МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
"МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
(ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ)"

ФАКУЛЬТЕТ ИННОВАЦИЙ И ВЫСОКИХ ТЕХНОЛОГИЙ
КАФЕДРА АНАЛИЗА ДАННЫХ

Выпускная квалификационная работа по направлению

01.03.02 «Прикладные математика и информатика»

НА ТЕМУ:

МУЛЬТИМОДАЛЬНЫЕ ТЕМАТИЧЕСКИЕ МОДЕЛИ СТАТЕЙ КОЛЛЕКТИВНЫХ БЛОГОВ ДЛЯ РАЗВЕДОЧНОГО ПОИСКА

Студент _____ Янина А.О.

Научный руководитель д.ф-м.н _____ Воронцов К.В.

Зам. зав. кафедрой д.ф-м.н, профессор _____ Бунина Е.И.

МОСКВА, 2016

Содержание

Часть 1

Введение

Выбор релевантного материала из большого корпуса статей — распространенная проблема информационного поиска и машинного обучения. Обычно задачи информационного поиска делят на два больших класса: поиск по четко сформулированному лаконичному запросу (known-item search) и разведочный поиск (exploratory search).

Традиционные информационно-поисковые техники фокусируются в основном на поиске по четкому короткому запросу. Разведочный поиск не так хорошо изучен. Такой поиск применяется, когда информационная потребность пользователя строго не формализуется, например, когда человеку необходимо быстро разобраться в смежной профессиональной сфере деятельности или узнать последние достижения науки в какой-то области. В таком случае обычного полнотекстового поиска может быть недостаточно: один-два запроса в поисковую систему не позволяют получить исчерпывающий ответ. Решение задач разведочного поиска требует от человека много сил и времени, автоматизация этого процесса также является непростым заданием. Этому способствует ряд причин: цели разведочного поиска зачастую неточно сформулированы, у пользователя может быть недостаточно знаний в предметной области, чтобы задать четкий запрос, поисковые мотивы и требования к найденным документам могут меняться в процессе поиска.

Таким образом, разведочный поиск — парадигма поиска, в которой в качестве поискового запроса задается не четко сформулированный текстовый запрос, а только обозначается тема, возможно, достаточно широко [1]. Формально в таком случае запросом может быть текстовый документ по заданной теме, коллекция документов или текстовое описание интересующей области знаний. В отличие

от полнотекстового поиска, разведочный поиск помогает решать более широкий спектр задач: систематизация данных, исследование, изучение нового материала, сравнительный анализ, планирование, синтез информации из разных источников и т.д. Подобных поисковых систем на данный момент не существует.

Целью данной работы является разработка нового метода решения задач разведочного поиска. Разрабатываемый метод предлагается строить на основе подходов, использующихся в тематическом моделировании. В рамках поставленной цели необходимо решить следующие задачи:

1. Построить тематическую модель для рекомендации статей и разведочного поиска.
2. Показать, что учет дополнительных модальностей улучшает качество разведочного поиска и рекомендательного ранжирования статей.
3. Предложить методику оценивания качества разведочного поиска.
4. Разработать технологию тематического поиска для решения задач разведочного поиска и показать его преимущество перед полнотекстовым поиском.

Тематическое моделирование — способ построения модели коллекции текстовых документов, которая определяет, к каким темам относится каждый из документов и какие слова или словосочетания образуют каждую тему [2]. Вероятностная тематическая модель (probabilistic topic model) коллекции текстовых документов описывает каждую тему дискретным распределением на множестве терминов, каждый документ — дискретным распределением на множестве тем. Задача построения вероятностной тематической модели является некорректно поставленной, так как искомое матричное разложение на произведение двух стохастических матриц определено не единственным способом. Данную проблему можно решить путем введения дополнительных ограничений на модель. В данном исследовании предлагается использовать аддитивную регуляризацию тематических моделей (APTM) [3]. APTM позволяет строить модели, удовлетворяющие нескольким ограничениям одновременно .

Основная часть работы по построению тематической модели заключалась в поиске оптимального набора регуляризаторов. Поиск регуляризаторов для построения тематической модели, улучшающих интерпретируемость тем — открытая

задача. В рамках данной работы была продумана стратегия регуляризации: найдены регуляризаторы для построения модели коллективного блога, подобраны траектории регуляризации, определены временные характеристики работы регуляризаторов относительно друг друга.

С точки зрения практического исполнения исследование демонстрирует возможности подхода ARTM на примере библиотеки с открытым кодом BigARTM. В ней реализованы регуляризаторы для построения тематической модели и метрики качества, необходимые для вычислительного эксперимента. С помощью средств BigARTM были построены тематические модели коллективного блога Хабрахабр.ру с разным набором модальностей. Затем было проведено сравнение полученных моделей по ряду признаков.

С задачами разведочного поиска плотно связаны задачи рекомендательного ранжирования статей. Оба класса задач можно решать при помощи тематического моделирования. В данной работе были изучены методы выдачи рекомендаций пользователям (см. 4) и предложены рекомендательные модели, использующие информацию о тематических профилях пользователей (см. 5). Изученные методы для построения тематических рекомендаций были использованы при построении тематической разведочной поисковой системы (см. 6.1). Затем с помощью разработанной методики оценивания качества разведочного поиска было проведено сравнение и выявлены преимущества разработанного тематического поисковика перед обычным полнотекстовым поиском (6.3).

Часть 2

Разведочный поиск информации

2.1 Концепция

Основная задача информационного поиска — помочь пользователю найти ответы на интересующие его вопросы. Типичный сценарий взаимодействия пользователя и поисковой системы выглядит так: пользователь формулирует короткий текстовый запрос, получает список релевантных отранжированных документов, затем анализирует результаты выдачи и если выдача его не устраивает, переформулирует запрос, чтобы выделить более специфичную тему или направить поиск в другом направлении. Таких поисковых итераций может быть много.

Исследования показывают [4], что пользователи используют длинные специализированные запросы менее, чем в половине случаев. Даже тогдв, когда пользователи точно знают, что нужно найти, они избегают использования длинных или сложных запросов, так как по опыту знают, что по простым и понятным запросам шанс найти релевантную информацию значительно выше. Это приводит к тому, что информационная потребность пользователя остается неточно сформулированной, детальная проработка запроса отсутствует. В результате, пользователи вынуждены изучать предметную область постепенно, запрос за запросом сужая круг поиска и приближаясь к необходимому ответу [4].

Зачастую информационную потребность сложно сформулировать в виде короткого списка ключевых слов. Например, если пользователю необходимо узнать последние достижения в конкретной научной области или быстро разобраться в смежной области знаний, одного-двух запросов в Google или Yandex может быть недостаточно для получения исчерпывающего ответа. Данная проблема

сохраняется, если поисковый запрос плохо сформулирован или пользователь не знаком с предметной областью (не знает ключевых слов, понятий, не разбирается в теме поиска).

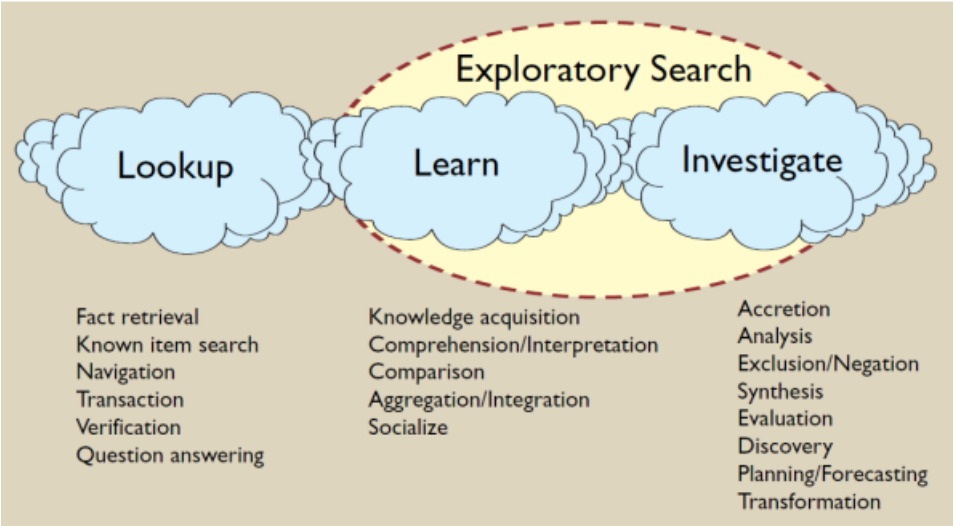

Рис. 2.1: Иллюстрация концепции разведочного поиска

Описанная парадигма подходит под определение разведочного поиска. Разведочный поиск — парадигма поиска, в которой в качестве поискового запроса задается не четко сформулированный текстовый запрос, а только обозначается тема, возможно, достаточно широко [1]. Таким образом, мы сталкиваемся с необходимостью помочь пользователю решить его поисковую задачу, когда он сам точно не знает, что ищет. Разведочный поиск представляет собой переход от аналитического подхода к нахождению соответствий между запросами и документами и полностью автоматизированного полнотекстового поиска по коротким запросам (запросам в поисковых системах).

Обычно разведочный поиск включает в себя несколько итераций поисковых запросов, а также зачастую несколько поисковых сессий. Разведочный поиск может требовать объединения усилий нескольких человек для достижения необходимого результата. Цель разведочного поиска — не только найти информацию, точно соответствующую запросу, но и понять, изучить тему, разобраться в новом вопросе. Таким образом, акцент делается не на поиске информации, а на саморазвитии ищущего.

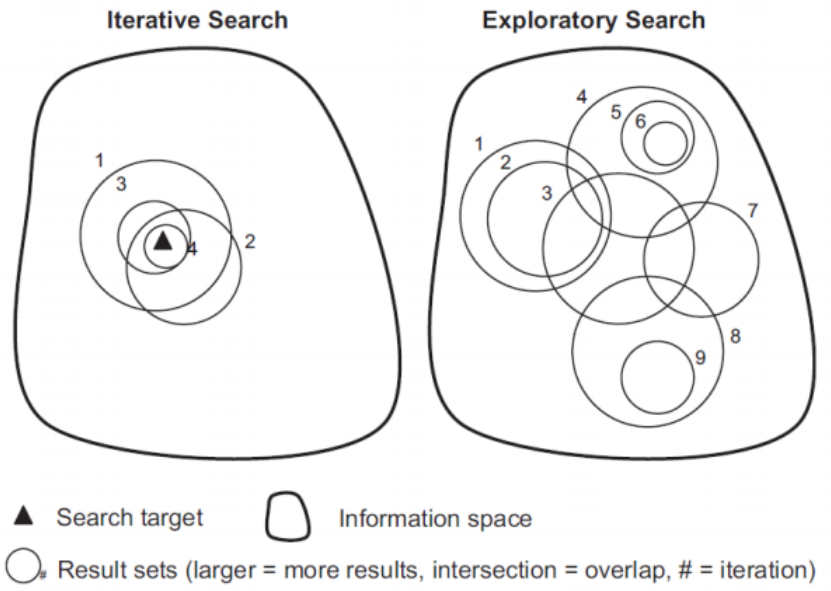

Рис. 2.2: Сравнение поиска по четкому запросу и разведочного поиска

При решении задач разведочного поиска нам могут помочь различные инструменты [1], например:

- Список тем и подтем выбранного фрагмента текста.
- Дорожная карта. Она представляет собой визуализацию кластеризации релевантных документов. Каждый кластер — это группа тематически схожих документов. Элементами кластеров являются документы (или их фрагменты).
- Тематическая иерархия. Она демонстрирует структуру предметной области. В целом, является визуализацией списка тем и подтем.
- Динамика тем. Показывает эволюцию предметной области во времени.
- Тематическая сегментация документа запроса.
- Суммаризация документа запроса.

2.2 Обзор существующих подходов

На настоящий момент разведочный поиск изучен менее подробно, чем поиск определенных документов по четко сформулированному запросу. Рассмотрим несколько существующих подходов к решению задач разведочного поиска.

В 2006 году Г.Марчионини в своей статье [5] изложил основную концепцию разведочного поиска и предложил интерактивную систему, поддерживающую разведочный поиск по государственным статистическим веб-сайтам, названую Relation Browser (RB). Эта система предназначена для поиска информации по большим базам данных, когда в качестве ответа на запрос пользователь ожидает увидеть не один-два результата, а целые коллекции документов на запрашиваемую тему для дальнейшего изучения полученного корпуса. RB предоставляет пользователю ограниченный набор параметров для поиска: тема, дата и время, формат данных. Для каждого из этих параметров можно выбрать одно из небольшого количества взможных значений. Предполагается, что Relation Browser можно использовать после поиска по ключевым словам в поисковой системе. В [5] показана эффективность этой системы по сравнению с использованием только обычного полнотекстового поиска.

В 2012 году С.Голденберг в своей диссертации предложил две системы, которые помогают пользователям решать задачи разведочного поиска [6]. Подход, предложенный в работе, группирует документы на основе анализа их метаинформации и визуализирует связи между документами с помощью node-link диаграммы. Этот алгоритм взвешивает информационные потребности пользователя и пытается сузить область поиска.

В 2014 году на конференции CIKM был представлен инструмент для упрощения поиска по Википедии и Yahoo Answers под названием DEESSE [7]. Эта система представляет информацию в виде графа, где для каждой ноды указана дополнительная информация: тема документа, категория (каждая статья относится к одной из нескольких больших групп), качество статьи и ее достоверность. Вершины графа соединяются исходя из их тематической схожести. Для обработки пользовательских запросов используются предподсчитанные тематические кластера: сначала ищется наиболее близкий к запросу кластер, затем внутри него отбираются наиболее релевантные документы. Создатели DEESSE отказались от привычной для поисковых систем организации выдачи — списка отранжированных по релевантности

документов. Вместо этого выдача отображается в виде тематических кластеров (bundles), что упрощает поиск интересующей информации среди найденных документов. DEESSE — мультиязычная система: она поддерживает английский и испанский языки.

На данный момент не существует широко распространенной удобной системы для решения задач разведочного поиска, поддерживающей различные источники информации и широкий спектр языков. Существуют решения для смежных задач. Так, в [8] предлагается прототип системы для организации документов в порядке, удобном для чтения и изучения. Статьи группируются в виде дерева: тексты более общей тематики находятся в корне, листья дерева — статьи узкой направленности. Такое представление информации взамен стандартной выдачи поисковика помогает более широко представить себе исследуемую область и, как следствие, решать задачи разведочного поиска быстрее и эффективнее.

Другой инструмент для упрощения процесса разведочного поиска — это системы закладок. Они помогают группировать тексты, сортировать статьи и упорядочивать найденные данные. В [9] построен теговый поисковый сервис, который объединяет в себе идеи разведочного поиска и системы закладок. В этой работе применяется коллаборативная фильтрация для рекомендации тегов пользователям. Далее построенные пользовательские профили из разных интернет-сервисов интегрируются в одну систему, которая помогает пользователям решать задачи поиска и систематизации информации путем рекомендации им статей с тегами, соответствующими их предпочтениям и интересам.

Описанные в обзоре результаты помогают решать близкие к тематическому поиску задачи. Однако комплексных систем тематического поиска по корпусам статей коллективных блогов, позволяющих решать задачи разведочного поиска, на данный момент не существует.

Часть 3

Тематическое моделирование

3.1 Применение тематического моделирования в разведочном поиске

Рассмотрим задачу разведочного поиска. На входе у нас есть текстовая коллекция и некоторый запрос. Мы знаем, какие термины и как часто встречаются в документах из коллекции. Эта информация позволяет узнать, к каким темам относится каждый документ и какими терминами образована каждая тема [10].

Системы полнотекстового поиска позволяют находить документы по словам, в разведочном поиске можно делать то же самое, только вместо слов использовать темы. В такой постановке задачи можно взять в качестве запроса несколько документов и совокупность их тем будет аналогом коротких запросов, которые используются для поиска в Yandex или Google. Таким образом, мы можем строить поисковые системы для разведочного поиска. В этой концепции можно использовать и инвертированные индексы: сначала из документов и запросов выделить темы, построить индекс, а затем искать темы из запроса по проиндексированной собранной коллекции.

Для того, чтобы осуществлять поиск по коллекции, необходимо знать тематический профиль каждого документа (вектор вероятностей, с которыми данный документ относится к данной теме). В решении этой задачи нам поможет вероятностное тематическое моделирование.

3.2 Постановка задачи вероятностного тематического моделирования

Пусть D — коллекция (множество текстовых документов), а W — словарь (множество всех употребляемых в документах из коллекции терминов). Терминами могут быть слова, биграммы, n-граммы, словосочетания. Каждый документ $d \in D$ представляет из себя последовательность терминов $(w_1, ..., w_{n_d}) \subset W$, где каждому термину ставится в соответствие число его вхождений n_{dw}. Таким образом, матрица частот F для текстовой коллекции D будет выглядеть так:

$$F = (f_{wd})_{W \times D} \tag{3.1}$$

$$f_{wd} = \frac{n_{dw}}{n_d} \tag{3.2}$$

Считаем, что существует конечное множество тем T, описывающее множество документов D. Коллекция документов рассматривается как случайная и независимая выборка троек $(w_i, d_i, t_i), i = 1..n$ из дискретного распределения $p(w, d, t)$ на конечном вероятностном пространстве $W \times D \times T$. При этом термины и документы — это наблюдаемые переменные, а тема документа является скрытой переменной. В данной модели используется гипотеза «мешка слов», согласно которой порядок, в котором термины встречаются в документе, не важен, а также гипотеза «мешка документов» (порядок документов в коллекции не важен).

Введем еще несколько понятий. $p(w|t)$ — вероятность встречания термина $w \in W$ в теме $t \in T$, $p(t|d)$ — вероятность встречания темы $t \in T$ в документе $d \in D$. Зная эти вероятности, получаем матрицу терминов тем Φ (3.3) и матрицу тем документов Θ (3.4):

$$\Phi = p(w|t)_{W \times T} \tag{3.3}$$

$$\Theta = p(t|d)_{T \times D} \tag{3.4}$$

Таким образом, задача построения тематической модели формулируется так: по заданной коллекции D найти множество тем T и оценить параметры модели $\phi_{wt} = p(w|t)$ и $\theta_{td} = p(t|d)$. Получается, что задача сводится к поиску матричного разложения заданной матрицы частот в виде произведения неизвестных матриц терминов тем (3.3) и тем документов (3.4):

$$F \approx \Theta_{W \times T} \times \Theta_{T \times D} \tag{3.5}$$

3.3 Вероятностный латентный семантический анализ

В вероятностном латентном семантическом анализе (PLSA) [11] для оценки матриц Φ и Θ предлагается максимизировать логарифм правдоподобия (плотности распределения) выборки при ограничениях неотрицательности и нормировки столбцов этих матриц(3.6):

$$\begin{gathered} L(\Phi,\Theta) \to \max_{\Phi,\Theta} \\ L(\Phi,\Theta) = ln \prod_{d\in D}\prod_{w\in d} p(w|d)^{n_{dw}} = \sum_{d\in D}\sum_{w\in d} n_{dw} ln \sum_{t\in T} \phi_{wt}\theta_{td} \\ \sum_{w\in W} \phi_{wt} = 1, \quad \phi_{wt} \geq 0, \quad \sum_{t\in T} \theta_{td} = 1, \quad \theta_{td} \geq 0. \end{gathered} \tag{3.6}$$

Поставленная задача решается с помощью итерационного EM-алгоритма. На E-шаге алгоритм по текущим значениям, вычисляет условные вероятности всех тем для каждой пары термин-документ:

$$p(t|d,w) = \frac{p(w|t)p(t|d)}{p(w|d)} \tag{3.7}$$

На M-шаге пересчитываются новые приближения параметров ϕ_{wt} и θ_{td} (3.9):

$$n_{dwt} \approx n_{dw} \cdot p(t|d,w) \tag{3.8}$$

$$\begin{gathered} \phi_{wt} = \frac{n_{wt}}{n_t} \qquad \theta_{td} = \frac{n_{dt}}{n_d} \\ n_{wt} = \sum_{d\in D} n_{dwt} \qquad n_t = \sum_{w\in W} n_{wt} \\ n_{dt} = \sum_{w\in d} n_{dwt} \qquad n_d = \sum_{t\in T} n_{dt} \end{gathered} \tag{3.9}$$

Начальные приближения ϕ_t и θ_d можно задавать нормированными случайными векторами из равномерного распределения. Кроме того, можно пройти по всей коллекции, выбрать для каждой пары (d,w) случайную тему и вычислить оценки аналогично 3.9: $\phi_{wt} = \frac{n_{wt}}{n_t}$, $\theta_{td} = \frac{n_{dt}}{n_d}$.

3.4 Аддитивная регуляризация тематических моделей

В общем виде задача тематического моделирования имеет бесконечно много решений. Если $F = \Phi\Theta$ — решение задачи, то для всех невырожденных матриц S, при которых матрицы $\Theta' = S^{-1}\Theta$ и $\Phi' = \Phi S$ являются стохастическими, $F = (\Phi S)(S^{-1}\Theta)$ также является решением. Такие задачи называют некорректно поставленными. Данную проблему можно решить с помощью регуляризации (добавлении к логарифму правдоподобия штрафного слагаемого, которая сужает множество решений). В [2] предлагается воспользоваться методом аддитивной регуляризации.

Допустим, что наряду с правдоподобием L требуется максимизировать еще n критериев $R_i(\Phi, \Theta)$, $i = 1, 2, \ldots, n$, называемых регуляризаторами. В байесовских методах обучения тематических моделей [11, 12, 13] регуляризатор $R(\Phi, \Theta)$ интерпретируется как логарифм априорного распределения, а оптимизационная задача соответствует принципу максимума апостериорной вероятности. Для практических задач необходимы сложные модели, совмещающие большое число дополнительных критериев-регуляризаторов. Байесовский вывод оказывается слишком громоздким для совмещения в одной модели более двух-трех регуляризаторов. Предлагаемая в [2] теория аддитивной регуляризации тематических моделей (APTM) позволяет решить эту проблему. В APTM регуляризатор не обязан иметь вероятностную интерпретацию. Таким образом, при аддитивной регуляризации тематических моделей мы максимизируем взвешенную сумму критириев-регуляризаторов $R_i(\Phi, \Theta)$ с логарифмом правдоподобия $L(\Phi, \Theta)$ (3.10):

$$\begin{gathered} L(\Phi, \Theta) + \sum_{i=1}^{r} \tau_i R_i(\Phi, \Theta) \to \max_{\Phi, \Theta} \\ \sum_{w \in W} \phi_{wt} = 1, \quad \phi_{wt} \geq 0, \quad \sum_{t \in T} \theta_{td} = 1, \quad \theta_{td} \geq 0. \end{gathered} \tag{3.10}$$

Здесь τ_i — неотрицательный коэффициент регуляризации.

Эту задачу можно решать аналогично 3.6 с помощью EM-алгоритма, немного видоизменив формулы M-шага:

$$\phi_{wt} = \frac{n_{wt}}{n_t} + \phi_{wt}\frac{\partial R(\Phi,\Theta)}{\partial \phi_{wt}}$$
$$\theta_{td} = \frac{n_{dt}}{n_d} + \theta_{td}\frac{\partial R(\Phi,\Theta)}{\partial \theta_{td}} \tag{3.11}$$

В формулах 3.11:

$$n_{wt} = \sum_{d\in D} n_{dwt} \qquad n_t = \sum_{w\in W} n_{wt}$$
$$n_{dt} = \sum_{w\in d} n_{dwt} \qquad n_d = \sum_{t\in T} n_{dt} \tag{3.12}$$

Добавление еще одного регуляризатора приводит к изменению формул для M-шага на аналогичную добавку. Это позволяет строить сложные модели с большим количеством регуляризаторов.

Теорема 3.1. *Пусть $R(\Phi,\Theta)$ непрерывно дифференцируема и точка (Φ,Θ) является локальным экстремумом задачи 3.10. Обозначим $p_{tdw} = p(t|d,w)$. Тогда для любой темы t и любого документа d выполняется система уравнений:*

$$p_{tdw} = norm_{t\in T}(\phi_{wt}\theta_{td}) \tag{3.13}$$

$$\phi_{wt} = norm_{w\in W}\left(n_{wt} + \phi_{wt}\frac{dR}{d\phi_w t}\right) \qquad n_{wt} = \sum_{d\in D} n_{dw}p_{tdw} \tag{3.14}$$

$$\theta_{td} = norm_{t\in T}\left(n_{td} + \theta_{td}\frac{dR}{d\theta_t d}\right) \qquad n_{td} = \sum_{w\in d} n_{dw}p_{tdw}, \tag{3.15}$$

где $normx_i = \dfrac{max(x_i,0)}{\sum\limits_{j\in I} max(x_j,0)}$, для всех $i \in I$.

Доказательство. Доказательство основано на применении условий Каруша-Куна-Такера и приводится в [2]. □

Данная теорема обобщается на случай мультимодальных тематических моделей, о которых рассказано в следующем разделе.

3.5 Мультимодальное тематическое моделирование

Многие тексты, помимо слов, содержат метаинформацию, например, имена авторов, теги, категории, тексты комментариев, отметки времени, лайки, метки классов и т.д.

В контексте построения тематической модели коллективного блога множество модальностей M состоит из 5 модальностей: лексическая составляющая (модальность терминов), автор, комментаторы, теги, категории (на Хабрахабр.ру они называются хабы). Вхождение элементов каждой модальности рассматривается точно так же, как вхождение терминов в текст.

Обобщим вероятностную тематическую модель на случай конечного числа модальностей M. Каждая модальность $m \in M$ имеет свой словарь W_m. Первая модальность соответствует терминам (словам, биграммам или словосочетаниям), остальные — метаданным.

Введем несколько понятий для задачи мультимодального тематического моделирования. $\forall m\ p(w,t)_m$ — вероятность встречания термина $w \in W^m$ в теме $t \in T$, $p(t,d)$ — вероятность встречания темы $t \in T$ в документе $d \in D$. Для каждой модальности $m \in M$ есть своя матрица терминов тем Φ_m:

$$\Phi_m = (p(w|t))_{W^m \times T} \quad \forall m \in M \tag{3.16}$$

Объединение матриц Φ_m, записанных в столбец, дает нам общую матрицу терминов тем модели:

$$\Phi = p(w|t)_{W \times T} \tag{3.17}$$

Матрица тем документов имеет такой же вид, как и для модели с единственной модальностью:

$$\Theta = (p(t|d))_{T \times D} \tag{3.18}$$

Запишем постановку задачи мультимодального тематического моделирования (3.19), представив логарифм правдоподобия в виде суммы по модальностям:

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

$$L(\Phi,\Theta)=\sum_{m\in M}L_m(\Phi_m,\Theta)=\sum_{m\in M}L(\Phi_m,\Theta)=$$

$$=\sum_{m\in M}ln\prod_{d\in D}\prod_{w\in d}p(w|d)^{n_{dw}}=\sum_{m\in M}\sum_{d\in D}\sum_{w\in d}n_{dw}ln\sum_{t\in T}\phi_{wt}\theta_{td} \tag{3.19}$$

$$\forall m\in M\sum_{w\in W^m}\phi_{wt}=1,\quad \phi_{wt}\geq 0,\quad \sum_{t\in T}\theta_{td}=1,\quad \theta_{td}\geq 0.$$

Первое слагаемое из суммы $\sum\limits_{m\in M}L_m(\Phi_m,\Theta)$ соответствует модальности терминов. Остальные слагаемые можно интерпретировать как регуляризаторы сооответствующих модальностей. Добавим в эту сумму коэффициенты регуляризации:

$$L(\Phi,\Theta)=\sum_{m\in M}\tau_m L_m(\Phi_m,\Theta)\to\max_{\Phi,\Theta} \tag{3.20}$$

Коэффициенты регуляризации позволяют сбалансировать модальности с учетом их важности.

Теорема 3.2. *Пусть $R(\Phi,\Theta)$ непрерывно дифференцируема и точка (Φ,Θ) является локальным экстремумом задачи 3.19. Обозначим $p_{tdw}=p(t|d,w)$. Тогда для любой темы t и любого документа d выполняется система уравнений:*

$$p_{tdw}=norm_{t\in T}(\phi_{wt}\theta_{td}) \tag{3.21}$$

$$\phi_{wt}=norm_{w\in W_m}\left(n_{wt}+\phi_{wt}\frac{dR}{d\phi_w t}\right)\qquad n_{wt}=\sum_{d\in D}\overline{n}_{dw}p_{tdw} \tag{3.22}$$

$$\theta_{td}=norm_{t\in T}\left(n_{td}+\theta_{td}\frac{dR}{d\theta_t d}\right)\qquad n_{td}=\sum_{w\in d}\overline{n}_{dw}p_{tdw}, \tag{3.23}$$

где $\overline{n}_{dw}=\tau_m\cdot n_{dw}\quad\forall m\in M$.

Доказательство. Доказательство аналогично доказательству 3.1 и приводится в [2]. □

Эта теорема — частный случай теоремы 1, когда модальность только одна. Таким образом, переход от одной модальности к нескольким сводится к двум дополнениям:

- Матрица Ф разбивается на блоки. Каждый из них соответствует своей модальности и нормируется отдельно.
- Исходные данные n_{dw} домножаются на коэффициенты регуляризации τ_m для каждой модальности.

Часть 4

Рекомендательное моделирование

4.1 Рекомендации как способ решения задач разведочного поиска

Глобальной целью разведочного поиска является необходимость разобраться в предметной области, получить новые знания, систематизировать информацию по выбранному вопросу. В то же время на разных этапах поиска возникают локальные цели, для достижения которых нужно дать ответы на вопросы:

1. Какие темы и области знаний затрагивает мой запрос?
2. Какова тематическая структура рассматриваемой предметной области?
3. Что еще известно по этим темам?
4. Какие еще документы (статьи, обзоры, лекции) по данной теме будут мне интересны и помогут разобраться в вопросе?

На эти вопросы, возникающие в процессе разведочного поиска, помогает ответить рекомендательное моделирование. Рекомендательные системы анализируют интересы пользователей и пытаются предсказать, что именно будет интересно конкретному пользователю в данный момент времени.

4.2 Основные методы рекомендательного моделирования

Рассмотрим два основных метода, используемых при создании рекомендательных систем — коллаборативная фильтрация [14, 15, 16] и контентно-основанные рекомендации [17]. Коллаборативные рекомендательные системы генерируют рекомендации на основе данных об оценках статей безотносительно к характеристикам конкретной статьи (т.е. ее темы и смысловой нагрузки). Прогнозы составляются индивидуально для каждого пользователя, хотя используемая информация собрана от многих участников. Делается допущение, что те, кто одинаково оценивали некоторые статьи в прошлом, будут давать похожие оценки другим статьям в будущем. Системы коллаборативной фильтрации находят пользователей, которые разделяют оценочные суждения прогнозируемого пользователя, а затем используют оценки этих пользователей для вычисления прогноза. При обработке данных о статьях коллективного блога Хабрахабр.ру будем считать, что пользователю понравилась статья, если он хотя бы один раз прокомментировал ее или является ее автором.

Контентно-основанные рекомендательные системы делают рекомендации на основе текстовой информации [17]. В данном методе необходимо использовать тематическое моделирование для определения тематики конкретной статьи. Из тематической модели получаем матрицу частот тем в документах Θ. По данным из этой матрицы можно оценить, какие темы превалируют в каждой статье. Пользователю рекомендуются статьи, похожие на те, которые он уже прочитал и оценил. Исходим из предположения, что пользователю, часто комментирующему статьи на определенную тему, будет интересно прочитать статьи схожей тематики.

Гибридный подход к коллаборативной фильтрации [18] объединяет в себе контентно-основанный подход и учет известных предпочтений группы пользователей для прогнозирования неизвестных предпочтений другого пользователя. Гибридное рекомендательное моделирование позволяет преодолеть ограничения контентно-ориентированного подхода и улучшить качество предсказаний. Этот подход также позволяет преодолеть проблему разреженности данных и потери информации.

В вычислительном эксперименте данной работы используются методы, основанные только на контентных и только на коллаборативных рекомендательных моделях, а также гибридный метод, который сочетает в себе коллаборативную

фильтрацию с контентно-основанными техниками.

Для того, чтобы оценить качество построенной рекомендательной модели, поделим пользователей на две группы: обучающая выборка и тестовая. Будем предсказывать предпочтения пользователей для тестовой выборки и сравнивать с реальными данными. Для этого используем условно-вероятностную модель. В рамках данного метода необходимо найти распределения тем в документах $p(t|d)$ (получаем эти вероятности из матрицы Θ) и распределения предпочтений пользователей из обучающей выборки $p(d|u)$. Тогда распределение тематических предпочтений пользователей из обучающей выборки находим, как показано в 4.1:

$$p(t|u) = \sum_{d \in D} p(t|d) \cdot p(d|u)_{train} \tag{4.1}$$

Распределения предпочтений пользователей из тестовой выборки вычисляем по формуле 4.2:

$$p(d|u)_{test} = \sum_{t \in T} p(d|t) \cdot p(t|u) \tag{4.2}$$

4.3 Измерение качества рекомендательных моделей

Для измерения качества рекомендательных моделей использовались метрики precision и recall. Precision — точность, доля интересных пользователю статей среди тех, что были рекомендованы к прочтению. Recall — полнота, доля статей, заинтересовавших пользователя среди всего списка релевантных статей по данному запросу. Введем обозначения:

- TP (true positive) — правильно рекомендованные статьи, которые должны понравиться пользователю,
- TN (true negative) — статьи, которые не нравятся пользователю и не были ему рекомендованы системой,
- FP (false positive) — ошибки первого рода, т.е. не интересные пользователю статьи, рекомендованные системой,
- FN (false negative) — ошибки второго рода, т.е. не рекомендованные пользователю статьи, которые оказались интересными.

Запишем формулы для метрик precision (P) и recall (R) с учетом введенных обозначений:

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{4.3}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{4.4}$$

Для больших корпусов документов precision и recall перестают быть информативными, так как рекомендация может содержать несколько тысяч релевантных статей. В этом случае для оценки качества лучше использовать Precision at k (Precision@k, P@k) и Recall at k (Recall@k, R@k) - метрики, применимые к первым наиболее популярным k документам. Так, $P@k$ — доля релевантных документов (тех, которые оказались интересны пользователю, рекомендации, которые он просмотрел) среди первых k документов из отранжированного списка рекомендаций. $R@k$ — доля релевантных документов из топ-k списка рекомендаций среди всех релевантных документов по данному запросу.

Введем обозначения для следующих метрик: average precision at k (ap@k) average recall at k (ar@k). Пусть m — количество документов, рекомендованных пользователю. Тогда:

$$ap@k = \frac{\sum_{n=1}^{k} p@n}{min(m,k)} \tag{4.5}$$

$$ar@k = \frac{\sum_{n=1}^{k} r@n}{min(m,k)} \tag{4.6}$$

Описанные выше метрики применяются для оценивания качества рекомендаций одному пользователю. Часто нужно посмотреть на среднее качество рекомендаций для нескольких пользователей, например, для аудитории сайта или блога. В таких случаях применяют метрику mean average precision at k (MAP@k). MAP@k для N пользователей вычисляется так:

$$MAP@k = \frac{\sum_{i=1}^{N} ap@k_i}{N} \tag{4.7}$$

Часть 5

Эксперимент по выдаче тематических рекомендаций

5.1 Исходные данные

Эксперимент проводился на основе данных о статьях коллективного блога Хабрахабр.ру. Проанализировано 132157 статей. Кроме текста статьи, доступна метаинформация о каждой статье. Она включает в себя:

- автор статьи
- ссылка на аккаунт автора
- количество просмотров статьи
- оценка статьи
- количество отметок «Мне нравится»
- текст комментариев с указанием их авторов
- теги
- хабы (категории)

Вся информация о статьях хранится в формате json. В рамках эксперимента использовались следующие метаданные: автор, комментарии, теги, хабы.

Предварительная обработка текстов включала в себя удаление слов, длина которых меньше двух букв (для исключения стоп-слов), удаление пунктуации,

приведение слов к нижнему регистру, замена буквы «ё» на буку «е» для единообразия написания слов, лемматизация при помощи морфологического анализатора pymorphy. Кроме того, для построения модальности «биграммы» в тематической модели, из текстов статей были выделены биграммы.

Всего на хабре 680368 пользователей, из них большая часть только читает статьи и ничего не пишет (605470 пользователей). Для эксперимента были выбраны 10000 наиболее активных пользователей.

5.2 Базовый эксперимент

Базовый эксперимент иллюстрирует простое базовое решение, с результатами которого будем сравнивать дальнейшие результаты эксперимента.

Для проведения эксперимента из метаданных к статьям была выделена информация, отвечающая предпочтениям пользователей. Данные для эксперимента включают в себя следующую информацию: id пользователя, список статей, которые пользователь комментировал с указанием времени, когда комментарий был оставлен. Данные были поделены на обучающую и тестовую выборки в отношении 1:1. Важно отметить, что разделение на обучающую и тестовую выборки производилось не по пользователям, а по статьям, таким образом, чтобы и тестовая, и обучающая выборки содержали информацию о предпочтениях каждого пользователя, отобранного для эксперимента. Данные делились по времени, когда был оставлен комментарий. Поскольку выбранная группа пользователей активна на сайте (часто оставляет комментарии), а период времени для эксперимента брался большим (данные за полгода), выборки получились достаточно сбалансированными (случаи, когда в обучающей выборке мало данных о конкретном пользователе, а в тестовую выборку попали почти все отметки об оставленных им комментариях, редки).

В итоге данные были представлены в виде таблицы из 3 столбцов: «id комментатора» «Номер статьи» «Количество комментариев». На основе этой информации с помощью библиотеки mrec было проведено рекомендательное моделирование. Модели были построены на основе трех подходов - «knn» (k ближайших соседей), взвешенной регуляризационной матричной факторизации (WRMF - Weighted Regularized Matrix Factorization) и самый простой подход, когда пользователю рекомендуются самые популярные статьи (в mrec этот

метод называется popularity-sum, сохраним это название). Для оценки качества рекомендаций использовались метрики качества Precision@k и Recall@k, $k \in [5, 10, 15, 20]$. Оказалось, что модель WRMF дает наилучшие оценки качества. Посмотрим на полученные значения метрик, усредненные по всем пользователям:

Table 5.1: Метрики качества precision@k и recall@k рекомендательных моделей из базового решения

Metric	KNN	Popularity-sum	WRMF
Precision@5	0.40	0.40	0.60
Precision@10	0.30	0.40	0.70
Precision@15	0.27	0.33	0.60
Precision@20	0.20	0.25	0.55
Recall@5	0.28	0.28	0.42
Recall@10	0.42	0.42	0.57
Recall@15	0.42	0.57	0.71
Recall@20	0.57	0.57	0.71

Метрики качества достаточно низкие. Посмотрим на $MAP@k$ (4.7):

Table 5.2: MAP@k рекомендательных моделей из базового решения

Metric	KNN	Popularity-sum	WRMF
MAP@5	0.42	0.45	0.60
MAP@10	0.37	0.42	0.61
MAP@15	0.31	0.41	0.62
MAP@20	0.27	0.42	0.56

MAP для всех моделей из базового решения достаточно низкий, что говорит о том, что использование только информации о предпочтениях пользователей и игнорирование тематики статей сказывается на качестве модели.

5.3 Построение тематических моделей

Эксперименты из базового решения показывают, что применение коллаборативной фильтрации в задаче рекомендации статей коллективного блога не дало высоких значений метрик качества. В базовом решении была задействована информации о предпочтениях пользователя (какой пользователь комментировал какие статьи), при этом остальная метаинформация и сам текст статьи никак не использовался. Выдвигается гипотеза, что использование информации о тематике статей поможет увеличить качество рекомендаций. Кроме того, необходимо проверить, что использование дополнительной информации о статьях (теги, хабы, авторство) улучшает качество рекомендательного моделирования. Проверим эти гипотезы на эксперименте.

Сначала нужно построить тематические модели, а затем на их основе выдавать рекомендации пользователям. Были построены три тематические модели, каждая из них использует разное количество информации о статьях. Первая из построенных моделей является контентно-основанной (в ней присутствует только одна модальность «слова», то есть никакая метаинформация не используется). Вторая модель построена на основе человеко-ориентированного (usage-based) подхода. В ней используется только информация о том, какие пользователи комментировали какие статьи. Соответственно эта модель также является унимодальной. Назовем эту модальность «комментарии». По количеству задействованной информации эта модель близка к моделям из базового решения, отличается только подход к задаче: мы будем использовать тематическое моделирование, а не KNN или WRMF. Наконец, третья модель является гибридной, то есть в ней объединен контентно-ориентированный подход и использование метаинформации. Эта модель — мультимодальная и включает в себя кроме основной модальности «слова» еще четыре дополнительные: автор, комментарии, теги и хабы.

Построение моделей было произведено при помощи библиотеки BigARTM. В каждую тематическую модель были включены три регуляризатора: декоррелирования распределений терминов в темах, сглаживания распределений терминов в темах разреживания распределений тем в документах. Регуляризатор декоррелирования распределений терминов в темах используется для повышения различности лексических ядер предметных тем. Кроме того, использовались фоновые темы для выделения слов общей лексики.

Для оценки качества модели были использованы следующие метрики качества: перплексия, разреженность терминов в темах для каждой из пяти модальностей: лексика, авторы, комментарии, теги, хабы, а также разреженность тем в документах, чистота и контрастность лексического ядра. Эти метрики служат промежуточными критериями качества, окончательная цель эксперимента: улучшить качество выдачи рекомендаций по сравнению с бейзлайн решением.

Важным параметром модели является количество тем (для каждой модели перед процессом обучения оно фиксируется). Поэтому до проведения основного эксперимента был проведен эксперимент по подбору оптимального количества тем для коллекции статей коллективного блога Хабра.

5.3.1 Эксперимент по определению оптимального количества тем

При поиске оптимального количества тем будем ориентироваться на два показателя. Во-первых, это метрики качества (перплексия, разреженность терминов в темах, разреженность тем в документах, чистота и контрастность лексического ядра), во-вторых, интерпретируемость получаемых тем. Эти показатели не всегда увеличиваются или уменьшаются согласованно друг с другом.

Сначала определим, как меняются метрики качества при изменении количества тем. Построим модели для $|T| = 10, 100, 500$ тем. Этот эксперимент показал, что увеличение количества тем улучшает значения метрик качества. На графиках ниже приведены значения метрик качества мультимодальной модели в зависимости от требуемого количества тем.

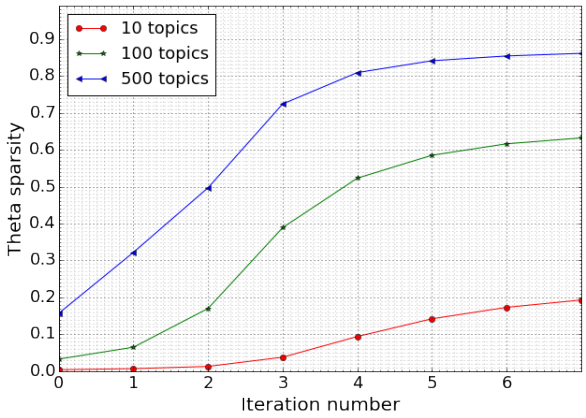

Рис. 5.1: Разреженность матрицы Θ

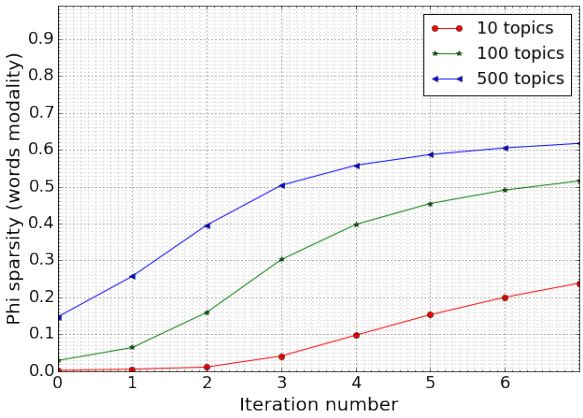

Рис. 5.2: Разреженность матрицы Φ

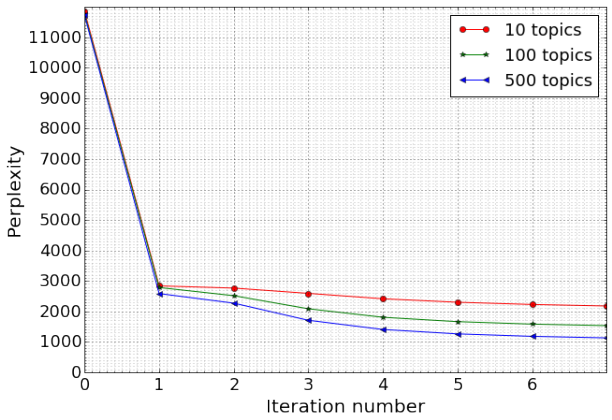

Рис. 5.3: Перплексия

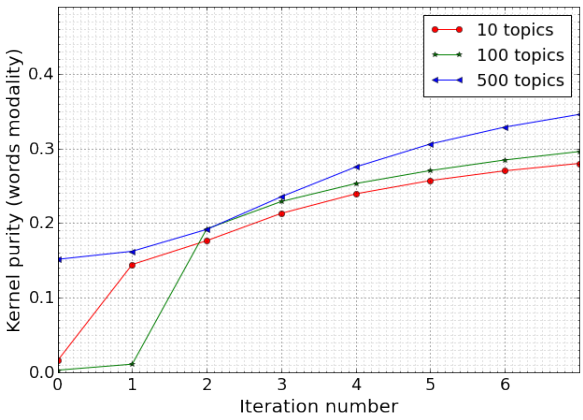

Рис. 5.4: Чистота лексического ядра

Дальнейшее увеличение количества тем ведет к тому, что темы становятся менее интерпретируемыми. Так, уже при 500 темах интерпретируемость полученных тем остается под вопросом. Проанализируем термины из 5 тем, выделенных с помощью модели на 500 темах:

Table 5.3: Термины для тем из мультимодальной модели ($|T| = 500$)

№	Термины
1	банк, банкомат, поступление, интеграция, адрес, распространение
2	шрифт, стиль, виджет, звонок, работа, написание
3	программирование, автоматизация, разработка, процесс, результаты
4	отладка, дебаг, код, флаг, уровень, медленный
5	дизайн, кнопка, итерация, постепенно, страница, дальний

Здесь нужно найти баланс между ростом метрик качества и получаемой интерпретируемостью тем. Анализируя интерпретируемость тем для моделей с $|T| = 100, 200, 300, 400, 500$ тем, приходим к выводу, что при количестве тем, равном 200, интерпретируемость тем остается высокой, при этом перплексия и остальные метрики имеют высокие значения.

Table 5.4: Термины для тем из мультимодальной модели ($|T| = 200$)

№	Термины
1	номер, оператор, звонок, абонент, связь, услуга, тариф, рубль
2	почта, адрес, платежный, кошелек, письмо, служба, счет
3	javascript, сервер, jquery, событие, клиент, ajax
4	изображение, распознавание, вершина, обработка, сжатие, граф
5	кластер, нагрузка, мониторинг, очередь, поток, соединение

Проведенной эксперимент показал, что оптимальное количество тем для коллекции статей коллективного блога Хабрахабр равно 200.

5.3.2 Сравнение мультимодальной и унимодальных тематических моделей

Будем строить три тематические модели (контентно-основанную с модальностью «слова», основанную на только на информации о предпочтениях пользователей с модальностью «комментарии» и мультимодальную). Количество тем равно 200, количество итераций — 8. При построении словаря для тематических моделей отрезаем «хвост»: 5% слов, которые употребляются очень часто (с большой долей вероятности это стоп-слова или слова общей лексики).

Рассмотрим метрики качества, полученные при построении трех моделей, и сравним результаты и качество построенных моделей. В таблице для мультимодальной модели модальности расположены в порядке: слова, пользователи, автор, теги, хабы. Например, Φ^1 — матрица Φ для модальности «слова», pur_t^5 — чистота лексического ядра модальности «хабы».

Table 5.5: Метрики качества, полученные при оценке качества контентно-основанной тематической модели

№	Перплексия	Разр. матрицыΦ	Разр. матрицыΘ	pur_t	con_t
1	10529.40	0.2361	0.334	0.0013	0.1609
2	1706.78	0.4589	0.683	0.2466	0.2304
3	1400.36	0.5617	0.854	0.6891	0.3304
4	1241.33	0.5982	0.859	0.7974	0.4047
5	1059.22	0.6184	0.855	0.8377	0.4480
6	919.83	0.6308	0.851	0.8608	0.4767
7	897.85	0.6390	0.848	0.8763	0.4969
8	684.17	0.6448	0.846	0.8871	0.5127

Table 5.6: Метрики качества, полученные при оценке качества тематической модели на основе человеко-ориентированного (usage-based) подхода

№	Перплексия	Разр. матрицыΦ	Разр. матрицыΘ	pur_t	con_t
1	10983.10	0.1209	0.877	0.0834	0.1608
2	295.35	0.2127	0.971	0.9591	0.5522
3	217.96	0.2152	0.974	0.9959	0.9518
4	163.40	0.2156	0.973	0.9966	0.9625
5	159.5092	0.2159	0.972	0.9972	0.9672
6	158.07	0.2161	0.971	0.9976	0.9703
7	157.08	0.2163	0.971	0.9979	0.9725
8	156.31	0.2165	0.970	0.9981	0.9740

Table 5.7: Перплексия и разреженности, полученные при оценке качества гибридной тематической модели с использованием модальностей

№	Перплексия	Разр. Φ^1	Разр. Φ^2	Разр. Φ^3	Разр. Φ^4	Разр. Φ^5	Разр. Θ
1	10880.03	0.1777	0.0198	0.0765	0.0443	0.0057	0.2281
2	2286.57	0.3904	0.0199	0.1359	0.0598	0.0058	0.4632
3	1865.53	0.5780	0.0198	0.1688	0.0598	0.0058	0.5731
4	1601.30	0.6143	0.0199	0.1812	0.0598	0.0058	0.7910
5	1362.46	0.6381	0.0199	0.1883	0.0599	0.0058	0.7930
6	1194.55	0.6596	0.0199	0.1930	0.0599	0.0058	0.7902
7	956.40	0.6840	0.0199	0.1963	0.0599	0.0058	0.7873
8	532.72	0.6943	0.0199	0.1986	0.0599	0.0058	0.7854

Table 5.8: Чистота и контрастность ядра, полученная при оценке качества гибридной тематической модели с использованием модальностей

№	pur_t^1	pur_t^2	pur_t^3	pur_t^4	pur_t^5	con_t^1	con_t^2	con_t^3	con_t^4	con_t^5
1	0.0006	0.9405	0.0023	0.1300	0.7821	0.1325	0.6901	0.1895	0.1492	0.3292
2	0.0631	0.8600	0.2336	0.1564	0.2000	0.1827	0.8600	0.2133	0.1162	0.2000
3	0.3783	0.8600	0.5956	0.1593	0.2000	0.3543	0.8600	0.2838	0.1457	0.2000
4	0.5493	0.8600	0.7052	0.1596	0.2000	0.4253	0.8600	0.3294	0.1515	0.2000
5	0.6286	0.8600	0.7566	0.1596	0.2000	0.4764	0.8600	0.3577	0.1545	0.2000
6	0.6756	0.8600	0.7883	0.1598	0.2000	0.5185	0.8600	0.3773	0.1561	0.2000
7	0.7082	0.8600	0.8102	0.1599	0.2000	0.5550	0.8600	0.3906	0.1567	0.2000
8	0.7315	0.8600	0.8262	0.1599	0.2000	0.6282	0.8600	0.4013	0.1574	0.2000

Для наглядности представим термины из 5 выделенных тем для контентно-основанной и гибридной моделей. Для модели, основанной только на информации о предпочтениях пользователей, можно просматривать списки пользователей, которые больше всего комментировали определенные статьи, но эта информация не очень наглядна (списки никнеймов пользователей не несут никакого интереса в контексте этой задачи).

Table 5.9: Термины из 5 тем, выделенных для контентно-основанной тематической модели

№ темы	Термины
1	чек, банкомат, qiwi, биткоин, visa, наличный, билайн, врач, касса
2	javascript, шрифт, стиль, css, иконка, chrome, виджет
3	письмо, услуга, домен, пароль, почта, оператор, регистрация, аккаунт, закон
4	вредоносный, антивирус, злоумышленник, отладчик, антивирусный, заразить
5	компиляция, макрос, статический, c++, наследование, флаг, паттерн

Table 5.10: Термины из 5 тем, выделенных для гибридной тематической модели с использованием модальностей

№ темы	Термины
1	Роскомнадзор, антипиратский закон, Ростелеком, вирусная реклама
2	письмо, клиент, почта, номер, адрес, услуга, пароль, отправить, звонок
3	книга, перевод, текст, заказчик, клиент, интерфейс, общение, идея, читать
4	изображение, точка, угол, кадр, пиксель, картинка, слой, распознавание
5	сервер, домен, хостинг, имя, файл, доступ, настройка, клиент, зона

Таким образом, систематизированные в приведенных выше таблицах термины позволяют в целом понять основную тематику и направленность каждой статьи. Отсюда делаем вывод, что интерпретируемость выделенных тем достаточно высока.

Результаты экспериментов по построению тематических моделей визуализированы. На приведенных ниже графиках представлены значения метрик качества, использованные для оценки трех построенных моделей.

Перед тем, как перейти к рекомендательному моделированию на основе построенных моделей, проведем анализ тематических моделей по промежуточным критериям (интерпретируемость тем и метрики качества). Анализ метрик качества для трех моделей показывает, что гибридная модель дает самое высокое качество среди трех построенных моделей, но не по всем критериям. Контрастность и

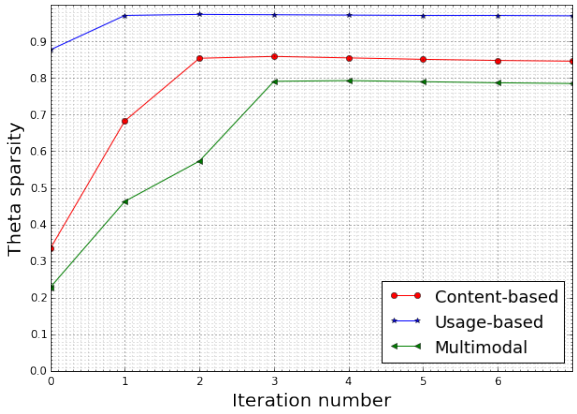

Рис. 5.5: Разреженность матрицы Θ

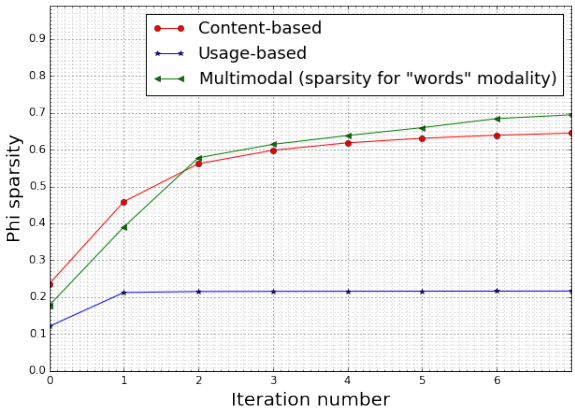

Рис. 5.6: Разреженность матрицы Φ

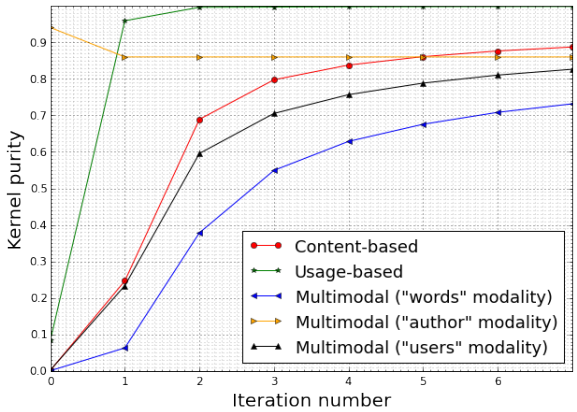

Рис. 5.7: Чистота лексического ядра

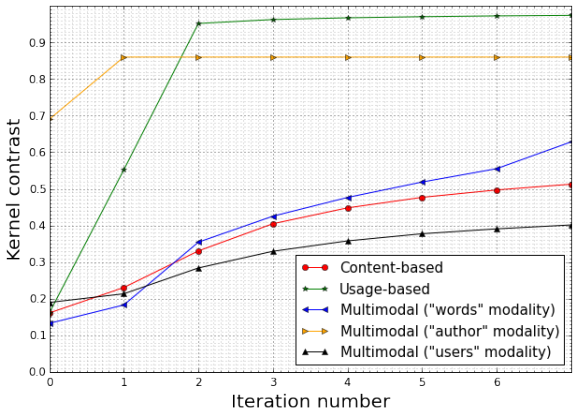

Рис. 5.8: Контрастность лексического ядра

разреженность матрицы Φ для мультимодальноной модели оказалась выше, чем для контентно-основанной модели и модели с одной модальностью «пользователи». Однако перплексия для мультимодальной модели выше перплексий унимодальных моделей, а разреженность матрицы Θ и чистота ядра немного ниже аналогичных характеристик первых двух моделей. Здесь нужно больше полагаться на интерпретируемость тем, кроме того, конечной целью эксперимента является повышение качества выдачи рекомендаций. Небольшое отставание мультимодальной модели по некоторым метрикам качества необязательно означает, что ее использование при дальнейшем рекомендательном моделировании не даст никакого прироста качества по сравнению с унимодальными моделями. Таким образом, в результате данного эксперимента было выявлено преимущество мультимодальной модели, но не по всем заявленным критериям.

5.4 Выдача тематических рекомендаций

В этой главе проверим гипотезу о том, что учет модальностей улучшает качество рекомендаций. Основная цель этой части эксперимента — сформировать для авторов статей рекомендации, наиболее полно отражающие интересующие их предметные темы, а затем проверить правильность созданных рекомендаций, подсчитать оценки качества.

На основе построенных тематических моделей строим рекомендательные модели. Для каждой тематической модели, описанной в предыдущей главе, была построена рекомендательная модель.

Опишем процесс построения модели. Сначала выборка с предпочтениями авторов была разделена на тестовую и обучающую в соотношении 1:1. С помощью условно-вероятностной модели было найдено распределение тематических предпочтений пользователей из обучающей выборки по формуле 5.1:

$$p(t|u) = \sum_{d \in D} p(t|d) \cdot p(d|u)_{train} \tag{5.1}$$

В 5.1 $p(t|d)$ — распределения тем в документах (получаем эти вероятности из тематического модели), $p(d|u)$ — распределения предпочтений пользователей из обучающей выборки.

Затем по формуле 5.2 вычисляются предпочтения пользователей из тестовой выборки:

$$p(d|u)_{test} = \sum_{t \in T} p(d|t) \cdot p(t|u) \tag{5.2}$$

Приведем пример рекоммендаций, построенных для одного случайного пользователя (его никнейм — iAndrey). Здесь отобразим топ5 рекомендаций и прочитанных пользователем статей:

Рекомендованные статьи:

1. ***Мобильные устройства — друзья или враги?***
2. Баг или фича с related ссылками?
3. ***Чтение RSS-потоков***
4. Борьба с комментариями-дубликатами

5. ***Кредитная карта + мобильный телефон***

Статьи, которые пользователь действительно читал:

1. ***Кредитная карта + мобильный телефон***
2. ***Мобильные устройства — друзья или враги?***
3. Старейшему сетевому изданию о музыке — 10 лет
4. Оптимизация стоимости при работе с Amazon S3
5. ***Чтение RSS-потоков***

Для этого пользователя рекомендации построены неплохо: 3 из 5 предложенных статей он действительно прокомментировал (а значит и прочитал).

На тестовой выборке были подсчитаны значения метрик Precision@k и Recall@k. Рассмотрим полученные значения метрик качества для трех рекомендательных моделей (контентно-ориентированная, человеко-ориентированная — usage-based, гибридная) и сравним модели между собой.

Table 5.11: Метрики качества рекомендательных моделей

Метрика качества	Контентно-ориентир.	Человеко-ориентир.	Гибридная модель
Precision@5	0.64	0.55	0.65
Precision@10	0.62	0.51	0.64
Precision@15	0.63	0.54	0.63
Precision@20	0.65	0.53	0.61
Recall@5	0.63	0.59	0.69
Recall@10	0.64	0.60	0.71
Recall@15	0.68	0.61	0.74
Recall@20	0.71	0.65	0.80

Анализ метрик Precision@k и Recall@k $k \in [5, 10, 15, 20]$ показывает, что рекомендации, полученные с помощью гибридной модели с модальностями оказались наиболее точными. Посчитаем MAP:

Table 5.12: MAP@k рекомендательных моделей

Метрика качества	Контентно-ориентир.	Человеко-ориентир.	Гибридная модель
MAP@5	0.62	0.48	0.64
MAP@10	0.63	0.51	0.65
MAP@15	0.65	0.53	0.66
MAP@20	0.64	0.52	0.68

Анализ метрик качества построенных рекомендательных моделей показал, что гибридный подход к выдаче рекомендаций на основе тематических моделей дает наиболее хорошие результаты. Все три модели побили базовое решение по выбранным метрикам качества, показав более высокие результаты по выдаче рекомендаций.

5.5 Сравнение тематического рекомендательного моделирования с базовым решением

Сравним результаты, полученные с применением тематического моделирования к задаче выдачи рекомендаций с базовым решением. На графике проиллюстрированы результаты для трех моделей из базового решения (рекомендательные модели на основе WRMF, KNN, popularity-sum, построенные с помощью библиотеки mrec) и для трех моделей, основанных на выделении тематики статей (контентно-ориентированная модель, модель на основе учета только предпочтений пользователей, гибридная модель).

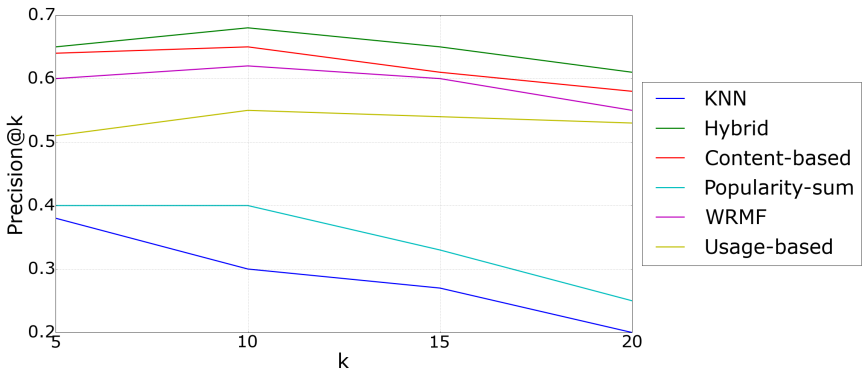

Рис. 5.9: $Precision@k, k \in [5, 10, 15, 20]$

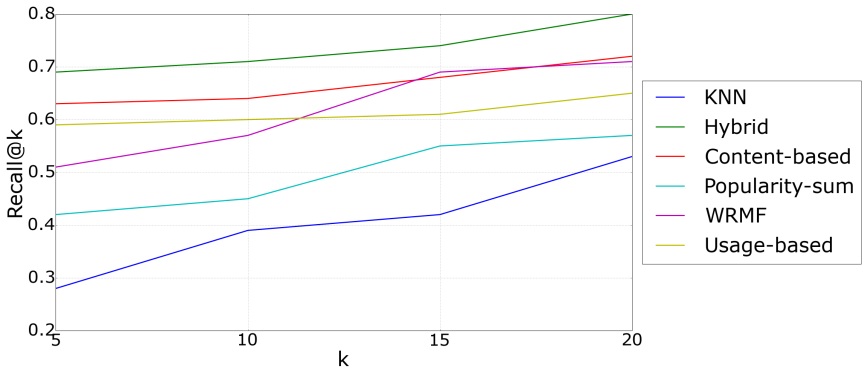

Рис. 5.10: $Recall@k, k \in [5, 10, 15, 20]$

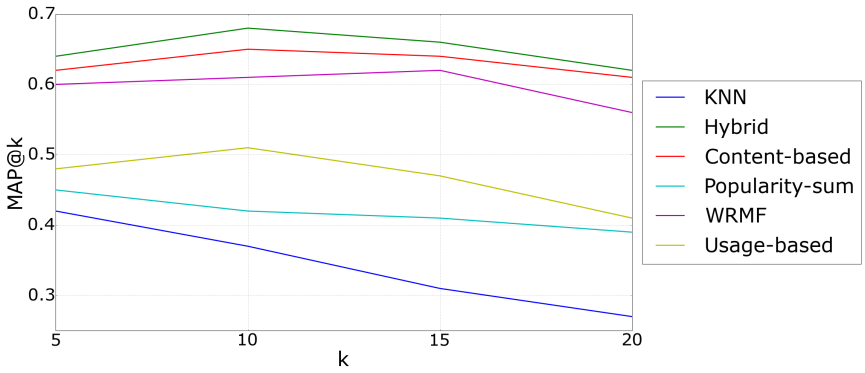

Рис. 5.11: $MAP@k, k \in [5, 10, 15, 20]$

На графике видно, что модель, объединяющая в себе контентно-основанный подход и учет известных предпочтений группы пользователей для прогнозирования неизвестных предпочтений другого пользователя (гибридная модель), сильно обгоняет все три модели из базового решения по метрике $MAP@k$. При этом модель на основе учета только информации об действиях пользователей в блоге (комментарии, лайки) уступает модели из базового решения (WRMF).

Часть 6

Разведочный тематический поиск

При решении задач разведочного поиска можно использовать те же методы, что и при решении задач рекомендации статей. Глобально перед пользователем стоит задача поиска и систематизации большого количества информации по новой для него теме, либо поиск с неявно сформулированным ожидаемым результатом. Необходимо порекомендовать пользователю статьи, которые соответствуют его информационной потребности, при этом она может быть сформулирована неточно или общо. На данном этапе важно понять, чем является запрос в задачах разведочного поиска и формализовать саму задачу поиска.

Запрос — это описание поискового намерения пользователя. Он может задаваться в виде статьи или набора статей по теме поиска, плана поиска или короткого текста с описанием поисковой задачи. В нашем эксперименте запрос — это текст объема примерно на один лист А4. В качестве типовой модельной ситуации тематического поиска мы рассматриваем поручение, которое менеджер информационного агентства мог бы дать своему подчинённому. Мы допускаем, что менеджер собрал из разных источников несколько абзацев текста, примерно задающих направление поиска, и достаточных, чтобы подчинённый правильно понял его поисковое намерение. Предполагается, что подчинённый тратит порядка часа времени на отработку такого сорта задания и может пользоваться любыми доступными ему средствами информационного поиска.

В этой главе мы предлагаем метод для быстрого решения задач тематического разведочного поиска, основанный на мультимодальном тематическом моделировании.

6.1 Алгоритм для тематического поиска

Пусть Q — текстовая коллекция запросов для разведочного поиска. Поскольку запрос в нашем эксперименте представлен в виде текста, для тематической модели запрос никак не отличается от обычной статьи. Будем строить тематическую модель на данных, состоящих из статей коллективного блога Хабра и текстов запросов из Q. Аналогично алгоритму выдачи рекомендаций из предыдущей главы находим тематический профиль каждого запроса $q \in Q$ по формуле 6.1:

$$p(t|q) = \Theta[q_i] \tag{6.1}$$

В 6.1 q_i — номер запроса q в текстовой коллекции (тематический профиль запроса — соответствующая строка из матрицы Θ).

Затем среди тематических профилей документов ищем k близких к профилю запроса векторов. Близость тематических профилей можно оценивать по-разному: с помощью евклидова расстояния, манхеттенского расстояния, косинусной меры. Мы будем использовать косинусную меру. Косинусная мера для двух векторов a и b:

$$cossim(a,b) = \frac{a \cdot b}{||a||||b||} = \frac{\sum\limits_{i=1}^{n} a_i \cdot b_i}{\sqrt{\sum\limits_{i=1}^{n} a_i^2} \cdot \sqrt{\sum\limits_{i=1}^{n} b_i^2}} \tag{6.2}$$

Полученный список документов нужно разметить на релевантные и нерелевантные, а затем оценить качество тематического поиска по метрикам precision@k, recall@k, map@k.

При условии наличия обученной тематической модели такой поиск занимает не более 2-3 секунд, тогда как человек тратит на решение задач разведочного поиска около часа. В следующей главе приведено подробное описание проведенного эксперимента и измерено качество работы созданного тематического поисковика на коллекции статей Хабрахабра.

6.2 Построение мультимодальной тематической модели для разведочного поиска

Тематическая модель коллективного блога Хабрахабр.ру, которая использовалась в эксперименте по построению рекомендаций из предыдущей главы, была

усовершенствована. Все настройки модели, количество модальностей, словари терминов остались прежними, были доработаны только стратегии регуляризации. Вместо перебора значений коэффициентов регуляризации по сетке, была использована более продвинутая техника для подбора коэффициентов при построении модели. Коротко опишем эту технику.

В подходе из предыдущей главы предлагалось добавлять регуляризаторы в модель одновременно, перед началом обучения модели. При этом предполагалось, что мы перебираем по сетке различные значения коэффициентов регуляризации и в итоге выбираем наилучшую модель (по критерию интерпретируемости тем и значениям метрик качества: перплексии, разреженности матриц Φ и Θ). Учитывая, что модель содержала три регуляризатора, а для каждого из них нужно было перебрать как минимум 3-5 значений для достижения высокого уровня качества модели, процесс подбора коэффициентов регуляризации занимал очень много времени (на коллекции статей коллективного блога Хабрахабр можель обучается в среднем 30 минут). Учитывая вышеизложенные проблемы, был разработан новый метод подбора коэффициентов.

В отличие от подхода из предыдущей главы, теперь регуляризаторы добавляются в модель последовательно, один за одним. Промежуток времени после добавления i-го регуляризатора до момента добавления $(i + 1)$-го регуляризатора будем называть глобальной итерацией. Каждая глобальная итерация содержит некоторое количество итераций EM-алгоритма при обучении модели (в нашем случае 8 итераций). На каждой глобальной итерации обучаем несколько моделей с разными значениями коэффициентов регуляризации для нового добавленного регуляризатора. Из этих моделей выбираем ту, которая позволяет улучшить одну (или несколько) метрик качества тематической модели без существенного понижения значений других метрик. Таким образом, на каждом этапе получаем модель с наиболее оптимально подобранными коэффициентами регуляризации. На следующей глобальной итерации выбираем наиболее оптимальную модель с предыдущей итерации, добавляем к ней новый регуляризатор и подбираем оптимальное значение коэффициента регуляризации для него. Такой подход дает значительный выигрыш по времени: вместо обучения $reg_1 \cdot reg_2 \cdot .. \cdot reg_n$ моделей (здесь reg_i — количество параметров для i-го регуляризатора, которые мы будем перебирать), нужно обучить всего лишь $reg_1 + reg_2 + .. + reg_n$. Кроме это преимущества, мы получаем возможность контролировать процесс обучения модели

и корректировать его за счет введения новых регуляризаторов с различными весами прямо в процессе обучения. Это позволяет получать более высокие значения метрик качества тематических моделей.

Проиллюстрируем описанную концепцию на примере. При обучении модели коллективного блога было использовано три регуляризатора:

- Декоррелирование распределений терминов в темах
- Сглаживание распределений терминов в темах
- Разреживание распределений тем в документах

Они вводились в модель в том порядке, в котором перечислены в списке. Для каждого регуляризатора перебиралось по три значения коэффициентов. Коэффициент декоррелирования распределений терминов был равен $10^6, 10^7, 10^8$ соответственно. Оказалось, что добавив с самого начала в модель декоррелирование с весом 10^8 мы получаем очень хорошие значения метрик качества (см. 6.1, 6.2, 6.3). Варьирование весов сглаживания и разреживания на последующих итерациях не позволяет побить первую модель, хотя и значительно приближлижает их к итоговой оптимальной по выбранным критериям модели.

Этот пример проиллюстрирован следующими графиками. Смотрим, как менялось значение перплексии, а также разреженности матриц Φ и Θ в процессе обучения модели (6.1, 6.2, 6.3).

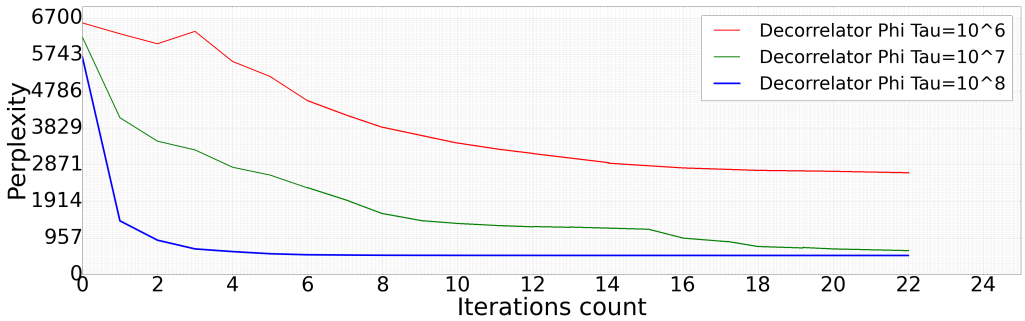

Рис. 6.1: Изменения перплексии при разных значениях коэффициента декоррелирования Φ

В результате итоговое значение перплексии подобранной модели было ниже, чем у модели из предыдущей главы. Кроме того, такой подход позволил значительно сократить время на подбор коэффициентов регуляризации и поиска оптимальной по заявленным критериям модели.

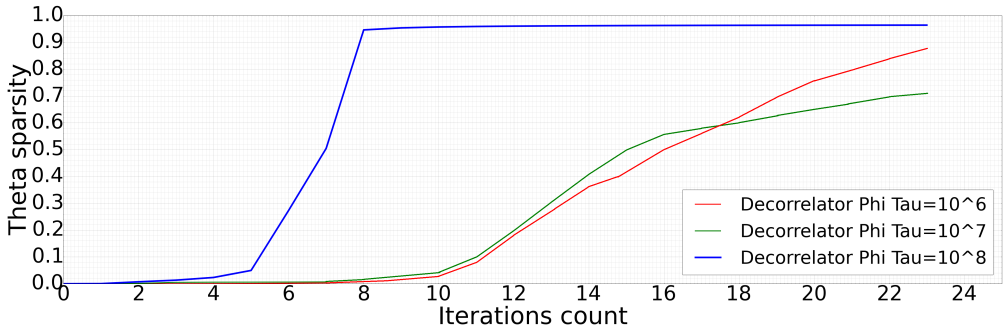

Рис. 6.2: Изменения разреженности матрицы Θ при разных значениях коэффициента декоррелирования Φ

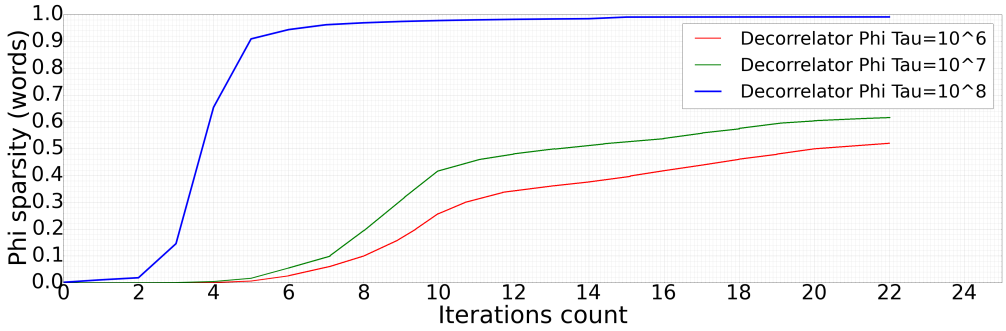

Рис. 6.3: Изменения разреженности матрицы Φ (модальность «термины») при разных значениях коэффициента декоррелирования Φ

6.3 Эксперимент по оцениванию качества разведочного поиска

Цель эксперимента из этой главы — показать, что с помощью мультимодального тематического моделирования можно решать задачи разведочного поиска более быстро и эффективно, чем стандартными средствами (поисковики, поиск по тегам, просмотр больших списков релевантных статей вручную).

На материалах коллективного блога Хабрахабр.ру был проведен эксперимент по оцениванию качества разработанного метода разведочного поиска. В процессе эксперимента было проведено сравнение разведочного поиска, осуществляемого людьми вручную и поиска с применением тематического поисковика. В эксперименте принимали участие 30 асессоров. Асессорам было предложено два задания:

1. Пользуясь любыми поисковыми средствами, осуществить разведочный поиск по предложенному запросу: найти как можно больше статей коллективного блога, релевантных запросу, а также замерять время, потраченное на

выполнение задания. На выходе нужно предоставить список ссылок на статьи с Хабрахабра и время, затраченное на поиск.

2. На том же запросе разметить результаты тематического поисковика. По каждой предложенной статье из выдачи нужно указать, является ли она релевантной запросу.

Всего для проведения эксперимента было сгенерировано 25 запросов. Каждый из них представляет собой текст примерно на одну страницу формата A4 с описанием направления поиска. Выбирались темы, подробно освещенные на Хабре, например: «Криптосистемы с открытым ключам», «Алгоритмы раскраски графов», «Облачные сервисы», «Марсоход Curiosity», «Космические проекты Илона Маска» и т.д. Все они связаны с технологиями или программмированием, что объясняется спецификой статей коллективного блога хабра. Каждый запрос был обработан тремя асессорами, далее результаты поиска усреднялись. В результатате после выполнения первого задания каждый асессор предоставлял список найденных статей и время, затраченное на поиск. Найденная статья считалась релевантной, только если как минимум два асессора из трех сочли ее соответствующей запросу. Обозначим через $founddocs$ количество документов, найденных суммарно тремя асессорами, через $reldocs$ — количество релевантных среди всех найденных документов. Тогда $precision$ по каждому запросу можно посчитать так:

$$precision = \frac{reldocs}{founddocs}$$

Результаты первого задания представлены в таблице. Каждая строка таблицы соответствует одному поисковому запросу.

Из данных таблицы видно, что в среднем на обработку запроса асессор тратит 30 минут. Важно отметить, что что только в 2 случаях из 25 $precision$ составил 1.0. В среднем, $precision = 0.88$. Это значит, что в большинстве случаев асессор находит не все релевантные запросу документы, либо включает в выдачу 1-2 нерелевантных документа. Этот факт имеет логичное объяснение: все люди мыслят по-разному, используют разные методики поиска, поэтому могут упускать из вида важные статьи. Эту проблему может помочь решить автоматический тематический поиск.

Для выполнения второго задания каждому асессору выдавался список статей, полученных с помощью тематического поисковика. Асессор размечал статьи из выдачи на релевантные и нерелевантные. Во втором задании каждый запрос

Table 6.1: Результаты асессорского разведочного поиска

Номер п/п	**Асессор 1**		**Асессор 2**		**Асессор 3**		**Всего**		**Среднее время (мин.)**	**Precision**
	Найдено док-ов	Время (мин.)	Найдено док-ов	Время (мин.)	Найдено док-ов	Время (мин.)	Найдено док-ов	Рел. док-ов		
1	8	50	9	40	8	55	9	8	48	**0.89**
2	21	40	25	50	23	30	25	23	40	**0.92**
3	8	10	10	15	8	20	10	8	15	**0.80**
4	19	15	15	15	16	20	20	16	17.5	**0.80**
5	18	40	15	30	20	50	18	17	40	**0.94**
6	51	30	50	30	55	60	55	51	40	**0.92**
7	10	15	12	20	11	15	12	11	16.7	**0.91**
8	10	25	12	30	9	30	12	10	28.3	**0.94**
9	***10***	***15***	***15***	***30***	***15***	***20***	***15***	***15***	***15***	***1.0***
10	17	20	15	30	20	25	20	17	25	**0.85**
11	5	20	6	30	5	20	6	5	23.3	**0.83**
12	20	30	21	30	18	25	21	19	28.3	**0.90**
13	7	10	5	12	10	10	10	7	10.6	**0.70**
14	19	20	20	20	20	20	20	19	20	**0.95**
15	30	60	25	40	23	50	30	25	40	**0.83**
16	10	20	12	20	10	25	12	10	21.6	**0.83**
17	7	15	8	20	8	10	8	7	15	**0.88**
18	15	40	15	30	16	30	16	15	33.3	**0.94**
19	20	50	22	55	18	45	22	20	50	**0.90**
20	10	30	15	25	17	20	17	15	25	**0.88**
21	8	30	9	30	10	30	10	8	30	**0.80**
22	5	30	7	20	10	10	10	7	20	**0.70**
23	35	70	30	40	32	50	35	30	53.3	**0.85**
24	21	25	20	30	25	35	25	20	30	**0.80**
25	***10***	***30***	***12***	***20***	***12***	***25***	***12***	***12***	***25***	***1.0***
						Итого:	**18**	**16**	**28.8**	**0.88**

также обрабатывался тремя асессорами и результаты усреднялись по аналогичному критерию: статья считалась релевантной только тогда, когда как минимум два асессора из трех посчитали ее подходящей.

По полученным результатам можно рассчитать метрики качества асессорского поиска и тематического разведочного поиска. Для каждого запроса были подсчитаны precision и recall. Полнота измерялась относительно статей, найденных и асессорами вручную, и тематическим поисковиком. Результаты представлены в таблице.

Из таблицы видно, что в среднем тематический поиск находит больше релевантных документов, соответственно *recall* у тематического поиска выше (0.91 против 0.89). При этом *precision* автоматического поиска ниже, чем у поиска вручную. Это ожидаемый результат: при поиске вручную случаи ошибочного отнесения нерелевантной статьи к списку редки, обычно это результат невнимательности или невдумчивого прочтения статьи. При автоматическом поиске ошибки могут возникать если, например, в документе освещается несколько тем, и тема, соответствующая запросу, является второстепенной. Такие ошибки не понижают полноту выдачи, но способствуют появлению в выдаче нерелевантных статей.

Одним из важнейших результатов этого этапа работы является то, что среди 25 запросов было обнаружено 8, для которых тематический поисковик дает $recall = 1.0$, это значит, что автоматический поисковик достаточно часто находит не только больше статей, чем асессоры, но и все возможные релевантные статьи по данной тематике (с поправкой на правила проведения эксперимента: полной выдачей по данному запросу мы считали найденные суммарно асессорами и тематическим поиском статьи).

Еще одно несомненное преимущество тематического поиска перед обычным полнотекстовым: выигрыш по времени. По данным из таблицы человек в среднем тратил 30 минут на запрос, наш тематический поисковик на обработку одного запроса тратит не больше 1-2 секунд.

Посмотрим на метрики качества по агрегированным данным для всех 25 запросов. Так же как и для задачи рекомендаций будем считать $Precision@k$ и $Recall@k$.

Анализ этих метрик также показывает преимущество тематического поиска перед полнотекстовым по полноте и отставание по метрике *precision*.

В результате проведенного эксперимента было показано, что тематический

Table 6.2: Сравнение результатов тематического и асессорского разведочного поиска

Номер п/п	**Асессоры**				**Тематический поиск**				**Асессоры + тем.поиск (рел. док-ы)**
	Найдено док-ов	Рел. док-ов	Precision	Recall	Найдено док-ов	Рел. док-ов	Precision	Recall	
1	9	8	0.89	0.80	12	10	0.83	1.0	10
2	25	23	0.92	0.95	25	24	0.92	1.0	24
3	10	8	0.80	0.88	11	9	0.72	1.0	9
4	20	16	0.80	0.94	22	17	0.77	1.0	17
5	18	17	0.94	0.85	20	17	0.85	0.85	20
6	55	51	0.92	1.0	57	48	0.84	0.94	51
7	12	11	0.91	1.0	14	8	0.57	1.0	11
8	12	10	0.94	0.83	10	9	0.90	0.75	12
9	15	15	1.0	0.88	20	17	0.85	1.0	17
10	20	17	0.85	0.94	21	15	0.71	0.83	18
11	6	5	0.83	0.83	8	5	0.63	0.83	6
12	21	19	0.90	0.90	25	20	0.80	0.95	21
13	10	7	0.70	0.88	10	7	0.70	0.88	8
14	20	19	0.95	0.95	21	18	0.86	0.9	20
15	30	25	0.83	1.0	32	25	0.78	1.0	25
16	12	10	0.83	0.9	10	8	0.80	0.72	11
17	7	7	0.88	0.88	10	7	0.70	0.88	8
18	15	15	0.94	0.93	23	14	0.60	0.88	16
19	20	20	0.90	0.95	20	18	0.90	0.86	21
20	17	15	0.88	0.94	18	15	0.83	0.94	16
21	10	8	0.80	0.80	15	9	0.60	0.9	10
22	10	7	0.70	0.88	10	6	0.60	0.88	8
23	35	30	0.85	0.93	32	28	0.88	0.88	32
24	25	20	0.80	0.80	27	23	0.85	0.92	25
25	12	12	1.0	1.0	15	12	0.80	1.0	12
Итого:	**18**	**15**	**0.87**	**0.89**	**20**	**18**	**0.77**	**0.91**	**18**

Table 6.3: Сравнение асессорского и тематического поиска по метрикам Precision@k и Recall@k

Метрика	**Асессоры**	**Тематический поисковик**
Precision@5	0.83	0.74
Precision@10	0.87	0.77
Precision@15	0.86	0.68
Precision@20	0.86	0.68
Recall@5	0.79	**0.83**
Recall@10	0.85	**0.88**
Recall@15	0.89	**0.91**
Recall@20	0.89	**0.91**

поиск при решении исследовательских задач работает не хуже, а зачастую и лучше обычного поиска вручную (с использованием поисковиков и других средств полнотекстового поиска). При этом мы получаем огромный выигрыш по времени: в среднем на обработку одного запроса человек тратил 30 минут, в то время, как тематический поиск работает практически мгновенно.

Часть 7

Заключение

7.1 Итоги работы

В данной работе разработан новый метод решения задач разведочного поиска, а также предложен способ построения рекомендаций, основанный на тематичеком моделировании. В рамках проведенного исследования была достигнута поставленная цель и решены сформулированные в начале исследования задачи. Подведем итоги по проделанной работе:

1. Были построены тематические модели с различным количеством модальностей: контентно-ориентированная модель (с одной модальностью «слова»), модель на основе данных об интересах пользователей (с одной модальностью «комментарии»), мультимодальная модель, учитывающая метаинформацию о статьях (она включала в себя пять модальностей: «слова», «комментарии», «автор», «теги», «категории»). Проведено сравнение моделей по критерию интерпретируемости тем, а также по значениям метрик качества; выявлено явное преимущество мультимодальной модели по сравнению с унимодальными.

2. На основе построенных тематических моделей были обучены рекомендательные модели и измерено качество выдачи рекомендаций пользователям. Выявлено, что учет дополнительных модальностей улучшает качество рекомендательного ранжирования статей: рекомендации построенные на основе мультимодальной тематической модели были наиболее точными. Результаты по построению тематических рекомендаций опубликованы [19, 20].

3. Был разработан метод решения задач разведочного поиска: тематический

поисковик, позволяющий по текстовому описанию поискового намерения пользователя находить статьи необходимой тематики.

4. Была предложена методика оценивания качества разведочного поиска.

5. Был проведен эксперимент с участием 30 асессоров по оцениванию качества тематического поиска.

6. В результате эксперимента было выявлено преимущество созданного тематического поисковика перед полнотекстовым поиском при решении задач разведочного поиска.

7.2 Дальнейшие исследования

Рассмотрим возможные направления дальнейшего исследования в рамках данной работы:

1. Попробовать в качестве меры близости тематических профилей документов другие расстояния (евклидово расстояние, взвешенное евклидово расстояние, манхеттенское расстояние), сравнить полученные результаты между собой.

2. Расширить коллекцию статей для исследования качества тематического поисковика. Например, исследовать качество разведочного поиска на статьях из Википедии, коллекции статей ПостНаука.

3. Расширить базу запросов для разведочного поиска. Увеличить количество запросов для эксперимента, а также расширить их тематическую направленность. На данный момент все запросы связаны с техническими темами: программирование, космические исследования, новые технологии, гаджеты и т.д. Нужно сгенерировать запросы, охватывающие другие области знаний.

4. Протестировать рекомендательную систему статей для Хабра в онлайне. Для этого необходимо разработать удобный веб-интерфейс рекомендаций, затем протестировать его с помощью асессоров (например, постоянных читателей Хабра). При успешном завершении тестирования продумать стратегию внедрения системы тематических рекомендаций на Хабрахабр.

Литература

[1] K.Vorontsov. Additive regularization of topic models: Towards exploratory search and other multi-criteria applications. *Yandex School of Data Analysis Conference*, 2015.

[2] A.Potapenko K.Vorontsov. Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization. *Communications in Computer and Information Science (CCIS)*, 436:29–46, 2014.

[3] К.В.Воронцов. Аддитивная регуляризация тематических моделей коллекций текстовых документов. *Доклады РАН*, 455(3):268–271, 2014.

[4] K.Konuyushkova K.Athukorala S.Kaski G.Jacucci D.Glowacka, T.Ruotsalo. Directing exploratory search: Reinforcement learning from user interactions with keywords. *Proceedings of the 2013 international conference on Intelligent user interfaces*, pages 117–128, March 2013.

[5] G.Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, April 2006.

[6] S.Goldenberg. Exploratory search in worktop. *Master-Thesis, Brown-University, Providence, Rhode Island, USA*, 2012.

[7] Y. Mejova M.Lalmas O.Van Laere, I.Bordino. Deesse: entity-driven exploratory and serendipitous search system. *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 2072–2074, 2014.

[8] S. Simske G. Koutrika, L. Liu. Generating reading orders over document collections. *2015 IEEE 31st International Conference on Data Engineering*, pages 507–518, March 2015.

[9] M.Gontek. User modeling for exploratory search on the social web. *Dissertation zur Erlangung des Doktorgrades der Philosophischen Fakultet der Universitet zu Keln im Fach Informationsverarbeitung*, June 2011.

[10] S.Gerrish Ch.Wang D.M.Blei J.Chang, J.Boyd-Graber. Reading tea leaves: How humans interpret topic models. *Advances in neural information processing systems*, pages 288–296, 2009.

[11] T.Hoffman. Probabilistic latent semantic analysis. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57, 1999.

[12] X.Cheng H.Wu, Y.Wang. Incremental probabilistic latent semantic analysis for automatic question recommendation. *Proceedings of the 2008 ACM conference on Recommender systems*, pages 99–106, 2008.

[13] M.I. Jordan D.M. Blei, A.Y. Ng. Latent dirichlet allocation. *Journal of Machine Learning Research*, pages 993–1022, January 2003.

[14] A.Banerjee H.Shan. Generalized probabilistic matrix factorizations for collaborative filtering. *Proceedings of the 2010 IEEE International Conference on Data Mining,*, pages 1025–1030, 2010.

[15] D.M.Blei Ch.Wang. Collaborative topic modeling for recommending scientific articles. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 448–456, 2011.

[16] T.M.Khoshgoftaar X.Su. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009.

[17] D.Billsus M.J.Pazzani. Content-based recommendation systems. *The adaptive web*, pages 325–341, 2007.

[18] C.Volinsky Y.Hu, Y.Koren. Collaborative filtering for implicit feedback datasets. *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 263–272, 2008.

[19] M.Apishev M.Dudarenko P.Romov A.Yanina K.Vorontsov, O.Frei. Non-bayesian additive regularization for multimodal topic modeling of large collections. *Proceedings*

of the 2015 Workshop on Topic Models: Post-Processing and Applications, pages 29–37, 2015.

[20] П.А. Ромов А.О.Янина М.А.Суворова М.А.Апишев К.В.Воронцов, А.И.Фрей. Bigartm: библиотека с открытым кодом для тематического моделирования больших текстовых коллекций. *Труды XVII Международной конференции DAMDID/RCDL'2015 «Аналитика и управление данными в областях с интенсивным использованием данных»*, 2015.

[21] J.D.Lafferty D.M. Blei. Correlated topic models. *Advances in neural information processing systems*, 18:147, 2006.

[22] K.Athukorala. Is exploratory search different? a comparison of information search behavior for exploratory and lookup tasks. *Journal of the association for information science and technology*, page 32, July 2015.

[23] R.Nagarajan P.Melville, R.J.Mooney. Content-boosted collaborative filtering for improved recommendations. *American Association for Artificial Intelligence*, pages 187–192, July 2002.