САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

На правах рукописи

Кижаева Наталья Александровна

ИССЛЕДОВАНИЕ ПАТТЕРНОВ В ТЕКСТАХ НА ОСНОВЕ ДИНАМИЧЕСКИХ МОДЕЛЕЙ

01.01.09 —
Дискретная математика и математическая кибернетика

Диссертация
на соискание ученой степени
кандидата физико-математических наук

Научный руководитель:
доктор физико-математических наук, профессор
Олег Николаевич Граничин

Санкт-Петербург
2018

Оглавление

Введение

Актуальность темы. На протяжении последних десятилетий наблюдается значительный рост объема текстовой информации, генерируемой каждый день. Этот огромный объем данных представляется в различных формах, таких как записи в социальных сетях, записи осмотра пациентов, данные медицинского страхования, статьи новостных агентств, отчеты о работе технических устройств и т.п. Текстовые данные — это пример неструктурированной информации, которая легко обрабатывается и воспринимается человеком, но является гораздо более сложной для понимания компьютером. Задача интеллектуального анализа текстов состоит в извлечении полезной информации из неструктурированных текстов, их автоматической категоризации, классификации и кластеризации. Автоматизированный анализ позволяет исследователям не только собирать и изучать объем материала, анализ которого вручную невозможен, но и выявлять закономерности, незаметные при простом прочтении.

Интеллектуальный анализ текстов является частью более широко класса задач интеллектуального анализа данных, машинного обучения и теории распознавания образов. Современные алгоритмы машинного обучения (классификации, кластеризации) и теории распознавания образов базируются на работах С.А.Айвазяна [1], М.А.Айзермана [2], Э.М.Бравермана [2], В.Н.Вапника [3], Ф.Розенблатта [121], Л.И.Розоноэра [2], Р.А.Фишера [10], В.Н.Фомина [11], И.Форджи [56], К.Фукунаги [58], Я.З.Цыпкина [12], [13], А.Я.Червоненкиса [3], Дж.Хартигана [69], Дж.Хопфилда [72] и др. Исследования рандомизированного машинного обучения начались с основополагающей статьи Вадьясагара [142] и в прошедшие десятилетия тема активно изучалась в научной литературе (О.Н. Граничин [61], М.Кампи [32], Б.Т. Поляк [8], Ю.С. Попков [116], М.В.Хлебников [8]).

Большинство методов интеллектуального анализа текстов рассматривают текст как статический объект, не учитывая процесс его написания или динамику последовательности изложения. В то же время дина-

мика текстового документа может служить его отличительной характеристикой, признаком, по которому в множестве текстов можно выделить группы схожих документов. Это, в свою очередь, открывает множество сфер применения: определение авторства текстов, выявление плагиата, поиск аномалий в отчетах о работе технических устройств и т. п.

Перечисленные факторы актуализируют разработки методов классификации текстовых документов, которые кроме статических характеристик текстов и их фрагментов учитывали бы связи (корреляции) между последовательностями отрывков (фрагментов текстовых документов).

Целью работы является исследование паттернов динамической модели текстовых документов.

Для достижения цели было необходимо решить следующие задачи:

- Разработать метод построения динамических моделей текстовых документов.
- Исследовать, является ли динамика изменений фрагментов текстового документа его отличительной характеристикой.
- Разработать и обосновать алгоритмы кластеризации динамических моделей.

Методы исследования. В диссертации применяются методы теории оценивания и оптимизации, функционального анализа, теории вероятностей и математической статистики.

Основные результаты. В работе получены следующие основные научные результаты:

1. Предложен метод построения динамических моделей текстовых документов.
2. Разработан и теоретически обоснован алгоритм классификации фрагментов текстовых документов, основанный на кластеризации спектрального представления динамических моделей текстовых документов.

3. Разработан и теоретически обоснован алгоритм классификации фрагментов текстовых документов, основанный на кластеризации динамических моделей текстовых документов с помощью расстояний на ядрах.

Научная новизна. Все основные научные результаты диссертации являются новыми.

Теоретическая ценность и практическая значимость. Теоретическая ценность работы состоит в предложенном новом методе построения динамической модели текста и в обосновании разработанных новых алгоритмов классификации фрагментов текстовых документов.

Предложенные новые методы находят применение во множестве прикладных и исследовательских задач: определение авторства текстов в литературных исследованиях, криминалистике, выявление плагиата и т. п. Анализ неструктурированной текстовой информации в отчетах технических устройств с помощью предложенного алгоритма предоставляет возможность выявления неоднородности стиля, а, значит, и возможного сбоя технического устройства.

Апробация работы. Материалы диссертации докладывались на семинарах кафедр системного программирования и теоретической кибернетики математико-механического факультета СПбГУ, семинарах Лаборатории анализа и моделирования социальных процессов СПбГУ, семинарах факультета интеллектуальной обработки информации колледжа ОРТ им. Брауде (Кармиэль, Израиль), на международных конференциях AINL-ISMW FRUCT Artificial Intelligence and Natural Language & Information Extraction, Social Media and Web Search (9-14 ноября, 2015, Санкт-Петербург, Россия), XXVIII Международная научная конференция по источниковедению и историографии стран Азии и Африки “Азия и Африка в меняющимся мире” (22-24 апреля, 2015, Санкт-Петербург, Россия), 2015 IEEE International Symposium on Intelligent Control (September 21-23, 2015, Sydney, Australia), 2017 IEEE Conference on Control Technology and Applications (August 27-30, 2017, Coast, Hawaii, USA), 8th International Scientific Conference on Physics and Control (PhysCon 2017) (July 17-19,

Florence, Italy).

Результаты диссертации были использованы в работах по грантам СПбГУ:

- “Исследование возможностей кластеризации рукописных текстов на арабском языке” 6.37.181.2014.

- “Определение формальных характеристик арабографических рукописей и их цифровая обработка” 2.37.175.2014.

Публикации. Основные результаты исследований опубликованы в 7 работах [5], [6], [7], [16], [60], [87], [88]. Из них три [16], [60], [88] опубликованы в изданиях, индексируемых в базе данных Scopus, и одна [87] в журнале, входящем в перечень рецензируемых научных журналов, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук.

Работы [7], [16], [60], [87], [88] написаны в соавторстве. В работах [7], [16], [60], [87], [88] Н.А. Кижаевой принадлежат формулировки и доказательства теорем, результаты моделирования, а соавторам — постановка задачи и выбор направления решения.

Структура и объем диссертации. Диссертация состоит из введения, трех глав, заключения, списка литературы, включающего 150 источников. Текст занимает 86 страниц и содержит 10 рисунков.

Во введении обосновывается актуальность темы диссертационной работы, формулируется цель и ставятся задачи исследования, кратко излагаются основные результаты.

В первой главе вводятся основные понятия и постановки задач исследований предметной области.

В п. 1.1 рассматриваются основные проблемы и задачи, которые возникают в сфере интеллектуального анализа текстовых данных. Ключевые задачи интеллектуального анализа текстов включают в себя извлечение информации, реферирование, обучение с учителем, обучение без учителя, извлечение мнений, анализ биомедицинских данных.

В п. 1.2 перечисляются этапы предварительной обработки и дается описание распространенных моделей представления текстовых данных. Предобработка текстов — важный этап большинства алгоритмов. Этап предобработки обычно состоит из токенизации, фильтрации, лемматизации и стемминга. Векторная модель — представление текстов в виде векторов из некоторого общего для всех текстов векторного пространства.

В п. 1.3 и 1.4 формулируются проблемы классификации и кластеризации и приводятся классические алгоритмы для их решения.

В п. 1.5 даны определения мер сходства и различия, приведены примеры широко используемых функций расстояния и схожести. Даны определения ядерных функций, упомянуты связанные с этими понятиями важные теоретические результаты.

Во второй главе предложен один из возможных методов построения динамической модели текста. На основе предложенной динамической модели были разработаны и обоснованы два метода классификации документов и их фрагментов. Первый метод основан на кластеризации периодограмм, второй использует кластеризацию с помощью расстояния основанного на некоторых ядрах. Сформулированы теоремы об однозначности и корректности построенных процедур классификации.

В п. 2.1 приводится метод построения динамической модели текста. Пусть $\{X_i\}_{i=1}^n$ — множество текстовых документов. Под текстовым документом будет понимать упорядоченное множество символов.

$\forall i = 1, \ldots, n$ разделим документ X_i на m_i последовательных фрагментов:

$$X_i = x_i^1 + \ldots + x_i^{m_i},$$

где "+" — операция конкатенации строк. Рассмотрим множество всех фрагментов $\overline{X} = \{x_i^j\}_{i \in 1..n, j \in 1..m_i}$.

Введем отображение V, которое сопоставляет фрагменту $x_i^j \in \overline{X}$ некоторое вероятностное распределение $\mathbf{P} \in \mathcal{P}_M$ из множества вероят-

ностных распределений на $\{1, \ldots, M\}$:

$$V : \overline{X} \to \mathcal{P}_M.$$

Таким образом

$$\mathbf{x}_i^j = V(x_i^j) \in \mathbb{R}^M.$$

Обозначим $\mathcal{X} = \{\mathbf{x}_i^j\}_{i \in 1..n, j \in 1..m_i}$ — множество всех фрагментов в векторном представлении.

Значение параметра M определяется выбранной векторной моделью.

Будем считать, что на множестве $\mathbb{R}^M \times \mathbb{R}^M$ определена некоторая функция похожести двух отрывков

$$r : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}.$$

Пусть $T > 0$. Для $i \in 1..n$, $j > T$, $\mathbf{x}_i^j \in \mathcal{X}$ обозначим через $\Delta_{x_i^j}$ множество предшествующих ему векторов-фрагментов: $\Delta_{\mathbf{x}_i^j} = \{\mathbf{x}_i^{j-T}, \ldots, \mathbf{x}_i^{j-1}\}$.

Каждая последовательность векторов-фрагментов $\Delta_{\mathbf{x}}$ с помощью описанной выше функции (2.3) порождает функцию $s_{\mathbf{x}}(\cdot) : \mathbb{R}^M \to \mathbb{R}$:

$$s_{\mathbf{x}}(\mathbf{y}) = \frac{1}{T} \sum_{\mathbf{x}' \in \Delta_{\mathbf{x}}} r(\mathbf{x}', \mathbf{y}),$$

которую будем называть *динамической моделью*.

Значения функции $s_{\mathbf{x}}(\mathbf{y})$ соответствуют средней похожести вектора-фрагмента $\mathbf{y}$ с каждым из векторов-фрагментов из $\Delta_{\mathbf{x}}$.

Таким образом, введено отображение

$$\phi : \mathbf{x}_i^j \to s_{\mathbf{x}}(\cdot).$$

В п. 2.2 предложен алгоритм кластеризации с помощью спектрального представления и правило классификации на его основе. Сформулирована теорема о корректности построенной процедуры.

Формулировка алгоритма:

X — множество текстов

T — параметр задержки

$k^\star$ — максимальное количество кластеров

Cl — алгоритм кластеризации

CLV — индекс алгоритма валидации кластеризации

1. Преобразовать документ $\mathcal{X}_i \in X$ во временной ряд $\mathcal{S}_i$ последовательно применив (2.6) и (2.7).

2. Для каждого временного ряда вычислить периодограмму $\mathrm{PG}(\mathcal{S}_i)$.

3. $for\ k = 2\ to\ k^*$

   - $\mathcal{T} = Cl(\{\mathrm{PG}(S_i)\}_{i\in 1..n}, k)$;
   - $ind_k = CLV(\mathcal{T})$;

4. Количество кластеров соответствует оптимальному числу кластеров, согласно значению индекса ind_k $\{k = 2, .., k^*\}$.

Правило классификации 1:

Два документа X_i и X_j относятся к одному классу l_k, если соответствующие им периодограммы $\mathrm{PG}(\mathcal{S}_i)$ и $\mathrm{PG}(\mathcal{S}_j)$ попали в один кластер k.

Теорема 1. *Кластеризация в пространстве* $\mathbb{F}$ *обеспечивает однозначность и корректность правила классификации.*

В п. 2.3 предложен алгоритм кластеризации с помощью расстояний на ядрах и правило классификации на его основе. Сформулирована теорема о корректности построенной процедуры.

Формулировка алгоритма: $\mathcal{X}$ — коллекция текстов

T — параметр задержки

k — число групп

1. Построить $\mathbb{X} = \{\mathbf{x}_i^j\}_{j=T+1}^m$.

2. Для каждого $\mathbf{x}$ построить динамическую модель $s_{\mathbf{x}}$ по (2.4).

3. Вычислить $F(\mathbf{x})$ для каждого $\mathbf{x}$ по (2.11).

4. Разделить множество $\mathcal{F}$ на k кластеров с помощью алгоритма кластеризации Cl.

Правило классификации 2

Два фрагмента $\mathbf{x}_i$ и $\mathbf{x}_j$ относятся к одному классу l_k, если соответствующие им вектора $F(\mathbf{x}_i)$ и $F(\mathbf{x}_j)$ попали в один кластер k.

Теорема 2. *Если $r(\mathbf{x}, \mathbf{y})$ — положительно определенное ядра и выполнено Предположение 1 кластеризация в пространстве $\mathcal{F}$ обеспечивает однозначность и корректность правила классификации.*

В третьей главе представлены результаты применения предложенных алгоритмов кластеризации к задаче определения авторства текстов нескольких серий популярных книг.

В п. 3.1 дается определение задачи определения авторства и приводится краткий обзор методов решения.

В п. 3.2 приводится результат применения алгоритма классификации текстов на основе кластеризации с помощью спектрального представления к задаче определения авторского стиля в двух коллекциях книг.

В п. 3.3 приводится результат применения алгоритма классификации текстов на основе кластеризации с помощью расстояния на ядрах к задаче определения авторского стиля в трех коллекциях книг.

Результаты применения предложенных алгоритмов к анализу серийных последовательностей книг показывают, что рассмотренная в диссертации новая динамическая модель фрагментов текстов дает для каждого автора некоторые новые уникальные характеристики его стиля.

В заключении диссертации подведены итоги проведенного и завершенного в рамках поставленных задач исследования.

Глава 1

Интеллектуальный анализ текстов

В этой главе рассматриваются основные проблемы и задачи, которые возникают в сфере интеллектуального анализа текстовых данных. Перечисляются этапы предварительной обработки и дается описание распространенных моделей представления текстовых данных. Формулируются проблемы классификации и кластеризации и приводятся классические алгоритмы для их решения.

1.1 Основные задачи

Извлечение информации — одна из ключевых задач интеллектуального анализа текстов, основной целью которой является получение структурированной информации (фактов) из неструктурированных или полуструктурированных текстовых данных. Часто служит промежуточным этапом в решении других задач анализа текстов. Так, например, определение именованных сущностей (англ. Name Entity Recognition) и их связей может выявить важную семантическую информацию в текстовых данных для улучшения результатов поисковой выдачи.

Реферирование. Во многих приложениях может быть необходимо резюмировать текст для того, чтобы предоставить краткий обзор большого документа или коллекции документов на определенную тему. Методы

реферирования можно разделить на два типа. При первом типе реферирования, в резюме содержатся информационные единицы из исходного текста. При втором типе, напротив, резюме может содержать "синтезированную" информацию, которая необязательно присутствовала в текстовых документах.

Обучение с учителем — класс методов машинного обучения, которые используют тренировочные данные (т. е. входные данные и соответствующие им выходные данные) для обучения регрессионной функции или классификатора. Так как множество прикладных задач можно переформулировать как задачу классификации, то часто под обучением с учителем понимают методы классификации. Множество традиционных алгоритмов машинного обучения таких, как байесовский классификатор, деревья решений, классификатор ближайших соседей применяются для решения задач интеллектуального анализа текстов.

Обучение без учителя. Для алгоритмов обучения без учителя не требуется набор тренировочных данных, поэтому их можно применять к текстовым данным без дополнительной обработки вручную. Наиболее распространенными методами обучения без учителя в сфере интеллектуального анализа текстов являются кластеризация и тематическое моделирование. Задача кластеризации заключается в нахождении разбиения корпуса текстов на группы, например, документов, относящихся к одной теме. Кластеризация и тематическое моделирование тесто связаны между собой. В тематическом моделировании используются вероятностные модели для определения "нежесткой" кластеризации, в которой для документа определяются вероятности принадлежности к кластеру, в противоположность "жесткому" разделению документов, когда один документ может принадлежать только одному кластеру.

Извлечение мнений. Значительное количество текстовых данных, доступных в сети Интернет, представляет собой отзывы о продуктах или мнение пользователей в социальных сетях. Анализ такого рода текстовой информации имеет широкое практическое применение: поддержка клиентов или бизнес-аналитика, проведение социальных исследований.

Анализ биомедицинских данных — анализ текстов на биомедицинскую тематику. Интеллектуальный анализ текстов в сфере биомедицины облегчает ученым доступ к информации, заключенной в огромном объеме биомедицинской литературы и амбулаторных карт пациентов. Множество алгоритмов анализа текстов также были адаптированы и расширены для применения к задаче распознавания различных биомедицинских сущностей, таких как последовательности генома, данные экспрессии генов и структуры белка.

1.2 Представление текста

Анализ большой коллекции документов — сложный процесс, поэтому важно ввести такое представление документов, которое облегчало бы дальнейшую работу с ними. Одной из самых распространенных моделей представления текстов является модель мешка слов (англ. bag of words), которая учитывает частоту появления слов, но игнорирует их порядок в тексте. Такая модель приводит к векторному представлению текста, которое далее можно анализировать с помощью алгоритмов понижения размерности. Среди основных алгоритмов понижения размерности можно упомянуть латентно-семантический анализ, вероятностный латентно-семантический анализ и тематическое моделирование.

1.2.1 Предобработка текстов

Предобработка текста — один из ключевых этапов большинства алгоритмов интеллектуального анализа текстов. Классическая методология категоризации текстов включает в себя этапы предобработки, извлечения признаков и классификации. Несмотря на то, что, как было показано в работах [54], [64], [138], извлечение признаков, их отбор и метод классификации вносят значительный вклад в процесс классификации, предобработка может серьезно повлиять на ее результат. Этап предобработки состоит из токенизации, фильтрации, лемматизации и стемминга.

Токенизация: разбиение последовательности символов на части (слова/фразы), называемые токенами. Также может включать в себя удаление определенных символов, например, знаков пунктуации.

Фильтрация заключается в удалении некоторых слов из текста. Распространенный вид фильтрации — удаление стоп-слов. Под стоп-словами понимаются такие слова, которые часто встречаются в тексте и не несут содержательной информации (предлоги, союзы и т. п.)

Лемматизация включает в себя морфологический анализ слов, при котором различные формы слова группируются для того, чтобы их можно было обрабатывать как один объект. При лемматизации документов для каждого слова необходимо определить часть речи. Так как определение части речи очень сложный процесс, подверженный ошибкам, на практике чаще пользуются методами стемминга.

Стемминг — процесс нахождения основы слова, которая не обязательно совпадает с его морфологическим корнем. Алгоритмы стемминга зависят от языка. Первый алгоритм для английского языка был предложен в 1968 году [98]. Наиболее распространенным на сегодняшний день является стеммер Портера [117]. Опубликованный в 1980 году, оригинальный алгоритм был предназначен для английского языка, но впоследствии автором были предложены стеммеры для распространенных индоевропейских языков, в том числе для русского языка.

1.2.2 Векторная модель

Векторная модель — представление текстов в виде векторов из некоторого общего для всех текстов векторного пространства. Этот подход является одним из основных инструментов в области интеллектуального анализа текстов, информационного поиска, классификации и кластеризации текстовых документов.

Каждая координата вектора в рамках модели соответствует отдельному терму. Определение терма зависит от сферы применения и в его роли могут выступать отдельные слова, группы слов, комбинации цифр

и букв. Если терм присутствует в документе, то соответствующее значение в векторе отлично от нуля. Существует несколько стандартных способов подсчета этих значений, известных также как веса термов. Это может быть булевский вес, равный 1, если терм встретился в документе и 0 в противном случае. Другой вариант — количество вхождений терма в документ. В классической векторной модели, предложенной Сэлтоном и др. [126], веса термов представляют собой произведение локальных и глобальных параметров. Такая модель известна как tf-idf (англ. term frequency — inverse document frequency, частота терма — обратная частота документа). Вектор весов $v_d = \mathrm{col}(w_{1,d}, w_{2,d}, \ldots, w_{N,d})$ для документа d определяется следующим образом:

$$w_{t,d} = \mathrm{tf}_{t,d} \cdot \log \frac{|D|}{|\{d' \in D \,|\, t \in d'\}|},$$

где

- $\mathrm{tf}_{t,d}$ — частота терма в документе (локальный параметр),
- $\log \frac{|D|}{|\{d' \in D \,|\, t \in d'\}|}$ — обратная частота документа в коллекции (глобальный параметр).

Здесь $|D|$ — общее количество документов в коллекции документов, $|\{d' \in D \,|\, t \in d'\}|$ — число документов, содержащих терм t.

Таким образом, терм будет иметь большой вес, если в некотором тексте он встречается часто, а в других — редко. С другой стороны, для распространенных термов веса будут небольшими.

Для моделирования коллекции документов вектора, соответствующие документам, группируют в матрицу так, что строка определяет терм, а каждый столбец соответствуют некоторому документу.

Последовательность N элементов (символов, термов, звуков, слогов) называется N-граммой. N-граммные модели используются для широкого круга исследований и разработок в области обработки естественного языка, как, например, распознавание речи, машинный перевод, извлечение информации.

1.3 Классификация

Задача классификация текстов широко изучается в таких областях знаний, как интеллектуальный анализ данных, машинное обучение и информационный поиск. Цель классификации заключается в присвоении текстовым документам меток определенных классов. Дан набор тренировочных данных $\mathcal{D} = \{d_1, \ldots, d_n\}$ такой, что для каждого документа d_i известна метка класса l_i — значение из множества $\mathcal{L} = \{l_1, \ldots, l_k\}$. Требуется найти модель классификации (классификатор) f, где

$$f : \mathcal{D} \to \mathcal{L}, \quad f(d) = l,$$

который мог бы присвоить правильную метку класса новому документу d (тестовый экземпляр). Обзор методов классификации приведен в [50], [76]. В статье [146] авторы оценивают различные виды классификаторов текстов. Многие алгоритмы классификации реализованы в различных программных системах и находятся в открытом доступе, как, например, BOW toolkit [101], Mallet [102].

Для оценивания качества модели классификации, случайная часть текстов откладывается (тестовый набор). После обучения на тренировочных данных, производится классификация текстов из тестового набора, сравниваются оценки меток классов с истинными метками. Для задач бинарной классификации, в которой метки принадлежат множеству $\{0, 1\}$, назовем объекты с меткой 1 положительными, а объекты с меткой 0 – отрицательными. Точность (precision) – это доля положительных объектов среди объектов, классифицированных алгоритмом как положительные. Полнота (recall) – доля правильно классифицированных объектов среди всех положительных объектов. F_1-мера – это геометрическое среднее точности и полноты

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall}.$$

1.3.1 Деревья решений

Дерево решений представляет собой иерархическую декомпозицию тренировочного множества данных, в которой использует предикат или условие на значение признака для иерархического разделения множества данных [112]. Разделение множества данных происходит рекурсивно в дереве решений, пока в листовых узлах не окажется минимальное количество элементов или пока не выполнятся условия любого другого критерия останова. Наиболее частотная метка класса в листовых узлах используется в модели классификации. Для тестового экземпляра применяется последовательность предикатов в узлах дерева с целью определить путь от корня к подходящему листу.

Для борьбы с переобучением некоторое множество листьев может быть удалено, для того чтобы отделить часть данных, не использующихся при построении дерева. Отделенный набор данных затем используется для определения следует ли удалять листовой узел или нет. В частности, если распределение класса в тренировочном наборе данных отличается от распределения классов в отделенном наборе данных, то предполагается, что листовой узел ведет к переобучению и должен быть удален. Подробный обзор и анализ алгоритмов на основе деревьев решений представлен в работах [52], [68], [105], [112].

В случае текстовых данных предикаты для деревьев решений определены в терминах слов исходной коллекции. Например, узел может быть разделен на два дочерних узла в зависимости от наличия или отсутствия одного или нескольких слов в тексте. В разных узлах на одном и том же уровне могут быть использованы разные слова в процессе разделения.

1.3.2 Байесовский классификатор

В вероятностном подходе к задаче классификации делаются предположения о том, как были сгенерированы данные (слова в документах) и предлагается вероятностная модель, основанная на этих предположениях. Используя тренировочные данные производится оценивание пара-

метров модели. Теорема Байеса используется при классификации новых элементов и при выборе класса, который с наибольшей вероятностью, сгенерировал этот элемент [103].

Наивный байесовский классификатор – один из самых простых, но в то же время популярных методов классификации. Моделирование распределения документов в каждом классе происходит в предположении, что разные термы появляются в тексте независимо друг от друга.

Существуют две основные модели, которые обычно используются для наивных классификаторов Байеса [103]. Обе модели стремятся найти апостериорную вероятность класса, основанную на распределении слов в документе. Единственное различие моделей в том, что одна учитывает частоты появления слов, в то время как вторая нет.

1. Модель многомерных испытаний Бернулли: в рамках модели документ представляется как вектор бинарных признаков, обозначающих наличие или отсутствие слов в документе. Таким образом, частота появления слов не учитывается. Подробное описание модели дано в работе [97].

2. Мультиномиальная модель: частоты слов (термов) в документе фиксируется, представляя документ как мешок слов. Различные вариации мультиномиальной модели описаны в [81], [104], [108], [112]. В работе [103] проведено сравнение между моделью многомерных испытаний Бернулли и мультиномиальной моделью. В результате сформулированы следующие выводы:

    - Если размер словаря небольшой, модель многомерных испытаний Бернулли может показывать лучшее качество.
    - Мультиномиальная модель всегда превосходит модель многомерных испытаний Бернулли для случая большого словаря и почти всегда превосходит, если размер словаря выбран оптимальным для каждой модели.

Обе модели предполагают, что документы сгенерированы параметризо-

ванной моделью смеси распределений. Приведем описание параметризованной модели, как она представлена в работе [103].

Модель смеси распределений включает компоненты смеси $c_j \in \mathcal{C} = \{c_1, \ldots, c_k\}$. Каждый документ $d_i = \{w_1, \ldots, w_{n_i}\}$ сгенерирован согласно априорному распределению $P(c_j|\theta)$. Таким образом можно вычислить правдоподобие документа как сумму вероятностей по всем компонентам

$$P(d_i|\theta) = \sum_{j=1}^{k} P(c_j|\theta)P(d_i|c_j;\theta).$$

Предполагается взаимно-однозначное соответствие между метками классов $\mathcal{L}\{l_1, \ldots, l_k\}$ и компонентами смеси. Таким образом, c_j определяет j-й компонент смеси и j-й класс. Пусть дан набор тренировочных данных, $\mathcal{D} = \{d_1, \ldots, d_{|\mathcal{D}|}\}$. Сначала оцениваются параметры вероятностной модели классификации $\hat{\theta}$. Далее, используя оценки этих параметров, производится классификация тестовых документов путем подсчета апостериорной вероятности каждого класса c_j, при условии тестового документа, и выбирается наиболее вероятный класс (класс с наибольшей вероятностью)

$$\begin{aligned} P(c_j|d_i;\hat{\theta}) &= \frac{P(c_j|\hat{\theta})P(d_i|c_j;\hat{\theta}_j)}{P(d_i|\hat{\theta})} = \\ &= \frac{P(c_j|\hat{\theta})P(w_1, w_2, \ldots, w_{n_i}|c_j;\hat{\theta}_j)}{\sum_{c\in\mathcal{C}} P(w_1, w_2, \ldots, w_{n_i}|c;\hat{\theta}_c)P(c|\hat{\theta})}. \end{aligned}$$

Согласно предположению, что слова в документе независимы получаем:

$$P(w_1, w_2, \ldots, w_{n_i}|c_j;\hat{\theta}_j) = \prod_{i=1}^{n_i} P(w_i|c_j;\hat{\theta}_j).$$

1.3.3 Линейный классификатор

Линейными классификаторами называются такие классификаторы, в которых результат линейного предсказателя имеет вид

$$\mathbf{p} = \mathbf{a} \cdot \mathbf{x} + \mathbf{b},$$

где $\mathbf{x} = \mathrm{col}(x_1, \ldots, x_n)$ – нормализованный вектор частот термов в документе, $\mathbf{a} = \mathrm{col}(a_1, \ldots, a_n)$ – вектор коэффициентов и $\mathbf{b}$ – скаляр. Такой вид предсказателя $\mathbf{p} = \mathbf{a} \cdot \mathbf{x} + \mathbf{b}$ естественно интерпретировать как разделяющую гиперплоскость между различными классами.

Метод опорных векторов был впервые предложен в [40], [140]. Основная идея алгоритма заключается нахождении оптимальной разделяющей гиперплоскости. Рассмотрим пример на Рис. 1.1. Два класса отмечены крестом и кружком, три разделяющие гиперплоскости A, B и C. Гиперплоскость A лучше всего разделяет классы, так как расстояние от любого объекта до нее наибольшее. Говорят, что гиперплоскость имеет наибольший зазор разделения. Вектор нормали к гиперплоскости указывает на направление в пространстве признаков, вдоль которого происходит максимальное различение.

Одним из преимуществ метода опорных векторов является его устойчивость к большой размерности, поскольку обучение происходит практически независимо от размерности признакового пространства. Отбор признаков редко требуется, так как для классификации выбираются элементы множества данных (опорные вектора). Как отмечено в работе [79], текстовые данные идеально подходят для этой модели классификатора из-за высокой размерности и разреженности данных. Метод опорных векторов популярен в приложениях распознавания образов, распознавания лиц, фильтрации спама [27], [49], [114]. Более глубокий анализ метода представлен в [78].

Регрессионные модели обычно применяются в задачах выявления зависимости между вещественнозначными признаками. Тем не менее бинарные значения меток классов можно рассматривать как частный слу-

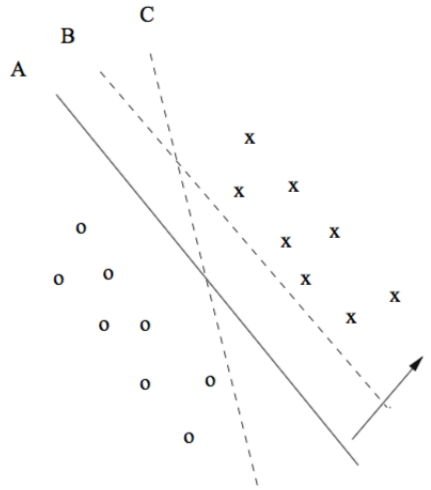

Рис. 1.1: Разделяющая плоскость.

чай веществennозначного признака и применять в задаче классификации регрессионные модели.

Одной из первых регрессионных моделей, примененной к задаче классификации текстов был метод наименьших квадратов [145]. Допустим, предсказанная метка класса $p_i = \mathbf{a} \cdot x_i + \mathbf{b}$. Истинная метка y_i известна и и требуется найти такие значения $\mathbf{a}$, b, которые минимизируют $\sum_{i=1}^{n}(p_i - y_i)^2$. Пусть $\mathbf{p}$ — $1 \times n$ вектор-индикатор бинарных значений классов, $\mathbf{b}$ равняется 0. Таким образом, если $\mathbf{x}$ — матрица $n \times d$ термдокумент, требуется найти такой вектор регрессионных коэффициентов $1 \times d$, который минимизирует $||\mathbf{a} \cdot \mathbf{x^T} - \mathbf{p}||$, где $||\cdot||$ — норма Фробениуса. Задачу можно обобщить на случай k классов, положив $\mathbf{p}$ — $k \times n$ матрица бинарных значений. В этой матрице в каждом столбце ровно одно значение 1, и соответствующий ей номер строки служит меткой класса для объекта. В работах [145], [146], [148] был проведен сравнительный анализ метода наименьших квадратов с множеством других методов классификации, и было показано, что метод наименьших квадратов оказывается достаточно робастным на практике.

Другим способом применения регрессионной модели к задаче классификации является логистическая регресия [111], где в качестве целевой функции, которую нужно оптимизировать, звыступает функция правдо-

подобия. Предполагается, что вероятность наблюдать метку y_i равняется

$$p(C = y_i|x_i) = \frac{\exp(\mathbf{a} \cdot x_i + \mathbf{b})}{1 + \exp(\mathbf{a} \cdot x_i + \mathbf{b})},$$

Или иначе

$$\log \frac{p(C = y_i|x_i)}{1 - p(C = y_i|x_i)} = \mathbf{a} \cdot x_i + \mathbf{b}.$$

Таким образом логистическая регрессия — это линейный классификатор, так как граница принятия решения является линейной функцией от признаков. В случае бинарной классификации значение $p(C = y_i|x_i)$ может быть использовано для определения метки класса (например, используя пороговое значение 0.5, если вероятность превышает порог, отнести объект к классу '1', в противном случае — к классу '0'). В случае многоклассовой классификации метка класса, имеющая наибольшее значение согласно $p(C = y_i|x_i)$, будет присвоена x_i. Пусть дан набор тренировочных данных $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, оценка параметров $\mathbf{a}$ происходит путем максимизации функции правдоподобия $\prod_{i=1}^{n} p(y_i|x_i)$.

1.3.4 Классификатор k ближайших соседей

Классификаторы на основе близости используют для классификации функции расстояния. Основная идея заключается в том, что документы, относящиеся к одному классу с большой вероятностью находятся близко друг другу согласно значению некоторой функции близости, например скалярного произведения или косинусной меры сходства [125]. Для того, чтобы классифицировать тестовый объект можно воспользоваться одним из следующих подходов:

- Определить k ближайших к тестовому объекту соседей в тренировочном наборе данных. Наиболее распространенный среди соседей класс возвращается как метка класса для тестового объекта. Примеры таких подходов изучены в [41], [66], [145]. Выбор k обычно варьируется от 20 до 40, в зависимости от размера словаря.

- На этапе предобработки данные в тренировочном наборе объединяются в группы документов, принадлежащих одному классу. Для каждой группы создается мета-документ, являющийся представителем класса. Подход k ближайших соседей, описанный выше, применяется к новому множеству мета-документов (обобщенных экземпляров [93]), а не к множеству исходных документов. Реферирование на этапе предобработки улучшает эффективность классификатора, так как сокращает количество вычислений расстояний. В случае, если в множестве данных большое количество выбросов, реферирование так же может повысить качество работы классификатора. В работах [67], [93], [120] описаны примеры таких подходов.

Как отмечено в статье [145], в классификаторах k ближайших соседей значительную роль играет отбор признаков и представление документов. В объемном корпусе документов большинство термов может не относиться к интересующей категории. Поэтому в [145] было предложен ряд методов выявления ассоциаций между словами и категориями. Далее эти ассоциации используются при построении признакового описания документа, так классификатор k ближайших соседей будет более чувствительным к классам в коллекции документов. Похожее наблюдение было сделано в статье [66], где было показано, что добавление весов термам (на основе их ассоциации с классом) повышает качество классификатора.

1.4 Кластеризация

Пусть $\mathcal{Z} = \{z^j\}_{j=1}^m$, $\rho(z, z')$ – метрика. Задача кластеризации заключается в нахождении разбиения множества $\mathcal{Z}$ на k кластеров таких, что

$$\mathcal{T}^k(\mathcal{Z}) = \{C_1, \ldots, C_k\},$$

$$\mathcal{Z} = \bigcup_{i=1}^{k} C_i, \quad C_i \cap C_j = \emptyset,\ i \neq j.$$

Для разбиения $\mathcal{T}^k(\mathcal{Z})$ функция $\gamma_{\mathcal{T}^k} : \mathcal{Z} \to \{1, \ldots, k\}$, соотносящая точки кластерам, определена следующим образом

$$\gamma_{\mathcal{T}^k}(z) = i \Leftrightarrow \in C_i, i = 1, \ldots, k.$$

Таким образом

$$C_i = \{z \in \mathcal{Z} | \gamma_{\mathcal{T}^k}(z) = i\}.$$

Для любого k для множества $\mathcal{Z}$ существуют различные разбиения $\mathcal{T}^k(\mathcal{Z})$.

Разбиение должно обладать следующим свойством: объекты, принадлежащие одному кластеру более "похожи" между собой, чем объекты, принадлежащие разным кластерам. Определим q_i – функцию "близости" к кластеру i, для любого $i = 1, \ldots, k$. Рассмотрим задачу минимизации

$$f(\mathcal{T}^k, z) = \sum_{i=1}^{k} \gamma_{\mathcal{T}^k}(z) q_i(\mathcal{T}^k, x) \to \min_{\mathcal{T}^k}. \tag{1.1}$$

Результат минимизации функции (1.1) зависит от z. Пусть вероятностное распределение $P(\cdot)$ определено на множестве $\mathcal{Z}$. Тогда можно рассматривать задачу минимизации функции качества

$$F(\mathcal{T}^k) = Ef(\mathcal{T}^k, z) = \sum_{i=1}^{k} \int_{C_i} q_i(\mathcal{T}^k, z) P(dz) \to \min_{\mathcal{T}^k} \tag{1.2}$$

В некоторых случаях можно ограничиться разбиением $\mathcal{T}^k$, которое полностью определяется множеством k векторов $c_1, \ldots, c_k \in \mathbb{R}^m$, которые формируют $m \times k$ матрицу $C = (c_1, \ldots, c_k)$ и для $i = 1, \ldots, k$ и $z \in \mathcal{Z}$ функции $q_(\cdot, z)$ зависят только от c_i, то есть $q_i(\cdot, \cdot) : \mathbb{R}^m \times \mathcal{Z} \to \mathbb{R}$. Правило разбиения можно задать следующим образом

$$C_i(\mathcal{Z}) = \{z \in \mathcal{Z} : q_i(c_i, z) < q_j(c_j, z),\ j = 1, \ldots, i-1 \\ q_i(c_i, z) \le q_j(c_j, z),\ j = i+1, \ldots, k\},\ i = 1, \ldots, k,$$

которое минимизирует (1.1). Вектора c_i, $i = 1, \ldots, k$ интерпретируются

как центры кластеров, когда $\mathcal{Z}$ — подмножество евклидова пространства $\mathbb{R}^m$. В этом случае функционал качества (1.2) принимает форму

$$F(\mathcal{T}^k) = \sum_{i=1}^{k} \int_{C_i} q_i(c_i, z) P(dz) \to \min_{\mathcal{T}^k}. \tag{1.3}$$

и может быть переписан в виде

$$F(C) = \int_{\mathcal{Z}} \langle l(C, z), q(C, z) \rangle P(dz) \to \min_{C}, \tag{1.4}$$

где $l(C, z)$ и $q(C, z)$ — вектора длины k такие, что первый состоит из значений характеристической функции $\mathbb{1}_{C_i(C)}(C, z)$, а второй из $q_i(c_i, z)$, $i = 1, \ldots, k$.

Такая формализация имеет простую геометрическую интерпретацию. Пусть распределение $P(\cdot)$ равномерно на $\mathcal{Z}$ и пусть функции

$$q_i(c_i, z) = ||c_i - z||^2, \ i = 1, \ldots, k$$

представляют расстояние до центров кластеров $c_1, c_2, \ldots, c_k$. Интеграл

$$\int_{C_i} ||c_i - z||^2 dz$$

определяет разброс точек x множества C_i. Функционал (1.4) принимает вид

$$F(C) = \sum_{i=1}^{l} \int_{C_i} ||c_i - z||^2 dz \to \min_{C}. \tag{1.5}$$

Таким образом, задача кластеризации свелась к задаче нахождения такого множества центров $\{c_1^*, \ldots, c_k^*\}$, для которых общий разброс точек минимален.

В области анализа текстов кластеризация может проходить на разных уровнях, в качестве кластеров могут выступать целые документы, абзацы, предложения или термы. Кластеризация активно применяется

в категоризации документов для улучшения поиска или просмотра. Например, в работе [42] авторы использовали алгоритмы кластеризации для составления оглавления большой коллекции документов, в [17] при помощи кластеризации строилась контекстная система информационного поиска.

Многие алгоритмы кластеризации можно применять к текстовым данным, используя, например, их векторное представление. Однако, текстовые данные имеют ряд особенностей представления:

1. Представление текста имеет высокую размерность, но сами данные разреженные. Другими словами, размер словаря коллекции может быть очень большим (например, порядка 10^5), но в отдельном документе встречаться всего несколько сотен слов.

2. Слова из словаря рассматриваемой коллекции документов обычно связаны между собой. Следует учитывать корреляцию между словами при разработке алгоритма кластеризации.

3. Документы в коллекции отличаются длиной (количеством встречающихся слов), важно производить нормализацию представлений текста.

Ряд алгоритмов, оптимизирующих представление текста, учитывая перечисленные выше характеристики, предложен в [124].

1.4.1 Иерархическая кластеризация

Название “иерархическая” объясняется тем, что в результате работы алгоритмов строится иерархия группы кластеров. Построение иерархии может происходить сверху-вниз (разделительная кластеризация) или снизу-вверх (агломеративная). Алгоритмы иерархической кластеризации относят к алгоритмам кластеризации, использующих функцию расстояния (похожести) $\rho(X, X')$ для определения близости текстовых документов. Обзор алгоритмов иерархической кластеризации представлен в [109], [110], [143].

При разделительном подходе один кластер, состоящий из всех документов коллекции, рекурсивно разделяется на под-кластеры. В агломеративном подходе, изначально каждый документ представляет отдельный кластер. Затем последовательно наиболее схожие кластеры объединяются, пока все документы не образуют единый кластер.

Для кластеров, состоящих из одного элемента определена функция расстояния

$$D(\{x\},\{x'\}) = \rho(x,x').$$

На каждой итерации слияния схожих кластеров вместо кластеров U и V образуется новый $W = U \cup V$. В 1967 году Лансом и Уильямсом [94] была предложена универсальная формула определения расстояния от нового кластера W до любого кластера S:

$$D(U \cup V, S) = \alpha_U D(U,S) + \alpha_V D(V,S) + \beta D(U,V) + \gamma |D(U,S) - D(V,S)|,$$

где $\alpha_U, \alpha_V, \beta, \gamma \in \mathbb{R}$ — параметры.

Наиболее распространены следующие три метода объединения кластеров в агломеративном подходе:

1. Метод одиночной связи (англ. single linkage)

$$D_{sl}(W,S) = \min_{w \in W, s \in S} \rho(w,s), \quad \alpha_U = \alpha_V = \frac{1}{2},\ \beta = 0,\ \gamma = -\frac{1}{2}.$$

2. Метод полной связи (англ. complete linkage)

$$D_{cl}(W,S) = \max_{w \in W, s \in S} \rho(w,s), \quad \alpha_U = \alpha_V = \frac{1}{2},\ \beta = 0,\ \gamma = \frac{1}{2}.$$

3. Метод средней связи (англ. group-average linkage)

$$D_{gal}(W,S) = \max_{w \in W, s \in S} \rho(w,s), \quad \alpha_U = \frac{|U|}{|W|} \alpha_V = \frac{|V|}{|W|},\ \beta = \gamma = 0.$$

1.4.2 Алгоритм k-средних

Алгоритм k-средних является одним из самых популярных методов кластеризации с заданным количеством кластеров. Принцип алгоритма основывается на поиске представителей кластеров (называемых центроидами), по одному для каждого кластера, и выборе разбиения на основе того, как кластеры рассеиваются вокруг этих точек. Таким образом кластеризация k-средних ищет разбиение, которое минимизирует функционал (1.5).

Приближенное решение этой задачи формулируется в виде следующего алгоритма:

Алгоритм 1 k-средних

Вход:

$\mathcal{Z}$ — множество

k — количество кластеров

$\widehat{\mathcal{T}}^k$ — начальное разбиение (необязательно)

Процедура:

1: *Инициализация:* Если начальное разбиение не задано, то сформировать начальное разбиение (обычно точки случайным образом относятся к кластерам).

2: *Минимизация:* Вычислить среднее (центроиду) точек каждого кластера.

3: *Классификация:* Отнести каждый элемент к текущей ближайшей центроиде.

4: Повторить 2-3 пока разбиение не стабилизируется, то есть центроиды перестанут изменяться.

Доказательство сходимости алгоритма кластеризации алгоритма k-средних требует проверки следующих двух утверждений:

- Переопределение точки к другому кластеру не увеличивает функцию ошибки.
- Пересчет центроида кластера не увеличивает функцию ошибки.

На практике элементы множества $\mathcal{Z}$ подаются на вход системе постепенно: $z^1, z^2, \ldots$. В этой связи итеративные алгоритмы, когда центроиды

пересчитываются одновременно с поступлением новых данных, гораздо более предпочтительны. Рандомизированный итеративный алгоритм k-средних был предложен в [4].

1.4.3 Тематическое моделирование

Задача тематического моделирования заключается в построении вероятностной порождающей модели корпуса текстовых документов. В рамках модели документы представляются как смеси тем, где тема — это вероятностное распределение слов.

Самые известные тематические модели — вероятностный латентно-семантический анализ (англ. Probabilistic Latent Semantic Analysis, PLSA) [70] и латентное размещение Дирихле (англ. Latent Dirichlet Allocation, LDA) [25]. Недостатком модели PLSA является невозможность вычислить вероятность документа, которого нет в коллекции текстовых документов. В [25] эта проблема была устранена введением априорного распределения Дирихле для тем в документах. Рассмотрим в этом разделе модель LDA.

Пусть $D = \{d_1, \ldots, d_{|D|}\}$ – коллекция документов и $V = \{w_1, \ldots, w_{|V|}\}$ – словарь коллекции. Тема z_j, $1 \leqslant j \leqslant K$ представляет собой мультиномиальное распределение $|V|$ слов, $p(w_i|z_j)$, $\sum_i^{|V|} p(w_i|z_j) = 1$. Модель LDA генерирует слова в два этапа: слова генерируются из тем, а темы из документов. Таким образом распределение слов в документе можно вычислить как:

$$p(w_i|d) = \sum_{j=1}^{K} p(w_i|z_j)p(z_j|d)$$

LDA предполагает следующий генеративный процесс для коллекции документов D:

1. Случайно выбрать для документа его распределение по темам $\theta \sim Dir(\alpha)$.

2. Для каждого слова в документе:

- Случайно выбрать тему из распределения θ_d, полученного на 1-м шаге.
- Случайно выбрать слово из распределения в выбранной теме z_i.

Совместное распределение модели (скрытых и наблюдаемых переменных) равняется

$$P(\phi_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}) = \prod_{j=1}^{K} P(\phi_j|\beta) \prod_{d=1}^{|D|} P(\theta_d|\alpha) \times \\ \times \left(\prod_{n=1}^{N} P(z_{d,n}|\theta_d) P(w_{d,n}|\phi_{1:K}, a_{d,n}) \right).$$

Далее для оценивания параметров необходимо вычислить апостериорное распределение скрытых переменных при условии наблюдаемых документов:

$$P(\phi_{1:K}, \theta_{1:D}, z_{1:D}|w_{1:D}) = \frac{P(\phi_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})}{P(w_{1:D})}.$$

Знаменатель дроби представляет собой вероятность наблюдать w при всех возможных параметрах модели и равняется сумме вероятностей совместного распределения по всем значениям скрытых переменных. Число всех возможных отнесений слов w к темам z растет экспоненциально с ростом длины документа. Поэтому на практике используют приближенные методы вывода апостериорной вероятности, например, вариационный вывод [25] или сэмплирование по Гиббсу [63].

Сэмплирование по Гиббсу вычисляет апостериорную вероятность для каждого слова следующим образом:

$$P(z_i = k|w_i = w, z_{-i}, w_{-i}, \alpha, \beta) = \frac{n_{k,-i}^{(d)} + \alpha}{\sum_{k'=1}^{K} n_{k',-i}^{(d)} + K\alpha} \times \frac{n_{w,-i}^{(k)} + \beta}{\sum_{w'=1}^{W} n_{w,-i}^{(k)} + W\beta},$$

где $z_i = k$ — назначение слова i теме k, z_{-i} — назначение всех остальных

слов к темам, $n_{w,-i}^{(k)}$ – количество раз, когда слово w было отнесено к теме k за исключением текущего отнесения. Аналогично, $n_{k,-i}^{(d)}$ – количество раз, когда тема k была отнесена к любому слову из документа d за исключением текущего отнесения. Теоретический обзор сэмплирования по Гиббсу представлен в [34], [62].

LDA часто используется как элемент более сложных моделей. В [37] авторы использовали LDA совместно с иерархией понятий для моделирования документов.

1.5 Меры сходства и различия

1.5.1 Определение мер сходства и различия и их свойства

Понятия сходства и различия широко используются в сфере искусственного интеллекта. Среди множества областей применения – интеллектуальный анализ данных, извлечение информации, распознавание образов, биоинформатика и нечеткая логика [23]. В широком смысле сходство и различие выражают результат сравнения двух объектов. Несмотря на интуитивность этих понятий, в литературе существует несколько способов их формализации.

Рассмотрим $\mathcal{X}$ — множество. Функция расстояния $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ удовлетворяет следующим свойствам [141]:

1. $d(x, x) = 0$ (тождественность.
2. $d(x, y) \geqslant 0$ (неотрицательность).
3. $d(x, y) = d(y, x)$ (симметричность).
4. $d(x, y) = 0 \implies x = y$. (определенность)
5. $d(x, y) + d(y, z) \geqslant d(x, z)$ (неравенство треугольника)

Если функция обладает только свойствами (1) и (2), то она называется функцией расстояния или функцией различия. Если функция различия удовлетворяет свойствам (1)–(3) и (5), то она называется полуметрикой, а если всем пяти свойствам, то метрикой. Пространство $(\mathcal{X}, d)$ называется пространством, снабженным полуметрикой, если d – полуметрика или метрическим пространством, если d – метрика. Принято формировать матрицу $D := (d(x_i, x_j)_{i,j-1,\ldots,n})$ из расстояний между объектами $x_1, \ldots, x_n \in \mathcal{X}$.

В контексте функций сходства рассмотрим следующие свойства [23]:

1. $s(x, x) > 0$.
2. $s(x, y) = s(y, x)$ (симметричность).
3. $s(x, x) \geqslant 0$ (неотрицательность).

Любую функцию, удовлетворяющую свойству (1) будем называть функцией сходства. Свойство (3) может выполняться не для всех функций сходства, например, оно не выполняется для коэффициентов корреляции или скалярного умножения.

Алгоритмы машинного обучения обычно используют либо функции сходства, либо функции различия. В общем случае выбор функции зависит от типа исследуемых данных, но может быть необходимо преобразовать функцию сходства в функцию различия и наоборот. Ниже представлено несколько способов такого преобразования.

- Если функция сходства является скалярным произведением в евклидовом пространстве, соответствующую функцию расстояния можно получить следующим образом:

$$d(x, y)^2 = \langle x - y, x - y \rangle = \langle x, x \rangle - 2\langle x, y \rangle + \langle y, y \rangle.$$

- Предположим, функция сходства нормирована, то есть $0 \leqslant s(x, y) \leqslant 1$ и $s(x, x) = 1$ для любых $x, y \in \mathcal{X}$. Тогда $d := 1 - s$ – функция расстояния.

- Для евклидовой функции расстояния соответствующая функция сходства может быть вычислена следующим образом:

$$s(x, y) := \frac{1}{2}(d(x, 0)^2 + d(y, 0)^2 - d(x, y)),$$

где 0 — некоторая произвольная точка в $\mathcal{X}$, играющая роль начала координат.

- Если d — функция различия, то неотрицательная убывающая функция от d будет являться функцией сходства. Например, $s(x, y) = \exp(-d(x, y)^2/t)$ при некотором параметре t или $s(x, y) = \frac{1}{1-d(x,y)}$

Далее приведены примеры наиболее известных функций расстояния (различия) и сходства. Пусть $x_i, x_j \in \mathcal{X}$, $P > 0$, $|\mathcal{X}| = N$. Обозначение x_{ik} означает k-й элемент x_i.

- Евклидово расстояние:

$$d_{Euclidean}(x_i, x_j) := \sqrt{\sum_{k=1}^{P}(x_{ik} - x_{jk})^2}.$$

- Взвешенное евклидово расстояние

$$d_{WEuclidean}(x_i, x_j) := \sqrt{\sum_{k=1}^{P} W_k(x_{ik} - x_{jk})^2},$$

где W_k обозначает вес k-го элемента вектора.

- Манхэттенское расстояние, расстояние городских кварталов

$$d_{Manhattan}(x_i, x_j) := \sum_{k=1}^{P} |x_{ik} - x_{jk}|.$$

- Расстояние Чебышёва

$$d_{Chebyshev} := max(|x_{i1} - x_{j1}|, \ldots, |x_{iP} - x_{jP}|).$$

- Расстояние Минковского

$$d_{Minkowski}(x_i, x_j) := \left(\sum_{k=1}^{P} |x_{ik} - x_{jk}|^l \right)^{\frac{1}{l}},$$

где $l \geqslant 1$.

- Расстояние Махаланобиса

$$d_{Mahalanobis} := \sqrt{\sum_{k=1}^{P} (x_i - x_j)^T \Sigma^{-1} (x_i - x_j)},$$

где Σ – матрица ковариации размера $P \times P$, ij-й элемент которой равен

$$\Sigma(i,j) := \frac{1}{N-1} \sum_{k=1}^{P} (x_{ik} - \bar{x_i})(x_{jk} - \bar{x_j}), \quad \bar{x} = \frac{1}{N} \sum_{k=1}^{N} x_{ik}.$$

- Корреляция Спирмена

Пусть $X = (x_1, \ldots, x_N)$, $Y = (y_1, \ldots, y_N)$

$$d_{Spearman} := 1 - \frac{6 \sum_{i=1}^{N} (R(x_i) - R(y_i))^2}{N(N^2 - 1)},$$

где $R(x_i), R(y_i)$ – ранги элементов x_i, y_i в последовательностях X, Y соответственно.

- Расстояние Канберра

$$d_{Canberra} := \sum_{i=1}^{P} \frac{|x_{ik} - x_{jk}|}{|x_{ik}| + |x_{jk}|}.$$

- Гауссовская ядерная функция сходства (радиальная базисная функция)

$$s_{Gaussian}(x_i, x_j) := a \exp(-\frac{||x_i - x_j||^2}{2\sigma^2}),$$

где σ — "масштабирующий" параметр.

- Гармоническое среднее

$$s_{Harmonic} := 2 \sum_{k=1}^{P} \frac{x_{ik} x_{jk}}{x_{ik} + x_{jk}}.$$

- Косинусная мера сходства

$$s_{Cosine} := \frac{\sum_{k=1}^{P} x_{ik} x_{jk}}{||x_i|| ||x_j||} = \frac{\langle x_i, x_j \rangle}{||x_i|| ||x_j||}.$$

- Корреляция Пирсона

$$\begin{aligned} s_{Pearson} :=& \frac{\sum_{k=1}^{P} (x_{ik} - \bar{x_i})(x_{jk} - \bar{x_j})}{||x_i - \bar{x_i}|| ||x_j - \bar{x_j}||} \\ =& \frac{\langle x_i - \bar{x_i}, x_j - \bar{x_j} \rangle}{||x_i - \bar{x_i}|| ||x_j - \bar{x_j}||} \\ =& S_{Cosine}(x_i - \bar{x_i}, x_j - \bar{x_j}). \end{aligned}$$

1.5.2 Ядерные функции и их свойства

В последнее время подход, основанный на ядерных функциях, стал распространенным и популярным методом в машинном обучении. Основной причиной такой популярности стал так называемый "Kernel trick", впервые предложенный в [2]. Идея заключается в переходе в пространство большей размерности, в котором нелинейная задача легко решается линейными методами. Исходное пространство объектов $\mathbb{X}$ вкладывается в пространство признаков $\mathbb{F}$

$$\phi : \mathbb{X} \to \mathbb{F}, \; x \to \phi(x),$$

в котором затем вычисляется скалярное произведение

$$K(x, x') = \langle \phi(x), \phi(x') \rangle. \tag{1.6}$$

Обычно, ядро $K(\cdot, \cdot)$ выбирается без какой-либо дополнительной информации об отображении ϕ, чтобы избежать расчетов в пространстве высокой размерности. Теорема Мерсера [127] утверждает, что любая непрерывная, симметричная, положительно определенная ядерная функция может быть представлена скалярным произведением в пространстве высокой размерности. Успех применения ядерных функций в методе опорных векторов [43] расширил применение этих функций на другие задачи машинного обучения: кластеризации (см. обзор в [55], [128]) и понижения размерности [106].

Далее приведем определения, свойства и известные результаты о вещественных положительно и отрицательно определенных ядрах.

Определение 1.

- *Симметричная функция* $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *называется положительно определенным ядром, если* $\forall n \geq 1$ *и для любых* $x, \ldots, x_n \in \mathcal{X}, c_1, \ldots, c_n \in \mathbb{R}$ *выполнено*

$$\sum_{i,j=1}^{n} c_i c_j K(x_i, x_j) \geqslant 0.$$

- *Симметричная функция* $N : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *называется отрицательно определенным ядром, если* $\forall n \geq 1$ *и для любых* $x, \ldots, x_n \in \mathcal{X}$ *выполнено*

$$\sum_{i,j=1}^{n} c_i c_j N(x_i, x_j) \leqslant 0$$

для $c_1, \ldots, c_n \in \mathbb{R}$ *таких, что* $\sum_{i=1}^{n} c_i = 0$.

Связь между положительно определенным и отрицательно определенным ядром выражается в следующих свойствах:

- Пусть $N(x, y)$ — отрицательно определенное ядро такое, что $N(z, z) = 0$ для некоторого $z \in \mathcal{X}$. Тогда

$$K(x, y) = N(x, z) + N(z, y) - N(x, y).$$

- Если $K(x, y)$ — положительно определенное ядро, то функция

$$N(x, y) = K(x, x) + K(y, y) - 2K(x, y)$$

является отрицательно определенным ядром, $N(x, x) = 0$ для любого $x \in \mathcal{X}$.

Следующие утверждения эквивалентны

- $K(x, y) = \exp(-tN(x, y))$ — положительно определенное ядро для любого $t > 0$.
- $N(x, y)$ — отрицательно определенное ядро.

Один из способов построить признаковое отображение, удовлетворяющее свойству (1.6), основано на следующей теореме.

Теорема. *Пусть $K(x, y)$ — положительно определенное ядро. Тогда существует единственное гильбертово пространство $\mathbb{H}$ вещественнозначных функций на $\mathcal{X}$ такое, что*

- *$k_y(\cdot) = K(\cdot, y) \in H$ для любого $y \in \mathcal{X}$,*
- *для любого $y \in \mathcal{X}$ и $f \in \mathbb{H}$ выполнено*

$$\langle f, k_y(\cdot) \rangle = f(y).$$

На основе этой теоремы признаковое отображение можно построить следующим образом

$$\phi : \mathcal{X} \to \mathbb{H}, \quad x \to k_x(\cdot).$$

В частности, по определению скалярного произведения

$$\langle k_x(\cdot), k_y(\cdot)\rangle = K(x, y)$$

для любых $x, y \in \mathcal{X}$. Таким образом положительно определенные ядра оказываются нелинейным обобщением функции схожести, порожденной скалярным произведением. Пространство $\mathbb{H}$ известно как воспроизводящее ядерное гильбертово пространство (англ. Reproducing Kernel Hilbert Space).

Теорема ([19]). *Пусть $N(x, y)$ отрицательно определенное ядро, такое, что $N(x, x) = 0$ для любого $x \in \mathcal{X}$. Тогда существует гильбертово пространство вещественнозначных функций на $\mathcal{X}$ и отображение $\phi : \mathcal{X} \to \mathbb{H}$ такое, что*

$$||\phi(x) - \phi(y)||^2 = N(x, y).$$

Теорема ([127]). *Сепарабельное метрическое пространство $(\mathcal{X}, \rho)$ может быть изометрически вложено в гильбертово пространство $\mathbb{H}$ тогда и только тогда, когда ρ^2 — отрицательно определенное ядро.*

Глава 2

Динамическая модель текстовых документов

В этой главе предложен один из возможных методов построения динамической модели текста. На основе предложенной динамической модели были разработаны и обоснованы два метода классификации документов и их фрагментов. Первый метод основан на кластеризации периодограмм, второй использует кластеризацию с помощью расстояния, основанного на некоторых ядрах. Сформулированы теоремы об однозначности и корректности построенных процедур классификации.

2.1 Динамическая модель текстовых документов

Пусть $\{X_i\}_{i=1}^n$ — множество текстовых документов. Под текстовым документом будет понимать упорядоченное множество символов.

$\forall i = 1, \ldots, n$ разделим документ X_i на m_i последовательных фрагментов:

$$X_i = x_i^1 + \ldots + x_i^{m_i}, \tag{2.1}$$

где "+" — операция конкатенации строк. Рассмотрим множество всех фрагментов $\overline{X} = \{x_i^j\}_{i \in 1..n, j \in 1..m_i}$.

Введем отображение V, которое сопоставляет фрагменту $x_i^j \in \overline{X}$ некоторое вероятностное распределение $P \in \mathcal{P}_M$ из множества вероятностных распределений на $\{1, \ldots, M\}$:

$$V : \overline{X} \to \mathcal{P}_M,$$

$$P \in \mathcal{P}_M : \quad \mathbf{P} = \{p_i\}_{i=1}^M, \quad p_i \geq 0, \ \sum_{i=1}^M p_i = 1.$$

Таким образом

$$\mathbf{x}_i^j = V(x_i^j) \in \mathbb{R}^M. \tag{2.2}$$

Обозначим $\mathcal{X} = \{\mathbf{x}_i^j\}_{i \in 1..n, j \in 1..m_i}$ — множество всех фрагментов в векторном представлении.

Значение параметра M определяется выбранной векторной моделью. Примеры распространенных векторных моделей приведены в п. 1.2.2. Пусть $\mathcal{V} = \{v_1, \ldots, v_A\}$ — множество всех термов в коллекции документов, называемое словарем. В случае модели "мешка слов" $M = |\mathcal{V}|$, текст представляется в виде распределения частот появления в нем всех термов из словаря. Модель ключевых слов является частным случаем предыдущей, текст представляется распределением частот появления слов из некоторого подмножества $\mathcal{V}' \subset \mathcal{V}$, таким образом $M = |\mathcal{V}'|$. В модели N-грамм, строится словарь всех N-грамм $\mathcal{V}_N$, встречающихся в документах из множества документов, в этом случае $M = |\mathcal{V}_N|$.

Будем считать, что на множестве $\mathbb{R}^M \times \mathbb{R}^M$ определена некоторая функция похожести двух фрагментов (см. п. 1.5.1)

$$r : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}. \tag{2.3}$$

Пусть $T > 0$. Для $i \in 1..n$, $j > T$, $\mathbf{x}_i^j \in \mathcal{X}$ обозначим через $\Delta_{x_i^j}$ множество предшествующих ему векторов-фрагментов: $\Delta_{\mathbf{x}_i^j} = \{\mathbf{x}_i^{j-T}, \ldots, \mathbf{x}_i^{j-1}\}$.

Каждая последовательность векторов-фрагментов $\Delta_{\mathbf{x}}$ с помощью опи-

санной выше функции (2.3) порождает функцию $s_{\mathbf{x}}(\cdot) : \mathbb{R}^M \to \mathbb{R}$:

$$s_{\mathbf{x}}(\mathbf{y}) = \frac{1}{T} \sum_{\mathbf{x}' \in \Delta_{\mathbf{x}}} r(\mathbf{x}', \mathbf{y}), \tag{2.4}$$

которую будем называть *динамической моделью*.

Значения функции $s_{\mathbf{x}}(\mathbf{y})$ соответствуют средней похожести вектора-фрагмента $\mathbf{y}$ с каждым из векторов-фрагментов из $\Delta_{\mathbf{x}}$.

Таким образом, введено отображение

$$\psi : \mathbf{x}_i^j \to s_{\mathbf{x}}(\cdot). \tag{2.5}$$

2.2 Паттерны динамической модели

2.2.1 Кластеризация спектральных представлений

Каждый документ из множества $\{X_i\}_{i=1}^n$ разделим на одинаковое количество последовательных фрагментов $\bar{m}$. Для каждого фрагмента получим его векторное представление согласно (2.2). Сопоставим документу последовательность векторов-фрагментов:

$$X_i \mapsto \{\mathbf{x}_i^j\}_{j \in 1..\bar{m}}. \tag{2.6}$$

Пусть $T > 0$. Для $j > T$, для каждого $\mathbf{x}_i^j \in \{\mathbf{x}_i^j\}_{j \in T+1..\bar{m}}$ построим динамическую модель $s_{\mathbf{x}_i^j}(\cdot)$. Рассмотрим последовательность выходов динамической модели:

$$\{\mathbf{x}_i^j\}_{j \in T+1..\bar{m}} \mapsto \{s_{\mathbf{x}_i^j}(\mathbf{x}_i^j)\}_{j \in T+1..\bar{m}}. \tag{2.7}$$

Последовательность (2.7) представляет собой временной ряд, соответствующий i-му документу.

Введем следующие обозначения:

- $\mathrm{s}_i^j = s_{\mathbf{x}_i^j}(\mathbf{x}_i^j)$, $j \in T+1..\bar{m}$, $i \in 1..n$ — средняя мера похожести фрагмента $\mathbf{x}_i^j$ и предшествующих ему фрагментов.

- $\mathcal{S}_i = \{\mathrm{s}_i^j\}_{i\in 1..n, j\in T+1..\bar{m}}$ — последовательность средних мер похожести, временной ряд.

- $\mathbb{S} = \{\mathcal{S}_i\}_{i\in 1..n}$ — множество последовательностей – временных рядов, соответствующих разным документам коллекции.

Периодограммой называется оценка спектральной плотности мощности сигнала, ее вычисление основано на подсчете коэффициентов преобразования Фурье с последующим усреднением [14].

Предположим сигнал $v(t)$ измерен в N точках:

$$\mathbf{v} = \{v_n,\ n = 0, 1, ..., N-1\}$$

с интервалом Δ. Для простоты положим N — четкое число. Дискретное преобразование Фурье $\mathbf{v}$ выражается как

$$V_k(\mathbf{x}) = \sum_{n=0}^{N-1} v_n w_n \exp\left(\frac{2\pi i n k}{N}\right),\ k = 0, ..., N-1, \tag{2.8}$$

где $i = \sqrt{-1}$, и w_n — функция окна. Периодограмма задана в $\left(\frac{N}{2}+1\right)$ частотах

$$f_k = \frac{k}{N\Delta} = 2f_c\frac{k}{N},\ k = 0, ..., \frac{N}{2},$$

где f_c — частота Найквиста, следующим образом:

- $P_{\mathbf{v}}(0) = P_v(f_0) = \frac{1}{W^2}\left|V_0(\mathbf{v})\right|^2.$

- $P_{\mathbf{v}}(k) = P_v(f_k) = \frac{1}{W^2}\left(\left|V_k(\mathbf{v})\right|^2 + \left|V_{N-k}(\mathbf{v})\right|^2\right),\ k = 1, ..., \left(\frac{N}{2}-1\right).$

- $P_{\mathbf{v}}(\frac{N}{2}) = P_{\mathbf{x}}(f_{\frac{N}{2}}) = \frac{1}{W^2}\left|V_{\frac{N}{2}}(\mathbf{x})\right|^2.$

для

$$W = N\sum_{n=0}^{N-1} w_n^2.$$

Для каждого временного ряда $\mathcal{S}_i$ вычислим его периодограмму.

$$\mathcal{S}_i \mapsto \mathrm{PG}(\mathcal{S}_i). \tag{2.9}$$

Обозначим $\mathbb{F} = \{\mathrm{PG}(\mathcal{S}_i)\}_{i\in 1..n}$ — множество всех периодограмм документов. Заметим, $\forall f \in \mathbb{F}\ f \in \mathbb{R}^{\bar{m}}$, будем называть $\mathbb{F}$ — пространством коэффициентов Фурье.

Будем кластеризовать элементы множества помощью алгоритма кластеризации Cl, минимизирующего функционал (1.5) из п.1.4.

Количество кластеров определяется значением индекса алгоритма валидации кластеризации (см., например, [30], [51], [53], [59], [69], [74], [91], [107], [137]).

Алгоритм 2

Вход:

X — множество текстов
T — параметр задержки
$k^\star$ — максимальное количество кластеров
Cl — алгоритм кластеризации
CLV — индекс алгоритма валидации кластеризации

Процедура:

1: Преобразовать документ $\mathcal{X}_i \in X$ во временной ряд $\mathcal{S}_i$ последовательно применив (2.6) и (2.7).
2: Для каждого временного ряда вычислить периодограмму $\mathrm{PG}(\mathcal{S}_i)$.
3: **for** $k = 2$ *to* k^* **do**
4: $\quad \mathcal{T} = Cl(\{\mathrm{PG}(S_i)\}_{i\in 1..n}, k)$;
5: $\quad ind_k = CLV(\mathcal{T})$;
6: **end for**
7: Количество кластеров соответствует оптимальному числу кластеров, согласно значению индекса ind_k $\{k = 2, .., k^*\}$.

Пусть в результате работы Алгоритма 2 периодограммы документов разделились на k кластеров $L_1, \ldots, L_k$. Тогда в пространстве временных рядов можно определить следующее правило классификации, относящее документ к одному из классов $l_1, \ldots, l_k$:

Правило классификации 1

Два документа X_i и X_j относятся к одному классу l_k, если соответствующие им периодограммы $\mathrm{PG}(\mathcal{S}_i)$ и $\mathrm{PG}(\mathcal{S}_j)$ попали в один кластер k.

Теорема 1. *Кластеризация в пространстве $\mathbb{F}$ обеспечивает однозначность и корректность правила классификации.*

Доказательство. По равенству Парсеваля [9] расстояния в $\mathbb{S}$ — пространстве последовательностей (временных рядов) и в пространстве коэффициентов преобразования Фурье $\mathbb{F}$ сохраняются. Следовательно, имеем взаимно-однозначное соответствие между кластерами в $\mathbb{S}$, пространстве последовательностей (временных рядов), и в пространстве периодограмм $\mathbb{F}$. Это обуславливает корректность построенной процедуры классификации. □

2.2.2 Кластеризация по расстояниям, основанным на ядрах

Каждый документ из множества $\{X_i\}_{i=1}^n$ разделим на последовательные фрагменты одинаковой длины. Аналогично п.2.2.1 для каждого фрагмента получим его векторное представление согласно (2.2). Сопоставим документу последовательность векторов-фрагментов:

$$X_i \mapsto \{\mathbf{x}_i^j\}_{j\in 1..m_i}.$$

Пусть $T > 0$, $\mathbb{X} = \{\mathbf{x}_i^j\}_{i\in 1..n, j\in T+1..\bar{m}}$ — множество векторов-фрагментов, для которых $j > T$, $\bar{m} = m_1 + \ldots + m_n$.

По формуле (2.4) $\forall \mathbf{x}_i^j$ построим динамическую модель:

$$\mathbf{x}_i^j \mapsto s_{\mathbf{x}_i^j}(\cdot).$$

Для строгого теоретического обоснования дальнейших выкладок предположим выполнение следующего условия для $s_{\mathbf{x}}(\cdot)$:

Предположение 1

$$s_{\mathbf{x}}(\mathbf{y}) \leq s_{\mathbf{x}}(\mathbf{x}) \quad \forall \mathbf{y} \in \mathbb{X}.$$

То есть каждый вектор-фрагмент наиболее тесно связан только со своими T предшественниками.

Введем функцию $D : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}$

$$D(\mathbf{x}, \mathbf{y}) = s_{\mathbf{x}}(\mathbf{x}) + s_{\mathbf{y}}(\mathbf{y}) - s_{\mathbf{x}}(\mathbf{y}) - s_{\mathbf{y}}(\mathbf{x}). \tag{2.10}$$

Произведем вложение пространства $(\mathbb{X}, D(\mathbf{x}, \mathbf{y}))$ в пространство R^m, где $m = m_1 + \ldots + m_n - nT = |\mathbb{X}|$:

$$F : (\mathbb{X}, D(\mathbf{x}, \mathbf{y})) \to (\mathbb{R}^m, \|\cdot\|).$$

Каждому вектору-фрагменту $\mathbf{x}_i^j$ сопоставим вектор $F \in R^m$ по следующему правилу:

$$F(\mathbf{x}_i^j) = \begin{pmatrix} D(\mathbf{x}_i^j, \mathbf{x}_1^{T+1}) \\ D(\mathbf{x}_i^j, \mathbf{x}_1^{T+2}) \\ \ldots \\ 0 \\ \ldots \\ D(\mathbf{x}_i^j, \mathbf{x}_n^{m_{n-1}}) \\ D(\mathbf{x}_i^j, \mathbf{x}_n^{m_n}) \end{pmatrix}. \tag{2.11}$$

Таким образом, $\forall j > T, i \in 1..n$ координаты вектора $F(\mathbf{x}_i^j)$ соответствуют расстояниям от вектора-фрагмента $\mathbf{x}_i^j$ до всех векторов-фрагментов из множества $\mathbb{X}$.

Рассмотрим пример вложения. Пусть $\mathbb{X} = \{\mathbf{x}_1^{t_1}, \mathbf{x}_2^{t_1}, \mathbf{x}_3^{t_1}\}$ и

- $D(\mathbf{x}_1^{t_1}, \mathbf{x}_2^{t_1}) = 0.5$,
- $D(\mathbf{x}_1^{t_1}, \mathbf{x}_3^{t_1}) = 1$,
- $D(\mathbf{x}_2^{t_1}, \mathbf{x}_3^{t_1}) = 0.2$.

Тогда соответствующие вектора F равны

$$F(\mathbf{x}_1^{t_1}) = \begin{pmatrix} 0 \\ 0.5 \\ 1 \end{pmatrix}, \; F(\mathbf{x}_2^{t_1}) = \begin{pmatrix} 0.5 \\ 0 \\ 0.2 \end{pmatrix}, \; F(\mathbf{x}_3^{t_1}) = \begin{pmatrix} 1 \\ 0.2 \\ 0 \end{pmatrix}.$$

Обозначим $\mathcal{F} = \{F(\mathbf{x}_i^j\}_{\mathbf{x}_i^j \in \mathbb{X}}$. Будем кластеризовать элементы множества $\mathcal{F}$ с помощью алгоритма кластеризации Cl, минимизирующего функционал (1.5) из п.1.4.

Алгоритм 3

Вход:

$\mathcal{X}$ — коллекция текстов
T — параметр задержки
k — число групп

Процедура:

1: Построить $\mathbb{X} = \{\mathbf{x}_i^j\}_{j=T+1}^m$.
2: Для каждого $\mathbf{x}$ построить динамическую модель $s_{\mathbf{x}}$ по (2.4).
3: Вычислить $F(\mathbf{x})$ для каждого $\mathbf{x}$ по (2.11).
4: Разделить множество $\mathcal{F}$ на k кластеров с помощью алгоритма кластеризации Cl.

Пусть в результате работы Алгоритма 3 вектора $F(\mathbf{x})$ разделились на k кластеров $L_1, \ldots, L_k$. Тогда в пространстве фрагментов можно определить следующее правило классификации, относящее фрагмент к одному из классов $l_1, \ldots, l_k$:

Правило классификации 2

Два фрагмента $\mathbf{x}_i$ и $\mathbf{x}_j$ относятся к одному классу l_k, если соответствующие им вектора $F(\mathbf{x}_i)$ и $F(\mathbf{x}_j)$ попали в один кластер k.

Теорема 2. *Если $r(\mathbf{x}, \mathbf{y})$ — положительно определенное ядра и выполнено Предположение 1 кластеризация в пространстве $\mathcal{F}$ обеспечивает однозначность и корректность правила классификации.*

Доказательство. Если $r(\mathbf{x}, \mathbf{y})$ — положительно определенное ядро, то при выполнении Предположения 1 функция $D(\mathbf{x}, \mathbf{y})$ является отрицательно определенным ядром, таким, что $D(\mathbf{x}, \mathbf{x}) = 0 \; \forall \mathbf{x} \in \mathbb{X}$.

Тогда по теореме из п. 1.5.2 существует гильбертово пространство $\mathbb{H}$ вещественнозначных функций на $\mathbb{X}$ и отображение $\phi : \mathbb{X} \to \mathbb{H}$ такое, что

$$||\phi(\mathbf{x}) - \phi(\mathbf{y})||^2 = D(\mathbf{x}, \mathbf{y}).$$

Полагая $\phi(\cdot) = F(\cdot)$ и $\mathbb{H} = \mathcal{F}$ имеем изометрическое вложение множества векторов-фрагментов $\mathbb{X}$ в множество векторов из $\mathcal{F}$. Следовательно, имеем соответствие между кластерами в пространстве векторов-фрагментов $\mathbb{X}$ и в пространстве векторов $\mathcal{F}$. Это и обуславливает корректность построенной процедуры классификации. □

Глава 3

Экспериментальные результаты

В главе приводятся результаты применения разработанных алгоритмов классификации фрагментов текстовых документов к задаче определения авторства текстов нескольких серий популярных книг. Полученные результаты указывают на то, что динамика изменений фрагментов текстовых документов является их отличительной характеристикой. Результаты хорошо согласуются с известными фактами.

3.1 Определение авторства текста

Задача определения авторства текста заключается в определение автора конкретного текста при помощи анализа текстовых документов, принадлежащим нескольким известным авторам. В этой сфере существует долгая история исследований. Исчерпывающие обзоры подходов даны в [80] и [134].

Для количественного определения различия двух текстов используют меры различия, которые является ключевой составляющей любого алгоритма определения авторства. Burrows's Delta [28] – одна из самых распространенных мер стилистического различия, была предложена в 2002 году. Различные ее модификации были разработаны и протестированы в работах [18], [71], [136].

Нормализованное расстояние сжатия (англ. Normalized Compression Distance) также успешно применяется в задачах кластеризации текстов и используется для оценки вычислительной сложности алгоритмов определения авторства (как, например, в [35], [113])

Некоторые подходы используют признаки на основе слов. Такого рода подходы можно разделить на три класса. В первом классе методов текст рассматривается как совокупность слов – служебных частей речи (например, предлоги, местоимения, союзы). При этом самостоятельные части речи игнорируются, так как они склонны быть сильно связанными с темой текста [149]. Второй класс подходов использует традиционную модель "мешка слов", выбирая в качестве признаков текста самостоятельные части речи [48]. Алгоритмы во втором классе основаны на предположении, что авторский стиль в большей степени определяется распределением вероятности появления слов, фраз или других синтаксических структур [100]. Они применимы в случае, когда есть явная связь между автором и темой текста. Стоит упомянуть среди них алгоритм локального дискриминативного тематического моделирования [144]. Последний класс методов рассматривает признаки на основе N-грамм — последовательностей N слов или символов [115]. Символьные N-граммы оказываются очень подходящими признаками для задачи определения авторства текстов. Они нечувствительны к грамматическим ошибкам, вычислительно эффективны и подходят для любых языков, так как их использование позволяет избежать сложной предобработки, например, токенизации для восточных языков. Ключевым аспектом этого подхода является правильный выбор длины N-граммы N. Большие значения N позволяют учесть контекст и тему текста, но также ведут к увеличению размерности признакового пространства. Меньшие значения N повышают чувствительность алгоритма к связям внутри слов, но не учитывают более широкий контекст. Для анализа синтаксической структуры, являющейся подходящей характеристикой авторского стиля, в работах [130], [131], [132] были представлены синтаксические N-граммы.

Гибридные методы сочетают несколько видов признаков (например,

[44]), таким образом используя стилистические и тематические признаки одновременно. Как было подчеркнуто в [123], не существует универсального признака, который позволит точно отличить разные авторские стили. Таким образом, необходимо анализировать необычайно широкий набор признаков с привлечением разных подходов [80] для того, чтобы получить подходящий результат. В задаче верификации авторства, авторский стиль становится самым важным признаком рассматриваемого текста [85], [134]. При верификации множество авторов-кандидатов состоит из одного автора [99]. Так как любую задачу идентификации автора можно привести к последовательности из задач верификации, последнюю считают фундаментальной [90], [133].

Существует два основных подхода к решению задачи верификации: внутренний и внешний. Внутренний подход работает только с предоставленными текстами (один известного авторства и один исследуемый, авторство которого под вопросом) и сводится к задаче одноклассовой классификации [57], [65], [77]. Такая задача возникает и при выявлении плагиата (например, [86], [96], [135], [147]). Внешние подходы преобразуют задачу верификации в задачу бинарной классификации. Среди алгоритмов такого рода стоит упомянуть “метод самозванцев” (Impostors Method) [89]. Решение об авторстве принимается на основании того, что документ известного авторства более похож на исследуемый текст по сравнению с текстами из множества “самозванцев”. Несмотря на то, что метод эффективен в целом, ее применимость имеет ряд ограничений. Например, может быть проблематично классифицировать пары “один автор” и “разные авторы”, когда исследуемые тексты разного жанра и тематики [90].

3.2 Классификация текстов на основе алгоритма кластеризации с помощью спектрального представления

В этом разделе приведены результаты применения Алгоритма 2 и Правила классификации 1 п. 2.2.1 к анализу авторского стиля в двух книжных циклах: “Основание” А. Азимова и “Рама” А. Кларка (произведения на английском языке).

Предобработка текстовых документов заключалась в удалении стоп-слов. Каждый документ делился на одинаковое количество отрывков $m = 256$. В качестве векторной модели была выбрана модель символьных 3-грамм. По коллекции текстовых документов строился словарь всех встречающихся в текстах 3-грамм, далее каждый фрагмент текста представлялся как распределение частот появления в нем 3-грамм из словаря. В качестве функции $r(\mathbf{x}, \mathbf{y})$ было выбрано $d_{Spearman}(\mathbf{x}, \mathbf{y})$ (см. п. 1.5.1), параметр $T = 10$ — учитывались 10 предшествующих фрагментов. Разделение периодограмм на кластеры происходило агломеративном иерархическим алгоритмом кластеризации с методом одиночной связи (см. п. 1.4.1), выбор оптимального количества кластеров — с помощью индекса Silhouette [122]. Значение индекса измеряет, как точки в одном кластере ближе друг к другу, в отличие от точек, попавших в другие кластеры. Точки с большими положительными значениями индекса около $+1$ хорошо сгруппированы. Точки, у которых отрицательные значения индекса, находятся в неправильном кластере. Среднее значение индекса, вычисленное для всех точек, характеризует качество разбиения на кластеры. Оптимальным считается разбиение с наибольшим значением индекса, так как оно обеспечивает с одной стороны, компактность кластеров, а другой — хорошо отделяет кластеры друг от друга.

Цикл “Основание” А.Азимова

Книжная серия “Основание” представляет собой цикл из 7 научно-фантастических романов А.Азимова. Первоначально цикл состоял из трех романов, спустя тридцать лет были написаны еще четыре книги. Ниже представлен список книг согласно внутренней хронологии событий в произведениях:

1. “Прелюдия к Основанию” (1988) (F1)
2. “На пути к Основанию” (1993) (F2)
3. “Основание” (1951) (F3)
4. “Основание и Империя” (1952) (F4)
5. “Второе Основание” (1953) (F5)
6. “Кризис Основания” (1982) (F6)
7. “Основание и Земля” (1986) (F7)

Рассмотрим две периодограммы, построенные для первой напечатанной книги “Основание” (красный) и последней напечатанной книгой “На пути к Основанию” Рис. 3.1.

Нестрого говоря, эти кривые похожи, но спектр мощности красной линии больше сосредоточен в низких частотах. Она имеет больше пиков, в то время как синяя — более гладкая. Можно считать, что со временем стиль “размывался”, в то время как стиль первой книги был “четким”. Результаты иерархической кластеризации на Рис. 3.2 так же указывают на изменение стиля.

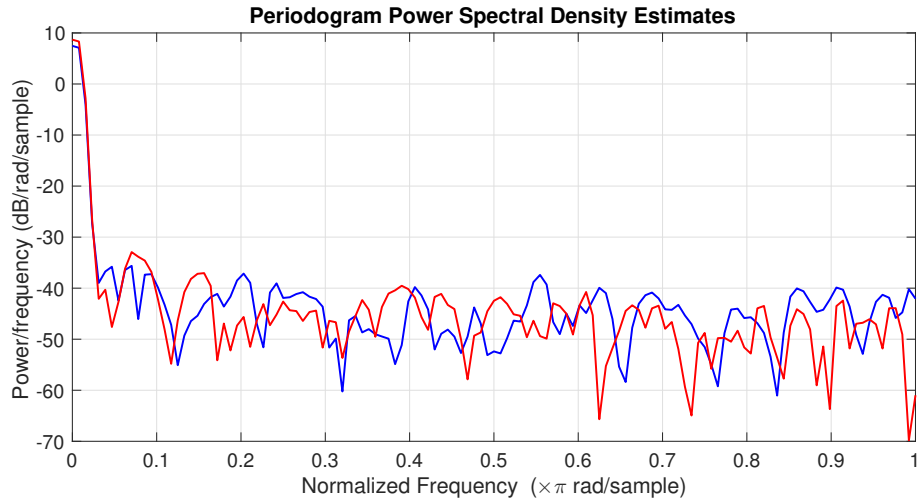

Рис. 3.1: Периодограммы, построенные для первой изданной (красный) и последней изданной книги (синий).

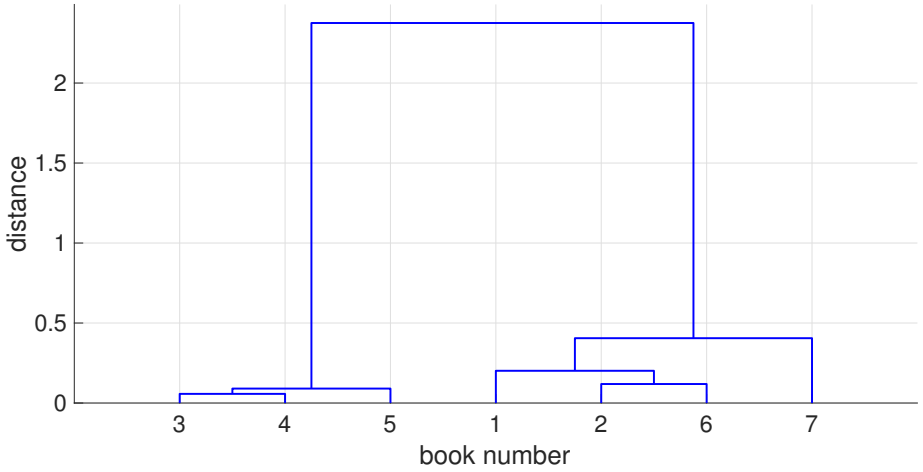

Рис. 3.2: Иерархическая кластеризации книг из цикла “Основание”.

Здесь точно обнаружены два кластера, отличающиеся по внутреннему стилю. В первый кластер попали три романа: “Основание”, “Основание и Империя”, “Второе основание”, которые были опубликованы в 1951, 1952 и 1953 годах. Другие четыре книги опубликованы в 1988, 1993, 1982 и 1986 годах, они расположены во второй группе. По всей видимости, такой большой перерыв (около 30 лет) между интервалами работы над книгами стал причиной изменения стиля в цикле.

Цикл “Рама” А.Кларка

1. “Свидание с Рамой” (1972) (R1)
2. “Рама II” (1989) (R2)
3. “Сад Рамы” (1991) (R3)
4. “Рама явленный” (1993) (R4)
5. “Яркие посланники” (1995) (R5)
6. “Двойное полнолуние” (1999) (R6)

Книжная серия “Рама” состоит из шести научно-фантастических романов (обозначены как ($R1-R6$)), началась в 1973 году с романа “Свидание с Рамой” Артура Кларка.

Следующая таблица содержит попарные расстояния между периодограммами в цикле произведений.

	$R1$	$R2$	$R3$	$R4$	$R5$	$R6$
$R1$	0.00	2.28	2.11	2.65	1.92	1.72
$R2$	2.28	0.00	0.18	0.37	0.35	0.56
$R3$	2.11	0.18	0.00	0.54	0.19	0.39
$R4$	2.65	0.37	0.54	0.00	0.73	0.93
$R5$	1.92	0.35	0.19	0.73	0.00	0.20
$R6$	1.72	0.56	0.39	0.93	0.20	0.00

Таблица 3.1: Расстояние между периодограммами в серии “Рама”.

Максимальное значение расстояния (2.65) достигается для книг $R1$ и $R4$. Соответствующие периодограммы представлены на Рис. 3.3 и обозначены красным и синим цветом соответственно.

Как и в рассмотренном цикле “Основание”, последняя изданная книга обладала более “гладким” стилем.

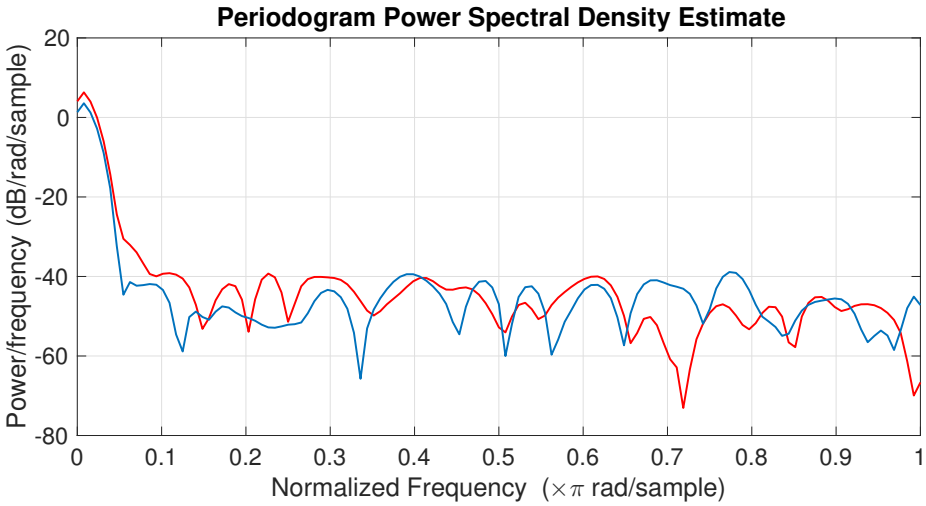

Рис. 3.3: Периодограммы, построенные для $R1$ и $R4$.

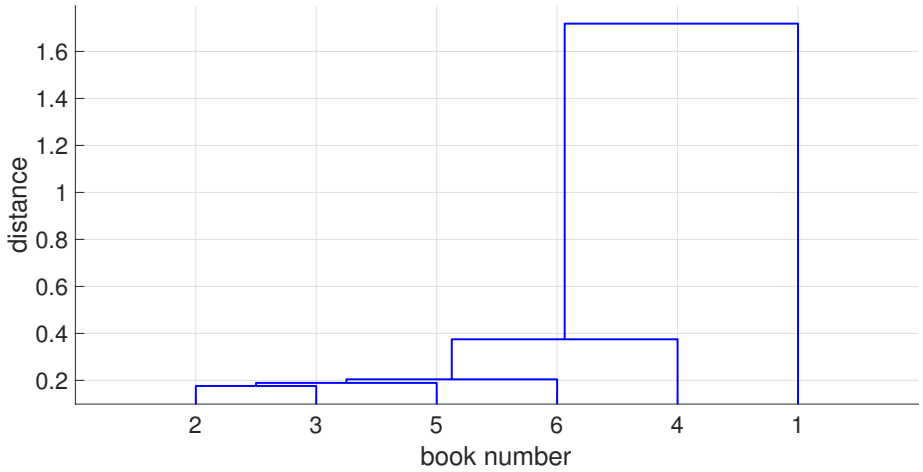

Рис. 3.4: Иерархическая кластеризация книг из цикла “Рама”.

Иерархическая кластеризация цикла (Рис. 3.4) чётко показывает структуру коллекции. С самого начала, только первая книга “Свидание с Рамой” была написана Кларком в одиночку. Он объединился с Джентри Ли для работы над следующими тремя романами “Рама-2”, “Сады Рамы” и “Рама явленный”. Существует распространенная точка зрения, что Ли занимался непосредственно сочинением, в то время как Кларк по большей части редактировал материал. Это объясняет, почему фокус и стиль совместно написанных романов ($R2 - R4$) настолько отличается от первоначального.

3.3 Классификация текстов на основе алгоритма кластеризации с помощью расстояний на ядрах

В этом разделе представлены результаты классификации текстов из п. 3.2 на основе Алгоритма 3 и Правила классификации 2, описанных в п. 2.2.2.

Изучаемые тексты делились на фрагменты одинаковой длины L (в экспериментах было выбрано $L = 2000$). В качестве векторной модели была выбрана модель служебных слов. Служебными словами называются вспомогательные части речи, такие, как предлоги, частицы, союзы, артикли. Такие слова можно рассматривать как некоторый стилистический "клей" языка, связывающий другие, более "содержательные" части речи. Частота появления в тексте служебных слов может представлять важную стилистическую характеристику, независящую от тематики текста. Каждый фрагмент текста представлялся в виде распределения частот появления в нем служебных слов из списка (307 слов для английского языка). Проводились эксперименты со двумя функциями $r(\mathbf{x}, \mathbf{y})$: $d_{Spearman}$ и $d_{Canberra}$ (см. п. 1.5.1). Параметр задержки $T = 10$.

Алгоритм 3 из Главы 2 можно применить к задаче верификации авторства следующим образом. Допустим даны два текста D_1 и D_2, которые нужно проверить на принадлежность одному автору. Разделим документы на фрагменты и классифицируем их на два класса, используя Алгоритм 3. Решение принимается в соответствии с отнесением к классу по правилу классификации 2.

Алгоритм 4 Алгоритм верификации авторства

Вход:

D_1 и D_2 – два текста для сравнения
T - параметр задержки
L - длина отрывков

Процедура:

1: Применить процедуру из Алг. 3 для текстов D_1 и D_2 с $k = 2$.
2: По правилу классификации отнести фрагменты к соответствующим классам. Текст попадает в тот класс, к которому относятся большинство его фрагментов.
3: Если D_1 и D_2 относятся к одному классу, то они написаны одним автором, в противном случае — разными.

Далее представлен общий алгоритм, обобщающий описанную методологию. Алгоритм кластеризации 5 разработан для идентификации автора в случае, когда исследуемый текст D_0 следует проверить относительно его предполагаемого авторства в коллекции:

$$\mathcal{D}_n = \{D_1, \ldots, D_n\}.$$

Основная идея та же: исследуемый текст и тренировочная коллекция разделяются на фрагменты, которые классифицируются. Рассматриваемый документ относится к тому классу, к которому отнесены большинство его фрагментов.

Число различных авторов обычно заранее известно и предполагается, что оно совпадает с количеством разных стилей (C_s) в коллекции. Заметим, что наличие C_s различных стилей в исходной коллекции должно быть гарантировано, так как несколько авторов могут писать тексты в соавторстве или длина отрывка L может быть выбрана так, что стили не могут быть разделены при выбранных параметрах. Для того, чтобы оценить разделение стилей в полученном результате алгоритма кластеризации воспользуемся скорректированным индексом Ранда [73].

Первоначально индекс Ранда [119] вводился для задач классификации, в которых результаты группировки сравнивались с “истинным” разделением на классы. Значения индекса находятся в отрезке от 0 до 1.

Значение 0 означает полное несоответствие двух разделений объектов. Если оба разделения совпадают, тогда индекс Ранда равен 1. Основным недостатком индекса является то, что его математическое ожидание для двух случайных разделений не постоянно.

Скорректированный индекс Ранда (Adjusted Rand Index — ARI) основан на обобщенном гипергеометрическом распределении, таком, что разделения строятся случайно с фиксированным числом элементов в каждом кластере. Математическое ожидание индекса равно 0 для независимых разделений, а его максимальное значение равно 1 для одинаковых разделений.

Рассмотрим следующие два разделения коллекции документов:

1. Разделение документов согласно определенным заранее стилям

$$\mathcal{D}_n = \bigcup_{m=1}^{C_s} C_{sm}\left(\mathcal{D}_n\right),$$

   где $C_{sm}\left(\mathcal{D}_{\mathfrak{n}}\right)$ состоит из всех документов, имеющих стиль C_{sm}, $m = 1, ..., S$.

2. Разделение $Cl_{(C_s)}$ документов на $C - s$ классов, полученное в результате Алгоритма 3 из п. 2.2.2:

$$\mathcal{D}_n = \bigcup_{i=1}^{C_s} Cl_{(C_s),i}\left(\mathcal{D}_n\right).$$

Составим таблицу сопряженности из всех значений

$$n_{si} = \left|C_{sm}\left(\mathcal{D}_{\mathfrak{n}}\right) \cap Cl_{(C_s),i}\left(\mathcal{D}_{\mathfrak{n}}\right)\right|,\ s,\ i = 1, ..., C_s,$$

и введем:

$$n_{s\cdot} = \sum_{i=1}^{C_s} n_{ci},\ n_{\cdot i} = \sum_{c=1}^{C_s} n_{ci}.$$

Скорректированный индекс Ранда (англ. Adjusted Rand index — ARI):

$$\mathrm{ARI}\left(Cl_{(C_s)}\right) = \frac{\sum\limits_{c,i}^{C_s}\binom{n_{ci}}{2} - \sum\limits_{c=1}^{C_s}\binom{n_{c\cdot}}{2}\sum\limits_{i=1}^{C_s}\binom{n_{\cdot i}}{2}/\binom{n}{2}}{\frac{1}{2}\left(\sum\limits_{c=1}^{C_s}\binom{n_{c\cdot}}{2} + \sum\limits_{i=1}^{C_s}\binom{n_{\cdot i}}{2}\right) - \sum\limits_{c=1}^{C_s}\binom{n_{c\cdot}}{2}\sum\limits_{i=1}^{C_s}\binom{n_{\cdot i}}{2}/\binom{n}{2}}.$$

Будем считать разделение $Cl_{(C_s)}$ состоятельным, если $\mathrm{ARI}\left(Cl_{(C_s)}\right) > C_{Rand}$, где C_{Rand} – заданное пороговое значение.

Алгоритм 5 Алгоритм идентификации авторства

Вход:

$\mathcal{D}_0$ — исследуемый документ

C_s — количество различных стилей в коллекции

$\mathcal{D}_n = \{D_1, \ldots, D_n\}$ — документы, отнесенные к S авторским стилям

Процедура:

Выбрать T, L, C_{Rand} — уровень значимости для скорректированного индекса Ранда

Составить новую коллекцию $\mathcal{D}_{n+1} = \{\mathcal{D}_i,\ i = 0, \ldots, n\}$.

if $C_s = 1$ **then**

    Вызвать Алг. 4 с входными данными $\mathcal{D}_{n+1}$ для сравнения стилей $\mathcal{D}_0$ и $\mathcal{D}_n$.

    STOP

else

    Вызвать Алг. 3 и получить разделение $Cl_{(C_s)}(\mathcal{D}_{n+1})$.

    Составить $Cl_{(C_s)}(\mathcal{D}_n$ из $Cl_{(C_s)}(\mathcal{D}_{n+1})$ и подсчитать $\mathrm{ARI}\left(Cl_{(C_s)}(\mathcal{D}_n)\right)$.

    if $\mathrm{ARI}\left(Cl_{(C_s)}(\mathcal{D}_n\right) \le C_{Rand}$ **then**

        Следует переопределить параметры алгоритма.

        Сообщение *“Коллекция не разделена”*.

        STOP

    else

        Отнести $\mathcal{D}_0$ к стилю, который наиболее частотен среди в соответствующем кластере.

    end if

end if

Заметим, что здесь вводится новый параметр C_{Rand}, используемый в алгоритме. Его цель заключается в оценке способности процедуры кластеризации разделить тренировочное множество. Если значение ARI, подсчитанное для исходной коллекции не превышает выбранного порога C_{Rand}, то невозможно предполагать, что тренировочная коллекция является надежной для текущего набора параметров.

В рассматриваемой задаче, неинформативные термины появляются в меньшинстве отрывков с относительно невысокой частотой. Таким образом, алгоритмы разделения нечувствительны к наличию таких терминов в конкретном отрывке, так как их частота является низкой для всех рассматриваемых отрывков. Естественно, число таких терминов может увеличиваться с уменьшением длины фрагмента L. Оценим информативность термина на основе его средней частоте появление во всей коллекции:

$$I(w_i) = \text{average}\left\{f\left(w_i, \mathcal{D}\right),\ \mathcal{D} \in \mathcal{D}\right\},$$

где $f\left(w_i, \mathcal{D}\right)$ — частота появления термина w_i в тексте $\mathcal{D} \in \mathcal{D}$. На следующем шаге только термины из множества

$$IW(T) = \left\{I(w_i) > Tr\right\}, \tag{3.1}$$

включены в построение векторной модели. Здесь Tr — определенное заранее пороговое значение.

Ключевыми параметрами в представленной методологии являются задержка T и длина фрагментов L. Задача выбора подходящего набора параметров является плохо обусловленной, так как разные комбинации параметров могут приводить к одинаковому поведению системы. Ясно, что большие значения T и L должны приводить к более стабильным результатам. Но, с другой стороны, количество отрывков текстов может уменьшиться до такой степени, что динамическая модель более не будет отражать динамику стиля и классификатор перестанет быть надежным. В связи с этим, необходимо сохранить баланс между значениями параметров и количеством отрывков при выборе конфигурации параметров.

По аналогии с [29], предлагается искать такие значения параметров, которые обеспечивают правильное разделение коллекций текстов, относящихся к различным стилям. Это идея воплощена в Алгоритме 6.

Рассмотрим две коллекции текстов с разными авторскими стилями

$$\mathcal{D}_{n_1} = \{D_{11}, \ldots, D_{1n_1}\}\,,\ \mathcal{D}_{n_2} = \{D_{21}, \ldots, D_{2n_2}\}$$

с двумя наборами возможных параметров

$$\mathcal{T} = \{T_1, \ldots, T_m\}\,,\ \mathcal{L} = \{L_1, ..., L_k\}\,.$$

Эти наборы параметров могут быть получены из предыдущих экспериментов или согласно общим представлениям. Процедура повторяется несколько раз (параметр $Iter$ в алгоритме).

Для каждого набора $T \in \mathfrak{T}$ и $L \in \mathfrak{L}$ два текста $D_{1j_1} \in \mathcal{D}_{n_1}$ и $D_{2j_2} \in \mathcal{D}_{n_2}$ выбираются случайным образом, разделяются на отрывки и кластеризуются при помощи Алг. 3, с целью определить индекс ARI между первоначальным и полученным разделением. После завершения итераций, когда среднее значение $A_0(T, L)$ индекса ARI найдено, следующее значение достигается для каждого $T \in \mathfrak{T}$

$$L^*(T) = \arg\min_{L \in \mathcal{L}} \{A_0(T, L)\ > C_{Rand}\}$$

и

$$T^* = \arg\min_{T \in \mathcal{T}} \{L^*(T)\}\,.$$

Здесь, C_{Rand} — определенный заранее порог. Пара $(T^*, L^*(T^*))$ — выбранная конфигурация.

Для определения подходящих значений для параметров использовался Алгоритм 6. В описанном следующие циклы книг использовались как заведомо различные наборы текстов:

- Цикл “Основание” А.Азимова,
- Цикл “Рама” А.Кларка

Алгоритм 6 Подбор параметров

Вход:

Две коллекции документов, относящиеся к двум разным стилям $\mathcal{D}_{n_1} = \{D_{11}, \ldots, D_{1n_1}\}$ и $\mathcal{D}_{n_2} = \{D_{21}, \ldots, D_{2n_2}\}$.

Процедура:

1: Выбрать C_{Rand} — уровень значимости для скорректированного индекса Ранда.

2: Выбрать $\mathcal{T} = \{T_1, ..., T_m\}$ — множество тестируемых значений T.

3: Выбрать $\mathcal{L} = \{L_1, ..., L_k\}$ — множество тестируемых значений L.

4: Выбрать $Iter$ — количество итераций.
5: **for** $T \in \mathcal{T}$ **do**
6: **for** $L \in \mathcal{L}$ **do**
7: **for** $i = 1 : Iter$ **do**

8: Случайным образом выбрать $D_{1j_1} \in \mathcal{D}_{n_1}$ и $D_{2j_2} \in \mathcal{D}_{n_2}$.

9: Составить $\mathcal{D}_{\mathfrak{i}} = \{D_{1j_1} \in \mathcal{D}_{n_1}, D_{2j_2} \in \mathcal{D}_{n_2}\}$.

10: Вызвать Алгоритм 3 и получить разбиение $Cl_2(\mathcal{D}_i)$.

11: Вычислить $R_i = \mathrm{ARI}\left(Cl_2(\mathcal{D}_i)\right)$.
12: **end for**

13: Вычислить $A_0(t, l) = \mathrm{mean}(R_i | i = 1, ..., Iter)$.
14: **end for**
15: **end for**

16: Для каждого $T \in \mathcal{T}$ найти

$$L^*(T) = \arg\min_{L \in \mathcal{L}} \left\{A_0(T, L) \ > C_{Rand}\right\}.$$

17: Найти

$$T^* = \arg\min_{T \in \mathcal{T}} \left\{L^*(T)\right\}.$$

18: Пара $(T^*, L^*(T^*))$ — выбранная конфигурация системы.

В этих наборах семь и шесть книг соответственно.

Каждая книга из первого набора сравнивалась с каждой книгой в другом наборе (всего 42 сравнения). Такой подход немного отличается от Алгоритма 6, где сравнивались произвольно выбранные документы. Проверялись три значения для параметра задержки с 10 последовательными значениями для размера отрывка $\mathcal{L} = \{500, 1000, ..., 5000\}$. На Рис. 3.5 представлены три графика скорректированного индекса Ранда, подсчитанного для выбранных значений T с использованием расстояния $d_{Spearman}$.

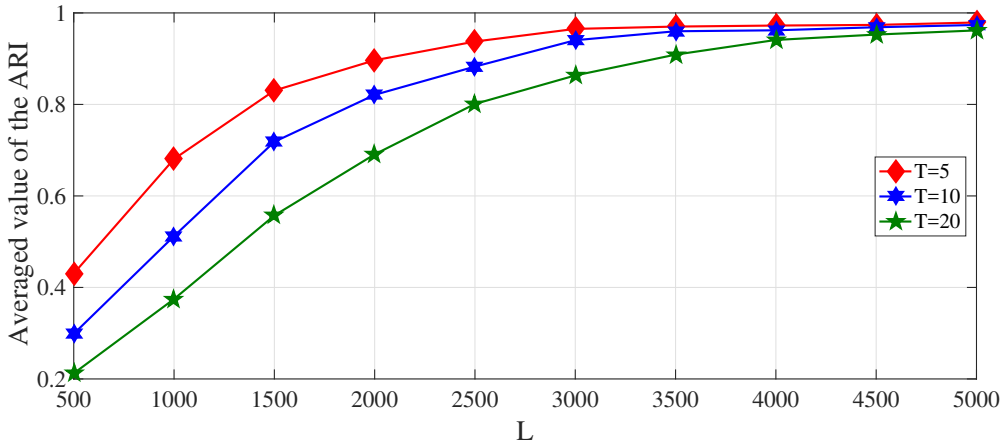

Рис. 3.5: Усредненные значения скорректированного индекса Ранда для расстояния $d_{Spearman}$.

Результаты полученные с помощью расстояния $d_{Canberra}$ очень похожи (см. Рис. 3.6).

Полагая $C_{Rand} = 0.9$ в алгоритме 6, получаем $L^*(T) = 2500, 2000, 1500$. Так, $T^* = 20$ и $L^*(T^*) = 2000$.

Таким образом, в экспериментах использовались следующие параметры:

- $T = 20$,

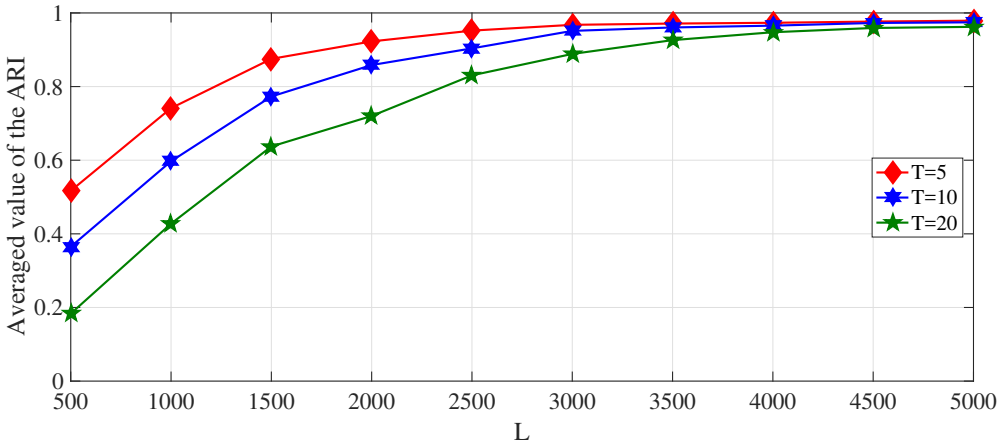

Рис. 3.6: Усредненные значения скорректированного индекса Ранда для расстояния $d_{Canberra}$.

- $L = 2000$,
- $Tr = 0.5$.

Двухступенчатый анализ кластеров — это масштабируемая методология, созданная для обработки больших наборов данных [38]. Общий подход состоит из двух основных шагов (ступеней). В начале, применяется разделяющий алгоритм, такой как K-средних, для того, чтобы сформировать небольшие так называемые “пред-кластеры”. Количество кластеров может быть заранее определено или оценено с помощью метода валидации кластеризации. Ожидается, что полученные кластеры достаточно консистентны, но не слишком малы, поскольку они будут использоваться на следующем шаге как отдельные наблюдения. После, процедура иерархической агломеративной кластеризации последовательно объединяет “пред-кластеры” в однородные группы. Агломеративнная иерархическая процедура начинает с одиночных кластеров и собирает их в группы, пока не будет достигнут критерий остановки. Ни один элемент не перемещается между кластерами.

Далее предложена похожая процедура. На первом шаге, книги из одной серии сравниваются между собой с помощью Алгоритма кластеризации 3 и Правила классификации 2 (см. пункт 2.2.2). Результаты представляются квадратной матрицей, где ’1’ обозначает, что для соответ-

ствующей пары книг найдено различие в стилях. Однако, на этом этапе необходимо классифицировать похожесть между книгами посредством общей связи между стилями. А именно, размещением книг в группы на основе их похожести или различия со всеми книгами в серии. Для реализации этого создаются кластеры по строкам от полученной двоичной матрицы классификации, с использованием односвзяного агломеративного иерархического алгоритма (см. п. 1.4.1). Процесс повторяется до тех пор, пока все элементы не будут собраны в один кластер. Результатом работы иерархической процедуры кластеризации является вложенная структура стилей.

Полученная дендрограмма является наглядным представлением развития авторского стиля.

Цикл "Основание" А.Азимова

На Таблицах 3.2 и 3.3 представлены результаты сравнения стилей, полученных с помощью расстояний $d_{Spearman}$ и $d_{Canberra}$.

	$F1$	$F2$	$F3$	$F4$	$F5$	$F6$	$F7$
$F1$	0	0	1	1	1	1	1
$F2$	0	0	1	1	1	1	1
$F3$	1	1	0	1	0	1	1
$F4$	1	1	1	0	0	1	1
$F5$	1	1	0	0	0	1	1
$F6$	1	1	1	1	1	0	1
$F7$	1	1	1	1	1	1	0

Таблица 3.2: Сравнение книг из цикла "Основание" с помощью расстояния $d_{Spearman}$.

На Таблице 3.2 видны следующие кластеры: $\{F1, F2\}$, $\{F3, F4, F5\}$, $\{F6\}$ и $\{F7\}$. Первые две строки и первые два столбца (первый кластер) на Таблице 3.2 содержат только '0'. Блок, соответствующий второму кластеру, состоит из семи '0' и только двух '1'. Шестой и седьмой столбец состоят только из '1', за исключением диагональных элементов. Дендрограмма, представленная на Рис. 3.7 (верхняя часть), подтверждает

	$F1$	$F2$	$F3$	$F4$	$F5$	$F6$	$F7$
$F1$	0	0	1	1	1	1	1
$F2$	0	0	1	1	1	1	1
$F3$	1	1	0	0	0	1	1
$F4$	1	1	0	0	1	1	1
$F5$	1	1	0	1	0	1	0
$F6$	1	1	1	1	1	0	0
$F7$	1	1	1	1	0	0	0

Таблица 3.3: Сравнение книг из цикла “Основание” с помощью расстояния $d_{Canberra}$.

результат разделения.

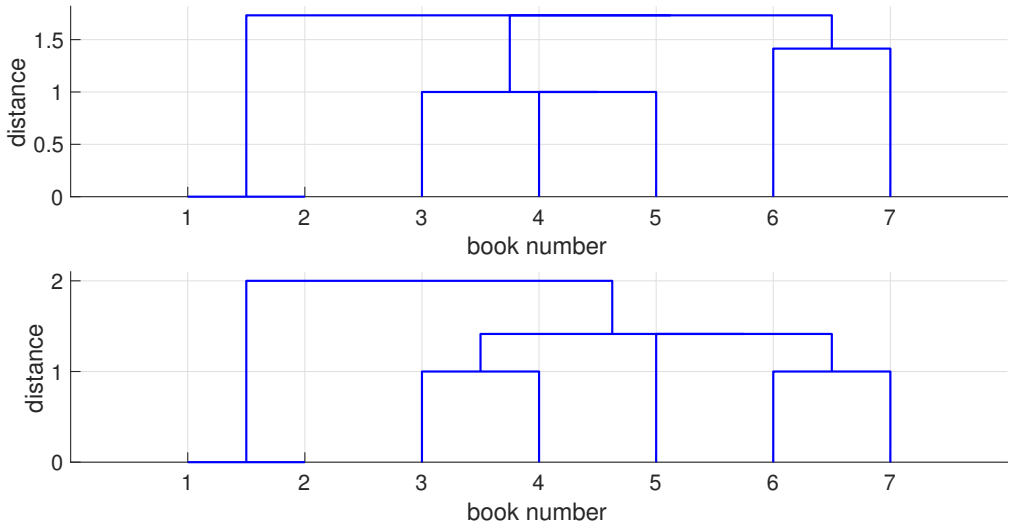

Рис. 3.7: Дендрограмма иерархии книг из цикла “Основание”.

Как можно увидеть из Таблицы 3.3 и Рис. 3.7 (нижняя часть), разделение по расстоянию $d_{Canberra}$ немного другое. Второй кластер содержит $\{F3\}$ и $\{F4\}$ и “Второе Основание” было отнесено к $\{F5,\ F6,\ F7\}$. Такой результат может быть следствием того, что А. Азимов собирался завершить цикл на книге “Второе Основание”, однако поклонники убедили его написать продолжение.

Цикл “Основание” изначально состоял из восьми небольших глав, которые были опубликованы между маем 1942 г. и январем 1950 г. Первый том серии, названный “Основание”, был выпущен в 1951 г. и состоял из четырех основных и одного побочного сюжета, который был выпущен позже. Остальные сюжеты были изданы в “Основание и Империя” (1952) и “Второе Основание” (1953), по два сюжета в каждом. Эта серия из трех томов известна как “Трилогия Основание” и она в точности совпадает со вторым кластером в полученном разделении. Перечисленные три книги образуют один и тот же кластер в обоих разделениях.

Четвёртый том под названием “Кризис Основания”, был написан в 1982 г., почти через 30 лет. Позже, в 1986 г., за ним последовал “Основание и Земля”. В них, Азимов пытается свести вместе три романа “Робот”, “Империя” и “Основание” в общей “Вселенной” и ввести понятие “Галаксия” для интегрированного коллективного разума. Эта пара книг составляет третий кластер во втором разделении и две разные группы в первом. Это может быть связано с различием в посыле этих книг. После А.Азимов написал предысторию, которая и составляет первый кластер. Таким образом, оба разделения полученных обсуждаемым методом хорошо описывают эволюцию авторского стиля.

Цикл “Рама” А.Кларка

Таблицы 3.4 и 3.5 содержат результаты сравнения стилей, полученных с помощью расстояний $d_{Spearman}$ и $d_{Canberra}$ соответственно.

	$R1$	$R2$	$R3$	$R4$	$R5$	$R6$
$R1$	0	1	1	1	1	1
$R2$	1	0	0	0	0	0
$R3$	1	0	0	0	0	1
$R4$	1	0	0	0	0	1
$R5$	1	0	0	0	0	1
$R6$	1	0	1	1	1	0

Таблица 3.4: Сравнение книг из цикла “Рама” с помощью расстояния $d_{Spearman}$.

	$R1$	$R2$	$R3$	$R4$	$R5$	$R6$
$R1$	0	1	1	1	1	1
$R2$	1	0	0	0	0	0
$R3$	1	0	0	1	0	1
$R4$	1	0	1	0	0	0
$R5$	1	0	0	0	0	0
$R6$	1	0	1	0	0	0

Таблица 3.5: Сравнение книг из цикла “Рама” с помощью расстояния $d_{Canberra}$.

На Рис. 3.8 изображены дендрограммы иерархии циклов.

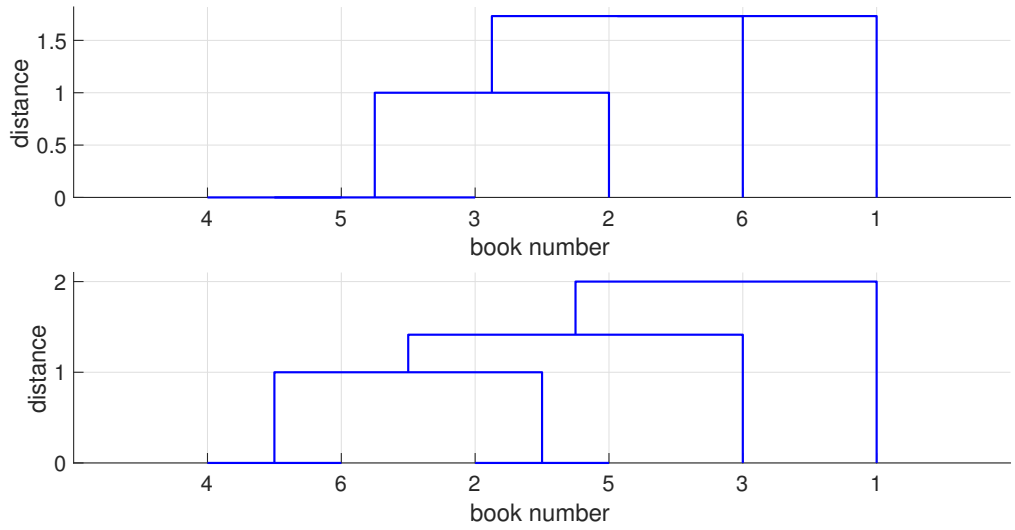

Рис. 3.8: Дендрограммы иерархии книг в цикле “Рама”.

Как и в результате эксперимента, описанного в п. 3.2, первое произведение, написанное только А.Кларком, выделяется в отдельный кластер.

Цикл “Властелин колец” Дж. Р. Р. Толкина

“Властелин колец” — роман-эпопея английского писателя Джона Рональда Руэла Толкина, одно из самых известных произведений жанра фэнтези. Роман был написан как единая книга, но при первом издании

его разделили на 3 части из-за объема. “Властелин колец” является продолжением повести “Хоббит”, опубликованной в 1937 году.

Были проанализированы следующие прозведения:

- “Хоббит” (1937) ($T1$),
- “Братство кольца” (1954) ($T2$),
- “Две башни” (1954) ($T3$),
- “Возвращение короля” (1955) ($T4$),
- “Сильмариллион” (1977) ($T5$).

На Таблицах 3.6, 3.7 и Рис. 3.9 отображены полученные результаты.

	$T1$	$T2$	$T3$	$T4$	$T5$
$T1$	0	0	0	1	1
$T2$	0	0	0	0	1
$T3$	0	0	0	0	1
$T4$	1	0	0	0	1
$T5$	1	1	1	1	0

Таблица 3.6: Сравнение цикла “Властелин колец” с помощью расстояния $d_{Spearman}$.

	$T1$	$T2$	$T3$	$T4$	$T5$
T1	0	0	0	1	1
T2	0	0	0	0	1
T3	0	0	0	0	1
T4	1	0	0	0	0
T5	1	1	1	0	0

Таблица 3.7: Сравнение книг из цикла “Властелин колец” с помощью расстояния $d_{Canberra}$.

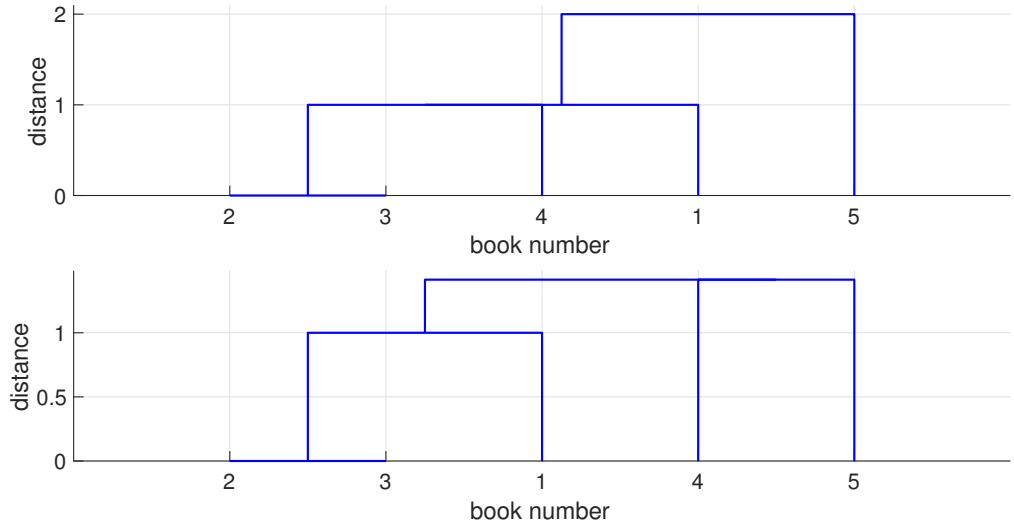

Рис. 3.9: Дендрограммы иерархии книг в цикле “Властелин колец”.

Из Таблицы 3.6 видно, что книги $T2$, $T3$ и $T4$ (основная часть трилогии) составляют полностью однородный кластер (только ’0’ в соответствующих блоках матрицы).

Такой результат предсказуем, так как книги были написаны в виде одного романа, который издателями в виду большого объема было решено разделить на три части.

Как и следовало ожидать, произведение $T1$ (“Хоббит”) тесно связан с этим кластером. Однако, стиль книги отличается от стиля $T4$.

Наконец, последнее произведение $T5$ расположено дальше от всех остальных. Это может объясняться тем, что “Сильмариллион” был составлен и издан сыном Дж. Толкина, Кристофером, в 1977. Ему пришлось добавлять новые тексты, чтобы исправить несоответствия в сюжете. Основное отличие в классификации, полученной с помощью расстояния $d_{Canberra}$ в расстоянии между произведениями $T4$ и $T5$. Тем не менее, общая структура книжной серии сохранена.

Результаты применения предложенных алгоритмов к анализу серийных последовательностей книг показывают, что рассмотренная в диссертации новая динамическая модель фрагментов текстов дает для каждого автора некоторые новые уникальные характеристики его стиля.

Заключение

Перечислим основные результаты диссертационной работы:

1. Предложен метод построения динамических моделей текстовых документов.

2. Разработан и теоретически обоснован алгоритм классификации текстовых документов, основанный на кластеризации спектрального представления динамических моделей текстовых документов.

3. Разработан и теоретически обоснован алгоритм классификации фрагментов текстовых документов, основанный на кластеризации динамических моделей текстовых документов с помощью расстояний на ядрах.

Литература

[1] *Айвазян С.А., Бухштабер В.М., Енюков И.С., Мешалкин Л.Д.* Прикладная статистика: Классификация и снижение размерности — М.: Финансы и статистика, 1989, 607 с.

[2] *Айзерман М. А., Браверман Э. М., Розоноэр Л. И.* Метод потенциальных функций в теории обучения машин. — 1970.

[3] *Вапник В.Н., Червоненкис А.Я.* Теория распознавания образов: статистические проблемы обучения — М.: Наука, 1974, 416 с.

[4] *Граничин О. Н., Измакова О. А.* Рандомизированный алгоритм стохастической аппроксимации в задаче самообучения // Автоматика и телемеханика — 2005 — No. 8 — P. 52–63.

[5] *Кижаева Н.А.* Тематическое моделирование и кластеризация текстов на арабском языке // Стохастическая оптимизация в информатике, 2013. — Т. 9, — №2. — С. 33–40.

[6] *Кижаева Н.А.* Динамическая модель процесса эволюции текстовых документов // Стохастическая оптимизация в информатике, 2018. — Т. 14. — №1. — С. 31–46.

[7] *Кижаева Н.А., Шалымов Д.С.* Определение авторского стиля текстов на основе статистического подхода двухвыборочного тестирования и метода К-ближайших соседей // Компьютерные инструменты в образовании, 2015. — №5. — С.14–23.

[8] *Поляк Б. Т., Хлебников М. В.* Метод главных компонент: робастные версии // Автоматика и телемеханика. — 2017. — №3. — С. — 130–148.

[9] *Садовничий В. А.* Теория операторов. — 1986.

[10] *Фишер Р.А.* Статистические методы для исследователей. М.: Госстатиздат, 1954, 267 с.

[11] *Фомин В.Н.* Математическая теория обучаемых опознающих систем — Л.: ЛГУ, 1976, 236 с.

[12] *Цыпкин Я.З.* Адаптация и обучение в автоматических системах. — М.: Наука. — 1968. — 400 с.

[13] *Цыпкин Я.З.* Основы теории обучающихся систем. — М.: Наука. — 1970. — 252 с.

[14] *Шахтарин Б., Ковригин В.А.* Методы спектрального оценивания случайных процессов. — Гелиос АРВ, 2005.

[15] *Alred J., Brusaw Ch.T., Oliu W.E.* Handbook of Technical Writing, Ninth Edition. — St. Martin's Press. — 2008.

[16] *Amelin K., Granichin O., Kizhaeva N., Volkovich Z.* Patterning of writing style evolution by means of dynamic similarity // Pattern Recognition, 2017,
https://doi.org/10.1016/j.patcog.2017.12.011

[17] *Anick P. G., Vaithyanathan S.* Exploiting clustering and phrases for context-based information retrieval // ACM SIGIR Forum. — ACM, 1997. — Т. 31. — №. SI. — P. 314-323.

[18] *Argamon S.* Interpreting Burrows's Delta: Geometric and probabilistic foundations // Literary and Linguistic Computing. — 2008. — Vol. 23, No. 2. — P. 131–147.

[19] *Aronszajn N.* Theory of reproducing kernels // Transactions of the American mathematical society — 1950. — Vol. 68 — No. 3 — P. 337–404.

[20] *Belanche L., Vázquez J. L., Vázquez M.* Distance-based kernels for real-valued data // Data Analysis, Machine Learning and Applications. — Springer, Berlin, Heidelberg, 2008. — P. 3-10.

[21] *Berg C., Christensen J. P. R., Ressel P.* Harmonic Analysis on Semigroups. — 1984.

[22] *Berkhin P.*A Survey of Clustering Data Mining Techniques // In: Proc. of the Grouping Multidimensional Data - Recent Advances in Clustering. — 2006. — P. 25–71.

[23] *Bibby J. M., Kent J. T., Mardia K. V.* Multivariate Analysis. — 1979.

[24] *Bishop C. M.* Pattern Recognition and Machine Learning // Springer. — 2006.

[25] *Blei D. M., Ng A. Y., Jordan M. I.* Latent dirichlet allocation // Journal of Machine Learning research. — 2003. — T. 3. — No. Jan. — P. 993–1022.

[26] *Bolshoy A., Volkovich Z., Kirzhner V., Barzily Z.* Genome Clustering: From Linguistic Models to Classification of Genetic Texts // Springer Science & Business Media. — 2010.

[27] *Burges C. J. C.* A tutorial on support vector machines for pattern recognition // Data Mining and Knowledge Discovery. — 1998. — T. 2. — №. 2. — P. 121-167.

[28] *Burrows J. F.* Delta: A measure of stylistic difference and a guide to likely authorship // Literary and Linguistic Computing. — 2002. — Vol. 17. — P. 267–287.

[29] *Cai C. S., Yang J., Shulin S. W.* A clustering based feature selection method using feature information distance for text data // In: Proc. of the Intelligent Computing Theories and Application: 12th International Conference. — 2016. — P. 122–132.

[30] *Calinski T., Harabasz J.* A dendrite method for cluster analysis // Communications in Statistics-theory and Methods. — 1974. — Vol. 3. — No. 1. — P. 1–27.

[31] *Calvo-Zaragoza J., On J.* An efficient approach for interactive sequential pattern recognition // Pattern Recognition. — 2017. — Vol. 64, No. Supplement C. — P. 295–304.

[32] *M. Campi* Classification with guaranteed probability of error // Machine learning. — 2010. — Vol. 80. — No. 1. — P. 63–84.

[33] *Cao F., Liang J., Jiang G.* An initialization method for the k-means algorithm using neighborhood model // Computers & Mathematics with Applications. — 2009. — Vol. 58, No. 3. — P. 474–483.

[34] *Carpenter B.* Integrating out multinomial parameters in latent Dirichlet allocation and naive Bayes for collapsed Gibbs sampling // Rapport Technique. — 2010. — T. 4. — P. 464.

[35] *Cerra D., Datcu M., Reinartz P.* Authorship analysis based on data compression // Pattern Recognition Letters. — 2014. — Vol. 42, No. Supplement C. — P. 79–84.

[36] *Cha S. H.* Comprehensive survey on distance/similarity measures between probability density functions // International Journal of Mathematical Models and Methods in Applied Sciences. — 2007. — Vol. 1, No. 4. — P. 300–307.

[37] *Chemudugunta C. et al.* Modeling documents by combining semantic concepts with unsupervised statistical learning // International Semantic Web Conference. — Springer, Berlin, Heidelberg, 2008. — P. 229-244.

[38] *Chiu T., Fang D., Chen J., Wang Y., Jeris C.* A robust and scalable clustering algorithm for mixed type attributes in large database environment // In: Proc. of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — 2001. — P. 263–268.

[39] *Cilibrasi R., Vitanyi P. M.* Clustering by compression // IEEE Transactions on Information Theory. — 2005. — Vol. 51. — P. 1523–1545.

[40] *Cohen K. B., Hunter L.* Getting started in text mining // PLoS computational biology. — 2008. — Т. 4. — №. 1. — P. e20.

[41] *Cohen W. W., Hirsh H.* Joins that generalize: text classification using WHIRL // In.: Proc. of ACM KDD Conference. — 1998. — P. 169-173.

[42] *Collier N., Nobata C., Tsujii J.* Extracting the names of genes and gene products with a hidden Markov model // In: Proc. of the 18th conference on Computational linguistics-Volume 1. — Association for Computational Linguistics, 2000. — P. 201-207.

[43] *Cortes C., Vapnik V.* Support-vector networks // Machine learning. — 1995. — Vol. 20. — №. 3. — P. 273–297.

[44] *Coyotl-Morales R. M., Villasenor-Pineda L., Montes-y-Gomez M., Rosso P.* Authorship attribution using word sequences // In: Proc. of the Iberoamerican Congress on Pattern Recognition. — 2006. — P. 844–853.

[45] *Coyotl-Morales R. M., Villasenor-Pineda L., Montes-y-Gomez M., Rosso P.* Grouping multidimensional data – Recent Advances in Clustering. — Springer. — 2006.

[46] *Deza M.M., Deza E.* Encyclopedia of Distances // Springer. — 2009.

[47] *Dhillon I., Guan Y., Kogan J.* Iterative clustering of high dimensional text data augmented by local search // In: Proc. of The 2nd IEEE Data Mining Conference. — 2002. — .

[48] *Diederich J., Kindermann J., Leopold E., Paas G.* Authorship attribution with support vector machines // Applied Intelligence. — 2003. — Vol. 19, No. 1. — P. 109–123.

[49] *Drucker H., Wu D., Vapnik V. N.* Support vector machines for spam categorization // IEEE Transactions on Neural networks. — 1999. — T. 10. — №. 5. — P. 1048-1054.

[50] *Duda R. O., Hart P. E., Stork D. G.* Pattern classification. — John Wiley & Sons, 2012.

[51] *Dudoit S., Fridlyand J.* A prediction-based resampling method for estimating the number of clusters in a dataset // Genome biology. — 2002. — Vol. 3. — No. 7. — P.‘112–129.

[52] *Dumais S. T. et al.* Latent semantic indexing (LSI) and TREC-2 // Nist Special Publication Sp. — 1994. — P. 105-105.

[53] *Dunn J. C.* Well-separated clusters and optimal fuzzy partitions // Journal of cybernetics. — 1974. — Vol. 4. — №. 1. — P. 95–104.

[54] *Feng G. et al.* A Bayesian feature selection paradigm for text classification // Information Processing & Management. — 2012. — T. 48. — №. 2. — P. 283-302.

[55] *Filippone M. et al.* A survey of kernel and spectral methods for clustering // Pattern recognition. — 2008 — Vol. 41 — №. 1 — P. 176–190.

[56] *Forgy E.W.* Cluster analysis of multivariate data – efficiency vs interpretability of classifications // Biometrics. — 1965. — No. 21. — P. 768—769.

[57] *Frery J., Largeron C., Juganaru-Mathieu M.* UJM at CLEF in author verification based on optimized classification trees // In: Proc. of the CLEF 2014.

[58] *Fukunaga K.* Introduction to Statistical Pattern Recognition. — New York: Academic Press. — 1972. — 618 p.

[59] *Gordon A. D.* Identifying genuine clusters in a classification // Computational Statistics & Data Analysis. — 1994. — Vol. 18. — No. 5. — P. 561–581.

[60] *Granichin O., Kizhaeva N., Shalymov D., Volkovich Z.* Writing style determination using the KNN text model // In: Proc. of the 2015 IEEE International Symposium on Intelligent Control. — Sydney, Australia, 2015. — September 21–23. — P. 900–905.

[61] *Granichin O., Volkovich V., Toledano-Kitai D.* Randomized Algorithms in Automatic Control and Data Mining. Springer-Verlag: Heidelberg New York Dordrecht London. — 2015. — 251 p.

[62] *Gregor H.* Parameter Estimation for Text Analysis. Technical report. — 2005.

[63] *Griffiths T. L., Steyvers M.* Finding scientific topics // In: Proc. of the National academy of Sciences. — 2004. — Vol. 101. — No. suppl 1. — P. 5228–5235.

[64] *Günal S. et al.* On feature extraction for spam e-mail detection // International Workshop on Multimedia Content Representation, Classification and Security. — Springer, Berlin, Heidelberg, 2006. — P. 635–642.

[65] *Halvani O., Steinebach M.*An efficient intrinsic authorship verification scheme based on ensemble learning // In: Proc. of the 9th International Conference on Availability, Reliability and Security. — 2014. — P. 571–578.

[66] *Han E. H. S., Karypis G., Kumar V.* Text categorization using weight adjusted k-nearest neighbor classification // Pacific-Asia Conference on Knowledge Discovery and Data Mining. — Springer, Berlin, Heidelberg, 2001. — P. 53–65.

[67] *Han E. H. S., Karypis G.* Centroid-based document classification: analysis and experimental results // European Conference on Principles of Data Mining and Knowledge discovery. — Springer, Berlin, Heidelberg, 2000. — P. 424–431.

[68] *Han J., Pei J., Kamber M.* Data Mining: Concepts and Techniques. — Elsevier, 2011.

[69] *Hartigan J. A.* Clustering Algorithms (Probability & Mathematical Statistics). — New York: Wiley, 1975, 351 p.

[70] *Hofmann T.* Probabilistic latent semantic indexing // ACM SIGIR Forum. — ACM, 2017 — Vol. 51 — No. 2 — P. 211-218.

[71] *Hoover D.L.* Testing Burrows's delta // Literary and Linguistic Computing. — 2004. — Vol. 19, No. 4. — P. 453–475.

[72] *Hopfield J.* Neurons with graded response have collective computational properties like those of two-state neurons // In: Proc. of the National Academy of Sciences. — 1984. — No. 81. — P. 3088—3092.

[73] *Hubert L., Arabie P.* Comparing partitions // Journal of Classification. — 1985. — Vol. 2, No. 1. — P. 193–218.

[74] *Hubert L., Schultz J.* Quadratic assignment as a general data analysis strategy // British journal of mathematical and statistical psychology. — 1976. — Vol. 29. — №. 2. — P. 190–241.

[75] *Hughes J. M., Foti N. J., Krakauer D. C., Rockmore D. N.* Quantitative patterns of stylistic influence in the evolution of literature // In: Proc. of the National Academy of Sciences. — 2012. — Vol. 109. — No. 20. — P. 7682–7686.

[76] *James M.* Classification Algorithms. — Wiley-Interscience, 1985.

[77] *Jankowska M., Keselj V., Milios E. E.* Proximity based one-class classification with common N-gram dissimilarity for authorship verification task // In: Proc. of the CLEF 2013 Evaluation Labs and Workshop. — 2013. — P. 23–26.

[78] *Joachims T.* A statistical learning model of text classification for support vector machines // In: Proc. of the 24th annual international

ACM SIGIR conference on Research and development in information retrieval. — ACM, 2001. — P. 128–136.

[79] *Joachims T.* Text categorization with support vector machines: Learning with many relevant features // European conference on machine learning. — Springer, Berlin, Heidelberg, 1998. — P. 137-142.

[80] *Juola P.* Authorship attribution // Foundations and trends in Information Retrieval. — 2006. — Vol. 1. — No. 3. — P. 33–334.

[81] *Kalt T., Croft W. B.* A new probabilistic model of text classification and retrieval. — Technical Report IR-78, University of Massachusetts Center for Intelligent Information Retrieval. — 1996.

[82] *Kaufman L., Rousseeuw P. J.* Finding Groups in Data: An Introduction to Cluster Analysis. // John Wiley. — 1990.

[83] *Kaufman L., Rousseeuw P. J.* Finding groups in data: an introduction to cluster analysis — John Wiley & Sons, 2009 — Vol. 344.

[84] *Kendall M. G., Gibbons J. D.* Rank Correlation Methods // Edward Arnold. — 1990.

[85] *Kestemont M., Luyckx K., Daelemans W., Crombez T.* Cross-Genre authorship verification using unmasking // English Studies. — 2012. — Vol. 93. — No. 3. — P. 340–356.

[86] *Kestemont M., Luyckx K., Daelemans W.* Intrinsic plagiarism detection using character trigram distance scores // In: Proc. of the PAN 2012 Lab Uncovering Plagiarism, Authorship, and Social Software Misuse held in conjunction with the CLEF 2012 Conference. — 2011. — P. 8.

[87] *Kizhaeva N., Shalymov D., Granichin O., Volkovich Z.* Studying of KNN two-sample test approach applications for writing style comparison of English and Russian text collections // In: Proc. of the AINL-ISMW FRUCT (Artificial Intelligence and Natural Language & Information Extraction, Social Media and Web Search). — ITMO University, FRUCT Oy, Finland. — Saint-Petersburg, Russia, 2015. — November 9–14. — P. 163–166.

[88] *Kizhaeva N., Volkovich Z., Granichin O., Granichina O., Kiyaev V.* Spectral profiling of writing process // In: Proc. of the 2017 IEEE

Conference on Control Technology and Applications. — Coast, Hawaii, USA, 2017. — August 27–30. — P. 2063–2068.

[89] *Koppel M., Schler J., Argamon S.* Computational methods in authorship attribution // Journal of the American Society for Information Science and Technology. — 2009. — Vol. 60, No. 1. — P. 9–26.

[90] *Koppel M., Winter Y.* Determining if two documents are written by the same author // Journal of the American Society for Information Science and Technology. — 2014. — Vol. 65, No. 1. — P. 178–187.

[91] *Krzanowski W. J., Lai Y. T.* A criterion for determining the number of groups in a data set using sum-of-squares clustering // Biometrics. — 1988. — P. 23–34.

[92] *Kulkarni V., Al-Rfou R., Perozzi B., Skiena S.* Statistically significant detection of linguistic change // In: Proc. of the 24th International Conference on World Wide Web. — 2015. — P. 11.

[93] *Lam W., Ho C. Y.* Using a generalized instance set for automatic text categorization // In: Proc. of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — ACM, 1998. — P. 81–89.

[94] *Lance G. N., Willams W. T.* A general theory of classification sorting strategies-Hierarchical System // Cognitive Journal. — 1967. — Vol. 9. — P. 373–380.

[95] *Lance G. N., Williams W. T.* Computer programs for hierarchical polythetic classification ("similarity analyses") // The Computer Journal. — 1966.. — Vol. 9. — No. 1. — P. 60–64.

[96] *Lemberg D., Soffer A., Volkovich Z.* New approach for plagiarism detection // International Journal of Applied Mathematics. — 2016. — Vol. 29. — No. 3. — P. 365–371.

[97] *Lewis D. D.* Naive (Bayes) at forty: The independence assumption in information retrieval // European conference on machine learning. — Springer, Berlin, Heidelberg, 1998. — P. 4-15.

[98] *Lovins J. B.* Development of a stemming algorithm // Mech. Translat. & Comp. Linguistics. — 1968. — Vol. 11. — №. 1-2. — P. 22–31.

[99] *Luyckx K., Daelemans W.* Authorship attribution and verification with many authors and limited data // In: Proc. of the 22nd International Conference on Computational Linguistics. — 2008. — P. 513–520.

[100] *Manning C., Schutze H.* Foundations of Statistical Natural Language Processing. — MIT Press. — 2003.

[101] *McCallum A. K.* Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering, 1996. — 1996.

[102] *McCallum A. K.* Mallet: A machine learning for language toolkit. — 2002.

[103] *McCallum A. et al.* A comparison of event models for naive bayes text classification // AAAI-98 workshop on learning for text categorization. — 1998. — T. 752. — №. 1. — P. 41-48.

[104] *McCallum A. et al.* Improving text classification by shrinkage in a hierarchy of classes // ICML. — 1998. — Vol. 98. — P. 359-367.

[105] *Mcauliffe J. D., Blei D. M.* Supervised topic models // Advances in neural information processing systems. — 2008. — P. 121–128.

[106] *Mika S. et al.* Kernel PCA and de-noising in feature spaces //Advances in neural information processing systems — 1999. — P. 536–542.

[107] *Milligan G. W., Cooper M. C.* An examination of procedures for determining the number of clusters in a data set // Psychometrika. — 1985. — Vol. 50. — No. 2. — P. 159–179.

[108] *Mitchell T. M. et al.* Machine learning // Burr Ridge, IL: McGraw Hill. — 1997. — Vol. 45. — No. 37. — P. 870–877.

[109] *Murtagh F.* A survey of recent advances in hierarchical clustering algorithms // The Computer Journal. — 1983. — T. 26. — №. 4. — P. 354-359.

[110] *Murtagh F.* Complexities of hierarchic clustering algorithms: state of the art // Computational Statistics Quarterly. — 1984. — Vol. 1. — №. 2. — P. 101–113.

[111] *Ng A. Y., Jordan M. I.* On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes // Advances in neural information processing systems — 2002 — P. 841–848.

[112] *Nigam K. et al.* Learning to classify text from labeled and unlabeled documents // AAAI/IAAI. — 1998. — Vol. 792.

[113] *Oliveira W., Justino E., Oliveira L.S.* Comparing compression models for authorship attribution // Forensic Science International. — 2013. — Vol. 228, No. 1. — P. 100–104.

[114] *Osuna E., Freund R., Girosit F.* Training support vector machines: an application to face detection // In: Proc. of the IEEE computer society conference on Computer vision and pattern recognition. — 1997. — P. 130–136.

[115] *Peng F., Schuurmans D., Keselj V., Wang S.* Augmenting naive bayes classifiers with statistical languages model // Information Retrieval. — 2004. — Vol. 7. — P. 317–345.

[116] *Popkov Yu. S., Dubnov Yu. A., Popkov A. Yu.* Randomized machine learning:

[117] *Porter M. F.* An algorithm for suffix stripping // Program. — 1980. — Vol. 14. — №. 3. — P. 130–137.

[118] *Rachev S.* Probability Metrics and the Stability of Stochastic Models // John Wiley & Son Ltd. — 1991.

[119] *Rand W.* Objective criteria for the evaluation of clustering methods // Journal of the American Statistical association. — 1971. — Vol. 66, No. 336. — P. 846–850.

[120] *Rocchio J. J.* Relevance feedback in information retrieval // The SMART Retrieval System: Experiments in Automatic Document Processing. — 1971. — P. 313–323.

[121] *Rosenblatt F.* Principles of Neurodynamics. — New York: Spartan Press. — 1962. — 616 p.

[122] *Rousseeuw P. J.* Silhouettes: a graphical aid to the interpretation and validation of cluster analysis // Journal of computational and applied mathematics — 1987 — Vol. 20 — P. 53–65.

[123] *Rudman J.* The state of authorship attribution studies: some problems and solutions // Computers and the Humanities. — 1998. — Vol. 31. — P. 351–365.

[124] *Salton G., Buckley C.* Term-weighting approaches in automatic text retrieval // Information processing & management — 1988 — Vol. 24 — No. 5 — P. 513–523.

[125] *Salton G., McGill M. J.* Introduction to Modern Information Retrieval McGraw-Hill New York. — 1983.

[126] *Salton G., Wong A., Yang C. S.* A vector space model for automatic indexing // Communications of the ACM. — 1975. — Vol. 18. — No. 11. — P. 613–620.

[127] *Schoenberg I. J.* Metric spaces and positive definite functions // Transactions of the American Mathematical Society. — 1938. — Vol. 44. — №. 3. — P. 522–536.

[128] *Scholkopf B., Smola A. J.* Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. — MIT press, 2001.

[129] *Shalymov D., Granichin O., Klebanov L., Volkovich Z.* Literary writing style recognition via a minimal spanning tree-based approach // Expert Systems with Applications. — 2016. — Vol. 61. — P. 145–153.

[130] *Sidorov G., Velasquez F., Stamatatos E., Gelbukh A., Chanona-Hernandez L.* Non-continuous syntactic N-grams // Expert Systems with Applications. — 2014. — Vol. 41. — No. 3. — P. 853–860.

[131] *Sidorov G.* Non-continuous Syntactic N-grams // International Journal of Computational Linguistics and Applications. — 2014. — Vol. 5, No. 1. — P. 139–158.

[132] *Sidorov G.* Non-continuous syntactic N-grams // Polibits. — 2013. — Vol. 48. — No. 1. — P. 67–75.

[133] *Stamatatos E., Daelemans W., Verhoeven B., Juola P., Lopez A., Potthast M., Stein B.*Overview of the Author Identification Task at PAN 2015 // In: Proc. of the CLEF (Working Notes). — 2015.

[134] *Stamatatos E.* A Survey of modern authorship attribution methods // Journal of the American Society for information Science and Technology. — 2009. — Vol. 60. — No. 3. — P. 538–556.

[135] *Stamatatos E.* Intrinsic plagiarism detection using character N-gram profiles // In: Proc. of the SEPLN 2009 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse. — 2009. — P. 38–46.

[136] *Stein S., Argamon S.* A mathematical explanation of Burrows's delta // In: Proc. of the Digital Humanities Conference. — 2006. — P. 207–209.

[137] *Sugar C. A., James G. M.* Finding the number of clusters in a dataset: An information-theoretic approach // Journal of the American Statistical Association. — 2003. — Vol. 98. — No. 463. — P. 750–763.

[138] *Tan S., Wang Y., Wu G.* Adapting centroid classifier for document categorization // Expert Systems with Applications. — 2011. — Т. 38. — №. 8. — P. 10264-10273.

[139] *Thompson R.* A note on restricted maximum likelihood estimation with an alternative outlier model // Journal of the Royal Statistical Society, Series B: Methodological. — 1985. — Vol. 47. — P. 53–55.

[140] *Vapnik V. N., Kotz S.* Estimation of dependences based on empirical data. — New York : Springer-Verlag, 1982. — Т. 40.

[141] *Veltkamp R. C., Hagedoorn M.* Shape similarity measures, properties and constructions // International Conference on Advances in Visual Information Systems. — Springer, Berlin, Heidelberg, 2000. — P. 467-476.

[142] *Vidyasagar M.* Randomized algorithms for robust controller synthesis using statistical learning theory // Automatica. – 2001. – Т. 37. – №. 10. – С. 1515-1528.

[143] *Willett P.* Recent trends in hierarchic document clustering: a critical review //Information Processing & Management. — 1988. — Т. 24. — №. 5. — P. 577-597.

[144] *Wu H., Bu J., Chen C., Zhu J., Zhang L., Liu H., Wang C., Cai D.* Locally discriminative topic modeling. — Elsevier. — 2012.

[145] *Yang Y., Chute C. G.* An example-based mapping method for text categorization and retrieval // ACM Transactions on Information Systems (TOIS). — 1994. — Т. 12. — №. 3. — P. 252-277.

[146] *Yang Y., Liu X.* A re-examination of text categorization methods // In: Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — ACM, 1999. — P. 42–49.

[147] *Zhang H., Chow T.W.S* A coarse-to-fine framework to efficiently thwart plagiarism // Pattern Recognition. — 2011. — Vol. 44, No. 2. — P. 471–487.

[148] *Zhang J., Yang Y.* Robustness of regularized linear classification methods in text categorization // In: Proc. of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. — ACM, 2003. — P. 190–197.

[149] *Zhao Y., Zobel J.* Effective and scalable authorship attribution using function words // In: Proc. of the Asia Information Retrieval Symposium. — 2000. — P. 174–189.

[150] *Zolotarev V. M.* Modern Theory of Summation of Random Variables. — Walter de Gruyter. — 1997.