для категоризации и тематического поиска научных публикаций.

Тематические модели могут учитывать различные особенности языка и текстовых коллекций. Существуют модели, выявляющие термины как устойчивые сочетания слов, отслеживающие изменения тематики во времени или внутри отдельных документов, строящие иерархические отношения между темами, учитывающие связи между документами через авторство или ссылки, и т. д. Многочисленные разновидности вероятностных тематических моделей описаны в обзоре [9].

§1.1 Вероятностная модель коллекции 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.

Алгоритм 1.1. Вероятностная модель порождения коллекции документов.

Вход: распределения $p(w\,|\,t)$, $p(t\,|\,d)$;
Выход: выборка пар (d_i, w_i), $i = 1, \ldots, n$;
для всех $d \in D$
  задать длину n_d документа d;
  для всех $i = 1, \ldots, n_d$
    выбрать случайную тему t из распределения $p(t\,|\,d)$;
    выбрать случайный термин w из распределения $p(w\,|\,t)$;
    добавить в выборку пару (d, w), при этом тема t «забывается»;

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.

Построить *тематическую модель* коллекции документов D — значит найти распределения $p(w\,|\,t)$ для всех тем $t \in T$ и распределения $p(t\,|\,d)$ для всех документов $d \in D$. Найденные распределения используются затем для решения прикладных задач. Распределение $p(t\,|\,d)$ является удобным признаковым описанием документа в задачах информационного поиска, классификации и категоризации документов.

Гипотеза условной независимости. Будем полагать, что появление слов в документе d, относящихся к теме t, описывается общим для всей коллекции распределением $p(w\,|\,t)$ и не зависит от документа d. Это предположение, называемое *гипотезой условной независимости*, допускает три эквивалентных представления:

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

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

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

где $p(t\,|\,d)$ и $p(w\,|\,t)$ — искомые распределения. Согласно порождающей модели (1.2), коллекция D — это выборка наблюдений (d, w), генерируемых Алгоритмом 1.1. Процесс порождения последовательности слов документа показан на рис. 1.

Гипотеза разреженности. Каждый документ d и каждый термин w связан, как правило, с небольшим числом тем t. Поэтому значительная часть вероятностей $p(t\,|\,d)$ и $p(w\,|\,t)$ должна обращаться в нуль. Документы, относящиеся к большому числу тем

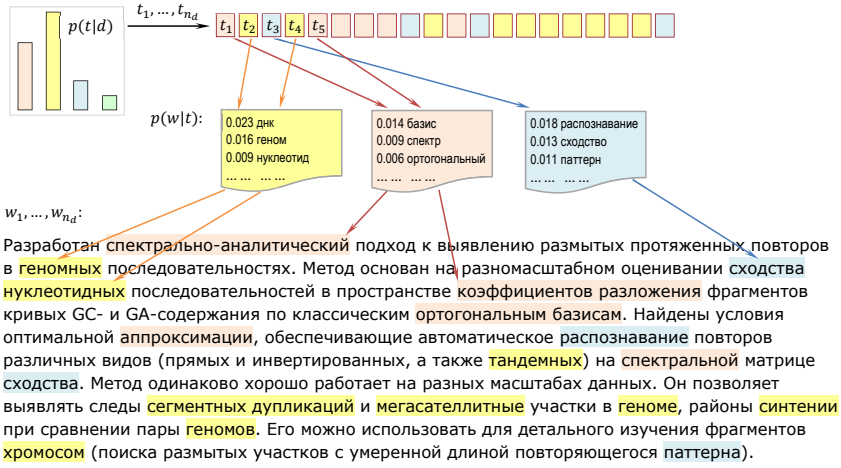


Рис. 1. Процесс порождения текстового документа вероятностной тематической моделью (1.2).

(энциклопедии, сборники статей), могут присутствовать в коллекции, но их не много. Терминов, относящихся к большому числу тем, также немного — это общеупотребительные слова (стоп-слова), бесполезные для определения тематики.

Частотные оценки условных 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 $\hat{p}$):

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

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):

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

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 $p(\cdot)$, согласно закону больших чисел. Частотная интерпретация даёт ясное понимание всех условных вероятностей, которые будут использоваться в дальнейшем.

Связь с задачами стохастического 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

$$F = \big(\hat{p}_{wd}\big)_{W\times D}, \quad \hat{p}_{wd} = \hat{p}(w\,|\,d) = n_{dw}/n_d,$$

в виде произведения $F \approx \Phi\Theta$ двух неизвестных матриц меньшего размера — *матрицы терминов тем* Φ и *матрицы тем документов* Θ:

$$\Phi = (\varphi_{wt})_{W\times T}, \quad \varphi_{wt} = p(w\,|\,t);$$
$$\Theta = (\theta_{td})_{T\times D}, \quad \theta_{td} = p(t\,|\,d).$$

Матрицы, столбцы которых неотрицательны и нормированы, следовательно, могут пониматься как дискретные распределения, называются *стохастическими*. Наиболее известное классическое представление $F \approx \Phi\Theta$ строится из $|T|$ главных компонент сингулярного разложения матрицы F и является решением задачи наименьших квадратов $\|F - \Phi\Theta\|^2 \to \min$. К сожалению, оно не подходит для тематического моделирования, так как в нём матрицы Φ, Θ ортогональны и в общем случае не являются стохастическими. Кроме того, квадратичная функция потерь плохо подходит для сравнения вероятностных распределений с «тяжёлыми хвостами».

1.1.1 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. Разработка хорошего *лемматизатора* (lemmatizer) требует составления грамматического словаря со всеми формами слов, либо аккуратной формализации правил языка со всеми исключениями, что является трудоёмким проектом. Известные лемматизаторы совершенствуются постепенно. Их недостатком является неполнота словарей, особенно по части специальной терминологии и неологизмов, которые во многих приложениях как раз и представляют наибольший интерес.

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.

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 терминами предметной области. Это отдельная довольно сложная задача, для решения которой используются тезаурусы, составленные экспертами [3], либо методы машинного обучения [19, 33], 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».

1.1.2 Метод максимума правдоподобия

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

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

где C — нормировочный множитель, 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

$$L(D;\Phi,\Theta) = \sum_{d\in D}\sum_{w\in d} n_{dw} \ln \sum_{t\in T} \varphi_{wt}\theta_{td} \;\to\; \max_{\Phi,\Theta} \tag{1.5}$$

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

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

1.1.3 Униграммная модель

Униграммная модель является простейшим примером вероятностной порождающей модели. Предполагается, что каждое слово появляется в тексте независимо от соседних слов. Модели, в которых учитываются пары, тройки, n-ки соседних слов, называются, соответственно, *биграммными*, *триграммными*, *n-граммными*. Мы рассмотрим два варианта униграммной модели.

Униграммная модель документов. Допустим, что слова каждого документа генерируются случайно и независимо из распределения $p(w|d) = \xi_{dw}$, своего для каждого документа d. Запишем задачу максимизации правдоподобия при ограничениях нормировки и неотрицательности на параметры модели ξ_{dw}:

$$\sum_{d\in D}\sum_{w\in d} n_{dw} \ln \xi_{dw} \to \max_{\xi}, \qquad \sum_{w\in W} \xi_{dw} = 1, \qquad \xi_{dw} \geqslant 0.$$

Запишем функцию Лагранжа, проигнорировав ограничения-неравенства (потом, получив решение, убедимся, что они выполнены автоматически):

$$\mathscr{L} = \sum_{d\in D}\Big(\sum_{w\in d} n_{dw} \ln \xi_{dw} - \lambda_d\Big(\sum_{w\in W} \xi_{dw} - 1\Big)\Big);$$

приравняем нулю производные по переменным ξ_{dw}:

$$\frac{\partial \mathscr{L}}{\partial \xi_{dw}} = \frac{n_{dw}}{\xi_{dw}} - \lambda_d = 0.$$

Суммируя по $w \in d$, получим значение двойственных переменных $\lambda_d = n_d$, и, подставляя их обратно в уравнение для ξ_{dw}, найдём, что искомый параметр ξ_{dw} равен частотной оценке условной вероятности слова w в документе d:

$$\xi_{dw} = \hat{p}(w|d) = n_{dw}/n_d. \tag{1.6}$$

Униграммная модель коллекции. Теперь предполодим, что слова каждого документа генерируются случайно и независимо из распределения $p(w|d) = \xi_w$, общего для всех документов коллекции. По аналогии с предыдущим случаем,

$$\sum_{d\in D}\sum_{w\in d} n_{dw} \ln \xi_{w} \to \max_{\xi}, \qquad \sum_{w\in W} \xi_{w} = 1, \qquad \xi_{w} \geqslant 0.$$

$$\mathscr{L} = \sum_{d\in D}\sum_{w\in d} n_{dw} \ln \xi_{w} - \lambda\Big(\sum_{w\in W} \xi_{w} - 1\Big);$$

$$\frac{\partial \mathscr{L}}{\partial \xi_{w}} = \frac{n_{w}}{\xi_{w}} - \lambda = 0;$$

откуда следует, что $\lambda = n$, и искомый параметр ξ_w равен частотной оценке вероятности слова w во всей коллекции:

$$\xi_{w} = \hat{p}(w) = n_{w}/n. \tag{1.7}$$

Обе униграммные модели имеют простые, интуитивно очевидные решения (1.6) и (1.7), но не являются тематическими. Тематическая модель (1.2) занимает между ними промежуточное положение. Набор её параметров богаче униграммной модели коллекции, но беднее униграммной модели документов.

§1.2 Вероятностный латентный семантический анализ

Вероятностный латентный семантический анализ PLSA (probabilistic latent semantic analysis) был предложен Томасом Хофманном в [14]. Задача максимизации правдоподобия (1.5) для вероятностной модели (1.2) не имеет простого аналитического решения и решается численно с помощью EM-алгоритма.

1.2.1 EM-алгоритм

EM-алгоритм — это итерационный процесс, в котором каждая итерация состоит из двух шагов — E (expectation) и M (maximization) [10]. 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 $p(t \mid d, w)$ всех тем $t \in T$ для каждого термина $w \in d$ в каждом документе d:

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

На M-шаге, наоборот, по условным вероятностям тем H_{dwt} вычисляется новое приближение параметров φ_{wt}, θ_{td}. Это легко сделать, если заметить, что величина

$$\hat{n}_{dwt} = n_{dw}p(t \mid d, w) = n_{dw}H_{dwt} \tag{1.9}$$

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}:

$$\varphi_{wt} = \frac{\hat{n}_{wt}}{\hat{n}_t}, \qquad \hat{n}_t = \sum_{w \in W} \hat{n}_{wt}, \qquad \hat{n}_{wt} = \sum_{d \in D} n_{dw}H_{dwt}. \tag{1.10}$$

$$\theta_{td} = \frac{\hat{n}_{dt}}{\hat{n}_d}, \qquad \hat{n}_d = \sum_{t \in T} \hat{n}_{dt}, \qquad \hat{n}_{dt} = \sum_{w \in d} n_{dw}H_{dwt}. \tag{1.11}$$

Эти простые, но не вполне строгие рассуждения поясняют суть EM-алгоритма. Покажем теперь, что оценки (1.10)–(1.11) действительно являются решением задачи максимизации правдоподобия (1.5) при фиксированных H_{dwt}. Запишем лагранжиан задачи (1.5) при ограничениях нормировки, проигнорировав ограничения неотрицательности (позже убедимся, что решение действительно неотрицательно):

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

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

$$\lambda_t = \sum_{d \in D} n_{dw} \frac{\theta_{td}}{p(w \mid d)}. \tag{1.12}$$

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

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

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

Вход: коллекция документов D, число тем $|T|$, начальные приближения Θ, 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 сойдутся;

Снова домножим обе части (1.12) 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

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

Обозначив числитель через $\hat{n}_{wt}$, получим (1.10). Проделав аналогичные действия с производной лагранжиана по θ_{td}, получим (1.11).

Заметим, что 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.

Эффективность EM-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}).

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. При этом 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.

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

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

Вход: коллекция документов D, число тем $|T|$, начальные приближения Θ, 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 сойдутся;

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 [10].

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 достаточно больших коллекциях частые обновления повышают скорость сходимости [2]. В Алгоритме 1.3 выбор условия обновления на шаге 9 оставлен на усмотрение разработчика.

На первой итерации проход коллекции делается без обновления счётчиков, чтобы в них накопилась информация о всей коллекции. 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 на предыдущей итерации. Таким образом, счётчики $\hat{n}_{wt}$ и $\hat{n}_{dt}$ всегда содержат результат одного последнего прохода коллекции.

Необходимость хранения трёхмерной матрицы n_{dwt} делает Алгоритм 1.3 неприменимым к большим коллекциям. Этот недостаток устраняется путём реорганизации итераций, либо применением сэмплирования. Далее рассматриваются оба способа.

1.2.3 Онлайновый EM-алгоритм

На больших коллекциях Алгоритмы 1.2 и 1.3 могут сходиться очень медленно. Причина в том, что за однократный проход по всем документам коллекции оценки распределений терминов в темах $\varphi_{wt} = \hat{n}_{wt}/\hat{n}_t$ уточняются огромное число раз и успевают сойтись, в то же время распределения тем в документах θ_d проходят лишь одну итерацию. На начальных итерациях, пока распределения θ_d не сошлись, вычислительный ресурс тратится впустую на достижение сходимости φ_t к прибли-

Алгоритм 1.4. PLSA-BatchEM: пакетный EM-алгоритм для модели PLSA.

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

инициализировать φ_{wt} для всех $w \in W$, $t \in T$;
повторять
  $\hat{n}_{wt} := 0$; $\hat{n}_t := 0$ для всех $w \in W$, $t \in T$;
  для всех $d \in D$
    инициализировать θ_{td} для всех $t \in T$;
    повторять
      $Z_w := \sum_{t \in T} \varphi_{wt}\theta_{td}$ для всех $w \in d$;
      $\theta_{td} := \frac{1}{n_d} \sum_{w \in d} n_{dw}\varphi_{wt}\theta_{td}/Z_w$ для всех $t \in T$;
    пока θ_d не сойдётся;
    увеличить $\hat{n}_{wt}$, $\hat{n}_t$ на $n_{dw}\varphi_{wt}\theta_{td}/Z_w$ для всех $w \in d$, $t \in T$;
  $\varphi_{wt} := \hat{n}_{wt}/\hat{n}_t$ для всех $w \in W$, $t \in T$;
пока Φ не сойдутся;

жениям, далёким от оптимальных. Суть этой проблемы в том, что параметры θ_{td} привязаны к отдельным документам d, а параметры φ_{wt} — ко всей коллекции.

Проблема решается реорганизацией шагов итерационного процесса. Проход каждого документа $d \in D$ производится несколько раз подряд. На каждом проходе документа выполняется E-шаг и обновляется распределение θ_d. Обновление распределений φ_t производится после каждого прохода коллекции. В результате распределения φ_t и θ_d сходятся более согласованно. Кроме того, реорганизация позволяет полностью отказаться от хранения трёхмерных массивов H_{dwt} или n_{dwt}.

Алгоритм 1.4 назван *пакетным* (batch algorithm), так как он может обрабатывать коллекцию по частям. Развитие этой идеи приводит к онлайновому алгоритму [13], одному из самых быстрых в тематическом моделировании. Он реализован в библиотеке онлайновых алгоритмов Vowpal Wabbit Джона Лэнгфорда.

Онлайновый алгоритм. В машинном обучении *онлайновыми* принято называть алгоритмы, способные адекватно настраивать параметры модели за один проход по выборке. Онлайновые алгоритмы используются для обработки потоковых данных. Во многих приложениях тематического моделирования коллекция документов огромна и пополняется динамически. При появлении нового документа d требуется вычислить распределение $\theta_{td} = p(t \mid d)$ и уточнить распределения $\varphi_{wt} = p(w \mid t)$ для всех терминов документа d. Чем больше коллекция, тем меньше новые документы влияют на Φ, и тем меньше итераций требуется для сходимости θ_d.

Онлайновый Алгоритм 1.5 является модицификацией пакетного Алгоритма 1.4. Теперь вся коллекция разбивается на пакеты документов $D_1, D_2, \ldots, D_j, \ldots$ Способ разбиения остаётся на усмотрение разработчика, в частности, пакеты могут пересекаться либо не пересекаться, просматриваться по одному разу либо многократно, выбираться случайно, по времени поступления, по времени публикации, и т. д. Размер первого пакета $|D_1|$ должен быть достаточным для получения хорошего начального

Алгоритм 1.5. PLSA-OEM: онлайновый EM-алгоритм для модели PLSA.

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

инициализировать φ_{wt} для всех $w \in W,\ t \in T$;
$\hat{n}_{wt} := 0,\ \hat{n}_t := 0$ для всех $w \in W,\ t \in T$;
для всех пакетов $D_j,\ j = 1, \ldots, J$
  повторять
    $\tilde{n}_{wt} := 0,\ \tilde{n}_t := 0$ для всех $w \in W,\ t \in T$;
    для всех $d \in D_j$
      инициализировать θ_{td} для всех $t \in T$;
      повторять
        $Z_w := \sum_{t \in T} \varphi_{wt}\theta_{td}$ для всех $w \in d$;
        $\theta_{td} := \frac{1}{n_d} \sum_{w \in d} n_{dw}\varphi_{wt}\theta_{td}/Z_w$ для всех $t \in T$;
      пока θ_d не сойдётся;
      увеличить $\tilde{n}_{wt}$, $\tilde{n}_t$ на $n_{dw}\varphi_{wt}\theta_{td}/Z_w$ для всех $w \in d,\ t \in T$;
    $\varphi_{wt} := \dfrac{\rho_j\hat{n}_{wt} + \tilde{n}_{wt}}{\rho_j\hat{n}_t + \tilde{n}_t}$ для всех $w \in W,\ t \in T$ таких, что $\tilde{n}_{wt} > 0$;
  пока Φ не сойдутся;
  $\hat{n}_{wt} := \rho_j\hat{n}_{wt} + \tilde{n}_{wt}$ для всех $w \in W,\ t \in T$;
  $\hat{n}_t := \rho_j\hat{n}_t + \tilde{n}_t$ для всех $t \in T$;

приближения Φ. Обработка каждого пакета производится пакетным Алгоритмом 1.4 при фиксированных φ_{wt}. Затем счётчики $\tilde{n}_{wt}$, вычисленные по обработанному пакету документов, складываются со счётчиками $\hat{n}_{wt}$, вычисленными по всем предыдущим пакетам, и происходит обновление φ_{wt}.

Если делается много проходов по коллекции или если значимость пакетов убывает по мере поступления новых, то старые счётчики домножаются на параметр $\rho_j \in (0, 1]$, задающий скорость забывания старых оценок: $\hat{n}_{wt} := \rho_j\hat{n}_{wt} + \tilde{n}_{wt}$. Таким образом, при поступлении каждого нового пакета документов D_j частоты слов во всех старых документах уменьшаются в ρ_j^{-1} раз. При $\rho_j = 1$ забывания нет и φ_{wt} вычисляются как обычные частотные оценки условных вероятностей.

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 [2].

Проблема разреживания условного распределения $p(t\,|\,d,w)$ адекватно решается с помощью стохастического EM-алгоритма (stochastic EM-algorithm, SEM) [7]. На M-шаге используется не само распределение $p(t\,|\,d,w)$, а эмпирическое распределение, построенное по искусственной выборке тем t_{dwi}, $i=1,\ldots,s$, сэмплированной для каждой пары (d,w) из распределения $p(t\,|\,d,w)$:

$$\hat{p}(t\,|\,d,w) = \frac{1}{s}\sum_{i=1}^{s}\big[t_{dwi}=t\big]. \tag{1.13}$$

Это позволяет добиться разреженности, и тем самым упростить задачу M-шага, сохранив свойства несмещённости оценок и сходимости EM-алгоритма. Размер сэмплируемой выборки s является параметром метода.

Для трансформации Алгоритма 1.3 в стохастический обобщённый EM-алгоритм (PLSA-SEM) достаточно сделать три изменения:

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

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

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

При $s=n_{dw}$ стохастический EM-алгоритм соответствует *сэмплированию Гиббса* [28] — одному из основных методов обучения вероятностных тематических моделей. В [1, 2] предложено *экономное сэмплирование*, когда s уменьшается до 3–5 тем, что приводит к большему разреживанию и экономии вычислительных ресурсов без существенной потери качества тематической модели.

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 $d\in D$, $w\in W$, $t\in T$.

Инициализация с частичным обучением применяется в случаях, когда темы известны заранее и имеются дополнительные данные о привязке некоторых документов или терминов к темам. Учёт этих данных улучшает интерпретируемость тем.

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:

$$\theta^0_{td} = \frac{1}{|T_d|}\big[t\in T_d\big]. \tag{1.14}$$

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

$$\varphi^0_{wt} = \frac{1}{|W_t|}\big[w\in W_t\big]. \tag{1.15}$$

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

$$\varphi^0_{wt} = \frac{\sum_{d\in D_t} n_{dw}}{\sum_{d\in D_t} n_d}.$$

Если нет никакой априорной информации о связи документов с темами, то последнюю формулу можно применить к случайным подмножествам документов D_t. В [12] предлагается брать один случайный документ.

Инициализация Θ по Φ. Если для всех тем известны начальные приближения φ_{wt}^0, то первая итерация EM-алгоритма при равномерном распределении $\theta_{td}^0 = 1/|T|$ даёт ещё одну интуитивно очевидную формулу инициализации:

$$\theta_{td} = \frac{1}{n_d}\sum_{w\in d} n_{dw} H_{dwt} = \sum_{w\in d}\frac{n_{dw}}{n_d}\frac{\varphi_{wt}}{\sum_s \varphi_{ws}} = \sum_{w\in d}\hat{p}(w\,|\,d)\,\hat{p}(t\,|\,w). \tag{1.16}$$

Здесь распределение тем в документе d оценивается путём усреднения распределений тем $p(t\,|\,w)$ по словам документа d, вычисленных по формуле Байеса.

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

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

§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 либо *отбор признаков*, приводящий к уменьшению числа параметров, либо *регуляризация* — наложение дополнительных ограничений на параметры. В частности, *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:

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

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

где $\Gamma(z)$ — гамма-функция.

Некоторые свойства распределения Дирихле. Математическое ожидание и дисперсия t-й координаты вектора θ_d равны, соответственно,

$$\mathsf{E}\theta_{td} = \int \theta_{td}\,\mathrm{Dir}(\theta_d;\alpha)\,d\theta_d = \frac{\alpha_t}{\alpha_0}, \qquad \mathsf{D}\theta_{td} = \frac{\alpha_t(\alpha_0-\alpha_t)}{\alpha_0^2(\alpha_0+1)}. \tag{1.17}$$

Векторный параметр α определяет степень разреженности векторов θ_d, порождаемых распределением $\mathrm{Dir}(\theta;\alpha)$. Если $\alpha_t = 1$ для всех t, то распределение Дирихле переходит в равномерное. Чем больше α_0, тем меньше дисперсия, и 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*.

Обоснования. Есть несколько доводов в пользу распределения Дирихле как байесовского регуляризатора вероятностных тематических моделей.

Во-первых, это достаточно широкое параметрическое семейство распределений на единичном симплексе, которое описывает как разреженные, так и сконцентрированные дискретные распределения.

Во-вторых, модель LDA хорошо подходит для описания кластерных структур. Чем меньше значения гиперпараметров α и β, тем сильнее разрежено распределение Дирихле, и тем дальше отстоят друг от друга порождаемые им векторы. Чем меньше α_0, тем сильнее различаются документы θ_d. Чем меньше β_0, тем сильнее различаются темы φ_t. Векторы $\varphi_t = p(w \mid t)$ в пространстве терминов $\mathbb{R}^{|W|}$ представляют центры тематических кластеров. Элементами кластеров являются векторы документов с эмпирическими распределениями $\hat{p}(w \mid d, t)$. Чем меньше гиперпараметры β, тем больше межкластерные расстояния по сравнению с внутрикластерными. Таким образом, гиперпараметры позволяют моделировать тематические кластерные структуры различной степени выраженности.

В-третьих, распределение Дирихле является сопряжённым к мультиномиальному, что упрощает вывод апостериорных оценок вероятностей θ_{td} и φ_{wt}. Именно математическое удобство распределения Дирихле в значительной степени определяет его популярность в тематическом моделировании.

Недостатки. Основной недостаток распределения Дирихле — отсутствие убедительных лингвистических обоснований. Предположение, что все распределения θ_d, $d \in D$ порождаются распределением Дирихле, да ещё и одним и тем же, кажется весьма произвольным. То же можно сказать и о порождении множества распределений φ_t для всех тем $t \in T$. Второй недостаток заключается в том, что параметры θ_{td} и φ_{wt} не могут обращаться в нуль, что противоречит гипотезе разреженности.

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 (1.17) по апостериорному распределению:

$$p(t|d,X_d,\alpha) = \int p(t|d)p(\theta_d|X_d,\alpha)\,d\theta_d = \int \theta_{td}\,\mathrm{Dir}(\theta_d,\alpha')\,d\theta_d = \frac{n_{td}+\alpha_t}{n_d+\alpha_0}. \tag{1.18}$$

Заменив величину n_{td} её оценкой $\hat{n}_{td}$, получим сглаженную байесовскую оценку параметра θ_{td} для EM-алгоритма, обобщающую (1.11):

$$\theta_{td} = \frac{\hat{n}_{dt}+\alpha_t}{\hat{n}_d+\alpha_0}. \tag{1.19}$$

Аналогично выводится сглаженная байесовская оценка и для φ_{wt}:

$$\varphi_{wt} = \frac{\hat{n}_{wt}+\beta_w}{\hat{n}_t+\beta_0}. \tag{1.20}$$

Замена в обобщённом EM-алгоритме частотных оценок условных вероятностей (1.10) и (1.11) сглаженными оценками (1.19) и (1.20) трансформирует PLSA в LDA. Более строгое обоснование EM-подобных алгоритмов приводится в [22, 28] для метода сэмплирования Гиббса и в [23] для метода вариационной байесовской аппроксимации.

В [5] показано, что эти и другие известные алгоритмы обучения LDA являются вариантами EM-алгоритма и отличаются, главным образом, формулой сглаживания частотных оценок вероятностей. Оптимизация гиперпараметров α и β, предложенная в [26, 27], 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.

1.3.2 Сэмплирование Гиббса

Сэмплирование Гиббса (Gibbs sampling, GS) применяется для решения задач статистического оценивания, когда вычисление или хранение функции распределении слишком ресурсоёмко, в то же время, генерация случайной выборки из этого распределения не вызывает затруднений. Тогда вместо исходного распределения используется его несмещённая эмпирическая оценка по выборке, сэмплированной из данного распределения.

Применение GS к тематической модели LDA предложено в [22], см. Алгоритм 1.6. Строгие доказательства приводятся в отчёте [28]. Однако гораздо проще понимать LDA-GS как специальный случай стохастического EM-алгоритма PLSA-SEM, когда для каждой пары (d,w) сэмплируется ровно $s = n_{dw}$ тем, а параметры φ_{wt}, θ_{td} обновляются после каждого вхождения термина w в документ d.

Алгоритм 1.6. LDA-GS: сэмплирование Гиббса для тематической модели LDA.

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

1 обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ для всех $d \in D,\ w \in W,\ t \in T$;
2 **повторять**
3 **для всех** $d \in D,\ w \in d,\ i = 1, \ldots, n_{dw}$
4 **если** не первая итерация **то**
5 $t := t_{dwi}$; уменьшить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на 1;
6 сэмплировать тему t_{dwi} из $p(t \mid d, w) \propto \dfrac{\hat{n}_{wt} + \beta_w}{\hat{n}_t + \beta_0} \dfrac{\hat{n}_{dt} + \alpha_t}{n_d + \alpha_0}$;
7 $t := t_{dwi}$; увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_t$ на 1;
8 **пока** Θ и Φ не сойдутся;

Есть ещё одно чисто техническое отличие. На шаге 5 счётчики уменьшаются на единицу. Тем самым i-е вхождение термина w в документ d не учитывается в оценке распределения $p(t \mid d, w)$, из которого сэмплируется тема t_{dwi}. Из теории следует, что эта деталь исключительно важна [28]. Однако эксперименты [1, 2] убеждают, что она практически не влияет на качество модели. Счётчики можно одновременно уменьшать для старой темы и увеличивать для новой, как в Алгоритме 1.3.

Единственное важное отличие LDA-GS от PLSA-SEM — это переход к сглаженным оценкам условных вероятностей (1.19) и (1.20). В экспериментах сэмплирование Гиббса действительно плохо работает без сглаживания.

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

1.3.3 Действительно ли сглаживание необходимо?

Согласно экспериментам [6], качество модели LDA существенно превосходит PLSA. По аналогии с задачами классификации и регрессии отсюда был сделан стандартный вывод, что модель PLSA имеет слишком много параметров θ_{td}, φ_{wt}, что и приводит к переобучению. Байесовская регуляризация накладывает ограничения на параметры, следовательно, должна сокращать эффективную размерность и уменьшать переобучение. Более тщательное сравнение PLSA и LDA показывает, что регуляризация Дирихле в тематических моделях играет совсем другую роль.

Регуляризация Дирихле приводит к сглаживанию частотных оценок условных вероятностей (1.19)–(1.20), что является единственным принципиальным отличием LDA от PLSA. Оптимальные значения гиперпараметров α_t и β_w обычно близки к нулю, как показывают эксперименты на реальных данных [27]. Оценки параметров φ_{wt} и θ_{td} в PLSA и LDA заметно отличаются только для терминов, крайне редких в теме, и тем, крайне редких в документе. Редкие темы и термины не несут статистически значимой информации о тематике коллекции. Скорее, их надо было бы проигнорировать как шум, но вместо этого LDA, наоборот, повышает оценку их вероятности.

Если из контрольных документов убрать небольшое число наиболее редких терминов, то качество PLSA и LDA практически совпадает [1, 2]. Исследования [18, 29, 17] также подтверждают, что для больших коллекций нет существенных различий в качестве моделей PLSA и LDA. Значимые отличия контрольной перплексии PLSA и LDA в ранних экспериментах [6] могут быть объяснены тем, что для них использовались существенно различные реализации алгоритмов обучения. В экспериментах [1, 2] для обучения моделей PLSA и LDA использовался один и тот же алгоритм, отличавшийся только сглаженными оценками в LDA.

Таким образом, роль регуляризации в LDA оказывается весьма скромной. Это не сокращение размерности и не уменьшение переобучения, а всего лишь более точное оценивание редких терминов и тем, которые можно считать статистически незначимыми. Кроме того, сглаживание необходимо в алгоритме сэмплирования Гиббса, чтобы все темы из распределения $p(t \mid d, w)$ имели шансы реализоваться; однако это требование скорее техническое и связано с конкретным методом.

§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 слова, 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.

Фон — это слова, имеющие значимые вероятности во многих темах, в частности, стоп-слова, не отброшенные на стадии предварительной обработки.

При построении разреженных тематических моделей с большим числом нулевых φ_{wt} и θ_{td} может оказаться, что $p(w \mid d) = 0$. В этом случае объяснить появление слова w иначе как шумовой или фоновой компонентой вообще невозможно.

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 [8]:

$$p(w \mid d) = \frac{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}{1 + \gamma + \varepsilon}; \qquad Z_{dw} = \sum_{t \in T} \varphi_{wt}\theta_{td}. \tag{1.21}$$

где *шумовая компонента* $\pi_{dw} \equiv p_{\text{ш}}(w \mid d)$ — неизвестное распределение терминов в документе d, *фоновая компонента* $\pi_w \equiv p_{\Phi}(w)$ — неизвестное распределение терминов во всей коллекции, γ и ε — неотрицательные параметры, задающие априорные вероятности тематической, шумовой и фоновой компонент: $\frac{1}{1+\gamma+\varepsilon}$, $\frac{\gamma}{1+\gamma+\varepsilon}$, $\frac{\varepsilon}{1+\gamma+\varepsilon}$.

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

$$L(D; \Phi, \Theta, \Pi) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln \frac{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}{1 + \gamma + \varepsilon} \;\to\; \max_{\Phi, \Theta, \Pi}, \tag{1.22}$$

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

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

Задача максимизации правдоподобия (1.5) для модели (1.21) решена в [1].

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

$$H_{dwt} = \frac{\varphi_{wt}\theta_{td}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}, \quad t \in T, \tag{1.23}$$

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

$$H_{dw} = \frac{\gamma\pi_{dw}}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}; \qquad H'_{dw} = \frac{\varepsilon\pi_w}{Z_{dw} + \gamma\pi_{dw} + \varepsilon\pi_w}. \tag{1.24}$$

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 H_{dwt}, H_{dw}, H'_{dw}.

M-шаг. Переменные θ_{td} и φ_{wt} вычисляются по прежним формулам (1.10) и (1.11). Формулы для переменных π_{dw}, π_w имеют естественную интерпретацию частотных оценок условных вероятностей шума и фона:

$$\pi_{dw} = \frac{\nu_{dw}}{\nu_d}, \qquad \nu_{dw} = n_{dw}H_{dw}, \qquad \nu_d = \sum_{w \in d} \nu_{dw},$$

$$\pi_w = \frac{\nu'_w}{\nu'}, \qquad \nu'_w = \sum_{d \in D} n_{dw}H'_{dw}, \qquad \nu' = \sum_{w \in W} \nu'_w,$$

где ν_d и ν' — оценки числа шумовых слов в документе d и фоновых слов в коллекции.

Неотрицательность π_{dw}, π_w гарантируется, если их начальные приближения неотрицательны. Если в начальном приближении значение π_{dw} или π_w не равно нулю, то оно так и останется ненулевым. Таким образом, переменные π_{dw} или π_w имеют ту же проблему разреженности, что и переменные φ_{wt} и θ_{td}.

Эвристика *сглаживания* вводится заменой частотных оценок (1.10)–(1.11) параметров φ_{wt} и θ_{td} байесовскими оценками (1.19)–(1.20).

Эвристика *сэмплирования* вводится заменой дискретного распределения $\tilde{H}_{dw} = \big(H_{dwt}, t \in T,\ H_{dw},\ H'_{dw}\big)$ его эмпирической оценкой, аналогичной (1.13).

1.4.2 Упрощённая робастная модель

Недостатком предыдущей модели является необходимость подбирать параметры γ, ε и хранить параметры π_{dw}, число которых сопоставимо с размером коллекции. Рассмотрим упрощённую робастную модель, которая вообще не требует дополнительных затрат памяти или времени. Фоновая компонента в ней отсутствует, а шумовая компонента π_{dw} включается только когда $Z_{dw} = 0$, то есть когда термин w в документе d оказывается нетематическим (например, вследствие разреживания):

$$p(w \,|\, d) = \nu_d Z_{dw} + \big[Z_{dw} = 0\big]\pi_{dw}, \tag{1.25}$$

где параметр ν_d определяется из условия нормировки $\sum_{w \in W} p(w \mid d) = 1$.

Максимизация правдоподобия (1.5) снова приводит к частотным оценкам условных вероятностей (1.10)–(1.11), но теперь H_{dwt} и $\hat{n}_{dwt}$ оцениваются только по тематическим терминам:

$$\hat{n}_{dwt} = \big[Z_{dw} > 0\big] n_{dw} H_{dwt}.$$

Оптимальное значение π_{dw} достаточно определять только для тех (d, w), при которых $Z_{dw} = 0$. Оно также выражается аналитически и совпадает с *униграммной оценкой* условной вероятности $p(w \mid d)$:

$$\pi_{dw} = n_{dw} / n_d.$$

Нормировочный множитель ν_d равен доле тематических терминов в документе:

$$\nu_d = \sum_{w \in W} \big[Z_{dw} > 0\big] \pi_{dw} = \frac{1}{n_d} \sum_{w \in d} \big[Z_{dw} > 0\big] n_{dw}.$$

Значения параметров π_{dw} и ν_d не нужны для вычисления тематической компоненты модели — матриц Φ и Θ, но могут понадобиться при вычислении функционалов качества модели, непосредственно зависящих от $p(w \mid d)$.

1.4.3 Принудительное разреживание

Согласно *гипотезе разреженности*, каждый документ d и каждый термин w связан с небольшим числом тем t, поэтому значительная часть вероятностей θ_{td} и φ_{wt} должна обращаться в нуль. Однако описанные выше алгоритмы обучения PLSA и LDA не определяют, какие именно из этих значений следует обратить в нуль.

Алгоритмы PLSA не оптимизирует структуру разреженности и требует задавать её через начальные приближения. Отдельные значения φ_{wt} и θ_{td} могут в ходе итераций сами собой приближаться к нулю, но, как правило, их доля не превышает 50%, что недостаточно для получения выигрыша в производительности.

Модель LDA также не является разреженной — априорные распределения Дирихле запрещают вероятностям φ_{wt} и θ_{td} и гиперпараметрам β_w и α_t принимать нулевые значения. При стремлении гиперпараметров к нулю распределения Дирихле порождают векторы φ_t и θ_d, компоненты которых стремятся к нулю, но никогда не обращаются в нуль. Сглаженные оценки (1.20)–(1.19), используемые в LDA, менее разрежены, чем несмещённые частотные оценки (1.10)–(1.11), используемые в PLSA.

В [2] исследуются различные стратегии *принудительного разреживания*, когда в конце каждой итерации (полного прохода всей коллекции D) обнуляется заданная доля r наименьших значений φ_{wt} и θ_{td}, так, чтобы сумма обнуляемых значений не превышала заданного порога R_φ или R_θ для распределений φ_t или θ_d соответственно. Разреживания включаются, начиная с итерации i_0, чтобы в распределениях правильно выделились малые вероятности, и делаются не на каждой итерации, чтобы модель успевала восстановить адекватность. Разреживание не совместимо со сглаживанием и может применяться только к PLSA.

В экспериментах на реальных коллекциях удавалось обнулить свыше 99% элементов матрицы Φ и около 95% элементов матрицы Θ без потери качества модели.

§1.5 Критерии 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.

Критерии качества тематических моделей делятся на две основные группы: внутренние и внешние. Внутренние критерии оценивают качество модели по той же коллекции документов, по которой эта модель строилась. Внешние критерии используют дополнительную информацию и показывают, насколько хорошо тематическая модель справляется с решением прикладной задачи классификации, категоризации или поиска текстовых документов.

1.5.1 Перплексия

Наиболее распространён внутренний критерий *перплексии* (perplexity), используемый для оценивания моделей языка в компьютерной лингвистике. Это мера несоответствия или «удивлённости» модели $p(w\,|\,d)$ терминам w, наблюдаемым в документах d коллекции D, определяемая через логарифм правдоподобия (1.5):

$$\mathscr{P}(D;p) = \exp\Big(-\frac{1}{n}L(D;\Phi,\Theta)\Big) = \exp\bigg(-\frac{1}{n}\sum_{d\in D}\sum_{w\in d} n_{dw}\ln p(w\,|\,d)\bigg). \tag{1.26}$$

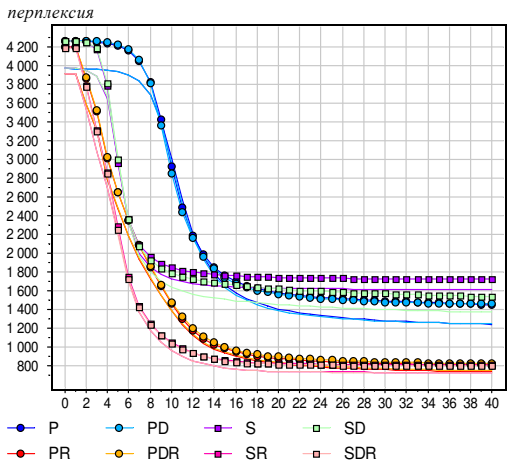
Чем меньше эта величина, тем лучше модель p предсказывает появление терминов w в документах d коллекции D.

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 лучше модель p описывает генерирующее распределение, и чем сильнее генерирующее распределение p отличается от равномерного, тем меньше перплексия. В нашем случае в (1.26) используются условные вероятности терминов $p(w\,|\,d)$, и интерпретация немного другая: если каждый документ генерируется из V равновероятных терминов (возможно, различных в разных документах), то перплексия сходится к V.

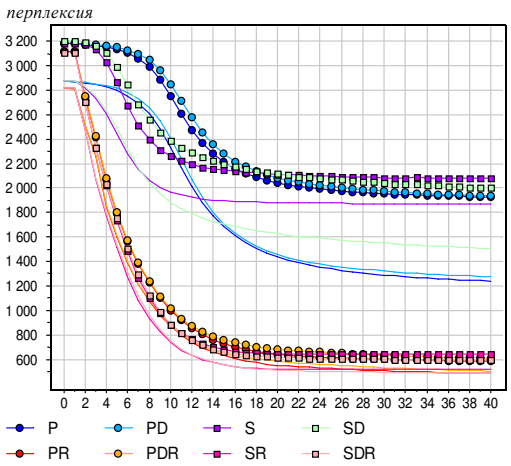
Перплексия контрольной выборки. Обозначим через $p_D(w\,|\,d)$ модель, построенную по обучающей коллекции документов D. Перплексия обучающей выборки $\mathscr{P}(D;p_D)$ является оптимистично смещённой (заниженной) характеристикой качества модели из-за эффекта переобучения. Обобщающую способность модели принято оценивать *перплексией контрольной выборки* (hold-out perplexity) $\mathscr{P}(D';p_D)$. 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 [5].

1.5.2 Эксперименты на реальных данных

Алгоритмы тематического моделирования принято сравнивать в экспериментах на общедоступных коллекциях документов. В [2] используются две коллекции,



RuDis



NIPS

Рис. 2. Зависимость контрольной перплексии от числа итераций для всевозможных сочетаний эвристик: D — сглаживание Дирихле ($\alpha_t = 0.5,\ \beta_w = 0.01$); R — робастность ($\gamma = 0.3,\ \varepsilon = 0.01$); S — сэмплирование ($s = n_{dw}$), P — пропорциональное распределение (1.9); $|T| = 100$. Тонкие кривые без точек — перплексия обучающей выборки.

доступные на странице «Коллекции документов для тематического моделирования» вики-ресурса `www.MachineLearning.ru`.

Коллекция RuDis содержит $|D| = 2000$ 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' состоит из 200 авторефератов. Предварительно производилась лемматизация и отбрасывались стоп-слова.

Коллекция NIPS содержит $|D| = 1566$ текстов статей научной конференции NIPS (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' состоит из 174 документов. Предварительно производился стемминг и отбрасывались стоп-слова.

На рис. 2 показан результат сравнения восьми алгоритмов, образуемых всевозможными комбинациями эвристик сглаживания, робастности и сэмплирования. Сравнение позволяет сделать следующие выводы:

1) для обеих задач робастные алгоритмы существенно превосходят неробастные и гораздо меньше переобучаются;

2) эвристика сглаживания для робастных алгоритмов оказывается избыточной.

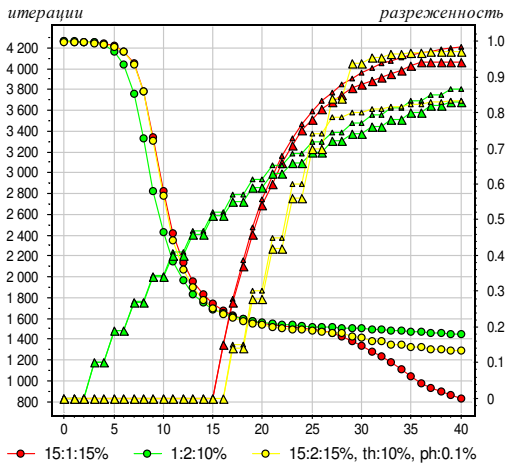
3) сэмплирование (1.13) немного хуже пропорционального распределения (1.9);

4) сэмплирование без сглаживания может приводить к увеличению перплексии.

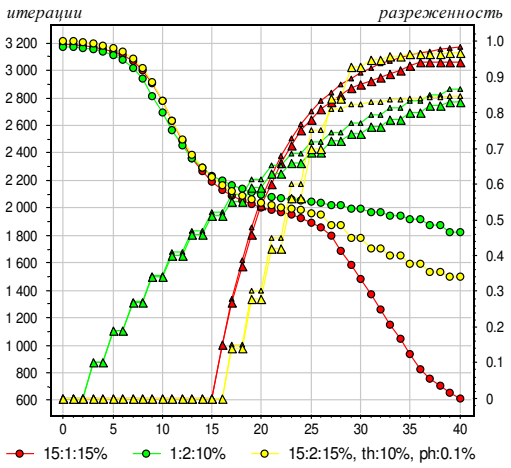
5) обучающая и контрольная перплексия приводят к одинаковым качественным выводам; нет особой необходимости вычислять контрольную перплексию.

На рис. 3 показаны результаты экспериментов с разреживанием. Наименьшая перплексия при одновременно наибольшей разреженности матрицы Φ достигается при совмещении упрощённой робастной модели, стохастического EM-алгоритма и принудительного разреживания.

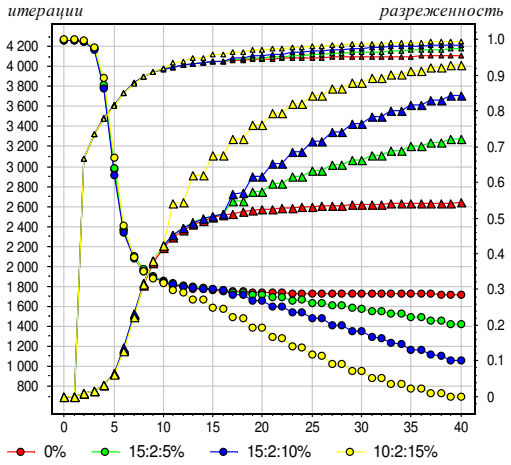
В робастных алгоритмах с шумом и фоном разреживание почти не влияет на перплексию и позволяет достигать сопоставимой разреженности, рис. 4.



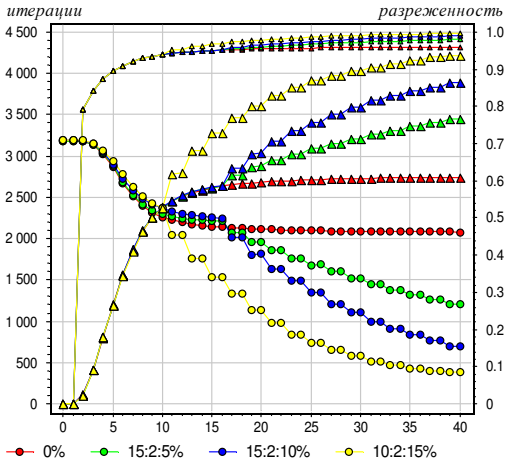
RuDis, агрессивное разреживание



NIPS, агрессивное разреживание



RuDis, SEM, через 2 итерации



NIPS, SEM, через 2 итерации

Рис. 3. Зависимость перплексии (○) и разреженности матриц Φ (△) и Θ (△) от числа итераций для рационального и стохастического EM-алгоритма при различных параметрах разреживания, обозначаемых $i_0{:}\delta{:}r$, th:R_θ, ph:R_φ. Число тем $|T| = 100$.

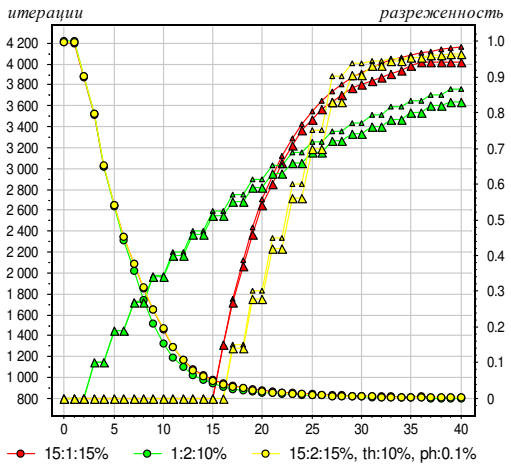
1.5.3 Критерии качества классификации документов

Если тематическая модель строится с целью классификации или категоризации документов, то качество модели естественно оценивать числом ошибок классификации документов, представленных $|T|$-мерными векторами тем $\theta_d = \big(p(t \mid d)\big)_{t \in T}$. Пусть Y — множество классов, c и a — индикаторные функции вида $Y \times D \to \{0, 1\}$,

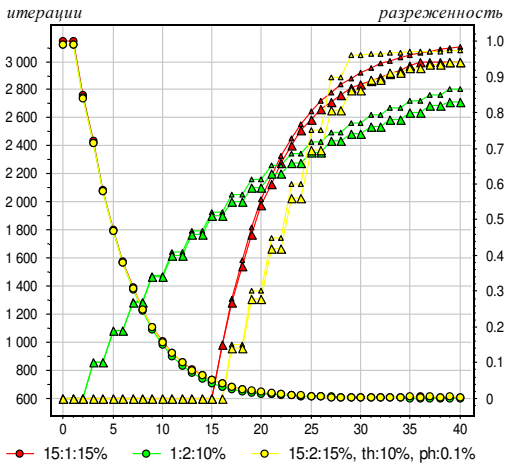
$$c(y, d) = [\text{документ } d \text{ принадлежит классу } y];$$
$$a(y, d) = [\text{классификатор относит документ } d \text{ к классу } y].$$

В роли классификатора может выступать какой-либо стандартный алгоритм. В задачах анализа текстов часто берут наивный байесовский классификатор, метод ближайших соседей kNN или метод опорных векторов SVM.

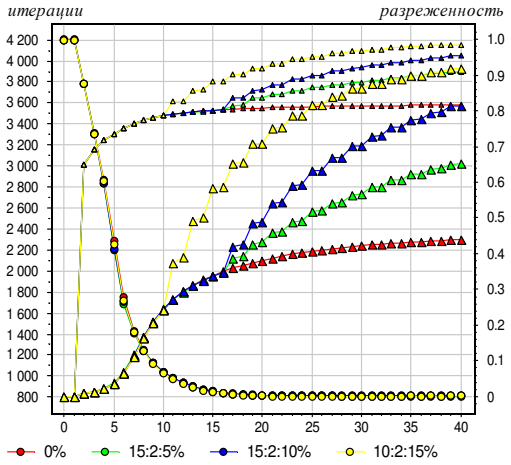
Качество классификации или категоризации принято измерять в терминах точности и полноты [21].



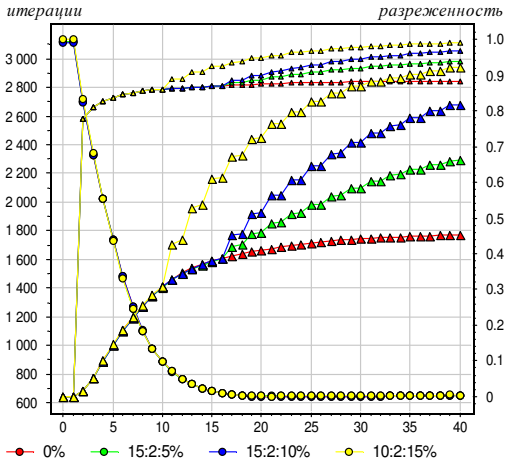
RuDis, агрессивное разреживание



NIPS, агрессивное разреживание



RuDis, SEM, через 2 итерации



NIPS, SEM, через 2 итерации

Рис. 4. Зависимость перплексии (○) и разреженности матриц Φ (△) и Θ (△) от числа итераций для рационального и стохастического *робастного* EM-алгоритма с параметрами робастности $\gamma = 0.3$, $\varepsilon = 0.01$ и параметрами разреживания $i_0{:}\delta{:}r$, $\mathsf{th}{:}R_\theta$, $\mathsf{ph}{:}R_\varphi$. Число тем $|T| = 100$.

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

$$P_y(a) = \frac{\sum\limits_{d \in D} a(y,d)c(y,d)}{\sum\limits_{d \in D} a(y,d)}.$$

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

$$R_y(a) = \frac{\sum\limits_{d \in D} a(y,d)c(y,d)}{\sum\limits_{d \in D} c(y,d)}.$$

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

Часто используется агрегированный показатель, называемый F_1-*мерой*:

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

Задачи категоризации, как правило, являются *многоклассовыми*, $|Y| \gg 2$. В таких случаях точность и полноту усредняют по всем классам одним из двух способов [21]: *макроусреднение*

$$P(a) = \frac{1}{|Y|}\sum_{y\in Y} P_y(a), \quad R(a) = \frac{1}{|Y|}\sum_{y\in Y} R_y(a),$$

и *микроусреднение*, менее чувствительное к ошибкам на классах с малым числом документов:

$$P_y(a) = \frac{\sum_{y\in Y}\sum_{d\in D} a(y,d)c(y,d)}{\sum_{y\in Y}\sum_{d\in D} a(y,d)}, \quad R_y(a) = \frac{\sum_{y\in Y}\sum_{d\in D} a(y,d)c(y,d)}{\sum_{y\in Y}\sum_{d\in D} c(y,d)}.$$

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

1.5.4 Эксперименты на модельных данных

Алгоритмы обучения тематических моделей тестируются на модельных данных. Для этого сначала задаются распределения $p(w\,|\,t)$ и $p(t\,|\,d)$, затем Алгоритмом 1.1 генерируется модельная коллекция. Хороший метод должен быть способен восстановить ту модель, которая породила данную коллекцию. Модельные данные можно генерировать различной длины n; можно добавлять в них шум — случайные пары (d_i, w_i) из распределения, заведомо плохо приближаемого тематической моделью; 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; можно задавать различное число тем $|T|$, а восстанавливать модель при другом числе тем, либо пытаться его определить. Эксперименты с варьированием модели данных позволяют исследовать устойчивость алгоритма 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 $p(w\,|\,t)$, $p(t\,|\,d)$, $p(w\,|\,d)$.

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] *Воронцов К. В., Потапенко А. А.* Модификации em-алгоритма для вероятностного тематического моделирования // *Машинное обучение и анализ данных.* — 2013 (в печати).

[3] *Лукашевич Н. В.* Тезаурусы в задачах информационного поиска. — Издательство МГУ имени М. В. Ломоносова, 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] *Celeux G., Chauveau D., Diebolt J.* On stochastic versions of the EM algorithm: Tech. Rep. RR-2514: INRIA, 1995.

[8] *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.

[9] *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.

[10] *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.

[11] *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.

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

[13] *Hoffman M. D., Blei D. M., Bach F. R.* Online learning for latent dirichlet allocation // NIPS. — Curran Associates, Inc., 2010. — Pp. 856–864.

[14] *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.

[15] *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.

[16] *Li X.-X., Sun C.-B., Lu P., Wang X.-J., Zhong Y.-X.* 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.

[17] *Lu Y., Mei Q., Zhai C.* Investigating task performance of probabilistic topic models: an empirical study of PLSA and LDA // *Information Retrieval.* — 2011. — Vol. 14, no. 2. — Pp. 178–203.

[18] *Masada T., Kiyasu S., Miyahara S.* Comparing LDA with pLSI as a dimensionality reduction method in document clustering // Proceedings of the 3rd International Conference on Large-scale knowledge resources: construction and application. — LKR'08. — Springer-Verlag, 2008. — Pp. 13–26.

[19] *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.

[20] *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.

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

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

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

[24] 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.

[25] *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.

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

[27] *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.

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

[29] *Wu Y., Ding Y., Wang X., Xu J.* A comparative study of topic models for topic clustering of chinese web news // Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on. — Vol. 5. — july 2010. — Pp. 236–240.

[30] *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. — Vol. 1. — IEEE Computer Society, 2010. — Pp. 209–213.

[31] *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.

[32] *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.

[33] *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.