САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

КАФЕДРА ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

Тимофеев Александр Иванович

Выпускная квалификационная работа бакалавра

Автоматическое выделение информативных тем документов с использованием латентного размещения Дирихле

Направление 010400.62

Прикладная математика, фундаментальная информатика и программирование

Научный руководитель,
старший преподаватель
Романенко Е. С.

Санкт-Петербург

2017

Содержание

Введение

Ежедневно собираются огромные объемы данных, при постоянном пополнении которых обработка и выделение требуемой информации становится нетривиальной задачей. С этой целью создаются специальные инструменты, предназначенные для организации, поиска и понимания огромного количества информации.

Тематическое моделирование предоставляет нам методы для организации, понимания и обобщения больших коллекций текстовой информации. И таким образом, помогает в обнаружении скрытых тематических характеристик коллекции. Тематическое моделирование может быть описано как метод поиска групп слов (тем) из набора документов, которые наилучшим образом представляют информацию в коллекции. Его также можно рассматривать как форму интеллектуального анализа текста – способ получения повторяющихся паттернов слов в текстовом материале.

Существует множество методов, которые используются для получения тематических моделей. Большинство из них принадлежат классу вероятностного тематического моделирования. Вероятностная тематическая модель представляет темы как дискретные распределения на множестве слов, а документы – как дискретное распределение на множестве тем. При построении тематической модели ставится задача восстановить эти распределения по данной коллекции документов. Поскольку документ может относиться сразу к нескольким темам, говорят, что тематическое моделирование осуществляет “нечеткую кластеризацию” [1].

Задача извлечения скрытых тематических характеристик текстовой коллекции тесно связана со многими другими прикладными задачами, в частности, задачами информационного поиска [2]. Это может быть анализ данных социальных сетей, классификация и кластеризация документов, для рекомендательных систем [3], и др.

На сегодняшний день разработано уже достаточно методов для построения тематических моделей, однако в основном они применимы лишь для извлечения тем, и слабо пригодны для реальных прикладных задач. Например, необходимо исследовать, как изменялись темы коллекции с течением времени. При условии наличия информации о времени создания документов коллекции можно анализировать информацию о перетекании одной темы в другую, возникновении абсолютно новых тем, либо исчезновении старых. К этой задаче можно добавить другую: выявление трендов определенных тем [4]. Цель данной работы – предложить подход для решения задачи, описанной выше, а именно задачи прогнозирования трендов скрытых тематик коллекции.

Постановка задачи

Пусть дана коллекция текстовых документов. Для каждого документа имеется множество оценок, присвоенных ему в определенный период времени. Задача заключается в следующем:

1. Для данной размеченной коллекции текстов определить оптимальное количество латентных тем.
2. Выделить латентные темы.
3. По возможности, среди выделенных тем определить наиболее информативные.
4. Исследовать информацию о перетекании тем со временем.
5. На основании полученной информации и информации о оценках документов построить временной ряд, описывающий динамику рейтингов скрытых тем.
6. По данному временному ряду построить прогнозирующую модель.

Глава 1. Вероятностные тематические модели

В данной главе описываются стандартные процедуры обработки текста перед построением тематической модели, а также классические методы вероятностного тематического моделирования

1.1. Предварительная обработка данных

Стоп-слова. Стоп-словами, или шумовыми, называются слова, которые встречаются во многих текстах различной тематики. Такие слова бесполезны для тематического моделирования, т.к. не несут информации о конкретной теме, и, следовательно, должны быть проигнорированы. В русском языке к ним относят, например, указательные слова, союзы, частицы. В английском языке примером могут служить артикли. Число уникальных стоп-слов в тексте обычно невелико. Их отбрасывание почти не влияет на длину словаря, но может приводить к заметному сокращению длины некоторых текстов.

Стемминг. Одно и то же слово может встретиться в тексте в разных грамматических формах (например, спряжения, склонения). Эти формы в основном несут схожий смысл, поэтому при построении тематической модели их не различают. Учитывая эти формы, мы получаем много больший размер словаря, увеличение ресурсоёмкости и снижение качества модели. Поэтому применяется техника стемминга. Стемминг — процесс нахождения основы слова путем отбрасывания изменяемых частей. Данная техника основана на морфологических правила языка. Т.к. эти правила обычно не строго сформулированы, и в каждом языке имеет место множество исключений, при стемминге имеется довольно крупный процент ошибок.

Отбрасывание редких слов. Нельзя сказать, что слова, встречающиеся в длинной коллекции слишком редко, например, только один раз, несут в себе много информации о какой-то теме. Поэтому такие слова, как и стоп-слова, можно отбрасывать.

Векторное представление текста. Современные алгоритмы машинного обучения работают с документами, которые представлены в виде векторов признаков. Используя статическую информацию о словах, можно преобразовать текст в векторное представление. Данная операция проводится следующим образом: рассматривается выборка текстов, каждый объект из которой это вектор, элементами которого является количество используемых слов во всей выборке.

Bag of Words. Bag of Words (мешок слов) — векторная модель текстового документа. В этой модели текст представляется как неупорядоченное множество содержащихся в нем слов. Коллекцию документов при этом можно рассматривать как простую выборку пар «документ–слово» *(d, w)*, где $d \in D$, $w \in W$. В мешке слов коллекция документов представляется в виде матрицы $T = \{t\}_{dw}$, в которой строки представляют собой векторы количеств вхождений каждого слова в документ d, а столбцы — векторы количеств вхождений слова w в каждый документ. Элемент t_{dw} соответственно равен количеству вхождений слова w в документ d.

1.2. Вероятностная модель

Пусть D – множество текстовых документов, W – множество уникальных терминов этой коллекции, так же именуемое словарем. В вероятностном тематическом моделировании *темой* называется некая совокупность слов, часто употребляющихся совместно в документах коллекции. Предполагается, что существует конечное множество тем T, и каждое употребление слова w в каждом документе d связано с некоторой заранее неизвестной темой $t \in T$. Коллекция документов обычно рассматривается как случайная и независимая выборка троек (w_i, d_i, t_i), $i = 1, \dots, n$ из дискретного распределения $p(w, d, t)$ на конечном множестве $W \times D \times T$. слова w и документы d являются наблюдаемыми

переменными, тема $t \in T$ является латентной (скрытой) переменной. Каждая тема представляется как дискретное распределение на множестве слов $\phi_{wt} = p(w|t)$, а каждый документ – как дискретное распределение на множестве тем $\theta_{td} = p(t|d)$.

В вероятностном тематическом моделировании принимается гипотеза условной независимости появления в документе слова *w*, связанного с темой *t*, от документа *d*:

$$p(w\,|t) \;=\; p(w \mid d, t). \qquad (1)$$

Из формулы полной вероятности и (2) получим модель коллекции в виде:

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

Вероятностная модель описывает процесс порождения документов по известным ϕ_{wt} и θ_{td}. Этот процесс можно описать следующим образом:

1. **для** каждого слова документа:
    1. выбрать тему из распределения θ_{td};
    2. выбрать слово из распределения ϕ_{wt}.

Построение тематической модели заключается в настройке параметров ϕ_{wt} и θ_{td} по заданной матрице вероятностей $p(w|d)$, т.е. по сути, является обратной задачей по отношению к порождению документов. Число тем $|T|$ обычно намного меньше $|D|$ и $|W|$, и задачу построения тематической модели можно трактовать как нахождение разложения заданной матрицы частот

$$F = \{p_{wd}\}_{W \times D}, \qquad p_{wd} = {}^{n_{dw}}/_{n_d}, \qquad p_d = \{p_{wd}\}_{d \in D}$$

в виде $F \;\approx\; \Phi\Theta$ двух стохастических матриц — матрицы *термины-темы* Φ и матрицы *темы-документы* Θ:

$$\Phi = \{\phi_{wt}\}_{W\times D}, \qquad \phi_t = \{\phi_{wt}\}_{w\in W}.$$

$$\Theta = \{\theta_{td}\}_{W\times D}, \qquad \theta_d = \{\theta_{td}\}_{d\in D}.$$

Матрицы F, Φ, Θ являются стохастическими, то есть имеют неотрицательные нормированные столбцы p_d, ϕ_t, θ_d, представляющие дискретные распределения.

1.3. Классическая PLSA модель

Модель вероятностного латентного семантического анализа (probabilistic latent semantic analysis) была предложена впервые в 1999 Томасом Хофманном [5]. Модель документа представляется как

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

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

Построение данной модели заключается в настройке параметров ϕ_{wt} и θ_{td} по заданной матрице вероятностей $p(w|d)$. Данная оптимизационная задача решается максимизацией логарифмического правдоподобия, с использованием, например, EM-алгоритма:

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

при условиях нормировки (3).

1.4. EM-алгоритм

EM-алгоритм – это итерационный процесс, каждая итерация которого состоит из двух шагов: E (expectation) и M (maximization). Рассмотрим его применение при построении модели PLSA.

Вначале задается начальное приближение параметров ϕ_{wt} и θ_{td}. В качестве начальных значений обычно берутся произвольные положительные нормированные векторы.

Далее на E-шаге считается, сколько слов w в документе d принадлежит теме t:

$$n_{dwt} = n_{dw} p(t|d,w) = n_{dw} \frac{p(w|t)p(t|d)}{p(w|d)} = n_{dw} \frac{\phi_{wt}\,\theta_{td}}{\sum_{s\in T} \phi_{ws}\,\theta_{sd}}.$$

А на M-шаге пересчитываются параметры модели:

$$\phi_{wt} = \frac{n_{wt}}{n_t}, \qquad n_{wt} = \sum\nolimits_{d\in D} n_{dwt}\,, \qquad n_t = \sum\nolimits_{w\in W} n_{wt},$$

$$\theta_{td} = \frac{n_{td}}{n_d}, \qquad n_{td} = \sum\nolimits_{w\in W} n_{dwt}\,, \qquad n_d = \sum\nolimits_{t\in T} n_{dwt}.$$

Эти формулы обычно для краткости записываются как

$$\phi_{wt} \propto n_{wt}, \qquad \theta_{td} \propto n_{td}.$$

1.5. Классическая LDA модель

Модель латентного размещения Дирихле (Latent Dirichlet Allocation) была предложена Дэвидом Блеем в 2003 [6].

Данная модель основана на тех же предположениях, что и PLSA, при дополнительных условиях, накладываемых на параметры модели. Принимаются предположения, что распределения слов в темах и тем в документах порождаются распределением Дирихле с параметрами α и β соответственно:

$$Dir(\theta_d;\alpha) = \frac{\Gamma(\alpha_0)}{\prod_t \Gamma(\alpha_t)} \prod_t \Gamma\left(\theta_{td}^{\alpha_t - 1}\right),$$

$$\alpha_t > 0, \qquad \theta_{td} > 0, \qquad \alpha_0 = \sum_t \alpha_t\,, \qquad \sum_t \theta_{td} = 1.$$

$$Dir(\phi_t;\beta) = \frac{\Gamma(\beta_0)}{\prod_w \Gamma(\beta_w)} \prod_w \Gamma\left(\phi_{wt}^{\beta_w - 1}\right),$$

$$\beta_w > 0, \qquad \phi_{wt} > 0, \qquad \beta_0 = \sum_w \beta_w, \qquad \sum_w \phi_{wt} = 1,$$

где Γ(x) – гамма-функция.

Два наиболее часто используемых метода обучения данной модели основаны на вариационном выводе и на сэмплировании Гиббса [7]. Первый метод подробно описан в [8], и по сути является вариантом EM-алгоритма, в котором на каждой итерации оценки ϕ_{wt} и θ_{td} сглаживаются в числителе и знаменателе:

$$\phi_{wt} = \frac{n_{wt} + \alpha_t}{n_t + \alpha_0},$$

$$\theta_{td} = \frac{n_{td} + \beta_w}{n_d + \beta_0}.$$

Процесс генерации документа согласно LDA модели выглядит следующим образом:

1. сгенерировать распределение $\theta_d \sim Dir(\alpha)$;
2. сгенерировать распределение $\phi_t \sim Dir(\beta)$;
3. **для** каждого слова документа:
    1. выбрать тему из распределения θ_{td};
    2. выбрать слово из распределения ϕ_{wt};

В этом и заключается принципиальное отличие LDA модели от PLSA.

Исследования [6] показывают, что при использовании LDA получается выделить более интерпретируемые темы в сравнении со стандартной PLSA моделью, в то время как никаких лингвистических обоснований использование распределений Дирихле не имеет.

1.6. Иерархическая LDA модель

Одним из недостатков скрытого размещения Дирихле является тенденция к извлечению слишком общих тем для заданного набора документов. В случае, когда некоторая концепция имеет ряд смыслов, которые часто употребляются совместно с основной концепцией, в результатах работы LDA с большой вероятностью будет присутствовать тема, которая включает как основной смысл концепции, так и все её аспекты. Часто необходимо, чтобы в отдельные темы была выделена не только основная концепция, но и различные её аспекты.

Другой проблемой, непосредственно связанной с ранее упомянутой, является ситуация, когда нет информации даже о приблизительном количестве тем в корпусе документов. Помимо параметров-распределений модели приходится подбирать также и количество тем для выделения.

В таких случаях используются иерархические тематические модели, позволяющие моделировать иерархию тем — от более общих до узких.

Первое обобщение LDA для построения иерархий было предложено автором LDA, Дэвидом Блеем, в [9]. Она основана на вложенном процессе китайского ресторана (Nested Chinese Restaurant Process, nCRP) [10]. Иерархия здесь представляется в виде дерева тем, глубина дерева известна и фиксирована. Модель сама определяет количество подтем в каждом узле, для этого априорное распределение на путь документа от корня к листу задается с помощью nCRP, принимающего в качестве параметров глубину дерева и коэффициент γ. Идея алгоритма состоит в следующем: чтобы построить путь от вершины дерева к корню, нужно для каждой вершины выбрать одну из существующих вершин-подтем, либо создать новую. Каждая вершина выбирается с вероятностью, пропорциональной количеству документов, уже относящихся к этой вершине. Новая вершина-подтема создается с

вероятностью, пропорциональной γ. В результате процесс генерации документа коллекции выглядит следующим образом:

1. начать с корня дерева;
2. **для** всех слоев дерева:
    1. выбрать подтему s с помощью nCRP;
3. сгенерировать распределение над множеством выбранных подтем $\theta_d \sim Dir(\alpha)$;
4. сгенерировать распределение над множеством слов $\phi_t \sim Dir(\beta)$;
5. **для** каждого слова документа:
    1. сгенерировать слой из распределения θ_d;
    2. сгенерировать слово из распределения ϕ_t.

Обучение модели производится с использованием вариации EM-алгоритма. Информацию о модификациях формул E- и M- шага можно найти в [8].

1.7. Подход ARTM

Задача построения указанных тематических моделей по коллекции документов является задачей стохастического разложения матрицы $F = \Phi\Theta$. Эта задача является некорректно поставленной, так как множество ее решений бесконечно: если $F = \Phi\Theta$ – решение, то $F = (\Phi S)(S^{-1}\,\Theta)$ также является решением при невырожденных S таких, что матрицы ΦS и $S^{-1}\Theta$ - стохастические. Выбор матрицы S в EM-подобных алгоритмах никак не контролируется и зависит от начального приближения. При неопределенности в оптимизационных задачах к основному критерию добавляют дополнительные – регуляризаторы. Такой подход к решению данной проблемы называется регуляризацией.

Аддитивная регуляризация тематических моделей (additive regularization for topic modeling) [11] основана стандартной PLSA модели и введении дополнительных критериев-регуляризаторов $R_i(\Phi, \Theta),\ i = 1, \dots, r$. Для

построения модели необходимо решить задачу максимизации линейной комбинации регуляризаторов с логарифмом правдоподобия $L(\Phi, \Theta)$ с положительными коэффициентами:

$$L(\Phi, \Theta) + \sum_{i=1}^{r} \tau_i R_i(\Phi, \Theta) \to \max_{\Phi, \Theta},$$

$$\sum_{w \in W} \phi_{wt} = 1, \phi_{wt} \geq 0; \quad \sum_{t \in T} \theta_{td} = 1\,, \theta_{td} \geq 0;\ \tau_i\ \geq 0.$$

Задача, как и в случае PLSA, заключается в оптимизации параметров Φ и Θ, и решается с использованием EM-алгоритма. В этом случае формула M-шага выглядит следующим образом:

$$\phi_{wt} \propto \left(n_{wt} + \phi_{wt} \frac{\partial R}{\partial \phi_{wt}}\right)_+, \qquad \theta_{td} \propto \left(n_{dt} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\right)_+,$$

где $(x)_+ = \max(x, 0)$.

1.8. LDA в ARTM

В терминах ARTM модель LDA выражается через сглаживающие регуляризаторы следующим образом:

$$R(\Phi, \Theta) = \beta_0 \sum_{t,w} \beta_{wt} ln\, \phi_{wt} + \alpha_0 \sum_{d,t} \alpha_{td} ln\, \theta_{td},$$

где β_0, α_0 – коэффициенты регуляризации. При этом векторы $\beta_0 \beta_t$ и $\alpha_0 \alpha_t$ соответствуют гиперпараметрам априорных распределений Дирихле.

Формула M шага EM-алгоритма при этом имеет вид:

$$\phi_{wt} \propto (n_{wt} + \beta_0 \beta_t)_+, \qquad \theta_{td} \propto (n_{dt} + \alpha_0 \alpha_t)_+.$$

1.9. Оценка качества модели

Перплексия. В отличие от множества других задач машинного обучения в тематическом моделировании нет точного понятия «ошибки».

Общепринятой мерой качества вероятностной тематической модели является перплексия (perplexity) контрольной выборки. Она рассчитывается через логарифм правдоподобия:

$$P(D^{test}) = \exp\left\{-\frac{1}{n}L(\Phi,\Theta)\right\} = exp\left\{-\frac{1}{n}\sum_{d\in D^{test}}\sum_{w\in W} n_{dw}\ln p(w|d)\right\},$$

Где $n = \sum_{d\in D^{test}}\sum_{w\in W} n_{dw}$ – длина тестовой коллекции.

Впервые ее начали использовать в вычислительной лингвистике при оценивании моделей языка [8]. Чем ниже значение перплексии, тем лучше модель описывает тестовые данные. Перплексия измеряется по контрольной выборке документов, не используемых для построения модели. Это позволяет избежать занижения оценки в результате переобучения.

Перпелексию можно интерпретировать как ожидаемый размер словаря с равномерным распределением слов, который необходим модели чтобы сгенерировать слово из тестовой выборки. С ее помощью можно сравнивать модели, обученные на одной коллекции, но, например, имеющие различные параметры.

Глава 2. Прогнозирование

В данной главе вводится метрика расстояния между распределениями, а также кратко рассматривается модель прогнозирования временных рядов ARIMA.

2.1. Дивергенция Йенсена-Шеннона

В теории вероятностей и математической статистике существуют различные методы сравнения двух вероятностных распределений. Одним из самых известных является дивергенция Кульбака-Лейблера [12]. В случае дискретных распределений P и Q с числом событий n мера Кульбака-Лейблера определяется как

$$D_{KL}(P||Q) = \sum_{i=1}^{n} p_i log \frac{p_i}{q_i}.$$

Недостатками данного функционала является его несимметричность и нарушение неравенства треугольника. И, хотя он часто рассматривается в качестве расстояния между вероятностными распределениями, нельзя утверждать, что данный функционал является метрикой в пространстве распределений.

В данной работе для определения расстояния между двумя вероятностными распределениями используется дивергенция Йенсена-Шеннона [13]. Она основана на расстоянии Кульбака-Лейблера с некоторыми полезными отличиями, в частности, симметричностью. Для двух дискретных распределений P и Q одинаковой размерности дивергенция Йенсена-Шеннона определяется как

$$D_{JSD}(P, Q) = \frac{1}{2} D_{KL}(P||M) + \frac{1}{2} D_{KL}(Q||M),$$

где $M = \frac{1}{2}(P + Q)$. Значения данного функционала лежат в границах [0; 1] при условии, что в вычислении D_{KL} использовался логарифм с основанием 2. При этом $D_{JSD}(P, Q) = 0$ тогда и только тогда, когда P совпадает с Q.

2.2. Модель ARIMA

Временным рядом называются данные, последовательно измеренные через некоторые промежутки времени. Стационарным временным рядом называется ряд, вероятностные свойства которого не меняются с течением времени. Временной ряд называется интегрированным порядка d, если разности ряда порядка d являются стационарными, в то время как разности меньшего порядка не являются стационарными. Для проверки временного ряда на стационарности применяется тест Дики-Фуллера [16].

Одним из подходов моделирования временных рядов является авторегрессионная модель (AR). Она подразумевает, что значение временного ряда в данный момент зависит линейно от предыдущих значений ряда. Порядком модели называют число предыдущих значений ряда, от которых зависит текущее.

$$X_t = c + \sum_{i=1}^{p} a_i X_{t-i} + \varepsilon_t,$$

где $c, a_i, i = 1, \dots, p$ – параметры модели, ε_t – белый шум, p – порядок модели.

Модель скользящего среднего (MA) является другим подходом к моделированию временного ряда. Она задается следующим образом:

$$X_t = \sum_{j=0}^{q} b_j \varepsilon_{t-j} + \varepsilon_t,$$

где $b_j, j = 0, \dots, q$ – параметры модели, ε_{t-j} – ошибки, q – порядок модели.

Модель авторегрессионного скользящего среднего (ARMA) является обобщением двух ранее упомянутых моделей. Она представляет временной ряд в виде суммы двух компонент: авторегрессионной модели порядка p и модели скользящего среднего порядка q. Данная модель применяется для анализа и прогнозирования стационарных временных рядов.

В случае нестационарных рядов используется модель ARIMA – интегрированная модель авторегрессионного скользящего среднего. Для нестационарного временного ряда X_t она строится как

$$\Delta^d X_t = c + \varepsilon_t + \sum_{i=1}^{p} a_i \Delta^d X_{t-i} + \sum_{j=0}^{q} b_j \varepsilon_{t-j},$$

где c, a_i, b_j – параметры модели d – порядок модели, Δ^d – оператор разности временного ряда порядка d. Порядок d модели определяется как порядок интегрированности временного ряда. При $d = 0$ (стационарности ряда X_t), модель совпадает с моделью ARMA.

Подход ARIMA к временным рядам заключается в том, что в первую очередь оценивается стационарность ряда. Различными тестами выявляется порядок интегрированности временного ряда, т.е. порядок модели d. Для определения порядков p и q может применяться исследование автокорреляционной функции и частной автокорреляции функции. Для определения коэффициентов применяются такие методы, как метод наименьших квадратов и метод максимального правдоподобия.

Глава 3. Эксперимент

В данной главе описывается ход работы над решением поставленной задачи.

Анализ начальных данных. Для исследования была выбрана открытая коллекция Book-Crossing Dataset [3]. Она содержит информацию о ISBN, названии, авторе и годе публикации более 270 000 книг, а также более 1млн. оценок книг пользователями сети. В качестве текстовых данных использовались аннотации к этим книгам. Коллекция не содержит аннотаций, поэтому они были получены с сайта www.lookupbyisbn.com путем запроса по ISBN. Для парсинга ответов сервиса использовалась библиотека BeautifulSoup4. Таким образом было получено около 90 000 описаний к книгам.

Было проанализировано распределение полученных книг по году публикации, которое представлено на рис. 1.

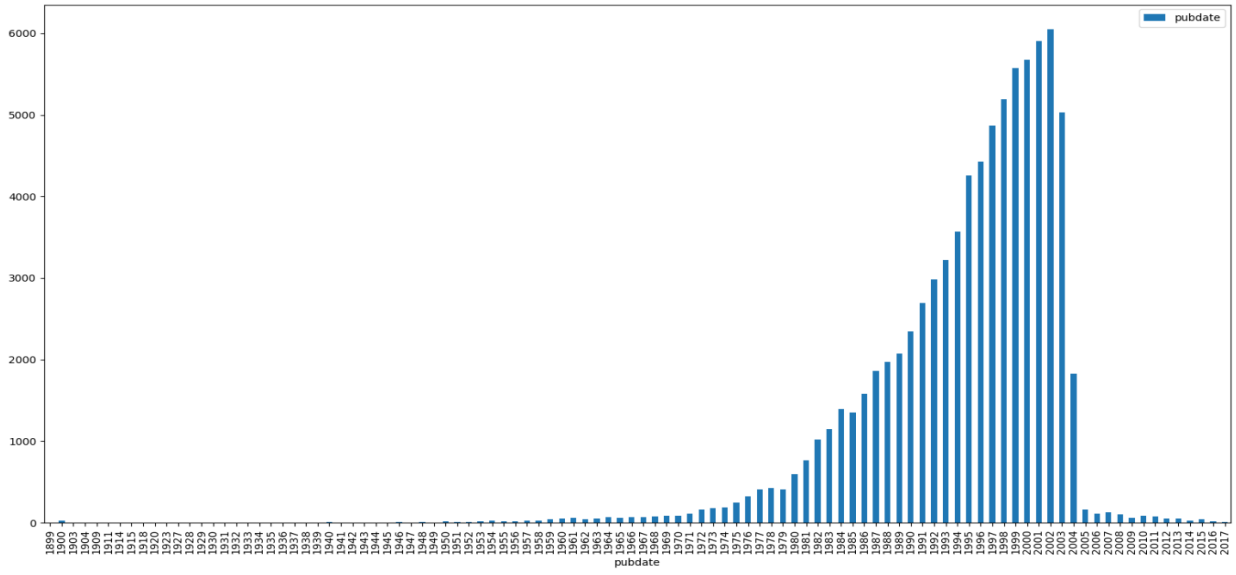

Рисунок 1. Распределение книг по году публикации

Из данного графика видно, что года 1997-2003 обладают достаточным объемом публикаций для обучения модели. Так для каждого года из данного

промежутка было рассмотрено распределение публикаций по месяцам, которое представлено на рис. 2.

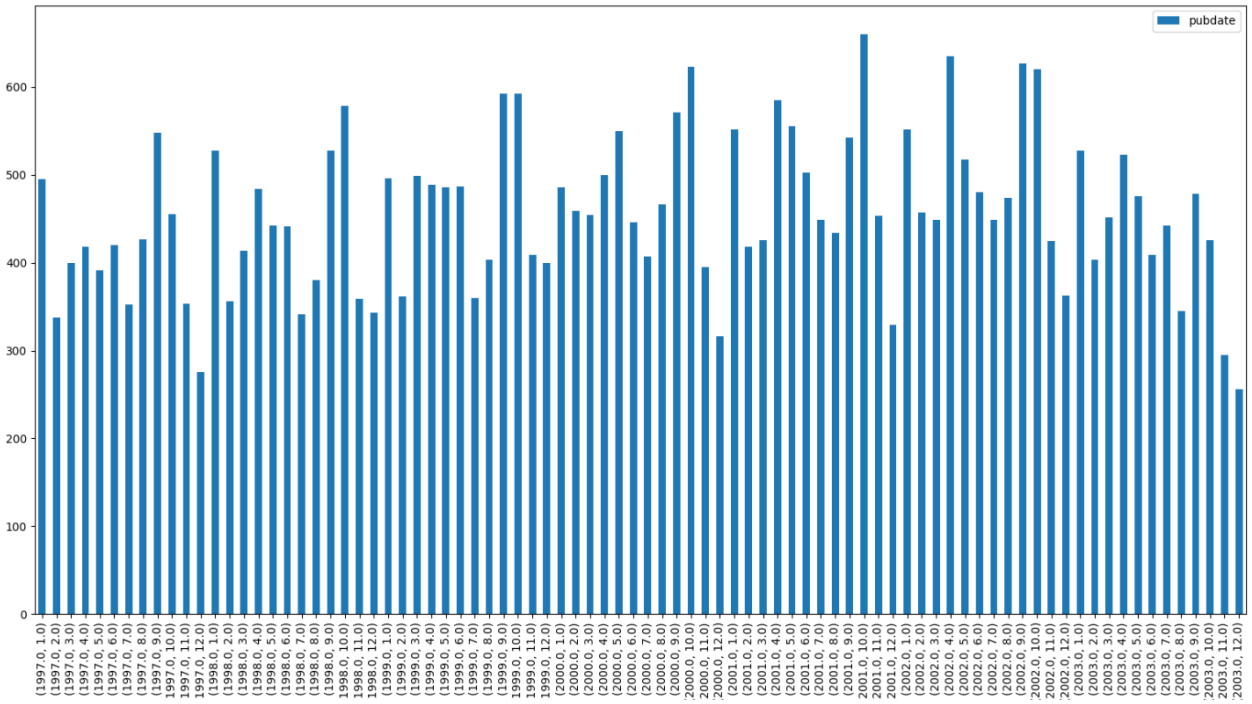

Рисунок 2. Распределение публикаций по месяцам

Из рис. 2 видно, что по месяцам данные распределены равномерно. Публикации за любой год могут быть использованы в качестве коллекции для обучения. Были выбраны документы за 2001, 2002 и 2003 года.

Выбранные данные подверглись предобработке – из каждого текста были выделены токены и стемы для дальнейшего приведения текстов в формат “Vowpal Wabbit”.

Таблица 1. Параметры коллекции

	2001 г.	2002 г.	2003 г.
Количество документов	5904	6048	5030
Размер словаря	33562	35063	31395
Количество слов в коллекции	454191	484313	415158

В рамках тематического моделирования коллекция обычно описывается такими параметрами, как длина коллекции, количество слов в словаре и число ненулевых счетчиков в “мешке слов”. Эта информация представлена в табл. 1.

Построение тематических моделей. С использованием BigARTM [14] были обучены три hLDA модели для каждого года. Сглаживающий регуляризатор Дирихле в каждой из них имеет параметры $\beta = 0.1$ и $\alpha = 0.3$. Иерархия имеет структуру дерева, на где на i-ом слое находится $f(i)$ вершин-тем, где

$$f(i) = \begin{cases} i, & 1 \leq i \leq 10 \\ 10 * (i - 9), & 11 \leq i \leq 24 \end{cases}.$$

Для тестирования модели использовался прием кросс-валидации: выборка делилась в отношении 3:1, на большей части обучалась модель, на меньшей – тестировалась. Графики перплексии на тестовом множестве приведены на рис. 3, 4, 5.

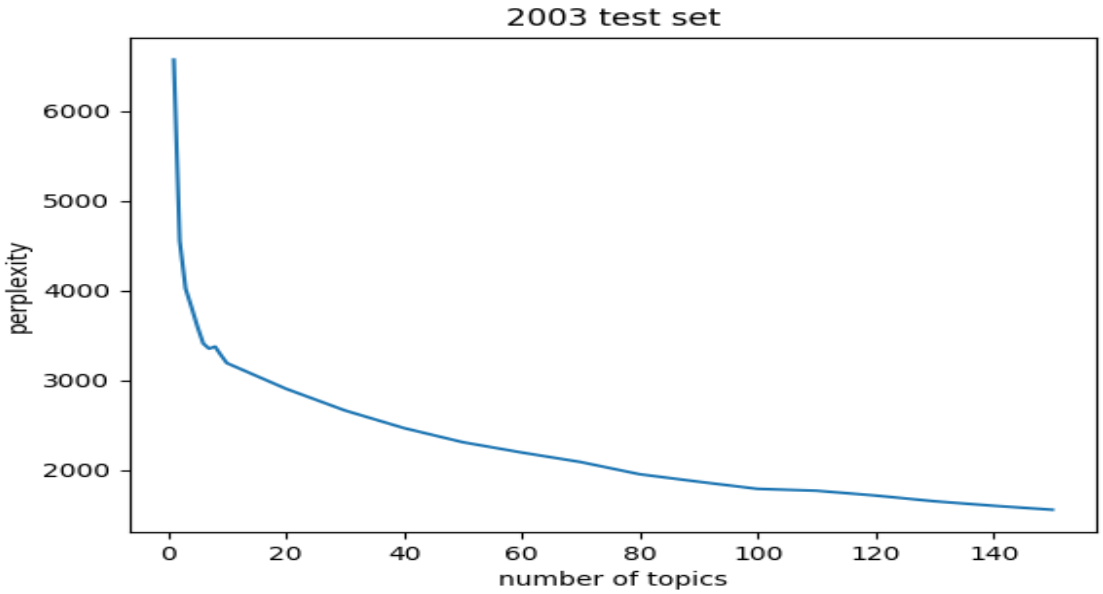

Рисунок 3. График перплесии на слоях, 2003г.

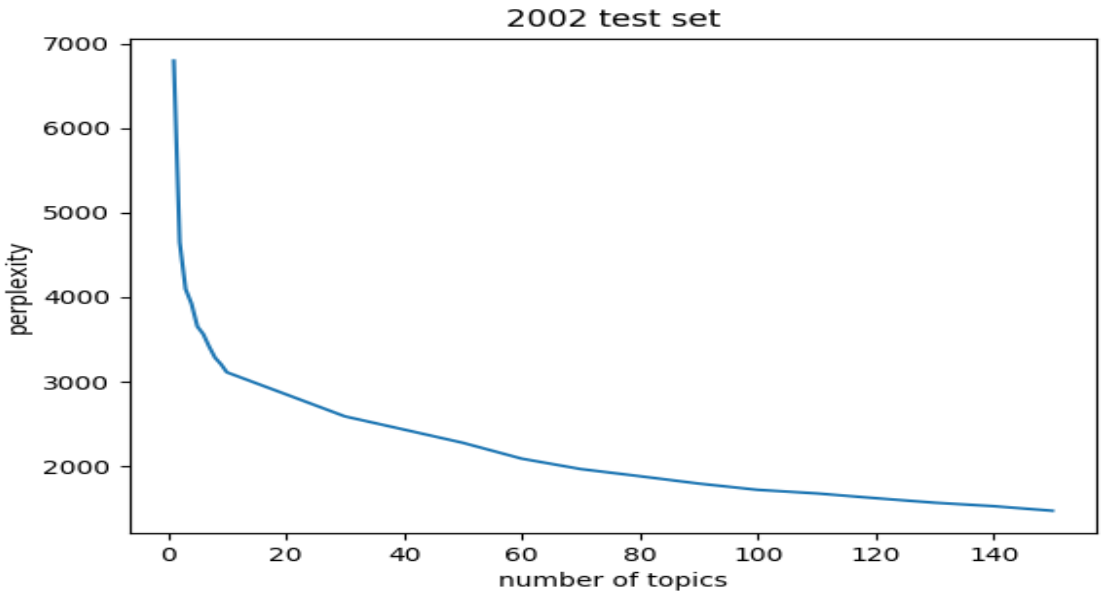

Рисунок 4. График перплесии на слоях, 2002г.

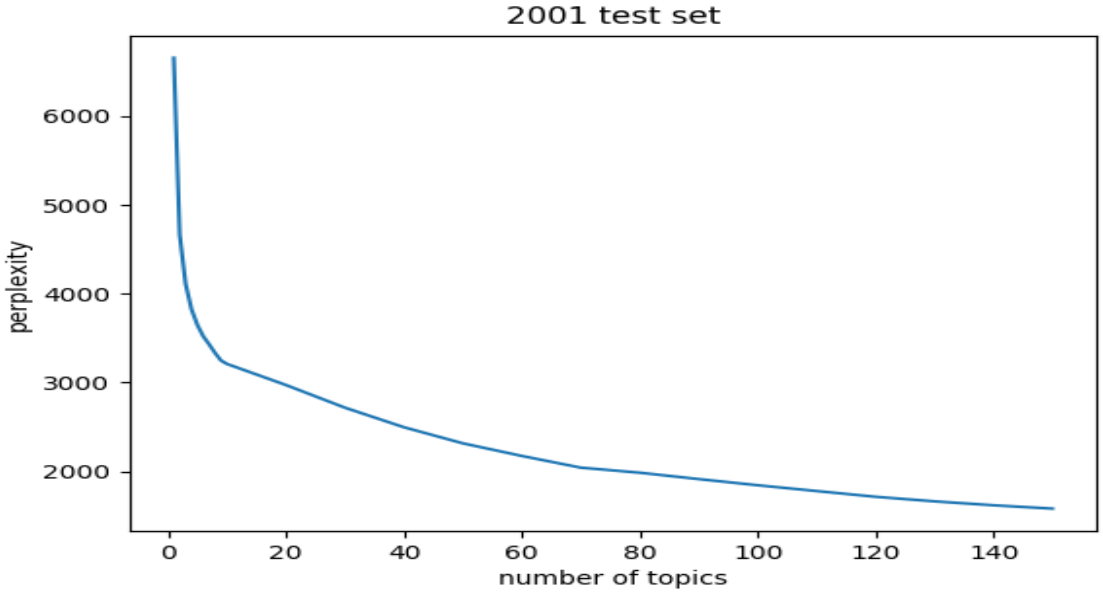

Рисунок 5. График перплексии на слоях, 2001г.

Как видно из графиков, начиная примерно с уровня 130 тем график перплексии сходится к прямой. В качестве оптимального числа тем было выбрано именно это значение.

Определение неинформативных тем. Для определения полезности темы была введена весовая функция для слов коллекции, представляющая

собой модифицированную версию TF-IDF меры. TF в данном случае высчитывалась не для одного документа, а для всей коллекции сразу:

$$TF - IDF'(t) = {}^{Nt}/_{N} \times \log_{10} {}^{D}/_{n_{dt}},$$

где Nt – количество слов t во всей коллекции, N – длина коллекции в словах, D – длина коллекции в документах, n_{dt} – количество документов, содержащих слово t. Таким образом, наибольший вес получили малоинформативные слова, присутствующие в большом количестве документов, и которые при этом являются достаточно частыми во всей коллекции. Вес темы T определялся по словам, значения вероятностей которых лежат правее значения третьего квартиля распределения T. Иными словами, выбирались 25% наиболее вероятных слов темы. За вес темы принималась сумма весов выбранных слов. Получившиеся значения весов для каждой темы за 2001, 2002 и 2003 гг. представлены на рисунках 6, 7, 8.

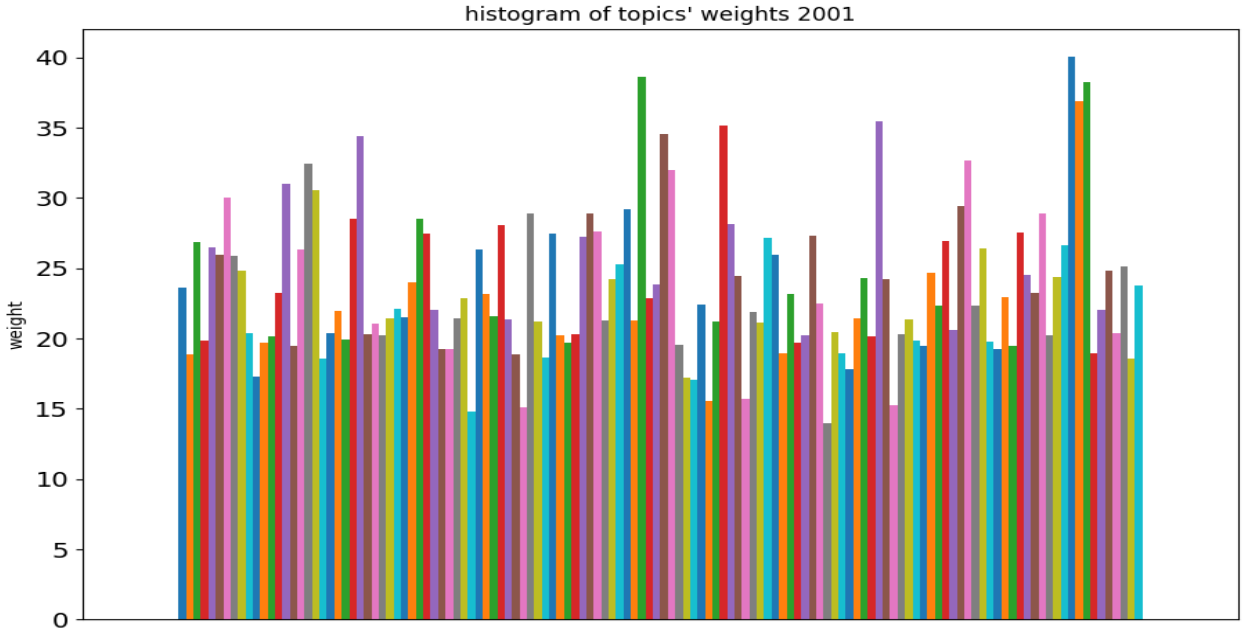

Рисунок 6. Гистограмма весов тем за 2001 г.

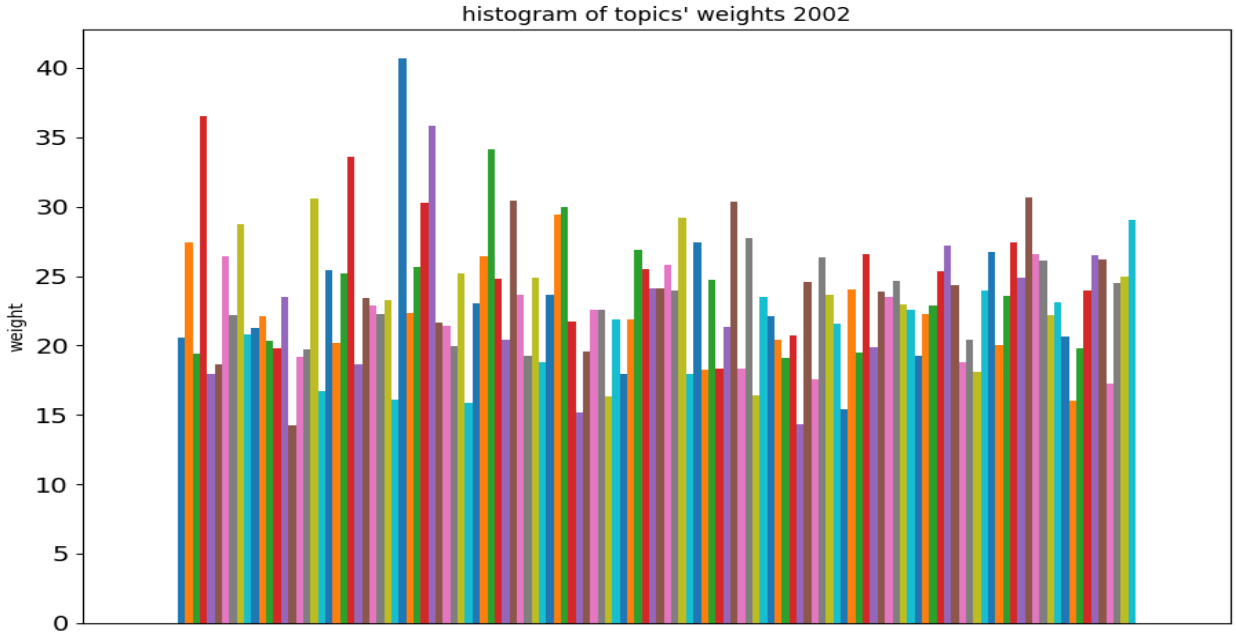

Рисунок 7. Гистограмма весов тем за 2002 г.

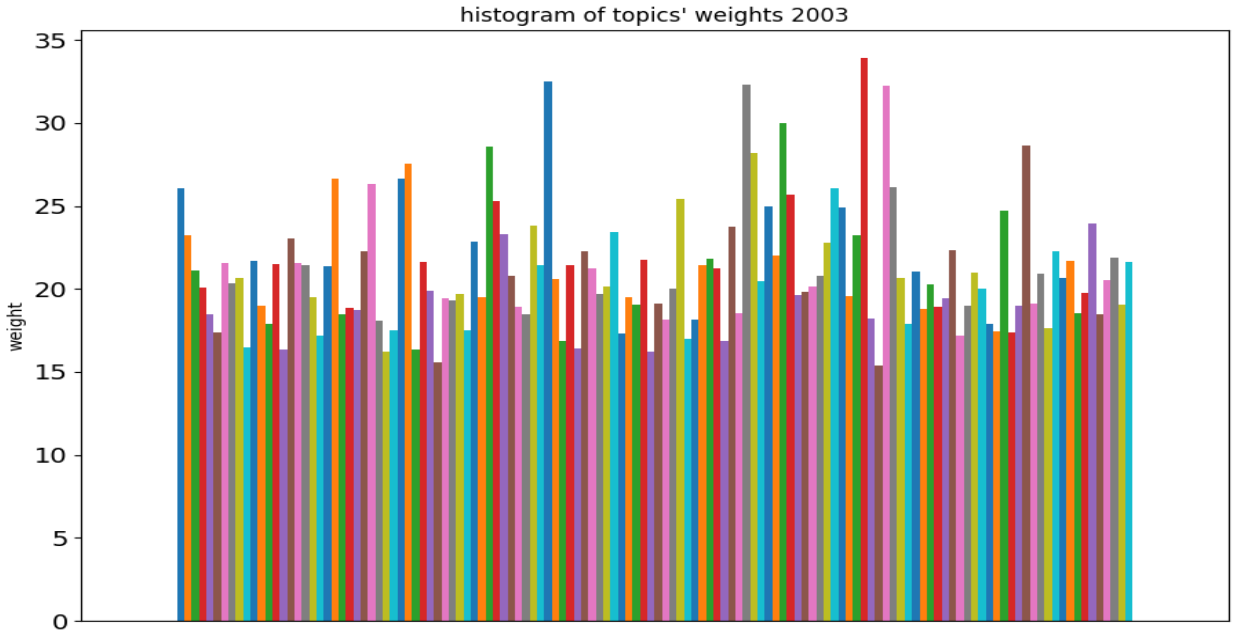

Рисунок 8. Гистограмма весов тем за 2003 г.

На приведенных гистограммах можно наблюдать немногочисленные выбросы. Было принято решение относить темы, соответствующие большим отклонениям от среднего, к неинформативным и в дальнейшей разработке не использовать.

Перетекание тем. Следующей задачей было определения близости двух тем, выделенных за разные года, для получения представления о перетекании тем. В ходе реализации данного этапа использовались различные подходы.

В качестве первого решения для каждой пары тем разных годов была вычислена дивергенция Йенсена-Шеннона как мера расстояния между двумя распределениями. Однако значения расстояний, полученные с ее использованием, оказались неудовлетворительными. На рисунке 9 можно видеть пары ближайших тем за 2002 и 2003 гг., а также значения расстояний между ними. Средняя величина оказалась равна 0.75, в то время как дивергенция Йенсена-Шеннона принимает значения от 0 до 1. При этом, нулевое значение можно получить только в случае совпадающих распределений.

topic2002	->JSD	-> topic2003
topic_0	-> 0.783578143181	-> topic_115
topic_1	-> 0.78949392527	-> topic_119
topic_2	-> 0.780505241019	-> topic_50
topic_3	-> 0.7932924207809999	-> topic_82
topic_4	-> 0.7976109856620001	-> topic_32
topic_5	-> 0.7261650257260001	-> topic_89
topic_6	-> 0.610961935224	-> topic_104
topic_7	-> 0.71070163076 3	-> topic_3
topic_8	-> 0.821110391347	-> topic_3
topic_9	-> 0.741703467482	-> topic_78
topic_10	-> 0.777741619658	-> topic_42
topic_11	-> 0.8004388847120001	-> topic_88
topic_12	-> 0.806663703544	-> topic_72
topic_13	-> 0.791867946648	-> topic_99
topic_14	-> 0.7848419072759999	-> topic_45
topic_15	-> 0.723200255493	-> topic_83
topic_16	-> 0.6303133744380001	-> topic_104
topic_17	-> 0.7086089649089999	-> topic_69
topic_18	-> 0.7135183974429999	-> topic_25
topic_19	-> 0.628619877644	-> topic_94
topic_20	-> 0.850959118823	-> topic_78

Рисунок 9. Расстояния между темами, выделенными за 2002 и 2003 гг.

topic2002	->JSD	-> topic2002-2003
topic_0	-> 0.7755494361699999	-> topic_3
topic_1	-> 0.680019035758	-> topic_79
topic_2	-> 0.7970716999230001	-> topic_37
topic_3	-> 0.653850657508	-> topic_113
topic_4	-> 0.7915497770450001	-> topic_126
topic_5	-> 0.785783537908	-> topic_45
topic_6	-> 0.725114598284	-> topic_43
topic_7	-> 0.762545392465	-> topic_25
topic_8	-> 0.6825243175	-> topic_47
topic_9	-> 0.7666817070849999	-> topic_47
topic_10	-> 0.731576929275	-> topic_28
topic_11	-> 0.803181037216	-> topic_102
topic_12	-> 0.485281965699	-> topic_72
topic_13	-> 0.7466566752420001	-> topic_43
topic_14	-> 0.7459839160580001	-> topic_45
topic_15	-> 0.6734807483640001	-> topic_54
topic_16	-> 0.5659390934049999	-> topic_97
topic_17	-> 0.7851807917	-> topic_65
topic_18	-> 0.7545519086239999	-> topic_6
topic_19	-> 0.748279381357	-> topic_129
topic_20	-> 0.652822624178	-> topic_65

Рисунок 10. Расстояния между темами, выделенными за 2002 и 2002+2003 гг.

В качестве второго решения был разработан следующий подход: на объединенном за два смежных года корпусе документов строилась hARTM модель, аналогичная описанным выше. Выделенные с помощью этой модели темы назовем темами-посредниками. В качестве меры расстояния по-прежнему используем дивергенцию Йенсена-Шеннона, однако для данной темы в качестве ближайшей к ней определим тему, ближайшую к ближайшей теме-посреднику данной темы (см. рис. 11). При использовании данного

подхода среднее значение расстояния между темой и темой-подтемой оказалось равным 0.7, примерные значения для 2002 и 2003 гг. можно увидеть на рис. 10. Таким образом, удалось добиться улучшения результатов при определении ближайших тем в разных годах.

Рисунок 11. Схема определения пары ближайших тем за разные года.

Прогнозирование. На данном этапе у нас имеется информация о дате публикации документов коллекции, рейтинги документов, а также информация о перетекании тем из года в год. Для того, чтобы отследить динамику рейтинга темы, принимаются следующие допущения:

1. Рейтинг документа отождествляется с рейтингом наиболее представленной в документе темы.
2. Дата выставления рейтинга близка к дате публикации документа (с точностью до месяца).

На основании данных предположений можно построить временной ряд с рейтингом каждой темы.

В ходе построения такого ряда выяснилось, что, хотя коллекция содержит более 1 млн. троек <пользователь, книга, оценка>, около 700 000 из них являются нулевым полями и не несут информации о рейтинге книги. В следствие этого при подсчете рейтинга темы за каждый месяц некоторые месяцы оказались пустыми – в эти месяцы не публиковались (и, согласно допущению 2, не оценивались) книги, представленной данной темой. В таких случаях значение рейтинга линейно интерполировалось по значениям рейтинга в ближайших месяцах.

Таким образом был определен временной ряд, который насчитывает 36 временных отметок (месяцы с 2001 по 2003 гг.). Для обучения

прогнозирующей модели использовались первые 27 точек, остальные 9 – для прогнозирования.

Для прогнозирования применялась модель *ARIMA(p, d, q)*. Для работы с ней необходимо привести ряд к стационарному виду. Стандартным приемом в этом случае является дифференцирование ряда. Подробнее описано в [15]. Определить стационарность ряда можно с помощью теста Дики-Фуллера.

Параметр d выбирался как степень дифференцирования ряда.

Параметры p и q подбирались на основании графиков функций автокорреляции и частной автокорреляции: p равно номеру временного лага, в котором функция частной автокорреляции впервые пересекает доверительный интервал; q же равно номеру лага, в котором функция автокорреляции впервые пересекает доверительный интервал.

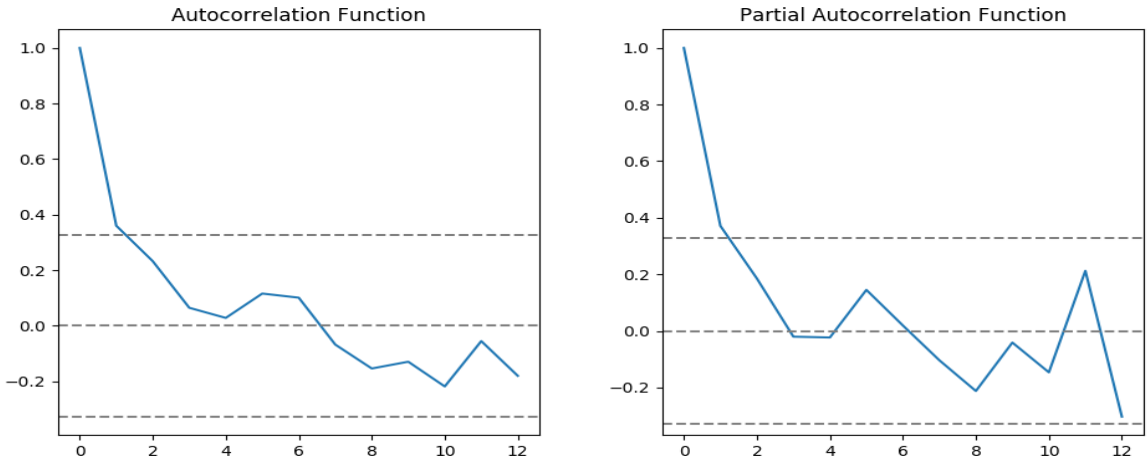

Рисунок 6. Графики функции автокорреляции и частной автокорреляции.

На рис. 9 представлены графики функций автокорреляции и частичной автокорреляции для одной из выделенных тем. На их основании были выбраны параметры модели $p = q = 1$. Результаты теста Дика-Фуллера на рис. 13.

```
Results of Dickey-Fuller Test:
Test Statistic                  -3.650892
p-value                          0.004859
#Lags Used                       0.000000
Number of Observations Used     27.000000
Critical Value (5%)             -2.976430
Critical Value (1%)             -3.699608
Critical Value (10%)            -2.627601
```

Рисунок 7. Тест Дики-Фуллера.

Значение статистики ниже критического значения для 5%, значит, с вероятностью 95% можно утверждать, что временной ряд стационарен и, следовательно, $d = 0$. После обучения модели с подобранными параметрами, был получен следующий прогноз (см. рис. 14). Значение среднеквадратичной ошибки на тестовом множестве получилось равным 0.826, что является неплохим показателем.

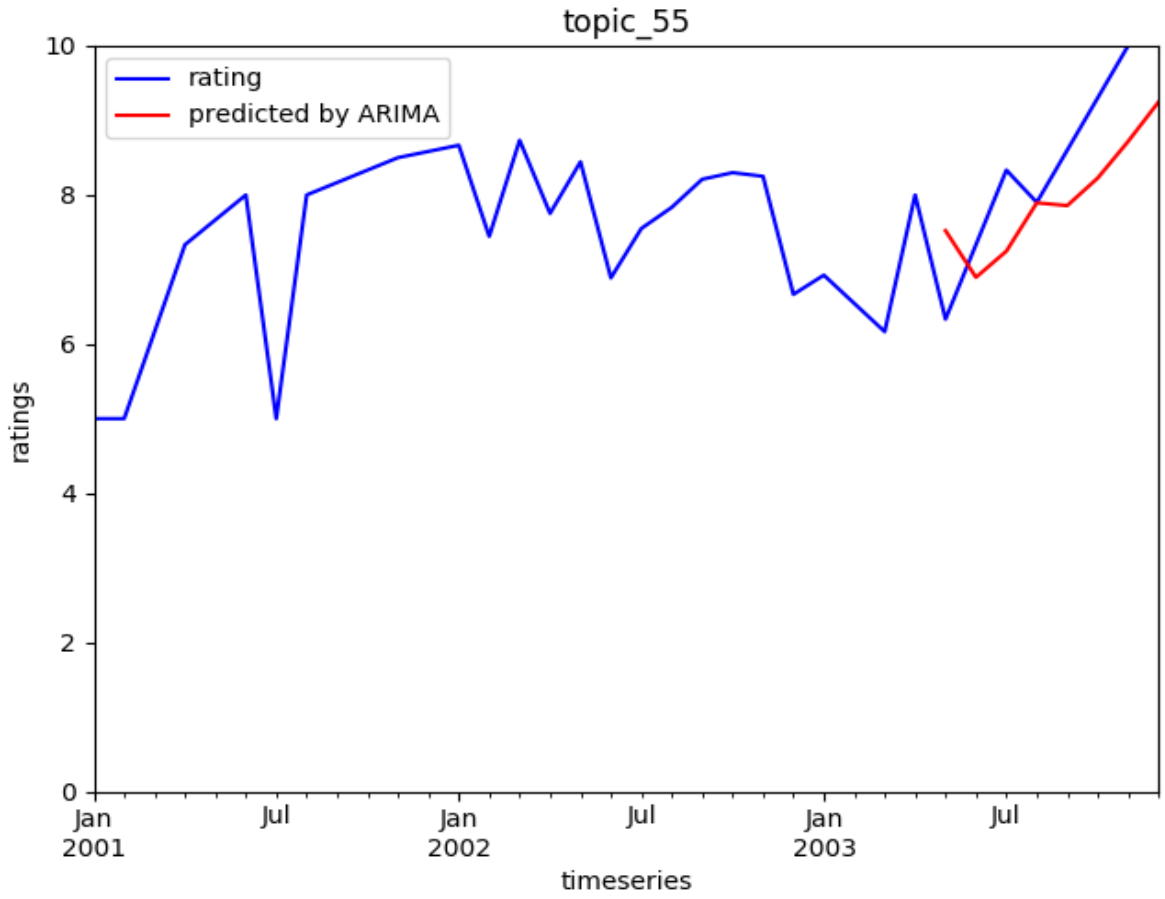

Рисунок 14. Прогноз ARIMA рейтинга 55 темы.

Заключение

В рамках данной работы описан ход решения задачи прогнозирования динамики рейтингов скрытых тем документов. В частности, выполнены следующие подзадачи:

1. определено оптимальное количество скрытых тем в текстовой коллекции;
2. выявлены скрытые темы в текстовой коллекции;
3. построена прогнозирующая модель для рейтингов выявленных тем.

Было продемонстрировано применение тематического моделирования для решения реальной прикладной задачи. В ходе работы был сделан ряд эвристических допущений, например, из-за специфики выбранной коллекции документов пришлось сделать предположение о совпадении даты оценивания книги с датой ее публикации. Несмотря на сделанные допущения, способ нахождения ближайших тем не был окончательно проработан и, безусловно, требует дополнительного анализа и корректировки в силу нетривиальности поставленной задачи. В дальнейшем следует использовать дополнительные источники информации для получения более точного представления о дате выставления рейтинга. Так же следует более полно исследовать вопрос фильтрации выделенных тем.

Список литературы

1. Коршунов Антон, Гомзин Андрей. Тематическое моделирование текстов на естественном языке // Труды Института системного программирования РАН, 2012. Т. 23. С. 215–244.
2. 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. P. 29–41.
3. 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. IEEE Computer Society, 2010. Vol. 1. P. 209–213.
4. 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. P. 1079–1088.
5. 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. P. 50–57.
6. Blei D. M., Ng A. Y., Jordan M. I. Latent Dirichlet allocation // Journal of Machine Learning Research, 2003.Vol. 3. P. 993–1022.
7. Wang Y. Distributed Gibbs sampling of latent dirichlet allocation: The gritty details, 2008.
8. К. В. Воронцов. Вероятностное тематическое моделирование, 2013. http://www.machinelearning.ru/wiki/images/2/22/Voron-2013-ptm.pdf
9. David M. Blei, Thomas Griffiths, Michael Jordan, Joshua Tenenbaum // Hierarchical topic models and the nested Chinese restaurant process. NIPS, 2003.
10. Chinese restaurant process on Wikipedia. https://en.wikipedia.org/wiki/Chinese_restaurant_process

11. К. В. Воронцов, А. И. Фрей, М. А. Апишев, А. А. Потапенко. Тематическое моделирование в BigARTM: теория, алгоритмы, приложения, 2015. http://www.machinelearning.ru/wiki/images/b/bc/Voron-2015-BigARTM.pdf
12. Kullback S., Leibler R.A. On information and sufficiency. // Annals of Mathematical Statistics, 1951 P. 79–86.
13. Jensen-Shannon divergence on Wikipedia. https://en.wikipedia.org/wiki/Jensen%E2%80%93Shannon_divergence
14. BigARTM. http://bigartm.org/
15. A comprehensive guide to create a Time Series Forecast. https://www.analyticsvidhya.com/blog/2016/02/time-series-forecasting-codes-python/
16. Dickey D. A., Fuller W. A. Distribution of the Estimators for Autoregressive Time Series with a Unit Root // Journal of the American Statistical Association, 1979. Vol. 74. P. 427–431.