МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование кафедры полностью)

01.03.02 Прикладная математика и информатика
(код и наименование направления подготовки, специальности)

Компьютерные технологии и математическое моделирование
(направленность(профиль))

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему Исследование алгоритмов тематического моделирования для новостных статей

Обучающийся А.С.Бондарева
(Инициалы Фамилия) (личная подпись)

Руководитель д.т.н., доцент, С.В. Мкртычев
(ученая степень(при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультант старший преподаватель, О.А. Головач
(ученая степень(при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2023

Аннотация

Тема выпускной квалификационной работы – «Исследование алгоритмов тематического моделирования для новостных статей».

Исследование и особенности практического применения алгоритмов тематического моделирования представляет актуальность и научно-практический интерес.

Объектом исследования бакалаврской работы является тематическое моделирование.

Предметом исследования бакалаврской работы являются алгоритмы тематического моделирования.

Цель бакалаврской работы – исследование и реализация алгоритмов тематического моделирования.

Методы исследования – методы и алгоритмы тематического моделирования, технологии реализации алгоритмов на языке высокого уровня.

Практическая значимость бакалаврской работы заключается в разработке и тестировании программы, реализующей эффективные алгоритмы тематического моделирования.

Результаты бакалаврской работы представляют научно-практический интерес и могут быть рекомендованы для анализа и программной реализации методов и алгоритмов тематического моделирования.

Бакалаврская работа состоит из 52 страницы текста, 16 рисунков, 3 таблиц и 28 источников.

Abstract

The title of the graduation work is: «Research of algorithms thematic modeling for news articles.».

The research and features of the practical application of thematic modeling algorithms is of relevance and scientific and practical interest.

The object of study of the bachelor's work is the thematic modeling.

The subject of research of the bachelor's work is the thematic modeling algorithms.

The aim of the bachelor's work is the research and implementation of thematic modeling algorithms.

Methods of research - methods and algorithms of thematic modeling, technology implementation of algorithms in high-level languages.

Practical significance of the bachelor's work is to develop and test a program that implements effective algorithms of thematic modeling.

The results of the bachelor's work are of scientific and practical interest and can be recommended for analysis and software implementation of methods and algorithms for thematic modeling.

The bachelor's thesis consists of 52 pages of text, 16 figures, 3 tables and 28 literature sources.

Оглавление

Введение

Когда мы ведем с кем-то дискуссию в реальной жизни, мы говорим на любую тему, чтобы выявить какой-то внутренний смысл. Точно так же подлежащее в NLP обозначает группу слов, которые каким-то образом связаны.

Тематическая модель автоматически находит темы в наборе документов. После этого можно использовать обученную модель, чтобы определить, какие из этих тем появляются в новых статьях. Модель также может определять, относятся ли части документа к конкретным темам.

Анализ текста на основе искусственного интеллекта использует широкий спектр методов или алгоритмов для естественной обработки языка, одним из которых является тематический анализ, используемый для автоматического определения тем в текстах.

Тематическое моделирование - это метод анализа текстов, который позволяет извлечь темы и подтемы из текстовых данных. Он используется в разных областях, таких как машинное обучение, информационный поиск и анализ данных. Тематическое моделирование основывается на представлении текстов в виде мешка слов и нахождении частых комбинаций слов, которые связываются в темы. Этот метод может раскрывать скрытые связи и смыслы, которые являются нетривиальными для выявления, работая с большими объемами информации, такими как текстовые документы, блоги, новости, твиты и т.д.

Исследование и особенности практического применения алгоритмов тематического моделирования представляет актуальность и научно-практический интерес.

В данной бакалаврской работе объектом исследования является тематическое моделирование.

Предметом исследования бакалаврской работы являются алгоритмы

тематического моделирования.

Цель работы заключается в исследовании и реализации этих алгоритмов, а также в разработке и тестировании программы.

Для достижения данной цели необходимо выполнить следующие задачи:

- выполнить постановку задачи исследования и проанализировать методы тематического моделирования;
- проанализировать алгоритмы тематического моделирования;
- разработать и протестировать программу, реализующую алгоритмы тематического моделирования.

Методы исследования – методы и алгоритмы тематического моделирования, технологии реализации алгоритмов на языках высокого уровня.

Практическая значимость работы заключается в разработке и тестировании программы, которая может эффективно реализовывать алгоритмы тематического моделирования.

Данная работа состоит из введения, трех глав, заключения и списка используемой литературы.

Первая главе работы посвящена постановке задачи исследования и анализу методов тематического моделирования.

Вторая глава работы посвящена обзору и анализу тематического моделирования.

В третьей главе рассматривается программная реализация и тестирование алгоритмов тематического моделирования.

В заключении описываются результаты выполнения выпускной квалификационной работы.

Бакалаврская работа состоит из 52 страницы текста, 16 рисунков, 3 таблиц и 28 источников.

Глава 1 Анализ методов тематического моделирования

Тематическое моделирование - это статистический метод обнаружения скрытых тем в наборе документов. Кластеризация - это метод машинного обучения для группировки похожих точек данных [5]. Оба метода группируют документы, но различаются тем, как они это делают.

Алгоритмы кластеризации группируют похожие элементы, а алгоритмы тематического моделирования определяют отношения между элементами. Тематическое моделирование использует статистический подход для поиска скрытых тем в наборе документов [5]. Кластеризация обычно используется для группировки элементов, чтобы их можно было анализировать как единое целое. Тематическое моделирование находит отношения между элементами и понимает скрытую структуру набора данных. Кроме того, тематическое моделирование не требует ручной маркировки точек данных, в то время как кластеризация требует ручной маркировки точек данных [5].

Чтобы сделать вывод о субъектах из неструктурированных данных, тематическое моделирование включает подсчет слов и группировку похожих словесных шаблонов. Предположим, что мы являемся фирмой-разработчиком программного обеспечения, заинтересованной в том, чтобы узнать, что потребители говорят о конкретных элементах нашего продукта, и нам потребуется использовать алгоритм тематического моделирования для изучения наших комментариев вместо того, чтобы тратить часы на то, чтобы выяснить, какие сообщения говорят о наших продуктах. интересующие темы.

Тематическая модель группирует сопоставимые отзывы, а также наиболее часто встречающиеся фразы и выражения, распознавая такие закономерности, как частотность слов и расстояние между словами. Используя эту информацию, мы можем быстро сделать вывод о том, о чем каждая группа текстов.

Тему нельзя строго определить ни семантически, ни эпистемологически.

Темы выявляются исключительно с помощью автоматического подсчета

правдоподобия совместной встречаемости слов. Слово может быть отнесено к нескольким темам, но с разной вероятностью, но слова-соседи для каждой темы у него будут разными.

Применение тематического моделирования стало разнообразным: контролируемые, неконтролируемые и полууправляемые подходы модифицируются и изобретаются для применения в интеллектуальном анализе текста, классификации текста, машинном обучении,поиске информации и системах рекомендаций.

Тематическое моделирование предоставляет методы автоматического извлечения тематической информации из наборов текстов, таких как статьи, книги, блоги и другие текстовые документы.

Это может помочь в следующем:

- обнаружение скрытых тем в коллекции.
- обнаружение и отслеживание событий в новостных потоках;
- построение профилей интересов пользователей в рекомендательных системах;
- классификация документов по обнаруженным темам.
- использование классификации для организации/обобщения/поиска документов.

Основные этапы интеллектуального анализа текста показаны на рисунке 1, первыми шагами в процессе анализа текста идут сбор данных из нескольких источников данных, такие как, например, новостные веб-страницы.

Затем был применен этап предварительной обработки для очистки данных, а после преобразования извлеченной информации в структурированный формат, для анализа шаблонов (видимых и скрытых) в данных.

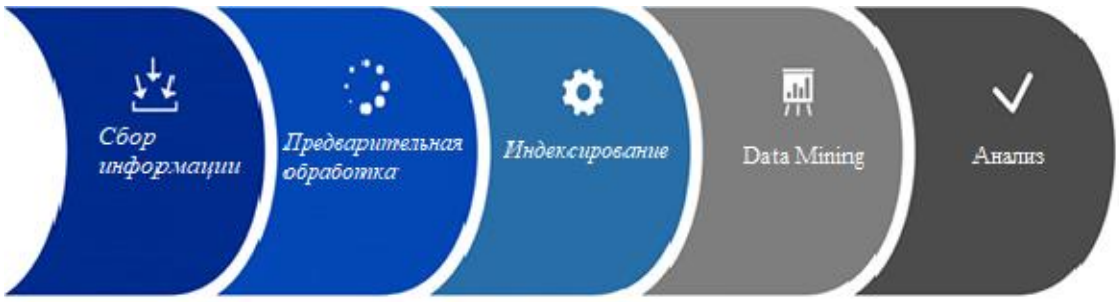

Рисунок 1 – Этапы, связанные с процессом интеллектуального анализа текста

«Наибольшее применение в современных приложениях находят подходы, основанные на Байесовских сетях - вероятностных моделях на ориентированных графах.

Относительно молодой областью исследований в теории самообучения являются вероятностные тематические модели. Одним из первых был предложен вероятностный латентно-семантический анализ (PLSA) [20], который основан на принципе максимума правдоподобия. Он был предложен в качестве альтернативы классическим методам кластеризации, которые основаны на вычислении функций расстояния. Затем был предложен метод латентного размещения Дирихле (LDA) и его многочисленные обобщения.

Вероятностные тематические модели отличаются от классических методов кластеризации тем, что они позволяют документу или термину относиться к нескольким темам с различными вероятностями, осуществляя таким образом «мягкую» кластеризацию. Каждая тема в вероятностной тематической модели описывается дискретным распределением на множестве терминов, а каждый документ – дискретным распределением на множестве тем. Другие тематические модели как правило являются расширением LDA, например, размещение пачинко улучшает LDA за счёт введения дополнительных корреляционных коэффициентов для каждого слова, которое составляет тему» [6].

Существуют различные методы, которые позволяют автоматически находить темы в наборе текстовых документов и определять, какие слова в

этих документах связаны с каждой темой.

Некоторые из них [6][23]:

- латентноеразмещениеДирихле(LDA, Latent Dirichlet allocation);
- неотрицательнаяматричнаяфакторизация (Non-negative matrix factorization (NMF or NNMF));
- латентно-семантическийанализ(Latentsemanticanalysis, LSA));
- модель распределенияПачинко(pachinkooallocation,PAM));
- вероятностный латентно-семантический анализ (Probabilisticlatentsemanticanalysis(PLSA));
- аддитивная регуляризация тематических моделей (ARTM,AdditiveRegularizationforTopicModeling).

Тем не менее, проводится много исследований по улучшению алгоритмов для понимания полного контекста документов.

Начнем с рассмотрения скрытого распределение Дирихле (LDA).

LDA, представленная [12], представляет собой вероятностную модель, которая считается самым популярным алгоритмом TM в реальных приложениях для извлечения тем из коллекций документов, поскольку она обеспечивает точные результаты и может быть обученаонлайн.Корпус организован как случайная смесь скрытых тем в модели LDA, и тема относится к распределению слов. Кроме того, LDA представляет собой генеративный неконтролируемый статистический алгоритм для извлечения тематической информации (тем) из коллекции документов в рамках байесовской статистической парадигмы. Модель LDA предполагает, что каждый документ состоит из различных тем, где каждая тема представляет собой распределение вероятностей по словам.

Существенным преимуществом использования модели LDA является то, что темы могут быть выведены из заданной коллекции без каких-либо предварительных знаний (рисунок 2) [11].

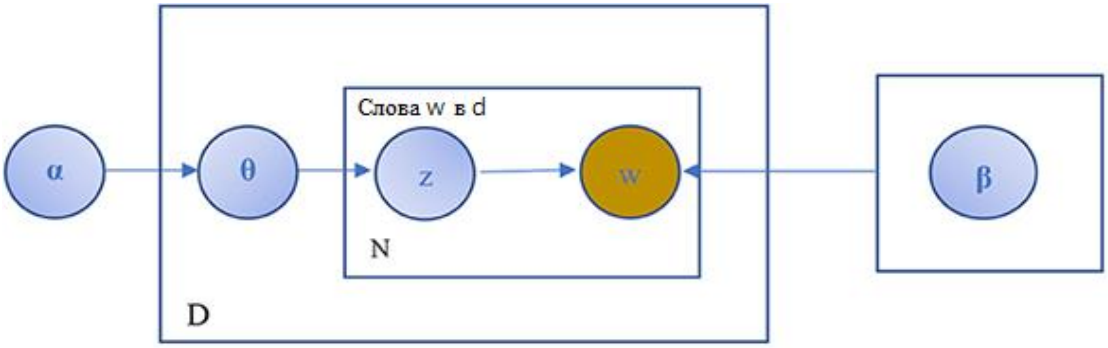

Рисунок 2 – Исходная структура тематической модели LDA

На рисунке 2.α - параметр, представляющий априор Дирихле для распределения тем документа, β - параметр, представляющий Дирихле для распределения слов, θ - вектор распределения тем по документу d, z - тема для выбранного слова в документе, w относится к конкретнымслова в N, таблица D - длина документов, а таблица N - количество слов в документе.

Простая модель LDA предоставляет мощный инструмент для обнаружения и использования скрытой тематической структуры в больших текстовых архивах. Однако одним из главных преимуществ формулирования LDA как вероятностной модели является то, что ее можно легко использовать в качестве модуля в более сложных моделях для более сложных целей [11].

Далее рассмотрим неотрицательную матричную факторизацию (NMF).

Неотрицательная матричная факторизация - это статистический метод, позволяющий уменьшить размерность входных корпусов. Он использует метод факторного анализа, чтобы придать сравнительно меньший вес словам с меньшей связностью. Рассмотрим матрицу терминов документа, полученную после удаления стоп-слов из корпуса. Матрица «термин-тема» и матрица «тема-документ» - это две матрицы, которые могут быть вынесены из матрицы [10].

Матричная факторизация может быть выполнена с использованием различных методов оптимизации. NMF можно выполнять быстрее и

эффективнее, используя иерархический чередующийся метод наименьших квадратов. В этом случае факторизация происходит путем обновления одного столбца за раз, в то время как другие столбцы остаются неизменными.

NMF - это неконтролируемый метод факторизации матриц (линейно-алгебраический), который способен выполнять как уменьшение размера, так и кластеризацию одновременно [10] [21]. Он может быть применен к многочисленным задачам ТМ; однако сообщалось лишь о нескольких работах по определению тем для коротких текстов.

В статье [28] представили модель NMF, целью которой является получение тем для коротких текстовых данных с использованием факторизирующей асимметричной матрицы корреляции терминов, матрицы термина-документа и матричного представления пакета слов текстового корпуса. [14] определил метод NMF как разложение неотрицательной матрицы D на неотрицательные множители U и V, $V \geq 0$ и $U \geq 0$. Модель NMF может извлекать релевантную информацию о темах без какого-либо предварительного понимания исходных данных. NMF обеспечивает хорошие результаты в нескольких задачах, таких как обработка изображений, анализ текста и процессы транскрипции (рисунок 3). Кроме того, он может обрабатывать декомпозицию непонятных данных, таких как видео.

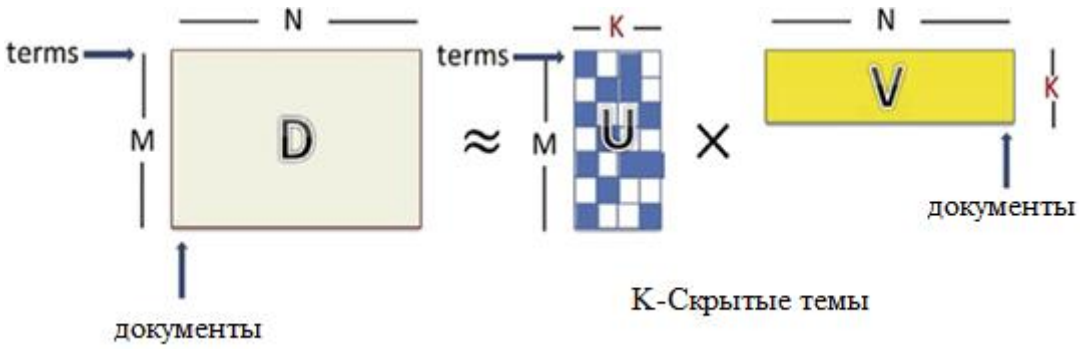

Рисунок 3 – Исходная структура тематической модели NMF

На рисунке 3, D ≈ UV, где U и V поэлементно неотрицательны, и для данного текста корпус разлагается на две матрицы, которые являются матрицей терминов U и матрицей документов V, соответствующей K координатным осям и N точкам в новом семантическом пространстве соответственно (каждая точка представляет один документ).

NMF по умолчанию создает разреженные представления. Это означает, что большинство записей близки к нулю и лишь очень немногие параметры имеют значимые значения. Это можно использовать, когда нам строго требуется меньше тем.NMF создает более последовательные темы по сравнению с LDA.

Далее перейдем к рассмотрению скрытого семантического анализа (LSA).

LSA: это метод в NLP, предложенный в[16], в частности, дистрибутивная семантика, который может использоваться в нескольких областях, таких как определение темы; он стал основой для выполнения многих передовых методов. Гипотезы распределения составляют теоретическую основу метода LSA, в котором говорится, что термины с аналогичным значением ближе с точки зрения их контекстуального использования, предполагая, что слова, близкие по своему значению, отображаются в связанных частях текстов [18]. Кроме того, он анализирует большие объемы необработанного текста на слова и разделяет их на значимые предложения или абзацы. LSA рассматривает как термины сходства текста, так и связанные термины, чтобы лучше понять тему. Кроме того, модель LSA может генерировать векторное представление для текстов, которое помогает группировать связанные слова. Наиболее распространенный вариант LSA основан на использовании сингулярного разложения матрицы (SVD). SVD - это метод факторизации матриц, который используется для разложения матрицы на три компоненты: матрицу левых сингулярных векторов (U), матрицу правых сингулярных векторов (V) и матрицу сингулярных значений (S). SVD может быть применен к любой матрице, включая матрицу, которая

представляет набор текстовых документов.

LSA использует SVD для нахождения скрытых семантических связей между словами в текстовых документах, тогда как SVD может быть применен напрямую к матрице документов-термов, чтобы находить скрытые семантические связи между документами.

На рисунке 4изучаются скрытые темы путем выполнения матричного разложения матрицы термина-документа; допустим, X - это матрица термина по документу, которая разлагается на три другие матрицы, S, W и P; умножая эти матрицы, мы возвращаем матрицу X с {X} = {S}{W} {P}; каждый абзац характеризуется столбцами, а строки характеризуют уникальные слова.

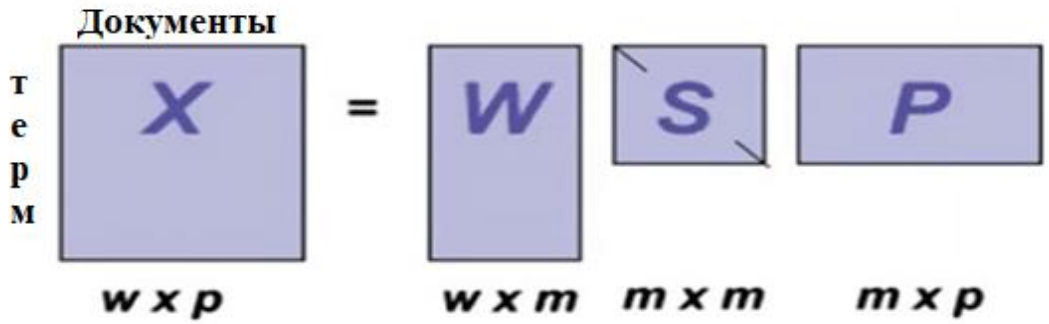

Рисунок 4 – Исходная структура метода SVD

Метод LSA использует матричное разложение для поиска скрытых семантических связей между словами в больших коллекциях текстов. LSA анализирует матрицу слов и документов, где каждый документ представлен в виде вектора, а каждое слово - в виде строки. Затем метод использует сингулярное разложение матрицы, чтобы найти скрытые семантические факторы, которые объясняют, какие слова часто употребляются вместе в документах.

Проверка сохраненной доли дисперсии, аналогичная анализу главных компонентов (PCA) или факторному анализу, для определения оптимальной размерности не подходит для LSA. Использование синонимического теста или

предсказание пропущенных слов - это два возможных метода нахождения правильной размерности. Когда темы LSA используются в качестве функций в методах контролируемого обучения, можно использовать измерения ошибок прогнозирования, чтобы найти идеальную размерность.

В отличии от LSA, PLSA работает с вероятностями появления слов и документов в темах. Для расчета этих вероятностей используется совершенно другой математический аппарат, который основан на правиле Байеса.

Хотя это выглядит совершенно по-другому и совсем по-другому подходит к проблеме, pLSA на самом деле просто добавляет вероятностную обработку тем и слов поверх LSA.Это гораздо более гибкая модель.

Основным недостатком PLSA является то, что он не учитывает контекст, в котором используются слова. PLSA использует вероятностную модель, которая моделирует распределение каждого слова в каждом документе как комбинацию распределений по темам. Для этого модели требуется огромное количество параметров, которые необходимо оценить из обучающих данных. Количество параметров растет линейно с числом слов и документов в коллекции, что приводит к проблемам с масштабируемостью и вычислительной сложности.В PLSA каждый документ представляется в виде набора тем, а каждая тема представляется в виде набора слов. Эти темы и слова выбираются таким образом, чтобы максимизировать вероятность появления слов в документах [9].

Для максимизации правдоподобия в PLSA используется итерационный процесс, известный как EM-алгоритм. Каждая итерация состоит из двух шагов: E (expectation) и M (maximization).

Алгоритм максимизации ожиданий, или сокращенно алгоритм EM, представляет собой подход к оценке максимального правдоподобия при наличии скрытых переменных. Алгоритм EM можно использовать в методах построения некоторых известных тематических моделей LDA, PLSA, ARTM и др.

Модель распределения Пачинко (PAM) представляет собой

усовершенствованный метод модели скрытого распределения Дирихле. Модель LDA выявляет корреляцию между словами, определяя темы на основе тематических отношений между словами, присутствующими в корпусе. Но PAM импровизирует, моделируя корреляцию между сгенерированными темами. Эта модель обладает большей силой в определении семантических отношений именно потому, что они также принимают во внимание отношения между темами. Модель названа в честь пачинко, популярной в Японии игры. Модель использует ориентированные акриловые графики, чтобы понять корреляцию между темами. DAG - это конечный ориентированный граф, показывающий, как связаны темы.

Далее рассмотрим подход ARTM (аддитивная регуляризация тематических моделей).

ARTM позволяет вводить в модель ограничения на частоты слов, на количество тем, на взаимодействие тем и другие ограничения, которые позволяют улучшить качество моделирования. Принцип работы ARTM заключается в добавлении регуляризаторов к функции правдоподобия модели, которые позволяют контролировать различные аспекты моделирования. Регуляризаторы могут быть настроены на основе предположений о структуре данных, что позволяет избежать переобучения модели и улучшить ее обобщающую способность.

BigARTM - это проект с открытым исходным кодом, основанный на аддитивной регуляризации в тематических моделях (ARTM), которая является небайесовской регуляризованной моделью и направлена на упрощение задачи вывода темы. BigARTM мотивирован предпосылкой, что предыдущие предположения Дирихле противоречат понятию разреженности в наших разделах документов, и что попытка учесть эту разреженность приводит к чрезмерно сложным моделям.

Мы используем аддитивную комбинацию регуляризаторов для сглаживания, синтаксического анализа и декорреляции, чтобы сделать темы более интерпретируемыми, разреженными. Фреймворк ARTM позволяет нам

выполнять все эти действия плавно, без сложных выводов и разработки новых алгоритмов. Все эти регуляризаторы были реализованы как часть набора инструментов тематического моделирования BigARTM с открытым исходным кодом.

Каждый документ состоит из статистического сочетания тем(т.е. статистического распределения всех тем, которое может быть получено путем «суммирования» всех распределений по всем темам, рассмотренным в корпусе). Что делают методы тематического моделирования, так это выясняют, какие темы присутствуют в документах внутри корпуса и в чем сила каждой из них.

Тематическое моделирование направлено на поиск тем (или кластеров) в корпусе текстов (например, писем или новостных статей), без предварительного ознакомления с этими темами. Здесь кроется настоящая сила тематического моделирования: вам не нужны никакие маркированные или аннотированные данные, только необработанные тексты, и из этого хаоса алгоритмы тематического моделирования найдут темы ваших текстов.

Для сравнения методов тематического моделирования используем таблицу 1.

Таблица 1 – Сравнение методов тематического моделирования

Методы ТМ	Преимущества	Недостатки
LSA	Решает проблему неполноты данных и учитывает синонимы слов. Уменьшает размерность TF-IDF с помощью разложения по сингулярным значениям. Не требует глубокой статистической подготовки и теории вероятности.	Сложность в определении темы в некоторых случаях и установлении количества тем. Не отражает корреляцию между несколькими темами. Не учитывает контекст в тексте.

Продолжение таблицы 1

NMF	Быстрая обработка большого количества данных в реальном времени. Способен извлекать значимые темы без предварительной информации или знания глубинного смысла в исходных данных.	Иногда дает семантически неверные результаты.
LDA	Обрабатывает длинные документы и может показывать прилагательные и существительные в темах. Работает с документами смешанной длины. Он использует статистическую модель для выявления скрытых тем в текстовых данных.	Не способен моделировать отношения между темами, которые помогают понять глубинные структуры документов. Алгоритм медленного процесса.
PLSA	PLSA использует вероятностную модель для моделирования текстовых данных. Метод работает на основе предположения о том, что каждый документ содержит несколько скрытых тем.	Невозможность управлять разреженностью получаемых на выходе матриц вероятностей. Линейный рост числа параметров PLSA с ростом числа документов в корпусе, что приводит к переобучению модели.
ARTM	ARTM позволяет использовать различные типы моделей и меры оценки. Это позволяет получать более точные и полезные результаты моделирования.	ARTM является трудоемким процессом, требуется большое количество компьютерных ресурсов для обработки текстов.

Таким образом, каждый метод имеет свои преимущества и недостатки, поэтому стоит проводить сравнительный анализ перед применением конкретного метода.

Выводы по главе 1

Первая глава бакалаврской работы была посвящена постановке задачи исследования, а также обзору и анализу методов тематического моделирования.

В результате проведенного исследования были получены следующие выводы:

- выбор метода тематического моделирования зависит от конкретной задачи и особенностей текстовых данных;

– методLSA использует сингулярное разложение для выявления общих паттернов в матрице. Недостатком этого метода является то, что он не учитывает контекст в тексте;
– метод NMF использует матричное разложение для выделения тем в текстовых данных;
– метод PLSA имеет высокую точность в моделировании текстовых данных, но требует большого количества итераций при обработке больших объемов данных;
– метод LDA имеет высокую точность в выявлении скрытых тем, а также хорошо работает с большими объемами текстовых данных. Однако он требует длительного времени компьютерной обработки;
– ARTM позволяет группировать документы в соответствии с темой их содержания. Это упрощает анализ больших объемов текстовой информации и помогает находить смежные темы и смысловые связи.

Метод PLSA используется для анализа крупных наборов данных, таких как текстовые документы, изображения и видео. Он может справляться со многими проблемами, с которыми сталкиваются другие методы, такие как разреженность, шум и многомерность. Метод LDA может быть применен к большим объемам данных и может обрабатывать тексты на нескольких языках. ARTM имеет высокую скорость обработки, что позволяет работать с большими объемами информации.

Далее будем рассматривать алгоритмы методов PLSA, LDAи ARTM.

Глава 2 Анализ алгоритмов тематического моделирования

2.1 Вероятностные тематические модели

С ростом объема и разнообразия цифровых данных становится все труднее находить нужную информацию. Однако, с помощью различных методов анализа текстовых данных, таких как вероятностные тематические модели, можно упростить процесс поиска и понимания содержания коллекций данных. существует множество инструментов, которые позволяют анализировать и визуализировать текстовые данные, что делает процесс анализа более удобным и понятным для пользователей[24].Поиск и ссылки являются двумя наиболее распространенными инструментами при работе с онлайн-информацией Поиск позволяет найти информацию на основе ключевых слов или фраз, которые пользователь вводит в поисковый запрос. Ссылки, с другой стороны, позволяют пользователям переходить на другие страницы и ресурсы, связанные с их интересами. Это хороший способ взаимодействия с нашим онлайн-архивом, но чего-то не хватает.

Вероятностные тематические модели позволяют представлять каждый документ в виде вероятностного распределения на темы, а каждую тему в виде вероятностного распределения на слова [3].

Вместо того, чтобы искать документы только с помощью поиска, по ключевым словам, мы могли бы сначала найти интересующую нас тему, а затем изучить документы, связанные с этой темой.

Вероятностная тематическая модель включает коллекцию текстовых документов ($D = \{d1,\ldots d|D|\}$), словарь термов ($W = \{w1,\ldots w|W|\}$), встречающихся в них, и конечное множество тем($T = \{t1,\ldots t|T|\}$). В качестве термов могут использоваться слова, коллокации, n-граммы, именованные существительные и т.д. Вероятностная тематическая модель предполагает, что каждое вхождение терма $w \in W$ в документ $d \in D$ связано с некоторой темой $t \in T$. Задача модели заключается в определении вероятностей

принадлежности каждого документа к каждой теме и вероятностей вхождения каждого терма в каждую тему. [7].

Вероятностная модель порождения данных. Согласно определению условной вероятности [4], формуле полной вероятности и гипотезе условной вероятности (1):

$$p(w|d) = \sum_{t \in T} p(w|t)p(t|d), \quad (1)$$

где p(t|d) и p(w|t)- искомые распределения.

Согласно модели порождения данных, коллекция D является выборкой наблюдений (d,w), где каждый документ d является случайной величиной, а каждый терм w в документе d также является случайной величиной. [4].

Алгоритм порождения коллекции текстов с помощью вероятностной модели:

Вход: распределения p(t|d), p(w|t);

Выход: выборка пар (di,wi), i = 1,…, n;

Шаг 1: для всех d ∈ D

Шаг 2: задать длину nd документа d;

Шаг 3: для всех i = 1,…, nd

Шаг 4: выбрать случайную тему tиз распределения p(t|d);

Шаг 5: выбрать случайный термин wиз распределения p(w|t);

Шаг 6: добавить в выборку пару (d,w), при этом тема t «забывается».

Процесс порождения последовательности слов документа показан на рисунке 5.

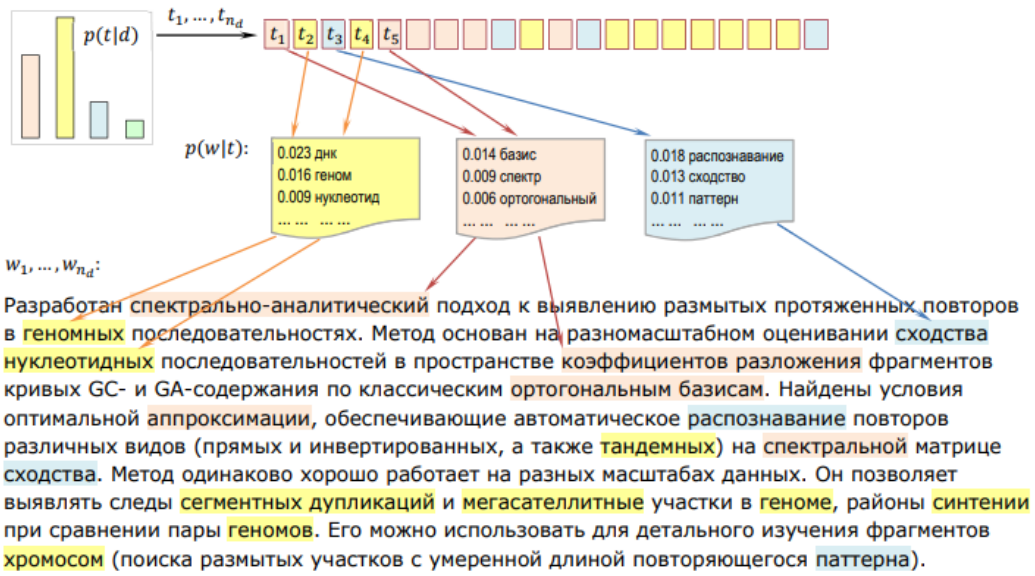

Рисунок 5 – Процесс порождения текстового документа вероятностной тематической моделью (1)

Построить тематическую модель коллекции документов D - значит найти множество тем T, распределения p(w|t) для всех тем t ∈ T и p(t|d) для всех документов d ∈ D. Распределения p(t|d) - это сжатые тематические операции документов, которые предполагается использовать для дальнейшего решения задач информационного поиска, классификации, категоризации, аннотирования, суммаризации текстовых документов [4].

Таким образом, каждый документ в коллекции порождается путем выбора распределения тем для документа и затем выбора слов для каждой темы в соответствии с распределением слов для этой темы.

В работе [15] рассмотрен наиболее популярный алгоритм Латентное размещение Дирихле. На практике исследователи используют одну из эвристик метода максимального правдоподобия, методы сингулярного разложения (SVD),метод моментов, алгоритм, основанный на неотрицательной матрице факторизации (NMF), вероятностные тематические модели, вероятностный латентно-семантический анализ, латентное размещение Дирихле.

2.2 Описание алгоритмов тематического моделирования

Алгоритмы тематического моделирования представляют тематическую структуру документа в виде вероятностного распределения слов. Их целью является обнаружение скрытой тематической структуры в больших архивах документов.

Темы, обнаруженные алгоритмами тематических моделей, представляют собой группы слов, где каждое слово имеет определенную степень ассоциации с темами. Это означает, что слово может принадлежать более чем к одной теме с разной степенью ассоциации с каждой темой [1]. В статистическом смысле темы можно рассматривать как распределение слов.

Начнем с рассмотрения модели pLSA.

«Подход pLSA моделирует вероятность p(w|d) появления термов w в документах d как смесь условно независимых мультиномиальных распределений. Компоненты смеси можно рассматривать как представления «тем» t. Наблюдаемые пары (d,w) встречаются независимо, что соответствует представлению документов в виде «мешка слов». Тематическая модель появления слов в документах выглядит следующим образом (2):

$$p(d,w) = p(d)p(w|d) = p(d)\sum_{t\in T} p(w|t)p(t|d) \quad (2)$$

где d-документ коллекции;

w-терм;

t-тема;

d-документ коллекции.

Когда строим тематическую модель нам требуется оценить параметры модели φwt = p(w|t) и θtd = p(t|d). Ставится задача максимизации логарифма правдоподобия (3)» [7]:

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

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

где $\Phi = (\varphi_{wt})W\times T$;

$\Theta = (\theta_{td})T\times D$;

n_{dw} - число вхождений терма w документ d.

«Задача решается с помощью EM-алгоритма. Вероятностные тематические модели находят локальный максимум логарифма правдоподобия. Известно, что pLSA не моделирует процесс выбора документов, распределение p(d), это частично генеративная модель.

Известно, что на каждом M-шаге нет необходимости слишком точно решать задачу максимизации правдоподобия. Достаточно сместиться в направлении максимума и затем выполнить E-шаг. Такой вариант EM-алгоритма называется обобщенным EM-алгоритмом (realizedEM-algorithm, GEM)» [17].

Алгоритм PLSA-EM:рациональный EM-алгоритм для модели PLSA:

Вход: коллекция документов D, число тем |T|, начальные приближения Θ иФ;

Выход: распределения Θ и Ф;

Шаг 1: пока не выполнится критерий остановки, повторять итерации:

Шаг 2: обнулить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_{t}$,$\hat{n}_{d}$,$\hat{n}_{dwt}$ для всех d ∈ D,w ∈ W,t ∈ T;

Шаг 3: для всехd ∈ D, w ∈ d;

Шаг 4: Z:=$\sum_{t\in T}\varphi_{wt}\theta_{td}$;

Шаг 5: для всех t ∈ T, таких, что $\varphi_{wt}\theta_{td} > 0$

Шаг 6: увеличить $\hat{n}_{wt}$, $\hat{n}_{dt}$, $\hat{n}_{t}$на $\delta = n_{dw}\varphi_{wt}\theta_{td}/Z$;

Шаг 7: φwt:=$\hat{n}_{wt}/\hat{n}_{t}$для всех w ∈ W, t ∈ T;

Шаг 8: θtd:=$\hat{n}_{dt}/\hat{n}_{d}$для всех d ∈ D, t ∈ T;

Алгоритм PLSA-GEM: рациональный EM-алгоритм для модели PLSA:

Вход: коллекция документов D, число тем |T|, начальные приближения Θ иФ;

Выход: распределения Θ и Ф;

Шаг 1: обнулить n̂wt, n̂dt, n̂t, n̂d, n̂dwt для всех d ∈ D, w ∈ W, t ∈ T;

Шаг 2: пока не выполнится критерий остановки, повторять итерации:

Шаг 3: для всех d ∈ D, w ∈ d;

Шаг 4: Z:=$\sum_{t\in T} \varphi_{wt}\theta_{td}$;

Шаг 5: для всехt ∈ Tтаких, что$n_{dwt} > 0$ или$\varphi_{wt}\theta_{td} > 0$;

Шаг 6: $\delta = n_{dw}\varphi_{wt}\theta_{td}/Z$;

Шаг 7: увеличить n̂wt, n̂dt, n̂tна (δ - ndwt);

Шаг 8: ndwt := δ;

Шаг 9: если пора обновить параметры Ф, Θ то

Шаг 10: φwt:=n̂wt/ n̂t для всех w ∈ W, t ∈ T таких, что n̂wt изменился;

Шаг 11: θtd:=n̂dt/n̂d для всех d ∈ D, t ∈ T таких, что n̂dt изменился.

Обобщенный EM-алгоритм дляPLSA/LDA отличается от стандартного более частым обновлением параметров θtdи φw по текущим значениям счетчиков $\hat{n}_{wt}$ и $\hat{n}_{dt}$. Возможные варианты обновлений – после каждого документа или заданного числа документов, после каждого термина (d,w) или заданного числа терминов, после каждого вхождения термина.

В алгоритмеPLSA-EM обновления происходят после каждого прохода коллекции.

В алгоритмеPLSA-GEM моменты обновления выбираются на шаге 9. В экспериментах на достаточно больших коллекциях частые обновления ускоряют сходимость.

При обновлении после каждого термина или каждого вхождения можно не хранить значения φ_{wt}, θ_{td}, а вычислять их каждый раз как частное двух счетчиков [4].

Недостатком алгоритма PLSA-GEM является необходимость хранить массив значений $\hat{n}_{dwt}$,t ∈ T для каждого термина (d,w) [4].

Модель pLSA имеет большое количество параметров, и она может быть склонна к переобучению, особенно если количество тем велико и данных для обучения недостаточно. Решением данной проблемы может стать небольшая модификация E-шага EM-алгоритма.

Другая трудность заключается в том, что алгоритм не разделяет тематические и фоновые компоненты смеси тем. Также модель не учитывает модальности данных, что может привести к потере информации и плохим результатам.Кроме того, модель pLSA не позволяет управлять разреженностью. Действительно, если в начале работы алгоритма $\varphi_{wt} = 0$ или $\theta_{td} = 0$, то и после завершенияработы алгоритма значения этих параметров останется равным 0. Этипроблемы могут быть решены с помощью регуляризации модели pLSA [7].

Существуют несколько альтернативных способов разреживания распределений p(t|d,w).Например, стохастический EM-алгоритм, сэмплирование Гиббса, робастный EM-алгоритм, разреживающийEM-алгоритм. Согласно статье [4] о модификации EM-алгоритма для вероятностного тематического моделирования, среди рассмотренных эвристик только сглаживание разреживание являются взаимно исключающими. В задачах машинного обучения им соответствуют два альтернативных подходя к понижению размерности - отбор признаков и регуляризация [4].

В современных исследованиях по тематическому моделированию преобладают методы регуляризации, которые помогают управлять сложностью модели и предотвращать переобучение, что делает их важным инструментом для анализа текстовых данных.

Далее рассмотрим модель LDA.

Латентное размещение Дирихле-широко известная вероятностная генеративная модель, которая используется для анализа тематической структуры в коллекции документов. Подобный подход схож с pLSAс той разницей, что в LDA накладываются дополнительные ограничения на вид

распределений φt~Dir(β) и θd~Dir(α) [25]. Это приводит к различным модификациям М-шага. Самая простая и популярная из них следующая (4) [7]:

$$\varphi_{wt} \infty n_{wt} + \beta_{w,} \theta_{td} \infty n_{td} + \alpha_t, \qquad (4)$$

что совпадает с регуляризатором сглаживания в подходе ARTM.

Латентное(скрытое) распределение Дирихле - это статистическая и графическая модель, которая используется для получения взаимосвязей между несколькими документами в корпусе. LDA использует алгоритм Гиббса с дополнительным расчетом уровня перплексии для оценки параметров модели.

Алгоритм Гиббса в LDA является итеративным алгоритмом, который позволяет оценивать распределения слов в темах и распределений тем в документах. Он основывается на предположении, что каждое слово в документе порождается из определенной темы [26]. Поэтому этот алгоритм позволяет оценивать вероятности тем и слов в темах из наблюдаемых данных (корпуса текстов).

Таким образом, мы можем получить гораздо более четкое представление о том, как связаны темы.

LDA является одной из наиболее распространенных моделей тематического моделирования в области обработки естественного языка, и было разработано множество расширений и модификаций для моделирования различных явлений. LDA является гибкой моделью тематического моделирования, которая может быть адаптирована для учета различных типов данных и особенностей коллекции документов. [8].

Тем не менее, построение комбинированных и многоцелевых тематических моделей остается сложной проблемой в байесовском подходе из-за сложного вывода в случае несопряженного предшествующего. Этот открытый вопрос мало обсуждается в литературе.

Другая трудность заключается в том, что предшествующий Дирихле

противоречит естественным предположениям о разреженности. Документ обычно содержит небольшое количество тем, а тема обычно состоит из небольшого количества терминов, специфичных для предметной области.

Следовательно, большинство слов и тем должны иметь нулевые вероятности. Разреженность помогает экономить память и время при моделировании больших текстовых коллекций.

Для решения вышеуказанных проблем мы вводим небайесовскийполувероятностный подход - аддитивную регуляризацию тематических моделей (ARTM). Изучение тематической модели из коллекции документов представляет собой некорректную задачу приближенной стохастической факторизации матрицы, которая имеет бесконечный набор решений.

Далее рассмотрим подход ARTM.

ARTM - это метод тематического моделирования, который использует аддитивную регуляризацию для учета различных свойств данных, таких как корреляция между темами, иерархическая структура или частые слова-спам. Он позволяет более эффективно учитывать специфические особенности данных и извлекать более точные и интерпретируемые темы.

Аддитивная регуляризация отличается от байесовского подхода в нескольких аспектах.

Во-первых, мы не стремимся построить полностью генеративную вероятностную модель текста. Многие требования к тематической модели могут быть более естественно формализованы в терминах критериев оптимизации, а не предварительных распределений. Регуляризаторы могут вообще не иметь вероятностной интерпретации. Структура регуляризованных моделей настолько проста, что их представление и объяснение в терминах графических моделей больше не требуется. Таким образом, ARTM придерживается тенденции избегать чрезмерных вероятностных допущений при обработке естественного языка.

Во-вторых, мы используем регуляризованный алгоритм максимизации

математического ожидания (EM) вместо более сложного байесовского вывода. Мы не используем сопряженные априоры, интегрирования и вариационные приближения. Несмотря на эти фундаментальные различия, оба подхода часто приводят к одинаковым или очень похожим алгоритмам обучения, но в ARTM вывод намного короче.

В-третьих, ARTM значительно упрощает как проектирование, так и вывод многоцелевых тематических моделей. На этапе проектирования мы формализуем каждое требование к модели в виде регуляризатора - критерия, который необходимо максимизировать. На этапе вывода мы просто дифференцируем каждый регуляризатор по параметрам модели.

Распределения φwt и θtd, получаемые при использовании модели PLSA, могут быть слабо разрежены, что может быть нежелательно при решении практических задач. Одним из способов борьбы со слабой разреженностью является применение методов регуляризации, которые позволяют ограничить сложность модели и уменьшить склонность к переобучению. Основная идея ARTM заключается в том, чтобы обеспечить гибкий способ добавления некоторой дополнительной информации о задаче к оптимизируемой функции правдоподобия [7]. Делается это с помощью взвешенной суммы критериев регуляризации Ri (5):

$$R(\Phi,\Theta) = \sum_i \tau_i R_i(\Phi,\Theta)\,, L(\Phi,\Theta) + R(\Phi,\Theta) \rightarrow \max_{\Phi,\Theta,} \qquad (5)$$

где $\tau \geq 0$ – коэффициенты регуляризации, регулируют силу действия регуляризатора и настраиваются экспериментально.

Для обучения тематической модели применяется EM-алгоритм.

Алгоритм ARTM-EM: EM-алгоритм для ARTM:

Вход: коллекция документов D, число тем |T|;

Выход: Θ и Φ;

Шаг 1: инициализировать вектор-столбцы φt, θd случайным образом;

Шаг 2: повторять

Шаг 3: обнулитьn̂wt, n̂dt длявсеxd ∈ D, w ∈ W, t ∈ T;

Шаг 4: для всеxd ∈ D, w ∈ d

Шаг 5: p(w | d):=$\sum_{t\in T} \varphi_{wt}\theta_{td}$;

Шаг 6: для всеxt ∈ T

Шаг 7: $p(t|d,w) = \varphi_{wt}\theta_{td}/p(w \mid d)$;

Шаг 8: увеличитьn̂wt, n̂td,нandwtp(t | d,w);

Шаг 9: $\varphi_{wt} \infty (n_{wt} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}})_+$ для всеxw ∈ W, t ∈ T;

Шаг 10: $\theta_{td} \infty (n_{td} + \theta_{td}\frac{\partial R}{\partial \theta_{td}})_+$ для всеxd ∈ D, t ∈ T;

Шаг 11:пока Θ и Φ не сойдутся;

Ограничения подхода APTM:

Коэффициенты регуляризации приходится подбирать вручную. Автоматическая коррекция стратегий регуляризации в APTM пока является открытой проблемой.

В частности, в ARTM используются следующие типы регуляризаторов: регуляризаторы сглаживания, разреживания и декоррелирования.

Регуляризация разреживания тем (Sparsity): заставляет модель выбирать только несколько тем для моделирования каждого документа. Это сделано для того, чтобы получить более интерпретируемые темы, т.к. каждая тема будет отвечать только за определенный аспект документа.

Разреживающий регуляризатор (6):

$$R(\Phi,\Theta) = -\beta_0 \sum_{t\in T}\sum_{w\in W}\beta_w \ln \varphi_{wt} - \alpha_0 \sum_{d\in D}\sum_{t\in T}\alpha_t \ln \theta_{td} \rightarrow^{max}_{\Phi,\Theta} \quad (6)$$

Сглаживание распределений терминов в темах. Используется для выделения фоновых тем, собирающих общую лексику языка или общую лексику данной коллекции.

Сглаживающий регуляризатор (7):

$$R(\Phi,\Theta) = \beta_0 \sum_{t\in T}\sum_{w\in W}\beta_w \ln\varphi_{wt} + \alpha_0 \sum_{d\in D}\sum_{t\in T}\alpha_t \ln\theta_{td} \rightarrow^{max}_{\Phi,\Theta} \quad (7)$$

Декорреляция тем как минимизация ковариаций между столбцами φt и φs матрицы Φ (8):

$$R(\Phi) = -\tau \sum_{t\in T}\sum_{s\in\frac{T}{t}}\sum_{w\in W}\varphi_{wt}\,\varphi_{ws} \rightarrow^{max}_{\Phi}, \quad (8)$$

$$\tau \geq 0 - \text{коэффециент регуляризации}$$

Декоррелирующий регуляризатор (8) стремится уменьшить пересечение между распределениями слов по темам φt. Это достигается путем добавления штрафа за корреляцию между темами в функцию потерь модели. Уменьшение пересечения между распределениями слов по темам, в свою очередь, повышает различность тем.

Различные регуляризаторы могут выполнять различные задачи при тематическом моделировании.

Они позволяют уменьшать веса тем, которые мало связаны с документами в коллекции, тем самым улучшая различность тем, также могут быть использованы для устранения разрывов в распределениях термов или тем, тем самым улучшая качество моделирования.

И если необходимо, выполнять тематическую сегментацию - разделение текста на предложения [2], строить иерархические модели и т.д.

Подход ARTM позволяет комбинировать различные регуляризаторы в одну общую модель путем преобразования М-шага [7].

Для удобства сравнения и оценки характеристик алгоритмов тематического моделирования используем таблицу 2.

Таблица 2 – Сравнение алгоритмов тематического моделирования

Метод ТМ	Алгоритм	Преимущества	Недостатки
PLSA	PLSA использует EM-алгоритм для оценки параметров модели.	PLSA разбивает текст на темы, и каждому слову приписывает вероятность присутствия в каждой из тем.	Не учитывает порядок слов в документе и не может моделировать зависимости между темами.
LDA	LDA использует алгоритм Гиббса с дополнительным расчетом уровня перплексии для оценки параметров модели.	LDA также использует вероятностное моделирование, но вместо условных вероятностей слов в контексте каждой темы, LDA представляет каждый документ как вероятностное распределение тем.	LDA может порождать темы, которые не имеют значения для пользователя, или не учитывать важные аспекты документов.
ARTM	ARTM использует регуляризаторы.	Помимо выявления тем, ARTM может учитывать другие характеристики, такие как содержание и спецификация тематических составляющих, а также включать внешние факторы, такие как источники текстов.	ARTM требует правильной настройки параметров моделирования, иначе результаты могут быть искажены или неверными.

Как показывает анализ, правильный подбор алгоритма должен обеспечивать как высокую точность моделирования, так и возможность интерпретации результатов.

Выводы по главе 2

Вторая глава посвящена обзору и анализу алгоритмов тематического моделирования.

Результаты проделанной работы позволили сделать следующие выводы:

– LDA является генеративной вероятностной моделью, которая используется для моделирования тематической структуры коллекции текстовых документов. Генеративная модель означает, что LDA моделирует процесс генерации коллекции текстовых документов, то есть описывает, как документы были созданы на основе скрытых тем и вероятностей слов в каждой теме.;

– PLSA также использует генеративную вероятностную модель, но учитывает только распределения слов в документах. Эта модель не учитывает возможные зависимости между словами из-за чего результаты моделирования не всегда точны;
– LDA и PLSA используют только вероятностные модели для выделения тем, в то время как ARTM может использовать как вероятностные, так и эвристические методы для нахождения наиболее вероятных тем.
– ARTM нацелен на улучшение качества тематического моделирования за счет введения регуляризации, которая позволяет учитывать различные априорные предположения о тематических структурах документов. Например, можно использовать регуляризацию для уменьшения перекрывающихся тем или для улучшения интерпретируемости тем;

Алгоритмы тематического моделирования являются важным инструментом для анализа текстовых данных. Они могут помочь выявить скрытые темы и отношения в больших наборах текстовых данных, что может помочь принимать более обоснованные решения на основе анализа данных.

Сравнительный анализ показывает, что LDA и PLSA имеют некоторые недостатки в точности тематического моделирования, особенно когда речь идет о коротких документах и когда слова в документах могут быть несвязанными. ARTM же более гибкий, учитывает все модельные факторы и, как показывают результаты сравнения, обеспечивает более точные прогнозы. В целом, ARTM представляет собой более прогрессивный метод тематического моделирования, который предлагает более точные, лучше различающиеся тематические модели, чем LDA и PLSA.

Глава 3 Программная реализация и сравнение алгоритмов тематического моделирования

Для реализации алгоритмов тематических моделей, будем использовать язык программирования Python.

Python- один из самых популярных языков программирования для машинного обучения и анализа данных. Акцент на удобочитаемости кода делает его очень простым в использовании, и у него есть большое сообщество участников, которые разработали широкий спектр вариантов для реализации моделей НЛП.

Одним из лучших вариантов тематического моделирования в Python является Gensim, надежная библиотека, которая предоставляет набор инструментов для реализации LSA, LDA и других алгоритмов тематического моделирования.

Исторически первые модели векторного представления слов были основаны на частотных распределениях слов. Одним из первых таких подходов был метод латентного семантического анализа (LSA), который использует матричное разложение для построения векторных представлений слов и документов.

Модели обучаемых векторных представлений слов (wordembeddings) получили большую популярность в последние годы благодаря развитию нейронных сетей и глубокого обучения. Основная идея таких моделей состоит в том, чтобы представить каждое слово в виде вектора в некотором пространстве, где семантически близкие слова имеют близкие векторы. Наиболее известным представителем таких моделей является семейство моделей word2vec[8].

Тематическое моделирование же развивалось параллельно. Вероятностный латентный семантический анализ (PLSA) был предложен Томасом Хоффманом в 1999 году [20]. Входными данными является матрица встречаемости слов, которая описывает коллекцию текстовых документов. На

выходе же модель выдает скрытые темы и их распределения в каждом документе.

Программная реализация.

Для начала нужно подготовить данные. Для всех текстов была проведена лемматизация - приведения слова к его базовой форме (лемме) с сохранением его смыслового значения [2], также убираем пунктуацию и стоп-слова. Удаление пунктуации и стоп-слов также позволяет уменьшить количество уникальных слов в коллекции и улучшить качество моделирования тем. Стоп-слова не несут семантической нагрузки и не влияют на определение темы документа, поэтому их удаление позволяет сосредоточиться на более значимых словах (рисунок 6).

```
with open('rus_stopwords.txt', 'r', encoding='utf-8') as f:
    sw = f.read().split('\n')

[ ]  with open('news.txt', 'r', encoding='utf-8') as f:
        text = f.read()

    texts = text.split('\n\n')

    punct = '[!"#$%&()*+,./:;<=>?@[\]^_`{|}~„“«»†*/\—–’]'
    nums = '[0-9]'

    texts = [re.sub('\n', ' ', text) for text in texts]
    texts = [re.sub(punct, '', text) for text in texts]
    texts = [re.sub(nums, '', text) for text in texts]

    tokenized_texts = []
    for text in texts:
        text = [w for w in text.split() if w not in sw]
        tokenized_texts.append(text)

    tokenized texts[0][:10]
```

Рисунок 6 – Удаление пунктуации и стоп-слов

Далее векторизуем тексты с помощью doc2vec.Doc2Vec можно рассматривать как расширение Word2Vec, целью которого является создание вектора представления документа. Word2Vec и Doc2Vec реализованы в

нескольких пакетах/библиотеках. Пакет python под названием gensim реализовал как Word2Vec, так и Doc2Vec.

Word2Vec - это широко используемый алгоритм, основанный на нейронных сетях, обычно называемый «глубоким обучением».

Используя большие объемы обычного текста, word2vec автоматически распознает взаимосвязи между словами. На выходе получаются векторы, по одному вектору на слово, с линейными соотношениями (рисунок 7).

```
print('Making dictionary...')
dictionary = corpora.Dictionary(tokenized_texts)
print('Original: {}'.format(dictionary))
dictionary.filter_extremes(no_below = 5, no_above = 0.9, keep_n=None)
dictionary.save('news.dict')
print('Filtered: {}'.format(dictionary))

print('Vectorizing corpus...')
corpus = [dictionary.doc2bow(text) for text in tokenized_texts]
corpora.MmCorpus.serialize('news.model', corpus)
```

Рисунок 7 – Векторизация текстов и представление в виде словаря и корпуса

Файл считывается в память по одному документу за раз, а не всю матрицу сразу.

Это позволяет вам обрабатывать корпуса, размер которых превышает объем доступной оперативной памяти, потоковым способом.

Определяем тематическую близость документов.

Распределение тем можно использовать для определения сходств в текстах.

На рисунке 8 представлен код реализации определения близости документов.

```
import random
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import style
style.use('ggplot')

# tf-idf векторах
tfidf = TfidfModel(corpus)
corpus_tfidf = tfidf[corpus]

# создание случайной выборки
sampling_tfidf = random.choices(corpus_tfidf, k=30)

# вычисление сходства по косинусной мере на tf-idf векторах
index = similarities.MatrixSimilarity(sampling_tfidf)
sims = index[sampling_tfidf]

plt.figure(figsize = (12,10))#размер карты
sns.heatmap(data=sims,annot= True, fmt=".2g", annot_kws={"size":6},
            cmap="Blues").set(xticklabels=[], yticklabels=[])#создание карты
plt.title("Матрица близости")
plt.show()
```

Рисунок 8 – Реализация определения близости документов

В нашем случае картинка была бы трудночитаемой, если бы мы взяли все документы, поскольку в нашем корпусе их более 3000. Поэтому для построения графика мы выберем 20 случайных (рисунок 9).

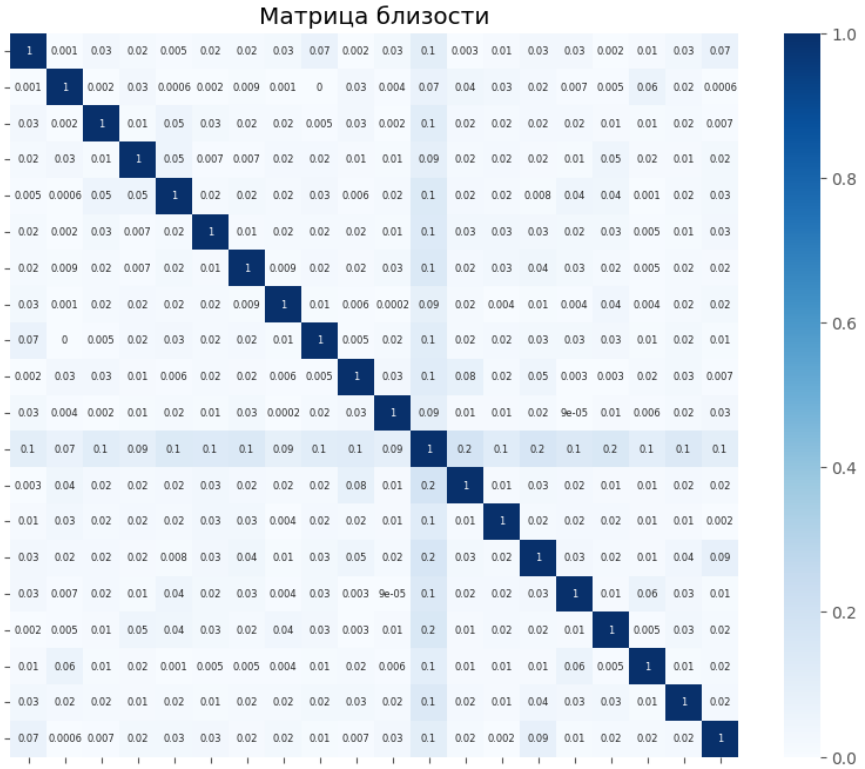

Рисунок 9 – Матрица близости документов

LDA. Данная модель реализована в библиотеке gensim. Один из ее плюсов в том, что можно дообучать готовую модель (в отличие от LSA и pLSA, где даже при добавлении одного нового документа приходится обучать модель с нуля) как показано на рисунке 10.

```
print("Training model...")

%time lda = ldamodel.LdaModel(corpus, id2word=dictionary, num_topics=20, chunksize=50, update_every=1, passes=2)
```

```
Training model...
CPU times: user 12.8 s, sys: 6.33 s, total: 19.2 s
Wall time: 12.7 s
```

Рисунок 10 – Обучение модели LDA

Визуализировать данные будем с помощью LDAVis. Используя pyLDAvis, данные LDA были разложены с помощью PCA (анализ главных компонентов), чтобы быть только 2-мерными. Таким образом, оно было сглажено для целей визуализации [13].

У нас есть круги, и центр каждого круга представляет положение нашей темы в пространстве скрытых объектов; расстояния между темами иллюстрируют, насколько похожи или не похожи темы, а площадь кругов пропорциональна количеству документов, содержащих каждую тему (рисунок 11).

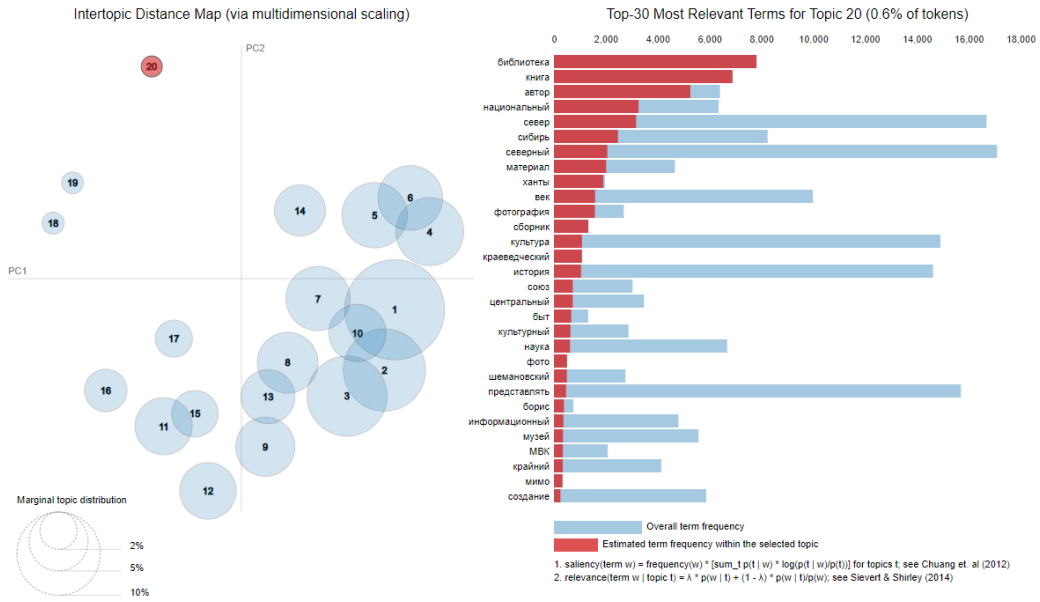

Рисунок 11 – Интерактивный график pyLDAvis

График LDAvis состоит из двух частей - двумерного ‘сглаженного’ отображения наших n-мерных данных LDA и интерактивной, изменяющейся горизонтальной гистограммы распределения сроков. Оба они показаны на рисунке 11. Следует отметить одну важную особенность: правая столбчатая диаграмма показывает термины в теме в порядке убывания релевантности, но столбики указывают частоту употребления терминов. Красный раздел представляет частотность термина исключительно в рамках конкретной темы; красный и синий представляют общую частотность термина в корпусе документов.

Чтобы визуализировать многомерные данные, нужно снизить их размерность. Например, векторы интенсивности пикселей, используемые для представления изображений, или векторы количества слов, используемые для представления документов, обычно имеют тысячи измерений.

Снижение размерности данных помогает облегчить анализ собранного набора данных, а именно сделать возможной его визуализацию. t-SNE начинается с определения сходства точек на основе расстояний между ними. Близлежащие точки считаются похожими, в то время как удаленные

считаются непохожими [22].

Возьмем 500 наиболее часто встречающихся слов и визуализируем их на графике (рисунки 12,13).

```
from nltk import FreqDist
from tqdm import tqdm_notebook as tqdm
from sklearn.manifold import TSNE

top_words = []


fd = FreqDist()
for s in tqdm(tokenized_texts):
    fd.update(s)

for w in fd.most_common(500):
    top_words.append(w[0])

print(top_words[:50:])
top_words_vec = model.wv[top_words]

%%time
tsne = TSNE(n_components=2, random_state=0)
top_words_tsne = tsne.fit_transform(top_words_vec)
```

Рисунок 12 – Код снижения размерности данных

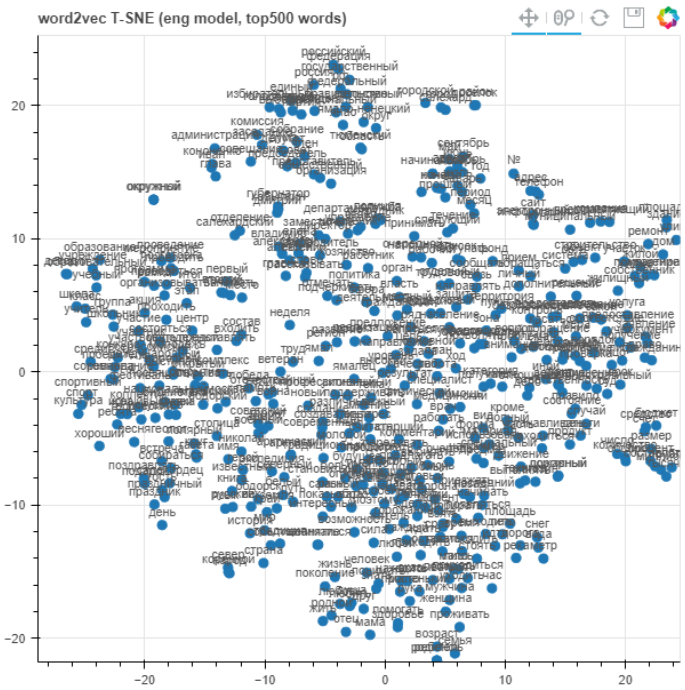

Рисунок 13 – Модель сходства слов

t-SNE может помочь нам определить разные способы написания одной и той же цифры или позволить вам находить синонимы слов/фразы со схожим значением при выполнении анализа NLP [22].

Зачем нужно что-то еще, если есть LDA? Проблема в том, что задача тематического моделирования имеет очень много (бесконечно много) решений, и для решения конкретных задач анализа текстов могут быть использованы и другие методы. Кроме того, важно выбирать подходящий метод, учитывая цель исследования и характеристики данных.

BigARTM является мощным инструментом для тематического моделирования, который позволяет обрабатывать большие коллекции текстовых данных и использовать различные методы и регуляризаторы для улучшения качества модели. Известно, что BigARTM хорошо сочетает в себе самые разные задачи, включая разбиение, сглаживание, декорреляцию тем и многие другие. Такая комбинация регуляризаторов значительно улучшает сразу несколько показателей качества практически без потери сложности.

BigARTM реализует несколько механизмов, которые позволяют настраивать и улучшать качество моделей тематического моделирования. Регуляризация служит для задания желаемых свойств тематической модели в виде оптимизационных критериев. Например, есть регуляризаторы, которые приводят к более интерпретируемым и разнообразным темам, улучшить качество модели, устранить шум в коллекции текстов и улучшить стабильность модели, учитывать временные факторы и т.д.

Мы будем использовать регуляризаторы SmoothSparsePhi, SmoothSparseTheta и DecorrelationPhi. Сначала импортируем все необходимые модули (BigARTM Python API в переменной PATH). Используем artm.ARTM – это полнофункциональный Python API для библиотеки BigARTM (рисунок 14).

```
%matplotlib inline
import glob
import os
import matplotlib.pyplot as plt

import artm

batch_vectorizer = None
if len(glob.glob(os.path.join('kos', '*.batch'))) < 1:
    batch_vectorizer = artm.BatchVectorizer(data_path='', data_format='bow_uci',
                                            collection_name='kos', target_folder='kos')
else:
    batch_vectorizer = artm.BatchVectorizer(data_path='kos', data_format='batches')
```

Рисунок 14 – Представление данных в виде одного класса

Библиотека Python API аналогично алгоритмам scikit-learn представляет входные данные в виде одного класса, называемого BatchVectorizer, рис. (14).

Dictionary - это объект BigARTM, содержащий информацию о коллекции (словарь, различные счетчики и значения, связанные с токенами). Предоставленный словарь будет использоваться для инициализации матрицы Ф (рисунок 15).

```
dictionary = batch_vectorizer.dictionary
topic_names = ['topic_{}'.format(i) for i in xrange(15)]

model_plsa = artm.ARTM(topic_names=topic_names, cache_theta=True,
                       scores=[artm.PerplexityScore(name='PerplexityScore',
                                                    dictionary=dictionary)])

model_artm = artm.ARTM(topic_names=topic_names, cache_theta=True,
                       scores=[artm.PerplexityScore(name='PerplexityScore',
                                                    dictionary=dictionary)],
                       regularizers=[artm.SmoothSparseThetaRegularizer(name='SparseTheta',
                                                                       tau=-0.15)])
```

Рисунок 15 – Создание словаря с информацией о коллекции документов

ARTM предоставляет возможность использовать все оценки BigARTM. Как только оценка была включена в модель, модель сохранит все свои значения, полученные во время каждого обновления матрицы Ф.

ARTM позволяет создавать свои собственные регуляризаторы, что

позволяет более гибко настраивать модель под конкретные задачи.Аддитивная регуляризация позволяет комбинировать различные регуляризаторы, управляя их вкладом в целевую функцию с помощью коэффициентов регуляризации. Это позволяет улучшить качество модели, учитывая несколько критериев одновременно, таких как интерпретируемость тем, разнообразие тем, разреженность матрицы термы-темы и т.д.ARTM также поддерживает условную регуляризация, которая позволяет задавать регуляризаторы только для определенных групп тем или документов, что позволяет более точно настраивать модель под конкретные задачи.

Оценка качества тематических моделей является сложной задачей, так как у нас нет информации о правильных или неправильных темах каждого документа.

Одним из распространенных методов оценки качества тематических моделей является перплексия. (perplexity), используемая для оценки качества модели языка в компьютерной лингвистике [8]. Она определяется как экспонента отрицательного логарифма правдоподобия модели на тестовой выборке. Чем меньше значение перплексии, тем лучше модель.

Переплексия - это метрика, позволяющая оценить качество модели, но не является единственной метрикой, используемой в тематическом моделировании.

Важным является и интерпретируемость тем, получаемых моделью, а также их релевантность и соответствие контексту.

Мы сравнили модели LDA, PLSAиARTM.

На рисунке 16 представлен график изменения переплексии в процессе обучения моделей.

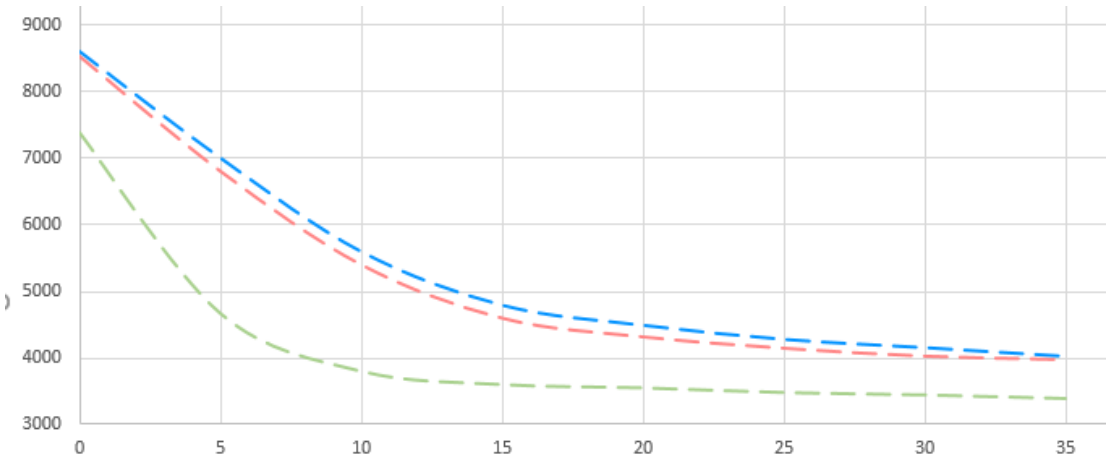

Рисунок 16 – График изменения переплексии в процессе обучения моделей

Итоговые значения переплексии представлены в таблице 3.

Таблица 3 – Значения перплексии для моделей LDA, PLSA, ARTM

Модель	Значение переплексии
LDA	3483.820
PLSA	4135.604
ARTM	4032.745

Модель LDA показала наилучшее значение перплексии, однако ее список топ-слов менее интерпретируемый. Модели PLSA и ARTM проявили сходные результаты по интерпретируемости, но ARTM показала более высокую точность по перплексии. [8].

Мы видим, что темы примерно равны с точки зрения интерпретируемости, но они более различны в ARTM. Темы в PLSA менее интерпретируемы, чем LDA.

Таким образом, можно выделить некоторые недостатки байесовского подхода в тематическом моделировании:

– Переформулировка критериев в терминах априорных распределений усложняет байесовский вывод;

– Обучение модели определяет не сами параметрв модели, а их распределение. Решается более трудная задача, чем это необходимо;
– Модели, которые используют априорное распределние Дирихиле, не моделируют каких-либо явлений естественного языка и не имееют убедительных лингвистических обоснований;
– Проблемы неединственности и неустойчивости.

Преимущества подхода ARTM:

– ARTM обучается значительно быстрее, чем байесовские модели, т.к. не требует итеративной оптимизации гиперпараметров и сэмплирования.;
– ARTM позволяет задавать несколько регуляризаторов, что позволяет более точно настраивать модель под конкретные задачи;
– Можно строить многоцелевые комбинированные модели;
– ARTM имеет простой и интуитивно понятный интерфейс, который не требует специальных знаний в области статистики и математики.

Существует множество различных методов и алгоритмов тематического моделирования. Для методов тематического моделирования создавались и разрабатывались различные дополнения, регуляризаторы для более точной работы алгоритмов. Некоторые алгоритмы более эффективны на небольших наборах данных, например, такие как анализ комментариев в социальных сетях или отзывов о конкретном продукте. Другие же, более эффективно работают с большим объемом текстов, такие как научные и новостные статьи.

В статье [19] «Использование тематического моделирования для кластеризации сообществ в социальной сети ВКонтакте» (Using topic modeling for communities clusterization in the VKontakte social network), рассматривается исследование на примере не больших наборов текста. Обучение модели ARTMзяняло меньше времени, чем обучение модели LDA. Все расчеты были выполнены на платформе GoogleColab.

Согласно [27] об аддитивной регуляризации, предположения Дирихле,

предшествующие в LDA, не соответствуют реальной разреженности распределения тем в документе. BigARTM не пытается построить полностью генеративную модель текста, в отличие от LDA; вместо этого он выбирает оптимизацию определенных критериев с помощью регуляризаторов. Эти регуляризаторы не требуют каких-либо вероятностных интерпретаций. Поэтому следует отметить, что с BigARTM проще формулировать многоцелевые тематические модели.

Задача тематического моделирования имеет бесконечное множество решений. Это дает нам свободу в выборе. Регуляризаторы дают возможность получить результат, удовлетворяющий одновременно нескольким критериям (таким как разреженность, интерпретируемость).

Выводы по главе 3

Третья глава посвящена программной реализации и тестированию алгоритмов тематического моделирования.

Результаты проделанной работы позволили сделать следующие выводы:

- темы примерно равны с точки зрения интерпретируемости, но они более различны в ARTM;
- LDA более устойчивый к шуму и более гибкий, чем PLSA.
- модель LDA показала наилучшее значение перплексии, чем PLSA и ARTM;
- темы в PLSA менее интерпретируемы, чем LDA, потому что PLSA не использует модель мешка слов, которая позволяет учесть взаимосвязи между словами.
- ARTM требуется большое количество вычислительных ресурсов, особенно при использовании больших коллекций текстовых данных.
- ARTM может быть более полезным, чем LDA или PLSA, когда требуется работать с многомодальными данными или когда требуется управлять качеством модели с помощью регуляризации.

– ARTM позволяет контролировать различные аспекты моделирования, такие как разреженность тем, совместная моделирование тем и метаданные (такими как авторы и даты документов.).
– ARTM объединяет преимущества LDA и PLSA и включает регуляризацию для улучшения точности модели. Он учитывает не только контекст и порядок слов в тексте, но также взаимодействия между темами. Это делает его более гибким и точным методом, чем LDA и PLSA.

В целом, выбор между LDA, PLSA и ARTM зависит от конкретных требований и задач, которые нужно решить. Если вы работаете с большими объемами текстовых данных и требуете точности, то LDA может быть лучшим выбором.

Если нужна большая точность и учет контекста и порядка слов, то PLSA может быть лучшим вариантом.

Если нужны гибкость и наилучшая точность, то ARTM может быть лучшим выбором.

Заключение

В ходе выполнения выпускной квалификационной работы на тему «Исследование алгоритмов тематического моделирования для новостных статей», проведено исследование, объектом которого являлись тематические модели, различные методы и алгоритмы тематического моделирования. Также были рассмотрены основы тематического моделирования, базовые модели.

Цель бакалаврской работы – исследование и реализация алгоритмов тематического моделирования.

В ходе данной работы поставлены и выполнены следующие задачи:

- выполнена постановка задачи исследования и проанализированы методы тематического моделирования: LSA, NMF, LDA, PLSA и ARTM. LSA не всегда достаточно точен в поиске семантических связей, так что два значения, которые являются близкими в контексте текста, могут быть несколько отделены друг от друга в новом пространстве. NMF требует тщательной предварительной обработки данных, так как может производить ошибки в анализе, если данные содержат шумы, выбросы или пропущенные значения. PLSA может справляться со многими проблемами, с которыми сталкиваются другие методы, такие как разреженность, шум и многомерность. LDA может автоматически определять темы в документах, не требуя от пользователя предварительно заданных тем (в отличие от других методов тематического моделирования). ARTM использует тот же формат мешка слов, что и LDA, но включает регуляризацию, что делает его более гибким и точным.;
- проанализированы алгоритмы тематического моделирования: LDA, PLSA и ARTM. Дано математическое описание алгоритмов. PLSA использует EM-алгоритм для оценки параметров модели. LDA использует алгоритм Гиббса с дополнительным расчетом уровня

перплексии для оценки параметров модели. ARTM позволяет структурировать модель и задавать дополнительные ограничения (регуляризации) на ее параметры.;

– выполнена программная реализация и тестирование алгоритмов тематического моделирования. Выполнена реализация данных алгоритмов на языке Python. Как показали результаты тестирования темы в PLSA менее интерпретируемы, чем LDA, а ARTM позволяет включать в модель различные типы ограничений и достигать более точной интерпретации тем. PLSA лучше учитывает контекст и порядок слов в тексте, что делает его более точным, чем LDA. Однако PLSA имеет ограничения в том, что он не учитывает отрицательные взаимодействия между темами. ARTM учитывает не только контекст и порядок слов в тексте, но также взаимодействия между темами.

Алгоритм LDA, как правило, является отправной точкой для тематического моделирования во многих случаях использования. BigARTM можно использовать как современную альтернативу.

Не существует идеального решения, объединяющего достоинства всех подходов, поскольку оптимизация одних характеристик может приводить к ухудшению других. Выбор компромиссного решения зависит от требований по времени адаптации под конкретную задачу, времени обучения моделей, вычислительным ресурсам, гибкости, масштабируемости и отказоустойчивости.

Результаты бакалаврской работы представляют научно-практический интерес и могут быть рекомендованы для анализа и программной реализации методов и алгоритмов тематического моделирования.

Список используемой литературы и используемых источников

1. Апишев М.А. «Эффективные реализации алгоритмов тематического моделирования» // Труды ИСП РАН. 2020. № 32:1. С. 137–152.

2. Бенгфорт Б., Билбро Р., Охеда Т. Прикладной анализ текстовых данных на Python. Машинное обучение и создание приложений обработки естественного языка. СПб.: Питер, 2019. 368 с.

3. Булатов В. Г., Ирхин И. А., Воронцов К. В., «Аддитивная регуляризация тематических моделей с быстрой векторизацией текста», Компьютерные исследования и моделирование, 12:6 (2020), 1515–1528

4. Воронцов К.В., Потапенко А.А. Модификации EM-алгоритма для вероятностного тематического моделирования // Машинное обучение и анализ данных. 2013. Т. 1, № 6. С. 657-686.

5. Коршунов А., Гомзин А. Тематическое моделирование текстов на естественном языке // Труды ИСП РАН. 2012 [Электронный ресурс]. URL: https://cyberleninka.ru/article/n/tematicheskoe-modelirovanie-tekstov-na-estestvennom-yazyke (дата обращения: 02.03.2023).

6. Митрофанова О.А. Моделирование тематики специальных текстов на основе алгоритма LDA. // Санкт-Петербург, 11—16 марта 2013 г.: Избранные труды. СПб.: Филологический факультет СПбГУ, а. 2014.-С. 220–233.

7. Сухарева А. В, Воронцов К. В., «Построение полного набора тем вероятностных тематических моделей», Интеллектуальные системы. Теория и приложения, 23:4, 2019, 7–23

8. Черкасов И.Е. «Сравнение алгоритмов тематического моделирования при определении тематик постов людей в социальной сети «ВКонтакте»» 2020, 45-49.

9. Bassiou N., Kotropoulos C. Online PLSA: Batch updating techniques including out-of-vocabulary words // Neural Networks and Learning Systems, IEEE Transactions on. 2014. Vol. 25, No. 11. P. 1953–1966.

10. Berry M. W., Browne M. Email surveillance using non-negative matrix

factorization. Computational and Mathematical Organization Theory. 2005. Vol. 11, P. 249–264.

11. Blei D., Carin L., Dunson D. Probabilistic Topic Models: A focus on graphical model design and applications to document and image analysis, IEEE signal. processing magazine. 2010. 27:6, 55.

12. Blei D., Ng A., Jordan M. Latent Dirichlet Allocation // Journal of Machine Learning Research. 2003.Vol. 3. P. 993-1022.

13. Carson, Sievert., Kenneth, E., Shirley. LDAvis: A method for visualizing and interpreting topics. (2014).63-70. doi: 10.3115/V1/W14-3110.

14. Chen Y., Zhang H., Liu R., Ye Z., Lin, J. Experimental explorations on short text topic mining between LDA and NMF based Schemes // Knowledge-Based Systems. 2019. Vol. 163. P. 1–13.

15. David Blei «Introduction to Probabilistic Topic Models|» //Communications of the ACM. 2012. P. 77–84.

16. Deerwester S., Dumais S. T., Furnas G. W., Landauer T. and Harshman R. Indexing by latent semantic analysis // J. Am. Soc. Inform. Sci.41. 1990. P. 391–407.

17. Dempster A.P., Laird N.M., Rubin D.B. Maximum Like lihood from Incomplete Data via the EM // Algorithm. Journal of the Royal Statistical Society. Series B (Methodological). 1977. Vol. 39. No. 1. P. 1-38.

18. Dudoit S. et al. Speed: comparison of discrimination methods for the classification of tumor using gene expression data // J. Amer. Stat. Assoc. 2002. Vol. 97. P. 77–87.

19. Gorshkov S., Ilyushin E., Chernysheva A., Goiko V., Namiot D. Using topic modeling for communities clusterization in the vkontakte social network // International Journal of Open Information Technologies. 2021. №5. URL: https://cyberleninka.ru/article/n/using-topic-modeling-for-communities-clusterization-in-the-vkontakte-social-network (дата обращения: 02.03.2023).

20. Hofmann T. Probabilistic Latent Semantic Indexing. In Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in

Information Retrieval, 1999. P. 50-57.

21. Kim J., He Y., Park H. Algorithms for non-negative matrix and tensor factorizations: a unified view based on block coordinate descent framework // J. Glob. Optim. 2014. Vol. 58. P. 285–319.

22. Maaten L.V., Hinton G.E. Visualizing Data using t-SNE // Journal of Machine Learning Research. 2008. Vol. 9. P. 2579-2605.

23. Nugroho R., Paris C., Nepal S., Yang J., Zhao W. A survey of recent methods on deriving topics from twitter: algorithm to evaluation // Knowl. Inf. Syst. 2020. Vol. 62. P. 2485–2519.

24. Steyvers M. Griffiths T. Probabilistic Topic Models. In T. Landauer, S. D. McNamara & W. Kintsch (ed.), 2007, Latent Semantic Analysis: A Road to Meaning. Laurence Erlbaum. (pp.427-448). Mahwah, New Jersey: Laurence Erlbaum Associates

25. Tan Y., Ou Z. Topic-weak-correlated latent dirichlet allocation, 7th International Symposium on Chinese Spoken Language Processing, IEEE, 2010. P. 224-228.

26. Teh Y.W., Newman D., Welling M. A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation. In Proc. of the 19th International Conference on Neural Information Processing, 2006, pp. 1353-1360.

27. Vorontsov, Konstantin V. and Anna Potapenko. «Additive regularization of topic models». Machine Learning 101 (2015): 303-323.

28. Yan, Xiaohui & Guo, Jiafeng & Lan, Yanyan & Cheng, Xueqi. A biterm topic model for short texts. WWW 2013 - Proceedings of the 22nd International Conference on World Wide Web. (2013) 1445-1456. 10.1145/2488388.2488514.