Московский физико-технический институт
(государственный университет)

На правах рукописи
УДК 519.254

Кузьмин Арсентий Александрович

ИЕРАРХИЧЕСКАЯ КЛАССИФИКАЦИЯ КОЛЛЕКЦИЙ ДОКУМЕНТОВ

05.13.17 — Теоретические основы информатики

Диссертация на соискание ученой степени
кандидата физико-математических наук

Научный руководитель:
д.ф.-м.н. В. В. Стрижов

Москва — 2017

Оглавление

Введение

Актуальность темы. В работе исследуются методы категоризации и классификации текстовых документов, автоматически структурирующие документы в виде иерархий тем и оптимизирующие уже существующие, выявляя в них тематические несоответствия [1, 2, 3, 4, 5, 6, 7, 8].

Тематическая модель – модель коллекции текстовых документов, которая определяет, к каким темам относится каждый документ коллекции. В работе исследуется фундаментальная проблема тематического моделирования – классификация документов из частично размеченных коллекций с экспертно заданной иерархической структурой тем [9, 10, 11, 12]. Решением задачи классификации является отображение подмножества неразмеченных документов коллекции во множество тем, наилучшим образом восстанавливающее экспертную классификацию согласно заданному критерию качества. В случае большого числа тем вместо единственного релевантного кластера предлагается ранжированный список кластеров согласно их релевантности документу. При несовпадении экспертного мнения и наиболее релевантного кластера, эксперт рассматривает следующие по релевантности кластеры в качестве альтернативных вариантов.

Коллекциями документов являются аннотации к научным работам [13], доклады на конференциях [14], текстовые сообщения в социальных сетях [15, 16], текстовая информация веб-сайтов [17], описания патентов, новостные сводки [18, 19] и описания фильмов [16]. Предполагается, что экспертное разделение документов на темы является эталонным. В связи со значительным размером коллекций и числом тем распределение документов по темам является для экспертов трудоемкой задачей. Поэтому автоматическая классификация неразмеченных документов и поиск небольшого числа наиболее подходящих тем для каждого неразмеченного документа для дальнейшего принятия решения экспертом являются актуальными задачами.

Для текстовой классификации и кластеризации были предложены жесткие методы, в которых каждому документу ставится в соответствие единственный кластер [20, 11], описательно вероятностные методы, в которых оценивается вероятность принадлежности документа каждому из кластеров [6, 21], смеси моделей [7] и вероятностные методы [22, 1, 2] в которых темы являются распределениями над множеством слов, а документы – распределениями над множеством тем. Для коллекций с большим числом тем были предложены иерархические методы, позволяющие учитывать взаимосвязи между темами [11, 23, 8].

Важной проблемой при построении метрических алгоритмов классификации и кластеризации является выбор метрики [24] как способа сравнения векторных представлений документов. В [25] для учета соотношения масштабов признаков рассматривается взвешенная метрика Минковского. Веса интерпретируются как важность слов. В данной работе исследуются способы оптимизации весов взвешенной метрики, а также различные способы векторного представления документов, наилучшим образом восстанавливающие эксперт-

ную классификацию. Альтернативой взвешенной функции расстояния является взвешенная функция сходства [26]. Для уменьшения числа параметров оптимизации предлагается энтропийный метод определения важности слов во взвешенной функции сходства через их энтропию относительно экспертной кластеризации на различных уровнях иерархии. Для иерархической классификации предлагается иерархическая взвешенная функция сходства, позволяющая учитывать сходство сразу со всей веткой дерева экспертной иерархической структуры коллекции.

Для оптимизации параметров иерархической функции сходства рассматривается вероятностная постановка задачи, в которой вероятность принадлежности кластеру оценивается как нормированная экспоненциальная функция softmax от значений иерархического сходства с кластерами. Задача поиска параметров иерархической взвешенной функции сводится к максимизации правдоподобия модели.

При наличии априорных распределений параметров аналитический байесовский вывод апостериорного распределения параметров иерархической функции сходства, и совместного апостериорного распределения параметров и классов неразмеченных документов не является возможным. В работах [27, 28] рассматриваются способы приближенного вариационного вывода и аппроксимации правдоподобия. В работе данные идеи используются для аналитического вывода апостериорного распределения параметров [29, 30], а также для аппроксимации совместного апостериорного распределения классов неразмеченных документов и параметров.

Для размеченных коллекций возникает задача верификации. Решением этой задачи является изменение у фиксированного набора документов их тем так, чтобы качество полученной модели стало максимальным. Для этого предлагается алгоритм построения иерархической модели, схожей с существующей, для выявления значимых тематических несоответствий в модели. Предлагаются варианты устранения несоответствий путем переноса некоторых документов в другие кластеры.

Для визуализации тематической модели были предложены различные подходы [31, 32]. В случае, когда документы представляются в виде действительных векторов, для их визуализации используются методы понижения размерности [33]. При этом кластеры из разных ветвей иерархической модели могут пересекаться. В данной работе предлагается метод построения плоской вложенной визуализации иерархической модели, при которой кластеры более низкого уровня остаются внутри кластеров более высокого уровня на плоскости. Предлагаемый подход опирается на методы, минимизирующие изменения относительного расстояния между документами и центрами кластеров иерархии [34].

Цели работы.

1. Исследовать метрические свойства описаний текстовых документов.

2. Предложить критерии качества модели иерархической классификации документов.
3. Построить оптимальную модель иерархической классификации.
4. Получить вариационные оценки апостериорных распределений параметров и гиперпараметров модели.
5. Разработать алгоритм построения модели и провести вычислительный эксперимент для сравнения различных подходов к решению задачи иерархической классификации документов.

Методы исследования. Для достижения поставленных целей используются методы иерархического тематического моделирования [22, 8, 11, 35, 23]. Для метрической иерархической кластеризации применяются методы плоской кластеризации [24, 36] совместно с агломеративным и дивизимным подходами [37, 11]. Для построения локально оптимальной взвешенной метрики используются методы отбора признаков [38] и методы условной оптимизации [39, 29]. Для сравнения документов при иерархической классификации используется взвешенная функция сходства [26], а для оптимизации ее параметров развивается энтропийный метод, предложенный в [37]. Для оптимизации параметров иерархической взвешенной функции сходства и энтропийной модели используются методы вариационного вывода [27, 28], байесовского вывода [40] и методы локальных вариаций [29]. Для построения оператора релевантности используются методы иерархической классификации [10, 9]. Для построения плоской вложенной визуализации иерархической тематической модели используются методы понижения размерности [34]. Для учета синонимичности слов используются языковые модели [41, 42] и методы оптимизации параметров нейронных сетей [29]. Кроме того, используются элементы теории вероятности и выпуклой оптимизации [39].

Основные положения, выносимые на защиту.

1. Предложен метод иерархической классификации коллекций документов на основе оператора релевантности.
2. Разработана и исследована вероятностная модель иерархической классификации.
3. Предложены методы оптимизации параметров и гиперпараметров модели.
4. Предложен способ вычисления иерархической вероятности класса документа и построения ранжированного списка для последующей экспертной оценки.
5. Разработан программный комплекс для экспертного построения программы конференции.

Научная новизна. Разработан новый подход иерархической классификации частично размеченных коллекций текстовых документов с экспертной

иерархической структурой. Предложена иерархическая взвешенная функция сходства документа и кластера, учитывающая иерархичность экспертной кластерной структуры. Предложен метод оценки важности слов с помощью энтропийной модели. Предложена вероятностная модель текстовой коллекции и способ аппроксимации совместного апостериорного распределения параметров модели и классов неразмеченных документов. Предложен способ представления иерархической функции сходства в виде многослойной нейронной сети и способ учета синонимичности слов. Введен оператор релевантности, ранжирующий кластеры тематической модели по убыванию релевантности новому документу. Для верификации экспертной тематической модели предложен метод построения модели, схожей с экспертной, и выявления наиболее значимых несоответствий. Предложен метод вложенной визуализации экспертной иерархической тематической модели на плоскости, а также выявленных несоответствий и вариантов повышения тематической целостности модели.

Теоретическая значимость. В данной диссертационной работе предложенные ранее функции расстояния обобщаются для учета важности признаков путем введения их весов. Взвешенная функция сходства обобщается на случай иерархических моделей. Вычисляются оценки весов взвешенной функции сходства с помощью обобщения энтропийного подхода. Для вероятностной модели коллекции документов, основанной на иерархической функции сходства, предлагается способ оценки апостериорного распределения параметров, а также совместного апостериорного распределения параметров и классов неразмеченных документов. Доказываются свойства полученных оценок.

Практическая значимость. Предложенные в работе методы предназначены для иерархической классификации коллекций текстов с учетом существующих экспертных моделей; выявления тематических несоответствий в экспертных моделях и значимого повышения тематической целостности уже построенных тематических моделей с помощью небольшого числа изменений; визуализации иерархических моделей и выявленных несоответствий на плоскости.

Степень достоверности и апробация работы. Достоверность результатов подтверждена математическими доказательствами, экспериментальной проверкой полученных методов на реальных задачах иерархической классификации коллекций тезисов конференции и коллекций сайтов индустриального сектора; публикациями результатов исследования в рецензируемых научных изданиях, в том числе рекомендованных ВАК. Результаты работы докладывались и обсуждались на следующих научных конференциях.

1. Международная конференция “26th European Conference on Operational Research”, 2013 [43].
2. Международная конференция “20th Conference of the International Federation of Operational Research Societies”, 2014 [44].

3. Всероссийская конференция “Математические методы распознавания образов” ММРО-17, 2015 [45].
4. Всероссийская конференция “58 научная конференция МФТИ”, 2015.
5. Всероссийская конференция “Ломоносов-2016”, 2016 [46].
6. Международная конференция “28th European Conference on Operational Research”, 2016 [47].

Работа поддержана грантами Российского фонда фундаментальных исследований и Министерства образования и науки РФ.

1. 14-07-31264, Российский фонд фундаментальных исследований в рамках гранта “Развитие методов визуализации иерархических тематических моделей”.
2. 07.524.11.4002, Министерство образования и науки РФ в рамках Государственного контракта “Система агрегирования и публикации научных документов ВебСервис: построение тематических моделей коллекции документов”.

Публикации по теме диссертации. Основные результаты по теме диссертации изложены в 10 печатных изданиях, 4 из которых изданы в журналах, рекомендованных ВАК.

1. Кузьмин А. А. Многоуровневая классификация при обнаружении движения цен // Машинное обучение и анализ данных, 3 (2012). С. 318-327 [48].
2. Кузьмин А. А., Адуенко А. А., Стрижов В. В. Выбор признаков и оптимизация метрики при кластеризации коллекции документов // Известия ТулГУ, 3 (2012). С. 119-131 [49].
3. Кузьмин А. А., Стрижов В. В. Проверка адекватности тематических моделей коллекции документов. // Программная инженерия, 4 (2013). С. 16-20 [50].
4. Kuzmin A. A., Aduenko A. A., Strijov V. V. Hierarchical thematic model visualizing algorithm // 26th European Conference on Operational Research, Rome, (2013). P. 155 [43].
5. Kuzmin A. A., Aduenko A. A., Strijov V. V. Thematic Classification for EURO/IFORS Conference Using Expert Model // 20th Conference of the International Federation of Operational Research Societies, Barcelona, (2014). P. 173 [44].
6. Кузьмин А. А., Адуенко А. А., Стрижов В. В. Тематическая классификация тезисов крупной конференции с использованием экспертной модели // Информационные технологии. 6 (2014). С. 22-26 [14].
7. Кузьмин А. А., Стрижов В. В. Построение иерархических тематических моделей крупных конференций // Математические методы распознавания образов ММРО-17. Тезисы докладов 17-й Всероссийской конференции с международным участием, г. Светлогорск: Торус пресс., (2015). С. 224–225 [45].

8. Кузьмин А. А., Адуенко А. А. Построение иерархических тематических моделей крупных конференций // Сборник тезисов 23 международной научной конференции студентов, аспирантов и молодых ученных "Ломоносов-2016" секция "Вычислительная математика и кибернетика", г. Москва: МАКС Пресс., (2016). С. 73–75 [46].
9. Kuzmin A. A., Aduenko A. A., Strijov V. V. Thematic Classification for EURO/IFORS Conference Using Expert Model // 28th European Conference on Operational Research, Poznan, (2016). P. 206 [47].
10. Златов А. С., Кузьмин А. А. Построение иерархической тематической модели крупной конференции // Искусственный интеллект и принятие решений, 3 (2016). С. 77-86 [21].

Личный вклад. Все приведенные результаты, кроме отдельно оговоренных случаев, получены диссертантом лично при научном руководстве д.ф.-м.н. В. В. Стрижова.

Структура и объем работы. Диссертация состоит из оглавления, введения, пяти разделов, заключения, списка иллюстраций, списка таблиц, перечня основных обозначений и списка литературы из 123 наименований. Основной текст занимает 120 страниц.

Краткое содержание работы по главам. В первой главе вводятся основные понятия и определения, формулируются задачи иерархической классификации и кластеризации. Рассматриваются основные этапы классификации и кластеризации коллекций документов существующими методами: предобработка коллекции текстовых документов, составление словаря коллекции, представление слов в виде векторов, представление документов в виде векторов, построение модели. Рассматриваются четыре основных подхода построения тематической модели: с помощью жестких методов, описательно-вероятностных методов, смесей моделей и вероятностных методов. Рассматриваются существующие варианты алгоритмов иерархической классификации.

Во второй главе предлагается алгоритм иерархической метрической кластеризации. Рассматривается взвешенная функция расстояния Минковского и ее свойства. Предлагается алгоритм оптимизации весов данной функции с помощью частично размеченной коллекции. Анализируются агломеративный и дивизимный методы построения иерархической тематической модели, а также сравниваются различные способы представления документов в виде действительных векторов.

В третьей главе рассматривается способ вычисления взвешенного сходства между векторными представлениями документов и кластеров. Для оптимизации весов данной функции предлагается энтропийный подход, использующий экспертную кластеризацию документов на различных уровнях иерархии. Предлагается иерархическая взвешенная функция сходства, характеризующая сход-

ство документа и ветки дерева иерархической модели. Предлагается оператор релевантности, ранжирующий кластеры нижнего уровня иерархической модели в порядке убывания сходства с неразиеченным документом. Вводится критерий качества оператора релевантности AUCH. Для оптимизации параметров иерархической функции сходства предлагается итеративный алгоритм, оптимизирующий функционал качества AUCH. Предлагается способ оценки вероятности принадлежности документа кластеру и строится вероятностная модель коллекции документов. Предлагается способ оптимизации параметров данной модели, максимизирующий правдоподобие модели по размеченным документам. Вводятся априорные распределения параметров модели, с помощью вариационного вывода строится оценка апостериорного распределения параметров. Для оценки вероятности принадлежности документа кластеру строится оценка совместного апостериорного распределения параметров модели и классов неразмеченных документов. Для учета синонимичности слов предлагается способ инициализации параметров с помощью векторных представлений слов и обученной языковой модели. С помощью предложенных методов классифицируются аннотации к докладам конференции EURO.

В четвертой главе рассматривается задача верификации экспертной иерархической тематической модели. Предлагается алгоритм построения модели, схожей с экспертной. Вводится понятие качества экспертной модели и система штрафов за ее изменение. С помощью предложенного метода проводится верификация экспертной тематической модели конференции EURO.

В пятой главе на базе предложенных методов описывается разработанный программный комплекс, позволяющий классифицировать неразмеченные текстовые документы с помощью экспертных моделей. Работа данного комплекса анализируется на двух текстовых коллекциях: коллекции аннотаций к докладам на крупной конференции EURO, и коллекции веб-сайтов компаний индустриального сектора. Результаты, полученные с помощью предложенных методов, сравниваются с результатами известных алгоритмов. Предлагается метод построения вложенной визуализации экспертной иерархической тематической модели на плоскости, а также выявленных несоответствий и способов их устранения.

Глава 1
Постановка задачи

Обработка тестовой информации является одной из наиболее важных задач в области интеллектуального анализа данных. Теоретические результаты в данной области находят непосредственное применение при решении прикладных задач, в частности, задач ранжирования поисковых выдач по запросу, задач информационного поиска, анализа текстов, построения тематических моделей коллекции текстов и терминологических словарей.

Определение 1. Словом w называется любой неразрывный набор символов.

Определение 2. Текстовым документом d называется множество слов $\{w_1, w_2, \ldots, w_n\}$. Размером документа $|d|$ называется число элементов данного множества.

Определение 3. Коллекцией документов D называется неупорядоченное множество документов $\{d_1, d_2, \ldots, d_n\}$. Размером коллекции $|D|$ называется число элементов данного множества.

Определение 4. Словарем W коллекции D называется упорядоченное подмножество неповторяющихся слов w и словосочетаний $w_1 w_2 \ldots w_n$, содержащихся в коллекции D.

В данной работе словарь W содержит всевозможные слова w из коллекции D и не содержит словосочетания, если не оговорено иное.

Определение 5. Кластером документов c называется подмножество документов коллекции D. Корневым кластером называется кластер, содержащий все документы коллекции D. Документ d имеет класс c, если $d \in c$. В общем случае, каждый документ может принадлежать произвольному числу классов.

Определение 6. Кластер c_1 является родительским кластером кластера c_2 если все документы d из c_2 содержатся в c_1. При этом кластер c_2 называется дочерним кластером c_1.

Определение 7. Тематической моделью M текстовой коллекции D называется разбиение D на кластеры $\{c_1, c_2, \ldots, c_n\}$ таким образом, чтобы каждый документ $d \in D$ принадлежал хотя бы одному кластеру помимо корневого.

Тематическая модель M коллекции D называется экспертной, если для каждого документа $d \in D$ его классы задавались экспертами. Тематическая модель $\hat{M}$ называется алгоритмической, если для документов классы задавались алгоритмическим образом. Коллекция D называется частично размеченной, если экспертная классификация известна только для подмножества документов. Кластерная структура коллекции задана экспертно, если изначально задан граф модели в виде графа кластеров.

Пусть каждый кластер модели M, кроме корневого, является чьим-то дочерним кластером. Модель M представляется в виде графа следующим образом. Каждому кластеру c ставится в соответствие вершина. Пусть $\{c_1, \ldots, c_n\}$ – множество родительских кластеров для кластера c. Вершина, соответствующая кластеру c, соединяется ребром со всеми вершинам, соответствующими кластерам $\{c_1, \ldots, c_n\}$, кроме тех, для которых хотя бы один из кластеров в $\{c_1, \ldots, c_n\}$ является дочерним.

Определение 8. Модель M называется иерархической, если она представима в виде направленного ациклического графа (DAG) [51, 52]. Уровнем кластера c в M называется сумма уровня корневого кластера и максимальной длины пути в графе модели от корневого кластера до c. Уровнем корневого кластера считается 1.

Каждый кластер $c_{l,k}$ индексируется двумя числами – уровнем l и порядковым номером на данном уровне k. Корневой кластер обозначается как $c_{1,1}$, число кластеров на уровне l обозначается K_l.

Определение 9. Кластеры иерархической модели M у которых нет дочерних кластеров называются терминальными. В общем случае уровни данных кластеров могут различаться.

Определение 10. Иерархическая модель называется сбалансированной, если у всех терминальных кластеров совпадает уровень.

В данной работе исследуется фундаментальная задача построения иерархических тематических моделей M с экспертно заданной кластерной структурой, классификации неразмеченных документов в данной структуре и верификации ранее построенной модели. Исследуются свойства иерархических тематических моделей. При построении модели M заданы:

1) частично размеченная коллекция документов D,
2) экспертная кластерная структура в виде дерева,
3) классы документов заданного подмножества D,
4) тип модели, как способ отнесения документа к элементу кластерной структуры.

Требуется определить положение каждого неразмеченного документа коллекции в структуре тематической модели. Ранее предложен ряд методов [9, 10, 11, 37] для решения подобных задач. Каждый из них использует определенные начальные условия и предположения о структуре. В большинстве из них можно выделить следующие этапы построения модели: 1) предобработка документов коллекции, 2) построение словаря коллекции, 3) представление документов в виде числовых векторов и 4) применение алгоритма построения тематической модели к полученному набору векторов. Рассмотрим каждый из этапов.

1.1. Предобработка документов

Основной целю предобработки документов является удаление неинформативных слов [53] и приведение оставшихся слов к их нормальной форме. Согласно [54, 55, 56] предобработка позволяет улучшить качество классификации документов для некоторых языков в 10–50 раз, а так же уменьшить размер словаря на 50%. Для удаления неинформативных слов и незначимых частей речи, таких как союзы и предлоги, используются словари стоп-слов [53]. Для нормализации слов существует три основных метода [57].

Метод удаления аффиксов. Для каждого слова из D существует последовательность суффиксов, в которой они присоединены к корню. На каждом шаге метод [58] удаляет с конца слова один суффикс всевозможными способами, сверяет получившееся слово со списком нормальных форм слов и при отсутствии совпадения рекурсивно ищет нормальную форму от оставшегося слова.

Метод разнообразия продолжений. Сегментами слова являются его отделяемые части — корень, суффикс, приставка. Пусть $A = \{a\}$ – множество символов языка коллекции D. Пусть

$$w = a_{i_1}a_{i_2}\dots a_{i_n}, \quad wa = a_{i_1}\dots a_{i_n}a, \quad w^m = a_{i_1}\dots a_{i_m},\ m \le n,$$

где w – представление слова в виде последовательности букв, wa – конкатенация слова w и буквы a, а w^m – первые m букв слова w. Пусть H – множество всех слов коллекции D, $H(w^m)$ – все слова из H, у которых первые m букв совпадают с w^m, а $S(w^m)$ — множество различных букв, встречающихся на $m+1$ позиции в словах из $H(w^m)$. Пусть κ^* – структурный параметр. Для поиска сегментов в [59] используются следующие методы.

1. Метод отсечения – считать w^m сегментом если $|S(w^m)| > \kappa^*$.
2. Метод пика и плато – считать w_m сегментом если
$$|S(w^{m+1})| - |S(w^m)| > \kappa^*.$$
3. Метод совпадений — считать w^m сегментом если w без первых m букв совпадает с другим словом w' из H.
4. Энтропийный метод. Энтропия $I(w^m)$ разнообразия продолжений последовательности букв w^m задается как
$$I(w^m) = -\sum_{a\in A} p(w^m a)\log p(w^m a), \quad p(w^m a) = \frac{|H(w^m a)|}{|H(w^m)|},$$
где $p(w^m a)$ – вероятность того, что случайное слово из $H(w^m)$ имеет в качестве продолжения букву $a \in A$. Последовательность букв w_m считается сегментом если
$$|I(w^{m+1})| - |I(w^m)| > \kappa^*.$$

Для нормализации, все слова коллекции делятся на сегменты описанными выше способами и для каждого слова выбирается нормальная форма из множества его сегментов.

Кластеризация слов. В [60] слова разбиваются на кластеры согласно заданной функции расстояния или сходства между словами. Словам из одно кластера ставится в соответствие одинаковая нормальная форма. Для кластеризации применяется метод полной связи [61]. Расстояние между словами w_1 и w_2 в этом методе определяется как расстояние Левенштейна [62]. Для вычисления сходства между словами, каждому слову ставится в соответствие набор всех его подпоследовательностей букв длинны n [63] и для полученных множеств вычисляется мера сходства Дайса [64].

1.2. Составление словаря коллекции

После предобработки коллекции D, словарь W содержит слова из коллекции D без повторений. Добавление в W устойчивых словосочетаний позволяет улучшить качество кластеризации коллекции [65, 66]. Устойчивым словосочетанием называется последовательность слов, N-грамма, часто встречающаяся в документах коллекции.

Для поиска устойчивых словосочетаний в [66] предлагается упорядочить всевозможные пары слов по значению ассоциативной меры

$$\text{T-Score}(w_1w_2, D) = \frac{|W|\,\text{tf}(w_1w_2, D) - \text{tf}(w_1, D)\,\text{tf}(w_2, D)}{|W|\sqrt{\text{tf}(w_1w_2, D)}},$$

где $\text{tf}(w, D)$ – частота слова w в коллекции D. Словосочетания с наибольшим значением T-Score являются наиболее устойчивыми и добавляются в словарь.

В [67] для отбора устойчивых словосочетаний используется взаимная информация:

$$\text{MI}(w_1w_2) = \log \frac{|W| \cdot \text{tf}(w_1w_2, D)}{\text{tf}(w_1, D) \cdot \text{tf}(w_2, D)}.$$

Чем больше ее значение для словосочетания w_1w_2, тем более устойчивым оно является. Однако данная мера не ограничена сверху, а $\text{tf}(w_1, D)$ и $\text{tf}(w_2, D)$ в знаменателе не зависят от того, встретилось ли слово w_1 вместе с w_2 или нет. Чтобы учесть эти недостатки, были предложены аналоги данной меры: дополненная взаимная информация [68] и нормализованная взаимная информация [69].

В [65] предлагается подход PLSA-ITER, основанный на алгоритме PLSA [1], в котором шаг построения модели PLSA чередуется с шагом генерации и добавления в словарь всевозможных словосочетаний слов, с наибольшей вероятностью характеризующих одну из тем, полученных с помощью PLSA. В [70] для увеличения числа словосочетаний предлагается в качестве N-грамм рассматривать последовательности слов с не более чем m пропусками.

1.3. Представление слов из словаря в виде векторов

Пусть каждому слову w из словаря W ставится в соответствие некоторый вектор $\mathbf{w}(w) \in \mathbb{R}^m$, а сходство двух слов $s_{\mathsf{w}}(w_i, w_j)$ выражается как косинус угла между их векторами:

$$s_{\mathsf{w}}(w_i, w_j) = \frac{\|\mathbf{w}(w_i) - \mathbf{w}(w_j)\|}{\|\mathbf{w}(w_i)\|\|\mathbf{w}(w_j)\|} = \frac{\langle \mathbf{w}(w_i), \mathbf{w}(w_j)\rangle}{\sqrt{\langle \mathbf{w}(w_i), \mathbf{w}(w_i)\rangle}\sqrt{\langle \mathbf{w}(w_j), \mathbf{w}(w_j)\rangle}}.$$

Тривиальным векторным представлением слова w_j, стоящего на позиции j в словаре W, является единичный вектор $\mathbf{e}(j)$ с единицей на позиции j:

$$\mathbf{w}(w_j) = \mathbf{e}(j) \in \mathbb{R}^{|W|}, \quad \mathbf{e}(j) = [0 \ldots 0\, 1\, 0 \ldots 0].$$

При этом сходство между двумя произвольными словами равно 1, если слова совпадают, и 0, если слова разные:

$$s_{\mathsf{w}}(w_i, w_j) = [i = j].$$

Чтобы сохранить информацию о синонимичности слов, для слов-синонимов используются схожие векторы. Для описания процесса оптимизации векторов $\mathbf{w}(w)$, рассмотрим языковую модель (LM, англ. Language Modeling).

Языковая модель. Языковая модель применяется в распознавании речи, распознавании печатного и рукописного текста, в области машинного перевода и проверки орфографии [71, 72, 73, 74, 75]. Она определяет вероятность $p(w_1, w_2, \ldots, w_n)$ последовательности слов $w_1 w_2 \ldots w_n$. В [76] предполагается, что вероятность слова зависит только от предыдущих слов:

$$p(w_1, w_2, \ldots, w_n) = p(w_1) \cdot p(w_2|w_1) \cdot \ldots \cdot p(w_n|w_{n-1}, \ldots, w_1). \tag{1.1}$$

Оценить $p(w_i|w_{i-1}, \ldots, w_1)$ при больших i сложно, поэтому предполагается, что слово зависит только от двух предыдущих. Данное предположение называется триграммным. Условные вероятности в (1.1) оцениваются как

$$p(w_i|w_{i-1}, \ldots, w_1) = p(w_i|w_{i-1}, w_{i-2}) = \frac{N(w_i w_{i-1} w_{i-2}, D)}{N(w_{i-1} w_{i-2}, D)}, \tag{1.2}$$

где $N(w_i w_{i-1} \ldots, D)$ – число последовательностей слов $w_i, w_{i-1}, \ldots$ в коллекции D.

Матрица оценок вероятностей $p(w_i|w_{i-1}, \ldots, w_1)$ будет разреженной, так как многие тройки слов не встретятся в D. Для решения проблемы разреженности были предложены различные формы дисконтирования, перераспределяющие вероятность встретившихся троек на другие схожие тройки слов. Применяется дисконтирование Катца [77] или Джелинека-Мерсера [78]. В [79] показывается,

что дисконтирование Кнесера-Нея [80] превосходит по качеству остальные техники дисконтирования, а в [76] приводится теоретическое обоснование этого. Для увеличения числа последовательностей слов в [70] рассматриваются всевозможные триграммы с пропусками.

Распределенное представление слова. Альтернативными методами оценки вероятности (1.1) являются модели [81, 82, 41, 42], использующие распределенное представление слов в виде действительных векторов $\mathbf{w}(w)$.

Пусть каждому слову w из словаря W соответствует вектор $\mathbf{w}(w) \in \mathbb{R}^m$, а в матрице $\mathbf{W}$ размера $|W| \cdot m$ строка с номером i соответствует векторному представлению слова с номером i из словаря W. Пусть w_t – слово с порядковым номером t в документе. Вероятность слова w_t быть словом w_i из словаря W при заданной последовательности предшествующих ему слов $w_{t-1} \dots w_{t-n+1}$ задается как i-й элемент параметрической вектор-функции $\mathbf{f} \in \mathbb{R}^{|W|}$:

$$p(w_t = w_i | w_{t-1}, \dots, w_{t-n+1}) = f\big(\mathbf{w}(w_{t-1}), \dots, \mathbf{w}(w_{t-n+1}), \boldsymbol{\theta}\big)_i.$$

В [81] функция $\mathbf{f}$ представляется в виде нейронной сети, показанной на рис. 1.1:

$$\mathbf{f}\big(\mathbf{w}(w_{t-1}), \dots, \mathbf{w}(w_{t-n+1}), \boldsymbol{\theta}\big) = \mathbf{f}\big(\mathbf{y}(\mathbf{x}, \boldsymbol{\theta})\big), \quad \text{где} \tag{1.3}$$

$$\mathbf{y}(\mathbf{g}, \boldsymbol{\theta}) = \mathbf{b} + \mathbf{V}\mathbf{x} + \mathbf{U}\mathbf{g}, \quad \mathbf{g} = \tanh(\mathbf{a}), \quad \mathbf{a} = \mathbf{d} + \mathbf{H}\mathbf{x}, \tag{1.4}$$

$$\mathbf{x} = [\mathbf{w}(w_{t-1}), \mathbf{w}(w_{t-2}), \dots, \mathbf{w}(w_{t-n+1})]^\intercal. \tag{1.5}$$

Здесь $\mathbf{x}$ – конкатенация векторных представлений слов, полученных из матрицы $\mathbf{W}$, $\mathbf{y}(\mathbf{x}, \boldsymbol{\theta})$ – вектор значений предактивации выходного слоя, $\mathbf{g}$ – вектор значений активации скрытого слоя, $\mathbf{a}$ – вектор значений предактивации скрытого слоя, $\mathbf{b}$ – вектор констант нейронов выходного слоя, $\mathbf{V}$ – матрица весов связей, соединяющих напрямую векторные представления слов с выходным слоем, $\mathbf{U}$ – матрица весов соединений между скрытым слоем и выходным слоем, $\mathbf{H}$ – матрица весов соединений входного слоя векторного представления слов $\mathbf{x}$ со скрытым слоем, а $\mathbf{d}$ – вектор констант нейронов выходного слоя.

В качестве функции активации выходного слоя используется функция $\mathrm{softmax}(\mathbf{y})$:

$$\mathbf{f}(\mathbf{y}) = \left[\frac{\exp(y_1)}{\sum_{i=1}^{|W|}\exp(y_i)}, \dots, \frac{\exp(y_{|W|})}{\sum_{i=1}^{|W|}\exp(y_i)}\right]^\intercal. \tag{1.6}$$

Число параметров данной модели $|W|(1+nm+h)+h(1+(n-1)m)$. Наибольший вес в данную сумму вносит член $|W|nm$. Таким образом, число свободных параметров растет линейно с размером словаря W, размерностью пространства векторного представления слов m и числом слов в рассматриваемых словосочетаниях.

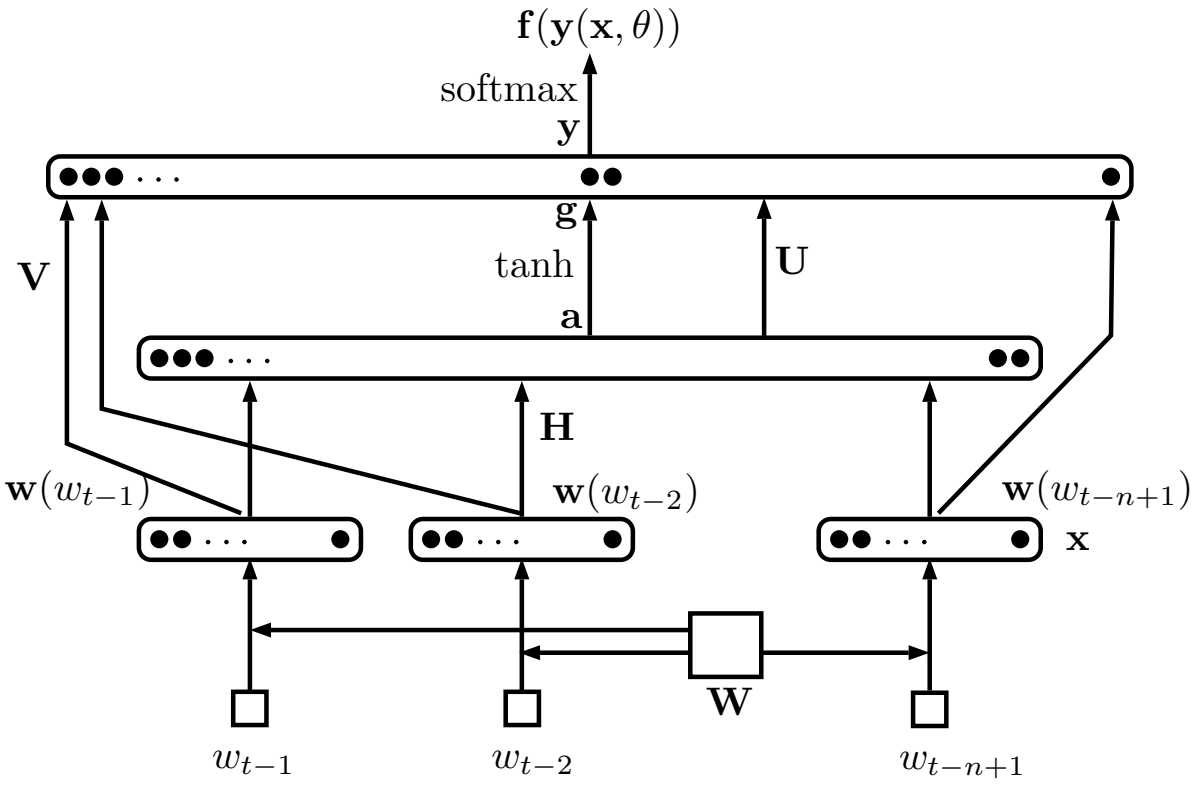


Рис. 1.1. Представление функции $\mathbf{f}$ в виде нейронной сети.

Качеством модели является логарифм правдоподобия последовательностей слов заданной длины из коллекции D:

$$L = \frac{1}{T}\sum_{t} \log p(w_t|w_{t-1},\ldots,w_{t-n+1}). \tag{1.7}$$

Оптимизация параметров нейронной сети. Для удобства, структура аргументов функции $\mathbf{f}$ опускается, и данная функция обозначается как $\mathbf{f}(\mathbf{x})$. Согласно (1.5) и (1.7), функция потерь для последовательности слов $w_t w_{t-1}\ldots,w_{t-n+1}$ имеет вид

$$\ell\big(\mathbf{f}(\mathbf{x}), w_t\big) = -\sum_{t'=1}^{|W|}[t=t']\log f(\mathbf{x})_{t'} = -\log f(\mathbf{x})_t. \tag{1.8}$$

Утверждение 1. Производные логарифмической функции потерь (1.8) по параметрам $\boldsymbol{\theta} = (\mathbf{b},\mathbf{d},\mathbf{V},\mathbf{U},\mathbf{H},\mathbf{W})$ для слова w_t равны

$$\frac{\partial}{\partial\theta_l}\ell\big(f(\mathbf{x}),w_t\big) = \sum_{k=1}^{|W|}\big([k=t]-f(\mathbf{x})_k\big)\frac{\partial y_k}{\partial\theta_l}, \quad \text{где}$$

$$\frac{\partial y_k}{\partial b_l} = [l=k], \qquad \frac{\partial y_k}{\partial v_{lm}} = [l=k]x_m, \qquad \frac{\partial y_k}{\partial u_{lm}} = [l=k]g_m, \tag{1.9}$$

$$\frac{\partial y_k}{\partial d_l} = \frac{4u_{kl}}{\big(\exp(a_l) + \exp(-a_l)\big)^2}, \qquad \frac{\partial y_k}{\partial h_{lm}} = \frac{4u_{kl}x_m}{\big(\exp(a_l) + \exp(-a_l)\big)^2}, \tag{1.10}$$

$$\frac{\partial y_k}{\partial W_{t,m}} = v_{km} + \sum_{i=1}^{h} \frac{4u_{ki}h_{im}}{\big(\exp(a_i) + \exp(-a_i)\big)^2}. \tag{1.11}$$

Доказательство. Пусть θ_j – один из параметров $(\mathbf{b}, \mathbf{d}, \mathbf{V}, \mathbf{U}, \mathbf{H}, \mathbf{C})$. Частные производные функции потерь по $f(\mathbf{x})_i$

$$\frac{\partial}{\partial f(\mathbf{x})_i}\ell\big(f(\mathbf{x}), w_t\big) = \frac{[i=t]}{f(\mathbf{x})_t}, \quad \nabla_{f(\mathbf{x})}\ell\big(f(\mathbf{x}), w_t\big) = -\frac{\mathbf{e}(t)}{f(\mathbf{x})_t},$$

где $\mathbf{e}(t)$ – единичный вектор с единицей на позиции t. Производная элемента m вектора softmax

$$\frac{\partial}{\partial y_k}\left(\frac{\exp(y_m)}{\sum_{m'}\exp(y_{m'})}\right) = \frac{\exp(y_m)}{\sum_{m'}\exp(y_{m'})}\left([k=m] - \frac{\exp(y_k)}{\sum_{m'}\exp(y_{m'})}\right) = \\ = f(\mathbf{x})_m\big([k=m] - f(\mathbf{x})_k\big). \tag{1.12}$$

Тогда производные функции потерь по значениям функции предактивации выходного слоя y_k

$$\frac{\partial}{\partial y_k}\ell\big(f(\mathbf{x}), w_t\big) = \sum_{k'=1}^{|W|} \frac{\partial\ell\big(f(\mathbf{x}), w_t\big)}{\partial f(\mathbf{x})_{k'}}\frac{\partial f(\mathbf{x})_{k'}}{\partial y_k} = \\ = \sum_{k'=1}^{|W|} \frac{[k'=t]f(\mathbf{x})_{k'}([k=k'] - f(\mathbf{x})_k)}{f(\mathbf{x})_t} = [k=t] - f(\mathbf{x})_k. \tag{1.13}$$

Согласно (1.4) частные производные y_k по параметрам $\mathbf{b}, \mathbf{V}, \mathbf{U}$ записываются как

$$\frac{\partial y_k}{\partial b_l} = [l=k], \qquad \frac{\partial y_k}{\partial v_{lm}} = [l=k]x_m, \qquad \frac{\partial y_k}{\partial u_{lm}} = [l=k]g_m, \tag{1.14}$$

Вспомогательные производные имеют вид

$$\frac{\partial y_k}{\partial \theta_l} = \sum_{p=1}^{h} \frac{\partial y_k}{\partial g_p}\frac{\partial g_p}{\partial a_p}\frac{\partial a_p}{\partial \theta_l}, \qquad \frac{\partial y_k}{\partial g_p} = u_{kp}, \tag{1.15}$$

$$\frac{\partial g_p}{\partial a_p} = \frac{\partial \tanh(a_p)}{\partial a_p} = \frac{4}{\big(\exp(a_p) + \exp(-a_p)\big)^2}. \tag{1.16}$$

Частные производные y_k по параметрам $\mathbf{d}, \mathbf{H}, \mathbf{x}$ равны

$$\frac{\partial y_k}{\partial d_l} = \frac{4u_{kl}}{\big(\exp(a_l) + \exp(-a_l)\big)^2}, \qquad \frac{\partial y_k}{\partial h_{lm}} = \frac{4u_{kl}x_m}{\big(\exp(a_l) + \exp(-a_l)\big)^2}, \tag{1.17}$$

$$\frac{\partial y_k}{\partial x_m} = v_{km} + \sum_{i=1}^{h} \frac{4u_{ki}h_{im}}{\big(\exp(a_i) + \exp(-a_i)\big)^2}. \tag{1.18}$$

Таким образом, производная функции потерь по параметру θ_l выражается как

$$\frac{\partial}{\partial \theta_l}\ell\big(f(\mathbf{x}), w_t\big) = \sum_{k=1}^{|W|}\big([k=t] - f(\mathbf{x})_t\big)\frac{\partial y_k}{\partial \theta_l},$$

где $\partial y_k/\partial\theta_l$ берутся из (1.14), (1.17), и (1.18). □

Для оптимизации параметров $\boldsymbol{\theta} = [\mathbf{b}, \mathbf{d}, \mathbf{V}, \mathbf{U}, \mathbf{H}, \mathbf{W}]$ используется метод стохастического градиентного спуска (англ. SGD):

$$\theta'_j \leftarrow \theta_j + \epsilon\frac{\partial \log \ell\big(f(\mathbf{x}), w_t\big)\big)}{\partial \theta_j}. \tag{1.19}$$

На каждом шаге SGD изменяется лишь небольшая часть элементов матрицы $\mathbf{W}$ – только строки тех слов, которые встретились в последовательности $w_t w_{t-1} \dots w_{t-n+1}$, полученной SGD на вход.

Основной вклад в вычислительную сложность данного алгоритма вносит вычисление суммы в знаменателе softmax. В [82, 41] были предложены иерархические варианты функции softmax, что снижало вычислительную сложность на величину, пропорциональную $|W|/\log(|W|)$. В [83, 84] предлагались подходы, позволяющие избежать вычисления данной суммы. В [81] предлагался способ распараллеливания алгоритма обучения нейронной сети (1.3).

В [42, 85] рассматриваются две альтернативные структуры нейронной сети без скрытого слоя: CBOW (англ. continuous bag-of-words) и Skip-gram, показанные на рис. 1.2 и 1.3 соответственно. В модели CBOW векторные представления слов усредняются и подаются сразу на выходной слой. В отличие от оригинального алгоритма, вычисляется вероятность среднего w_t слова в последовательности как по предыдущим $w_{t-2}w_{t-1}$ так и по последующим $w_{t+1}w_{t+2}$. В модели Skip-gram по текущему слову w_t в последовательности считается вероятность соседних с ним слов $w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2}$.

За счет отсутствия скрытого слоя вычислительная сложность данной модели уменьшается, что позволяет обучать ее на больших объемах данных, не уменьшая при этом размерность векторного представления слов. В [42] данные модели обучались на коллекции из $6 \cdot 10^9$ слов, используя векторы слов $\mathbf{w}(w) \in \mathbb{R}^{1000}$, в то время как алгоритм со скрытым слоем использовал пространство с размерностью в десять раз меньше $\mathbf{w}(w) \in \mathbb{R}^{100}$.

Для сравнения качества представления слов в виде векторов используются семантические и синтаксические списки пар слов, составленные экспертно. Алгоритму необходимо предсказать второе слово, используя первое. Ответ считается верным в случае совпадения второго слова. Алгоритмы CBOW и Skip-gram в [42] показали более высокие результаты.

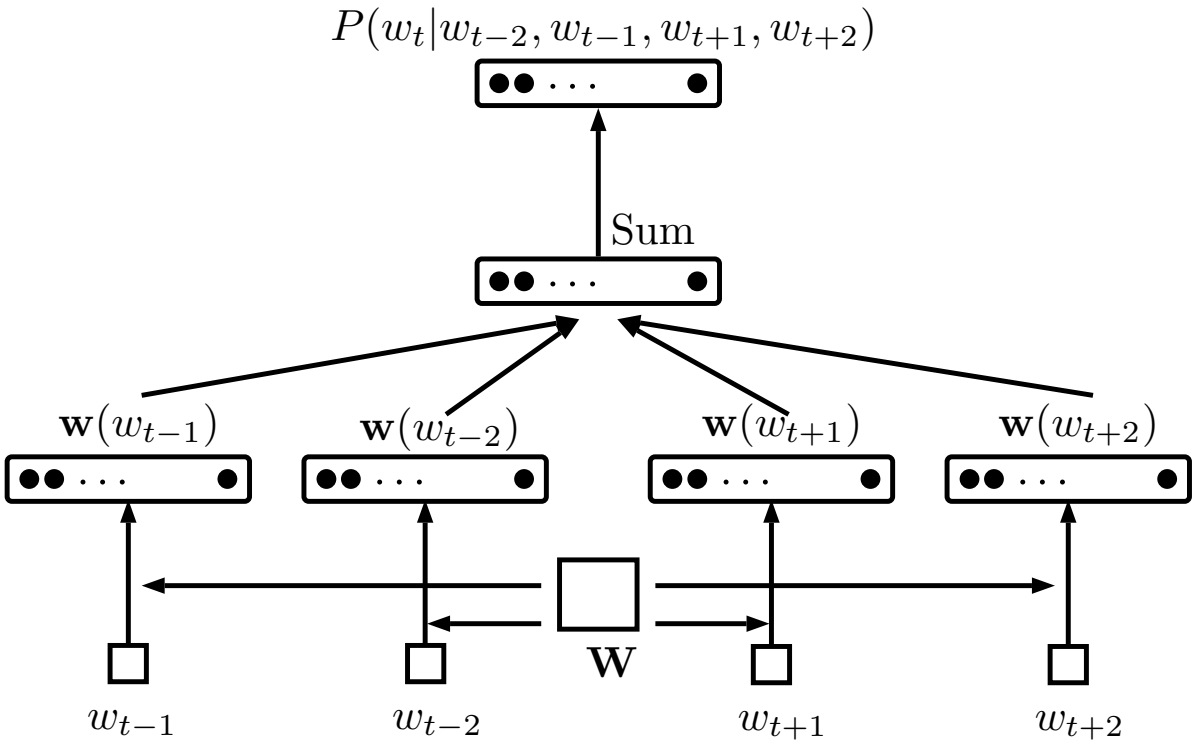


Рис. 1.2. Модель CBOW.

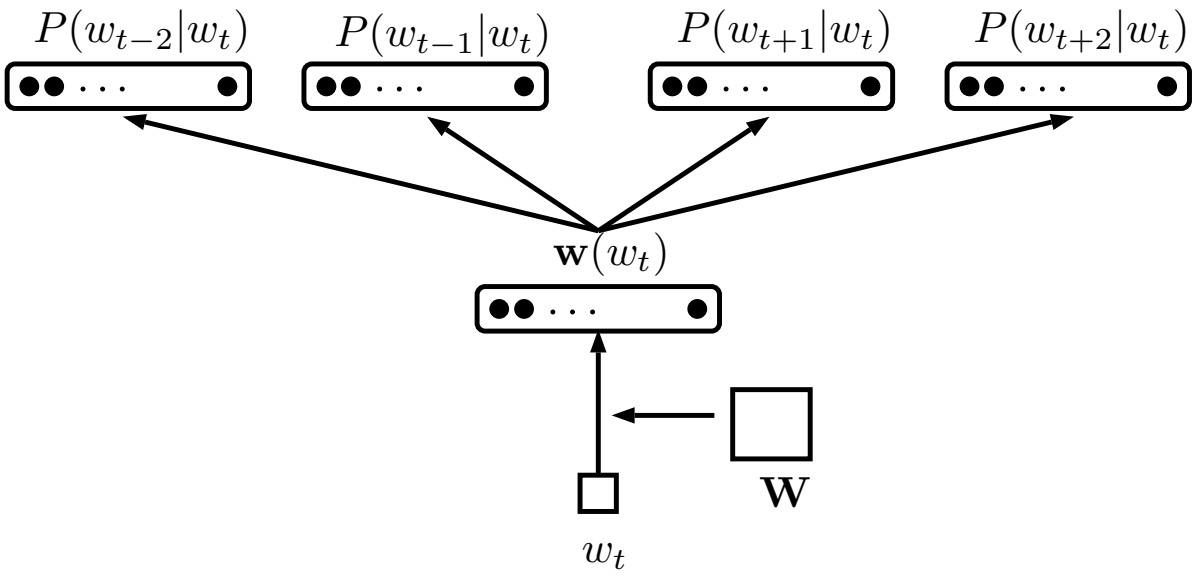


Рис. 1.3. Модель Skip-gram.

1.4. Представление документа в виде вектора

Для решения задач кластеризации и классификации, документы d представляются в виде векторов $\mathbf{x} \in \mathbb{R}^{|W|}$ [16, 86, 14]:

$$\mathbf{x} = [\phi(w_1, d, D), \cdots, \phi(w_{|W|}, d, D)]^{\intercal}, \tag{1.20}$$

где $\phi(w_m, d, D)$ – функция, ставящая в соответствие слову w_m из W действительное число.

В [87] документ d представляется в виде целочисленного вектора $\mathbf{x} \in \mathbb{R}^{|W|}$, где на позиции m стоит $x_m = N(w_m, d)$ – число слов w_m из словаря W в d. Предполагается, что порядок слов в документе не несет дополнительной информации для кластеризации и классификации. В [49] данный подход сравнивался с представлением документа в виде бинарного вектора, где $x_m = [N(w_m, d) > 0]$.

Для учета важности слов при решении задач классификации и кластеризации используются:

1) взвешенные метрики или функции сходства [25],
2) алгоритмы отбора признаков [19, 88, 89, 90],
3) функции $\phi(w, d, D)$, учитывающие частотные особенности слов в коллекции [86].

В последнем подходе используются комбинации частотных показателей слов $\mathrm{tf} \cdot \mathrm{idf}$ (англ. tf – term frequency, idf – inverse document frequency) [91]:

$$\mathrm{tf}(w, d) = \frac{N(w, d)}{|d|}, \quad \mathrm{idf}(w, D) = \log \frac{|D|}{\sum_d [N(w, d) > 0]}, \tag{1.21}$$

где $|d|$ – число слов в документе d. Элемент x_m представляется как

$$x_m = \phi(w_m, d, D) = \mathrm{tf}(w_m, d) \cdot \mathrm{idf}(w_m, D).$$

В [86] ищется оптимальный вид функции $\phi(w_m, d, D)$ методом порождения моделей [92]. Комбинация (1.22) нормализованных tf и idf (ntf и ndf соответственно) сравнивается с (1.21).

$$\text{ES-LG}(\mathrm{ntf}, \mathrm{ndf}) = \exp\left(\sqrt{\log\left(\frac{\mathrm{ntf} + \mathrm{ndf}}{\mathrm{ndf}}\right)}\right), \quad \text{где} \tag{1.22}$$

$$\mathrm{ntf}(w, d, D) = N(w, d) \log\left(1 + \varphi \frac{\sum_{d'} N(w, d')}{|D| N(w, d)}\right), \tag{1.23}$$

$$\mathrm{ndf} = \frac{\sum_d [N(w, d) > 0]}{|D|}. \tag{1.24}$$

В выражении (1.23) структурный параметр φ оптимизируется по коллекции D.

Распределенное представление документа. Недостатками представления (1.20) являются большая размерность получаемых векторов и отсутствие возможности учитывать синонимичность слов. В [93] предлагается метод paragraph vector – альтернативный способ векторного представления документов. Для этого к нейронной сети CBOW, показанной на рис. 1.2, в качестве еще

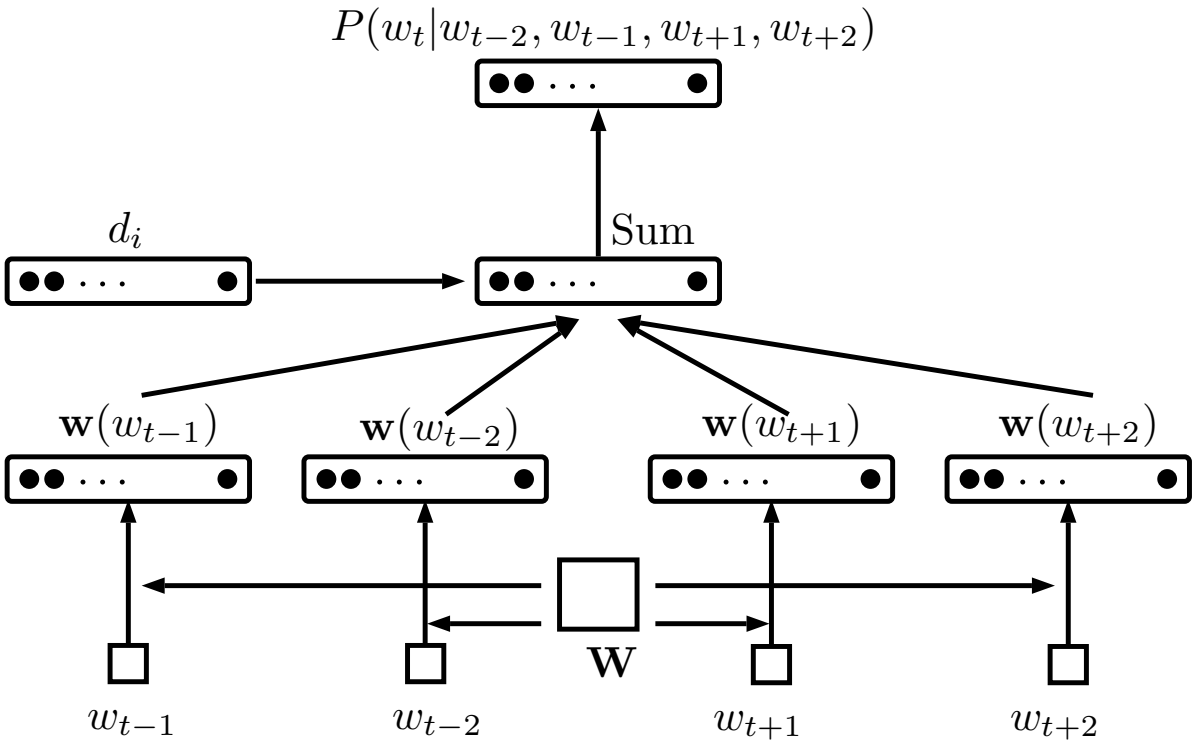


Рис. 1.4. Модель paragraph vector.

одного входа добавляется общий вектор для всех обучающих последовательностей слов из фиксированного документа d_i, рис. 1.4. После обучения, этот вектор является представлением документа d_i.

В [94] используется рекурсивная нейронная сеть для свертки векторных представлений слов предложения в один вектор, соответствующий этому предложению. После этого, векторные представления предложений с помощью рекурсивной нейронной сети сворачиваются в векторное представление документа.

1.5. Жесткие иерархические модели

Алгоритмы текстовой кластеризации разделяются на четыре типа по тому, каким способом они описывают документ и кластер в коллекции [6], см. таблицу 1.1.

Определение 11. Тематическая модель M коллекции документов D называется жесткой, если каждый документ $d \in D$ принадлежит только одному кластеру нижнего уровня и всем его родительским кластерам.

Жесткие модели ищут класс для каждого документа. Описательно-вероятностные модели являются расширением жестких моделей. В них дополнительно оценивается вероятность принадлежности каждого документа каждому из классов. В смесях моделей классы представляются в виде распределений

Таблица 1.1. Основные типы алгоритмов текстовой кластеризации.

Тип моделей	Документ	Кластер	Пример алгоритма
Жесткие	вектор	вектор	k-means [20], SVM [11]
Описательно-вероятностные	вектор	вероятность	DPM [6], nDPM, hDPM [21], Probabilistic SVM [95], Нейронные сети [37]
Смеси	вектор	распределение	mixture of Gaussian [96], vMF [7]
Вероятностные	распределение	распределение	LDA [1], PAM [51], hPAM [52], HDP [4], hHDP [23], ARTM [22]

векторных представлений документов. В вероятностных моделях и документы и классы являются распределениями. Так, в [1] документы являются распределениями над классами, классы являются распределениями над словарем коллекции.

Для построения жестких тематических моделей применяются алгоритмы кластеризации произвольных объектов в метрическом или неметрическом [97, 98] пространстве. Документы представляются в виде векторов. Для построения тематической модели M выбирается функция расстояния или сходства векторов документов, при помощи которой документы сравниваются и объединяются в кластеры.

В [25] рассматривается способ применения взвешенных метрик Минковского в качестве функции расстояния (1.25):

$$\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}) = \sqrt[p]{\sum_{m=1}^{|W|} |\lambda_m|^p |x_m - y_m|^p}, \tag{1.25}$$

где λ_m есть важность слова w_m из словаря W при кластеризации и классификации. Чтобы убрать зависимость метрики Минковского от числа слов в документе, векторы документов нормализуются:

$$\mathbf{x} \mapsto \frac{\mathbf{x}}{||\mathbf{x}||}.$$

В [26] для сравнения векторов документов используется взвешенная коси-

нусная функция сходства (1.26):

$$s(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x}^\intercal \mathbf{\Lambda} \mathbf{y}}{\sqrt{\mathbf{x}^\intercal \mathbf{\Lambda} \mathbf{x}} \sqrt{\mathbf{y}^\intercal \mathbf{\Lambda} \mathbf{y}}}, \quad \text{где} \quad \mathbf{\Lambda} = \operatorname{diag}(\lambda_1, \cdots, \lambda_{|W|}). \tag{1.26}$$

При большом числе ненулевых элементов в матрице весов $\mathbf{\Lambda}$ общего вида возникает проблема переобученности, так как число оптимизируемых параметров растет пропорционально $|W|^2$. Чтобы избежать этого в качестве $\mathbf{\Lambda}$ используется диагональная матрица. В [16, 14] используется частный случай (1.26) – косинусная функция сходства с $\mathbf{\Lambda} = \mathbf{I}$.

Утверждение 2. При единичной матрице $\mathbf{\Lambda} = \mathbf{I}$ взвешенное косинусное расстояние, задаваемое как $\rho_s(\cdot, \cdot) = 1 - s(\cdot, \cdot)$, не является метрикой.

Доказательство. Действительно, пусть

$$\mathbf{x} = (1, 0, 0),\ \mathbf{y} = (1, 1, 0),\ \mathbf{z} = (0, 1, 0),\ \mathbf{\Lambda} = \operatorname{diag}(1, 1, 1).$$

Для заданных точек для ρ_s не выполняется неравенство треугольника:

$$\rho_s(\mathbf{x}, \mathbf{y}) + \rho_s(\mathbf{x}, \mathbf{z}) < 0.59 < 1 = \rho_s(\mathbf{x}, \mathbf{z}).$$

□

В [99] с помощью косинусной функции сходства при $\mathbf{\Lambda} = \operatorname{diag}(1, \ldots, 1)$ задается ангулярное расстояние $\rho'_s(\mathbf{x}, \mathbf{y})$, которое задает метрику.

$$\rho'_s(\mathbf{x}, \mathbf{y}) = 1 - \frac{2 \cdot \cos^{-1}\big(s(\mathbf{x}, \mathbf{y})\big)}{\pi}.$$

Однако для произвольной диагональной матрицы $\mathbf{\Lambda}$ метрические свойства не всегда выполняются.

Утверждение 3. При матрице $\mathbf{\Lambda} \neq \mathbf{I}$, ρ'_s не является метрикой в общем случае.

Доказательство. Пусть

$$\mathbf{\Lambda} = \operatorname{diag}(1, 1, 10),\ \mathbf{x} = (1, 0, 1),\ \mathbf{y} = (1, 1, 0),\ \mathbf{z} = (0, 1, 1).$$

Для заданных точек для ρ'_s не выполняется неравенство треугольника. □

Утверждение 4. При $\mathbf{\Lambda} = \mathbf{I}$, $p = 2$ и нормированных векторах документов $\mathbf{x} \mapsto \frac{\mathbf{x}}{||\mathbf{x}||}$, метрика Минковского и функция сходства совпадают с точностью до линейного преобразования:

Доказательство.

$$\rho(\mathbf{x},\mathbf{y}) = ||\mathbf{x}-\mathbf{y}||^2 = (\mathbf{x}-\mathbf{y})(\mathbf{x}-\mathbf{y}) = \\ = ||\mathbf{x}||^2 + ||\mathbf{y}||^2 - 2(\mathbf{x}\cdot\mathbf{y}) = 2\big(1 - s(\mathbf{x},\mathbf{y})\big). \quad (1.27)$$

□

Алгоритм плоской кластеризации. Рассмотрим двухуровневую тематическую модель M, состоящую из корня $c_{1,1}$ и кластеров второго уровня $\{c_{2,k}\}, k \in \{1 \dots K_2\}$, где K_2 – фиксированное число кластеров второго уровня. Обозначим за $\{\boldsymbol{\mu}(c_{2,k})\}$ центры этих кластеров. Пусть $\hat{c}(\mathbf{x})$ – кластер, к которому алгоритм отнес документ $\mathbf{x}$. Задача кластеризации формулируется следующим образом:

$$\ell(M) = \frac{1}{|D|}\sum_{n=1}^{|D|} \rho\left(\mathbf{x}_n, \boldsymbol{\mu}\big(\hat{c}(\mathbf{x}_n)\big)\right) \to \min_M, \quad (1.28)$$

где M берется по всем возможным разбиениям коллекции D, согласно определению 7. Требуется найти такое разбиение документов коллекции D на кластеры, чтобы центроиды кластеров построенной модели M доставляли минимум функционалу $\ell(M)$.

Для решения данной задачи применяется алгоритм k-means [20, 100, 36]. В [24] приводится обобщенный вариант данного алгоритма, адаптирующий шаг пересчета координат центров кластеров для произвольной функции расстояния:

1) инициализировать положения центров $\boldsymbol{\mu}(c_{2,k})$, выбрав случайным образом векторы k_2 документов,
2) присвоить каждому документу $\mathbf{x}$ метку кластера ближайшего центроида,
3) обновить положения центроидов кластеров:

$$\boldsymbol{\mu}(c_{2,k}) := \arg\min_{\boldsymbol{\mu}(c_{2,k})} \sum_{n:\hat{c}(\mathbf{x}_n)=c_{2,k}} \rho\left(\mathbf{x}_n, \boldsymbol{\mu}\big(\hat{c}(\mathbf{x}_n)\big)\right),$$

4) если центроид хотя бы одного кластера $c_{k,2}$ изменился, вернуться на второй шаг.

Шаги 2 и 3 данного алгоритма не увеличивают целевую функцию $\ell(M)$. Поэтому, так как существует лишь конечное число разбиений $|D|$ объектов на k_2 кластеров, алгоритм гарантированно сойдется за конечное число шагов.

Недостатком алгоритма является сходимость к локальному минимуму, поэтому предлагается несколько раз запустить алгоритм с различными начальными условиями. В [101] для поиска локальных минимумов с меньшим значением $\ell(M)$ предлагается некоторым документам присваивать метки случайных кластеров на каждом шаге.

В [97] предлагается линейный приближенный алгоритм для решения задачи (1.28). В качестве функции расстояния при этом используется расстояние Кульбака-Лейблера [102], расстояние Итакуры-Саито, расстояние Махалонобиса [103] и некоторые случаи дивергенции Брегмана [104].

Иерархическая кластеризация. Существует два типа алгоритмов жесткой иерархической кластеризации.

Дивизимные [11, 37]: изначально, все документы находятся в одном кластере совпадающем с вершиной дерева иерархической кластерной структуры. При построении каждого следующего уровня $l+1$ каждый кластер уровня l делится на кластеры меньшего размера, например, методом k-means.

Агломеративные [50]: изначально, все документы рассматриваются как отдельные кластеры уровня $h+1$, при построении более высоких уровней $l < h+1$ центроиды кластеров уровня $l+1$ рассматриваются как объекты, которые объединяются в кластеры уровня l.

1.6. Вероятностные модели

После предобработки коллекции каждому документу $d \in D$ ставится в соответствие вектор $\mathbf{x}_d$, где $x_{w,d}$ – число слов w в документе d. Пусть $T = \{t_i\}$ – множество тем и каждое слово w принадлежит теме t с вероятностью $p(w|t)$. Предполагается, что слова в документе независимы. Вероятность появления слова w из темы t в документе d описывается дискретным распределением $p(d, w, t)$ на $D \times W \times T$.

Пусть заданы условные вероятности $p(w|t)$ и $p(t|d)$. Процесс генерации нового документа описывается следующим алгоритмом:

1) выбрать длину документа d: $N \sim \text{Poisson}$,
2) для каждого из N слов документа d
   выбрать тему из $p(t|d)$,
   выбрать слово w из $p(w|t)$.

В [1] рассматривается обратная задача: по существующей текстовой коллекции D найти вероятности $p(w|t)$ и $p(t|d)$. При этом документы d и слова w являются наблюдаемыми переменным, а темы t являются скрытыми или латентными переменными.

Одной из гипотез, на которой построены методы вероятностного тематического моделирования, является гипотеза условной независимости [1]: вероятность $p(w|t)$ не зависит от документа d. Она формализуется как $p(w|t, d) = p(w|t)$. Это эквивалентно

$$p(w, d|t) = \frac{p(w|d, t)p(d, t)}{p(t)} = p(w|t)p(d|t). \tag{1.29}$$

Согласно гипотезе условной независимости:

$$p(w|d) = \frac{p(w,d)}{p(d)} = \frac{\sum_t p(w,d|t)p(w)}{p(d)} = \frac{\sum_t p(w,d|t)p(w)}{p(d)} = \sum_t p(t|d)p(w|t).$$

Число n_{wd} слов w в документе d и общее число слов в документе n_d задаются как:

$$n_{wd} = N(w,d), \quad n_d = \sum_w N(w,d).$$

Матрицей частот коллекции D называется:

$$\mathbf{F} = (\hat{p}_{wd})_{W\times D}, \text{ где } \hat{p}_{wd} = \hat{p}(w|d) = \frac{n_{wd}}{n_d}.$$

Задача поиска неизвестных вероятностей $p(w|t)$ и $p(t|d)$ сводится к поиску разложения матрицы $\mathbf{F}$ на матрицу слов-тем $\mathbf{\Phi}$ и матрицу тем-документов $\mathbf{\Theta}$:

$$\mathbf{F} \approx \mathbf{\Phi\Theta}, \tag{1.30}$$

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

Модель PLSA. В модели PLSA [1] максимизируется логарифм правдоподобия коллекции D при ограничениях нормировки и неотрицательности:

$$L(\mathbf{\Phi},\mathbf{\Theta}) = \ln\prod_{d\in D}\prod_{w\in W} p(w|d)^{n_{dw}} = \sum_{d\in D}\sum_{w\in W} n_{dw}\ln\sum_{t\in T}\phi_{wt}\theta_{td} \to \max_{\Phi,\Theta}, \tag{1.31}$$

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

В [105] приводится вывод оценок искомых переменных θ_{td} и ϕ_{wt} с помощью метода множителей Лагранжа. Доказывается, что стационарная точка (1.31) удовлетворяет системе уравнений:

$$p_{tdw} = \frac{\phi_{wt}\theta_{td}}{\sum_{s\in T}\phi_{ws}\theta_{sd}},$$

$$\phi_{wt} = \frac{n_{wt}}{n_t}, \quad n_{wt} = \sum_{d\in D} n_{dw}p_{tdw}, \quad n_t = \sum_{w\in W} n_{wt},$$

$$\theta_{td} = \frac{n_{td}}{n_d}, \quad n_{td} = \sum_{w\in D} n_{dw}p_{tdw}, \quad n_d = \sum_{t\in T} n_{td}.$$

Данная задача решается EM-алгоритмом [1]:

1) на E шаге по текущим θ_{td} и ϕ_{wt} вычисляются p_{tdw},

2) на M шаге по p_{tdw} вычисляются новые оценки для θ_{td} и ϕ_{wt}.

В [105] приводится вариант рационального EM-алгоритма, в котором E-шаг встраивается в M-шаг, при этом отпадает необходимость хранить в памяти всю трехмерную матрицу значений p_{tdw}. Недостатками данного подхода является наличие большого числа параметров [2], что может приводить к переобученности. Одним из способов предотвращения переобучения является введение различных вариантов регуляризации [22].

Модель LDA. Альтернативным вариантом уменьшения числа параметров является введение априорного предположения о виде распределений. В [2] предполагается, что векторы документов $\boldsymbol{\theta}_d$ порождаются распределениями Дирихле с параметрами $\boldsymbol{\alpha} \in \mathbb{R}^{|T|}$. Процесс порождения документа принимает следующий вид:

1) выбрать $\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha})$,
2) для каждого нового слова w_n
   выбрать тему $t \sim \mathrm{Mult}(\boldsymbol{\theta})$,
   выбрать слово $w_n \sim \mathrm{Mult}(\boldsymbol{\beta}_t)$.

В модели LDA в силу свойств распределения Дирихле неявно предполагается независимость тем, присутствующих в документе, что не всегда выполняется на практике [5]. Для учета корреляции тем в [5] предлагается для каждого документа выбирать темы из логнормального распределения [106] с заданным вектором средних значений $\boldsymbol{\varepsilon}$ и ковариационной матрицей $\boldsymbol{\Sigma}$. В этом случае процесс генерации документа выглядит следующим образом:

1) выбрать $\boldsymbol{\theta} \sim \ln \mathcal{N}(\boldsymbol{\varepsilon}, \boldsymbol{\Sigma})$,
2) для каждого слова w_n из N
   выбрать тему $t \sim \mathrm{Mult}(\theta)$,
   выбрать слово $w_n \sim \mathrm{Mult}(\boldsymbol{\beta}_t)$.

Однако логнормальное распределение не является сопряженным с мультиномиальным распределением, что усложняет байесовский вывод, поэтому для оптимизации параметров распределений используются приближенные методы.

Адаптивная регуляризация ARTM. PLSA и LDA не решают проблему корректного восстановления матриц $\boldsymbol{\Phi}$ и $\boldsymbol{\Theta}$ по отдельности, так как правдоподобие (1.31) зависит только от произведения $\boldsymbol{\Phi} \cdot \boldsymbol{\Theta}$ [22]. В результате оптимизации, матрицы $\boldsymbol{\Phi}$ и $\boldsymbol{\Theta}$ восстанавливаются с точностью до некоторого преобразования $\boldsymbol{\Phi} \cdot \mathbf{U}^{-1}$ и $\mathbf{U} \cdot \boldsymbol{\Theta}$. Для решения данной проблемы в [22] к функции правдоподобия (1.31) добавляются регуляризаторы $\Omega_i(\boldsymbol{\Phi}, \boldsymbol{\Theta})$ нужного вида в зависимости от прикладной задачи с неотрицательными коэффициентами τ_i:

$$\Omega(\boldsymbol{\Phi}, \boldsymbol{\Theta}) = \sum_i^r \tau_i \Omega_i(\boldsymbol{\Phi}, \boldsymbol{\Theta}), \qquad L(\boldsymbol{\Phi}, \boldsymbol{\Theta}) + \Omega(\boldsymbol{\Phi}, \boldsymbol{\Theta}) \to \max_{\boldsymbol{\Phi}, \boldsymbol{\Theta}}. \tag{1.32}$$

Теорема 1 (Воронцов К. В.)**.** Если функция $\Omega(\mathbf{\Phi}, \mathbf{\Theta})$ непрерывно дифференцируема и $\mathbf{\Phi}, \mathbf{\Theta}$ – точка локального экстремума задачи (1.32), то для всех регулярных тем t и регулярных документов d справедлива система уравнений:

$$p_{tdw} = \frac{\phi_{wt}\theta_{td}}{\sum_{s\in T}\phi_{ws}\theta_{sd}}, \tag{1.33}$$

$$\phi_{wt} \propto \left(n_{wt} + \phi_{wt}\frac{\partial\Omega}{\partial\phi_{wt}}\right)_+, \qquad n_{wt} = \sum_{d\in D} n_{dw}p_{tdw}, \tag{1.34}$$

$$\theta_{td} \propto \left(n_{td} + \theta_{td}\frac{\partial\Omega}{\partial\theta_{td}}\right)_+, \qquad n_{td} = \sum_{w\in D} n_{dw}p_{tdw}. \tag{1.35}$$

При этом тема t называется регулярной, если $n_{wt} + \phi_{wt}\frac{\partial\Omega}{\partial\phi_{wt}} > 0$ хотя бы для одного слова $w \in W$. Аналогично, документ d называется регулярным, если $n_{td}+\theta_{td}\frac{\partial\Omega}{\partial\theta_{td}} > 0$ хотя бы для одной темы $t \in T$. В противном случае тема t или документ d называется перерегуляризированными. Из (1.34) и (1.35) следует, что перерегуляризированные темы и документы имеют $\boldsymbol{\phi}_t = \mathbf{0}$ и $\boldsymbol{\theta}_d = \mathbf{0}$ соответственно, поэтому эти темы и документы не учитываются в модели. Таким образом происходит автоматический отсев нерелевантных тем и документов.

Процесс Дирихле. В алгоритмах PLSA [1] и LDA [2] структура тем изначально задана. Однако в задачах построения тематических моделей неразмеченных коллекций изначально задать число тем экспертно не всегда возможно. В [4] предлагается метод, позволяющий избавиться от этого структурного параметра, используя иерархический процесс Дирихле в качестве модели генерации выборки документов. Пусть Dir – распределение Дирихле [29]. Процесс Дирихле определяется следующим образом [107].

Определение 12. Пусть задано измеримое пространство $\mathbb{R}^{|W|}$ с базовой вероятностной мерой G_0. Процессом Дирихле $\mathrm{DP}(\alpha_0, G_0)$ с параметром концентрации $\alpha_0 > 0$ является распределение случайной вероятностной меры G над $\mathbb{R}^{|W|}$, для которого выполняется следующее: для любого конечного разбиения $\mathbb{R}^{|W|}$ на измеримые непересекающиеся подмножества $\{A_i\}_{i=1}^r$, случайный вектор $[G(A_1), \ldots, G(A_r)]$ имеет конечномерное распределение Дирихле с параметрами $\alpha_0 G_0(A_1), \ldots, \alpha_0 G_0(A_r)$:

$$[G(A_1), \ldots, G(A_r)] \sim \mathrm{Dir}\big(\alpha_0 G_0(A_1), \ldots, \alpha_0 G_0(A_r)\big). \tag{1.36}$$

Утверждение 5 (Сесураман Д. [108])**.** С вероятностью единица реализация процесса Дирихле G является дискретной вероятностной мерой.

Доказательство. Плотность, соответствующая G, записывается в виде обобщенной функции

$$G(x) = \sum_{k=1}^{\infty} \pi_k \delta_{\boldsymbol{\theta}_k}(x), \quad \pi_k = \pi'_k \cdot \prod_{i=1}^{k-1}(1 - \pi'_i), \tag{1.37}$$

где $\boldsymbol{\theta}_k \sim G_0$, значения π'_k выбирается из бета-распределения $\mathrm{Beta}(1, \alpha_0)$, а $\delta_{\boldsymbol{\theta}}(x)$ – дельта-функция Дирака, понимаемая как вероятностная мера, сконцентрированная в $\boldsymbol{\theta}$. □

При генерации документа с помощью процесса Дирихле $\mathrm{DP}(\alpha_0, G_0)$ его слова разбиваются на темы. Согласно утверждению 5, мера $G \sim \mathrm{DP}(\alpha_0, G_0)$ является дискретным распределением над счетным набором векторов из $\mathbb{R}^{|W|}$. С помощью G для каждого нового слова выбирается вектор параметров темы $\boldsymbol{\theta}_i \in \mathbb{R}^{|W|}$ и новое слово генерируется из мультиномиального распределения с параметрами $\boldsymbol{\theta}_i$. Чтобы для всех документов векторы параметров тем были общими, в [4] предлагается алгоритм HDP, использующий иерархический процесс Дирихле. Процедура генерации нового документа для иерархического процесса Дирихле выглядит следующим образом.

1. Базовая мера G_0 определяется процессом Дирихле: $G_0 \sim \mathrm{DP}(\gamma, H)$. Она задает счетный набор векторов всевозможных тем для коллекции D.
2. Для каждого нового документа d_j генерируется вероятностная мера $G_j \sim \mathrm{DP}(\alpha_0, G_0)$, определяющая пропорции тем для данного документа. Таким образом, для всех документов используется общий набор тем, задаваемых G_0.
3. Для нового слова w_i согласно G_j генерируются параметры $\boldsymbol{\theta}_i \sim G_j$.
4. Генерируется новое слово $w_i \sim \mathrm{Mult}(\boldsymbol{\theta}_i)$.

Пусть на третьем шаге описанной выше процедуры уже выбрано k уникальных векторов тем $\{\boldsymbol{\theta}_j\}$. Согласно свойствам процедуры генерации [109, 110, 111] условное распределение вектора параметров темы $\boldsymbol{\theta}_i$ следующего слова w_i с учетом предыдущих реализаций $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{i-1}$ имеет вид:

$$\boldsymbol{\theta}_i | \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{i-1}, G_j \sim \sum_{a=1}^{i-1} \frac{1}{i - 1 + \alpha_0} \delta_{\boldsymbol{\theta}_a} + \frac{\alpha_0}{i - 1 + \alpha_0} G_j = \sum_{a=1}^{k} \frac{m_a}{i - 1 + \alpha_0} \delta_{\boldsymbol{\theta}_a} + \frac{\alpha_0}{i - 1 + \alpha_0} G_j, \tag{1.38}$$

где m_a – число слов с вектором темы $\boldsymbol{\theta}_a$. Таким образом, вектор параметров темы для нового слова w_i выбирается из уже встречавшихся k векторов тем с вероятностью

$$p_k = \frac{m_k}{i - 1 + \alpha_0}, \tag{1.39}$$

либо выбирается как вектор новой темы согласно G_j с вероятностью

$$p_{k+1} = \frac{\alpha_0}{i - 1 + \alpha_0}. \tag{1.40}$$

1.7. Иерархические вероятностные модели

Иерархические вероятностные модели позволяют учитывать и выявлять иерархическую структуру тем в коллекции D. Для этого делаются априорные предположения о типе распределений подтем и слов в темах более высокого уровня. Эти предположения определяют свойства иерархической модели, см. таблицу 1.2.

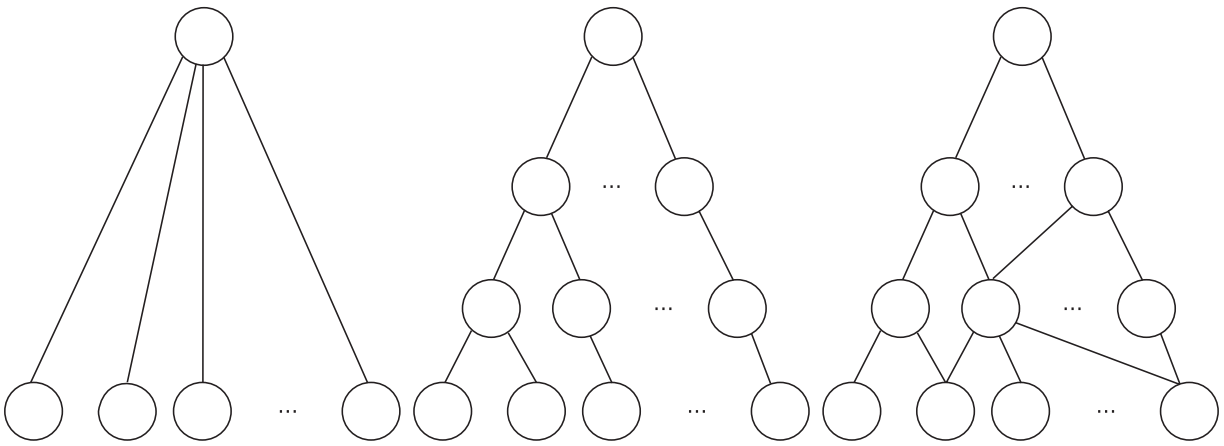
Таблица 1.2. Алгоритмы построения иерархических вероятностных тематических моделей.

	Структура в виде DAG	Внутренние узлы генерируют слова	Подбор числа кластеров на уровне	Подбор числа уровней
hLDA	−	+	+	−
PAM	+	−	−	−
HPAM	+	+	−	−
NPBPAM	+	−	+	−
hvHDP	−	+	+	+
htHDP	−	−	+	+

Как показано на рисунке 1.5 в., структура в виде DAG позволяет темам иметь общие дочерние темы [112]. Возможность генерировать слова во внутренних узлах позволяет анализировать слова, часто встречающиеся в теме данного узла. Возможность строить модели с нефиксированным числом кластеров на уровнях [3, 35] и с нефиксированным числом уровней в иерархии тем [23] достигается с помощью использования схожих с (1.39) и (1.40) процедур генерации слов и тем.

Модель hLDA. В [3] предлагается использовать древовидную структуру тем. Пусть $t_{1,1}$ корневая тема иерархической структуры тем с h уровнями, а каждая тема $t_{l,k}$ имеет счетное число возможных дочерних подтем. Процесс генерации документа $d \in D$ описывается следующим образом.

1. Для каждой темы t_{l,k_l} начиная с $t_{1,1}$, $l \in \{1, \ldots, h\}$
   выбрать следующую тему $t_{l+1,k_{l+1}}$ согласно (1.40) и (1.39) во множестве дочерних тем темы t_{l,k_l}.
2. Выбрать пропорцию тем, полученных на шаге 1: $\boldsymbol{\alpha} \in \mathbb{R}^h, \quad \boldsymbol{\alpha} \sim \mathrm{Dir}$.
3. Для генерации слов



а. Структура тем в LDA. б. Структура тем в hLDA. в. Структура тем в PAM.

Рис. 1.5. Структура тем в различных алгоритмах.

выбрать t^* из множества тем, выбранных на шаге 1: $t^* \sim \text{Mult}(\boldsymbol{\alpha})$,
выбрать новое слово w из темы t^*: $w \sim p(w|\boldsymbol{\theta}_{t^*})$.

Все темы, выбранные на первом шаге, определяют конечное поддерево, состоящее из h уровней дерева возможных тем. Число тем на каждом уровне этого поддерева не является параметром, а определяется в процессе генерации темы для каждого нового документа.

Модели PAM и HPAM. Вместо древовидной структуры тем, алгоритмы PAM и HPAM используют направленный ациклический граф DAG [112, 8], в котором темы уровня l могут иметь общие дочерние подтемы на уровне $l+1$. Пример DAG приведен на рис. 1.5 в. Отличием HPAM от PAM является возможность генерировать слова не только в темах нижнего уровня h, но и в темах более высоких уровней. Для этого к теме $t_{l,k}$ помимо вектора параметров $\boldsymbol{\theta}_{l,k}$ мультиномиального распределения подтем следующего уровня, добавляется вектор параметров $\boldsymbol{\phi}_{l,k} \in \mathbb{R}^{|W|}$ мультиномиального распределения слов из W для этой темы. Процедура генерации нового документа d для HPAM имеет следующий вид.

1. Для каждой темы $t_{l,k}$ выбрать вектор параметров $\boldsymbol{\theta}_{l,k} \sim \text{Dir}$.
2. Пока не будет выбрано слово w, начиная с $t_{1,1}$
   выбрать номер узла следующего уровня: $k_{l+1} \sim \text{Mult}(\boldsymbol{\theta}_{l,k_l})$,
   если $k_{l+1} = 0$ или $l = h$, выбрать слово $w \sim \text{Mult}(\boldsymbol{\phi}_{l,k_l})$.

Параметры распределений настраиваются с помощью сэмплирования Гиббса [113].

Непараметрические модели NPBPAM, hvHDP и htHDP. Чтобы не задавать начальные значения структурных параметров, задающих число тем

на каждом уровне, мультиномиальные распределения подтем в темах на каждом уровне алгоритма PAM заменяются на процессы Дирихле [35]. Чтобы не задавать начальное значение числа уровней, в [23] используются два подхода: hvHDP, в котором внутренние узлы иерархии являются распределениями над множеством тем и слов, и htHDP, в котором только листовые элементы иерархии являются распределениями над множеством слов, а внутренние узлы – распределениями над множеством тем. Уровни строятся снизу вверх как при агломеративной жесткой кластеризации.

На первом шаге в hvHDP и htHDP с помощью HDP строится набор общих тем $T_h = t_{h,1}, \ldots, t_{h,K_h}$, каждой теме соответствует вектор параметров мультиномиального распределения $\boldsymbol{\phi}_{h,k} \in \mathbb{R}^{|W|}$ над множеством слов, а каждому документу d_n соответствует пропорция полученных тем $\boldsymbol{\theta}_n \in \mathbb{R}^{K_h}$.

В hvHDP на шаге l объектами являются векторы параметров тем $\boldsymbol{\phi}$, полученные на предыдущем шаге. Вместо матрицы документ-слово используется матрица тема-слово, имеющая размерность $|T_l| \times |W|$.

В htHDP на шаге l объектами являются векторы пропорций тем $\boldsymbol{\theta}_n$, полученных на предыдущем шаге. Вместо матрицы документ-слово используется матрица документ-тема размерности $|D| \times |T_l|$.

На каждом шаге число тем уменьшается, алгоритм останавливается, когда на очередной итерации останется только одна тема [23].

1.8. Описательно-вероятностные модели и смеси моделей

Описательно-вероятностные модели и смеси моделей комбинируют вероятностные предположения о процессе порождения документов коллекции с векторными представлениями документов и кластеров. Документ в данных подходах относится к каждой из тем с определенной вероятностью.

Смеси моделей, алгоритм vMF. В [7] документы описываются векторами $\mathbf{x} = [x_1, \ldots, x_{|W|}]^\intercal$, в которых $x_m = N(w_m, d)$. В качестве сходства документов $\mathbf{x}$ и $\mathbf{y}$ используется корреляция Пирсона:

$$\rho(\mathbf{x}, \mathbf{y}) = \frac{(\mathbf{x} - \overline{\mathbf{x}})^\intercal (\mathbf{y} - \overline{\mathbf{y}})}{\sqrt{(\mathbf{x} - \overline{\mathbf{x}})^\intercal (\mathbf{x} - \overline{\mathbf{x}})}\sqrt{(\mathbf{y} - \overline{\mathbf{y}})^\intercal (\mathbf{y} - \overline{\mathbf{y}})}}, \quad \text{где} \tag{1.41}$$

$$\overline{x} = \frac{1}{|W|} \sum_{m=1}^{|W|} x_m, \quad \overline{\mathbf{x}} = [\overline{x}, \overline{x}, \ldots, \overline{x}].$$

Пусть векторы $\mathbf{x}$ документов нормированы следующим образом:

$$\mathbf{x} \mapsto \frac{(\mathbf{x} - \overline{\mathbf{x}})}{\sqrt{(\mathbf{x} - \overline{\mathbf{x}})^\intercal (\mathbf{x} - \overline{\mathbf{x}})}}. \tag{1.42}$$

В этом случае корреляция Пирсона имеет вид $\rho(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\intercal \mathbf{y}$ и является частным случаем косинусной меры близости (1.26).

Пусть тема с номером k описывается распределением фон Мизеса-Фишера (vMF) [114, 115] с параметрами $\boldsymbol{\theta}_k$ и γ_k, а документ $\mathbf{x}$ описывается как смесь распределений тем в пропорции $\boldsymbol{\alpha}$:

$$p(\mathbf{x}|\Theta, \boldsymbol{\gamma}) = \sum_{k=1}^{K} \alpha_k z_{|W|}(\gamma_k) \exp\left(\gamma_k \boldsymbol{\theta}_k^\intercal \mathbf{x}\right), \quad \|\boldsymbol{\alpha}\| = 1, \quad \|\boldsymbol{\theta}_k\| = 1, \quad \boldsymbol{\alpha} \geq \mathbf{0}. \quad (1.43)$$

Чем больше параметр γ_k, тем больше концентрация документов из темы k вокруг ее направления $\boldsymbol{\theta}_k$. Нормализующий множитель $z_{|W|}(\gamma)$ задается как:

$$z_{|W|}(\gamma) = \frac{\gamma^{|W|/2}}{(2\pi)^{|W|/2} I_{d/2-1}(\gamma)}, \quad (1.44)$$

где $I_r(\cdot)$ – модифицированная функция Бесселя первого рода порядка r.

Описательно-вероятностные модели, DPM. В [6] документы и кластеры описываются векторами, но каждый документ может принадлежать многим кластерам одновременно.

Слова в документах делятся на информативные и неинформативные. Пусть словарь W содержит все слова коллекции D без повторений, подмножество $\hat{W} \subseteq W$ – словарь информативных слов, $t \in T$ – темы, а $\mathbf{x}$ – векторное представление документа d, в котором на позиции m стоит число слов w_m в d. Аналогично вероятностному подходу (1.29), вводится гипотеза условной независимости в следующей форме:

$$p(t, \mathbf{x}|w) = p(t|w)p(\mathbf{x}|w) \quad \text{или}$$

$$p(t|w, \mathbf{x}) = \frac{p(t, \mathbf{x}|w)}{p(\mathbf{x}|w)} = \frac{p(t|x)p(\mathbf{x}|w)}{p(\mathbf{x}|w)} = p(t|w),$$

где $p(t, \mathbf{x}|w)$ – совместное распределение тем и документов при условии, что встретилось слово w. Предполагается, что неинформативное слово $w \in W \setminus \hat{W}$ не влияет на тему документа: $p(t|w, \mathbf{x}) = p(t|\mathbf{x})$. Тогда вероятность документа $\mathbf{x}$ принадлежать теме t

$$p(t|\mathbf{x}) = \sum_{w \in W} p(t|w, \mathbf{x})p(w|\mathbf{x}) = \sum_{w \in \hat{W}} p(t|w, \mathbf{x})p(w|\mathbf{x}) +$$
$$+ \sum_{w \in W \setminus \hat{W}} p(t|w, \mathbf{x})p(w|\mathbf{x}) = \sum_{w \in \hat{W}} p(t|w)p(w|\mathbf{x}) + \sum_{w \in W \setminus \hat{W}} p(w|\mathbf{x})p(t|\mathbf{x}). \quad (1.45)$$

В [6] предполагается, что для всех документов вероятность встретить неинформативное слово одинакова:

$$\sum_{w \in W \setminus \hat{W}} p(w|\mathbf{x}) = r. \quad (1.46)$$

С учетом этого предположения вероятность $p(t|\mathbf{x})$ приобретает вид

$$p(t|\mathbf{x}) = \frac{1}{1-r} \sum_{w \in \hat{W}} p(t|w) p(w|\mathbf{x}). \tag{1.47}$$

Тема документа $\mathbf{x}$ определяется как

$$t^* = \arg\max_t p(t|\mathbf{x}). \tag{1.48}$$

Пусть

$$\mathrm{tf}'(w_m, \mathbf{x}) = \frac{x_m}{\|\mathbf{x}\|}, \quad \mathrm{idf}'(w_m) = \sqrt{\frac{|D|}{\sum_{d \in D} \frac{x_m}{\|\mathbf{x}\|}}}. \tag{1.49}$$

Значения $p(w|\mathbf{x}), p(t), p(x|t)$ оцениваются как

$$p(w_m|\mathbf{x}) = \frac{x_m}{\|\mathbf{x}\|}, \quad p(t) = \frac{N(t,D)}{|D|}, \quad p(w|t) = \frac{1}{N(t,D)} \sum_{\mathbf{x}' \in t} p(w_m|\mathbf{x}'), \tag{1.50}$$

где $N(t,D)$ – число документов с темой t в коллекции D. Найденные вероятности (1.50) подставляются в (1.47). После перехода к новому представлению документа d в виде вектора с компонентами $x_m = \mathrm{tf}'(w_m, d) \cdot \mathrm{idf}'(w_m)$, вероятность темы

$$p(t|\mathbf{x}) = \frac{1}{1-r} \frac{N(t,D)}{|D|} \mathbf{x}^{\mathsf{T}} \mathbf{t}, \text{ где } \mathbf{t} = \frac{1}{N(t,D)} \sum_{\mathbf{x}' \in t} \mathbf{x}' \text{ – центр темы } t. \tag{1.51}$$

1.9. Иерархическая классификация документов

В данном разделе рассматриваются алгоритмы текстовой иерархической классификации. Решением задачи иерархической классификации является отображение, ставящее в соответствие каждому документу $\mathbf{x}$ набор меток кластеров $\{k_1, \ldots, k_h\}$ всех уровней иерархии, наилучшим образом восстанавливающее экспертную классификацию размеченного подмножества документов согласно заданному критерию качества. В качестве экспертных иерархических структур кластеров рассматриваются деревья кластеров. В этом случае задача иерархической классификации может быть представлена в виде задачи плоской классификации документа $\mathbf{x}$ в множестве кластеров нижнего уровня h. Использование дополнительной информации об иерархичности структуры и экспертной классификации размеченных документов на остальных уровнях иерархии позволяет увеличить качество построенного решения.

Иерархический наивный байес. Документ d представляется как целочисленный вектор размерности словаря W. Предполагается что каждому классу $c_{h,k}$ соответствует распределение с параметрами $\boldsymbol{\theta}$, а новый документ генерируется по следующей схеме:

1) выбрать класс $c_{h,k}$ из $p(c|\boldsymbol{\theta})$,
2) сгенерировать документ согласно распределению $p(\mathbf{x}|c_{h,k},\boldsymbol{\theta})$.

Предполагается, что слово в документе зависит только от класса документа и не зависит от контекста и его позиции в документе. Вероятность документа, принадлежащего классу $c_{h,k}$ имеет вид

$$p(d|c_{h,k}) = p(|d|)\prod_{w\in d} p(w|c_{h,k}). \tag{1.52}$$

Согласно формуле байеса, вероятность документа принадлежать классу $c_{h,k}$

$$p(c_{h,k}|d) = \frac{p(c_{h,k})\prod_{w\in d} p(w|c_{h,k})}{\sum_{c\in C} p(c)\prod_{w'\in d} p(w'|c)}. \tag{1.53}$$

Параметрами модели (1.53) являются вероятности слов $p(w_m|c_{h,k}) = \theta_{mk}$ и априорные вероятности классов $p(c_{h,k}) = \theta_{0k}$. Эти параметры оцениваются с помощью размеченной коллекции документов D:

$$\hat{\theta}_{mk} = \frac{1+\sum_{d\in D} N(w_m,d)p(c_{h,k}|d)}{|W|+\sum_{w\in W}\sum_{d'\in D} N(w,d')p(c_{h,k}|d')}, \qquad \hat{\theta}_{0k} = \frac{1}{|D|}\sum_{d\in D} p(c_{k,h}|d), \tag{1.54}$$

$$p(c_{h,k}|d) = [c(d) = c_{h,k}].$$

При оценивании вероятностей слов $\hat{\theta}_{mk}$ для кластера $c_{h,k}$ в том случае, когда число документов $|c_{h,k}| \ll |W|$, многие слова ни разу не встретятся, а оценки вероятностей встретившихся слов будут завышены, несмотря на сглаживание Лапласа в (1.54). Чтобы избежать этого, при иерархической классификации в [9] предлагается алгоритм hNB, в котором параметры кластеров более низких уровней усредняются с параметрами их родительских кластеров. Для параметров кластера $c_{h,k}$ нижнего уровня используется следующая оценка:

$$\hat{\theta}_{mk} = \lambda_{k,1}\hat{\theta}_{mk,1} + \lambda_{k,2}\hat{\theta}_{mk,2} + \ldots + \lambda_{k,h}\hat{\theta}_{mk,h}, \qquad \sum_{l=1}^{h}\lambda_{k,l} = 1, \tag{1.55}$$

где $\{\hat{\theta}_{mk,l}\}$ – оценка параметров родительского кластера $B^{h-l}(c_{h,k})$ на уровне l кластера $c_{h,k}$, а оператор B возвращает родительский кластер заданного кластера. Применив $h-l$ раз оператор B к кластеру нижнего уровня h получаем его родительский кластер на уровне l. Веса $\lambda_{k,l}$ в (1.55) настраиваются максимизацией правдоподобия. Усреднение параметров (1.55) значительно улучшает качество алгоритма иерархической классификации [9].

Иерархический мультиклассовый svm. Для адаптации svm к многоклассовой классификации в [13] на каждом кластере $c_{l,k}$ уровня l обучается

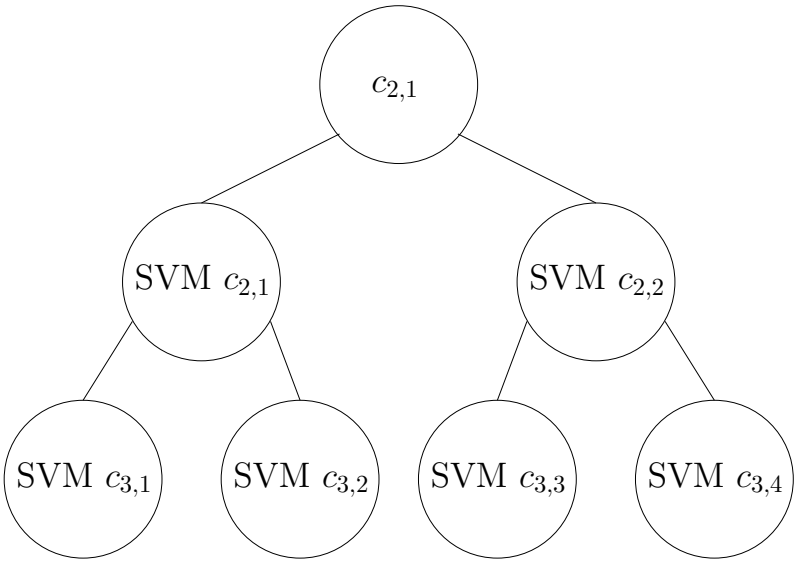


Рис. 1.6. Иерархия алгоритмов SVM.

двухклассовый svm. При этом документы из данного кластера рассматриваются как объекты класса 1, а все остальные документы – как объекты класса 0.

Вероятность документа $\mathbf{x}$ принадлежать кластеру $c_{l,k}$ оценивается с помощью метода Платта [95]

$$p(c_{l,k}|\mathbf{x}) = \frac{1}{1+\exp\big(a_{l,k}f(\mathbf{x})+b_{l,k}\big)}, \tag{1.56}$$

где $f(\mathbf{x})$ – результат, полученный с помощью SVM на объекте $\mathbf{x}$, а $b_{l,k}$ и $a_{l,k}$ – параметры.

Для ранжирования кластеров нижнего уровня иерархии по убыванию релевантности новому документу используется следующий подход. Пусть $C_h(c_{l,k})$ – множество кластеров уровня h, являющихся дочерними кластерами для кластера $c_{l,k}$, а $k(c)$ – индекс кластера c в ранжированном по релевантности списке кластеров нижнего уровня. На шаге l для всех кластеров $c_{l,k}$ уровня l кластеры $C_h(c_{l,k})$ в ранжированном списке переставляются таким образом, чтобы для любых двух кластеров из $C_h(c_{l,k})$ выполнялось соотношение:

$$k(c_1) < k(c_2) \Rightarrow p\big(B^{h-l-1}(c_1)|\mathbf{x}\big) \geq p\big(B^{h-l-1}(c_2)|\mathbf{x}\big).$$

Пусть для кластеров с рис. 1.6 для некоторого документа выполняются соотношения

$$p(c_{2,1}) > p(c_{2,2}), \quad p(c_{3,1}) > p(c_{3,2}), \quad p(c_{3,4}) > p(c_{3,3}).$$

Тогда ранжированный список кластеров нижнего уровня $h = 3$ для данной иерархии и документа имеет вид:

$$\left(\overbrace{c_{3,1}, c_{3,2}}^{c_{2,1}}, \overbrace{c_{3,4}, c_{3,3}}^{c_{2,2}}\right).$$

ARTM для коллекции с экспертной моделью. Пусть $t(d)$ – экспертная тема документа d, а матрица $\mathbf{Z} \in \mathbb{R}^{|T|\times|D|}$ определяется экспертной классификацией:

$$z_{td} = [t(d) = t].$$

В [10] для учета экспертной кластеризации в алгоритме ARTM, описанном в разделе 1.6., используется регуляризатор

$$\Omega(\boldsymbol{\Theta}, \mathbf{Z}) = -\|\boldsymbol{\Theta} - \mathbf{Z}\|_1. \tag{1.57}$$

Задача поиска разложения (1.30) с регуляризатором (1.57) имеет вид:

$$\boldsymbol{\Phi}^*, \boldsymbol{\Theta}^* = \arg\max_{\boldsymbol{\Phi},\boldsymbol{\Theta}} \sum_{d\in D}\sum_{w\in W} n_{dw} \ln \sum_{t\in T} \phi_{wt}\theta_{td} + \tau \left(\sum_{d\in D}\sum_{t\in T} \theta_{td}(2z_{td} - 1) \right), \tag{1.58}$$

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

Так как регуляризатор $\Omega(\boldsymbol{\Theta}, \mathbf{Z})$ непрерывно дифференцируем, согласно теореме 1 в локальном максимуме ϕ_{wt} и θ_{td} задаются как (1.34) и (1.35). Так как $\Omega(\boldsymbol{\Theta}, \mathbf{Z})$ зависит только от θ_{td}, используется только (1.35), и формула для M шага EM-алгоритма имеет вид:

$$\theta_{td} = \frac{\eta_{td}}{\sum\limits_{t\in T} \eta_{td}}, \quad \eta_{td} = \left(\sum_{w\in W} n_{dw} \frac{\phi_{wt}\theta_{td}}{\sum\limits_{t'\in T} \phi_{wt'}\theta_{t'd}} + \tau\theta_{td}(2z_{td} - 1) \right)_+. \tag{1.59}$$

ARTM для коллекции с иерархической экспертной моделью, SuhiPLSA Пусть $\mathbf{Z}_l$ – матрица экспертной классификации для уровня l. Для учета иерархической экспертной модели регуляризатор (1.57) заменяется на

$$\Omega_h(\boldsymbol{\Theta}, \mathbf{Z}_2, \ldots, \mathbf{Z}_h) = \sum_{l=2}^{h} \sum_{d\in D} \sum_{t\in\{t_{l,i}\}} |z_{td,l} - \theta_{td,l}|, \quad \text{где} \tag{1.60}$$

$$\theta_{td,l} = \frac{1}{|T_l(d)|} \sum_{t'\in T_l(d)} \theta_{t'd,h}, \quad T_l(d) = \{t \,:\, B^{h-l}(t) = B^{h-l}\big(t(d)\big)\}.$$

$T_l(d)$ – множество тем уровня h, у которых родительская тема на уровне l совпадает с родительской темой уровня l экспертной темы $t(d)$ документа d уровня h. В выражении для M шага (1.59) с регуляризатором (1.60) переменная η_{td} принимает вид

$$\eta_{td} = \left(\sum_{w\in W} n_{dw} \frac{\phi_{wt}\theta_{td}}{\sum\limits_{t\in T} \phi_{wt}\theta_{td}} + \sum_{l=2}^{h} \tau_l \theta_{t_l d,l}(2z_{t_l d,l} - 1) \right)_+, \quad t_l = B^{h-l}(t).$$

После настройки модели столбцы матрицы слово-тема $\mathbf{\Phi}^*$ определяют векторы $\boldsymbol{\phi}_{h,k}$ тем $t_{h,k}$, соответствующие экспертным кластерам $c_{h,k}$. Для построения ранжированного списка кластеров нижнего уровня в порядке убывания релевантности новому документу $\mathbf{x}$ используется косинусная мера сходства (1.26) с $\mathbf{\Lambda} = \mathbf{I}$:

$$\left(c_{h,k_1}, c_{h,k_2}, \ldots, c_{h,k_{K_h}}\right): \quad s(\boldsymbol{\phi}_{h,k_1}, \mathbf{x}) \geq s(\boldsymbol{\phi}_{h,k_2}, \mathbf{x}) \geq \cdots \geq s(\boldsymbol{\phi}_{h,k_{K_h}}, \mathbf{x}).$$

Глава 2
Отбор признаков и метрическая кластеризация

В данной главе предлагается способ выбора и оптимизации взвешенной метрики ρ для плоской и иерархической кластеризации и документов. Пусть имеется размеченная коллекция D с экспертной тематической моделью M в виде дерева, в котором каждому документу d соответствует единственный кластер нижнего уровня $c_{h,k}$.

2.1. Выбор взвешенной метрики

Каждый документ представляется в виде вектора с помощью его частотных характеристик, см. раздел 1.4. На позиции x_m векторного представления документа $\mathbf{x}$ ставится

1) булево значение $[N(w_m, d) > 0]$,
2) значение произведения $\mathrm{tf}(w_m, d) \cdot \mathrm{idf}(w_m, D)$,
3) число слов w_m в документе $N(w_m, d)$.

Для сравнения документов вводится взвешенная функция расстояния Минковского с фиксированным параметром $p \geq 1$ и вектором важности слов $\boldsymbol{\lambda}$

$$\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}) = \sqrt[p]{\sum_{m=1}^{|W|} \lambda_m |x_m - y_m|^p}, \quad \text{где} \quad \boldsymbol{\lambda} \geq \mathbf{0}, \quad \|\boldsymbol{\lambda}\|_1 = 1. \tag{2.1}$$

Утверждение 6. Функция расстояния $\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y})$ (2.1) является метрикой.

Доказательство. Все свойства метрики выполняются.

1. $\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{y}$, так как все разности вида $x_m - y_m$ равны нулю.
2. $\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}) = \rho(\boldsymbol{\lambda}, \mathbf{y}, \mathbf{x})$, так как $|x_m - y_m| = |y_m - x_m|$.
3. Функция расстояния с $\boldsymbol{\lambda}_0 = [1, \ldots, 1]$ является расстоянием Минковского, которое является метрикой при $p \geq 1$, см. [116]. Для пары векторов $\mathbf{x}$ и $\mathbf{y}$ взвешенная функция расстояния переписывается в виде

$$\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}) = \sqrt[p]{\sum_{m=1}^{|W|} \lambda_m |x_m - y_m|^p} = \sqrt[p]{\sum_{m=1}^{|W|} |(x_m - y_m) \cdot \sqrt[p]{\lambda_m}|^p} =$$
$$= \rho(\boldsymbol{\lambda}_0,\ \mathbf{x} \cdot \sqrt[p]{\boldsymbol{\lambda}},\ \mathbf{y} \cdot \sqrt[p]{\boldsymbol{\lambda}}), \quad \mathbf{x} \cdot \boldsymbol{\lambda} = [x_1\lambda_1, \ldots, x_{|W|}\lambda_{|W|}], \tag{2.2}$$

где $\rho(\boldsymbol{\lambda}_0, \cdot, \cdot)$ – метрика Минковского. Неравенство треугольника для взвешенной функции расстояния записывается как

$$\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}) = \rho(\boldsymbol{\lambda}_0,\ \mathbf{x} \cdot \sqrt[p]{\boldsymbol{\lambda}},\ \mathbf{y} \cdot \sqrt[p]{\boldsymbol{\lambda}}) \leq \rho(\boldsymbol{\lambda}_0,\ \mathbf{x} \cdot \sqrt[p]{\boldsymbol{\lambda}},\ \mathbf{x}' \cdot \sqrt[p]{\boldsymbol{\lambda}}) +$$
$$+ \rho(\boldsymbol{\lambda}_0,\ \mathbf{x}' \cdot \sqrt[p]{\boldsymbol{\lambda}},\ \mathbf{y} \cdot \sqrt[p]{\boldsymbol{\lambda}}) = \rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{x}') + \rho(\boldsymbol{\lambda}, \mathbf{x}', \mathbf{y}). \tag{2.3}$$

□

Частными случаями взвешенной метрики (2.1) являются взвешенное расстояние городских кварталов при $p = 1$ и взвешенное евклидово расстояние при $p = 2$.

Функция качества метрики. Пусть $\mathcal{I}$ – множество индексов документов коллекции D, а $c(\mathbf{x})$ – экспертный кластер документа $\mathbf{x}$. Для произвольного документа $\mathbf{x}$ индексы всех остальных документов разбиваются на подмножество $\mathcal{P}(\mathbf{x})$ индексов документов из того же экспертного кластера $c(\mathbf{x})$ и подмножество индексов $\mathcal{N}(\mathbf{x})$ документов из остальных кластеров:

$$\mathcal{P}(\mathbf{x}) = \{n \in \mathcal{I} | c(\mathbf{x}) = c(\mathbf{x}_n)\},$$

$$\mathcal{N}(\mathbf{x}) = \{n \in \mathcal{I} | c(\mathbf{x}) \neq c(\mathbf{x}_n)\}.$$

Расстояние от документа $\mathbf{x}$ до k ближайших соседей своего класса $r_k(\mathbf{x})$ и до k ближайших соседей чужих классов $\overline{r}_k(\mathbf{x})$ задаются как

$$r_k(\mathbf{x}) = \min_{\mathcal{B} \subset \mathcal{P}(\mathbf{x}) : |\mathcal{B}| = k} \sum_{\mathbf{y} \in D_{\mathcal{B}}} \rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}),$$

$$\overline{r}_k(\mathbf{x}) = \min_{\mathcal{B} \subset \mathcal{N}(\mathbf{x}) : |\mathcal{B}| = k} \sum_{\mathbf{y} \in D_{\mathcal{B}}} \rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y}),$$

где $\mathcal{B}$ состоит из неповторяющихся элементов. Вспомогательная функция близости $s_\rho(\mathbf{x})$ объекта $\mathbf{x}$ к объектам своего класса задается как

$$s_\rho(\mathbf{x}) = \frac{\overline{r}_k(\mathbf{x}) - r_k(\mathbf{x})}{\overline{r}_k(\mathbf{x}) + r_k(\mathbf{x})}. \tag{2.4}$$

Функция $s_\rho(\mathbf{x})$ обладает следующим свойством:

$$s_\rho(\mathbf{x}) \approx \begin{cases} -1, & \text{объект } \mathbf{x} \text{ близок к объектам чужого класса,} \\ 0, & \text{объект } \mathbf{x} \text{ пограничный,} \\ +1, & \text{объект } \mathbf{x} \text{ близок к объектам своего класса.} \end{cases}$$

В предположении гипотезы компактности наилучшей метрикой для документа $\mathbf{x}$ будет та, у которой $s_\rho(\mathbf{x}) \approx 1$. Функция качества метрики ρ для всех документов коллекции задается как

$$V(\rho, \boldsymbol{\lambda}, D) = \frac{1}{|D|} \sum_{\mathbf{x} \in D} s_\rho(\mathbf{x}), \tag{2.5}$$

где ρ и $s_\rho(\mathbf{x})$ определены в (2.1) и (2.4). Задача выбора оптимальной метрики сводится к нахождению ее весов с помощью максимизации функции качества метрики $V(\rho, \boldsymbol{\lambda}, D)$:

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} V(\rho, \boldsymbol{\lambda}, D).$$

2.2. Алгоритм оптимизации весов метрики

Множество взвешенных метрик (2.1) задается набором весовых коэффициентов $\boldsymbol{\lambda}$. Их оптимизация осуществляется алгоритмом последовательного добавлении признаков [38]. Слово с индексом m называется активным признаком, если соответствующий ему вес λ_m больше нуля. Пусть $\mathcal{A}$ – множество активных признаков. Выборка D разбивается случайным образом на обучающее $D_{\mathcal{V}}$ и контрольное $D_{\mathcal{T}}$. Изначально $\boldsymbol{\lambda} = \mathbf{0}$ и $\mathcal{A} = \varnothing$. На каждом шаге алгоритма в $\mathcal{A}$ добавляется наилучший признак следующим образом.

1. Для каждого признака $m \notin \mathcal{A}$ найти набор весов $\hat{\boldsymbol{\lambda}}_m$ для множества признаков $\mathcal{A}_m = \{\mathcal{A} \cup m\}$, доставляющий максимум функции качества
$$\hat{\boldsymbol{\lambda}}_m = \arg\max_{\boldsymbol{\lambda}_m} V(\rho, \boldsymbol{\lambda}_m, D_{\mathcal{V}}), \quad \|\boldsymbol{\lambda}_m\|_1 = 1, \quad \lambda_{m,i} = 0,\ i \notin \mathcal{A}_m. \tag{2.6}$$
2. Найти признак m^* которому соответствует максимальное качество
$$m^* = \arg\max_{m \notin \mathcal{A}} V(\rho, \hat{\boldsymbol{\lambda}}_m, D_{\mathcal{V}})$$
и добавить его во множество $\mathcal{A}$, обновив вектор весов $\boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}}_{m^*}$.

Алгоритм повторяется до тех пор, пока значение функции качества $V(\rho, \boldsymbol{\lambda}, D_{\mathcal{T}})$ на контрольной выборке $D_{\mathcal{T}}$ увеличивается.

Утверждение 7. При условии $|D| \sim |D_{\mathcal{V}}| \sim |D_{\mathcal{T}}|$ сложность описанного выше алгоритма $O(|D|^2 k^2 |W|^2 t)$, где t – число итераций алгоритма оптимизации (2.6).

Доказательство. Для вычисления качества метрики (2.5) для каждого документа необходимо найти k ближайших соседей и вычислить (2.4). Это делается за $O(|D|^2 \log_2^2 |D|)$ с помощью сортировки. При $k < \log_2 |D|$ это можно сделать за $O(|D|^2 k^2)$.

На каждом шаге выбирается признак, дающий максимальный прирост качества метрики. Для поиска данного признака необходимо перебрать все возможные и найти для каждого оптимальный вес, решив задачу (2.6). На каждой итерации алгоритма оптимизации, необходимо вычислить качество метрики, таким образом для добавления очередного признака необходимо $O(|D|^2 k^2 |W| t)$ операций. Локально оптимальный набор признаков содержит $O(|W|)$ элементов, таким образом сложность всего алгоритма $O(|D|^2 k^2 |W|^2 t)$. □

Чтобы снизить сложность, вместо функции качества V используется среднее внутрикластерное расстояние
$$F_0 = \frac{\sum\limits_{\mathbf{x},\mathbf{y} \in D} [c(\mathbf{x}) = c(\mathbf{y})] \rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y})}{\sum\limits_{\mathbf{x},\mathbf{y} \in D} [c(\mathbf{x}) = c(\mathbf{y})]} \to \min_{\boldsymbol{\lambda}}, \quad \|\boldsymbol{\lambda}\|_1 = 1, \boldsymbol{\lambda} \geq \mathbf{0}, \tag{2.7}$$

где $[\cdot = \cdot]$ – индикаторная функция.

Утверждение 8. Решением оптимизационной задачи (2.7) для взвешенной метрики городских кварталов (2.1) с $p = 1$ является единичный вектор $\mathbf{e}(\cdot)$.

Доказательство. F_0 является линейной функцией относительно весов $\boldsymbol{\lambda}$. Ее требуется минимизировать на множестве $\|\boldsymbol{\lambda}\|_1 = 1, \boldsymbol{\lambda} \geq \mathbf{0}$, которое является выпуклым. У этой задачи существует решение, причем минимум реализуется в вершине $|W|$-мерного симплекса. В каждой вершине этого симплекса все координаты равны нулю, кроме одной, которая равна единице. □

Таким образом при оптимизации (2.7) отбирается единственный признак. Поэтому необходимо учитывать не только внутрикластерное расстояние, но и межкластерное. Оно определяется как

$$F_1 = \frac{\sum\limits_{\mathbf{x},\mathbf{y} \in D} [c(\mathbf{x}) \neq c(\mathbf{y})] \rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y})}{\sum\limits_{\mathbf{x},\mathbf{y} \in D} [c(\mathbf{x}) \neq c(\mathbf{y})]} \to \max_{\boldsymbol{\lambda}}, \quad \|\boldsymbol{\lambda}\|_1 = 1, \boldsymbol{\lambda} \geq \mathbf{0}, \tag{2.8}$$

и вместо функции качества (2.5) используется:

$$F = \frac{F_0}{F_1} \to \min_{\boldsymbol{\lambda}}. \tag{2.9}$$

2.3. Сравнение экспертной и алгоритмической модели

Пусть $\hat{c}(\mathbf{x})$ – номер кластера документа $\mathbf{x}$ на уровне h в построенной алгоритмом тематической модели $\hat{M}$, а $c(\mathbf{x})$ – номер его кластера в экспертной тематической модели M. Пусть $B(c)$ – оператор, возвращающий родительский кластер заданного кластера c на следующем уровней. Родительским кластером документа $\mathbf{x}$ на уровне l является кластер $B^{h-l}\big(c(\mathbf{x})\big)$. Обозначим $\boldsymbol{\mu}(c)$ координаты центра кластера c.

Определение 13. Ошибка классификации документа $\mathbf{x}$ в алгоритмической модели $\hat{M}$ относительно экспертной модели M задается как суммарное расстояние между центрами экспертных и алгоритмических кластеров документа $\mathbf{x}$ на всех уровнях иерархии

$$\upsilon(\mathbf{x}, M, \hat{M}) = \sum_{l=1}^{h} \rho\left(\boldsymbol{\mu}\left(B^{h-l}\big(c(\mathbf{x})\big)\right), \boldsymbol{\mu}\left(B^{h-l}\big(\hat{c}(\mathbf{x})\big)\right)\right). \tag{2.10}$$

Определение 14. Расстояние $\Upsilon(M, \hat{M})$ между экспертной моделью M и алгоритмической моделью $\hat{M}$ определяется как сумма ошибки классификации неразмеченных документов

$$\Upsilon(M, \hat{M}) = \sum_{\mathbf{x} \in D} \upsilon(\mathbf{x}, M, \hat{M}). \tag{2.11}$$

2.4. Анализ метрических свойств описаний документов

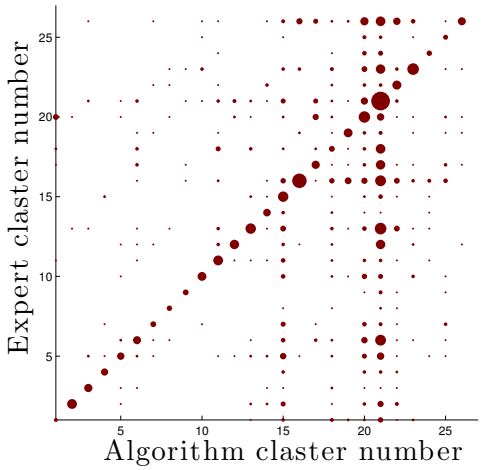
Для анализа способов векторного представления документов и свойств алгоритма построения метрики с оптимальным набором весов $\boldsymbol{\lambda}$ был проведен эксперимент плоской кластеризации $|D| = 1342$ тезисов научной конференции European Conference on Operational Research на $K_2 = 26$ кластеров. Количество кластеров выбрано в согласии с числом научных областей (англ. Areas) на конференции. Рассматривались три способа векторного представления документа, описанные в разделе 2.1.

Кластеризация документов. Для кластеризации использовался алгоритм k-means, описанный в разделе 1.5., с функцией расстояния (2.1). Начальные положение центров кластеров задавались согласно экспертной кластеризации размеченных документов. Для оценки качества полученной кластеризации использовалась функция (2.11). На рис. 2.1 а.-г. показаны расхождения между построенной и экспертной моделью. Эти графики строились следующим образом.

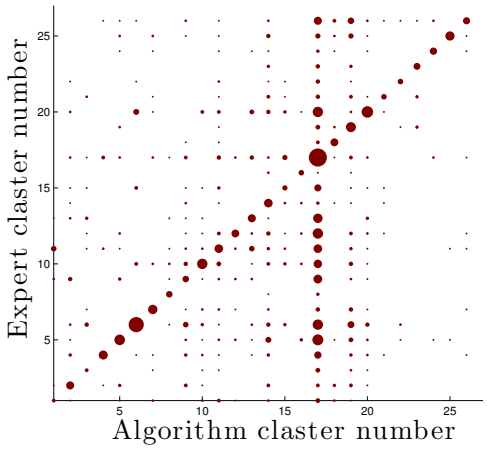
1. При помощи метода главных компонент центры полученных кластеров нумеровались таким образом, чтобы наиболее близкие кластеры имели близкие порядковые номера.
2. Все документы откладывались на плоскости: координатой документа по оси абсцисс являлся номер кластера в алгоритмической модели $\hat{M}$, а по оси ординат – номер кластера в экспертной модели M. В каждой точке рисовался круг с радиусом пропорциональным числу документов, попавших в эту точку. Наибольшему кругу соответствовало 80 документов.

Чем дальше документ находился от прямой $y = x$ на данной плоскости, тем сильнее отличались тематики кластеров, к которым он был отнесен алгоритмом и экспертом. Как видно из рис. 2.1 для булевых признаков характерен меньший разброс документов относительно диагонали. Это подтверждается значениями расстояния (2.11) между построенной моделью и экспертной, указанными в таблице 2.1. При использовании дополнительной настройки весов $\boldsymbol{\lambda}$ алгоритм кластеризации показывал более высокие результаты. Алгоритм оптимизации $\boldsymbol{\lambda}$ позволил удалить некоторые неинформативные слова, например, “matrix”, “compute”, “activity”, встречающиеся почти во всех кластерах, и не несущие информации о кластеризации, так как большая часть тезисов конференции EURO посвящена методам оптимизации.

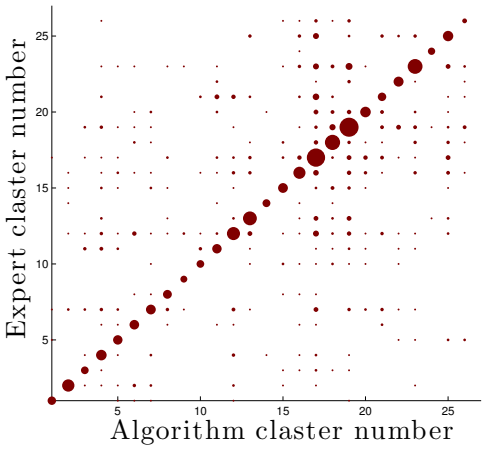
Для визуализации кластеризации документов и сравнения алгоритмической кластеризации с экспертной строились гистограммы, изображенные на рис. 2.2. Количество столбцов совпадало с количеством кластеров, каждому документу присваивался цвет экспертного кластера. Высота части столбца одного цвета показывала число документов, экспертно отнесенных к кластеру с данным цветом, которое алгоритм отнес к кластеру, соответствующему номеру столбца. Рис. 2.2 а. построен по экспертной кластеризации, поэтому каждый столбец



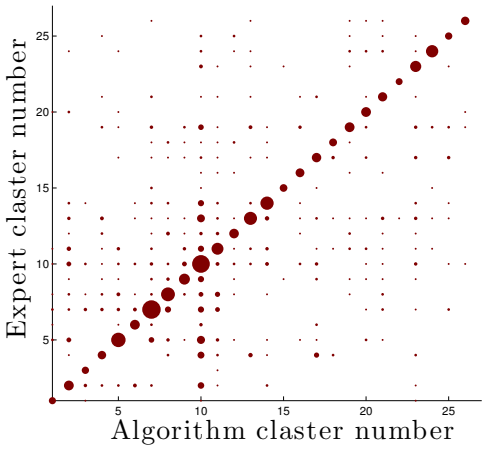
а. Признаки – $\mathrm{tf} \cdot \mathrm{idf}$.



б. Признаки – число появлений слов.



в. Булевы признаки с оптимальной метрикой.



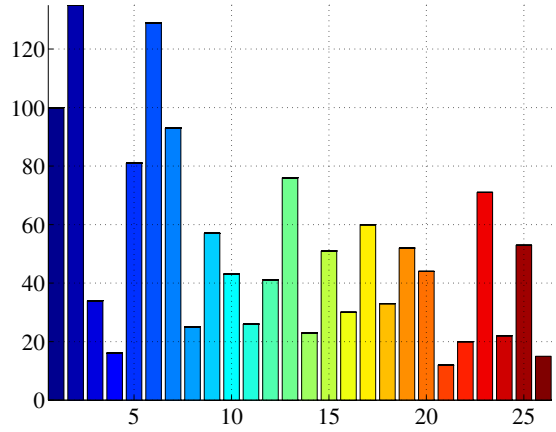
г. Булевы признаки.

Рис. 2.1. Сравнение экспертной и алгоритмической кластеризации.

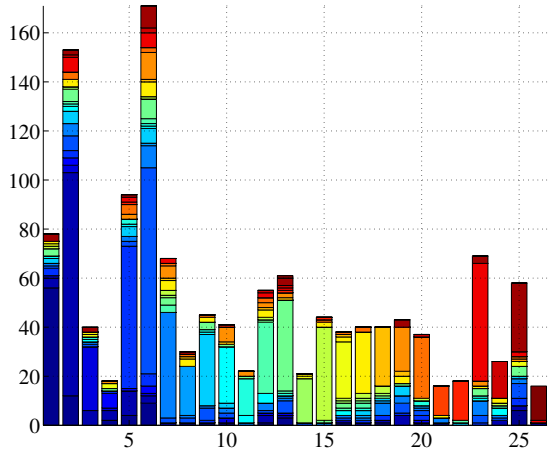
Таблица 2.1. Значение функции ошибки для разных способов построения набора признаков.

Способ построения признака	Появление слова в документе	Критерий $\mathrm{tf} \cdot \mathrm{idf}$	Число появлений слова
Υ (2.11), единичные веса $\boldsymbol{\lambda} = \mathbf{1}$	398	710	771
Υ, оптимальные веса $\boldsymbol{\lambda}$	**364**	630	650

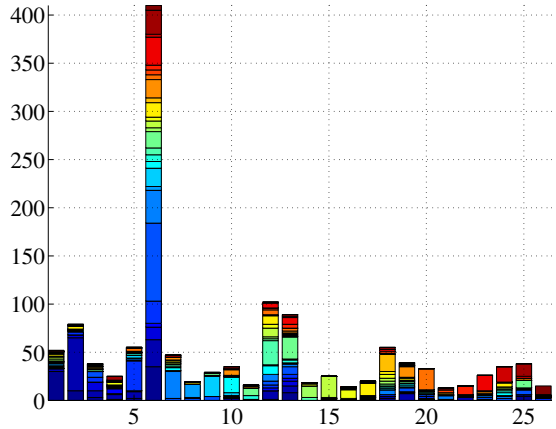
имеет документы только одного цвета, являющегося цветом кластера с номером данного столбца. На рис. 2.2 б.-г. показаны алгоритмические распределения документов по кластерам для различных способов векторного представления документов.



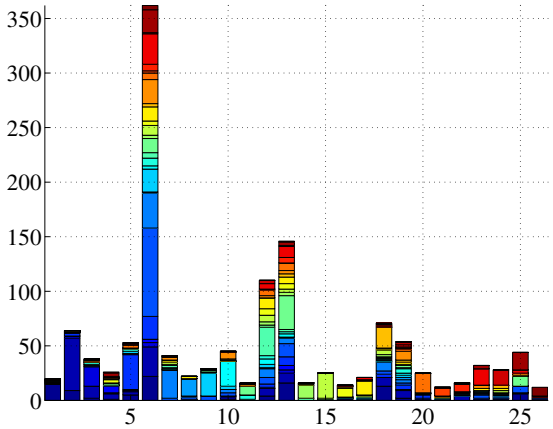
а. Экспертная кластеризация.



б. Булевы признаки с оптимальным набором весов $\boldsymbol{\lambda}$.



в. Признаки – число повторов слов.



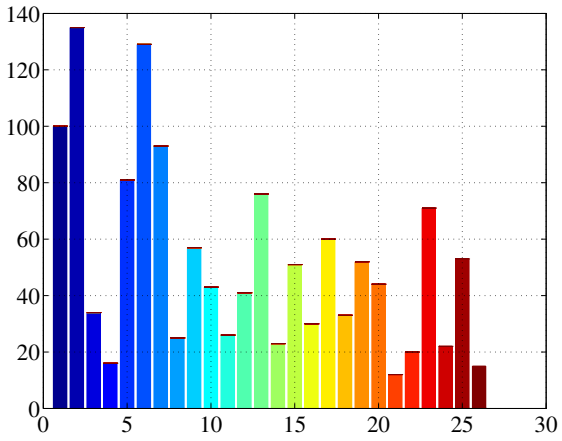
г. Признаки – $\mathrm{tf} \cdot \mathrm{idf}$.

Рис. 2.2. Перераспределение документов по кластерам для экспертной и алгоритмической кластеризации.

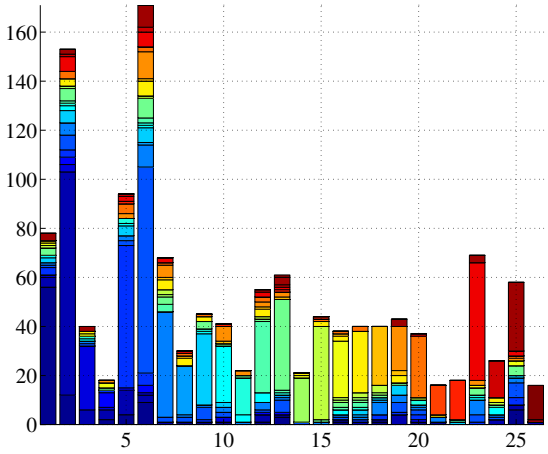
2.5. Анализ алгоритмов иерархической кластеризации

Для анализа агломеративного и дивизимного подходов построения иерархической тематической модели, описанных в разделе 1.5., рассматривался еще один уровень экспертной тематической модели конференции EURO. Каждый

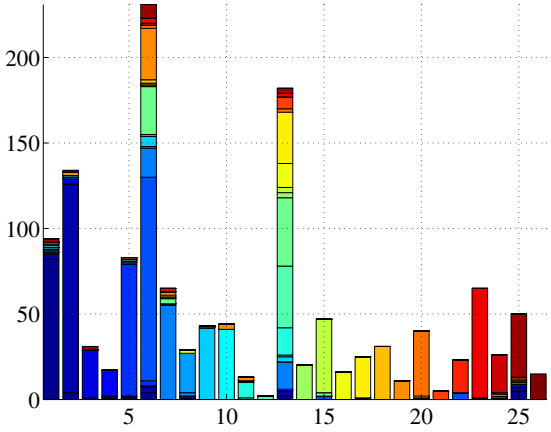
кластер уровня Area (Область) разбивался на некоторое количество кластеров уровня Stream (Направление). Структура экспертной тематической модели изображена на рис. 5.1.



а. Экспертная кластеризация.



б. Дивизимный алгоритм.



в. Агломеративный алгоритм.

Рис. 2.3. Распределение документов по кластерам для экспертной и алгоритмической иерархической кластеризации.

Для оценки качества работы алгоритмов вычислялось количество несоответствий между алгоритмической $\hat{M}$ и экспертной M моделями: 1) количество документов, которые относятся экспертной и алгоритмической моделью к различным кластерам уровня Area, 2) к различным кластерам уровня Stream, 3) расстояние между моделями Υ (2.11). Полученные результаты для дивизимного и агломеративного алгоритмов приведены в таблице 2.2.

На рис. 2.3 показано перераспределение документов по кластерам уровня Area. Рис. 2.3 а. построен по экспертной кластеризации, результат работы дивизимного алгоритма изображен на рис. 2.3 б., а результат работы агломератив-

Таблица 2.2. Количество различий и расстояние $\Upsilon(M, \hat{M})$ для разных способов построения иерархической тематической модели.

Вид алгоритма	Число различий на уровне Area	Число различий на уровне Stream	Значение функционала Υ (2.11)
Дивизимный	486	555	1700
Агломеративный	**342**	**208**	**500**

ного алгоритма на рис. 2.3 в. Рис. 2.3 в. показывает, что в результате использования агломеративного алгоритма, при кластеризации в столбцы попадают не отдельные документы из других столбцов, а целые кластеры уровня Stream. Таким образом несоответствия между экспертной и алгоритмической моделями носят более систематический характер. С другой стороны, при использовании агломеративного алгоритма появились два кластера (6-ой и 13-ый), к которым вместо части документов из других кластеров уровня Area были отнесены сразу целые кластеры уровня Stream, в результате чего количество документов в этих кластерах уровня Area стало отличаться от экспертного примерно в два раза.

Глава 3
Иерархическая классификация неразмеченных документов

В данной главе предлагается способ ранжирования кластеров нижнего уровня иерархической тематической модели по убыванию их релевантности неразмеченному документу.

3.1. Иерархическая функция сходства

Взвешенная косинусная функция сходства $s(\mathbf{x},\ \mathbf{y})$ двух документов $\mathbf{x}$ и $\mathbf{y}$ задается как

$$s(\mathbf{x},\ \mathbf{y}) = \frac{\mathbf{x}^\intercal \mathbf{\Lambda} \mathbf{y}}{\sqrt{\mathbf{x}^\intercal \mathbf{\Lambda} \mathbf{x}}\sqrt{\mathbf{y}^\intercal \mathbf{\Lambda} \mathbf{y}}}. \tag{3.1}$$

В случае равенства знаменателя нулю значение функции сходства считается равной нулю. Симметричная неотрицательно определенная матрица $\mathbf{\Lambda} = \mathbf{\Lambda}^\intercal$ введена для учета важности признаков. В данной главе предполагается, что эта матрица имеет диагональный вид $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1,\ \ldots,\ \lambda_{|W|})$, $\lambda_m \geq 0$, так как оптимизация всех элементов матрицы $\mathbf{\Lambda}$ размера $|W| \times |W|$ приводит к неадекватному увеличению сложности модели. Для удобства дальнейшего изложения все векторные представления документов нормируются следующим образом:

$$\mathbf{x} \mapsto \frac{\mathbf{x}}{\sqrt{\mathbf{x}^\intercal \mathbf{\Lambda} \mathbf{x}}}. \tag{3.2}$$

С учетом нормировки функция сходства (3.1) приобретает вид

$$s(\mathbf{x},\ \mathbf{y}) = \mathbf{x}^\intercal \mathbf{\Lambda} \mathbf{y}.$$

Взвешенная косинусная функция сходства (3.1) позволяет быть близкими документам $\mathbf{x}$ и $\mathbf{y}$, содержащими различное число слов, но одинаковый словарный состав с учетом нормировки. Так как все компоненты векторов $\mathbf{x}$, $\mathbf{y}$ неотрицательны, то $s(\mathbf{x}, \mathbf{y}) \in [0, 1]$, причем $s(\mathbf{x}, \mathbf{y}) = 1$ достигается для документов, словарный состав которых одинаков. Пусть $A(c_{l,k_1}, c_{l,k_2})$ – множество всех пар документов $\{(\mathbf{x}, \mathbf{y})\}$ из кластеров c_{l,k_1} и c_{l,k_2} таких, что $\mathbf{x} \in c_{l,k_1}$, $\mathbf{y} \in c_{l,k_2}$.

Определение 15. Сходство s_c двух кластеров c_{l,k_1} и c_{l,k_2} уровня l задается как среднее сходство между всеми парами документов $(\mathbf{x}, \mathbf{y}) \in A(c_{l,k_1}, c_{l,k_2})$

$$s_\mathsf{c}(c_{l,k_1},\ c_{l,k_2}) = \frac{1}{|A(c_{l,k_1}, c_{l,k_2})|} \sum_{(\mathbf{x},\,\mathbf{y}) \in A(c_{l,k_1}, c_{l,k_2})} s(\mathbf{x},\ \mathbf{y}), \tag{3.3}$$

При вычислении внутрикластерного сходства для c_{l,k_1}, в качестве множества пар рассматривается $A(c_{l,k_1}, c_{l,k_1})$. Обозначим $\boldsymbol{\mu}(c_{l,k_1})$ средний вектор кластера c_{l,k_1}:

$$\boldsymbol{\mu}(c_{l,k_1}) = \frac{1}{|c_{l,k_1}|} \sum_{\mathbf{x} \in c_{l,k_1}} \mathbf{x}. \tag{3.4}$$

Утверждение 9. Сходство (3.3) между парой кластеров c_{l,k_1} и c_{l,k_2} определяется средними векторами кластеров, их размерами и матрицей $\mathbf{\Lambda}$.

Доказательство. В соответствии с (3.1) и (3.3)

$$s_{\mathsf{c}}(c_{l,k_1},\, c_{l,k_2}) = \frac{1}{|c_{l,k_1}||c_{l,k_2}|} \sum_{\mathbf{x}\in c_{l,k_1}} \sum_{\mathbf{y}\in c_{l,k_2}} \mathbf{x}^{\mathsf{T}}\mathbf{\Lambda}\mathbf{y} =$$

$$= \left(\frac{1}{|c_{l,k_1}|} \sum_{\mathbf{x}\in c_{l,k_2}} \mathbf{x}\right)^{\mathsf{T}} \mathbf{\Lambda} \left(\frac{1}{|c_{l,k_2}|} \sum_{\mathbf{y}\in c_{l,k_1}} \mathbf{y}\right) = \boldsymbol{\mu}(c_{l,k_1})^{\mathsf{T}}\mathbf{\Lambda}\boldsymbol{\mu}(c_{l,k_2}).$$

Аналогично для внутрикластерного сходства:

$$s_{\mathsf{c}}(c_{l,k_1},\, c_{l,k_1}) = \frac{1}{|c_{l,k_1}|} \sum_{\mathbf{x}\in c_{l,k_1}} \frac{1}{|c_{l,k_1}|-1} \sum_{\mathbf{y}\neq\mathbf{x},\, \mathbf{y}\in c_{l,k_1}} \mathbf{x}^{\mathsf{T}}\mathbf{\Lambda}\mathbf{y} =$$

$$= \frac{1}{|c_{l,k_1}|} \sum_{\mathbf{x}\in c_{l,k_1}} \frac{1}{|c_{l,k_1}|-1} \mathbf{x}^{\mathsf{T}}\mathbf{\Lambda}\big(|c_{l,k_1}|\boldsymbol{\mu}(c_{l,k_1}) - \mathbf{x}\big) =$$

$$= \frac{1}{|c_{l,k_1}|} \sum_{\mathbf{x}\in c_{l,k_1}} \frac{|c_{l,k_1}|}{|c_{l,k_1}|-1} \mathbf{x}^{\mathsf{T}}\mathbf{\Lambda}\boldsymbol{\mu}(c_{l,k_1}) - \frac{1}{|c_{l,k_1}|-1} =$$

$$= \frac{|c_{l,k_1}|}{|c_{l,k_1}|-1} \boldsymbol{\mu}(c_{l,k_1})^{\mathsf{T}}\mathbf{\Lambda}\boldsymbol{\mu}(c_{l,k_1}) - \frac{1}{|c_{l,k_1}|-1}.$$

В последнем выражении учтена нормировка $\mathbf{x}^{\mathsf{T}}\mathbf{\Lambda}\mathbf{x} = 1$. □

Таким образом, сходство между парой кластеров определено только средними векторами кластеров, что позволяет его эффективно пересчитывать при изменении состава кластеров.

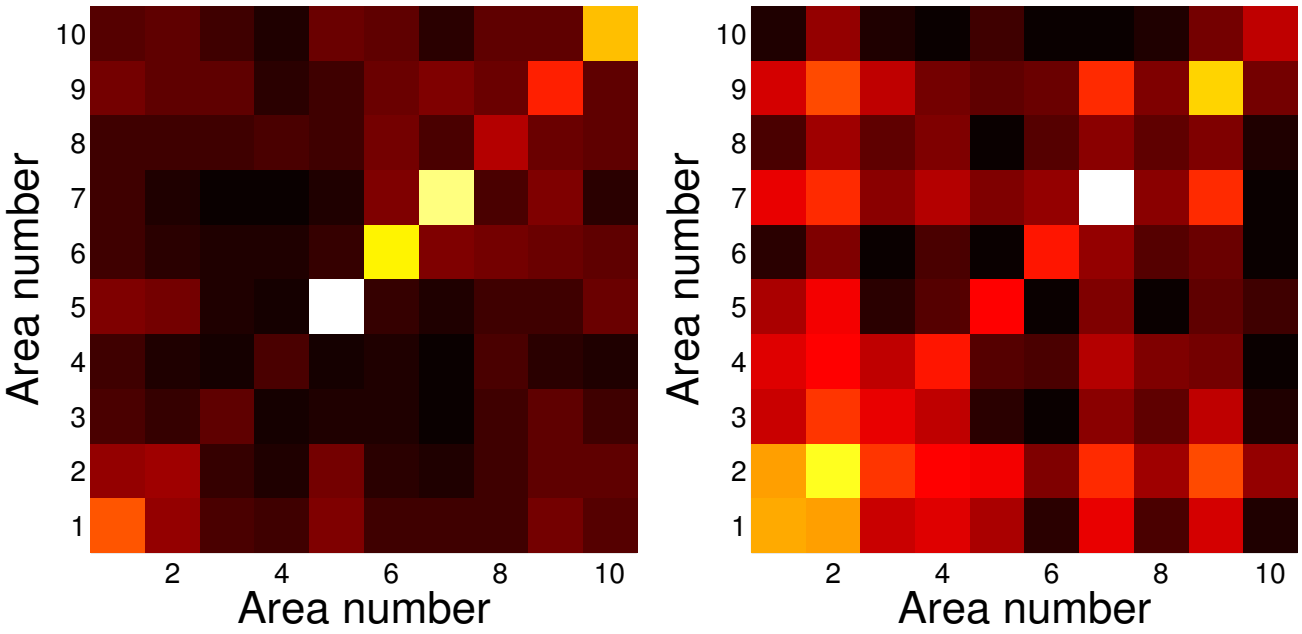
Сравнение способов векторного представления документа. Согласно разделу 1.4., документ представим в виде вектора как с помощью частоты слов, встречающихся в нем, так и с помощью языковых моделей, например, paragraph vector [93]. Для оценки качества модели, по аналогии с (2.7) и (2.8) вводится внутрикластерное (3.5) и межкластерное (3.6) сходство на заданном уровне l иерархии.

$$F_0(l) = \frac{1}{K_l} \sum_{k=1}^{K_l} s_{\mathsf{c}}(c_{l,k},\, c_{l,k}), \tag{3.5}$$

$$F_1(l) = \frac{2}{K_l(K_l-1)} \sum_{k=1}^{K_l} \sum_{k'=k+1}^{K_l} s_{\mathsf{c}}(c_{l,k},\, c_{l,k'}). \tag{3.6}$$

Качество кластерной структуры оценивается как

$$F(l) = \frac{F_0(l)}{F_1(l)} \to \max. \tag{3.7}$$



а. Частотное представление, $F(l) = 1.98$.

б. Представление с помощью языковой модели, $F(l) = 1.31$.

Рис. 3.1. Сходство между экспертными кластерами для различных представлений документов.

Предполагается, что экспертная модель M является эталонной и выбранная функция сходства и способ векторного представления документа должны отличать экспертные кластеры друг от друга. На рис. 3.1 приведена визуализация матрицы значений парного сходства (3.3) между экспертными кластерами. Значения на диагонали соответствуют внутрикластерному сходству, значения вне диагонали соответствуют межкластерному сходству. Если представление хорошо описывает экспертную модель, то значения диагональных элементов должны быть больше значений внедиагональных элементов. На рис. 3.1 а. показана матрица парного расстояния, построенная с помощью частотного представления документов в виде векторов. Ее диагональные элементы выделяются лучше, а результат (3.7) отношения внутрикластерного сходства к межкластерному равен 1.98, что значительно превосходит результат 1.31 для векторного представления документов с помощью языковой модели paragraph vector.

Сходство документа с кластером. Сходство документа $\mathbf{x}$ с кластером $c_{l,k}$ определяется как

$$s(\mathbf{x},\ c_{l,k}) = \mathbf{x}^{\mathsf{T}} \mathbf{\Lambda} \boldsymbol{\mu}(c_{l,k}) \tag{3.8}$$

с учетом введенной нормировки (3.2) и введенной функции сходства кластера к кластеру (3.3). Выражение (3.8) можно рассматривать как сходство между двумя кластерами (3.3), в котором документ $\mathbf{x}$ рассматривается как одноэлементный кластер. Для классификации неразмеченного документа $\mathbf{x}$ решается

следующая задача:

$$\hat{c}(\mathbf{x}) = \underset{c_{h,k}, k \in \{1 \ldots K_h\}}{\arg\max} \; s(\mathbf{x}, c_{h,k}). \tag{3.9}$$

В случае древовидной кластерной структуры, для определения принадлежности документа $\mathbf{x}$ к кластерам $c_{l,k}$ на каждом уровне l иерархии достаточно решить задачу (3.9) только для кластеров нижнего уровня $c_{h,k}$, так как это определит кластеры для данного документа на всех остальных уровнях. Однако, при малом размере кластеров нижнего уровня данный подход является неустойчивым. Добавление или удаление одного документа из данного кластера приведет к значительному изменению его среднего вектора $\boldsymbol{\mu}(c_{h,k})$, и, как следствие, сходства с данным кластером документов $\mathbf{x}$, уже находящихся в нем.

Другим способом решения задачи иерархической классификации является подход сверху вниз. Пусть на шаге l для документа $\mathbf{x}$ определены кластеры на первых l уровнях, и на уровне l этот документ попал в кластер $c_{l,k}$. Кластер уровня $l+1$ определяется как решение задачи (3.9) для дочерних кластеров $\{c_{l+1,k'} : B(c_{l+1,k'}) = c_{l,k}\}$, где $B(c)$ – оператор, возвращающий родительский кластер указанного кластера c с более высокого уровня иерархии. Результатом применения данного оператора l раз является родительский кластер, лежащий на l уровней выше.

Данный подход является более стабильным, так как при изменении состава кластера последнего уровня c_{h,k_h} на $\tilde{c}_{h,k_h}$, выполняется условие:

$$\begin{aligned} \|\boldsymbol{\mu}(c_{h,k_h}) - \boldsymbol{\mu}(\tilde{c}_{h,k_h})\| \;\geq\; \|\boldsymbol{\mu}\big(B\big(c_{h,k_h}\big)\big) - \boldsymbol{\mu}\big(B\big(\tilde{c}_{h,k_h}\big)\big)\| \;\geq \\ \geq \;\ldots\; \geq \; \|\boldsymbol{\mu}\big(B^{h-2}\big(c_{h,k_h}\big) - \boldsymbol{\mu}\big(B^{h-2}\big(\tilde{c}_{h,k_h}\big)\big)\|, \end{aligned} \tag{3.10}$$

поэтому изменения в кластеризации остальных документов будут, скорее всего, только на нижних уровнях иерархии. Однако при таком подходе отнесение документа на верхнем уровне иерархии l в неверный кластер сделает невозможным его попадание в нужные кластеры на более низких уровнях.

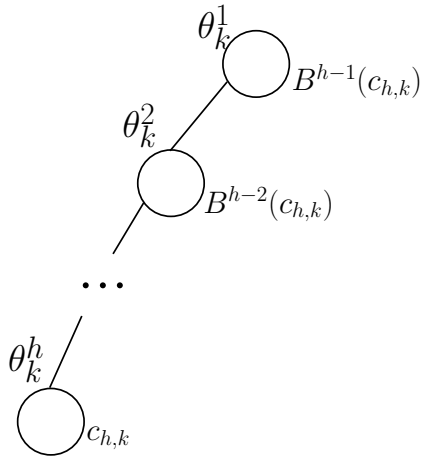


Рис. 3.2. Вычисление сходства с веткой иерархической структуры.

Пусть каждому кластеру $c_{h,k}$ нижнего уровня иерархии соответствует вектор весов $\boldsymbol{\theta}_k \in \mathbb{R}^h$, см. рис. 3.2. Сходство документа $\mathbf{x}$ с веткой дерева, начинающейся с кластера нижнего уровня $c_{h,k}$ определяется как взвешенная сумма сходства документа $\mathbf{x}$ и всех кластеров данной ветки

$$s_{\mathsf{h}}(\mathbf{x},\ c_{h,\,k}) = \sum_{l=1}^{h} \theta_k^l s\big(\mathbf{x},\ B^{h-l}(c_{h,\,k})\big). \tag{3.11}$$

При этом учитывается факт, что документ, схожий с кластером c_{h,k_h} нижнего уровня должен быть схожим со всеми его родительскими кластерами. Пусть центр родительского кластера уровня l кластера нижнего уровня $c_{h,k}$ обозначается как

$$\boldsymbol{\mu}_{l,k} = \boldsymbol{\mu}\big(B^{h-l}(c_{h,k})\big).$$

Для каждого кластера нижнего уровня $c_{h,k}$ из векторов $\boldsymbol{\mu}_{l,k}$ составим матрицу $\mathbf{M}_k$ так, чтобы столбец с номером l соответствовал вектору центра $\boldsymbol{\mu}_{l,k}$:

$$\mathbf{M}_k = [\boldsymbol{\mu}_{1,k}, \ldots, \boldsymbol{\mu}_{h,k}].$$

Определение 16. Иерархическое сходство документа $\mathbf{x}$ с кластером $c_{h,\,k}$ нижнего уровня h определяется как

$$s_{\mathsf{h}}(\mathbf{x}, c_{h,\,k}) = \sum_{l=1}^{h} \theta_k^l \mathbf{x}^{\mathsf{T}} \boldsymbol{\Lambda} \boldsymbol{\mu}_{l,k} = \mathbf{x}^{\mathsf{T}} \boldsymbol{\Lambda} \left(\sum_{l=1}^{h} \theta_k^l \boldsymbol{\mu}_{l,k} \right) = \mathbf{x}^{\mathsf{T}} \boldsymbol{\Lambda} \mathbf{M}_k \boldsymbol{\theta}_k. \tag{3.12}$$

Задача (3.9) иерархической классификации для иерархической функции сходства (3.12) принимает вид:

$$\hat{c}(\mathbf{x}) = \underset{c_{h,k},\ k \in \{1 \ldots K_h\}}{\arg\max} s_{\mathsf{h}}(\mathbf{x}, c_{h,k}). \tag{3.13}$$

3.2. Оператор релевантности

При большом количестве кластеров нижнего уровня, алгоритм классификации, решающий задачу (3.13) ошибается чаще. В данном разделе рассматривается более общая постановка задачи классификации, в которой для каждого документа необходимо найти ранжированный список кластеров нижнего уровня по убыванию их релевантности документу $\mathbf{x}$. При этом решением задачи классификации является кластер, стоящий в перестановке на первой позиции. При несовпадении экспертного мнения с данным решением, эксперт рассматривает кластер, стоящий на следующей позициях в перестановке в качестве альтернативного решения, и т.д. Пусть S^k – множество перестановок порядка k.

Определение 17. Оператором релевантности R называется оператор, ставящий в соответствие документу $\mathbf{x} \in \mathbb{R}^{|W|}$, перестановку кластеров нижнего уровня, отсортированных по убыванию релевантности документу $\mathbf{x}$:

$$R : \mathbb{R}^{|W|} \to S^{K_h}. \tag{3.14}$$

Кластер $c_{h,k}$ является наиболее релевантным для документа $\mathbf{x}$ относительно оператора релевантности R, если номер k данного кластера стоит на первом месте в перестановке, возвращаемой R.

Оценка качества оператора релевантности. Пусть имеется коллекция D с экспертной иерархической тематической моделью M. Пусть $c(\mathbf{x})$ – экспертный кластер документа $\mathbf{x}$ на уровне h. Средняя позиция экспертного кластера в перестановках $R(\cdot)$ определяется как

$$Q(R) = \frac{1}{|D|} \sum_{n=1}^{|D|} \text{pos}\big(R(\mathbf{x}_n),\, c(\mathbf{x}_n)\big), \tag{3.15}$$

где функция $\text{pos}(R(\mathbf{x}),\, c(\mathbf{x}))$ возвращает позицию экспертного кластера $c(\mathbf{x})$ в перестановке, возвращаемой $R(\mathbf{x})$. Чем меньше значение $Q(R)$, тем меньше номер позиции экспертного кластера в перестановке, которую возвращает предложенный оператор релевантности R.

Пусть кумулятивная гистограмма строится следующим образом. Столбец с номером j принимает значение

$$\frac{1}{|D|} |\{\mathbf{x} : \text{pos}\big(R(\mathbf{x}),\, c(\mathbf{x})\big) \leq j\}|, \tag{3.16}$$

где $\{\mathbf{x} : \text{pos}\big(R(\mathbf{x}),\, c(\mathbf{x})\big) \leq j\}$ – множество всех документов, для которых номер позиции экспертного кластер в перестановке $R(\mathbf{x})$ меньше либо равен j.

Определение 18. Качеством оператора релевантности R называется $\text{AUCH}(R)$ – нормированная на число кластеров площадь под верхней огибающей кумулятивной гистограммы (3.16):

$$\text{AUCH}(R) = \frac{1}{K_h |D|} \sum_{j=1}^{K_h} |\{\mathbf{x} : \text{pos}\big(R(\mathbf{x}),\, c(\mathbf{x})\big) \leq j\}|. \tag{3.17}$$

Значение $\text{AUCH}(R) = 1$ соответствует случаю, когда экспертный кластер оказывается в соответствии с R наиболее релевантным для каждого из документов выборки D.

Утверждение 10. Максимизация критерия качества $\text{AUCH}(R)$ эквивалентна минимизации средней позиции экспертного кластера в перестановке.

Доказательство. Заметим, что каждый документ вносит вклад $1/|D|$ в каждый столбец кумулятивной гистограммы начиная со столбца с номером $K_h + 1 - \text{pos}\big(R(\mathbf{x}),\, c(\mathbf{x})\big)$. Перегруппируем сумму в (3.17) так, чтобы суммирование шло по документам

$$\text{AUCH}(R) = \frac{1}{K_h|D|}\sum_{j=1}^{K_h}|\{\mathbf{x}:\, \text{pos}\big(R(\mathbf{x}),\, c(\mathbf{x})\big) \leq j\}| =$$

$$= \frac{1}{K_h|D|}\sum_{n=1}^{|D|}\big(K_h + 1 - \text{pos}\big(R(\mathbf{x}),\, c(\mathbf{x})\big)\big) = 1 + \frac{1}{K_h} - \frac{1}{K_h}Q(R).$$

Таким образом, критерий качества $\text{AUCH}(R)$ является линейным преобразованием средней позиции экспертного кластера в перестановке $Q(R)$. Так как число кластеров нижнего уровня K_h не изменяется в процессе оптимизации, то максимизация критерия $\text{AUCH}(R)$ эквивалентна минимизации средней позиции позиции экспертного кластера в перестановке. □

3.3. Энтропийная модель важности слов

Матрица $\mathbf{\Lambda}$ в функции сходства (3.13) позволяет учесть важность слов для экспертной кластеризации. В данном разделе рассматривается способ определения диагональных элементов данной матрицы по коллекции с экспертной иерархической тематической моделью. Для этого энтропийный подход оценки важности признаков, предложенный в [37], обобщен на иерархический случай.

Энтропия слова относительно кластеризации. Слова, отделяющие одни кластеры от других в экспертной тематической модели, являются наиболее важными. Рассмотрим следующую ситуацию. Пусть все документы из кластера $c_{l,k}$ содержат слово w, а документы из остальных кластеров $c_{l,k'}, k' \neq k$ не содержат слово w. Пусть нужно классифицировать документ, содержащий слово w. Справедливо предположить, что данный документ относится к кластеру $c_{l,k}$. Данный пример показывает, как некоторое слово w может отделять одни кластеры от других. Эта идея формализуется с помощью энтропии слов.

Пусть $p^l_{m,k} = p(c_{l,k}|w_m)$ – вероятность кластера при заданном слове w_m. Оценка $p^l_{m,k}$ через средние векторы $\{\boldsymbol{\mu}(c_{l,k})\}$ кластеров уровня l:

$$\mathbf{p}^l_m = \big[\mu(c_{l,1})_m,\, \ldots,\, \mu(c_{l,K_l})_m\big]^\intercal, \quad \mathbf{p}^l_m \mapsto \frac{\mathbf{p}^l_m}{\|\mathbf{p}^l_m\|_1}.$$

Определение 19. Энтропией слова w_m относительно экспертной кластеризации документов на уровне l называется

$$\mathsf{H}^l(w_m) = -\sum_{k=1}^{K_l} p^l_{m,k}\log(p^l_{m,k}). \tag{3.18}$$

Минимальное значение энтропии $\mathsf{H}^l(w_m) = 0$ соответствует случаю, когда слово w_m встречается в документах только одного кластера уровня l, выделяя тем самым его из остальных. Случай, когда $p^l_{m,k} = \text{const}$ для всех $k = 1 \dots K_l$ соответствует максимальному значению энтропии и случаю, когда слово w_m является неинформативным.

Определение важности слов через их энтропию. В случае произвольной диагональной матрицы $\mathbf{\Lambda}$, число переменных оптимизации равно размерности словаря $|W|$. Чтобы избежать переобучения, используется модель, ставящая в соответствие важности λ_m слова w_m значение функции, зависящей от энтропии данного слова и структурного параметра α_l:

$$\lambda_m = 1 + \alpha_l \log\bigl(1 + \mathsf{H}^l(w_m)\bigr). \tag{3.19}$$

Структурный параметр α_l определяет, с каким весом учитывается энтропия слова относительно экспертной кластеризации на уровне l. В иерархическом случае имеются значения энтропии слов для каждого из уровней экспертной кластеризации, и модель (3.19) принимает вид:

$$\lambda_m = 1 + \sum_{l=1}^{h} \alpha_l \log\bigl(1 + \mathsf{H}^l(w_m)\bigr). \tag{3.20}$$

Так как для каждого слова w_m и уровня l можно изначально вычислить значение $\log\bigl(1 + \mathsf{H}^l(w_m)\bigr)$, обозначим

$$\iota_{ml} = \log\bigl(1 + \mathsf{H}^l(w_m)\bigr).$$

Модель (3.20) принимает вид

$$\lambda_m = 1 + \boldsymbol{\alpha}^\intercal \boldsymbol{\iota}_m. \tag{3.21}$$

3.4. Учет векторного представления слов в функции сходства

Предложенная иерархическая взвешенная функция сходства (3.12) учитывает важность слов при классификации новых документов. Однако диагональная матрица $\mathbf{\Lambda}$ не позволяет учесть связь между словами синонимами при вычислении сходства документа и кластера.

Рассмотрим следующий модельный пример. Пусть имеется словарь $W = \{\text{math, mathematics}\}$ и кластер c состоящий из одного документа d, который содержит единственное слово “mathematics”. Требуется вычислить сходство данного кластера с документом d', состоящим из единственного слова “math”. Средний вектор кластера $\boldsymbol{\mu}(c)$ и векторное представление документа $\mathbf{x}(d')$ имеют вид:

$$\boldsymbol{\mu}(c) = [0,\ 1]^\intercal, \quad \mathbf{x}(d') = [1,\ 0]^\intercal. \tag{3.22}$$

Сходство данного документа и кластера равно

$$s(c, \mathbf{x}) = \boldsymbol{\mu}(c)^\intercal \boldsymbol{\Lambda} \mathbf{x} = 0,$$

что противоречит аналогичному результату, полученному с помощью обученной языковой модели (см. раздел 1.3.)

$$s_{\text{word2vec}}(c, \mathbf{x}) = \mathbf{w}(\text{“math”})^\intercal \mathbf{w}(\text{“mathematics”}) = 0.7, \tag{3.23}$$

так как в данном случае сходство кластера из одного слова и документа из одного слова можно свести к сходству векторных представлений данных слов.

Чтобы учесть синонимичность слов, центры кластеров c адаптируются следующим образом. Для каждого слова w_{m_1} из словаря ищется наиболее близкое к нему слово w_{m_2}, принадлежащее этому кластеру, с помощью обученной языковой модели. На позицию m_1 вектора $\boldsymbol{\mu}(c)$ ставится значение скалярного произведения векторных представлений слов w_{m_1} и w_{m_2}

$$\mu(c)_{m_1} = \mathbf{w}(w_{m_1})^\intercal \mathbf{w}(w_{m_2}). \tag{3.24}$$

При таком способе определения центров кластеров $\boldsymbol{\mu}$, центр кластера c из примера (3.22) равен

$$\boldsymbol{\mu}(c) = [0.7,\ 1]^\intercal.$$

Его сходство (3.23) с документом d' в этом случае равно 0.7, что не противоречит результату, полученному с помощью языковой моделью.

Областью значений скалярного произведения $\mathbf{w}(w_{m_1})^\intercal \mathbf{w}(w_{m_2})$ является отрезок $[-1, 1]$. Для синонимов это скалярное произведение близко к 1, поэтому чтобы учитывать только синонимы, используется дополнительное преобразование

$$\mu(c)'_m \mapsto \begin{cases} f\big(\mu(c)'_m\big) = \big(1 + \cos\big(-\pi\left(1.5 - \mu(c)_m\right)\big)\big)^p, & \text{если } \mu(c)_m \geq 0.5 \\ 0, & \text{иначе.} \end{cases} \tag{3.25}$$

Утверждение 11. Функция $f(\cdot)$ из преобразования (3.25) монотонно возрастает на интервале $(0.5, 1)$ при $p > 1$.

Доказательство. При $x = 1$, $f(x) = 1$, при $x = 0.5$, $f(x) = 0$. Производная $f(x)$ имеет вид

$$f'(x) = \pi p\big(1 + \cos\big(-\pi(1.5 - x)\big)\big)^{p-1} \sin\big(\pi(1.5 - x)\big). \tag{3.26}$$

Так как на интервале $(0.5, 1)$

$$\sin\big(\pi(1.5 - x)\big) > 0,$$

$$\big(1 + \cos\big(-\pi(1.5 - x)\big)\big)^{p-1} > 0,$$

то $f'(x) > 0$ и функция возрастает. □

Чем больше структурный параметр p, тем медленнее начинает возрастать функция f при значениях, близких к 0.5. Это позволяет регулировать допустимый уровень шума в синонимах. Так, при $p \to \infty$ учитываются только совпадающие слова, а вес синонимов стремиться к нулю.

3.5. Оптимизация параметров иерархической функции сходства

Иерархическая функция сходства содержит два набора параметров: параметры энтропийной модели $\boldsymbol{\alpha}$ и весовые параметры $\boldsymbol{\theta} = \{\boldsymbol{\theta}_k\}$ для каждого кластера нижнего уровня $c_{h,k}$. В данном разделе рассматривается способ оптимизации данных параметров с помощью максимизации функции качества AUCH. Обучающая выборка разбивается на три части $D = D_{\mathcal{V}_0} \cup D_{\mathcal{V}_1} \cup D_{\mathcal{V}_2}$, начальные значения параметров задаются как

$$\boldsymbol{\alpha} = \mathbf{0}, \quad \boldsymbol{\theta}_k = \left[\frac{1}{h}, \ldots, \frac{1}{h}\right]. \tag{3.27}$$

Алгоритм оптимизации разбивается на два шага, повторяющихся итеративно:

1) найти оптимальные значения $\boldsymbol{\alpha}$ при фиксированных параметрах $\boldsymbol{\theta}_k$ по подвыборке $D_{\mathcal{V}_1}$,
2) найти оптимальные значения параметров $\boldsymbol{\theta}_k$ при фиксированных значениях $\boldsymbol{\alpha}$ по выборке $D_{\mathcal{V}_2}$. Если $\Delta\boldsymbol{\theta}_k$ и $\Delta\boldsymbol{\alpha}$ больше заданного порога, вернуться на шаг 1.

Далее каждый из шагов описывается более подробно.

Оптимизация параметров энтропийной модели. По выборке $D_{\mathcal{V}_0}$ строятся центры кластеров $\{\boldsymbol{\mu}(c_{l,k})\}$ и по ним вычисляется энтропия слов относительно каждого уровня экспертной иерархии. Для оценки параметров модели (3.21) решается задача максимизации $\mathrm{AUCH}(R)$ (3.15) по $\alpha_1, \ldots, \alpha_h$ при фиксированных значениях $\boldsymbol{\theta}$ по выборке $D_{\mathcal{V}_1}$:

$$\boldsymbol{\alpha}^* = \arg\max_{\boldsymbol{\alpha}} \mathrm{AUCH}(R). \tag{3.28}$$

При этом должна сохраняться нормировка $\mathbf{x}^\intercal \boldsymbol{\Lambda} \mathbf{x} = 1$, поэтому после изменения $\boldsymbol{\Lambda}$ производится перенормировка векторов $\mathbf{x}$ и пересчет средних векторов $\boldsymbol{\mu}(c_{l,k})$.

Оптимизация весовых параметров. Задача оптимизации параметров $\boldsymbol{\theta}$ по выборке $D_{\mathcal{V}_2}$ при фиксированных значениях $\boldsymbol{\alpha}$ сводится к максимизации сходства документов $\mathbf{x} \in D_{\mathcal{V}_2}$ с их экспертными кластерами. Данная задача формулируется следующим образом:

$$\boldsymbol{\theta}_k^* = \arg\max_{\boldsymbol{\theta}_k} \sum_{\mathbf{x} \in c_{h,k}} \mathbf{x}^\intercal \boldsymbol{\Lambda} \mathbf{M}_k \boldsymbol{\theta}_k + \psi \|\boldsymbol{\theta}_k - \mathbf{h}\|_2^2, \tag{3.29}$$

$$\|\boldsymbol{\theta}_k\|_1 = 1, \quad \boldsymbol{\theta}_k \geq \mathbf{0}, \quad k \in \{1 \ldots K_h\}, \quad \mathbf{h} = \left[\frac{1}{h}, \ldots, \frac{1}{h}\right]^\intercal, \tag{3.30}$$

где ψ – структурный параметр регуляризации. На данном шаге значения параметров $\boldsymbol{\alpha}$ и, как следствие, матрицы $\boldsymbol{\Lambda}$ фиксированы, поэтому для всех документов $\mathbf{x} \in D_{\mathcal{V}_2}$ известны значения $\mathbf{x}^\intercal \boldsymbol{\Lambda} \mathbf{M}_k$, а задача (3.29) является задачей квадратичного программирования при $\psi \neq 0$.

Утверждение 12. При параметре регуляризации $\psi = 0$, результатом оптимизаци (3.29) является тривиальное решение в виде единичного вектора.

Доказательство. При $\psi = 0$ задача становится линейной относительно $\boldsymbol{\theta}_k$. Ее решение с учетом ограничений (3.30) достигается в вершине симплекса

$$\boldsymbol{\theta}_k = \mathbf{e}(l^*), \qquad l^* = \arg\max_l \sum_{\mathbf{x} \in c_{h,k}} \left(\mathbf{x}^\intercal \boldsymbol{\Lambda} \mathbf{M}_k\right)_l,$$

где $\mathbf{e}(l)$ – единичный вектор с единицей на позиции l. □

Теорема 2. Пусть выполняются соотношения

$$|D_{\mathcal{V}_0}| \sim |D_{\mathcal{V}_1}| \sim |D_{\mathcal{V}_2}| \sim |D|, \quad K < |D|, \quad h^3 \log_2 h < K < |D|,$$

где K – суммарное число кластеров на всех уровнях. Сложность приведенного выше оптимизационного алгоритма $O(ba^h|D||W|hK_h)$, где b – число повторений шагов 2 и 3, а a – число значений каждого из α_l в оптимизационной сетке.

Доказательство. В данном алгоритме чередуются два шага – оптимизация параметров $\boldsymbol{\alpha}$ и весовых параметров $\boldsymbol{\theta}_k$. Так как функционал AUCH является дискретным, в базовом случае для оптимизация $\boldsymbol{\alpha}$ используется сетка с a значениями каждого элемента вектора $\boldsymbol{\alpha}$. Для оптимизации $\boldsymbol{\theta}_k$ используется метод внутренней точки, так как задача является выпуклой. Распишем каждый из этапов.

1. Вычислить начальные значения
   центров кластеров $O(|D_{\mathcal{V}_0}||W|h)$,
   энтропии слов $O(|W|K)$, где $K = \sum_{l=1}^{h} K_l$.
2. Найти оптимальные $\boldsymbol{\alpha}$, вычислив AUCH при фиксированных $\boldsymbol{\theta}_k$ для всех a^h значений сетки $\boldsymbol{\alpha}$:
   иерархическое сходство с кластерами уровня h: $O(|D_{\mathcal{V}_1}||W|hK_h)$,
   качество AUCH: $O(|D_{\mathcal{V}_1}|K_h \ln K_h)$.
3. Найти оптимальные $\boldsymbol{\theta}_k$, решив K_h задач квадратичного программирования (3.29) при фиксированном $\boldsymbol{\alpha}$:
   вычислить параметры задачи $O(|D_{\mathcal{V}_2}||W|h)$,
   решить задачу $O(h^3 L)$, где $L \sim \log_2(h)$.

С учетом условия теоремы, наиболее трудоемким местом данного алгоритма является вычисление иерархического сходства a^h раз на шаге 2. Так как шаги 2 и 3 повторяются b раз, итоговая сложность алгоритма

$$O(ba^h|D||W|hK_h). \tag{3.31}$$

□

Условия в формулировке теоремы выполняются для большинства задач. Так как выборка разбивается на пропорциональные части, а число кластеров меньше любой из частей, то

$$|D_{\mathcal{V}_0}| \sim |D_{\mathcal{V}_1}| \sim |D_{\mathcal{V}_2}| \sim |D|, \quad K < |D|.$$

Число кластеров K чаще всего растет экспоненциально с увеличением числа уровней, поэтому даже при небольших h выполняется

$$h^3 \log_2 h < K.$$

Число итераций в проводимых экспериментах было $b \sim 10$. Чтобы уменьшить сложность алгоритма при большом числе уровней h, вместо учета энтропии на всех уровнях согласно модели (3.20) используется модель (3.19) для одного фиксированного уровня. В этом случае, в сложности (3.31) вместо a^h стоит a, что значительно уменьшает вычислительную сложность.

3.6. Оптимизация правдоподобия модели

Критерий качества AUCH (3.17) является дискретным, что усложняет его оптимизацию по параметрам модели $\boldsymbol{\theta}$ и $\boldsymbol{\alpha}$. Так, сложность (3.31) растет экспоненциально при увеличении числа уровней, что делает данный алгоритм плохо масштабируемым. Чтобы получить вычислительно эффективный метод оптимизации параметров, вместо максимизации критерия AUCH максимизируется правдоподобие модели:

$$L(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\alpha}) = \prod_{n=1}^{N} \prod_{k=1}^{K_h} p(z_{nk} = 1|\mathbf{x}_n, \boldsymbol{\theta}_k, \boldsymbol{\alpha})^{z_{nk}}, \tag{3.32}$$

где $z_{nk} = [\mathbf{x}_n \in c_{h,k}]$ – элемент матрицы экспертной классификации. Для удобства обозначим значение иерархического сходства s_{h} документа $\mathbf{x}_n$ и кластера $c_{h,k}$

$$s_{\mathsf{h}}(\mathbf{x}_n, c_{h,k}) = s_{n,k}, \quad \mathbf{s}_{\mathsf{h}}(\mathbf{x}_n) = \big[s_{n,1}, \ldots, s_{n,K_h}\big].$$

Вероятность документа принадлежать кластеру нижнего уровня $c_{h,k}$ оценивается с помощью функции softmax (1.6) от результата иерархической функции сходства:

$$p(\mathbf{x} \in c_{h,k}) = \operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\mathbf{x})\big)_k = \frac{\exp\big(s_{\mathsf{h}}(\mathbf{x}, c_{h,k})\big)}{\sum_{k'=1}^{K_h} \exp\big(s_{\mathsf{h}}(\mathbf{x}, c_{h,k'})\big)}. \tag{3.33}$$

Максимизация правдоподобия эквивалентна максимизации логарифма правдоподобия

$$\log L(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta},\boldsymbol{\alpha}) = \sum_{n=1}^{N}\sum_{k=1}^{K_h} z_{nk}\log p(z_{nk}=1|\mathbf{x}_n,\boldsymbol{\theta}_k,\boldsymbol{\alpha}). \tag{3.34}$$

Утверждение 13. Производные логарифма правдоподобия (3.34) по параметрам $\boldsymbol{\theta}$ и $\boldsymbol{\alpha}$ имеют вид

$$\begin{aligned}
\nabla_{\boldsymbol{\alpha}}\log L &= \sum_{m=1}^{|W|}\boldsymbol{\iota}_m\sum_{n=1}^{N}\sum_{k=1}^{K_h} x_{nm}(\mathbf{M}_k\boldsymbol{\theta}_k)_m\big[z_{nk}-\operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\mathbf{x}_n)\big)_k\big],\\
\nabla_{\boldsymbol{\theta}_k}\log L &= \sum_{n=1}^{N}\mathbf{M}_k^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{x}_n\big[z_{nk}-\operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\mathbf{x}_n)\big)_k\big].
\end{aligned} \tag{3.35}$$

Доказательство. Производные по параметрам $\boldsymbol{\alpha}$:

$$\nabla_{\boldsymbol{\alpha}}\log L = \nabla_{\boldsymbol{\alpha}}\sum_{n=1}^{N}\sum_{k=1}^{K_h} z_{nk}\left(\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k - \log\sum_{k'=1}^{K_h}\exp\big(\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_{k'}\boldsymbol{\theta}_{k'}\big)\right). \tag{3.36}$$

Подставим значения элементов матрицы $\boldsymbol{\Lambda}$ согласно энтропийной модели (3.21). Воспользуемся тем, что $\sum_{k=1}^{K_h} z_{nk} = 1$, после чего переменную суммирования k' в сумме под логарифмом переобозначим на k:

$$\begin{aligned}
\nabla_{\boldsymbol{\alpha}}\log L = \nabla_{\boldsymbol{\alpha}}&\sum_{n=1}^{N}\sum_{k=1}^{K_h}\sum_{m=1}^{|W|} z_{nk}x_{nm}(1+\boldsymbol{\alpha}^{\mathsf{T}}\boldsymbol{\iota}_m)(\mathbf{M}_k\boldsymbol{\theta}_k)_m -\\
&-\nabla_{\boldsymbol{\alpha}}\sum_{n=1}^{N}\log\sum_{k=1}^{K_h}\exp\big(\sum_{m=1}^{|W|} x_{nm}(1+\boldsymbol{\alpha}^{\mathsf{T}}\boldsymbol{\iota}_m)(\mathbf{M}_k\boldsymbol{\theta}_k)_m\big).
\end{aligned} \tag{3.37}$$

Взяв градиент и сгруппировав сгруппировав softmax из получившихся множителей, получаем

$$\nabla_{\boldsymbol{\alpha}}\log L = \sum_{m=1}^{|W|}\boldsymbol{\iota}_m\sum_{n=1}^{N}\sum_{k=1}^{K_h} x_{nm}(\mathbf{M}_k\boldsymbol{\theta}_k)_m\big[z_{nk}-\operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\mathbf{x}_n)\big)_k\big]. \tag{3.38}$$

Проводя аналогичные преобразования, градиент по параметрам $\boldsymbol{\theta}_k$:

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}_k}\log L =& \nabla_{\boldsymbol{\theta}_k}\sum_{n=1}^{N}\sum_{k=1}^{K_h} z_{nk}\left(\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k - \log\sum_{k'=1}^{K_h}\exp\big(\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_{k'}\boldsymbol{\theta}_{k'}\big)\right) =\\
=&\sum_{n=1}^{N}\mathbf{M}_k^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{x}_n\big[z_{nk}-\operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\mathbf{x}_n)\big)_k\big].
\end{aligned} \tag{3.39}$$

□

Настройка параметров $\boldsymbol{\theta}$ и $\boldsymbol{\alpha}$ производится с помощью градиентного спуска:

$$\begin{aligned} \boldsymbol{\theta}'_k &= \boldsymbol{\theta}_k + \psi_1 \nabla_{\boldsymbol{\theta}_k} \log L, \\ \boldsymbol{\alpha}' &= \boldsymbol{\alpha} + \psi_2 \nabla_{\boldsymbol{\alpha}} \log L. \end{aligned} \tag{3.40}$$

Представление иерархического сходства в виде нейронной сети. Функция softmax от иерархической функции сходства (3.33) представима в виде нейронной сети, изображенной на рис. 3.3. На вход слоя IL подается документ $\mathbf{x}$. Оба скрытых слоя HL_1, HL_2 являются линейными с тождественной функцией активации. Первый скрытый слой HL_1 разбивается на h частей, каждая из которых соответствует кластерам $c_{l,k}$ определенного уровня l. Данный слой состоит из $\sum_{l=1}^{h} K_l$ нейронов. Значением функции нейронов слоя HL_1 является значение сходства документа с соответствующим кластером. Каждый нейрон второго скрытого слоя HL_2 вычисляет иерархическое сходства (3.12) с соответствующим кластером $c_{h,k}$ нижнего уровня. Выходным слоем является softmax (1.6).

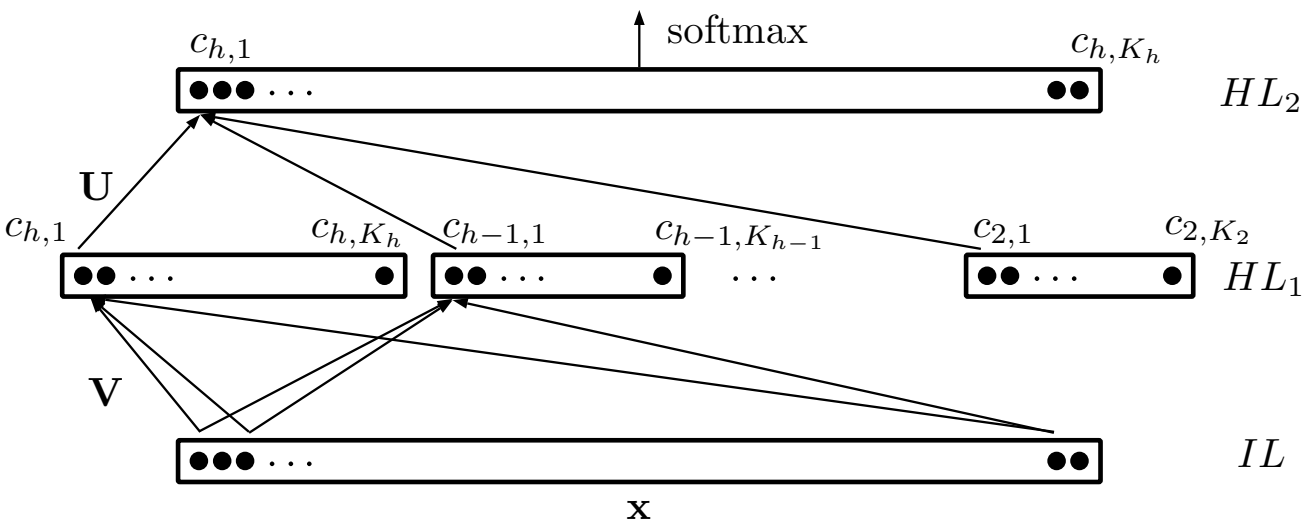


Рис. 3.3. Представление иерархической функции сходства в виде нейронной сети.

Инициализация параметров нейронной сети. Чтобы получить аналог иерархической функции сходства (3.12), начальные значения весов связей между слоями задаются следующим образом. Каждому кластеру $c_{l,k}$ в матрице $\mathbf{V}$ соответствует строка $\mathbf{v}_j^{\mathsf{T}}$ с номером j

$$\mathbf{v}_j = \boldsymbol{\Lambda}\boldsymbol{\mu}(c_{l,k}), \qquad j = k + \sum_{i=l-1}^{h} K_i, \tag{3.41}$$

где $\boldsymbol{\mu}(c_{l,k})$ – средний вектор кластера $c_{l,k}$. В этом случае значением активации нейрона с номером j слоя HL_1 является сходство документа $\mathbf{x}$ и кластера $c_{l,k}$.

Матрица весов $\mathbf{U}$ задается таким образом, чтобы каждый нейрон слоя HL_2 был соединен со всеми его родительскими кластерами, представленными нейронами слоя HL_1. Таким образом, строка $\mathbf{u}_k^\intercal$ матрицы $\mathbf{U}$ записывается в виде

$$\mathbf{u}_k^\intercal = [\,0 \ldots 0\, \theta_k^h\, 0 \ldots 0\, \theta_k^{h-1}\, 0 \ldots 0\, \theta_k^2\, 0 \ldots 0\,],$$

где параметр θ_k^l стоит на позиции

$$\operatorname{idx}\left(B^{h-l}(c_{h,k})\right) + \sum_{i=0}^{h-l} K_i,$$

а $\operatorname{idx}(\cdot)$ – функция, возвращающая индекс кластера на его уровне, $\operatorname{idx}(c_{l,k}) = k$.

Полученная нейронная сеть обучается методом обратного распространения ошибки. Так как элементы матрицы $\mathbf{V}$ являются произведением матрицы важности на вектора центров кластеров (3.41), то оптимизироваться будут значения данного произведения, а не только параметры модели $\boldsymbol{\alpha}$, влияющие на значения матрицы $\boldsymbol{\Lambda}$.

Изначально многие значения элементов матрицы $\mathbf{V}$ и $\mathbf{U}$ нулевые, так как каждый кластер содержит небольшое подмножество различных слов относительно размера всего словаря, а матрица $\mathbf{U}$ задает связи кластеров в экспертной иерархии. Однако в результате оптимизации (3.32) с помощью метода обратного распространения ошибки, большинство значений становятся отличными от нуля. Чтобы избежать этого, необходимо обнулять градиент изначально нулевых значений на каждом шаге градиентного спуска.

3.7. Байесовские оценки параметров иерархической функции сходства.

В данном разделе выводится байесовский итерационный алгоритм оптимизации параметров модели (3.12), позволяющий одновременно оптимизировать все параметры модели по обучающей выборке с учетом их априорного распределения. В разделе 3.5. чтобы избежать тривиального решения для $\boldsymbol{\theta}_k$, к введенной функции качества (3.29) добавлялся квадратичный регуляризатор. В данном разделе для введения ограничений на параметры используются следующие вероятностные предположения:

$$p(\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\alpha}|\mathbf{0}, a^{-1}\mathbf{I}), \quad p(\boldsymbol{\theta}_k) = \mathcal{N}(\boldsymbol{\theta}_k|\mathbf{m}_k, \mathbf{V}_k^{-1}). \tag{3.42}$$

Так как $\boldsymbol{\alpha}$ характеризует влияние энтропии слов на их важность, а при нулевом значении $\boldsymbol{\alpha}$ влияние энтропии не учитывается, априорное распределение $p(\boldsymbol{\alpha})$ имеет нулевое математическое ожидание и диагональную корреляционную матрицу. Вектор параметров $\boldsymbol{\theta}_k$ имеет неизвестное математическое ожидание и ковариационную матрицу, на эти гиперпараметры накладываются априорные распределения

$$p(\mathbf{m}_k|\mathbf{V}_k) = \mathcal{N}(\mathbf{m}_k|\mathbf{m}_0, (b\mathbf{V}_k)^{-1}), \quad p(\mathbf{V}_k) = \mathcal{W}(\mathbf{V}_k|\mathbf{W}, \nu), \tag{3.43}$$

где $\mathcal{W}$ – распределение Уишарта.

Правдоподобие модели задается как (3.32), где вероятность документа принадлежать кластеру $c_{h,k}$ нижнего уровня оценивается с помощью функции softmax (3.33). При дальнейшем изложении считается, что матрица $\mathbf{X}$ векторных представлений документов $\mathbf{x}$ известна и не указывается в параметрах условных распределений. Общая вероятностная модель имеет вид:

$$p(\mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) = L(\mathbf{Z}|\boldsymbol{\theta}, \boldsymbol{\alpha}) p(\boldsymbol{\theta}|\mathbf{m}, \mathbf{V}) p(\mathbf{m}|\mathbf{V}) p(\mathbf{V}) p(\boldsymbol{\alpha}). \tag{3.44}$$

Из-за нелинейной зависимости правдоподобия L от параметров модели $\boldsymbol{\theta}$ и $\boldsymbol{\alpha}$, аналитический вывод апостериорного распределения параметров с учетом обучающей выборки невозможен. Для получения оценок используется вариационный вывод [28, 27, 117].

Вариационный вывод. Для любой функции плотности вероятности q, распределение $p(\mathbf{Z})$ наблюдаемых переменных $\mathbf{Z}$ представимо в виде

$$\begin{aligned} \ln p(\mathbf{Z}) &= \mathcal{L}(q) + \mathrm{KL}(q\|p), \quad \text{где} \\ \mathcal{L}(q) &= \int q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) \ln \left(\frac{p(\mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})}{q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})} \right) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{m} \mathrm{d}\mathbf{V} \mathrm{d}\boldsymbol{\alpha}, \\ \mathrm{KL}(q\|p) &= -\int q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) \ln \left(\frac{p(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z})}{q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})} \right) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{m} \mathrm{d}\mathbf{V} \mathrm{d}\boldsymbol{\alpha}. \end{aligned} \tag{3.45}$$

Дивергенция Кульбака-Лейблера KL больше либо равна нулю, поэтому $\mathcal{L}(q)$ является нижней границей $\ln p(\mathbf{Z})$. В случае, когда вычисление $p(\mathbf{Z})$ и $p(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z})$ в явном виде невозможно, в [29] предлагается максимизировать нижнюю границу $\mathcal{L}(q)$ по неизвестному распределению скрытых параметров $q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$, что эквивалентно минимизации дивергенции KL, так как левая часть не зависит от q. При произвольной функции q, результатом максимизации нижней границы является $q = p(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z})$, так как дивергенция $\mathrm{KL}(q\|p) = 0$ тогда и только тогда, когда $q = p$. Однако если апостериорное распределение $p(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z})$ аналитически получить невозможно, то и построить $q = p$ также невозможно. Поэтому в [29] предлагается выбрать определенный класс функций q, и среди них найти такую, при которой $\mathcal{L}(q)$ будет максимальной.

В нашем случае в качестве класса функций q рассматривается

$$q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) = q(\boldsymbol{\theta}) q(\mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}), \tag{3.46}$$

где $q(\boldsymbol{\theta})$ и $q(\mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$ – факторы, на которые распадается $q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$. Нижнюю границу $\mathcal{L}$ можно представить в виде

$$\begin{aligned} \mathcal{L}(q) &= \int q(\boldsymbol{\theta}) q(\mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) \ln \left(\frac{p(\mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})}{q(\boldsymbol{\theta}) q(\mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})} \right) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{m} \mathrm{d}\mathbf{V} \mathrm{d}\boldsymbol{\alpha} \\ &= \int q(\boldsymbol{\theta}) \ln \tilde{p}(\mathbf{Z}, \boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} - \int q(\boldsymbol{\theta}) \ln q(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} + \mathrm{const}(\boldsymbol{\theta}), \quad \text{где} \end{aligned} \tag{3.47}$$

$$\begin{aligned}\ln\tilde{p}(\mathbf{Z},\boldsymbol{\theta}) = \int q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha})\ln p(\mathbf{Z},\boldsymbol{\theta},\mathbf{m},\mathbf{V},\boldsymbol{\alpha})\mathrm{d}\mathbf{m}\mathrm{d}\mathbf{V}\mathrm{d}\boldsymbol{\alpha} = \\ = \mathsf{E}_{\mathbf{m},\mathbf{V},\boldsymbol{\alpha}}\big[\ln p(\mathbf{Z},\boldsymbol{\theta},\mathbf{m},\mathbf{V},\boldsymbol{\alpha})\big] + \mathrm{const}\,.\end{aligned} \tag{3.48}$$

Первые два члена во второй строчки выражения (3.47) являются дивергенцией Кульбака-Лейблера между $q(\boldsymbol{\theta})$ и $\tilde{p}(\mathbf{Z},\boldsymbol{\theta})$, поэтому при фиксированной функции $q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha})$ максимум по $q(\boldsymbol{\theta})$ достигается при

$$\ln q(\boldsymbol{\theta}) = \mathsf{E}_{\mathbf{m},\mathbf{V},\boldsymbol{\alpha}}\big[\ln p(\mathbf{Z},\boldsymbol{\theta},\mathbf{m},\mathbf{V},\boldsymbol{\alpha})\big] + \mathrm{const}(\boldsymbol{\theta}), \tag{3.49}$$

где $\mathrm{const}(\boldsymbol{\theta})$ – некоторая функция, не зависящая от $\boldsymbol{\theta}$. Проведя аналогичную группировку относительно $q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha})$ получаем максимум $\mathcal{L}$ при фиксированном факторе $q(\boldsymbol{\theta})$:

$$\ln q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha}) = \mathsf{E}_{\boldsymbol{\theta}}\big[\ln p(\mathbf{Z},\boldsymbol{\theta},\mathbf{m},\mathbf{V},\boldsymbol{\alpha})\big] + \mathrm{const}(\mathbf{m},\mathbf{V},\boldsymbol{\alpha}). \tag{3.50}$$

Утверждение 14. Алгоритм оптимизации параметров распределения q, итеративно обновляющий факторы (3.49) и (3.50), сходится.

Доказательство. При пересчете каждого из факторов согласно (3.49) и (3.50) минимизируется дивергенция $KL(q\|\tilde{p})$, поэтому значение $\mathcal{L}(q)$ не убывает на каждом шаге. Так как нижняя граница $\mathcal{L}(q)$ ограничена сверху, данная процедура сойдется. □

Так как вероятность принадлежать классу задается с помощью функции softmax, знаменатель которой содержит сумму экспонент, аналитический вывод (3.50) и (3.49) невозможен. Поэтому вместо softmax используется ее верхняя параметрическая оценка.

Оценка функции softmax. Воспользуемся локальным вариационным методом [118] для получения верхней границы для данной функции. Обозначим

$$g(\mathbf{x}) = \sum_{k=1}^{K_h}\exp(x_k).$$

На рис. 3.4 изображена функция $\tilde{g}(\mathbf{x}) = -\ln\big(g(\mathbf{x})\big)$. Она является вогнутой функцией, поэтому касательная плоскость

$$y(\mathbf{x},\boldsymbol{\xi}) = -\ln\big(g(\boldsymbol{\xi})\big) - \nabla\ln\big(g(\boldsymbol{\xi})\big)^{\mathsf{T}}(\mathbf{x}-\boldsymbol{\xi}), \tag{3.51}$$

проходящая через точку $\boldsymbol{\xi}$, как показано на рис. 3.6, лежит всегда выше:

$$-\ln\big(g(\boldsymbol{\xi})\big) \le y(\mathbf{x},\boldsymbol{\xi}),$$

при этом в точке $\mathbf{x} = \boldsymbol{\xi}$ их значения совпадают. Взяв экспоненту от левой и правой части получившегося неравенства и подставив значение градиента функции $g(\mathbf{x})$ получаем верхнюю оценку

$$\frac{1}{g(\mathbf{x})} \leq \frac{1}{g(\boldsymbol{\xi})} \exp\left(\sum_{k=1}^{K_h} \frac{\exp(\xi_k)}{g(\boldsymbol{\xi})} (\xi_k - x_k) \right). \tag{3.52}$$

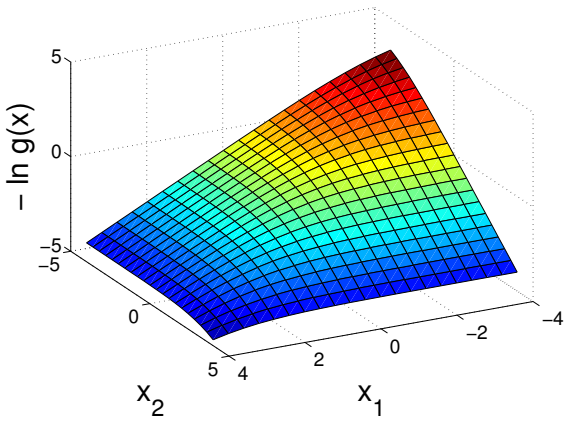


Рис. 3.4. Значения функции $\tilde{g} = -\ln g(\mathbf{x})$ в случае размерности $\mathbf{x}$ равной два.

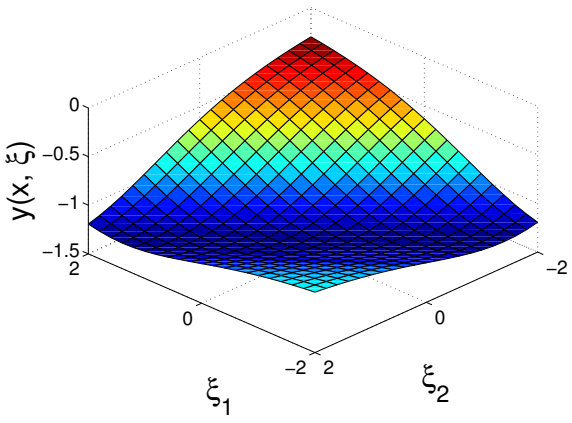


Рис. 3.5. Зависимость $y(\mathbf{x}, \boldsymbol{\xi})$ от $\boldsymbol{\xi}$ при фиксированном значении $\mathbf{x}$.

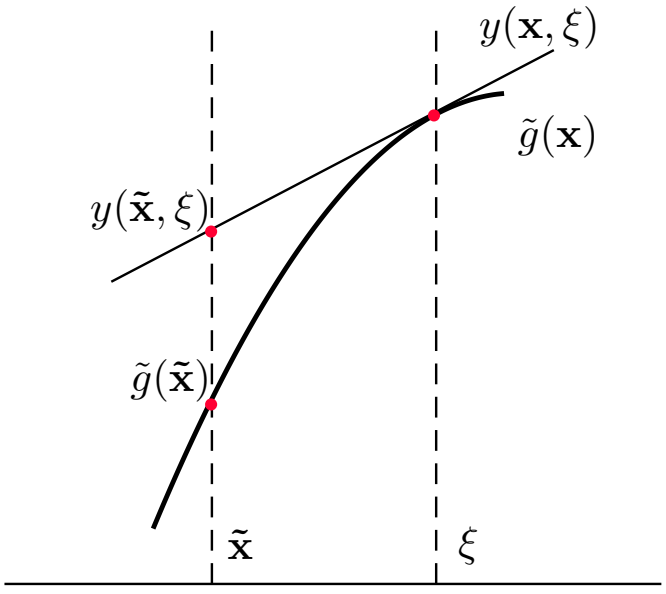


Рис. 3.6. Функция $\tilde{g} = -\ln g(\mathbf{x})$ и касательная к ней в точке $\boldsymbol{\xi}$.

Показатель экспоненты в правой части неравенства линейно зависит от $\mathbf{x}$, поэтому при свертке данной оценки с распределениями из экспоненциального семейства получаются аналогичные распределения. Используя данную оценку

функции softmax для каждого $\mathbf{x}_n$, мы получаем верхнюю оценку $\mathcal{L}(q)$:

$$\mathcal{L}(q) \leq \hat{\mathcal{L}}(q, \boldsymbol{\xi}), \tag{3.53}$$

где $\boldsymbol{\xi} = \{\boldsymbol{\xi}_n\}$. Чтобы найти наиболее близкую границу для $\mathcal{L}(q)$, значения $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$ минимизируются по $\boldsymbol{\xi}$. В дальнейшем ищется аппроксимация апостериорного распределения q, максимизирующая оценку нижней границы $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$ при фиксированных значениях $\boldsymbol{\xi}$, после чего оценка нижней границы $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$, минимизируется по вариационным параметрам $\boldsymbol{\xi}$ при фиксированных факторах q. Совместную модель (3.44), в которой логарифм правдоподобия был замещен его верхней оценкой, будем обозначать $\hat{p}(\mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$.

Оценка апостериорного распределения. Воспользовавшись верхней оценкой (3.52) для функции softmax в правдоподобии (3.32) и взяв логарифм от обеих частей

$$\begin{aligned}\ln \hat{L}(\mathbf{Z}|\boldsymbol{\theta}, \boldsymbol{\alpha}) &= \sum_{n=1}^{N} - \ln g(\boldsymbol{\xi}_n) + \sum_{k=1}^{K_h} z_{nk} \left(s_{n,k} + \sum_{k'=1}^{K_h} \frac{\exp(\xi_{nk'})}{g(\boldsymbol{\xi}_n)} (\xi_{nk'} - s_{n,k'}) \right) = \\ &= \sum_{n=1}^{N} - \ln g(\boldsymbol{\xi}_n) + \sum_{k=1}^{K_h} \left[\mathbf{x}_n^{\intercal} \boldsymbol{\Lambda} \mathbf{M}_k \boldsymbol{\theta}_k \left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)} \right) + \frac{\xi_{nk} \exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)} \right].\end{aligned} \tag{3.54}$$

Согласно предположениям (3.42) и (3.43), априорные распределения параметров имеют вид

$$\begin{aligned} p(\boldsymbol{\alpha}) &= \frac{a^{h/2}}{(2\pi)^{h/2}} \exp\left(-\frac{a}{2} \boldsymbol{\alpha}^{\intercal} \boldsymbol{\alpha} \right), \\ p(\mathbf{m}_k | \mathbf{m}_0, (b\mathbf{V}_k)^{-1}) &= \frac{|b\mathbf{V}_k|^{1/2}}{(2\pi)^{h/2}} \exp\left(-\frac{b}{2} (\mathbf{m}_k - \mathbf{m}_0)^{\intercal} \mathbf{V}_k (\mathbf{m}_k - \mathbf{m}_0) \right), \\ p(\boldsymbol{\theta}_k | \mathbf{m}_k, \mathbf{V}_k^{-1}) &= \frac{|\mathbf{V}_k|^{1/2}}{(2\pi)^{h/2}} \exp\left(-\frac{1}{2} (\boldsymbol{\theta}_k - \mathbf{m}_k)^{\intercal} \mathbf{V}_k (\boldsymbol{\theta}_k - \mathbf{m}_k) \right), \\ p(\mathbf{V}_k) &= B(\mathbf{W}, \nu) |\mathbf{V}_k|^{(\nu - h - 1)/2} \exp\left(-\frac{1}{2} \operatorname{Tr}\left(\mathbf{W}^{-1} \mathbf{V}_k \right) \right), \\ B(\mathbf{W}, \nu) &= |\mathbf{W}|^{-\nu/2} \left(2^{h\nu/2} \pi^{h(h-1)/4} \prod_{l=1}^{h} \Gamma\left(\frac{\nu + 1 - l}{2} \right) \right)^{-1}. \end{aligned} \tag{3.55}$$

Согласно (3.50), логарифм фактора $q(\mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$ в случае оценки $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$ при

фиксированных значениях $\boldsymbol{\xi}$ имеет вид

$$\begin{aligned}\ln q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha}) &= \mathsf{E}_{\boldsymbol{\theta}}\big(\ln \hat{p}(\mathbf{Z},\boldsymbol{\theta},\mathbf{m},\mathbf{V},\boldsymbol{\alpha})\big) + \text{const}(\mathbf{m},\mathbf{V},\boldsymbol{\alpha}) = \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K_h} \mathbf{x}_n^{\intercal}\boldsymbol{\Lambda}\mathbf{M}_k\mathsf{E}_{\boldsymbol{\theta}_k}\boldsymbol{\theta}_k\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right) + \\ &+ \frac{1}{2}\sum_{k=1}^{K_h}\ln|\mathbf{V}_k| - \mathsf{E}_{\boldsymbol{\theta}_k}(\boldsymbol{\theta}_k - \mathbf{m}_k)^{\intercal}\mathbf{V}_k(\boldsymbol{\theta}_k - \mathbf{m}_k) + \\ &+ \ln|b\mathbf{V}_k| - b(\mathbf{m}_k - \mathbf{m}_0)^{\intercal}\mathbf{V}_k(\mathbf{m}_k - \mathbf{m}_0) + \\ &+ (\nu - h - 1)\ln|\mathbf{V}_k| - \text{Tr}\big(\mathbf{W}^{-1}\mathbf{V}_k\big) - \\ &- a\boldsymbol{\alpha}^{\intercal}\boldsymbol{\alpha} + \text{const}(\mathbf{m},\mathbf{V},\boldsymbol{\alpha}).\end{aligned} \tag{3.56}$$

Таким образом, фактор $q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha})$ разбивается на произведение факторов

$$q(\mathbf{m},\mathbf{V},\boldsymbol{\alpha}) = q(\boldsymbol{\alpha})\prod_{k=1}^{K_h} q(\mathbf{m}_k|\mathbf{V}_k)q(\mathbf{V}_k).$$

Лемма 1. Фактор $q(\mathbf{m}_k|\mathbf{V}_k)$ имеет нормальное распределение.

Доказательство. Группировка слагаемых (3.56), содержащих $\mathbf{m}_k$ дает

$$\begin{aligned}\ln q(\mathbf{m}_k) &= -\frac{1}{2}\mathbf{m}_k^{\intercal}(1+b)\mathbf{V}_k\mathbf{m}_k + \mathbf{m}_k^{\intercal}(1+b)\mathbf{V}_k\frac{(\mathsf{E}\boldsymbol{\theta}_k + b\mathbf{m}_0)}{1+b} + \text{const}(\mathbf{m}_k) = \\ &= -\frac{1}{2}(\mathbf{m}_k - \mathbf{m}_{0k})^{\intercal}(b'\mathbf{V}_k)(\mathbf{m}_k - \mathbf{m}_{0k}) + \text{const}(\mathbf{m}_k).\end{aligned} \tag{3.57}$$

Логарифм фактора $q(\mathbf{m}_k|\mathbf{V}_k)$ имеет вид квадратичной формы, значит $q(\mathbf{m}_k|\mathbf{V}_k)$ имеет нормальное распределение

$$\begin{aligned} q(\mathbf{m}_k) &= \mathcal{N}(\mathbf{m}_k|\mathbf{m}_{0k},(b'\mathbf{V}_k)^{-1}), \\ \mathbf{m}_{0k} &= \frac{\mathsf{E}\boldsymbol{\theta}_k + b\mathbf{m}_0}{1+b}, \\ b' &= 1+b. \end{aligned} \tag{3.58}$$

□

Лемма 2. Фактор $q(\mathbf{V}_k)$ имеет распределение Уишарта.

Доказательство. Для группировки слагаемых, содержащих $\mathbf{V}_k$,

из $\ln q(\mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$ вычитается $\ln q(\mathbf{m}|\mathbf{V})$:

$$\begin{aligned}\ln q(\mathbf{V}_k) = \ln q(\mathbf{m}_k, \mathbf{V}_k) - \ln q(\mathbf{m}_k|\mathbf{V}_k) = \frac{1}{2}\ln|\mathbf{V}_k|^{(\nu-h-1+1)} + \text{const}(\mathbf{V}_k) - \\ - \frac{1}{2}\operatorname{Tr}\left(\frac{1}{1+b}(\mathsf{E}\boldsymbol{\theta}_k + b\mathbf{m}_0)(\mathsf{E}\boldsymbol{\theta}_k + b\mathbf{m}_0)^\intercal + b\mathbf{m}_0\mathbf{m}_0^\intercal + \mathsf{E}\left[\boldsymbol{\theta}_k\boldsymbol{\theta}_k^\intercal\right] + \mathbf{W}^{-1}\right) = \\ = \ln\mathcal{W}(\mathbf{W}_k, \nu') + \text{const}(\mathbf{V}_k),\end{aligned} \tag{3.59}$$

где параметры $\mathbf{W}_k$ и ν' задаются как

$$\begin{aligned}\mathbf{W}_k^{-1} &= b'\mathbf{m}_{0k}\mathbf{m}_{0k}^\intercal + b\mathbf{m}_0\mathbf{m}_0^\intercal + \mathsf{E}\left[\boldsymbol{\theta}_k\boldsymbol{\theta}_k^\intercal\right] + \mathbf{W}^{-1}, \\ \nu' &= \nu + 1.\end{aligned} \tag{3.60}$$

□

Лемма 3. Фактор $q(\boldsymbol{\alpha})$ имеет нормальное распределение.

Доказательство. Фактор $q(\boldsymbol{\alpha})$ задается оставшимися слагаемыми (3.56)

$$\begin{aligned}\ln q(\boldsymbol{\alpha}) &= -\frac{a}{2}\boldsymbol{\alpha}^\intercal\boldsymbol{\alpha} + \sum_{n=1}^{N}\sum_{k=1}^{K_h}\mathbf{x}_n^\intercal\boldsymbol{\Lambda}\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right) + \text{const}(\boldsymbol{\alpha}) = \\ &= -\frac{a}{2}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^\intercal\mathbf{I}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + \text{const}(\boldsymbol{\alpha}), \\ \boldsymbol{\alpha}_0 &= \frac{1}{a}\sum_{n=1}^{N}\sum_{m=1}^{|W|}x_{nm}\boldsymbol{\iota}_m\sum_{k=1}^{K_h}(\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k)_m\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right).\end{aligned} \tag{3.61}$$

Таким образом, фактор $q(\boldsymbol{\alpha})$ имеет нормальное распределение с параметрами $\mathcal{N}(\boldsymbol{\alpha}_0, a^{-1}\mathbf{I})$. □

Согласно (3.49), логарифм фактора $q(\boldsymbol{\theta})$ в случае оценки $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$ при фиксированных значениях $\boldsymbol{\xi}$ имеет вид

$$\begin{aligned}\ln q(\boldsymbol{\theta}) &= \mathsf{E}_{\mathbf{m},\mathbf{V},\boldsymbol{\alpha}}\big(\ln\hat{p}(\mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})\big) + \text{const}(\boldsymbol{\theta}) = \\ &= \mathsf{E}_{\boldsymbol{\alpha}}\big(\ln L(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\alpha})\big) + \mathsf{E}_{\mathbf{m},\mathbf{V}}\big(\ln p(\boldsymbol{\theta}|\mathbf{m}, \mathbf{V})\big) + \text{const}(\boldsymbol{\theta}) = \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K_h}\mathbf{x}_n^\intercal\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right) - \\ &- \frac{1}{2}\sum_{k=1}^{K_h}\mathsf{E}_{\mathbf{m}_k,\mathbf{V}_k}(\boldsymbol{\theta}_k - \mathbf{m}_k)^\intercal\mathbf{V}_k(\boldsymbol{\theta}_k - \mathbf{m}_k) + \text{const}(\boldsymbol{\theta}).\end{aligned} \tag{3.62}$$

Лемма 4. Фактор $q(\boldsymbol{\theta})$ имеет вид

$$q(\boldsymbol{\theta}) = \prod_{k=1}^{K_h} q(\boldsymbol{\theta}_k), \tag{3.63}$$

где каждый из факторов $q(\boldsymbol{\theta}_k)$ распределен нормально.

Доказательство. $\ln q(\boldsymbol{\theta})$ представим в виде суммы слагаемых, каждое из которых зависит только от $\boldsymbol{\theta}_k$, а значит $q(\boldsymbol{\theta})$ представим в виде (3.63). Согласно леммам 1 и 2 факторы $q(\mathbf{m}_k, \mathbf{V}_k)$ имеют распределения Гаусса-Уишарта (3.55), значит математическое ожидание в (3.62) вычисляется как

$$\mathsf{E}_{\mathbf{m}_k,\mathbf{V}_k}(\boldsymbol{\theta}_k - \mathbf{m}_k)^\intercal \mathbf{V}_k(\boldsymbol{\theta}_k - \mathbf{m}_k) = h(b')^{-1} + \nu'(\boldsymbol{\theta}_k - \mathbf{m}_{0k})^\intercal \mathbf{W}_k(\boldsymbol{\theta}_k - \mathbf{m}_{0k}), \quad (3.64)$$

а фактор $q(\boldsymbol{\theta}_k)$ принимает вид

$$\begin{aligned} \ln q(\boldsymbol{\theta}_k) &= -\frac{\nu'}{2}(\boldsymbol{\theta}_k - \mathbf{m}'_{0k})^\intercal \mathbf{W}_k(\boldsymbol{\theta}_k - \mathbf{m}'_{0k}) + \text{const}(\boldsymbol{\theta}_k), \\ \mathbf{m}'_{0k} &= \mathbf{m}_{0k} + \frac{1}{\nu'}(\mathbf{W}_k^{-1})^\intercal \mathbf{M}_k^\intercal \mathsf{E}_{\boldsymbol{\alpha}} \boldsymbol{\Lambda} \sum_{n=1}^{N} \mathbf{x}_n \left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)} \right). \end{aligned} \quad (3.65)$$

Таким образом, фактор $q(\boldsymbol{\theta}_k)$ имеет нормальное распределение $\mathcal{N}(\mathbf{m}'_{0k}, (\nu'\mathbf{W}_k)^{-1})$. □

Теорема 3. Функцией q из класса (3.46), заданной оптимальными оценками факторов (3.50) и (3.49), в которых правдоподобие L (3.32) оценивается с помощью верхней оценки функции softmax (3.52), а априорные распределения параметров $\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}$ задаются как (3.43) и (3.42), является

$$\begin{aligned} q(\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) &= q(\boldsymbol{\alpha}) \prod_{k=1}^{K_h} q(\boldsymbol{\theta}_k) q(\mathbf{m}_k|\mathbf{V}_k) q(\mathbf{V}_k), \\ q(\boldsymbol{\theta}_k) &\sim \mathcal{N}(\mathbf{m}'_{0k}, (\nu'\mathbf{V}_k)^{-1}), \\ q(\mathbf{m}_k|\mathbf{V}_k) &\sim \mathcal{N}(\mathbf{m}_{0k}, (b'\mathbf{V}_k)^{-1}), \\ q(\mathbf{V}_k) &\sim \mathcal{W}(\mathbf{W}_k, \nu'), \\ q(\boldsymbol{\alpha}) &\sim \mathcal{N}(\boldsymbol{\alpha}_0, a^{-1}\mathbf{I}), \end{aligned} \quad (3.66)$$

с параметрами, заданными (3.65), (3.58), (3.60) и (3.61) соответственно.

Доказательство. Подставляя результаты лемм 1-4 в (3.46) получаем утверждение теоремы. □

Теорема 4. Значение вариационного параметра ξ_{nk}, минимизирующее оценку $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$ при фиксированной функции q, совпадает со значением иерархической функции сходства в точке $\mathbf{x}_n$ для класса k, использующей в качестве параметров

$$\begin{aligned} \hat{\boldsymbol{\theta}}_k &= \mathsf{E}\boldsymbol{\theta}_k = \mathbf{m}'_{0k}, \\ \hat{\boldsymbol{\alpha}} &= \mathsf{E}\boldsymbol{\alpha} = \boldsymbol{\alpha}_0. \end{aligned} \quad (3.67)$$

Доказательство. Обозначим

$$\tilde{\mathbf{\Lambda}} = \mathrm{diag}(\tilde{\lambda}_1, \ldots, \tilde{\lambda}_{|W|}), \quad \tilde{\lambda}_m = 1 + \boldsymbol{\alpha}_0^\intercal \boldsymbol{\iota}_m,$$

$$\tilde{s}_{n,k} = \mathbf{x}_n^\intercal \tilde{\mathbf{\Lambda}} \mathbf{M}_k \mathbf{m}'_{0k}, \quad \tilde{\mathbf{s}}_n = [\tilde{s}_{n,1}, \ldots, \tilde{s}_{n,K_h}]^\intercal.$$

Согласно (3.45), (3.52) и (3.46)

$$\begin{aligned} \mathcal{L}(q) \leq \hat{\mathcal{L}}(q, \boldsymbol{\xi}) = \int q(\boldsymbol{\theta}) q(\boldsymbol{\alpha}) \ln p(\mathbf{Z}|\boldsymbol{\theta}, \boldsymbol{\alpha}, \boldsymbol{\xi}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\boldsymbol{\alpha} + \mathrm{const}(\boldsymbol{\xi}) = \\ = \sum_{n=1}^{N} - \ln g(\boldsymbol{\xi}_n) + \sum_{k=1}^{K_h} \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)} \left(\xi_{nk} - \mathbf{x}_n^\intercal \tilde{\mathbf{\Lambda}} \mathbf{M}_k \mathbf{m}'_{0k} \right) + \mathrm{const}(\boldsymbol{\xi}). \end{aligned} \tag{3.68}$$

Выражение (3.68) представимо в виде суммы выражений (3.51) с точностью до константы $\mathrm{const}(\boldsymbol{\xi})$, не зависящей от $\boldsymbol{\xi}$:

$$\hat{\mathcal{L}}(q, \boldsymbol{\xi}) = \sum_n y(\tilde{\mathbf{s}}_n, \boldsymbol{\xi}_n) + \mathrm{const}(\boldsymbol{\xi}_n). \tag{3.69}$$

Выбирая $\boldsymbol{\xi}_n$, мы выбираем точку, через которую проходит касательная $y(\tilde{\mathbf{s}}_n, \boldsymbol{\xi}_n)$, как показано на рис. 3.6. На данном рисунке значение $y(\tilde{\mathbf{s}}_n, \boldsymbol{\xi}_n)$ определяет ординату пересечения касательной и прямой $\mathbf{x} = \tilde{\mathbf{s}}_n$. С учетом вогнутости $\tilde{g}$, минимум достигается в точке

$$\boldsymbol{\xi}_n = \tilde{\mathbf{s}}_n. \tag{3.70}$$

□

Стоит отметить, что данный минимум не является единственным, как показано на рис. 3.5. При добавлении ко всем компонентам вектора $\boldsymbol{\xi}_n$ одинакового значения, результат (3.69) не изменится.

Алгоритм оптимизации параметров апостериорного распределения. Параметры распределения (3.66) содержат неизвестные математические ожидания $\mathsf{E}\boldsymbol{\theta}_k, \mathsf{E}[\boldsymbol{\theta}_k \boldsymbol{\theta}_k^\intercal], \mathsf{E}\mathbf{\Lambda}$ и вариационные параметры $\boldsymbol{\xi}_n$. Используя результаты теорем 3 и 4 получаем следующий EM-алгоритм для поиска их значений:

1. Инициализировать параметры

$$\mathbf{W}, \nu, \mathbf{m}_0, a, b, \mathbf{W}_k = \mathbf{W}, \nu' = \nu + 1, b' = b + 1, \mathbf{m}_{0k} = \mathbf{m}_0, \boldsymbol{\xi}_n.$$

2. Вычислить $\mathsf{E}\boldsymbol{\theta}_k, \mathsf{E}[\boldsymbol{\theta}_k \boldsymbol{\theta}_k^\intercal]$ с помощью распределений $q(\boldsymbol{\theta}_k)$

$$\begin{aligned} \mathsf{E}\boldsymbol{\theta}_k &= \mathbf{m}'_{0k}, \\ \mathsf{E}[\boldsymbol{\theta}_k \boldsymbol{\theta}_k^\intercal] &= (\nu' \mathbf{W}_k)^{-1} + \mathbf{m}'_{0k} (\mathbf{m}'_{0k})^\intercal, \end{aligned} \tag{3.71}$$

   и пересчитать значения параметров распределений $q(\mathbf{m}), q(\mathbf{V}), q(\boldsymbol{\alpha})$ согласно (3.58), (3.60) и (3.61).

3. Вычислить $\mathsf{E}\boldsymbol{\Lambda}$ с помощью распределения $q(\boldsymbol{\alpha})$

$$\mathsf{E}\boldsymbol{\Lambda} = \tilde{\boldsymbol{\Lambda}} = \operatorname{diag}(\{\lambda'_m\}), \quad \lambda'_m = 1 + \boldsymbol{\alpha}_0^{\mathsf{T}}\boldsymbol{\iota}_m, \tag{3.72}$$

   и пересчитать значения параметров распределения $q(\boldsymbol{\theta}_k)$ согласно (3.65).
4. Пересчитать вариационные параметры $\boldsymbol{\xi}_n = \tilde{\mathbf{s}}_n$ согласно (3.70). Если на одном из шагов 2-4 параметры изменились значимо, вернуться на шаг 2.

Шаг 2 и шаг 3 данного алгоритма являются E шагом, на котором обновляются параметры распределений. Шаг 4 является M шагом, на котором минимизируется $\hat{\mathcal{L}}(q, \boldsymbol{\xi})$ по вариационным параметрам $\boldsymbol{\xi}$.

Предсказание класса нового документа. Обозначим $\tilde{z}_{tk}$ случайную величину равную единице, если неразмеченный документ $\tilde{\mathbf{x}}_t$ принадлежит кластеру $c_{h,k}$ нижнего уровня, и нулю иначе. Для предсказание класса $\tilde{\mathbf{x}}_t$ с помощью найденной аппроксимации q апостериорного распределения строятся два оператора релевантности.

Оператор R_1 ранжирует кластеры нижнего уровня по значению иерархической функции сходства с параметрами $\boldsymbol{\theta}_k^{\text{MAP}}$ и $\boldsymbol{\alpha}^{\text{MAP}}$, максимизирующими q. Согласно теор. 3, факторы $q(\boldsymbol{\theta}_k)$ и $q(\boldsymbol{\alpha})$ имеют нормальные распределения, поэтому $\boldsymbol{\theta}_k^{\text{MAP}}$ и $\boldsymbol{\alpha}^{\text{MAP}}$ совпадают с их математическим ожиданием.

Оператор R_2 ранжирует кластеры нижнего уровня по вероятности $p(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t)$ принадлежности документа $\tilde{\mathbf{x}}_t$ кластеру $c_{h,k}$ нижнего уровня:

$$p(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t) = \int \operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\tilde{\mathbf{x}}_t|\boldsymbol{\theta}, \boldsymbol{\alpha})\big)_k q(\boldsymbol{\theta}, \boldsymbol{\alpha}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\boldsymbol{\alpha}. \tag{3.73}$$

Интеграл (3.73) не берется аналитически из-за суммы экспонент в знаменателе softmax. Воспользовавшись верхней оценкой (3.52) получаем

$$\begin{aligned}\operatorname{softmax}\big(\mathbf{s}_{\mathsf{h}}(\tilde{\mathbf{x}}_t|\boldsymbol{\theta}, \boldsymbol{\alpha})\big)_k \equiv p(\tilde{z}_{tk}|\boldsymbol{\theta}, \boldsymbol{\alpha}, \tilde{\mathbf{x}}_t) \leq \tilde{p}(\tilde{z}_{tk}|\boldsymbol{\theta}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\xi}}_t, \tilde{\mathbf{x}}_t) \equiv \\ \equiv \frac{1}{g(\tilde{\boldsymbol{\xi}}_t)} \exp\left(s_{t,k} + \sum_{k'=1}^{K_h} \frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)}(\tilde{\xi}_{tk'} - s_{t,k'})\right), \quad s_{t,k} = \tilde{\mathbf{x}}_t^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k. \end{aligned} \tag{3.74}$$

Однако подстановка верхней границы (3.74) в (3.73) все равно не позволяет вычислить интеграл (3.73), так как экспонента в (3.74) неявно содержит произведение параметров $\boldsymbol{\theta}$ и $\boldsymbol{\alpha}$. Воспользуемся методом локальных вариаций для аппроксимации полученной экспоненты в (3.74). Для любой точки x для функции $\exp(x)$ выполняется неравенство

$$\exp(x) \geq \exp(\psi) + \exp(\psi)(x - \psi), \tag{3.75}$$

обращающееся в равенство в точке $\psi = x$. Подставляя (3.75) в (3.74) получаем

$$\tilde{p}(\tilde{z}_{tk}|\boldsymbol{\theta}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\xi}}_t, \tilde{\mathbf{x}}_t) \geq \frac{\exp(\psi_{tk})}{g(\tilde{\boldsymbol{\xi}}_t)} \left(1 - \psi_{tk} + s_{t,k} + \sum_{k'=1}^{K_h} \frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)}(\tilde{\xi}_{tk'} - s_{t,k'})\right). \tag{3.76}$$

Подставляя (3.76) в (3.73) и вычисляя интеграл, получаем оценку вероятности класса

$$\hat{p}(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t, \tilde{\boldsymbol{\xi}}_t, \psi_{tk}) = \frac{\exp(\psi_{tk})}{g(\tilde{\boldsymbol{\xi}}_t)} \left(1 - \psi_{tk} + \tilde{s}_{t,k} + \sum_{k'=1}^{K_h} \frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)} (\tilde{\xi}_{tk'} - \tilde{s}_{t,k'})\right), \quad (3.77)$$

$$\text{где} \quad \tilde{s}_{t,k} = \tilde{\mathbf{x}}_t^{\mathrm{T}} \tilde{\boldsymbol{\Lambda}} \mathbf{M}_k \mathbf{m}'_{0k}.$$

Параметры $\tilde{\boldsymbol{\xi}}_t, \psi_{tk}$ находятся с помощью оптимизации

$$\hat{p}(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t)^* = \max_{\psi_{tk}} \min_{\tilde{\boldsymbol{\xi}}_t} \hat{p}(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t, \tilde{\boldsymbol{\xi}}_t, \psi_{tk}). \quad (3.78)$$

Теорема 5. Значение качества AUCH(R_1) и AUCH(R_2) построенных операторов релевантности при оптимальных значениях вариационных параметров $\tilde{\boldsymbol{\xi}}_t$ и ψ_{tk} совпадает.

Доказательство. Структура выражения (3.77) относительно ψ_{tk} совпадает со структурой (3.75). Так как (3.75) принимает максимальное значение по ψ в точке x, оптимальным значением ψ_{tk} для (3.77) будет

$$\psi_{tk} = \tilde{s}_{t,k} + \sum_{k'=1}^{K_h} \frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)} (\tilde{\xi}_{tk'} - \tilde{s}_{t,k'}).$$

Подставляя его в (3.77), получаем

$$\hat{p}(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t, \tilde{\boldsymbol{\xi}}_t) = \frac{1}{g(\tilde{\boldsymbol{\xi}}_t)} \exp\left(\tilde{s}_{t,k} + \sum_{k'=1}^{K_h} \frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)} (\tilde{\xi}_{tk'} - \tilde{s}_{t,k'})\right). \quad (3.79)$$

В свою очередь из выражения (3.79) можно выделить часть, совпадающую с (3.52), которая принимает минимальное значение в точке

$$\tilde{\xi}_{tk} = \tilde{s}_{t,k}.$$

Подставляя найденные $\tilde{\boldsymbol{\xi}}_t$ в (3.79), получаем выражение

$$\hat{p}(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t) = \frac{\exp(\tilde{s}_{t,k})}{\sum_{k'=1}^{K_h} \exp(\tilde{s}_{t,k'})},$$

совпадающее с softmax от иерархической функции сходства, использующей математическое ожидание параметров $\boldsymbol{\theta}$ и $\boldsymbol{\alpha}$ от найденной аппроксимации апостериорного распределения q. Данные математические ожидания совпадают с $\boldsymbol{\theta}_k^{\mathrm{MAP}}$ и $\boldsymbol{\alpha}^{\mathrm{MAP}}$, поэтому ранжирование в случае R_1 и R_2 дает одинаковый результат. □

Аппроксимация совместного апостериорного распределения. При построении оператора R_2, конечной целью описанного выше вариационного вывода является получение оценки $\hat{p}(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t)$ вероятности принадлежности неразмеченного документа $\tilde{\mathbf{x}}_t$ кластеру $c_{h,k}$. Однако зная оценку апостериорного распределения q, взять интеграл (3.73) аналитически не получается. Рассмотрим альтернативный подход, в котором с помощью вариационного вывода вместо аппроксимации апостериорного распределения параметров ищется аппроксимация совместного апостериорного распределения параметров и классов $\tilde{\mathbf{Z}}$ неразмеченных документов. Данное распределение является оценкой всего подынтегрального выражения (3.73) и позволяет взять интеграл, не используя дополнительных оценок.

Совместное распределение (3.44) и распределение меток неизвестных классов $\tilde{\mathbf{Z}}$ имеет вид

$$p(\tilde{\mathbf{Z}}, \mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) = p(\tilde{\mathbf{Z}}|\boldsymbol{\theta}, \boldsymbol{\alpha})p(\mathbf{Z}, \boldsymbol{\theta}, \boldsymbol{\alpha}, \mathbf{m}, \mathbf{V}). \tag{3.80}$$

В качестве $q(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})$ будем искать аппроксимацию распределения $p(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z})$. Предполагается, что q факторизуется следующим образом:

$$q(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) = q(\boldsymbol{\theta})q(\boldsymbol{\alpha}, \mathbf{m}, \mathbf{V})q(\tilde{\mathbf{Z}}). \tag{3.81}$$

Стоит отметить, что подобное предположение о факторизации не влечет независимость распределения меток классов $\tilde{\mathbf{Z}}$ и параметров $\boldsymbol{\theta}, \boldsymbol{\alpha}$. Наоборот, параметры распределения $q(\tilde{\mathbf{Z}})$ будут выражаться через параметры остальных распределений таким образом, чтобы найденная аппроксимация q была максимально приближена к $p(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z})$. Выписав выражение для нижней границы $\mathcal{L}(q)$ и сгруппировав его аналогично (3.47), получаем вид оптимальных факторов q, максимизирующих $\mathcal{L}(q)$ при фиксированных остальных факторах:

$$\begin{aligned}
\ln q(\boldsymbol{\theta}) &= \mathsf{E}_{\boldsymbol{\alpha},\mathbf{m},\mathbf{V},\tilde{\mathbf{Z}}}\big[\ln p(\tilde{\mathbf{Z}}, \mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})\big] + \text{const}(\boldsymbol{\theta}), \\
\ln q(\boldsymbol{\alpha}, \mathbf{m}, \mathbf{V}) &= \mathsf{E}_{\boldsymbol{\theta},\tilde{\mathbf{Z}}}\big[\ln p(\tilde{\mathbf{Z}}, \mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})\big] + \text{const}(\boldsymbol{\alpha}, \mathbf{m}, \mathbf{V}), \\
\ln q(\tilde{\mathbf{Z}}) &= \mathsf{E}_{\boldsymbol{\alpha},\mathbf{m},\mathbf{V},\boldsymbol{\theta}}\big[\ln p(\tilde{\mathbf{Z}}, \mathbf{Z}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha})\big] + \text{const}(\tilde{\mathbf{Z}}).
\end{aligned} \tag{3.82}$$

Подставляя в выражение (3.80) совместную модель (3.44), априорные распределения параметров (3.55), правдоподобие (3.32) вместо $p(\mathbf{Z}|\boldsymbol{\theta}, \boldsymbol{\alpha})$ и $p(\tilde{\mathbf{Z}}|\boldsymbol{\theta}, \boldsymbol{\alpha})$, воспользовавшись верхней оценкой softmax (3.52) и взяв логарифм

от обоих частей получившегося равенства, получаем

$$\begin{aligned}\ln\hat{p} = &\sum_{t=1}^{T} -\ln g(\tilde{\boldsymbol{\xi}}_t) + \sum_{k=1}^{K_h} \tilde{z}_{tk}\left[\tilde{\mathbf{x}}_t^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k + \left(\sum_{k'=1}^{K_h}\frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)}(\tilde{\xi}_{tk'} - \tilde{\mathbf{x}}_t^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_{k'}\boldsymbol{\theta}_{k'})\right)\right] + \\ &+ \sum_{n=1}^{N} -\ln g(\boldsymbol{\xi}_n) + \sum_{k=1}^{K_h}\left[\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right) + \frac{\xi_{nk}\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right] + \\ &+ \frac{h}{2}\ln a - \frac{h\ln 2\pi}{2} - \frac{a}{2}\boldsymbol{\alpha}^{\mathsf{T}}\boldsymbol{\alpha} + \\ &+ \sum_{k=1}^{K_h}\frac{\ln|\mathbf{V}_k|}{2} - \frac{h}{2}\ln 2\pi - \frac{1}{2}(\boldsymbol{\theta}_k - \mathbf{m}_k)^{\mathsf{T}}\mathbf{V}_k(\boldsymbol{\theta}_k - \mathbf{m}_k) + \\ &+ \frac{\ln|b\mathbf{V}_k|}{2} - \frac{h}{2}\ln 2\pi - \frac{b}{2}(\mathbf{m}_k - \mathbf{m}_0)^{\mathsf{T}}\mathbf{V}_k(\mathbf{m}_k - \mathbf{m}_0) + \\ &+ \ln B(\mathbf{W},\nu) + \frac{(\nu - h - 1)}{2}\ln|\mathbf{V}_k| - \frac{1}{2}\operatorname{Tr}\bigl(\mathbf{W}^{-1}\mathbf{V}_k\bigr),\end{aligned} \tag{3.83}$$

где $B(\mathbf{W},\nu)$ – нормировочный множитель в распределении Уишарта (3.55), индекс $t \in \{1,\ldots,T\}$ соответствует документам $\tilde{\mathbf{x}}_t$ тестовой выборки $D_{\mathcal{T}}$, T – число документов тестовой выборки, а вектор вариационных параметров $\tilde{\boldsymbol{\xi}}_t$ соответствует верхней оценке softmax для документа $\tilde{\mathbf{x}}_t$. Подставляя (3.83) в (3.82) и группируя соответствующие слагаемые, получаем следующие леммы.

Лемма 5. Фактор $q(\boldsymbol{\alpha})$ из (3.82) имеет нормальное распределение.

Доказательство. Оставляя в (3.83) только члены, зависящие от $\boldsymbol{\alpha}$ получаем:

$$\begin{aligned}\ln q(\boldsymbol{\alpha}) = &\sum_{t=1}^{T}\sum_{k=1}^{K_h}\mathsf{E}\tilde{z}_{tk}\tilde{\mathbf{x}}_t^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k - \mathsf{E}\tilde{z}_{tk}\left(\sum_{k'=1}^{K_h}\frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)}\tilde{\mathbf{x}}_t^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_{k'}\mathsf{E}\boldsymbol{\theta}_{k'}\right) + \\ &+ \sum_{n=1}^{N}\sum_{k=1}^{K_h}\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right) - \frac{a}{2}\boldsymbol{\alpha}^{\mathsf{T}}\boldsymbol{\alpha} + a\boldsymbol{\alpha}^{\mathsf{T}}\boldsymbol{\alpha}_0 + \operatorname{const}(\boldsymbol{\alpha}).\end{aligned} \tag{3.84}$$

Преобразуем иерархическую функцию сходства так, чтобы она явным образом зависела от $\boldsymbol{\alpha}$:

$$\begin{aligned}\mathbf{x}_n^{\mathsf{T}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k = \sum_{m=1}^{|W|}x_{nm}(\mathbf{M}_k\theta_k)_m + \boldsymbol{\alpha}^{\mathsf{T}}\sum_{m=1}^{|W|}x_{nm}(\mathbf{M}_k\theta_k)_m\boldsymbol{\iota}_m = \\ = \boldsymbol{\alpha}^{\mathsf{T}}\sum_{m=1}^{|W|}x_{nm}(\mathbf{M}_k\theta_k)_m\boldsymbol{\iota}_m + \operatorname{const}(\boldsymbol{\alpha}).\end{aligned} \tag{3.85}$$

Подставив (3.85) в (3.86) получаем выражение, которое после группировки слагаемых, содержащих $\boldsymbol{\alpha}$, принимает вид квадратичной формы

$$\ln q(\boldsymbol{\alpha}) = -\frac{a}{2}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^{\intercal}\mathbf{I}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + \text{const}(\boldsymbol{\alpha}), \tag{3.86}$$

где $\boldsymbol{\alpha}_0$ задается как

$$\begin{aligned}\boldsymbol{\alpha}_0 = \frac{1}{a}\sum_{m=1}^{|W|}\sum_{t=1}^{T}\tilde{x}_{tm}\boldsymbol{\iota}_m\sum_{k=1}^{K_h}(\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k)_m\left[\mathsf{E}\tilde{z}_{tk} - \frac{\exp(\tilde{\xi}_{tk})}{g(\tilde{\boldsymbol{\xi}}_t)}\left(\sum_{k'=1}^{K_h}\mathsf{E}\tilde{z}_{tk'}\right)\right] + \\ + \frac{1}{a}\sum_{m=1}^{|W|}\sum_{n=1}^{N}x_{nm}\boldsymbol{\iota}_m\sum_{k=1}^{K_h}(\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k)_m\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right).\end{aligned} \tag{3.87}$$

□

Так как при умножении (3.44) на $p(\tilde{\mathbf{Z}}|\boldsymbol{\theta}, \boldsymbol{\alpha})$, слагаемые, зависящие от $\mathbf{m}_k$ и $\mathbf{V}_k$ никак не изменились, то факторы $q(\mathbf{m}_k, \mathbf{V}_k)$ имеют вид, полученный в леммах 1 и 2. Получим выражение для факторов $q(\boldsymbol{\theta}_k)$.

Лемма 6. Фактор $q(\boldsymbol{\theta})$ из (3.82) разбивается на произведение факторов $q(\boldsymbol{\theta}_k)$, каждый из которых имеет нормальное распределение.

Доказательство. Оставляя в (3.83) только члены, зависящие от $\boldsymbol{\theta}$ получаем:

$$\begin{aligned}\ln q(\boldsymbol{\theta}) = &\sum_{t=1}^{T}\sum_{k=1}^{K_h}\mathsf{E}\tilde{z}_{tk}\tilde{\mathbf{x}}_t^{\intercal}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k - \mathsf{E}\tilde{z}_{tk}\left(\sum_{k'=1}^{K_h}\frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)}\tilde{\mathbf{x}}_t^{\intercal}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\mathbf{M}_{k'}\boldsymbol{\theta}_{k'}\right) + \\ &+ \sum_{n=1}^{N}\sum_{k=1}^{K_h}\mathbf{x}_n^{\intercal}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\mathbf{M}_k\boldsymbol{\theta}_k\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right) - \\ &- \frac{1}{2}\sum_{k=1}^{K_h}\mathsf{E}_{\mathbf{m}_k,\mathbf{V}_k}(\boldsymbol{\theta}_k - \mathbf{m}_k)^{\intercal}\mathbf{V}_k(\boldsymbol{\theta}_k - \mathbf{m}_k) + \text{const}(\boldsymbol{\theta}).\end{aligned} \tag{3.88}$$

Подставляя математическое ожидание (3.64) и группируя (3.88) относительно $\boldsymbol{\theta}_k$ получаем, что фактор $q(\boldsymbol{\theta})$ представим в виде произведения нормальных распределений, так как его логарифм разбивается на сумму квадратичных форм относительно $\boldsymbol{\theta}_k$:

$$\ln q(\boldsymbol{\theta}) = \sum_{k=1}^{K_h} -\frac{\nu'}{2}(\boldsymbol{\theta}_k - \mathbf{m}'_{0k})^{\intercal}\mathbf{W}_k(\boldsymbol{\theta}_k - \mathbf{m}'_{0k}) + \text{const}(\boldsymbol{\theta}), \tag{3.89}$$

где параметры $\mathbf{m}_{0k}$ выражаются как

$$\begin{aligned}\mathbf{m}'_{0k} = \mathbf{m}_{0k} + \frac{1}{\nu'}(\mathbf{W}_k^{-1})^{\intercal}\mathbf{M}_k^{\intercal}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\sum_{t=1}^{T}\tilde{\mathbf{x}}_t\left[\mathsf{E}\tilde{z}_{tk} - \frac{\exp(\tilde{\xi}_{tk})}{g(\tilde{\boldsymbol{\xi}}_t)}\left(\sum_{k'=1}^{K_h}\mathsf{E}\tilde{z}_{tk'}\right)\right] + \\ + \frac{1}{\nu'}(\mathbf{W}_k^{-1})^{\intercal}\mathbf{M}_k^{\intercal}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\sum_{n=1}^{N}\mathbf{x}_n\left(z_{nk} - \frac{\exp(\xi_{nk})}{g(\boldsymbol{\xi}_n)}\right).\end{aligned} \tag{3.90}$$

□

Лемма 7. Фактор $q(\tilde{\mathbf{Z}})$ из (3.82) факторизуется на произведение факторов $q(\tilde{z}_{tk})$, каждый из которых имеет вид распределения Бернулли.

Доказательство. Логарифм распределения Бернулли представим в виде

$$\ln p(\tilde{z}_{tk}) = \ln p_{tk}^{\tilde{z}_{tk}}(1-p_{tk})^{1-\tilde{z}_{tk}} = \tilde{z}_{tk}\ln p_{tk} + (1-\tilde{z}_{tk})\ln(1-p_{tk}) = \\ = \tilde{z}_{tk}\ln\left(\frac{1}{1-p_{tk}} - 1\right) + \ln(1-p_{tk}). \quad (3.91)$$

Взяв математическое ожидание (3.83) по всем параметрам кроме $\tilde{z}_{tk}$ и оставив только слагаемые, зависящие от $\tilde{z}_{tk}$, получаем

$$q(\tilde{z}_{tk}) = \tilde{z}_{tk}\Big[-\ln g(\tilde{\boldsymbol{\xi}}_t) + \zeta_{tk}\Big] + \text{const}(\tilde{z}_{tk}) = \tilde{z}_{tk}\ln\left[\frac{1}{g(\tilde{\boldsymbol{\xi}}_t)}\exp(\zeta_{tk})\right] + \text{const}(\tilde{z}_{tk}), \quad (3.92)$$

где для удобства введено обозначение

$$\zeta_{tk} = \tilde{\mathbf{x}}_t^{\mathsf{T}}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\mathbf{M}_k\mathsf{E}\boldsymbol{\theta}_k + \sum_{k'=1}^{K_h}\frac{\exp(\tilde{\xi}_{tk'})}{g(\tilde{\boldsymbol{\xi}}_t)}(\tilde{\xi}_{tk'} - \tilde{\mathbf{x}}_t^{\mathsf{T}}\mathsf{E}_{\boldsymbol{\alpha}}\boldsymbol{\Lambda}\mathbf{M}_{k'}\mathsf{E}\boldsymbol{\theta}_{k'}). \quad (3.93)$$

Параметр распределения p_{tk} является решением уравнения

$$\frac{1}{1-p_{tk}} - 1 = \frac{1}{g(\tilde{\boldsymbol{\xi}}_t)}\exp(\zeta_{tk}) \Rightarrow p_{tk} = \frac{\exp(\zeta_{tk})}{\exp(\zeta_{tk}) + g(\tilde{\boldsymbol{\xi}}_t)}. \quad (3.94)$$

Таким образом, факторы $q(\tilde{z}_{tk})$ имеют распределение Бернулли с параметрами p_{tk}, заданными (3.94). □

Теорема 6. Функцией q из класса (3.81), заданной оптимальными оценками факторов (3.82), в которых правдоподобие L (3.32) оценивается с помощью верхней оценки функции softmax (3.52), а априорные распределения параметров $\boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}$ задаются как (3.43) и (3.42), является

$$q(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}) = q(\boldsymbol{\alpha})\prod_{k=1}^{K_h} q(\boldsymbol{\theta}_k)q(\mathbf{m}_k|\mathbf{V}_k)q(\mathbf{V}_k)\prod_{t=1}^{|T|} q(\tilde{z}_{tk}),$$
$$q(\boldsymbol{\theta}_k) \sim \mathcal{N}(\mathbf{m}'_{0k}, (\nu'\mathbf{V}_k)^{-1}),$$
$$q(\mathbf{m}_k|\mathbf{V}_k) \sim \mathcal{N}(\mathbf{m}_{0k}, (b'\mathbf{V}_k)^{-1}),$$
$$q(\mathbf{V}_k) \sim \mathcal{W}(\mathbf{W}_k, \nu'),$$
$$q(\boldsymbol{\alpha}) \sim \mathcal{N}(\tilde{\boldsymbol{\alpha}}_0, a^{-1}\mathbf{I}),$$
$$q(\tilde{z}_{tk}) \sim \text{Bern}(p_{tk}), \quad (3.95)$$

с параметрами, заданными (3.58), (3.60), (3.87), (3.90), и (3.94) соответственно.

Доказательство. Согласно (3.82), подставляем результаты лемм 1, 2, 5, 6, 7 в (3.81), что дает утверждение теоремы. □

Для настройки параметров данных распределений используется аналогичный EM-алгоритм, как и в случае оценки апостериорного распределения. На E шаг добавляется вычисление математического ожидания $\mathsf{E}\tilde{z}_{tk} = p_{tk}$ и пересчет параметров p_{tk}. Применяя результаты теоремы 4 для вариационных параметров $\tilde{\boldsymbol{\xi}}_t$, получаем их оптимальные значения при фиксированных параметрах q:

$$\tilde{\xi}_{tk} = \tilde{\mathbf{x}}_t^{\mathsf{T}} \tilde{\boldsymbol{\Lambda}} \mathbf{M}_k \mathbf{m}'_{0k}.$$

Прогноз класса нового документа. Используя найденную оценку (3.95) совместного апостериорного распределения параметров и классов неразмеченных документов, получаем искомую вероятность $p(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t)$:

$$p(\tilde{z}_{tk}|\tilde{\mathbf{x}}_t) = \int p(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z}, \tilde{\mathbf{x}}_t) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{m} \mathrm{d}\mathbf{V} \mathrm{d}\boldsymbol{\alpha} \mathrm{d}\tilde{\mathbf{Z}}_{(-tk)} \approx$$

$$\approx \int q(\tilde{\mathbf{Z}}, \boldsymbol{\theta}, \mathbf{m}, \mathbf{V}, \boldsymbol{\alpha}|\mathbf{Z}, \tilde{\mathbf{x}}_t) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\mathbf{m} \mathrm{d}\mathbf{V} \mathrm{d}\boldsymbol{\alpha} \mathrm{d}\tilde{\mathbf{Z}}_{(-tk)} = \mathrm{Bern}(p_{tk}), \quad (3.96)$$

где $\mathrm{d}\tilde{\mathbf{Z}}_{(-tk)}$ означает интегрирование по всем $\tilde{z}$ кроме $\tilde{z}_{tk}$. Задача классификации нового документа записывается как

$$z(\tilde{\mathbf{x}}_t) = \arg\max_k p_{tk},$$

а оператор релевантности R строится путем ранжирования кластеров по вероятности p_{tk}.

Теорема 7. Пусть выполняются следующие соотношения:

$$|D_{\mathcal{V}}| \sim |D_{\mathcal{T}}| \sim |D|, K_h < |D|, \quad h < K_h,$$

Сложность EM-алгоритма настройки параметров распределения q из теоремы 6 равна $O(b|D||W|hK_h)$, где b – число EM шагов.

Доказательство. В данном алгоритме чередуются два шага – оптимизация параметров распределения факторов q и вариационных параметров $\boldsymbol{\xi}$ и $\tilde{\boldsymbol{\xi}}$. На E шаге пересчитываются параметры

$\boldsymbol{\alpha}$ за $O(|W|K_h h + |W||D_{\mathcal{V}}|K_h + |W|) \sim O(|W||D_{\mathcal{V}}|K_h)$,
$\boldsymbol{\theta}_k$ за $O\big(h^2|W| + |W| + (|D_{\mathcal{V}}| + |D_{\mathcal{T}}|)(|W| + K_h)\big) \sim O(|W||D|)$,
$\mathbf{W}_k^{-1}$ за $O(h^2)$,
$\mathbf{m}_{0k}$ за $O(h)$,
p_{tk} за $O(|D_{\mathcal{T}}|W||K_h h)$.

На M шаге ищутся оптимальные вариационные параметры $\boldsymbol{\xi}$ и $\tilde{\boldsymbol{\xi}}$. Данный шаг сводится к вычислению иерархической функции сходства за $O(|W||D|K_h h)$.

С учетом условия теоремы, наиболее трудоемким местом данного алгоритма является вычисление иерархического сходства. Так как EM-алгоритм делает b шагов, то общая сложность

$$O(b|D||W|hK_h). \tag{3.97}$$

□

В поставленных экспериментах, значение $b \sim 10$. Таким образом, асимптотическая сложность данного алгоритма в a^h меньше асимптотической сложности 3.31 алгоритма прямой оптимизации AUCH. При этом обучающая выборка не разбивается на части, и все параметры настраиваются по всей обучающей выборке $D_{\mathcal{V}}$.

3.8. Построение тематической модели конференции

Для анализа предложенных алгоритмов использовалась коллекция D, содержащая 5318 тезисов конференции EURO с экспертной тематической моделью M. Модель M состояла из $k_2 = 24$ кластеров второго уровня Area и $k_3 = 163$ кластера уровня Stream, см. рис. 5.1. Для оптимизации параметров алгоритма использовалась подвыборка $D_{\mathcal{V}}$, содержащая 3655 тезисов. В качестве набора неразмеченных документов $D_{\mathcal{T}}$ использовалась подвыборка размером в 1663 документа. Словарь коллекции W состоял из $|W| = 1675$ слов.

Базовый оператор релевантности. Базовый оператор релевантности $R_1(\mathbf{x})$ строился следующим образом. Все кластеры нижнего уровня сортировались по убыванию их размера в выборке $D_{\mathcal{V}}$. Пусть $c_{3,k_1}, \ldots, c_{3,k_{K_h}}$ – порядок кластеров по убыванию их размера, тогда

$$|c_{3,k_1}| \geq |c_{3,k_2}| \geq \ldots \geq |c_{3,k_{K_h}}|.$$

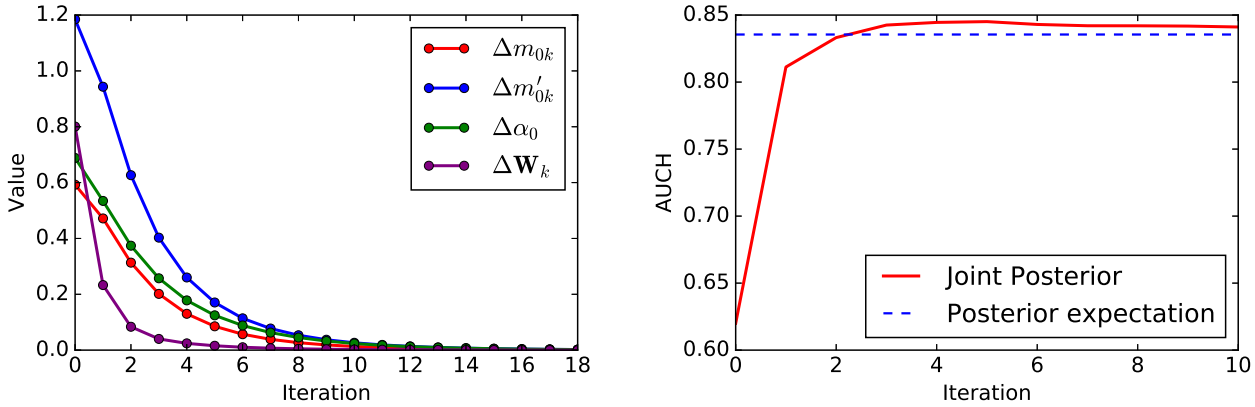
Оператор $R_1(\mathbf{x})$ независимо от документа $\mathbf{x}$ возвращал одну и ту же перестановку номеров кластеров нижнего уровня $(k_1, k_2, \ldots, k_{K_h})$, то есть

$$R_1(\mathbf{x}) = (k_1, k_2, \ldots, k_{K_h}).$$

Оптимизация параметров иерархической функции сходства. Для поиска оптимальных параметров $\boldsymbol{\alpha}$ и $\boldsymbol{\theta}$, использовалась модель (3.80) совместного распределения параметров и классов неразмеченных документов из выборки $D_{\mathcal{T}}$. Параметры распределения q, аппроксимирующего совместное апостериорное распределение параметров модели и классов неразмеченных документов, оценивались согласно результатам теоремы 6 с помощью EM-алгоритма

по выборке $D_{\mathcal{V}}$. Зависимость изменения параметров от итерации показана на рис. 3.7 а. Искомый оператор релевантности R строился путем ранжирования классов по убыванию найденной вероятности (3.96) для каждого документа $\tilde{\mathbf{x}}_t$ из $D_{\mathcal{T}}$.

На рис. 3.7 б. приводится сравнение значения качества AUCH описанного выше оператора релевантности R (красная линяя), со значением качества AUCH оператора релевантности R' построенного с помощью иерархической функции сходства, использующей оценки максимума апостериорной вероятности $\boldsymbol{\theta}_k^{\text{MAP}}$ и $\boldsymbol{\alpha}^{\text{MAP}}$, найденные согласно результатам теоремы 3 (пунктирная синяя линяя). Видно, что при оптимальных значениях параметров, качество оператора релевантности R выше.



а. Сходимость параметров модели.

б. Сравнение качества AUCH операторов релевантности.

Рис. 3.7. Иллюстрация свойств вариационного вывода параметров модели и сравнение способов построения оператора релевантности с помощью оценок апостериорного распределения и совместного апостериорного распределения.

Сравнение базового оператора релевантности с предложенным. Результаты базового операторов релевантности R_1 сравнивались с результатами предложенного оператора R на выборке $D_{\mathcal{T}}$. Из таблицы 3.1 видно, что предложенный оператор релевантности R значительно превосходит базовый оператор R_1. На рис. 3.8 приведены огибающие кумулятивных гистограмм (3.17) операторов R и R_1. На рис. 3.9 а. и б. приведены гистограммы распределения (3.16) для $k \in [1,\ K_3]$, показывающие для столбца с абсциссой k долю документов, у которых экспертный Stream лежит на позиции k в перестановке, возвращаемой оператором релевантности.

На рис. 3.10 показана визуализация матрицы значений парного сходства экспертных кластеров уровня Area. На рис. 3.10 а. приведены результаты для функции сходства, использующей матрицу важности слов $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^*$ с оптимальными значениями параметра $\boldsymbol{\alpha}$, а на рис. 3.10 б. – без оптимизации, $\boldsymbol{\Lambda} = \mathbf{I}$. Из

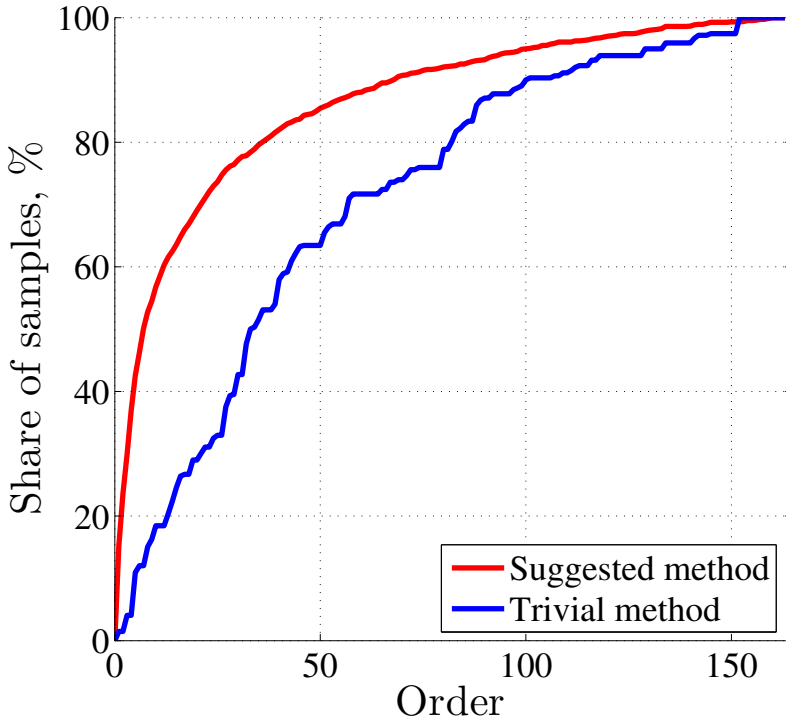
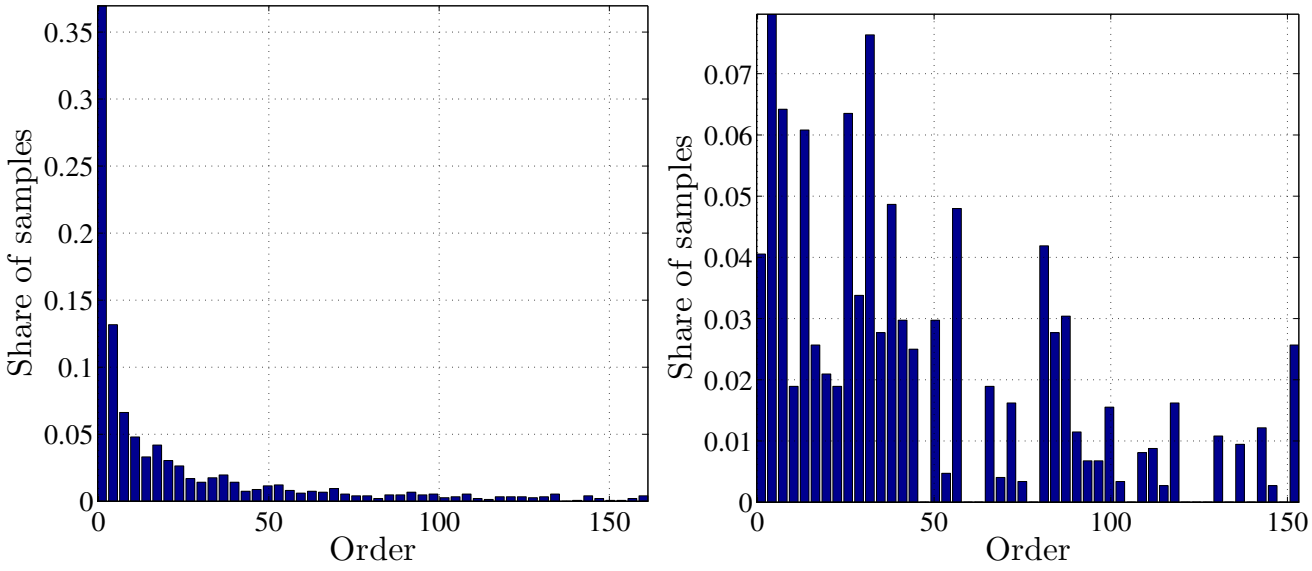


Рис. 3.8. Сравнение операторов релевантности $R(\cdot)$ и $R_1(\cdot)$ по AUCH.

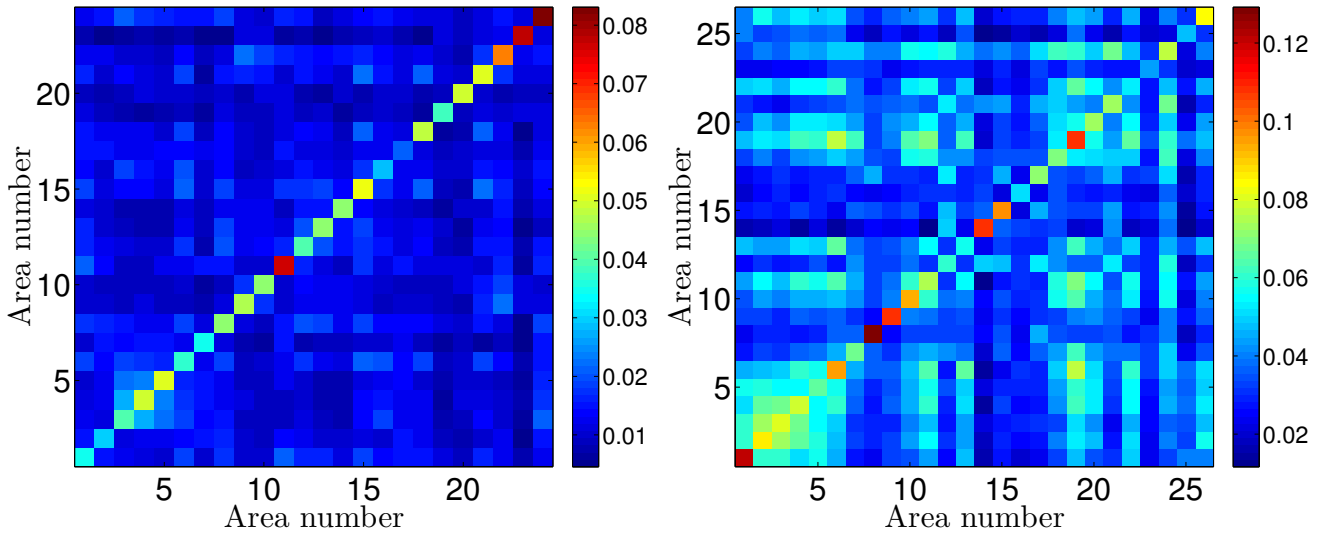


а. Гистограмма распределения документов по позиции их экспертного stream по $R(\cdot)$.

б. Гистограмма распределения документов по позиции их экспертного stream по $R_1(\cdot)$.

Рис. 3.9. Иллюстрация свойств операторов релевантности $R_1(\cdot)$ и $R(\cdot)$.

рисунков видно, что после оптимизации значений $\mathbf{\Lambda}$, внутрикластерные сходства, соответствующие диагональным элементам, стали заметно больше недиагональных элементов, соответствующих значениям межкластерного сходства.



а. Оптимальные веса, $\mathbf{\Lambda} = \mathbf{\Lambda}^*$. б. Единичные веса, $\mathbf{\Lambda} = \mathbf{I}$.

Рис. 3.10. Средние парные сходства экспертных кластеров.

Таблица 3.1. Значения функционалов качества для сравниваемых операторов релевантности Q (3.15) и AUCH (3.17).

Функционал качества \ Оператор релевантности	$R_1(\cdot)$	$R(\cdot)$
AUCH$(\cdot)$	0.72	**0.84**

Так, при $\mathbf{\Lambda}^*$ среднее значение внутрикластерного сходства документов равно 0.047, а межкластерного 0.012.

Глава 4
Верификация тематической модели

Пусть задана экспертная тематическая модель M коллекции документов D и ее критерий качества $\Xi(M)$. Решением задачи верификации является изменение значений классов у фиксированного числа документов в M таким образом, чтобы качество $\Xi(\hat{M})$ полученной тематической модели $\hat{M}$ было максимальным.

4.1. Построение иерархической модели схожей с экспертной

По аналогии с функционалом качества (3.7) для плоского случая, вводится функционал качества иерархической тематической модели как комбинация внутри- и межкластерного сходства.

Определение 20. Качество $\Xi(M)$ иерархической тематической модели M определяется как

$$\Xi(M) = \sum_{l=1}^{h} \left[\frac{1-\beta}{K_l} \sum_{k=1}^{K_l} |c_{l,k}| s_{\mathsf{c}}(c_{l,k},\ c_{l,k}) - \frac{2\beta}{K_l(K_l-1)} \sum_{k=1}^{K_l} \sum_{k'=k+1}^{K_l} s_{\mathsf{c}}(c_{l,k},\ c_{l,k'}) \right]. \tag{4.1}$$

Структурный параметр $\beta \in [0,1]$ – отвечает за приоритет межкластерного сходства. При $\beta \to 0$ качество определяется только внутрикластерным сходством, и наоборот при $\beta \to 1$ качество определяется только межкластерным сходством.

Алгоритм неметрической иерархической кластеризации документов. В качестве начального приближения $\hat{M}$ используется экспертная тематическая модель M. На каждом шаге алгоритма выбирается один документ $\mathbf{x}$ из коллекции D и переносится в другой кластер таким образом, чтобы значение функционала качества Ξ (4.1) максимально возросло. Это повторяется пока кластеризация не стабилизируется в терминах Ξ (4.1). Стоит отметить, что при переносе одного документа изменение параметров модели носит локальный характер, что делает перенос и оценку изменения качества (4.1) вычислительно-эффективной операцией

Утверждение 15. При переносе документа $\mathbf{x} \in c_{h,k}$ в кластер $c_{h,k'}$ новые средние векторы кластеров $c_{l,k}$ и $c_{l,k'}$ определяются по $\mathbf{x}$ и старым средним векторам как

$$\boldsymbol{\mu}(c_{l,k}) \to \frac{|c_{l,k}|}{|c_{l,k}|-1}\boldsymbol{\mu}(c_{l,k}) - \frac{1}{|c_{l,k}|-1}\mathbf{x},$$
$$\boldsymbol{\mu}(c_{l,k'}) \to \frac{|c_{l,k'}|}{|c_{l,k'}|+1}\boldsymbol{\mu}(c_{l,k'}) + \frac{1}{|c_{l,k'}|+1}\mathbf{x}.$$

Доказательство. Пусть $\tilde{c}_{l,k}$ и $\tilde{c}_{l,k'}$ – кластеры после изменения, тогда

$$\boldsymbol{\mu}(\tilde{c}_{l,k}) = \frac{1}{|\tilde{c}_{l,k}|}\sum_{\mathbf{y}\in\tilde{c}_{l,k}}\mathbf{y} = \frac{1}{|c_{l,k}|-1}\left(-\mathbf{x}+\sum_{\mathbf{y}\in c_{l,k}}\mathbf{y}\right) = \frac{|c_{l,k}|}{|c_{l,k}|-1}\boldsymbol{\mu}(c_{l,k}) - \frac{1}{|c_{l,k}|-1}\mathbf{x},$$

$$\boldsymbol{\mu}(\tilde{c}_{l,k'}) = \frac{1}{|\tilde{c}_{l,k'}|}\sum_{\mathbf{y}\in\tilde{c}_{l,k'}}\mathbf{y} = \frac{1}{|c_{l,k'}|+1}\left(\mathbf{x}+\sum_{\mathbf{y}\in c_{l,k'}}\mathbf{y}\right) = \frac{|c_{l,k'}|}{|c_{l,k'}|+1}\boldsymbol{\mu}(c_{l,k'}) + \frac{1}{|c_{l,k'}|+1}\mathbf{x}.$$

□

В силу вложенности тематической модели, в качестве возможных изменений кластеризации документа $\mathbf{x}$ рассматриваются только переносы документов из одного кластера нижнего уровня в другой, так как кластер документа на нижнем уровне однозначно определяет его принадлежность к кластерам более высоких уровней.

Построение модели, схожей с экспертной. Для верификации тематической модели необходимо построить тематическую модель, схожую с экспертной. Для этого алгоритм кластеризации модифицируется следующим образом. Сопоставим каждому документу $\mathbf{x}$ вектор длины $h-1$:

$$\boldsymbol{\zeta}(\mathbf{x}) = \left[[B^{h-2}\big(c(\mathbf{x})\big) = B^{h-2}\big(\hat{c}(\mathbf{x})\big)], \ldots, [B^{0}\big(c(\mathbf{x})\big) = B^{0}\big(\hat{c}(\mathbf{x})\big)]\right].$$

Элемент вектора $[B^{h-l}\big(c(\mathbf{x})\big) = B^{h-l}\big(\hat{c}(\mathbf{x})\big)]$ равняется одному, если на уровне l экспертный кластер $B^{h-l}(c(\mathbf{x}))$ для данного документа $\mathbf{x}$ совпадает с его алгоритмическим кластером $B^{h-l}\big(\hat{c}(\mathbf{x})\big)$. В силу вложенности тематической модели, векторы $\boldsymbol{\zeta}(\mathbf{x})$ могут иметь только следующий вид:

$$\boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), \hat{c}(\mathbf{x})\big) = [1,\ldots 1,\, 0,\ldots,0],$$

где первые m единиц означают, что для первых m уровней экспертная кластеризация и алгоритмическая кластеризация совпали, а последующие $h-m-1$ нулей означают, что для оставшихся $h-m-1$ уровней алгоритмическая кластеризация для данного документа отличается от экспертной. Всего возможно h вариантов вектора $\boldsymbol{\zeta}(\mathbf{x})$. Сумма элементов $\|\boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), \hat{c}(\mathbf{x})\big)\|_1$ показывает, на скольких уровнях совпадает экспертная кластеризация с алгоритмической для документа $\mathbf{x}$.

Каждой операции переноса документа $\mathbf{x}$ из кластера $c_{h,k}$ в кластер $c_{h,k'}$ ставится в соответствие пара векторов $\boldsymbol{\zeta}$ вида

$$\boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), c_{h,k}\big) \mapsto \boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), c_{h,k'}\big).$$

Каждому уникальному варианту переноса ставится в соответствие штраф δ за его осуществление. Всего возможно h^2 различных штрафов. Для их определе-

ния используется таблица размером $h \times h$. Пример таблицы штрафов для случая $h = 3$ показан в таблице. 4.1. Предполагается, что $\delta_{11} = \cdots = \delta_{hh} = 0$, так как переносы документа $\mathbf{x}$ такого вида не добавляют отличий в кластеризации.

Обозначим

$$\boldsymbol{\zeta} = \boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), c_{h,k}\big), \quad \boldsymbol{\zeta}' = \boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), c_{h,k'}\big), \quad \boldsymbol{\zeta}'' = \boldsymbol{\zeta}\big(\mathbf{x}, c(\mathbf{x}), c_{h,k''}\big),$$

Ξ_1 – значение оптимизируемой функции Ξ (3.1) до переноса $\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}'$ документа $\mathbf{x}$, и Ξ_2 – ее значение после переноса. Перенос $\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}'$ документа $\mathbf{x}$ осуществляется только при выполнении условия:

$$\Xi_2 - \Xi_1 \geq \gamma\delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}'), \tag{4.2}$$

где $\delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}')$ – штраф, соответствующий переносу $\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}'$, а $\gamma \geq 0$ – весовой множитель штрафов, регулирующий допустимую степень несоответствия построенной кластеризации и экспертной.

Таблица 4.1. Матрица штрафа $\mathbf{F}$.

Из \ В	$(1,1)$	$(1,0)$	$(0,0)$
$(1,1)$	δ_{11}	δ_{12}	δ_{13}
$(1,0)$	δ_{21}	δ_{22}	δ_{23}
$(0,0)$	δ_{31}	δ_{32}	δ_{33}

Таблица 4.2. Матрица штрафа $\tilde{\mathbf{F}}$.

Из \ В	$(1,1)$	$(1,0)$	$(0,0)$
$(1,1)$	0	0.002	0.005
$(1,0)$	-0.001	0	0.003
$(0,0)$	-0.003	-0.002	0

Различные штрафы позволяют учитывать существующую экспертную модель с различным весом. Если требуется выявить небольшое число наиболее сильных тематических противоречий, то штрафы на перемещение документа из его экспертного кластера задаются большие. Если целью является построить модель, не основываясь на экспертной, то штрафы следует устремить к нулю. Элементы матрицы штрафа должны удовлетворять следующим условиям.

1. Чем больше создается различий с экспертной моделью в результате перемещения документа, тем больше величина штрафа за этот перенос:
$$\|\boldsymbol{\zeta}'\|_1 < \|\boldsymbol{\zeta}''\|_1 \Rightarrow \delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}') > \delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}'').$$
2. Для любых двух последовательных переносов, каждый из которых увеличивает число различий, должно выполняться свойство транзитивности:
$$\|\boldsymbol{\zeta}\|_1 > \|\boldsymbol{\zeta}'\|_1, \quad \|\boldsymbol{\zeta}'\|_1 > \|\boldsymbol{\zeta}''\|_1 \Rightarrow \delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}') + \delta(\boldsymbol{\zeta}' \mapsto \boldsymbol{\zeta}'') = \delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}'').$$
3. Штраф за перенос, уменьшающий число различий, отрицательный.
$$\|\boldsymbol{\zeta}\|_1 < \|\boldsymbol{\zeta}'\|_1 \Rightarrow \delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}') < 0.$$
4. Сумма порогов для последовательности переносов, возвращающих документ в исходный кластер, больше нуля:
$$\|\boldsymbol{\zeta}\|_1 - \|\boldsymbol{\zeta}''\|_1 = 0 \Rightarrow \delta(\boldsymbol{\zeta} \mapsto \boldsymbol{\zeta}') + \delta(\boldsymbol{\zeta}' \mapsto \boldsymbol{\zeta}'') > 0.$$

4.2. Верификация тематической модели конференции

Для анализа работы предложенных алгоритмов проводилась верификация тематической модели конференции EURO. В качестве исходных данных был взят набор из 2313 тезисов конференции и ее экспертная тематическая модель, состоявшая из трех уровней $h = 3$, как показано на рис. 5.1.

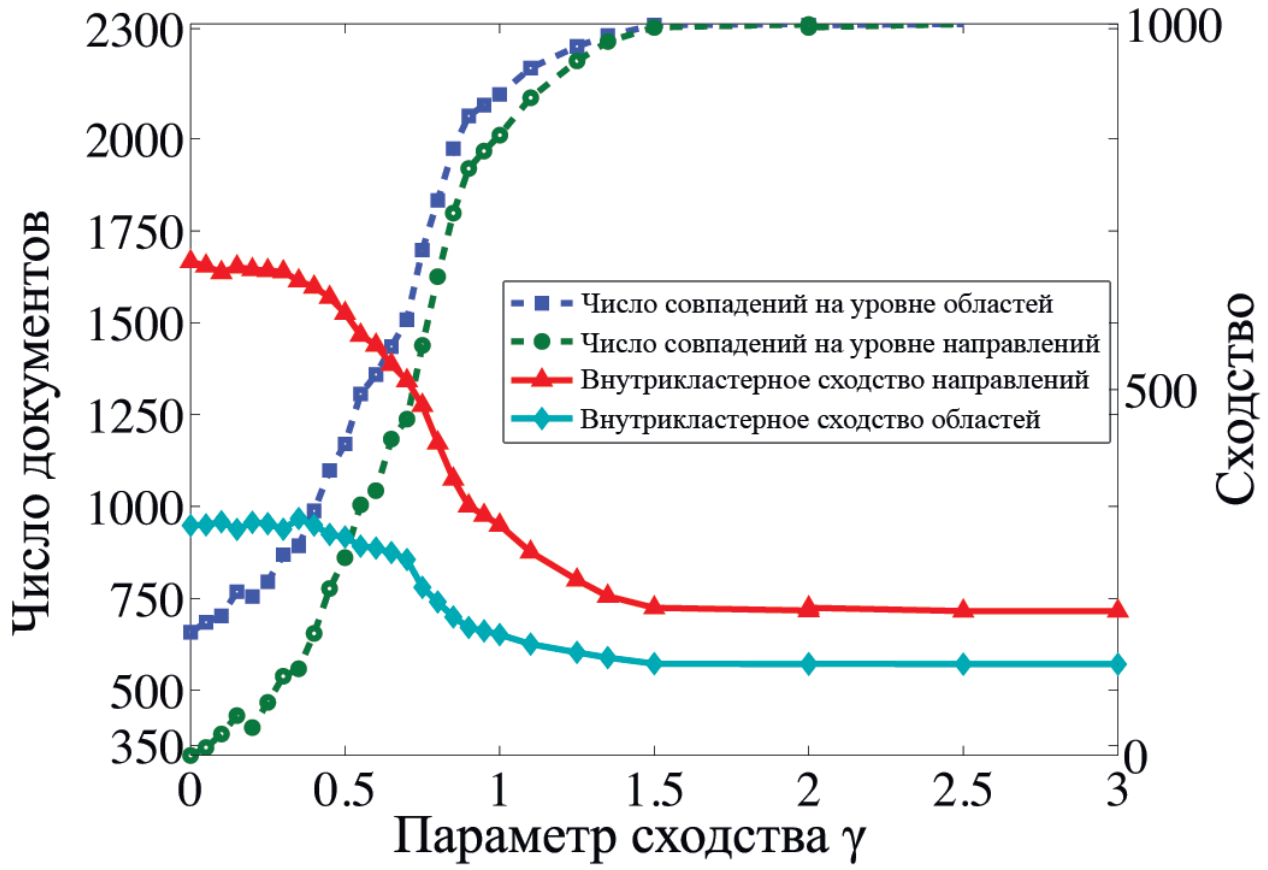
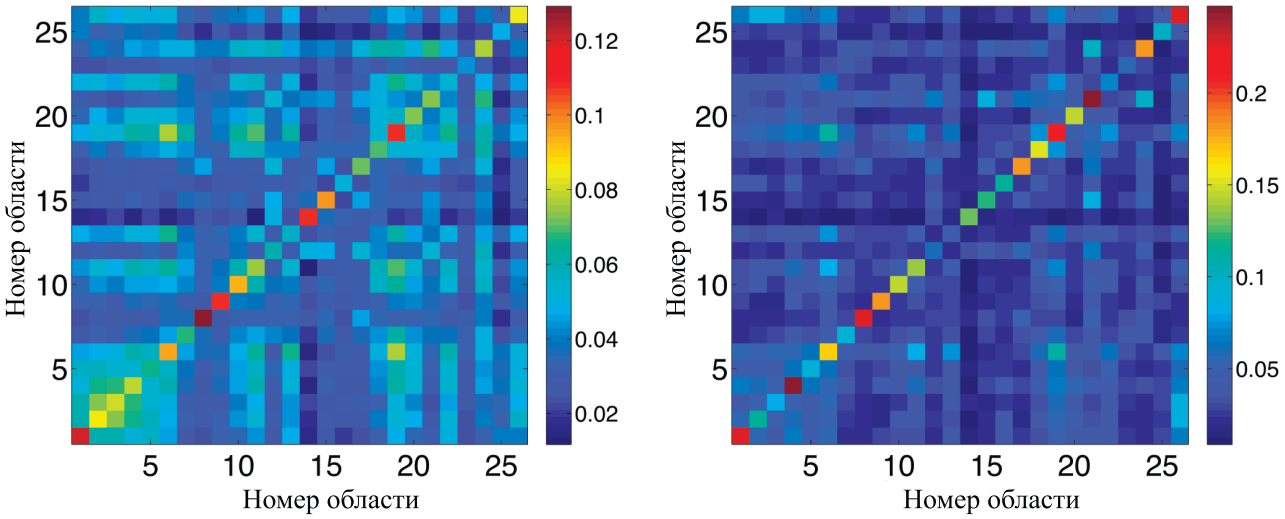


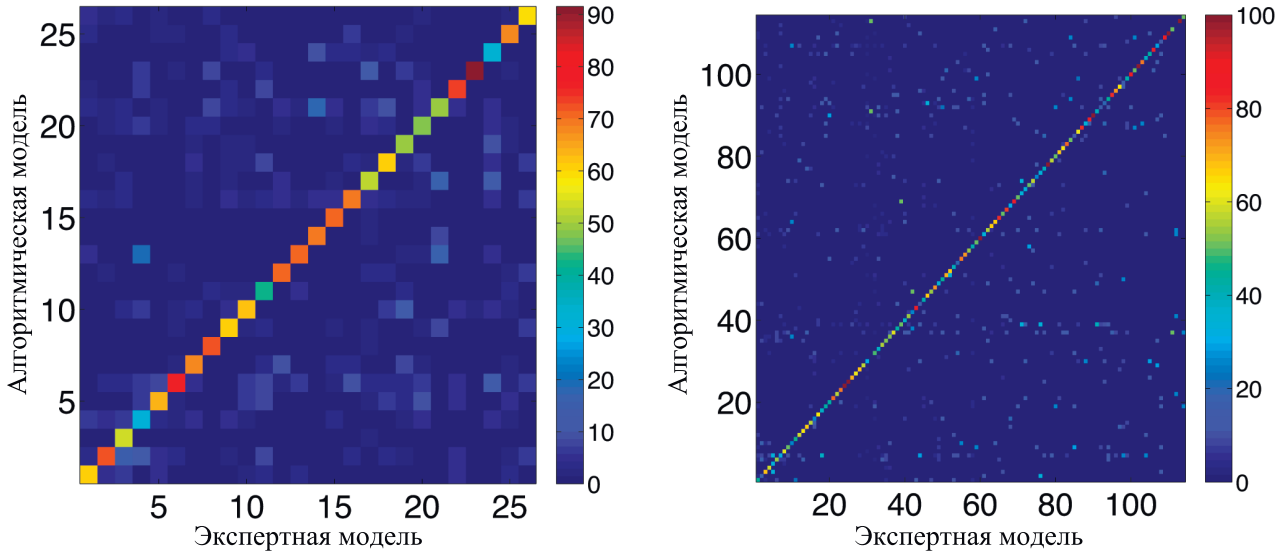
Рис. 4.1. Зависимость внутри- и межкластерного сходства на уровнях областей и направлений от параметра γ.

С помощью алгоритма, описанного в разделе 4.1. строилась алгоритмическая тематическая модель с параметром оптимизируемой функции Ξ (4.1) $\beta = 0.1$. В таблице 4.2 приведена матрица штрафов, использованная для построения модели. При изменении параметра γ в условии для переноса (4.2), изменялось сходство алгоритмической и экспертной модели. Результаты кластеризации, соответствующие разным значений параметра γ приведены на рис. 4.1. По левой оси отложено количество документов, для которых экспертная и алгоритмическая кластеризации совпали, по правой оси значения среднего внутрикластерного сходства (3.3), а по нижней оси отложено соответствующее значение параметра γ. Чем больше задавался штраф γ, тем меньше документов попадали в чужие кластеры, но и внутрикластерное сходство становилось меньше. Так при значении $\gamma > 2$, 99% документов попали в свои экспертные кластеры.



а. Экспертная кластеризация. б. Построенная кластеризация.

Рис. 4.2. Сравнение среднего сходства по областям.



а. Распределение по областям. б. Распределение по направлениям.

Рис. 4.3. Процентное распределение документов по областям и направлениям.

На рис. 4.2 а приведены результаты визуализации матрицы парного сходства кластеров уровня Area до процесса верификации, а на рис. 4.2 б – после. По осям отложены номера областей. Цвет клетки (x, y) соответствует значению сходства (3.3) между кластером Area с номером x и кластером Area с номером y. Клетки диагонали (x, x) соответствуют внутрикластерному сходству, а клетки (x, y), $x \neq y$ — межкластерному.

На рис. 4.3 клетка с координатами (x, y) показывает количество докумен-

тов, которые эксперт отнес к кластеру с номером x, а алгоритм к кластеру с номером y. На диагонали находятся документы, для которых экспертная кластеризация совпала с алгоритмической.

Глава 5
Анализ прикладных задач

Глава содержит анализ свойств предложенных моделей и рекомендации по их использованию. Качество предложенных моделей сравнивается с качеством известных решений.

5.1. Иерархическая классификация тезисов крупной конференции

Ежегодно программный комитет крупной конференции решает задачу построения иерархической модели тезисов конференции. Рассмотрим такую модель на примере конференции European Conference on Operational Research (EURO). Конференция содержит в себе 26 главных областей. Каждая область содержит в себе $10-15$ научных направлений, каждое направление делится на $5-10$ сессий, а каждая сессия состоит из четырех докладов. Для подачи заявки участники конференции присылают программному комитету тезисы их докладов –документы состоящие из не более чем 600 символов, и выбирают три ключевых слова, наиболее связанных по их мнению с тематикой работы. Все участники делятся на две группы – вновь поступившие и приглашенные. У приглашенных участников заранее известна сессия, в которой они будут выступать. Для остальных участников эксперты из программного комитета должны выбрать наиболее подходящую сессию на основании содержания полученных документов и выбранных ключевых слов. Данная конференция представлена в виде дерева на рис. 5.1. Для построения данной иерархической тематической модели привлекается до 200 экспертов из различных областей.

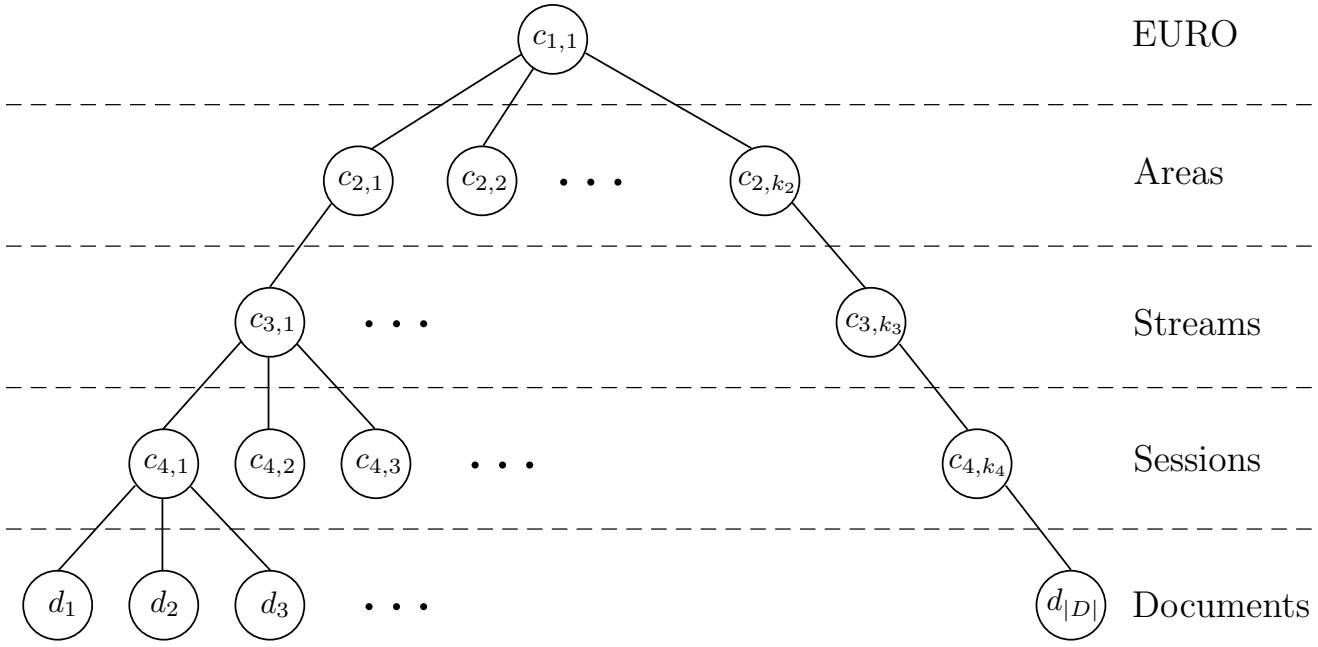


Рис. 5.1. Иерархическая структура конференции в виде дерева.

Из года в год кластерная структура уровня Area и кластерная структура уровня Stream изменяются незначительно. Для исследования свойств предложенных моделей и методов, описанных в главе 3, была построена экспертная система, позволяющая искать релевантные кластеры для неразмеченных документов, используя экспертные тематические модели конференций прошлых лет.

Предобработка текстовой коллекции. Для создания коллекции документов использовались программы конференций EURO и IFORS за период с 2006 по 2016 год [119]. Для задачи классификации рассматривались только уровни Area и Stream иерархической модели. Процедура получения экспертных тематических моделей и предобработанных текстов тезисов показана на рис. 5.2. На вход подавались тексты тезисов в формате pdf, пример тезиса представлен на рис. 5.3 а. С помощью программы pdftotext данные файлы переформатировались в простой текст EURO.txt, рис. 5.3 б. С помощью программы-парсера из EURO.txt выделялись названия Area, Stream и текст тезиса. Они записывались в новый файл через разделитель "##", как показано на рис. 5.3 в. Для нормализации слов в полученном структурированном файле использовался пакет NLTK [120]. Из текста отбрасывались стоп-слова, а остальные слова приводились к нормальной форме с помощью лемматизатора, ипользующего семантическую сеть wordNet [121] для английского языка, после чего использовался стеммер Snowball [122] и отбрасывались все слова, для которых не было векторного представления в обученной модели word2vec [42]. На выходе получался документ, показанный на рис. 5.3 г.

Описание коллекции. Все документы объединялись в одну общую коллекцию. Чтобы уменьшить различие тематических моделей конференций разных лет и построить общую экспертную тематическую модель, была проведена следующая процедура:

1) модель конференции EURO 2016 была взята за основу общей модели,
2) для каждого кластера уровня Stream и Area конференций 2010-2015 года эксперты либо находили соответствующий ему кластер в общей модели, либо добавляли новый кластер в общую модель,
3) для конференций 2006-2010 года автоматически искался соответствующий кластер в общей модели с учетом всевозможных перестановок слов в названии кластера без учета регистра, знаков препинания и артиклей. Если соответствующий кластер не был найден, то документы из данного кластера не добавлялись в модель.

В результате была получена предобработанная коллекция с экспертной иерархической тематической моделью:

1) размер коллекции $|D| = 15527$ документов,
2) размер словаря $|W| = 24304$ слова,
3) число кластеров второго уровня (Area) $K_2 = 26$,

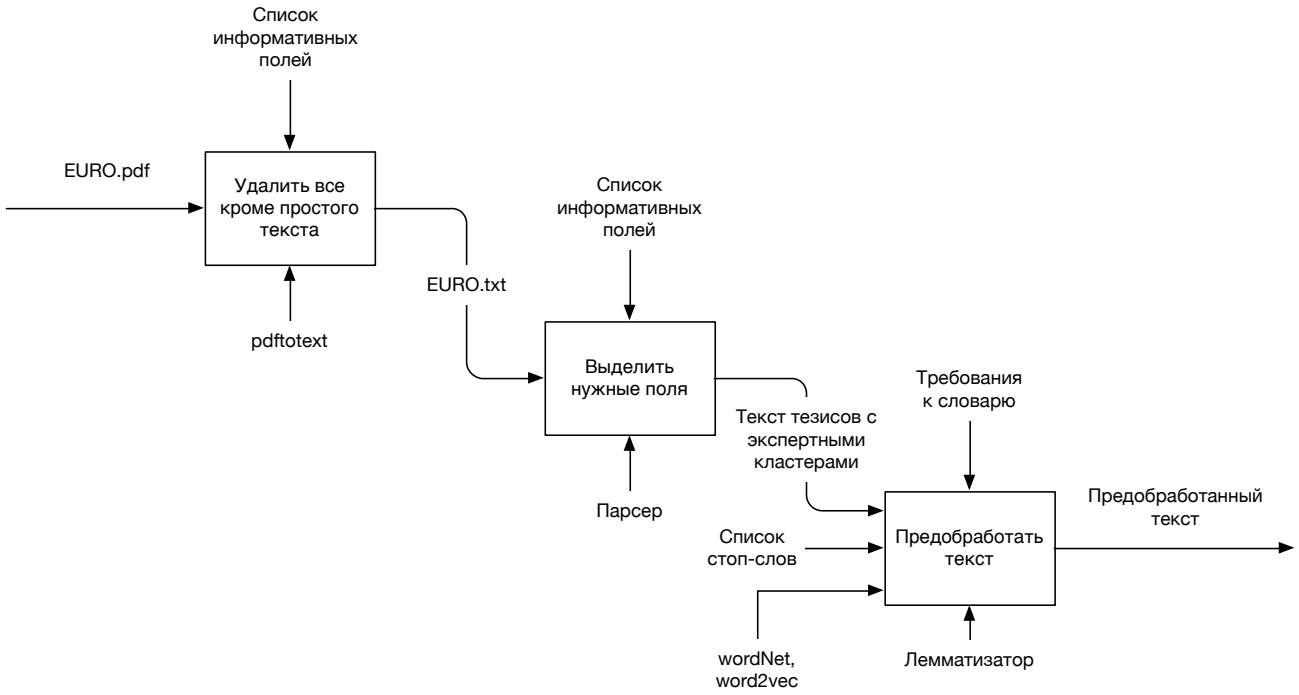


Рис. 5.2. Процесс предобработки программ конференций EURO.

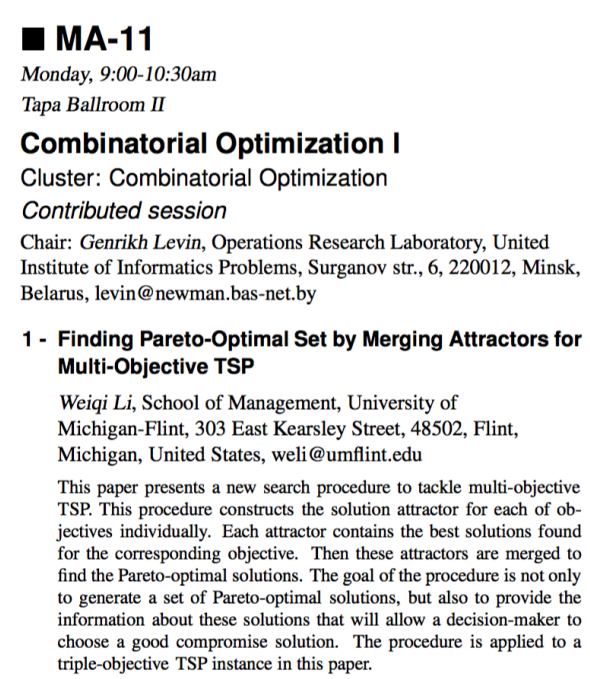
4) число кластеров третьего уровня (Stream) $K_3 = 264$.

Результаты классификации неразмеченных тезисов конференции. Для проверки предложенных методов – иерархической взвешенной функции сходства hSim (3.12) и ее аналогу, построенному с помощью обученной языковой модели и векторного представления слов hSimWV (3.24), их результаты сравнивались с результатами других алгоритмов, описанных в разделе 1.9.: иерархическим наивным байесом hNB, вероятностной моделью SuhiPLSA и иерархическим мультиклассовым svm.

Коллекция документов D делилась на две части: обучающую $D_{\mathcal{V}}$ и тестовую $D_{\mathcal{T}}$. Для анализа работы алгоритмов при различном объеме данных для обучения, размер обучающей выборки $|D_{\mathcal{V}}|$ менялся от 500 документов до 10000. Из тестовой выборки $D_{\mathcal{T}}$ случайным образом выбиралась подвыборка $D_{\mathcal{T}'}$, в которой число документов было фиксированным, $|D_{\mathcal{T}'}| = 5000$, вне зависимости от размера обучающей выборки.

По обучающей выборке $D_{\mathcal{V}}$ с помощью каждого из алгоритмов строился оператор релевантности R (3.14), возвращающий ранжированный список кластеров нижнего уровня в порядке убывания их релевантности новому документу. Качество построенных операторов оценивалось на тестовой выборке $D_{\mathcal{T}'}$ как площадь под кумулятивной гистограммой AUCH (3.17).

Рассматривались два случая – плоской классификации для одного уровня Area, и иерархической классификации для модели с уровнями Area и Stream. В таблицах 5.1 и 5.2 приведены значения AUCH для операторов релевантности,



■ **MA-11**

Monday, 9:00-10:30am

Tapa Ballroom II

Combinatorial Optimization I

Cluster: Combinatorial Optimization

Contributed session

Chair: *Genrikh Levin*, Operations Research Laboratory, United Institute of Informatics Problems, Surganov str., 6, 220012, Minsk, Belarus, levin@newman.bas-net.by

1 - Finding Pareto-Optimal Set by Merging Attractors for Multi-Objective TSP

Weiqi Li, School of Management, University of Michigan-Flint, 303 East Kearsley Street, 48502, Flint, Michigan, United States, weli@umflint.edu

This paper presents a new search procedure to tackle multi-objective TSP. This procedure constructs the solution attractor for each of objectives individually. Each attractor contains the best solutions found for the corresponding objective. Then these attractors are merged to find the Pareto-optimal solutions. The goal of the procedure is not only to generate a set of Pareto-optimal solutions, but also to provide the information about these solutions that will allow a decision-maker to choose a good compromise solution. The procedure is applied to a triple-objective TSP instance in this paper.

а. Pdf файл с тезисом программы конференции EURO.

```
MA-11
Monday, 9:00-10:30am
Tapa Ballroom II
Combinatorial Optimization I
Cluster: Combinatorial Optimization
Contributed session
Chair: Genrikh Levin, Operations Research Laboratory,
United
Institute of Informatics Problems, Surganov str., 6,
220012, Minsk,
Belarus, levin@newman.bas-net.by
1 - Finding Pareto-Optimal Set by Merging Attractors
for
Multi-Objective TSP
Weiqi Li, School of Management, University of
Michigan-Flint, 303 East Kearsley Street, 48502,
Flint,
Michigan, United States, weli@umflint.edu
This paper presents a new search procedure to tackle
multi-objective
TSP. This procedure constructs the solution attractor
for each of ob-
jectives individually. Each attractor contains the
best solutions found
for the corresponding objective. Then these attractors
are merged to
find the Pareto-optimal solutions. The goal of the
procedure is not only
to generate a set of Pareto-optimal solutions, but
also to provide the
information about these solutions that will allow a
decision-maker to
choose a good compromise solution. The procedure is
applied to a
triple-objective TSP instance in this paper.
```

б. Текстовое представление тезиса из программы конференции.

```
discrete optimization mixed integer linear nonlinear
programming
##
combinatorial optimization
##
finding pareto optimal set by merging attractors for
multi objective tsp weiqi li school of management
university of michigan flint east kearsley street
flint michigan united states weli umflint edu this
paper presents a new search procedure to tackle multi
objective tsp this procedure constructs the solution
attractor for each of objectives individually each
attractor contains the best solutions found for the
corresponding objective then these attractors are
merged to find the pareto optimal solutions the goal
of the procedure is not only to generate a set of
pareto optimal solutions but also to provide the
information about these solutions that will allow a
decision maker to choose a good compromise solution
the procedure is applied to a triple objective tsp
instance in this paper
```

в. Area, Stream и текст тезиса конференции в структурированном виде.

```
discrete optimization mixed integer linear nonlinear
programming
##
combinatorial optimization
##
find pareto optim set merg attractor multi object tsp
weiqi li school manag univers michigan flint east
kearsley street flint michigan unit state weli umflint
edu paper present new search procedur tackl multi
object tsp procedur construct solut attractor object
individu attractor contain best solut find correspond
object attractor merg find pareto optim solut goal
procedur onli generat set pareto optim solut also
provid inform solut allow decis maker choos good
compromis solut procedur appli tripl object tsp
instanc paper
```

г. Нормализованный текст тезиса конференции.

Рис. 5.3. Предобработка коллекции тезисов EURO.

построенных с помощью заданных алгоритмов для различного числа документов в обучающей выборке для плоского и иерархического случая соответственно. Для плоского случая сравнивались алгоритмы svm, hNB, suhiPLSA и hSim, а для иерархического случая svm, hNB, suhiPLSA, hSim и hSimWV. Жирным шрифтом выделены лучшие статистически эквивалентные результаты для каж-

дого размера выборки. На рис. 5.4 данные из таблиц показаны в виде графиков зависимостей значения AUCH по оси ординат от размера выборки по оси абсцисс.

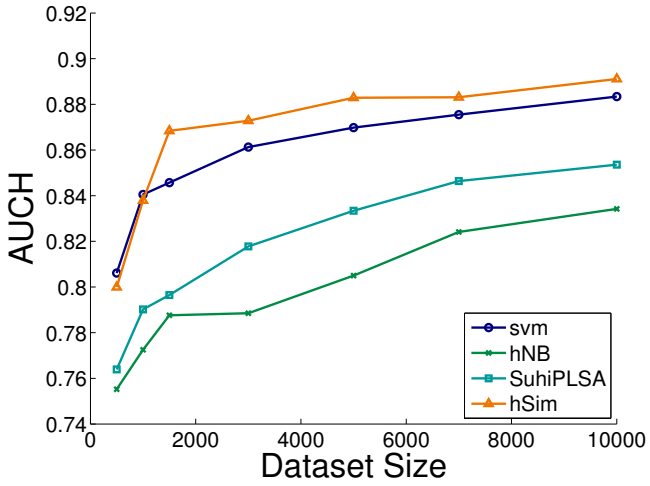
Таблица 5.1. Значения функционала качества AUCH (3.17) на уровне Area для операторов релевантности, построенных с помощью сравниваемых алгоритмов.

Размер выборки $\lvert D_{\mathcal{V}}\rvert$ / Алгоритм	500	1000	1500	3000	5000	7000	10000
svm	**0.81**	**0.84**	0.85	0.86	0.87	0.88	0.88
hNB	0.76	0.77	0.79	0.79	0.81	0.82	0.83
suhiPLSA	0.76	0.79	0.80	0.82	0.83	0.85	0.85
hSim	0.80	**0.84**	**0.87**	**0.87**	**0.88**	**0.88**	**0.89**

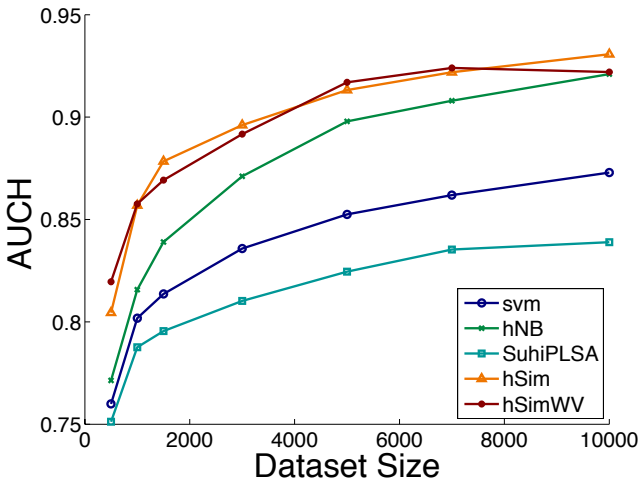
Таблица 5.2. Значения функционала качества AUCH (3.17) на уровне Stream для операторов релевантности, построенных с помощью сравниваемых алгоритмов.

Размер выборки $\lvert D_{\mathcal{V}}\rvert$ / Алгоритм	500	1000	1500	3000	5000	7000	10000
svm	0.76	0.80	0.81	0.84	0.85	0.86	0.87
hNB	0.77	0.82	0.84	0.87	0.90	0.91	0.92
suhiPLSA	0.75	0.79	0.80	0.81	0.82	0.84	0.84
hSim	0.80	**0.86**	**0.88**	**0.90**	**0.91**	**0.92**	**0.93**
hSimWV	**0.82**	**0.86**	0.87	**0.89**	**0.92**	**0.92**	0.92

Из таблицы 5.1 видно, что для плоской классификации на уровне Area данной иерархической модели при размере выборки меньше 1000 документов svm показывает более высокий результат, однако при увеличении числа документов в обучении иерархическая взвешенная функция сходства hSim показывает значительно более высокий результат. Алгоритм, основанный на наивном байесовском предположении, имеет наименьший показатель качества при любом размере выборки.

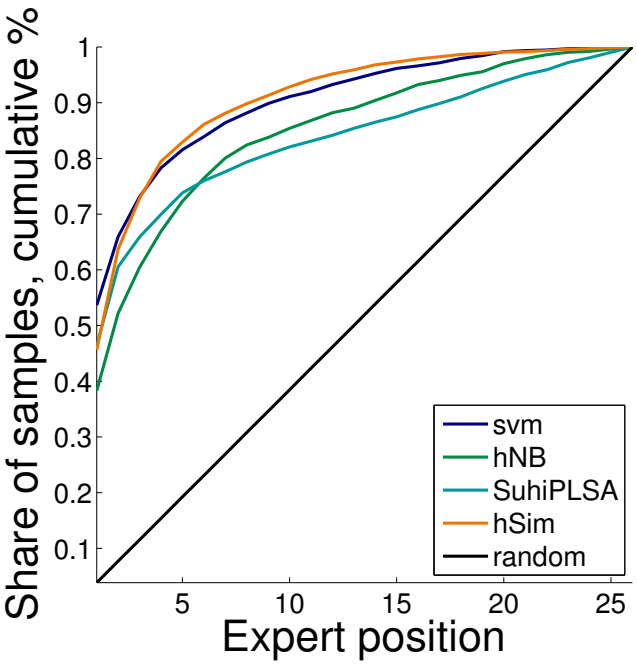


а. Случай плоской классификации.

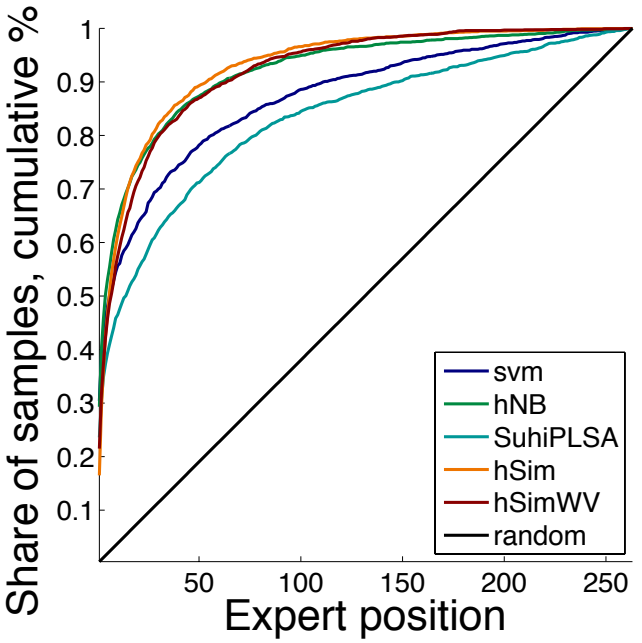


б. Случай иерархической классификации.

Рис. 5.4. Зависимость значений AUCH от размера обучающей выборки для операторов релевантности $R(\cdot)$, построенных с помощью алгоритмов svm, hNB, suhiPLSA, hSim и hSimWV.



а. Случай плоской классификации.



б. Случай иерархической классификации.

Рис. 5.5. Огибающие кумулятивных гистограмм операторов релевантности $R(\cdot)$, построенных с помощью алгоритмов svm, hNB, suhiPLSA, hSim, hSimWV.

Для случая иерархической классификации при размере выборки меньше 1000 документов наилучшее качество показывает алгоритм hSimWV, но

при дальнейшем увеличении количества документов в обучающей выборке алгоритм hSim показывает либо превосходящие, либо эквивалентные результаты по критерию AUCH (3.17), см. таблицу 5.2. Стоит отметить, что в отличие от плоского случая, для иерархической классификации алгоритм hNB показывает значительно лучшие результаты чем иерархическая версия svm и вероятностная модель suhiPLSA.

На рис. 5.5 показаны огибающие кумулятивных гистограмм (3.16) для обучающей выборки размером $D_{\mathcal{V}} = 10000$ документов для плоской и иерархической классификации. Точка на кривой с координатами (x, y) означает, что для данного алгоритма, у доли документов x экспертный кластер лежит на первых y позициях в ранжированных списках, полученных с помощью оператора релевантности (3.14). Из рис. 5.5 видно, что hSim для данного размера обучающей выборки показывает наилучший результат.

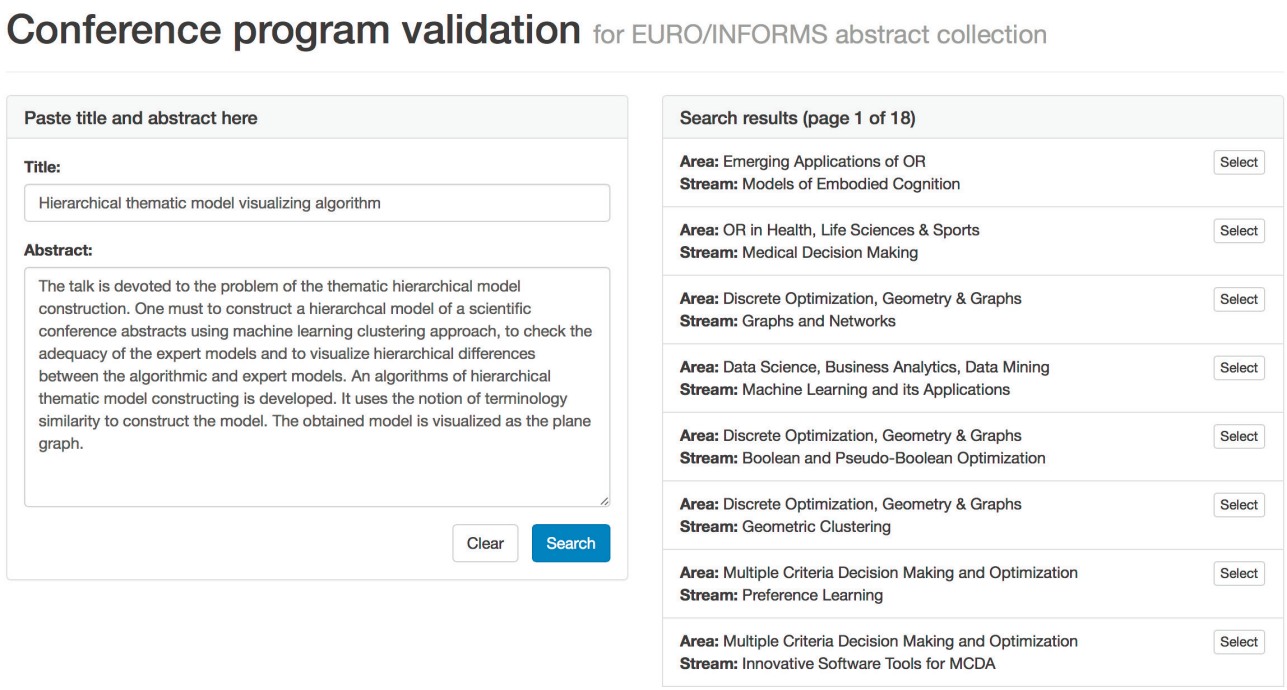


Рис. 5.6. Экспертная система для поиска релевантных кластеров для неразмеченных документов.

Экспертная система. Для программного комитета конференции EURO была реализована экспертная система, изображенная на рис. 5.6. В поля “Title” и “Abstract” вставлялись название и аннотация к неразмеченному документу. После нажатия на кнопку “Search” введенный текст предобрабатывался способом, описанным в разделе 5.1. Документу ставилось в соответствие его векторное описание $\mathbf{x}$. Оператор релевантности R (3.14), построенный с помощью предложенного алгоритма hSim, ставил в соответствие $\mathbf{x}$ ранжированный список кластеров общей экспертной тематической модели, который отображался в поле “Search results”. Эксперт выбирал из этого списка подходящий кластер

для данного документа, после чего документ добавлялся в обучающую подвыборку $D_{\mathcal{V}}$ и оценки параметров модели пересчитывались согласно алгоритму, описанному в разделе 3.5.

5.2. Визуализация иерархической тематической модели на плоскости

В данном разделе описывается алгоритм визуализации имеющейся экспертной иерархической модели на плоскости, обладающей свойством вложенности. Предлагается способ визуализации тематических несоответствий, выявленных методами, описанными в разделе 4.1.

Определение 21. Визуализацию иерархической модели M на плоскости будем называть вложенной, если границы кластеров не пересекаются, а дочерние кластеры лежат внутри родительских. Пример вложенной визуализации изображен на рис. 5.7.

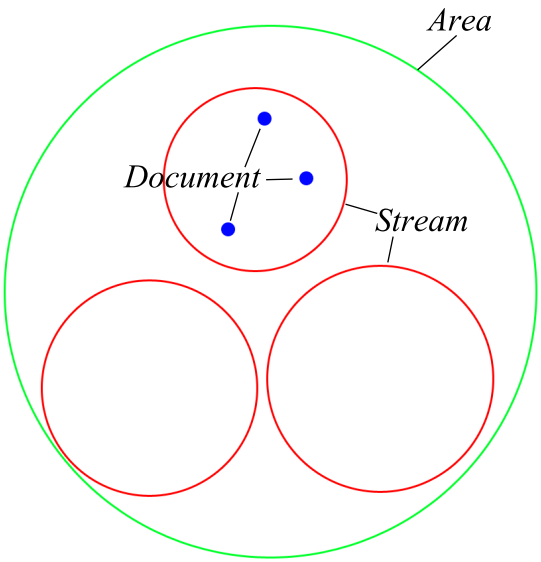


Рис. 5.7. Вложенная визуализация иерархической модели.

Построение вложенной визуализации. Пусть $\boldsymbol{\mu}(c_{l,1}), \ldots, \boldsymbol{\mu}(c_{l,K_l})$ – координаты центров кластеров на l-м уровне иерархии в исходном пространстве, а $\hat{\boldsymbol{\mu}}(c_{l,1}), \ldots, \hat{\boldsymbol{\mu}}(c_{l,K_l})$ – их координаты на плоскости. Пусть $r(c_{l,1}), \ldots, r(c_{l,K_l})$ – радиусы кластеров, понимаемые как расстояния от центра кластера до самого далекого документа из этого кластера в исходном пространстве, а $\hat{r}(c_{l,1}), \ldots, \hat{r}(c_{l,K_l})$ – их радиусы на плоскости. В качестве расстояния в исходном пространстве используется евклидова метрика $\rho(\cdot, \cdot)$.

Начиная с верхнего уровня иерархии выполняется проекция центров кластеров на плоскость с помощью проекции Саммона [34] и вычисляются двумерные координаты проекций центров кластеров $\hat{\boldsymbol{\mu}}(c_{l,1}),\ \ldots,\ \hat{\boldsymbol{\mu}}(c_{l,K_l})$. Радиус кластера на плоскости определяется как

$$\hat{r}(c_{l,k}) = \min_{k' \neq k\,:\, c_{l,k'} \in B(c_{l,k})} \frac{r(c_{l,k})}{r(c_{l,k}) + r(c_{l,k'})} \rho\big(\hat{\boldsymbol{\mu}}(c_{l,k}),\ \hat{\boldsymbol{\mu}}(c_{l,k'})\big), \tag{5.1}$$

где $B(c_{l,k})$ – родительский кластер кластера $c_{l,k}$ на уровне $l-1$. Для вложения следующего уровня иерархии в предыдущий делается следующее. Рассматривается кластер $c_{l,k}$ на уровне l иерархии. Пусть в нем содержатся кластеры $c_{l+1,k_1},\ \ldots,\ c_{l+1,k_q}$ уровня $l+1$. Их центры проецируются на плоскость и вычисляются их радиусы $\hat{r}(c_{l+1,k_1}), \ldots, \hat{r}(c_{l+1,k_q})$ описанным выше способом. Центр масс полученной системы проекций совмещается с центром $\hat{\boldsymbol{\mu}}(c_{l,k})$ рассматриваемого кластера на плоскости. Расстояние $\hat{\rho}$ от центра $\hat{\boldsymbol{\mu}}(c_{l,k})$ до максимально удаленной границы дочернего кластера уровня $l+1$ определяется как

$$\hat{\rho} = \max_{k' \in \{k_1,\,\ldots,\,k_q\}} \rho\big(\hat{\boldsymbol{\mu}}(c_{l,k}), \hat{\boldsymbol{\mu}}(c_{l+1,k'})\big) + \hat{r}(c_{l+1,k'}). \tag{5.2}$$

Для сохранения вложенности структуры проводится гомотетия: стягивание, если $\hat{\rho} > \hat{r}(c_{l,k})$ и растяжение, если $\hat{\rho} < \hat{r}(c_{l,k})$ с коэффициентом $\hat{r}(c_{l,k})/\hat{\rho}$ и центром $\hat{\boldsymbol{\mu}}(c_{l,k})$. После этого наиболее удаленные документы дочерних кластеров $c_{l,k_1},\ \ldots,\ c_{l+1,k_q}$ будут находиться внутри рассматриваемого родительского кластера $c_{l,k}$. Эта операция повторяется для всех кластеров по возрастанию уровня кластера. Получается плоская визуализация иерархии, обладающая свойством вложенности.

Результат визуализации На рис. 5.8, 5.9, 5.10 показаны результаты визуализации экспертной модели M алгоритмом, описанным выше. Центры уровня областей отмечены меткой – «×», уровня направлений – «+», а документы d отображаются метками – «○». Вокруг центра каждого кластера проводится его граница. Все объекты, лежащие внутри границы данного кластера, принадлежат ему.

Цвет документа d определяется степенью отличия (2.10) его экспертной кластеризации от верифицированной модели $\hat{M}$, построенной с помощью алгоритма, описанного в разделе 4.1. с весовым параметром матрицы штрафов γ (4.2). Полученный диапазон значений некорректности отображается в цветовую шкалу от RGB $(255; 0; 0)$ – красный (документ, для которого алгоритмическая и экспертная кластеризации отличаются сильнее всего) до RGB $(0; 255; 0)$ – зеленый (документ, для которых экспертная и алгоритмическая кластеризации совпадают).

Как видно из рис. 5.8, 5.9, 5.10, при увеличении значений штрафов γ число документов, отображаемых как некорректно классифицированные, растет.

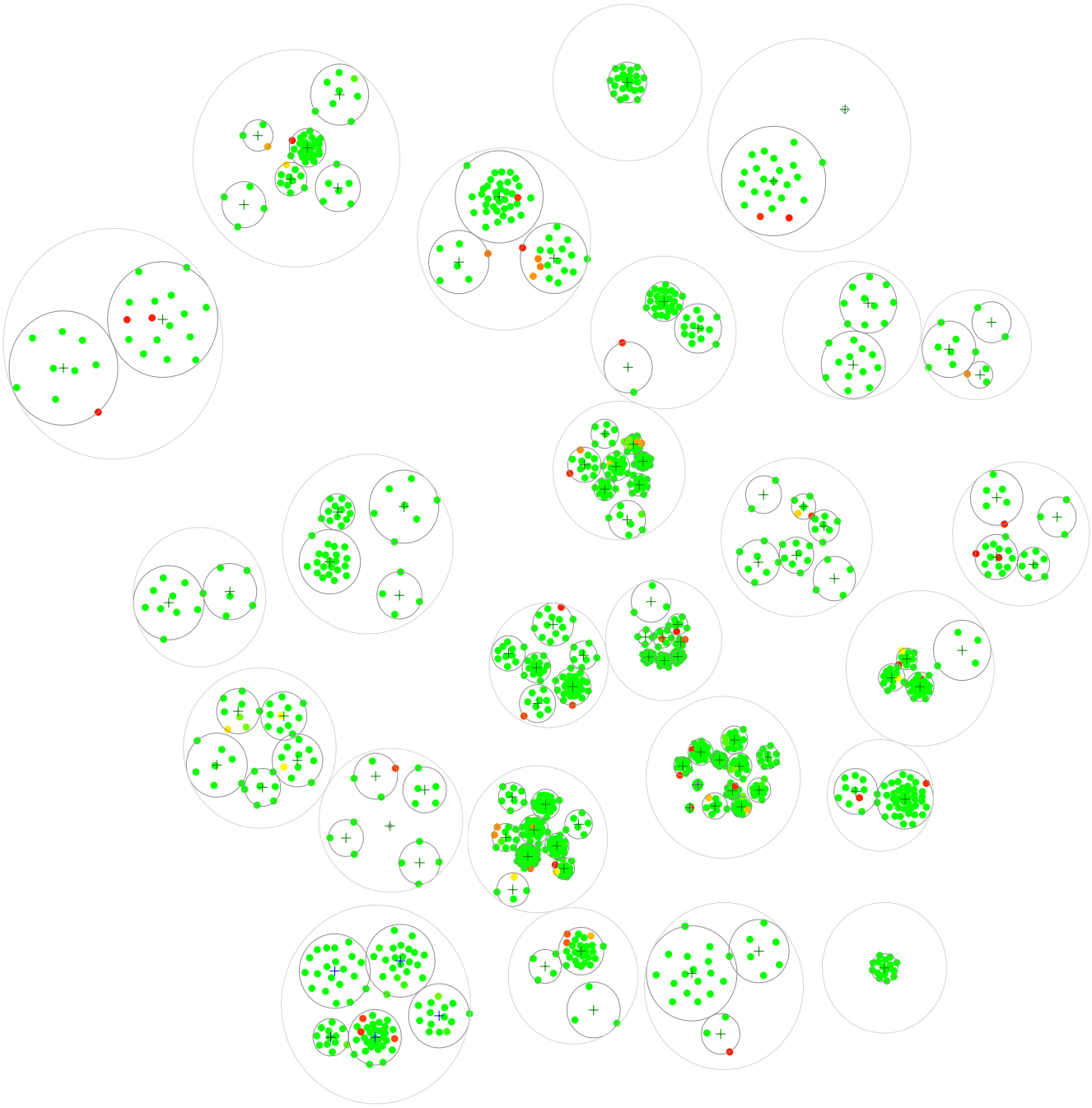


Рис. 5.8. Иерархическая визуализация несоответствий с большими штрафами $\gamma = 1.25$.

Построенная визуализация экспертной иерархической тематической модели на плоскости показывает

1) сходство между кластерами одного уровня иерархии l, лежащими в одном родительском кластере: схожие Area лежат на плоскости рядом, два тематически схожих Stream, лежащие внутри одной Area, также находятся рядом на плоскости,
2) качество отнесения документа к кластеру: если документ находится вдали от других документов из его родительского кластера или на краю кластера, а алгоритм выделяет его красным цветом, то данный документ отлича-

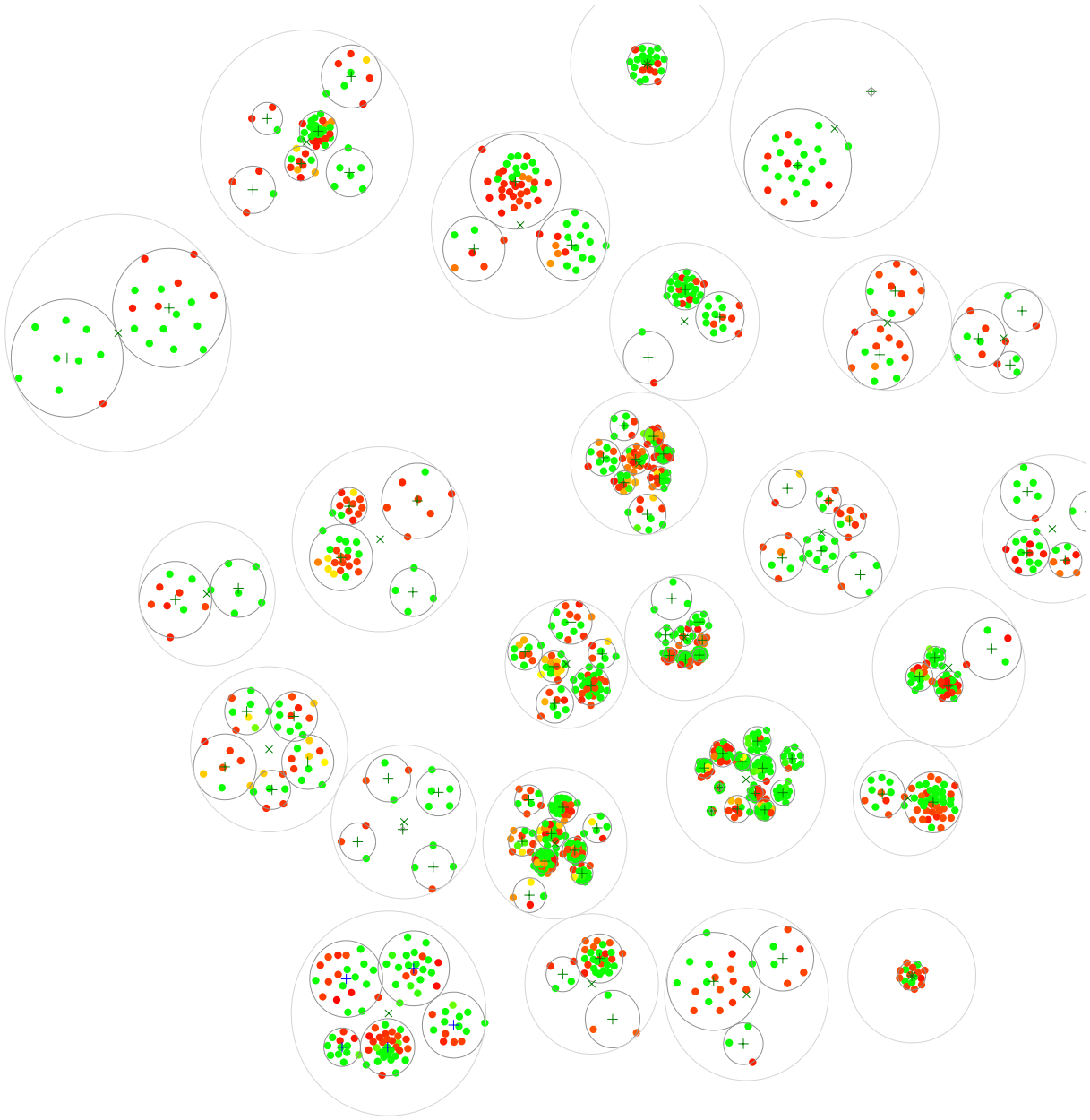


Рис. 5.9. Иерархическая визуализация несоответствий со средними штрафами $\gamma = 0.7$.

ется по терминологическому составу от остальных документов, лежащих в данном кластере.

Визуализация переноса документов. Для каждого документа, для которого экспертная и алгоритмическая кластеризации на совпали, алгоритм верификации, описанный в разделе 4.1. предлагает более релевантный кластер. На рис. 5.11 изображен пример переноса документов в их алгоритмические кластеры.

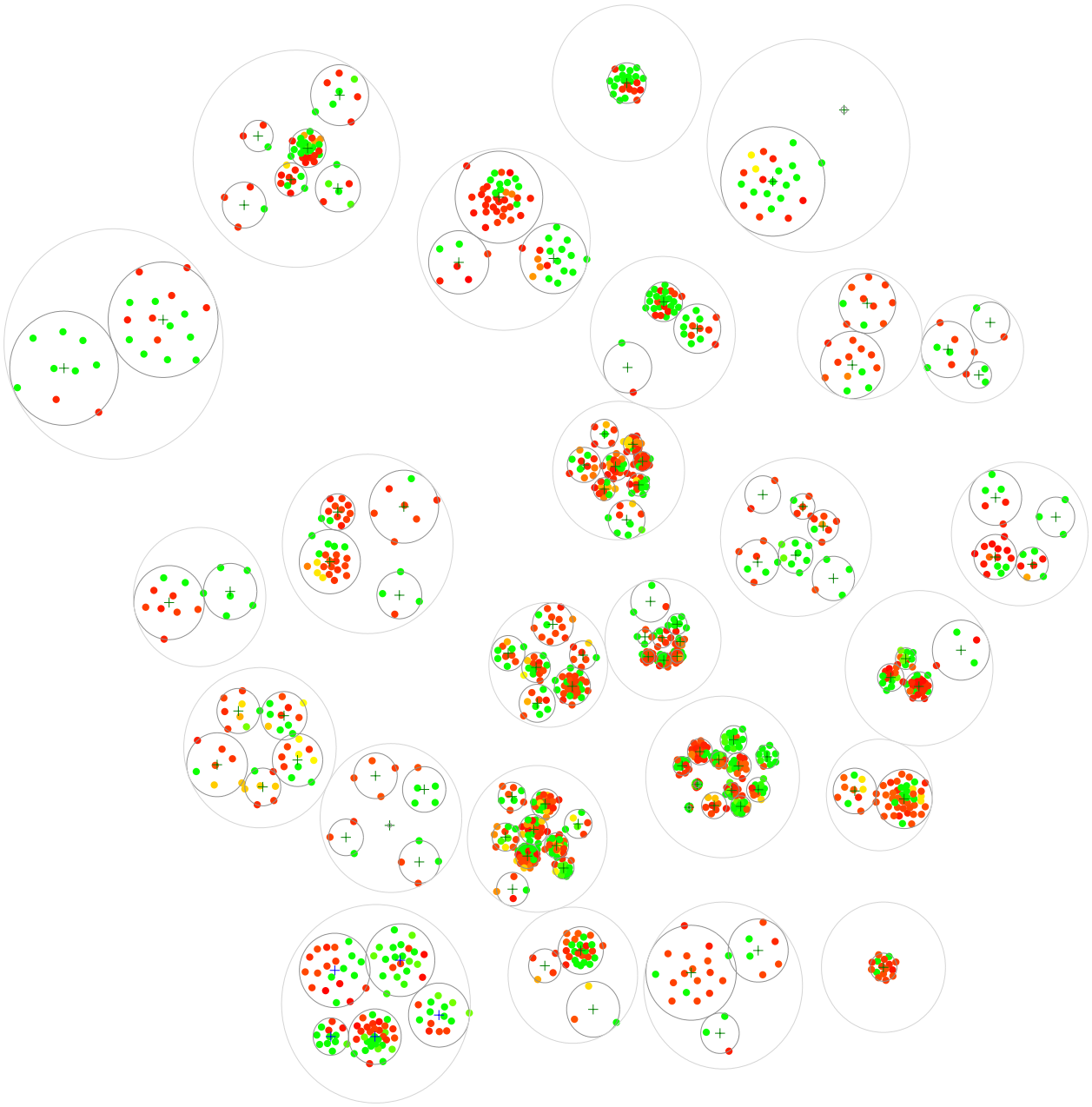


Рис. 5.10. Иерархическая визуализация несоответствий с малыми штрафами $\gamma = 0.5$.

5.3. Иерархическая классификация веб-сайтов индустриального сектора

Для выявления компаний индустриального сектора, работающих в определенной сфере, была построена экспертная система, позволяющая определять для заданного веб-сайта компании наиболее релевантные сферы, используя экспертную кластерную структуру и набор уже размеченных экспертами веб-сайтов.

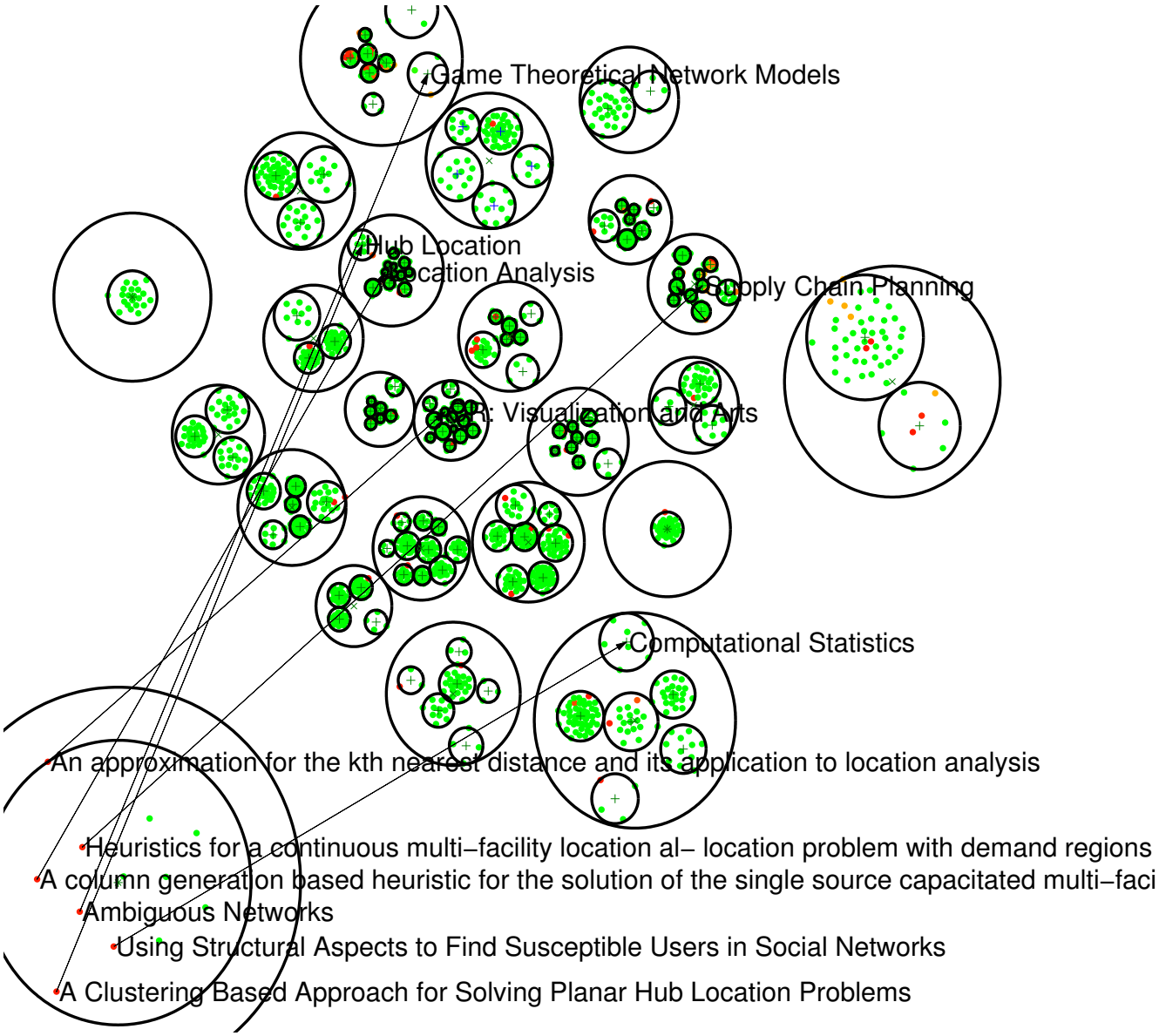


Рис. 5.11. Перенос документов.

Описание данных. Рассматривалась коллекция из 1036 веб-сайтов индустриального сектора, разбитая экспертами на 11 индустрий и 78 подиндустрий [123]. Каждый веб-сайт был представлен в виде набора HTML документов. Данная структура представлена в виде дерева на рис. 5.12.

Для получения матрицы документ признак все HTML-документы, соответствующие одному сайту, объединялись в один. Из полученного документа удалялись все специальные символы и теги разметки после чего использовалась процедура предобработки текстовых документов, описанная в разделе 5.1. В результате была получена предобработанная коллекция с экспертной иерархической тематической моделью:

1) размер коллекции $|D| = 1036$ документов,
2) размер словаря $|W| = 18775$ слова,
3) число кластеров второго уровня $K_2 = 11$,

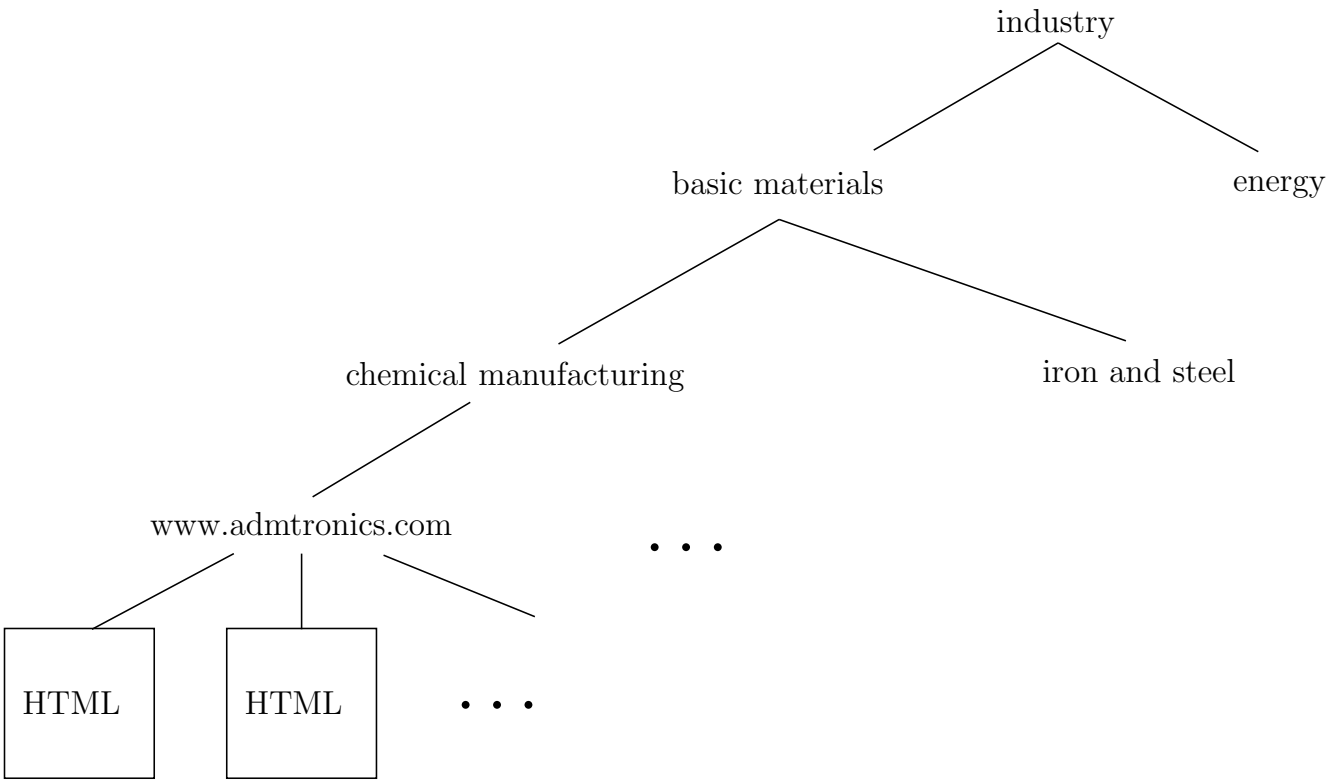


Рис. 5.12. Экспертная иерархическая структура коллекции сайтов индустриального сектора.

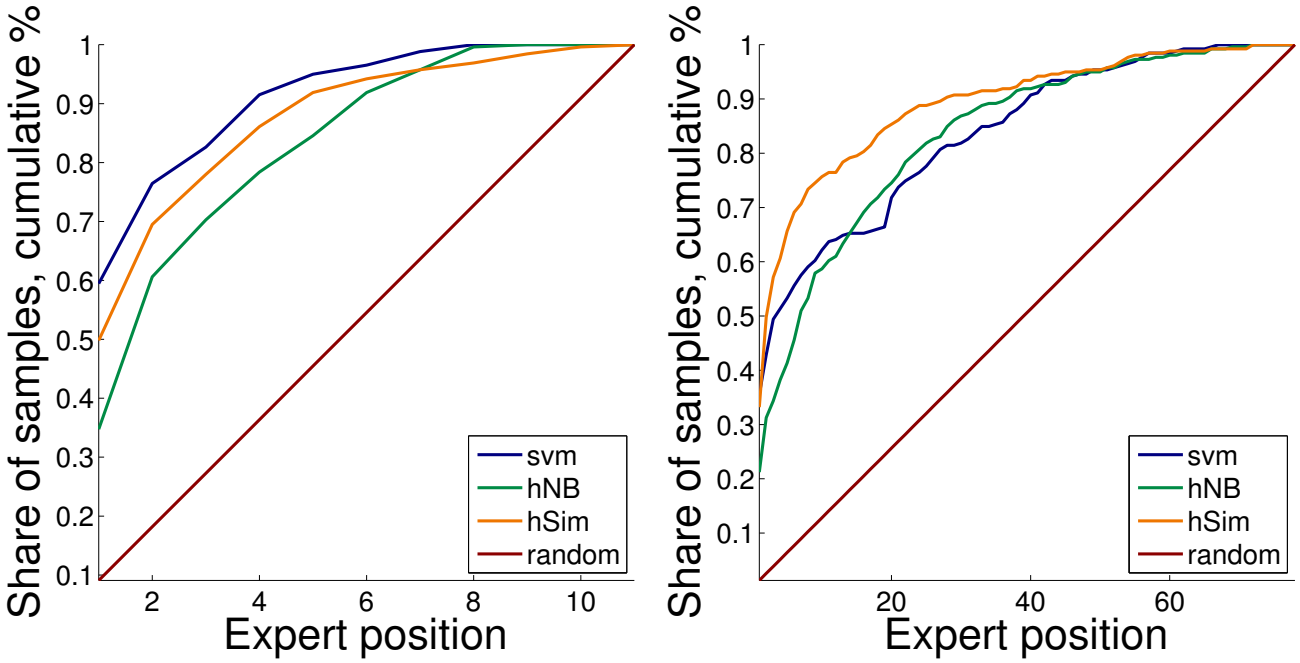
4) число кластеров третьего уровня $K_3 = 78$.

Результаты классификации сайтов индустриального сектора. Коллекция разбивалась на обучающую $D_{\mathcal{V}}$ и тестовую $D_{\mathcal{T}}$ в пропорции $3:1$. Операторы релевантности R (3.14), основанные на иерархической функции сходства hSim (3.12) и алгоритмах svm и hNB, описанных в разделе 1.9., настраивались по выборке $D_{\mathcal{V}}$, после чего оценивалось их качество AUCH (3.17). В таблице 5.3 приведены соответствующие значения AUCH. Для плоской кластеризации svm показал наилучший результат, для задачи иерархической классификации, оператор релевантности, построенный с помощью иерархической функции сходства hSim, показал наилучший результат.

На рис. 5.13 показаны соответствующие результатам из таблицы 5.3 огибающие кумулятивных гистограмм (3.16) для плоской и иерархической классификации выборки $D_{\mathcal{T}}$ с помощью данных алгоритмов.

Таблица 5.3. Значения функционала качества AUCH (3.17) для операторов релевантности, построенных с помощью сравниваемых алгоритмов.

Алгоритм \ Тип классификации	Плоская	Иерархическая
svm	**0.86**	0.83
hNB	0.80	0.83
hSim	0.83	**0.89**



а. Случай плоской классификации.

б. Случай иерархической классификации.

Рис. 5.13. Огибающие кумулятивных гистограмм операторов релевантности $R(\cdot)$, построенных с помощью алгоритмов svm, hNB, hSim.

Заключение

Основные результаты диссертационной работы заключаются в следующем.

В главе 1 введены основные понятия, поставлены задачи иерархической кластеризации и классификации и разобраны основные этапы построения иерархических тематических моделей коллекции документов: методы предобработки текстовых документов, методы составления словаря коллекции, методы представления слов и документов в виде векторов действительного пространства и методы иерархической кластеризации и классификации, включающие в себя алгоритмы построения жестких, вероятностных, описательно-вероятностных моделей и смесей моделей.

В главе 2 проанализированы способы векторного представления документа с помощью булевых, целочисленных и частотных признаков слов. Для кластеризации полученных векторов использована взвешенная метрика на базе расстояния Минковского. Предложен способ оценки качества этой метрики в зависимости от весов и алгоритм построения локально оптимального набора весов. Проведено сравнение агломеративного и дивизимного подходов иерархической кластеризации. Свойства алгоритмов анализировались при решении задачи построения плоской и иерархической кластерной модели коллекции аннотаций к докладам крупной конференции.

В главе 3 предложена взвешенная функция сходства документа и кластера, учитывающая информативность слов в задачах кластеризации и классификации. Предложен алгоритм оптимизации параметров взвешенной функции сходства, использующий энтропию слов относительно экспертной кластеризации на различных уровнях иерархии. Предложен иерархический вариант взвешенной функции сходства, позволяющий вычислять сходство документа с веткой экспертной иерархической кластерной структуры, как сходство с кластером нижнего уровня, а также со всеми его родительскими кластерами с заданными весами. Для классификации нового документа предложен оператор релевантности, возвращающий ранжированный список кластеров нижнего уровня по убыванию их релевантности этому документу. Предложен метод иерархической классификации на основе данного оператора. Для оценки качества оператора релевантности введен критерий качества AUCH. Предложена вероятностная модель иерархической классификации, построена вероятностная модель коллекции, разработан способ оценки вероятности принадлежности документа кластеру нижнего уровня с помощью иерархической функции сходства. Предложен алгоритм оптимизации параметров и гиперпараметров вероятностной модели, максимизирующий ее правдоподобие по размеченным документам. Для случая, когда на параметры модели накладываются априорные распределения, получена аппроксимация апостериорного распределения параметров, а также совместного апостериорного распределения параметров и классов неразмеченных документов. Получены аналитические оценки вероятности принадлежности неразмеченных документов кластерам нижнего уровня экспертной

иерархической структуры. Предложен способ учета синонимичности слов с помощью векторных представлений слов.

В главе 4 рассмотрена задача верификации экспертной тематической модели. Введен функционал качества экспертной модели. Предложен неметрический алгоритм построения иерархической тематической модели, схожей с экспертной, изменяющий класс документа в экспертной модели, если при этом прирост качества больше заданного штрафа за такое изменение. Предложен критерий выбора соотношения штрафов за различные виды переносов.

В главе 5 проведен анализ свойств предложенных методов. Описан реализованный программный комплекс, классифицирующий аннотации докладов крупной конференции EURO с помощью экспертных тематических моделей прошедших конференций. Построена экспертная система, позволяющая классифицировать веб-сайты компаний индустриального сектора. Проведено сравнение предложенных алгоритмов с известными решениями. Предложенные алгоритмы показали более высокие результаты. Для визуализации результатов верификации экспертной модели предложен метод вложенной визуализации иерархической модели на плоскости. На полученном изображении модели выделялись выявленные тематические несоответствия и предлагались способы их устранения.

Список основных обозначений

Матрицы обозначены заглавными жирными буквами, векторы — жирными прописными буквами.

d — текстовый документ

D — коллекция документов

w — слово, любой неразрывный набор символов. $w_1 w_2 \dots w_n$ – словосочетание, состоящее из слов w_1, $w_2, \dots, w_n$

W — словарь коллекции, содержащий различные слова w и словосочетания $w_1 w_2 \dots w_n$ из документов коллекции D

$N(w_1 w_2 \dots w_n, D)$ — число слов или словосочетаний $w_1 w_2 \dots w_n$ в множестве из одного или нескольких документов D

$N(t, D)$ — число документов с темой t в коллекции D

$\mathbf{w}(w)$ — представление слова w в виде вектора

$\mathbf{x}$ — представление документа d в виде вектора

M — тематическая модель

h — число уровней в сбалансированной иерархической модели

K_l — число кластеров на уровне l в тематической модели

$c_{l,k}$ — кластер на уровне l с порядковым номером k среди кластеров данного уровня

$c(\mathbf{x})$ — экспертный кластер документа $\mathbf{x}$ на уровне h

$\hat{c}(\mathbf{x})$ — алгоритмический кластер документа $\mathbf{x}$ на уровне h

$\mathbf{Z}$ — матрица экспертной классификации, $z_{nk} = [\mathbf{x}_n \in c_{h,k}]$

$\boldsymbol{\mu}(c_{l,k})$ — вектор центра кластера $c_{l,k}$, $\boldsymbol{\mu}_{l,k}$ – средний вектор родительского кластера на уровне l кластера нижнего уровня $c_{h,k}$.

$\mathbf{M}_k = [\boldsymbol{\mu}_{1,k}, \dots, \boldsymbol{\mu}_{h,k}]$ – матрица центров родительских кластеров для кластера нижнего уровня $c_{h,k}$

$r(c_{l,k})$ — радиус кластера $c_{l,k}$

$\hat{r}(c_{l,k})$ — радиус кластера $c_{l,k}$ на плоскости

$B^{h-l}(c_{h,k})$ — родительский кластер кластера $c_{h,k}$ на уровне l

λ_m — важность слова с номером m из словаря W

$\boldsymbol{\Lambda}$ — матрица важности слов

$\mathsf{H}^l(w)$ — энтропия слова w относительно экспертной кластеризации на уровне l

$\boldsymbol{\alpha}$ — вектор структурных параметров энтропийной модели оценки важности слов по экспертной кластеризации

$s_{\mathsf{w}}(w_1, w_2)$ — сходство слов w_1 и w_2

$s(\mathbf{x}, \mathbf{y})$ — сходство документов $\mathbf{x}$ и $\mathbf{y}$

$s(\mathbf{x}, c_{l,k})$ — сходство документа $\mathbf{x}$ и кластера $c_{l,k}$

$s_{\mathsf{h}}(\mathbf{x}_n, c_{h,k})$ — иерархическое сходство документа $\mathbf{x}_n$ и кластера нижнего уровня $c_{h,k}$, сокращенное обозначение – $s_{n,k}$

$\mathbf{s}_n = [s_{n,1}, \dots, s_{n,K_h}]^{\mathsf{T}}$ – вектор значений иерархического сходства документа $\mathbf{x}_n$ со всеми кластерами нижнего уровня

$s_{\mathsf{c}}(c_{l,k}, c_{l,k'})$ — сходство кластеров $c_{l,k}$ и $c_{l,k'}$

$\rho(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{y})$ — взвешенное расстояние между документами $\mathbf{x}$ и $\mathbf{y}$
$\boldsymbol{\theta}$ — вектор параметров модели
$\mathrm{tf}(w, D)$ — частота слова w в одном или нескольких документах D
$\mathrm{idf}(w, D)$ — обратная частота слова w в документах коллекции D
q — аппроксимация апостериорного распределения параметров или совместного апостериорного распределения параметров и классов неразмеченных документов
$\mathcal{L}(q)$ — нижняя граница распределения $p(\mathbf{Z})$
$\boldsymbol{\xi}_n$ — вектор вариационных параметров, соответствующий оценке функции softmax в точке $\mathbf{s}_n$
$R(\mathbf{x})$ — оператор релевантности, возвращающий ранжированный список кластеров нижнего уровня в порядке убывания их релевантности документу $\mathbf{x}$
$\mathrm{AUCH}(R)$ — критерий качества оператора релевантности, площадь под огибающей кумулятивной гистограммы
$Q(R)$ — средняя позиция экспертного кластера в перестановках, возвращаемых оператором релевантности R
$\Xi(M)$ — качество иерархической тематической модели как взвешенная сумма средних внутрикластерных и межкластерных сходств
$\upsilon(\mathbf{x}, M, \hat{M})$ — ошибка классификации документа $\mathbf{x}$ в построенной алгоритмической модели $\hat{M}$ относительно экспертной модели M
$\Upsilon(M, \hat{M})$ — расстояние между экспертной моделью M и алгоритмической моделью $\hat{M}$
$V(\rho)$ — функция качества метрики ρ
$\mathcal{I}$ — индексы документов коллекции D. $\mathcal{V}, \mathcal{T}$ — индексы документов для обучения и тестирования модели соответственно.
$\mathrm{pos}(\mathsf{q}, k)$ — позиция числа k в перестановке q
$\mathbf{F}$ — матрица штрафов
γ — весовой множитель матрицы штрафов
δ — элемент $\mathbf{F}$, штраф за перенос документа
$\mathbf{e}(k)$ — единичный вектор, с единицей на позиции k
$\mathbb{R}$ — множество действительных чисел
$|\cdot|$ — число элементов в множестве
$[i = j]$ — индикаторная функция

Список иллюстраций

Список таблиц

Литература

1. *Hofmann Thomas.* Probabilistic Latent Semantic Indexing // Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — SIGIR '99. — New York, NY, USA: ACM, 1999. — Pp. 50–57.
2. *Blei D. M., Ng A. Y., Jordan M. I.* Latent dirichlet allocation // *Journal of Machine Learning Research.* — 2003. — Vol. 3. — Pp. 993–1022.
3. *Blei David M., Griffiths Thomas L., Jordan Michael I.* The Nested Chinese Restaurant Process and Bayesian Nonparametric Inference of Topic Hierarchies // *J. ACM.* — 2010. — Vol. 57, no. 2. — Pp. 7:1–7:30.
4. Hierarchical Dirichlet Processes / Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, David M. Blei // *Journal of the American Statistical Association.* — 2006. — Vol. 101, no. 476. — Pp. 1566–1581.
5. *Blei D., Lafferty J.* Correlated Topic Models // *Advaces in neural information processing systems.* — 2006. — Vol. 18. — P. 147.
6. Keep It Simple with Time: A Reexamination of Probabilistic Topic Detection Models / Qi He, Kuiyu Chang, Ee-Peng Lim, A. Banerjee // *Pattern Analysis and Machine Intelligence, IEEE Transactions on.* — 2010. — Vol. 32, no. 10. — Pp. 1795–1808.
7. Generative Model-based Clustering of Directional Data / Arindam Banerjee, Inderjit Dhillon, Joydeep Ghosh, Suvrit Sra // Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — KDD '03. — New York, NY, USA: ACM, 2003. — Pp. 19–28.
8. *Mimno David, Li Wei, McCallum Andrew.* Mixtures of Hierarchical Topics with Pachinko Allocation // Proceedings of the 24th International Conference on Machine Learning. — ICML '07. — New York, NY, USA: ACM, 2007. — Pp. 633–640.
9. Improving Text Classification by Shrinkage in a Hierarchy of Classes / Andrew McCallum, Ronald Rosenfeld, Tom M. Mitchell, Andrew Y. Ng // Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, 1998. — 1998. — Pp. 359–367.
10. Supervised Topic Classification for Modeling a Hierarchical Conference Structure / Mikhail Kuznetsov, Marianne Clausel, Massih-Reza Amini et al. // Neural Information Processing - 22nd International Conference, ICONIP 2015, Istanbul, Turkey, 2015, Proceedings, Part I. — 2015. — Pp. 90–97.
11. *Hao Pei-Yi, Chiang Jung-Hsien, Tu Yi-Kun.* Hierarchically SVM classification based on support vector clustering method and its application to document categorization // *Expert Systems with Applications.* — 2007. — Vol. 33, no. 3. — Pp. 627–635.
12. *Агеев М. С., Добров Б. В., Лукашевич Н. В.* Автоматическая рубрикация текстов: методы и проблемы. — 2008. — Vol. 150, no. 4. — Pp. 25–40.

13. *Joachims Thorsten.* Text categorization with Support Vector Machines: Learning with many relevant features // Machine Learning: ECML-98: 10th European Conference on Machine Learning Chemnitz, Germany, 1998 Proceedings. — Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. — Pp. 137–142.
14. *Кузьмин А. А., Адуенко А. А., Стрижов В. В.* Тематическая классификация тезисов крупной конференции с использованием экспертной модели // *Информационные технологи.* — 2014. — Т. 6. — С. 22–26.
15. *Loukachevitch N. V., Rubtsova Y. V.* Overcoming Time Gap and Data Sparsity in Tweet Sentiment Analysis // Computational Linguistics and Intellectual Technologies. International Conference "Dialog 2016"Proceedings. — Изд-во РГГУ, Москва, 2016. — Pp. 416–426.
16. *Schedl Markus.* #Nowplaying Madonna: A Large-scale Evaluation on Estimating Similarities Between Music Artists and Between Movies from Microblogs // *Inf. Retr.* — 2012. — Vol. 15, no. 3-4. — Pp. 183–217.
17. Deep Classification in Large-scale Text Hierarchies / Gui-Rong Xue, Dikan Xing, Qiang Yang, Yong Yu // Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — New York, NY, USA: ACM, 2008. — Pp. 619–626.
18. *Ikonomakis M, Kotsiantis S, Tampakas V.* Text classification using machine learning techniques. // *WSEAS Transactions on Computers.* — 2005. — Vol. 4, no. 8. — Pp. 966–974.
19. *Gong Linghui, Zeng Jianping, Zhang Shiyong.* Text stream clustering algorithm based on adaptive feature selection // *Expert Systems with Applications.* — 2011. — Vol. 38, no. 3. — Pp. 1393–1399.
20. *Hartigan J. A., Wong M. A.* Algorithm AS 136: A k-means clustering algorithm // *Applied Statistics.* — 1979. — Vol. 28, no. 1. — Pp. 100–108.
21. *Златов А. С., Кузьмин А. А.* Построение иерархической тематической модели крупной конференции // *Искусственный Интеллект и Принятие Решений.* — 2016. — Т. 0. — С. 00–00.
22. *Vorontsov Konstantin, Potapenko Anna, Plavin Alexander.* Additive Regularization of Topic Models for Topic Selection and Sparse Factorization // Statistical Learning and Data Sciences. — Springer International Publishing, 2015. — Vol. 9047 of *Lecture Notes in Computer Science.* — Pp. 193–202.
23. *Zavitsanos Elias, Paliouras Georgios, Vouros George A.* Non-Parametric Estimation of Topic Hierarchies from Texts with Hierarchical Dirichlet Processes // *J. Mach. Learn. Res.* — 2011. — Vol. 12. — Pp. 2749–2775.
24. *Leisch Friedrich.* A Toolbox for K-centroids Cluster Analysis // *Comput. Stat. Data Anal.* — 2006. — Vol. 51, no. 2. — Pp. 526–544.
25. *Cordeiro de Amorim Renato, Mirkin Boris.* Minkowski Metric, Feature Weighting and Anomalous Cluster Initializing in K-Means Clustering // *Pattern Recogn.* — 2012. — Vol. 45, no. 3. — Pp. 1061–1075.

26. *Yih Wen-tau.* Learning Term-weighting Functions for Similarity Measures // Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2. — EMNLP '09. — Stroudsburg, PA, USA: Association for Computational Linguistics, 2009. — Pp. 793–802.
27. *Gershman Samuel, Hoffman Matthew D., Blei David M.* Nonparametric variational inference // Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK. — 2012.
28. *Blei David M., Kucukelbir Alp, McAuliffe Jon D.* Variational Inference: Review for Statisticians // *CoRR*. — 2016. — Vol. abs/1601.00670.
29. *Bishop C.M.* Pattern Recognition and Machine Learning. Information Science and Statistics. — Springer, 2006.
30. *Kuznetsov M. P., Tokmakova A. A., Strijov V. V.* Analytic and stochastic methods of structure parameter estimation // *Informatica*. — 2016. — Vol. 27, no. 3. — Pp. 607–624. http://strijov.com/papers/HyperOptimizationEng.pdf.
31. *Millar Jeremy R., Peterson Gilbert L., Mendenhall Michael J.* Document Clustering and Visualization with Latent Dirichlet Allocation and Self-Organizing Maps. // FLAIRS Conference. — AAAI Press, 2009.
32. Multi-document Summarization by Visualizing Topical Content / Rie Kubota Ando, Branimir K. Boguraev, Roy J. Byrd, Mary S. Neff // Proceedings of the 2000 NAACL-ANLP Workshop on Automatic Summarization. — NAACL-ANLP-AutoSum '00. — Association for Computational Linguistics, 2000. — Pp. 79–98.
33. *Lee John A., Verleysen Michel.* Nonlinear Dimensionality Reduction. — Springer Publishing Company, Incorporated, 2007.
34. *Sammon J. W.* A Nonlinear Mapping for Data Structure Analysis // *IEEE Trans. Comput.* — 1969. — Vol. 18, no. 5. — Pp. 401–409.
35. *Li Wei, Blei David M., McCallum Andrew.* Nonparametric Bayes Pachinko Allocation // *CoRR*. — 2012. — Vol. abs/1206.5270.
36. *Kogan Jacob, Teboulle Marc, Nicholas Charles.* Data Driven Similarity Measures for k-Means Like Clustering Algorithms // *Information Retrieval*. — 2005. — Vol. 8, no. 2. — Pp. 331–349.
37. *Ruiz Miguel E., Srinivasan Padmini.* Hierarchical Text Categorization Using Neural Networks // *Information Retrieval*. — 2002. — Vol. 5, no. 1. — Pp. 87–118.
38. *Воронцов К. В.* Лекции по методам оценивания и выбора моделей. — URL: http://www.ccas.ru/voron/download/Modeling.pdf. (дата обращения: 26.09.2016).
39. *Boyd Stephen, Vandenberghe Lieven.* Convex Optimization. — New York, NY, USA: Cambridge University Press, 2004.
40. *Стрижов В. В.* Порождение и выбор моделей в задачах регрессии и классификации: Ph.D. thesis / Вычислительный центр РАН. — 2014.

41. *Mnih Andriy, Hinton Geoffrey.* A scalable hierarchical distributed language model // NIPS. — MIT Press, 2009.
42. Efficient Estimation of Word Representations in Vector Space / Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean // *CoRR.* — 2013. — Vol. abs/1301.3781.
43. *Kuzmin A. A., Aduenko A. A., Strijov V. V.* Hierarchical thematic model visualizing algorithm // 26th European Conference on Operational Research. — Rome: 2013. — P. 155.
44. *Kuzmin A. A., Aduenko A. A., Strijov V. V.* Thematic Classification for EURO/IFORS Conference Using Expert Model // 20th Conference of the International Federation of Operational Research Societies. — Barcelona: 2014. — P. 173.
45. *Кузьмин А. А., Стрижов В. В.* Построение иерархических тематических моделей крупных конференций // Математические методы распознавания образов ММРО-17. Тезисы докладов 17-й Всероссийской конференции с международным участием. — г. Светлогорск, Калининградская область: Торус пресс., 2015. — Pp. 224–225.
46. *Кузьмин А. А., Адуенко А. А.* Построение иерархических тематических моделей крупных конференций // Сборник тезисов 23 международной научной конференции студентов, аспирантов и молодых ученных "Ломоносов-2016" секция "Вычислительная математика и кибернетика". — г. Москва: МАКС Пресс., 2016. — Pp. 73–75.
47. *Kuzmin A. A., Aduenko A. A., Strijov V. V.* Thematic Classification for EURO/IFORS Conference Using Expert Model // 28th European Conference on Operational Research. — Poznan: 2016. — P. 206.
48. *Кузьмин А. А.* Многоуровневая классификация при обнаружении движения цен // *Машинное обучение и анализ данных.* — 2012. — Т. 3. — С. 318–327.
49. *Кузьмин А. А., Адуенко А. А., Стрижов В. В.* Выбор признаков и оптимизация метрики при кластеризации коллекции документов // *Известия ТулГУ.* — 2012. — Т. 3. — С. 119–131.
50. *Kuzmin A.A, Strijov V.V.* Validation of the thematic models for document collections // *Informacionnie technologii.* — 2013. — Т. 4. — С. 16–20.
51. *Li Wei, McCallum Andrew.* Pachinko Allocation: DAG-structured Mixture Models of Topic Correlations // Proceedings of the 23rd International Conference on Machine Learning. — ICML '06. — New York, NY, USA: ACM, 2006. — Pp. 577–584.
52. *Mimno David, Li Wei, McCallum Andrew.* Mixtures of Hierarchical Topics with Pachinko Allocation // Proceedings of the 24th International Conference on Machine Learning. — ICML '07. — New York, NY, USA: ACM, 2007. — Pp. 633–640.

53. *Makrehchi Masoud, Kamel Mohamed S.* Automatic Extraction of Domain-specific Stopwords from Labeled Documents // Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval. — ECIR'08. — Berlin, Heidelberg: Springer-Verlag, 2008. — Pp. 222–233.
54. *Savoy Jacques.* Searching Strategies for the Hungarian Language // *Inf. Process. Manage.* — 2008. — Vol. 44, no. 1. — Pp. 310–324.
55. *Ramanathan A. Rao D.* A lightweight stemmer for Hindi // Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics. — EACL'03. — 2003.
56. *Krovetz Robert.* Viewing Morphology As an Inference Process // Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — SIGIR '93. — New York, NY, USA: ACM, 1993. — Pp. 191–202.
57. *Frakes W. B.* Stemming Algorithms // Information Retrieval. — Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992. — Pp. 131–160.
58. *Lovins J. B.* Development of a stemming algorithm // *Mechanical Translation and Computational Linguistics.* — 1968. — Vol. 11. — Pp. 22–31.
59. *Hafer M., Weiss S.* Word Segmentation by Letter Successor Varieties // *Information Storage and Retrieval.* — 1974. — Vol. 10. — Pp. 371–385.
60. YASS: Yet Another Suffix Stripper / Prasenjit Majumder, Mandar Mitra, Swapan K. Parui at al. // *ACM Trans. Inf. Syst.* — 2007. — Vol. 25, no. 4.
61. *Jain A. K., Murty M. N., Flynn P. J.* Data Clustering: A Review // *ACM Comput. Surv.* — 1999. — Vol. 31, no. 3. — Pp. 264–323.
62. *Levenshtein Vladimir.* Binary codes capable of correcting deletions, insertions, and reversals // *Doklady Akademii Nauk SSSR.* — 1965. — Vol. 163, no. 4. — Pp. 845–848.
63. *Adamson George W., Boreham Jillian.* The use of an association measure based on character structure to identify semantically related pairs of words and document titles // *Information Storage and Retrieval.* — 1974. — Vol. 10, no. 7–8. — Pp. 253–260.
64. *Dice L. R.* Measures of the Amount of Ecologic Association Between Species // *Ecology.* — 1945. — Vol. 26, no. 3. — Pp. 297–302.
65. *Нокель М. А., Лукашевич Н. В.* Тематические модели: добавление биграмм и учет сходства между униграммами и биграммами // *Вычислительные методы и программирование.* — 2015. — Vol. 16, no. 2. — Pp. 215–234.
66. *Lau Jey Han, Baldwin Timothy, Newman David.* On Collocations and Topic Models // *ACM Trans. Speech Lang. Process.* — 2013. — Vol. 10, no. 3. — Pp. 10:1–10:14.
67. *Church Kenneth Ward, Hanks Patrick.* Word Association Norms, Mutual Information, and Lexicography // *Comput. Linguist.* — 1990. — Vol. 16, no. 1. — Pp. 22–29.

68. Augmented Mutual Information for Multi-Word Term Extraction / W. Zhang, T. Yoshida, T. Ho, X. Tang // *International Journal of Innovative Computing.* — 2008. — Vol. 8, no. 2. — Pp. 543–554.
69. *Bouma Gerlof.* Normalized (Pointwise) Mutual Information in Collocation Extraction // *Proceedings of the Biennal GSCL Conference.* — 2009. — Pp. 31–40.
70. A Closer Look at Skip-gram Modelling / David Guthrie, Ben Allison, W. Liu et al. // Proceedings of the Fifth international Conference on Language Resources and Evaluation (LREC-2006). — Genoa, Italy: 2006.
71. *Church Kenneth Ward.* A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text // Proceedings of the Second Conference on Applied Natural Language Processing. — ANLC '88. — Stroudsburg, PA, USA: Association for Computational Linguistics, 1988. — Pp. 136–143.
72. A Statistical Approach to Machine Translation / Peter F. Brown, John Cocke, Stephen A. Della Pietra et al. // *Comput. Linguist.* — 1990. — Vol. 16, no. 2. — Pp. 79–85.
73. *Hull Jonathon.* Combining syntactic knowledge and visual text recognition: A hidden Markov model for part of speech tagging in a word recognition algorithm. — Stroudsburg, PA, USA: American Association for Artificial Intelligence, AAAI Press, 1992. — Pp. 77–83.
74. *Kernighan Mark D., Church Kenneth W., Gale William A.* A Spelling Correction Program Based on a Noisy Channel Model // Proceedings of the 13th Conference on Computational Linguistics - Volume 2. — COLING '90. — Stroudsburg, PA, USA: Association for Computational Linguistics, 1990. — Pp. 205–210.
75. *Srihari S. N., Baltus Charlotte M.* Combining statistical and syntactic methods in recognizing handwritten sentences. — Stroudsburg, PA, USA: American Association for Artificial Intelligence, AAAI Press, 1992. — Pp. 121–127.
76. *Goodman Joshua T.* A Bit of Progress in Language Modeling // *Comput. Speech Lang.* — 2001. — Vol. 15, no. 4. — Pp. 403–434.
77. *Katz Slava M.* Estimation of probabilities from sparse data for the language model component of a speech recognizer // IEEE Transactions on Acoustics, Speech and Signal Processing. — 1987. — Pp. 400–401.
78. *Jelinek Fred, Mercer Robert L.* Interpolated estimation of Markov source parameters from sparse data // Proceedings, Workshop on Pattern Recognition in Practice. — Amsterdam: North Holland, 1980. — Pp. 381–397.
79. *Chen Stanley F., Goodman Joshua.* An Empirical Study of Smoothing Techniques for Language Modeling // Proceedings of the 34th Annual Meeting on Association for Computational Linguistics. — ACL '96. — Stroudsburg, PA, USA: Association for Computational Linguistics, 1996. — Pp. 310–318.

80. *Ney H., Essen U., Kneser R.* On Structuring Probabilistic Dependencies in Stochastic Language Modelling // *Computer Speech and Language.* — 1994. — Vol. 8. — Pp. 1–38.
81. A Neural Probabilistic Language Model / Yoshua Bengio, Réjean Ducharme, Pascal Vincent, Christian Janvin // *J. Mach. Learn. Res.* — 2003. — Vol. 3. — Pp. 1137–1155.
82. *Morin Frederic, Bengio Yoshua.* Hierarchical Probabilistic Neural Network Language Model // Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics. — 2005.
83. *Collobert Ronan, Weston Jason.* A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning // Proceedings of the 25th International Conference on Machine Learning. — ICML '08. — New York, NY, USA: ACM, 2008. — Pp. 160–167.
84. Improving Word Representations via Global Context and Multiple Word Prototypes / Eric H. Huang, Richard Socher, Christopher D. Manning, Andrew Y. Ng // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1. — ACL '12. — Stroudsburg, PA, USA: Association for Computational Linguistics, 2012. — Pp. 873–882.
85. Distributed Representations of Words and Phrases and their Compositionality / Tomas Mikolov, Ilya Sutskever, Kai Chen et al. // Proceedings of Neural Information Processing Systems (NIPS). — 2013. — Pp. 3111–3119.
86. Exploring the Space of IR Functions / Parantapa Goswami, Simon Moura, Eric Gaussier at al. // Advances in Information Retrieval. — Springer International Publishing, 2014. — Vol. 8416 of *Lecture Notes in Computer Science.* — Pp. 372–384.
87. *Salton Gerard, McGill Michael J.* Introduction to Modern Information Retrieval. — New York, NY, USA: McGraw-Hill, Inc., 1986.
88. *Адуенко А. А., Стрижов В. В.* Совместный выбор объектов и признаков в задачах многоклассовой классификации коллекции документов // *Инфокоммуникационные технологии.* — 2014. — Vol. 1. — Pp. 47–54.
89. *Katrutsa A. M., Strijov V. V.* Stresstest procedure for feature selection algorithms // *Chemometrics and Intelligent Laboratory Systems.* — 2015. — Vol. 142. — Pp. 172–183.
90. Attribute Selection Based on FRiS-Compactness / N Zagoruiko, I Borisova, V Dyubanov, O Kutnenko // *JMLR Proceedings.* — 2010. — Vol. 10. — Pp. 35–44.
91. *Srivastava Asho, Sahami Mehran.* Text mining : classification, clustering, and applications. — Boca Raton, FL: CRC Press, 2009.

92. *Strijov Vadim, Weber Gerhard Wilhelm.* Nonlinear Regression Model Generation Using Hyperparameter Optimization // *Comput. Math. Appl.* — 2010. — Vol. 60, no. 4. — Pp. 981–988.
93. *Le Quoc V., Mikolov Tomas.* Distributed Representations of Sentences and Documents // *CoRR.* — 2014. — Vol. abs/1405.4053.
94. Parsing Natural Scenes and Natural Language with Recursive Neural Networks / Richard Socher, Cliff Chiung-Yu Lin, Andrew Y. Ng, Christopher D. Manning // ICML. — Omnipress, 2011. — Pp. 129–136.
95. *Platt John C.* Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods // Advances in large margin classifiers. — MIT Press, 1999. — Pp. 61–74.
96. *Brants Thorsten, Chen Francine, Farahat Ayman.* A System for New Event Detection // Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. — SIGIR '03. — New York, NY, USA: ACM, 2003. — Pp. 330–337.
97. *Ackermann Marcel R., Blömer Johannes, Sohler Christian.* Clustering for Metric and Nonmetric Distance Measures // *ACM Trans. Algorithms.* — 2010. — Vol. 6, no. 4. — Pp. 1–26.
98. Metric concentration search procedure using reduced matrix of pairwise distances / A. M. Katrutsa, M. P. Kuznetsov, K. V. Rudakov, V. V. Strijov // *Intelligent Data Analysis.* — 2015. — Vol. 19(5). — Pp. 1091–1108.
99. *Zhang Jin, Korfhage Robert R.* A Distance and Angle Similarity Measure Method // *J. Am. Soc. Inf. Sci.* — 1999. — Vol. 50, no. 9. — Pp. 772–778.
100. *Loohach Richa, Garg Kanwal.* Effect of Distance Functions on Simple K-means Clustering Algorithm // *International Journal of Computer Applications.* — 2012. — Vol. 49, no. 6. — Pp. 7–9.
101. *Hand DJ, Krzanowski WJ.* Optimising k-means clustering results with standard software packages // *Computational statistics and Data analysis.* — 2005. — Vol. 49. — Pp. 969–973.
102. *Kullback S., Leibler R. A.* On Information and Sufficiency // *Ann. Math. Statist.* — 1951. — Vol. 22, no. 1. — Pp. 79–86.
103. *Mahalanobis P. C.* On the generalised distance in statistics // Proceedings National Institute of Science, India. — Vol. 2. — 1936. — Pp. 49–55.
104. Clustering with Bregman Divergences / Arindam Banerjee, Srujana Merugu, Inderjit S. Dhillon, Joydeep Ghosh // *J. Mach. Learn. Res.* — 2005. — Vol. 6. — Pp. 1705–1749.
105. *Воронцов К. В.* Вероятностное тематическое моделирование. — URL: http://www.machinelearning.ru/wiki/images/2/22/Voron-2013-ptm.pdf. (дата обращения: 26.09.2016).
106. *Aitchison J.* The Statistical Analysis of Compositional Data. — London, UK, UK: Chapman & Hall, Ltd., 1986.

107. *Ferguson Thomas S.* A Bayesian Analysis of Some Nonparametric Problems // *The Annals of Statistics.* — 1973. — Vol. 1, no. 2. — Pp. 209–230.
108. *Sethuraman J.* A constructive definition of Dirichlet priors // *Statistica Sinica.* — 1994. — Vol. 4. — Pp. 639–650.
109. *Blackwell D., MacQueen J. B.* Ferguson distributions via Polya urn schemes // *Ann. Statist.* — 1973. — Vol. 1. — Pp. 353–355.
110. *Aldous DavidJ.* Exchangeability and related topics // École d'Été de Probabilités de Saint-Flour XIII — 1983. — Springer Berlin Heidelberg, 1985. — Vol. 1117 of *Lecture Notes in Mathematics.* — Pp. 1–198.
111. *Johnson Norman L., Kotz Samuel.* Urn Models and Their Applications: An Approach to Modern Discrete Probability Theory. — New York: Wiley, 1977.
112. *Li Wei, McCallum Andrew.* Pachinko Allocation: DAG-structured Mixture Models of Topic Correlations // Proceedings of the 23rd International Conference on Machine Learning. — ICML '06. — New York, NY, USA: ACM, 2006. — Pp. 577–584.
113. An Introduction to MCMC for Machine Learning / Christophe Andrieu, Nando de Freitas, Arnaud Doucet, MichaelI. Jordan // *Machine Learning.* — 2003. — Vol. 50, no. 1-2. — Pp. 5–43.
114. *Mardia K. V., Jupp. P.* Directional Statistics (2nd edition). — John Wiley and Sons Ltd., 2000.
115. *Dhillon Inderjit S., Sra Suvrit.* Modeling Data using Directional Distributions: Tech. Rep. TR-03-06: The University of Texas, Department of Computer Sciences, 2003.
116. *Константинов Р. В.* Функциональный анализ. Курс лекций. — Долгопрудный: МФТИ, 2009.
117. Stochastic Variational Inference / Matthew D. Hoffman, David M. Blei, Chong Wang, John Paisley // *J. Mach. Learn. Res.* — 2013. — Vol. 14, no. 1. — Pp. 1303–1347.
118. *Gibbs M.* Bayesian Gaussian Processes for Regression and Classification: Ph.D. thesis. — 1997.
119. Collection of EURO and IFORS abstracts. — URL: https://sourceforge.net/p/mlalgorithms/code/HEAD/tree/PhDThesis/Kuzmin/Data/EURO/. (last checked: 26.09.2016).
120. *Loper Edward, Bird Steven.* NLTK: The Natural Language Toolkit // Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. — Philadelphia: 2002. — Pp. 62–69.
121. *Miller George A.* WordNet: A Lexical Database for English // *Commun. ACM.* — 1995. — Vol. 38, no. 11. — Pp. 39–41.
122. *Porter M. F.* Readings in Information Retrieval. — San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997. — Pp. 313–316.

123. Collection of industry sector websites. — URL: https://sourceforge.net/p/mlalgorithms/code/HEAD/tree/PhDThesis/Kuzmin/Data/Industry_Sector/. (last checked: 26.09.2016).