Вероятностное тематическое моделирование
(задание по спецкурсу, весна 2016)

Воронцов Константин Вячеславович [voron@forecsys.ru]

11 марта 2016 г.

1 Теорминимум по тематическому моделированию

Пусть D — множество (коллекция) текстовых документов, W — множество (словарь) терминов. Известно число n_{dw} вхождений каждого из терминов w в каждый документ $d \in D$. Предполагается, что каждый термин в каждом документе связан с некоторой неизвестной (латентной) темой t из множества тем T. Задача заключается в том, чтобы выявить латентные темы по наблюдаемой коллекции D. *Тематическая модель* описывает вероятность появления терминов в документе:

$$p(w \mid d) = \sum_{t \in T} \phi_{wt}\, \theta_{td}, \tag{1}$$

где $\phi_{wt} = p(w \mid t)$ — неизвестное распределение на множестве терминов, определяющее тему t; $\theta_{td} = p(t \mid d)$ — неизвестное распределение на множестве тем, описывающее тематику документа d.

1.1 PLSA

В *вероятностном латентном семантическом анализе* PLSA [6] для обучения модели (1) по коллекции документов D максимизируется логарифм правдоподобия при ограничениях нормировки и неотрицательности:

$$L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln \sum_{t \in T} \phi_{wt} \theta_{td} \;\to\; \max_{\Phi, \Theta}; \tag{2}$$

$$\sum_{w \in W} \phi_{wt} = 1, \quad \phi_{wt} \geqslant 0; \qquad \sum_{t \in T} \theta_{td} = 1, \quad \theta_{td} \geqslant 0. \tag{3}$$

Задача (2), (3) является некорректно поставленной, поскольку имеет в общем случае бесконечно много решений. Произведение *матрицы терминов тем* $\Phi = (\phi_{wt})_{W \times T}$ и *матрицы тем документов* $\Theta = (\theta_{td})_{T \times D}$ определено с точностью до невырожденного линейного преобразования: $\Phi\Theta = (\Phi S)(S^{-1}\Theta)$. Решения данной задачи, получаемые с помощью итерационных численных методов, неустойчивы и существенно зависят от начального приближения.

1.2 ARTM

Аддитивная регуляризация тематических моделей ARTM [1, 10, 9] — это общий подход к использованию дополнительной информации для построения устойчивых моделей. Максимизируется линейная комбинация логарифма правдоподобия и дополнительных критериев $R_i(\Phi, \Theta)$, $i = 1, \ldots, k$, называемых *регуляризаторами*:

$$R(\Phi, \Theta) = \sum_{i=1}^{k} \tau_i R_i(\Phi, \Theta), \qquad L(\Phi, \Theta) + R(\Phi, \Theta) \;\to\; \max_{\Phi, \Theta}, \tag{4}$$

где τ_i — с неотрицательные *коэффициенты регуляризации.*

Если функция $R(\Phi, \Theta)$ непрерывно дифференцируема, то точка (Φ, Θ) локального экстремума задачи (4), (3) удовлетворяет системе уравнений со вспомогательными переменными $p_{tdw} = p(t \,|\, d, w)$:

$$p_{tdw} = \underset{t \in T}{\operatorname{norm}}\big(\phi_{wt}\theta_{td}\big); \tag{5}$$

$$\phi_{wt} = \underset{w \in W}{\operatorname{norm}}\left(n_{wt} + \phi_{wt}\frac{\partial R}{\partial \phi_{wt}}\right); \qquad n_{wt} = \sum_{d \in D} n_{dw} p_{tdw}; \tag{6}$$

$$\theta_{td} = \underset{t \in T}{\operatorname{norm}}\left(n_{td} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right); \qquad n_{td} = \sum_{w \in d} n_{dw} p_{tdw}; \tag{7}$$

где оператор неотрицательного нормирования $\underset{k \in K}{\operatorname{norm}} x_k = \frac{\max\{x_k, 0\}}{\sum_{s \in K} \max\{x_s, 0\}}$ преобразует произвольный вектор $(x_k)_{k \in K}$ в вектор вероятностей дискретного распределения.

Решение данной системы уравнений методом простых итераций приводит к EM-алгоритму. Сначала производится инициализация столбцов матриц Φ и Θ. Затем на каждой итерации выполняются два шага. На E-шаге (5) при фиксированных Φ и Θ вычисляются условные распределения $p_{tdw} = p(t \,|\, d, w)$. На M-шаге (6)–(7) при фиксированных p_{tdw} вычисляется следующее приближение матриц Φ и Θ.

1.3 Онлайновый алгоритм

В библиотеке тематического моделирования с открытым кодом BigARTM (`http://bigartm.org`) используется онлайновая реализация EM-алгоритма, ориентированная на обработку сверхбольших коллекций [9].

Идея онлайнового алгоритма заключается в том, что коллекция D разбивается на пакеты (batch) $D_1, \ldots, D_B$, которые могут обрабатываться параллельно, см. Алгоритм 1.1. Каждый пакет обрабатывается при фиксированной матрице Φ, затем обновления матрицы Φ, полученные от разных вычислителей, объединяются, и обновлённая версия матрицы Φ снова раздаётся вычислителям.

В BigARTM реализовано несколько вариантов EM-алгоритма. Далее в этом разделе используются обозначения из Python-интерфейса BigARTM.

Наиболее простой вариант — оффлайновый EM-алгоритм, реализованный функцией `fit_offline()`. Коллекция сканируется `num_collection_passes` раз, при этом матрицы Φ и Θ хранятся в памяти и обновляются один раз после каждого прохода по коллекции. Недостаток этого метода состоит в его плохой масштабируемости. Для больших коллекций матрица Θ, как правило, не может быть помещена в память.

Алгоритм 1.1. Онлайновый EM-алгоритм для АРТМ.

Вход: коллекция D, разбитая на пакеты $D_1, \ldots, D_B$; коэффициент $\rho \in (0, 1]$;
Выход: матрица Φ;

инициализировать ϕ_{wt} для всех $w \in W,\ t \in T$;
$n_{wt} := 0,\ \tilde{n}_{wt} := 0$ для всех $w \in W,\ t \in T$;
для всех пакетов D_b, $b = 1, \ldots, B$
    $(\tilde{n}_{wt}) := (\tilde{n}_{wt}) +$ `ProcessBatch`(D_b, Φ);
    если пора выполнить синхронизацию **то**
        $n_{wt} := (1 - \rho) n_{wt} + \rho \tilde{n}_{wt}$ для всех $w \in W,\ t \in T$;
        $\phi_{wt} := \underset{w \in W}{\operatorname{norm}}\big(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}}\big)$ для всех $w \in W,\ t \in T$;
        $\tilde{n}_{wt} := 0$ для всех $w \in W,\ t \in T$;

Функция `ProcessBatch`(D_b, Φ)
Вход: пакет D_b, матрица $\Phi = (\phi_{wt})$;
Выход: матрица $(\tilde{n}_{wt})$;

$\tilde{n}_{wt} := 0$ для всех $w \in W,\ t \in T$;
для всех $d \in D_b$
    инициализировать $\theta_{td} := \frac{1}{|T|}$ для всех $t \in T$;
    повторять
        $p_{tdw} := \underset{t \in T}{\operatorname{norm}}\big(\phi_{wt} \theta_{td}\big)$ для всех $w \in d,\ t \in T$;
        $\theta_{td} := \underset{t \in T}{\operatorname{norm}}\Big(\sum_{w \in d} n_{dw} p_{tdw} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\Big)$ для всех $t \in T$;
    пока θ_d не сойдётся;
    $\tilde{n}_{wt} := \tilde{n}_{wt} + n_{dw} p_{tdw}$ для всех $w \in d,\ t \in T$;

Кроме того, обновления матрицы Φ оказываются слишком редкими: приходится делать много проходов коллекции, каждый из которых занимает существенное время.

Альтернативой является онлайновый алгоритм 1.1, реализованный функцией `fit_online()`. Коллекция D разбивается на пакеты $D_1, \ldots, D_B$, и матрица Φ обновляется через каждые (`update_every`) пакетов. Если `update_every=1`, то обновление происходит после обработки каждого пакета. Функция делает только один проход по коллекции, при этом матрица Θ не хранится в памяти (`cache_theta=False`). Для каждого документа его Θ-столбец инициализируется равномерным распределением (`reuse_theta=False`), и производится несколько (`num_document_passes`) проходов по документу, чтобы распределение для документа сошлось.

Параметры `cache_theta`, `reuse_theta` и `num_document_passes` вынесены в конструктор ARTM-модели. Для оффлайнового сценария задаётся `cache_theta=True`, `reuse_theta=True` и `num_document_passes=1`. Возможен и гибридный подход, когда матрица Θ не хранится и обучается налету, но обновления Φ происходят после каждого прохода коллекции согласно `fit_offline()`.

Помимо уже описанных параметров, онлайновый алгоритм использует веса `decay_weight` и `apply_weight` для взвешивания старых счетчиков и счетчиков, подсчитанных, начиная с последнего обновления матрицы Φ. Один из подходов для

выбора этих весов описан в статье [5]:

$$\texttt{apply\_weight} = \rho, \quad \texttt{decay\_weight} = 1 - \rho, \quad \rho = (\tau_0 + j)^{-\kappа},$$

где j — номер итерации (обновления матрицы Φ), а τ_0 и $\kappа$ — новые параметры. Эти параметры (или веса напрямую) можно задавать в функции `fit_online()`. По умолчанию веса рассчитываются по значениям $\tau_0 = 1024,\ \kappа = 0.7$.

1.4 Синтетические данные

Некоторые задания предполагают выполнение вычислительных экспериментов на синтетических (модельных) данных. Каждый эксперимент заключается в многократном восстановлении синтетических (то есть искусственно сгенерированных и потому известных экспериментатору) «истинных» матриц Φ_0, Θ_0 при различных значениях некоторой выбранной характеристики задачи.

Отклонение восстановленных распределений $p(i \mid j)$ от исходных $p_0(i \mid j)$ измеряется средним расстоянием Хеллингера

$$H(p, p_0) = \frac{1}{m} \sum_{j=1}^{m} \sqrt{\frac{1}{2} \sum_{i=1}^{n} \Big(\sqrt{p(i \mid j)} - \sqrt{p_0(i \mid j)} \Big)^2},$$

как для самих матриц Φ, Θ, так и для их произведения $\Phi\Theta$:

$D_\Phi(\Phi, \Phi_0) = H(\Phi, \Phi_0)$;
$D_\Theta(\Theta, \Theta_0) = H(\Theta, \Theta_0)$;
$D_{\Phi\Theta}(\Phi\Theta, \Phi_0\Theta_0) = H(\Phi\Theta, \Phi_0\Theta_0)$.

Алгоритм 1.2. Восстановление стохастического матричного разложения.

1 сгенерировать синтетические матрицы Φ_0 и Θ_0;
2 сгенерировать коллекцию D;
3 EM-алгоритм: восстановить по коллекции D матрицы Φ и Θ;
4 венгерский алгоритм: найти соответствие между темами в (Φ_0, Θ_0) и (Φ, Θ);
5 вычислить расстояния D_Φ, D_Θ, $D_{\Phi\Theta}$;

Генерация реалистичных синтетических данных должна учитывать гипотезы разреженности, слабой корреляции тем, наличия фоновых тем. Возможно взять в качестве Φ_0, Θ_0 результат тематического моделирования реальной коллекции.

Простейший способ сгенерировать коллекцию по матрицам Φ_0, Θ_0:

$$n_{dw} = n_d \sum_{t \in T} \phi^0_{wt}\, \theta^0_{td},$$

где n_d — длина документа d. В этом случае существует точное решение, доставляющее нулевые значения расстояниям Хеллингера.

Однако такие синтетические данные не реалистичны тем, что счётчики терминов n_{dw} не являются целыми числами. Следующий способ генерирует коллекцию с минимальным уровнем шума:

$$n_{dw} = \text{pround}\Big(n_d \sum_{t \in T} \phi_{wt}\, \theta_{td} \Big),$$

где pround — функция вероятностного округления: число r округляется до $\lfloor r \rfloor$ с вероятностью $1-\{r\}$ и до $\lceil r \rceil$ с вероятностью $\{r\}$. Использование обычного округления приводит к расхождению сгенерированных данных с моделью, особенно в результате округлений к нулю и единице. Эксперименты с EM-алгоритмом лучше начинать без округления, чтобы проверить, сходятся ли расстояния $D_{\Phi\Theta}$ к нулю.

Задания предполагают построение графиков зависимости всех трёх расстояний от выбранной характеристики задачи.

На этапе тестирования модели строятся зависимости перплексии и расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ от номера итерации в EM-алгоритме. Это необходимо для выяснения необходимого числа итераций при заданных параметрах задачи.

Также строятся зависимости расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ и числа восстановленных тем от выбранных параметров синтетической задачи или метода тематического моделирования. Тема считается восстановленной, если для неё найдено взаимно однозначное соответствие с исходной темой.

2 Эксперименты с онлайновым алгоритмом

Цель экспериментов — исследование зависимости качества тематической модели от параметров онлайнового алгоритма: размера пакетов, числа итераций по документу, схемы взвешивания. Предполагается выработать рекомендации по установке этих параметров в зависимости от размера коллекции и длины документов. Эксперименты проводятся на реальных текстовых коллекциях.

Задание 1. Подобрать оптимальную стратегию выбора параметров онлайнового алгоритма `batch_size`, `update_every`, `num_document_passes`, `decay_weight`, `apply_weight` (либо τ_0 и κ). Результат должен быть сформулирован в виде рекомендаций вида «При увеличении параметра A стоит пропорционально уменьшать параметр B; это приведет к более долгому обучению, но лучшему качеству» или «При числе документов $|D|$ и средней длине $\bar{n}_d$ рекомендуется выбрать A таким-то». Для этого необходимо провести серии экспериментов на нескольких коллекциях различного размера.

Оптимальность стратегии понимается в смысле «наилучшее качество модели за наименьшее время». Поскольку одна итерация в онлайновом алгоритме может иметь совершенно разную вычислительную сложность, рекомендуется учитывать время работы процессора. Для сравнения разных работ в отчет необходимо включить время, затраченное на один проход по коллекции функции `fit_online()` с параметрами по умолчанию `num_document_passes=10`, `update_every=1`, а также указать число ядер процессора.

Качество модели оценивается рядом критериев, доступных в BigARTM: перплексия, разреженность Φ и Θ, чистота, контрастность и когерентность тем. Все эти критерии можно подключить к модели (`scores.add`), тогда они будут подсчитаны во время обучения при каждом обновлении матрицы Φ и доступны в `scores_tracker`. В таком сценарии перплексия будет каждый раз подсчитываться на части данных, обработанной между последними обновлениями. Чтобы производить итоговое сравнение моделей на одних и тех же данных одного и того же объема, рекомендуется выделить контрольную выборку, и после обучения модели оценить перплексию на ней. Для этого можно воспользоваться функциями `transform()` и `get_score()`.

3 Эксперименты с тематической сегментацией

Цель экспериментов. Проверка возможности восстановления сегментной тематической структуры документов. Исследование качества её восстановления в зависимости от длины сегментов и доли фоновых слов. Эксперименты проводятся на синтетических документах с известной сегментной тематической структурой.

Задание 2.

4 Эксперименты с аннотированием и поиском

Цель экспериментов. Совместная проверка алгоритмов аннотирования и тематического поиска. Аннотирование — это выделение в документе наиболее репрезентативных фраз и формирование его аннотации (краткого реферата). Тематический поиск позволяет по аннотации определять тематику и находить исходный документ. Чем меньше средняя позиция исходного документа в поисковой выдаче, тем лучше и аннотирование, и поиск. Исследуется зависимость качества поиска от длины аннотации, параметров алгоритмов аннотирования и поиска. Эксперименты проводятся на реальных текстовых коллекциях.

Задание 3.

5 Эксперименты с устойчивостью

Цель экспериментов. Проверка предположения, что разреженность и различность исходных тем способствует их лучшему восстановлению, особенно при правильном подборе комбинации регуляризаторов. Эксперименты проводятся на синтетических данных с известными «истинными» матрицами Φ_0 и Θ_0.

Известны различные условия единственности неотрицательных матричных разложений [3, 7, 8, 4]. В частности, разложение может быть единственным при сильной разреженности матриц Φ и Θ. Однако остаются не ясны вопросы: какой должна быть структура разреженности, т. е. как должны быть расположены нулевые элементы в матрицах Φ и Θ; какие дополнительные ограничения необходимо на них наложить, чтобы решение стало устойчивым; возможно ли связать эти ограничения с требованиями интерпретируемости (понятности) тем. Исследовательские задания направлены на поиск ответов на эти вопросы.

Задание 4. Проверяется гипотеза, что чем выше разреженность матриц Φ и Θ, тем выше устойчивость решения.

Вторая гипотеза: регуляризатор разреживания повышает устойчивость решения при условии, что (Φ_0, Θ_0) разрежены.

Строится зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ и числа восстановленных тем от разреженности (доли ненулевых элементов) исходных матриц.

Это задание повторяет эксперимент В.Глушаченкова, описанный в [2].

Задание 5. Проверяется гипотеза, что чем выше различность тем (столбцов Φ), тем выше устойчивость решения.

Вторая гипотеза: регуляризатор декоррелирования повышает устойчивость решения при условии, что темы (столбцы Φ_0) попарно существенно различны.

Третья гипотеза: регуляризаторы разреживания и декоррелирования вместе ещё сильнее повышают устойчивость решения при условии, что темы разрежены и попарно существенно различны.

Строится зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ и числа восстановленных тем от средней корреляции между темой и ближайшей к ней темой в исходной матрице Φ.

Задание 6. Проверяется гипотеза, что наличие фоновых тем не сильно мешает восстановить основные предметные темы.

Вторая гипотеза: при наличии плотных фоновых тем и существенно различных разреженных предметных тем в (Φ_0, Θ_0) совместное применение регуляризаторов разреживания, сглаживания и декорреляции повышает устойчивость решения.

Третья гипотеза: ошибка при назначении числа фоновых тем не сильно влияет на результат.

Строится зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ и числа восстановленных тем (только по предметным темам), от доли фоновых тем в документах коллекции. Затем, при фиксированной реалистичной доле фоновых тем (30%–70%) исследуется зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ от средней корреляции между темой и ближайшей к ней темой в исходной матрице Φ.

Задание 7. Проверяется гипотеза, что добавление в коллекцию «виртуальных документов», каждый из которых содержит список топовых слов одной темы, повышает устойчивость решения (применяется регуляризатор для частичного обучения [2]).

Вторая гипотеза: достаточно лишь небольшого числа топовых слов.

Третья гипотеза: достаточно задать топовые слова лишь небольшой части тем.

Строится зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ и числа восстановленных тем от числа виртуальных документов и суммарного числа слов в виртуальных документах.

Задание 8. Проверяется гипотеза, что если матрицы Φ_0, Θ_0 разрежены, темы попарно существенно различны и число тем велико (скажем, $|T| = 1000$), то при использовании много меньшего числа тем в EM-алгоритме (скажем, $|T| = 100$) многие темы восстанавливаются, но каждый раз разные в зависимости от случайного начального приближения.

Вторая гипотеза: при использовании регуляризаторов разреживания и декоррелирования восстановленные темы получаются более близкими к исходным.

Третья гипотеза: при добавлении регуляризатора сглаживания фоновых тем большее число предметных тем восстанавливаются правильно.

Строится зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$ и числа восстановленных тем от разреженности (доли ненулевых элементов) исходных матриц.

Задание 9. Проверяется гипотеза, что если матрицы Φ_0, Θ_0 разрежены, темы попарно существенно различны и число тем не велико (скажем, $|T| = 100$), но в EM-алгоритме число тем изначально велико ($|T'| = 1000$ и выше), то в результате совместного применения регуляризаторов разреживания, декоррелирования и отбора тем (строкового разреживания Θ) исходные темы восстанавливаются более устойчиво.

Строится зависимость расстояний D_Φ, D_Θ, $D_{\Phi\Theta}$, числа отобранных тем и числа восстановленных тем от номера итерации и от начального числа тем $|T'|$.

Список литературы

[1] *Воронцов К. В.* Аддитивная регуляризация тематических моделей коллекций текстовых документов // *Доклады РАН.* — 2014. — Т. 456, № 3. — С. 268–271.

[2] *Воронцов К. В.* Вероятностное тематическое моделирование. — 2014. http://www.MachineLearning.ru/wiki/images/2/22/Voron-2013-ptm.pdf.

[3] *Donoho D., Stodden V.* When does non-negative matrix factorization give a correct decomposition into parts? // Advances in Neural Information Processing Systems 2003 / Ed. by S. Thrun, L. Saul, B. Schölkopf. — Cambridge, MA: MIT Press, 2004. http://www-stat.stanford.edu/ donoho/Reports/2003/NMFCDP.pdf.

[4] *Gillis N.* Sparse and unique nonnegative matrix factorization through data preprocessing // *Journal of Machine Learning Research.* — 2012. — Vol. 13, no. 1. — Pp. 3349–3386.

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

[6] *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.

[7] *Lauerberg H., Christensen M., Pumbley M., Hansen L., Jensen S.* Theorems on positive data: On the uniqueness of nmf // *Computational Intelligence and Neuroscience.* — 2008. — Vol. 2008. — P. 10.

[8] *Schachtner R., Pöppel G., Lang E. W.* Towards unique solutions of non-negative matrix factorization problems by a determinant criterion // *Digital Signal Processing.* — 2011. — Vol. 21, no. 4. — Pp. 528–534.

[9] *Vorontsov K., Frei O., Apishev M., Romov P., Suvorova M., Yanina A.* Non-bayesian additive regularization for multimodal topic modeling of large collections // Proceedings of the 2015 Workshop on Topic Models: Post-Processing and Applications. — New York, NY, USA: ACM, 2015. — Pp. 29–37.

[10] *Vorontsov K. V., Potapenko A. A.* Additive regularization of topic models // *Machine Learning, Special Issue on Data Analysis and Intelligent Optimization with Applications.* — 2015. — Vol. 101, no. 1. — Pp. 303–323.