МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ М.В. ЛОМОНОСОВА

ФИЛИАЛ МГУ В ГОРОДЕ СЕВАСТОПОЛЕ

Направление подготовки «Прикладная математика и информатика»
01.03.02 (бакалавр)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

ИМЕНОВАНИЕ И СУММАРИЗАЦИЯ ТЕМ В ВЕРОЯТНОСТНОМ ТЕМАТИЧЕСКОМ МОДЕЛИРОВАНИИ С ИСПОЛЬЗОВАНИЕМ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ

Выполнил: Ильин Данила Владимирович
студент учебной группы ПМ-401

Научный руководитель: Зав. кафедрой
ММП, профессор РАН, д.ф.-м.н.,
Воронцов Константин Вячеславович

Севастополь – 2025

ОГЛАВЛЕНИЕ

СПИСОК ПРИНЯТЫХ СОКРАЩЕНИЙ, ОБОЗНАЧЕНИЙ И СИМВОЛОВ

LLM – большие языковые модели

ВКР – выпускная квалификационная работа

ВТМ – вероятностные тематические модели

ARTM – тематическое моделирование с аддитивной регуляризацией

АННОТАЦИЯ

В выпускной квалификационной работе предлагается метод автоматической абстрактивной суммаризации и именования тем для коллекции научных текстов, сочетающий в себе тематическое моделирование с аддитивной регуляризацией и генеративной способностью современных больших языковых моделей. Также, рассматривается метод автоматического оценивания получившихся вариантов на основе парного сравнения вариантов в подходе «LLM as a Judge», с дальнейшим построением статистической модели Брэдли – Терри для выбора наилучшего варианта.

Ключевые слова: тематическое моделирование, большие языковые модели, модель, коллекция научных текстов, мера Жаккара, метод.

ВВЕДЕНИЕ

Современный рост объемов научной информации затрудняет ориентацию и анализ больших коллекций научных публикаций. Даже при наличии специализированных поисковых систем и цифровых библиотек, исследователь сталкивается с необходимостью изучения множества релевантных, но объемных и разнородных текстов. Это определяет актуальность задач автоматического анализа и структурирования текстовой информации. Одним из перспективных направлений в этой области является тематическое моделирование, позволяющее выявлять скрытые тематики текстовых коллекций и формировать их краткие интерпретации без привлечения ручной разметки.

Тематическое моделирование относится к методам машинного обучения без учителя и представляет собой построение вероятностных моделей, описывающих распределение тем по документам и слов по темам. Одним из современных и активно развиваемых подходов является метод аддитивной регуляризации тематических моделей (ARTM), позволяющий гибко управлять свойствами модели, улучшать интерпретируемость тем и устойчивость результатов.

Другим мощным инструментом анализа и генерации текстов выступают большие языковые модели (Large Language Models, LLM), продемонстрировавшие высокую эффективность в задачах суммаризации, генерации аннотаций, переформулирования и логической интерпретации текста. Несмотря на их выразительность, LLM характеризуются высокой вычислительной затратностью и ограничениями на длину входных данных.

Актуальной задачей, возникающей при использовании тематических моделей, является автоматическое именование тем (topic labeling) и формирование кратких описаний (topic summarization), что позволяет сделать модель более понятной и полезной для пользователя. Задача суммаризации отдельных

тем, в отличие от классической многодокументной суммаризации, до сих пор не получила достаточного внимания в научной литературе.

Целью исследования является разработка и экспериментальная проверка метода абстрактивной суммаризации и именования тем, объединяющего в себе методы тематического моделирования с аддитивной регуляризацией и генеративной способностью больших языковых моделей.

Объектом исследования являются тематические модели, построенные на коллекции научных текстов, а также большие языковые модели.

Предмет исследования – методы автоматического именования и суммаризации тем с использованием ARTM и LLM.

Методы исследования включают: тематическое моделирование, получение векторных вложений текстов и их кластеризацию, генерацию текстов с использованием LLM, а также автоматическую оценку качества генераций.

Практическая значимость данной работы заключается в разработке нового подхода к суммаризации и именованию тем, направленного на повышение интерпретируемости результатов тематического моделирования при анализе текстовых коллекций. Предложен также метод автоматической оценки качества генераций без привлечения экспертов, что делает возможным масштабируемое применение. Полученные решения могут быть востребованы в системах научной аналитики, цифровых библиотеках, образовательных платформах и интеллектуальных поисковых системах.

РАЗДЕЛ I
ПОСТАНОВКА ЗАДАЧИ

Несмотря на активное развитие тематических моделей и языковых моделей в последние годы, проблема генерации человекопонятных названий и описаний тем остаётся недостаточно изученной. Анализ научной литературы показал, что подходы, объединяющие вероятностные тематические модели и большие языковые модели (LLM), не получили достаточного внимания в научной литературе.

Целью исследования является разработка и экспериментальная проверка метода абстрактивной суммаризации и именования тем, объединяющего в себе методы тематического моделирования с аддитивной регуляризацией и генеративной способностью больших языковых моделей.

Для достижения поставленной цели необходимо было решить следующие задачи:

– Изучить теоретические основы тематического моделирования и методов генерации текстов с помощью LLM.

– Проанализировать существующие подходы к автоматическому именованию и суммаризации тем.

– Построить тематическую модель с использованием ARTM на коллекции научных текстов.

– Реализовать алгоритм, объединяющий результаты тематического моделирования и генеративные возможности LLM для генерации названий тем и их описаний.

– Разработать автоматизированный метод оценки качества генераций без привлечения экспертов.

– Провести экспериментальную проверку качества предложенного подхода на корпусе научных публикаций.

Объектом исследования являются тематические и языковые модели, применяемые к коллекциям научных текстов.

Предметом исследования выступают методы автоматического именования и суммаризации тем, основанные на интеграции ARTM и LLM.

Научная новизна исследования заключается в создании метода суммаризации и именования тем с использованием совместной работы тематических моделей и LLM, и оцениванием результатов без привлечения экспертов.

РАЗДЕЛ II
ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ РАССМАТРИВАЕМОЙ ЗАДАЧИ

2.1 Обзор теории вероятностных тематических моделей

Пусть D – конечное множество текстовых документов, W – конечное множество всех употребляемых в них термов, T – конечное множество тем.

Таким образом, каждый документ $d \in D$ является последовательностью термов $d = \{w_1, \ldots, w_{n(d)}\}$, где $n(d)$ – количество термов в документе d и $w_i \in W$.

Будем называть последовательность троек (w_i, d_i, t_i) коллекцией документов:

$$\Omega_n = \{(w_i, d_i, t_i \mid i = 1, \ldots, n)\} \quad (2.1)$$

Определим начальные гипотезы:

– Гипотеза «мешка слов»:

Будем считать, что порядок термов в документах не имеет значения для определения тематики, что позволяет рассматривать каждый документ, как мультимножество – подмножество термов $d \subset W$, в котором каждый терм $w \in d$ повторяется n_{dw} раз.

– Гипотеза о вероятностном порождении данных:

Множество $\Omega = D \times W \times T$ является конечным вероятностным пространством с неизвестной функцией вероятности $p(d,w,t)$. Выборка троек вида (d_i,w_i,t_i) является коллекцией документов Коллекция документов это выборка троек вида (d_i,w_i,t_i), порождаемых случайно и независимо друг от друга из распределения $p(d,w,t)$.

– Гипотеза условной независимости.

Появление термов в документе по теме *t* зависит от темы, но не от документа, и описывается общим для всех документов распределением *p(w|t):*

$$p(w|d,t) = p(w|t) \tag{2.2}$$

Согласно формуле полной вероятности и гипотезе условной независимости, распределение термов в документе описывается при помощи распределений термов в темах и тем в документах:

$$p(w|d) = \sum_t p(w|t,d) \cdot p(t|d) = \sum_t p(w|t) \cdot p(t|d) \tag{2.3}$$

Введем новые обозначения и перепишем формулу:

$$p(w|d) = \sum_t \varphi_{wt} \cdot \theta_{td} \tag{2.4}$$

Вероятностная модель (2.4) описывает порождение коллекции по известным распределениям.

Задачей тематического моделирования является по заданной коллекции документов *D* требуется найти φ_{wt} и θ_{td}, при которых вероятностная модель хорошо приближает частотные оценки условных вероятностей:

$$\hat{p}(w|d) = p_{dw} = \frac{n_{dw}}{n_d} \tag{2.5}$$

Также, задачу тематического моделирования можно рассматривать как задачу приближенного низкорангового стохастического матричного разложения.

Перепишем (2.4) в матричном виде:

$$P \approx \Phi\Theta \tag{2.6}$$

Матрица P размеров |W| × |D| состоит из частот термов в документах, матрица Φ размеров |W| × |T| состоит из частот термов относительно тем, и матрица Θ размеров |T| × |D| состоит из частот тем в документах.

Во всех трех матрицах столбцы неотрицательны, нормированы и определяют дискретные вероятностные распределения. Такие матрица называются стохастическими. Ранг правой части не превышает числа тем, которое во много раз меньше количества термов и документов, а в левой части матрица, в общем случае, имеет полный ранг. Таким образом, матрица P, в общем случае, имеет полный ранг, поэтому не может быть в точности равняться правой части.

Запишем частотные оценки условных вероятностей:

$$p(d,w) = \frac{n_{dw}}{n} \tag{2.7}$$

$$p(d) = \frac{n_d}{n} \tag{2.8}$$

$$p(w) = \frac{n_w}{n} \tag{2.9}$$

$$p(w|d) = \frac{n_{dw}}{n_d} \tag{2.10}$$

В формулах (2.7) – (2.10) ndw – число вхождений терма w в документ d, nd – длина документа d в термах, nw – количество вхождений терма w во все документы коллекции, n – длина коллекции в термах.

Частотные оценки условных вероятностей, связанные со скрытой переменной t:

$$p(t) = \frac{n_t}{n} \tag{2.11}$$

$$p(w|t) = \frac{n_{wt}}{n_t} \tag{2.12}$$

$$p(t|d) = \frac{n_{td}}{n_d} \tag{2.13}$$

$$p(t|d,w) = \frac{n_{tdw}}{n_{dw}} \tag{2.14}$$

В формулах (2.11) – (2.14) ntdw – количество троек, в который терм w из документа d связан с темой t, ntd – количество троек, в которых термы документа d связаны с темой t, nwt – количество троек, в которых терм w связан с темой t, nt – число троек, связанных с темой t.

Заметим, что все оценки (2.11) – (2.14) можно выразить через ntdw:

$$n_{tdw} = n_{dw} p(t|d,w) \tag{2.15}$$

$$n_{td} = \sum_w n_{tdw} = \sum_w n_{dw} p(t|d,w) \tag{2.16}$$

$$n_{wt} = \sum_d n_{tdw} = \sum_d n_{dw} p(t|d,w) \tag{2.17}$$

$$n_t = \sum_d \sum_w n_{tdw} = \sum_d \sum_w n_{dw} p(t|d,w) \tag{2.18}$$

Задача называется корректно поставленной по Адамару, если ее решение существует, единственно и устойчиво. Задача низкорангового стохастического матричного разложения является некорректно поставленной, так как в общем случае решений существует бесконечно много: если $\Phi\Theta$ – решение, то и $(\Phi S)(S^{-1}\Theta)$ – решение, если S – невырожденная и при условии, что матрицы $S^{-1}\Theta$ и ΦS – стохастические.

Аддитивная регуляризация тематических моделей основана на максимизации логарифма правдаподобия и регуляризаторов $R_i(\Phi, \Theta)$ с неотрицательными коэффициентами регуляризации τ_i, с ограничениями неотрицательности и нормировки:

$$\sum_d \sum_w n_{dw} \ln \sum_t \varphi_{wt}\theta_{td} + R(\Phi, \Theta) \rightarrow max \quad (2.19)$$

$$R(\Phi, \Theta) = \sum_{i=1}^{k} \tau_i R_i(\Phi, \Theta) \quad (2.20)$$

$$\sum_w \varphi_{wt} = 1, \varphi_{wt} \geq 0 \quad (2.21)$$

$$\sum_t \theta_{td} = 1, \theta_{td} \geq 0 \quad (2.22)$$

Регуляризаторы необходимы для доопределения задачи, при помощи добавления к основному критерию дополнительных критериев. Регуляризаторы помогают учитывать специфику решаемой задачи и знания предметной области.

Теорема 2.1. Пусть функция $R(\Phi, \Theta)$ является непрерывна дифференцируема. *Тогда точка* (Φ, Θ) *локального экстремума задачи (2.19) с ограничениями (2.21) – (2.22) удовлетворяет системе уравнений со вспомогательными переменными* p_{tdw}*= p(t|d, w), если из решения исключить нулевые столбцы матриц* Φ, Θ*:*

$$p_{tdw} = norm_t(\varphi_{wt}\theta_{td}) \quad (2.23)$$

$$\varphi_{wt} = norm_w\left(n_{wt} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}}\right), n_{wt} = \sum_d n_{dw}p_{tdw} \quad (2.24)$$

$$\theta_{td} = norm_t\left(n_{td} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right), n_{td} = \sum_w n_{dw}p_{tdw} \quad (2.25)$$

Доказательство данной теоремы приводится в [1].

Для вычисления параметров модели используется регуляризованный ЕМ-алгоритм:

1. Выбираются начальные приближения $\varphi_{wt}, \theta_{td}$
2. Производятся вычисления по формуле (2.24)
3. Производятся вычисления по формуле (2.25)
4. Повторять шаги 2) – 3) пока решение не сойдется.

2.2 Обзор теории нейросетевых моделей для решения задач NLP

В данном разделе будут рассмотрены механизмы внимания и архитектура Transformer нейронных сетей, предназначенных для решения задач обработки естественного языка [2–4].

При обработке естественного языка важным этапом является выбор представления слов в векторном пространстве – реализация векторных вложений (embeddings). Вложение слова – это вектор некоторой размерности с вещественными компонентами. Компоненты этого вектора подбираются на этапе обучения, так, чтобы близкие, в семантическом смысле, слова, находились близко и в векторном пространстве.

При построении векторных вложений слов существует проблема неоднозначности значения из-за контекста. К примеру, взяв исключительно слово «замок» нельзя однозначно определить его смысл. Но если мы рассмотрим слово в контексте целого предложения, к примеру, «дверь не открывается, видимо, поломался замок» – мы понимаем, что речь здесь явно не про здание.

Механизм внимания обновляет вектор вложения каждого слова, добавляя к нему информацию об окружении этого слова, тем самым, помогая решить проблему неоднозначности.

Пусть есть три матрицы: Q – матрица запросов (query), K – матрица ключей (key), V – матрица значений (value). Тогда функцию внимания можно определить следующим образом:

$$Attn(Q, K, V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{E}}\right) \cdot V \qquad (2.26)$$

Операция *softmax* принимает на вход вектор из действительных чисел и превращает его в вектор распределения вероятностей, сумма которых равна единице:

$$softmax(x_i) = \frac{e^{x_i}}{\sum_i e^{x_i}} \quad (2.27)$$

Размерности всех матриц есть $N \times E$, где N – количество векторов вложений для слов предложения, E – размерность вектора вложений.

Рассмотрим теперь механизм само – внимания (self-attention):

$$Atnn(V, V, V) = softmax\left(\frac{V \cdot V^T}{\sqrt{E}}\right) \cdot V \quad (2.28)$$

Самовнимание позволяет модели определить важность каждой части входной последовательности относительно всех слов в предложении, что позволяет учесть контекст.

Следующим важным этапом развития механизма внимания стало изобретения механизма многоголового внимания (multihead attention):

$$MultiHead(Q, K, V) = Concat(h_1, \dots, h_H) \cdot W^O \quad (2.29)$$

$$h_i = Attn(Q \cdot W_i^Q, K \cdot W_i^K, V \cdot W_i^V) \quad (2.30)$$

Каждая голова h_i работает со своей частью входной последовательности размера E/H, где H – количество голов. Для поданных на вход матриц производится линейное преобразование, а далее, все h_i конкатенируются между собой и преобразуются при помощи матричного умножения на W^O, и на выходе получается итоговая матрица.

Таким образом, многоголовое внимание обращает внимание на разные части входной последовательности, что позволяет модели лучше извлекать локальную информацию о контексте.

Перейдем к рассмотрению архитектуры глубокой нейронной сети Transformer:

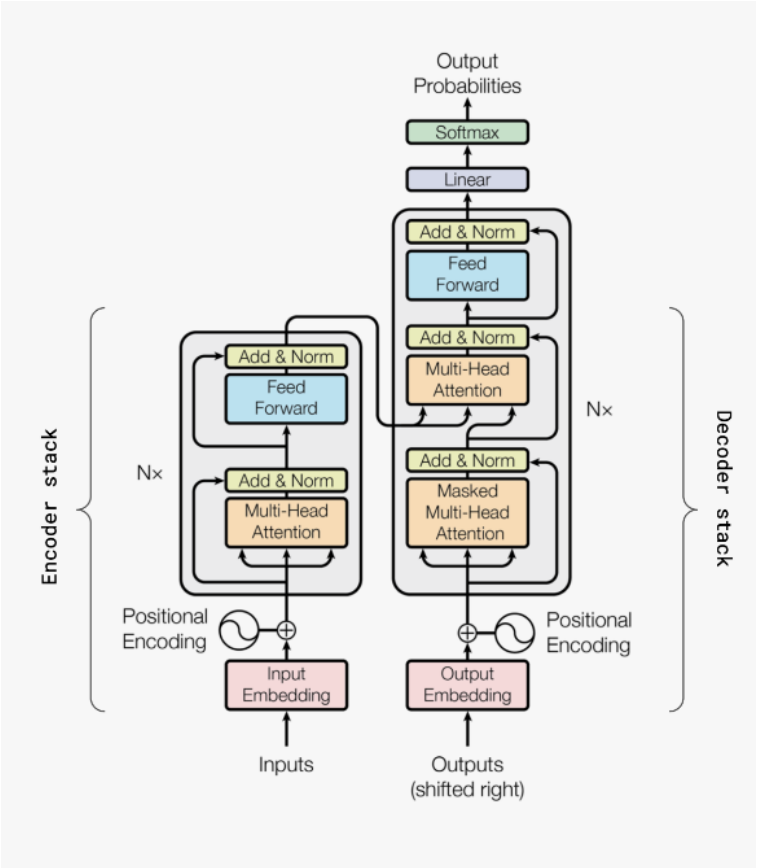


Рисунок 2.1 Архитектура Transformer

Источник: [11]

Архитектура Transformer состоит из двух частей: энкодера и декодера. Устройства блока энкодера представлено на схеме слева, устройство блока декодера представлено на правой части схемы.

Рассмотрим подробнее структуру энкодер блока:

На вход энкодеру подается тензор размера $N \times B \times E$, где N – количество слов входной последовательности, B – число примеров, которые обрабатываются одновременно (размер батча), E – размерность векторов вложений.

Полезно учитывать еще и информацию по поводу положения слова во входной последовательности, поэтому в архитектуре Transformer используется позиционное кодирование. В статье [2] используются следующие позиционные вложения:

$$PosEmb(pos, 2i) = \sin\left(\frac{pos}{10000^{\frac{2i}{E}}}\right) \quad (2.31)$$

$$PosEmb(pos, 2i+1) = \cos\left(\frac{pos}{10000^{\frac{2i}{E}}}\right) \quad (2.32)$$

Полученные позиционные вложения складываются со входными вложениями и подаются дальше по сети.

Далее, применяется многоголовое самовнимание, с наборами матриц преобразований W_i^Q, W_i^K, W_i^V и W^O, где $i \in [1, H]$, H – количество голов. Далее результат передается в слой Add&Norm, который работает следующим образом: выход многоголового внимания складывается с исходным тензором и нормируется при помощи LayerNorm:

$$LayerNorm(x) = \frac{x - E[x]}{\sqrt{Var[x] + \varepsilon}} \cdot \gamma + \beta \quad (2.33)$$

где $E[x]$ – математическое ожидание, $Var[x]$ – дисперсия, γ, β – обучаемые параметры, ε – некоторое маленькое число, необходимое для того, чтобы избежать деление на ноль.

Получившейся тензор поступает на вход полносвязного слоя *FFN* с двумя линейными преобразованиями и функцией активации *ReLU:*

$$ReLU(x) = \max(0, x) \quad (2.34)$$

$$FFN(x) = ReLU(x \cdot W_1 + b_1) \cdot W_2 + b_2 \quad (2.35)$$

Далее выход полносвязного слоя подается на вход слою *Add&Norm,* где он суммируется с исходным тензором и нормируется при помощи *LayerNorm.* Таким образом, получается тензор исходной формы $N \times B \times E$.

Рассмотрим устройство блока декодера:

Первоначально, на вход блоку подается входная целевая последовательность для которой вычисляется ее последовательность векторов вложений для каждого слова.

Затем, получившийся результат складывается с позиционными вложениями для извлечения дополнительной информации о позиции каждого слова, и отправляется на вход следующему слою.

Далее, векторы подаются на вход слою многоголового внимания для уточнения контекстного смысла слов, но в отличии от подобного слоя в энкодере, данный слой применяется с маской.

Обсудим подробнее принцип маскирования. Первый случай, когда используется маскирование – формирование последовательности входных предложений, которые имеют разные длины. Тогда предложения длины меньше максимальной длины предложения в этой последовательности, заполняются служебным токеном <PAD>, с выделенным ему индексом 0. Для того, чтобы нейронная сеть не обращала на них внимание, используются маска, которая ставит значение True там, где стоит токен <PAD> и False в ином случае. Технически, это означает, что в матрице $Q * K$ в колонках ключей для слов <PAD> будет стоять значение $-inf$, так как после использования функции *softmax,* в данных позициях будет стоять 0.

Второй случай – когда модели нельзя использовать информацию, заглядывая в будущее. К примеру, пусть на очередном шаге в блоке декодера был обработан токен w_i, тогда нейронная сеть имеет право обрабатывать все токены до $w_1,\ldots,w_{i-1}$, но токены стоящие после i-ого модель не должна видеть. Технически, это реализовывается через замещение всех элементов выше диагонали в матрице $Q * K$ на $-inf$.

Таким образом, результат добавления позиционных вложений к исходным вложениям подается на вход маскированному многоголовому самовниманию, далее выход суммируется со входом и нормируется в слое *Add&Norm*.

Далее, начинает свою работу еще один блок многоголового внимания, где в качестве запроса (матрица *Q*) выступает выход предыдущего слоя, а ключами и значениями (матрицы *K, V*) выступают выходы энкодера. Выход снова суммируется со входом и нормируется в слое Add&Norm.

В окончании блока декодера работает слой полносвязной сети из двух слоев с функцией активации *ReLU* (аналогично экнодеру).

Таким образом, при помощи совместной работы блока энкодера и декодера формируется блок Transformer. Таких блоков может быть несколько, они наслаиваются друг на друга. В окончании, стоит линейный слой размером совпадающим с размером словаря с функцией активации *softmax,* для вычисления следующего токена для генерации.

Отметим, что описанная схема работы представляет собой схему этапа обучения, во время которого, глубокая нейронная сеть уточняет свои веса при помощи решения задачи машинного перевода. На вход блоку энкодера подается предложение на одном языке, на вход блоку декодера подается предложение на другом языке, перевод которого, модель и должна сгенерировать.

В режиме инференса (evaluate) может включать в себя как работу лишь одного из блоков, так и их совместную работу.

2.3 Решение задачи именования тем и суммаризации.

Одним из ключевых вызовов при использовании тематических моделей является необходимость интерпретации выделенных тем. Традиционно каждая тема представляется списком наиболее вероятных слов, однако подобное представление часто оказывается недостаточным для точного понимания темы человеком. Поэтому особое внимание в научной литературе уделяется задачам автоматического именования и суммаризации тем.

Задача автоматического именования темы (automatic topic labeling) состоит в том, чтобы подобрать для каждой темы лаконичный релевантный заголовок из заранее заготовленного списка фраз-кандидатов, проследив, чтобы разные темы не получили слишком похожие заголовки. Впервые эта задача была поставлена и решена в [5]. В последующих работах решение немного улучшалось, не меняясь концептуально [6–8]. Краткий заголовок темы полезен для навигации по списку тем и изучения модели, однако для понимания темы пользователем его также недостаточно.

Другой подход мог бы заключаться в автоматической суммаризации темы, когда генерируется краткое изложение или аннотация темы в виде связного текста. Задача суммаризации текстов имеет длинную предысторию и хорошо исследована, начиная с 50-х годов [9] и заканчивая современными обзорами [10, 11]. В частности, в 2019 году была опубликована статья [12], аннотация которой завершалась фразой: «Note: The abstract above was not written by the authors, it was generated by one of the models presented in this paper». Несмотря на успехи методов суммаризации и очевидную практическую востребованность, задача суммаризации отдельных тем в тематических моделях до сих пор никем не решалась, и даже не ставилась. Определённый шаг сделан в недавней работе [13], где ставится задача выделения тематических фраз, содержащих ключевые слова заданной темы. Поиск таких фраз несомненно полезен, но не является полноценной суммаризацией темы.

При этом есть немало исследований, в которых тематическое моделирование используется как вспомогательный инструмент на одном из этапов обработки данных для суммаризации коллекции документов (multi-document summarization) [14, 15]. Идея заключается в том, чтобы с помощью тематической модели выделить основные темы текстовой коллекции, затем из каждой темы отобрать наиболее репрезентативные фразы, и тем самым повысить полноту суммаризации. Очевидно, что используемые в этих работах методы поиска и выделения фраз по заданной теме могут быть применены также и для суммаризации темы.

Серьёзной проблемой в области суммаризации текстов является оценивание качества решения. Общепринятым способом измерения качества уже более 20 лет остаётся метрика ROUGE [16, 17], хотя её недостатки давно осознаны и хорошо известны сообществу. Эта метрика позволяет сравнивать автоматическую суммаризацию с одной или несколькими суммаризациями, заранее написанными людьми. Сравнение основано на подсчёте совместно используемых слов или словосочетаний. При этом никак не оценивается связность, логичность, отсутствие фактических, речевых или стилистических ошибок.

При оценивании качества именования тем в [5] и последующих работах использовалось исключительно экспертное оценивание. К сожалению, такой подход не позволяет масштабировать экспериментальные исследования моделей. Для каждой модели именования (или суммаризации) приходится привлекать экспертов, что долго, дорого и субъективно. Отдельной проблемой является создание масштабируемой меры качества, которую можно было бы один раз откалибровать, хотя бы для заданной текстовой коллекции, чтобы впоследствии по полученной размеченной выборке оценивать и сравнивать различные модели.

Таким образом, несмотря на значительный интерес к интерпретируемости тем, задачи автоматического именования и суммаризации тем до сих пор остаются открытыми и требуют разработки новых, более точных и масштабируемых решений.

2.4 Вывод по разделу 2

Таким образом, в данном разделе были рассмотрены теоретические материалы по вероятностным тематическим моделям и глубоким нейронным сетям, основанным на архитектуре Transformer.

Также, были рассмотрены работы, связанные с решением задач именования и суммаризации тем. Были выявлены актуальные проблемы, такие как, отсутствие методов автоматического оценивания без экспертов, что затрудняет

масштабировать исследования, и проблема малого интереса к вопросу абстрактивной суммаризации тем.

Таким образом, настоящее исследование направлено на создание методов автоматического именования и суммаризации тем в вероятностных тематических моделях.

РАЗДЕЛ III
ИССЛЕДОВАНИЕ И ПОСТРОЕНИЕ РЕШЕНИЯ ЗАДАЧИ

3.1 Схема практической части исследования.

Для достижения поставленных целей был разработан план практической части исследования:

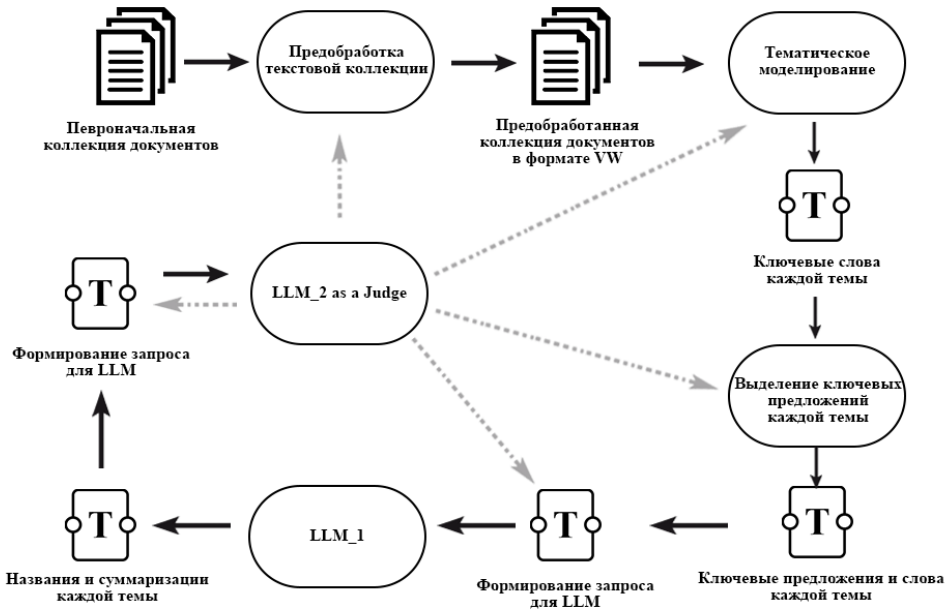


Рисунок 3.1 Схема исследования

Источник: авторская схема

Практическая часть состоит из следующих этапов:

– Этап предобработки текстовой коллекци;

– Этап тематического моделирования, суть которого, заключается в разработке вероятностных тематических моделей при помощи добавления разных регуляризаторов;

– Следующий этап – это получение ключевых слов для каждой темы (семантические ядра) и получение ключевых предложений для того, чтобы

сформировать как можно больше полезной информации, необходимой для суммаризации и именования тем;

– Далее следует этап получения пар (название темы, суммаризация темы) при помощи LLM. Для данных целей были использованы разные стратегии формирования запросов: zero–shot, few–shot и iterative–improving;

– После того, как для всех тем и разных стратегий были собраны все пары (название, суммаризация) наступает этап парного сравнения вариантов относительно каждой из тем, при помощи моделей-оценщиков («LLM as a Judge»);

– Финальным этапом служит построение статистической модели Брэдли–Терри для выявления наилучшей комбинации модель–стратегия запроса.

Рассмотрим подробно каждый из этапов.

3.2 Регуляризаторы.

В данной работе были использованы следующие регуляризаторы: разреживание Φ, разреживание Θ и декоррелирование тем.

Регуляризатор разреживания Φ:

$$R_{\varphi} = \tau \cdot \sum_{w,t} \log(\varphi_{wt}) \tag{3.1}$$

Также, рассмотрим производную регуляризатора по φ, которая используется в EM – алгоритме:

$$\frac{\partial R_{\varphi}}{\partial \varphi_{wt}} = \frac{\tau}{\varphi_{wt}} \tag{3.2}$$

Цель данного регуляризатора сделать распределение слов по темам разреженным: каждый терм должен быть связан с немногими темами и темы должны содержать как можно больше ключевых слов, и как можно меньше

«шумовых». Достигается это посредством штрафа за плотные распределения (в данном случае, $\tau < 0$).

Рассмотрим регуляризатор разреживания Θ:

$$R_\theta = \tau \cdot \sum_{t,d} \log(\theta_{td}) \tag{3.3}$$

Также, рассмотрим производную регуляризатора по θ, которая используется в EM – алгоритме:

$$\frac{\partial R_\varphi}{\partial \theta_{td}} = \frac{\tau}{\theta_{td}} \tag{3.4}$$

Цель данного регуляризатора сделать распределение тем по документам разреженным: каждый документ должен относиться к небольшому количеству тем. Достигается это посредством штрафа за плотные распределения (в данном случае $\tau < 0$).

Рассмотрим регуляризатор декоррелирования тем:

$$R(\Phi) = -\frac{\tau}{2} \cdot \sum_{t \in T} \sum_{s \in T \setminus t} \sum_{w \in W} \varphi_{wt} \varphi_{ws} \tag{3.5}$$

Производная данного регуляризатора имеет вид:

$$\frac{\partial R}{\partial \varphi_{wt}} = -\tau \cdot \varphi_{wt} \cdot \sum_{s \in T \setminus t} \varphi_{ws} \tag{3.6}$$

Регуляризатор контрастирует строки матрицы Ф, то есть, в каждой строке вероятности более значимых термов увеличиваются, а остальные уменьшаются и могут обращаться в ноль.

Таким образом, обновленный EM – алгоритм принимает следующий вид:

$$p_{tdw} = norm(\varphi_{wt}\theta_{td}), \tag{3.7}$$

$$\varphi_{wt} = norm\left[n_{wt} + \varphi_{wt}\left(\frac{\tau_\varphi}{\varphi_{wt}} - \tau_\Phi \cdot \sum_{s \neq t} \varphi_{ws}\right)\right], \tag{3.8}$$

$$\theta_{td} = norm\left(n_{td} + \frac{\tau_\theta}{\theta_{td}}\right). \tag{3.9}$$

3.3 Отбор ключевых предложений.

Для того, чтобы LLM смогла узнать больше информации о теме, было решено добавлять в запрос помимо ключевых слов предложений еще и ключевые предложения.

Поиск этих самых предложений заключался в следующем: зафиксируем некоторую тему *t* и ее семантическое ядро D_t, далее, для каждого документа, у которого вероятность принадлежности к теме *t* максимальна среди всех остальных вероятностей, и для каждого его предложения s_d будем вычислять меру Жаккара:

$$Jac = \frac{|s_d \cap D_t|}{|s_d \cup D_t|} \tag{3.10}$$

Но данный подход имеет существенный недостаток – если само предложение маленькое, и в нем есть пару ключевых слов из семантического ядра, то оценка может быть неоправданно большой. То есть, мы будем получать короткие, неинформативные предложения, которые не смогут обогатить наш запрос полезной информацией о теме.

Таким образом, формула для определения оценки важности предложения была переделана:

$$Jac = \alpha \cdot \frac{|s_d \cap D_t|}{|s_d \cup D_t|} + (1 - \alpha) \cdot \frac{|s_d \cap D_t|}{|s_d|} \quad (3.11)$$

По сравнению с обычной мерой Жаккара, данная формула учитывает длину предложения и поощряет длинные предложения с большим количеством ключевых тематических слов, что позволяет находить более информативные предложения.

3.4 Стратегии формирования запросов

Ряд исследований показывает, что разные стратегии формирования запросов к LLM способны варьировать качество итогового ответа [18–20]. Таким образом, в данной работе необходимо было определиться какие запросы будут подаваться на вход большой языковой модели, чтобы выяснить оптимальную стратегию «общения» с ними.

В данной работе использовались три стратегии: zero – shot prompting, few – shot prompting, iterative – improving prompting. Рассмотрим каждую из них.

Zero – shot prompting:

Данный вид запросов является базовым, в том смысле, что в запросе просто указывается задание для большой языковой модели, но без каких-либо инструкций и подсказок к тому, как это делать.

В данной работе использовался следующий запрос:

“You will receive top tokens and sentences on a certain topic. Your task is to formulate a title and a short summary of the topic (5-7 sentences). Write in a formal style. Your task is to guess what the topic could be about, where small fragments and sentences were taken from (they are only a tiny part of the information about the topic). Therefore, you should write generically and offer simple options (don't invent

*too much). Guess what topic all this information might relate to.\nTop tokens:\n{***top_tokens***}\nTop sentences:\n{***top_sentences***}. Write like this: ***Name:***\nSome_name\n***Summarization:***\nSome_summarization"*

Few – shot prompting:

Данный вид запросов очень схож с zero – shot запросами, но добавляются еще и пояснительные примеры: в данной работе, модели показывались как положительные, так и отрицательные примеры, чтобы LLM смогла лучше понять конечную цель.

Пример запроса:

*"You will receive top tokens and sentences on a certain topic. Your task is to formulate a title and a short summary of the topic (5-7 sentences). Write in a formal style. Your goal is to guess what the topic could be about based on small fragments and sentences (they're only a tiny part of the information about the topic). Therefore, you should write generically and offer simple, broad descriptions without adding too much invented detail.\nWrite exactly in the following format:\n**Example 1:**\nTop tokens:\nTask decomposition, Data partitioning, Load balancing, Synchronization, Scalability, Race condition, Message passing, Shared memory, Lock-free, Granularity\nTop sentences:\n\"The proposed divide-and-conquer approach demonstrates superior task decomposition for heterogeneous clusters.\"\n\"Dynamic load balancing via work stealing significantly reduces idle time in multithreaded applications.\"\n\"A hybrid MPI–OpenMP framework yields scalable performance across hundreds of nodes.\"\n\"Lock-free data structures eliminate bottlenecks associated with traditional mutex-based synchronization.\"\n\"Cache-aware data partitioning improves locality and reduces false sharing in multicore environments.\"\n**Name:**\nParallelization Algorithms\n**Summarization:**\nParallelization algorithms explore methods for splitting computational work into simultaneously executable units to accelerate processing. They address challenges of task decomposition, data distribution, synchronization, and communication to optimize resource utilization across multicore and distributed systems. By balancing granu-*

*larity and overhead, and often combining shared-memory and message-passing paradigms, these algorithms scale applications from desktop machines to large clusters.\n---\n**Example 2:**\nTop tokens:\nEigenvalues, Singular value decomposition, Sparse matrix, LU factorization, Vector space, Orthogonality, Condition number, Krylov subspace, QR decomposition, Norm\nTop sentences:\n\"We derive error bounds for the QR decomposition of ill-conditioned matrices.\"\n\"Sparse matrix–vector multiplication on GPUs offers unprecedented acceleration.\"\n\"The Lanczos algorithm outperforms classical methods for computing large symmetric eigenvalues.\"\n\"An optimized LU factorization with partial pivoting reduces numerical instability.\"\n\"Singular value decomposition enables robust dimensionality reduction in data analysis.\"\n**Name:**\nLinear Algebra\n**Summarization:**\nLinear algebra studies vector spaces and matrix operations fundamental to numerical computation and data analysis. Core techniques include matrix factorizations, eigenvalue problems, and sparse-system solvers, which underpin applications ranging from scientific simulation to machine learning. Stability, efficiency, and dimensionality considerations guide the development of direct and iterative methods across CPU and GPU architectures.\n---\n**Example 3:**\nTop tokens:\nMinimally invasive, Robotic assistance, Hemostasis, Anastomosis, Laparoscopy, Suturing technique, Perioperative management, Wound healing, Surgical site infection, Enhanced recovery\nTop sentences:\n\"Robotic-assisted laparoscopy reduces intraoperative blood loss and shortens hospital stay.\"\n\"The novel hemostatic agent demonstrated rapid clot formation in vivo.\"\n\"Enhanced recovery protocols show significant improvement in postoperative pain control.\"\n\"Perioperative antibiotic stewardship reduces the incidence of surgical site infections.\"\n\"Assessment of wound healing using biocompatible mesh overlays in hernia repair.\"\n**Name:**\nSurgery\n**Summarization:**\nSurgery covers the planning, execution, and postoperative management of invasive procedures to diagnose or treat medical conditions. Techniques range from traditional open operations to minimally invasive and robotic-assisted approaches, each aiming to improve precision, reduce risk, and enhance recovery. Key aspects include hemostasis, anastomosis, wound healing, and*

*perioperative protocols to minimize complications and optimize patient outcomes.\n---\nNow your turn:\nTop tokens:\n{***top_tokens***}\nTop sentences:\n{***top_sentences***}\n\nWrite your answer in the following format:\n**Name:**\n(Some_name)\n**Summarization:**\n(Some_summarization)"*

Iterative – improving prompting:

Суть данной стратегии в отправлении нескольких запросов: сначала посылается запрос с просьбой составить «черновой» вариант название и суммаризации темы, а после отправляется запрос на улучшение.

Пример запроса:

*"Generate draft version of the topic name and its short summary (from 5 to 7 sentences) in academic style, based on the top tokens and sentences of the topic:\ntop tokens:\n{***top_tokens***}\ntop sentences:\n{***top_sentences***}\nAnswer will be in next format: **title**:\nsome_title\nsummary:\nsome_summary"*

*"It's previous answer: {***answer***}\nThe answer was based on these top tokens: {***top_tokens***} and top sentences: {***top_sentences***}. Study the answer carefully and improve it. Do not repeat the top sentences but write a brief summarisation of the (5-7) sentences. Keep the format of your answer: **Name:**\nsome_name\n**Summary:**\nsome_summary".*

3.5 Парное сравнение предложенных вариантов

Пусть у нас есть два множества: M – множество пар (семантическое ядро, ключевые предложения) для каждой темы, относительно выбранных построенных тематических моделей и множество стратегий запросов Pr (функции от семантического ядра и ключевых предложений):

$$m_i^{(t)} = \left(D_t^{(i)}, S_t^{(i)}\right),\ m_i = \left\{m_i^{(t)} \mid t \in T\right\} \tag{3.12}$$

$$M = \{m_i\}_{i=1}^{k} \tag{3.13}$$

$$pr_i^{(t)} = pr_i\left(m^{(t)}\right) \quad (3.14)$$

$$Pr = \{pr_i()\}_{i=1}^{n} \quad (3.15)$$

Определим множество получившихся запросов:

$$Pr_M = \{pr(m) \mid m \in M, pr \in Pr\} = \{p_1^{(t)}, \dots, p_{kn}^{(t)} \mid t \in T\} \quad (3.16)$$

Таким образом, множество (3.16) содержит запросы, которые в свою очередь, зависят от тематической модели (в данном случае, считаем, что тематическая модель – это пара (семантическое ядро, ключевые предложения), которая зависит от темы *t*).

После того, как были получены запросы по всем стратегиям и для всех тем, начинается этап сбора ответов LLM:

$$L(Pr_M) = \{LLM(p) \mid p \in Pr_M\} = \{l_1^{(t)}, \dots, l_{kn}^{(t)} \mid t \in T\} \quad (3.17)$$

$$l_i^{(t)} = (name, summary)_t$$

Когда все ответы большой языковой модели были собраны, необходимо определить, какой из вариантов комбинирования тематической модели и стратегии запроса был лучше. Для этого было выбрано провести парное сравнение по следующему алгоритму:

1. Относительно каждой темы мы формируем пары для сравнений:

$$\left\{\left(l_i^{(t)}, l_j^{(t)}\right) \,\middle|\, l_i^{(t)}, l_j^{(t)} \in L(Pr_M), i \neq j\right\} \quad (3.18)$$

2. Формируем запрос *prompt(l_i, l_j)* для модели – оценщика J_k.
3. Получаем ответ от каждой модели – оценщика:

$$J_k^{(t)}\left(prompt\left(l_i^{(t)}, l_j^{(t)}\right)\right) \in \left\{l_i^{(t)}, l_j^{(t)}\right\},\ J_k \in J = \{J_1, \dots, J_h\} \quad (3.19)$$

Таким образом, для каждой темы мы попарно сравним все полученные варианты названия и суммаризации. Для независимости решений использовалось несколько моделей.

Следовательно, по итогу, мы получим следующую матрицу:

$$\mathrm{R} = \left(\mathrm{r_{ij}}\right)_{i,j=1}^{|kn|} \quad (3.20)$$

$$r_{ij} = \sum_{t\in T}\sum_{k=1}^{h}\left[J_k^{(t)}(prompt\left(l_i^{(t)}, l_j^{(t)}\right)) = l_i^{(t)}\right] \quad (3.21)$$

То есть, если i–ый вариант оказался лучше j–его варианта по версии J_k оценщика, то в позицию (i,j) мы добавляем 1. Здесь под вариантом

Таким образом, для каждой комбинации тематической модели и стратегии формирования запросов мы получили варианты названий и суммаризаций тем, и попарно сравнили их относительно каждой темы, получив в конце суммарную итоговую матрицу, где описано, какой вариант комбинирования сколько раз победил другой вариант по мнению всех моделей – оценщиков.

3.6 Выбор наилучшей стратегии

Получив матрицу итогов необходимо определить лучший вариант. Стратегия брать вариант, победивший больше всех раз не самое объективное решение, так как могут присутствовать сложные взаимосвязи между «участниками». Таким образом, нужен другой подход.

В данной работе было решено использовать статистическую модель Брэдли – Терри (BT – model) для определения вектора сил: в данном случае, сила – это некоторая оценка для каждого варианта, и победителем будет тот, чья сила больше остальных.

Прямая задача модели BT по имеющимся оценкам участников определить вероятности выигрыша i–го игрока над j–ым:

$$\mathrm{P(i > j)} = \frac{e^{\beta_i}}{e^{\beta_i + e^{\beta_j}}} = \sigma(\beta_i - \beta_j) \quad (3.22)$$

Обратная задача заключается в определении вектора оценок для каждого участника по имеющейся информации о результатах турнира – то есть, важна информация, кто кого и сколько раз победил. Тогда вектор оценок находится при помощи максимизации логарифма правдоподобия:

$$\ln \prod_{ij}[P(i > j)]^{w_{ij}} = \sum_{ij}[w_{ij} \cdot \ln(\beta_i) - w_{ij} \ln(\beta_i + \beta_j)] \rightarrow max \quad (3.23)$$

В работе [22] было найдено неявное решение оптимизационной задачи, которое можно рассчитать численными методами:

$$\beta_i = \frac{\sum_j w_{ij}\beta_j/(\beta_i + \beta_j)}{\sum_j w_{ij}/(\beta_i + \beta_j)} \quad (3.24)$$

Таким образом, модель BT позволяет определить самого сильного игрока, что в нашем случае эквивалентно нахождению самой лучшей комбинации тематических моделей и стратегий запросов.

РАЗДЕЛ IV
ОПИСАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ

4.1 Использованная коллекция текстов.

В данной работе, для построения тематической модели использовалась коллекция научных текстов “ccdv/arxiv-summarization”, предназначенная для построения моделей абстрактивной суммаризации научных статей.

Каждая запись в данном датасете представляет из себя тройку вида (идентификатор, полный текст статьи, аннотация статьи). Всего в коллекции содержится 13073 записей. Все текста написаны на английском языке.

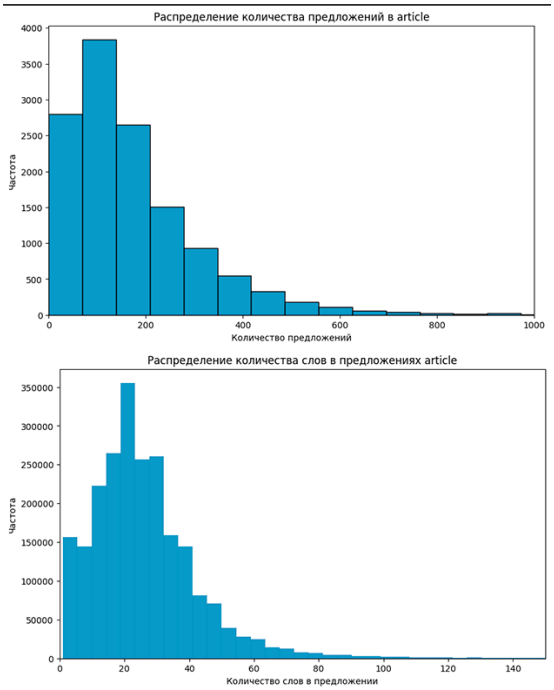


Рисунок 4.1 Распределение слов и предложений

Источник: скриншот авторской программы

На представленных графиках показаны распределения количества предложений в текстах и их длин. Таким образом, большинство текстов содержит от 50 до 250 предложений, и в большинстве предложений от 10 до 30 слов.

4.2 Предобработка текстовых данных

Один из важных этапов при работе с текстовыми данными – это этап предобработки текста. Необходимость данного процесса обусловлена тем, что:

– Тематические модели ориентируются на частоты встречаемости термов в документах, таким образом, если не убрать пунктуацию и общую лексику, слова, то модели будут работать с зашумленными данными и не смогут качественно находить и разделять скрытые тематики.

– Многие модели обработки естественного языка не способны напрямую работать с исходными текстами, и необходимо перевести текст в определенный формат. Тематические модели работают с частотными словарями, поэтому необходимо перед созданием и обучением модели подготовить частотный словарь исходной коллекции.

Таким образом, предобработка текстовой коллекции является ключевым этапом тематического моделирования, так как качество модели напрямую зависит от качества входных данных.

Рассмотрим проделанный процесс предобработки текстов:

– Все символы были переведены в нижний регистр, чтобы убрать возможные различия между одинаковыми словами: слова, начинающиеся с заглавной и строчной буквы не должны считаться разными («Наука» и «наука»).

– Была удалена HTML и LaTex разметка, а также пунктуация. Эти символы не несут никакой тематически значимой информации и лишь будут зашумлять входные данные. Документы, в которых встречалась разметка LaTex были удалены из коллекции.

– Удалены стоп-слова – малоинформативных распространенных слов, которые встречаются в большинстве документов и не помогают модели разделять темы между собой. Был использован подготовленный список стоп-слов из программного модуля nltk.

– Была произведена лемматизация – процесс приведения слова к нормальной словарной форме (лемме), учитывая грамматические правила языка.

После предобработки текстовой коллекции осталось 13038 научных статей.

4.3 Выбор оптимального количества скрытых тематик

Количество тем – ключевой параметр тематических моделей, который необходимо определить заранее, до построения моделей. Он напрямую влияет на качество и интерпретируемость модели: если тем слишком мало, скорее всего, получившиеся темы будут слишком общими и будут объединять слишком разрозненные понятия, и наоборот, если тем слишком много, многие будут дублирующими или разреженными.

Для определения тем изначально был использован алгоритм K-Means в цикле по количеству кластеров. Стратегия была следующая: на очередном шаге обучаем K-Means на заданном количестве тем, а после считаем метрики качества – WCSS и Silhouette. После проведения всех итераций, находим «локоть» на графике WCSS и смотрим на значение Silhouette. Локоть – это точка, после которой скорость изменения метрики резко уменьшается.

Были получены следующие результаты:

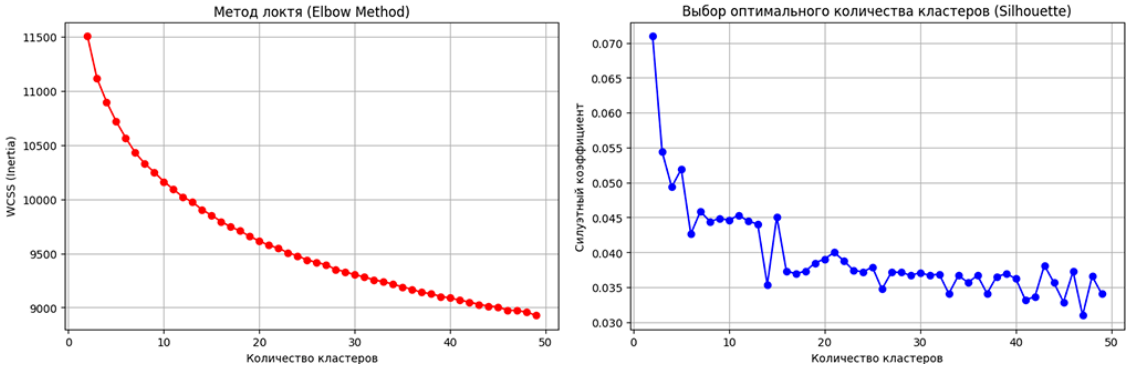


Рисунок 4.2 Результаты кластеризации

Источник: скриншот авторской программы

Заметим, что в графике WCSS нет очевидно выраженного локтя и график Silhouette идет на спад с увеличением количества кластеров, а также принимает крайне низкие значения (менее 0.1), что свидетельствует, что K-Means не смог выполнить качественную кластеризацию.

Такое могло произойти по той причине, что текста скорее относятся к нескольким тематикам, и потому, векторные вложения статей в своем гиперпространстве имеют слабовыраженную разделимость и классические методы кластеризации здесь не могут справиться.

Было принято решение искать оптимальное количество тем другим способом:

1. Разобьем исходную выборку на обучающую (10043 объекта) и тестовую (2995 объектов).
2. Будем строить и обучать модель PLSA в цикле по количеству тем от 10 до 30.
3. На каждой итерации будем вычислять значения перплексии на обучающей и тестовой выборке, средние значения чистоты и контрастности.
4. Построим графики, получившихся значений и проведем анализ.

Разбиение исходной выборки на две необходимо для мониторинга и предотвращения переобучения моделей.

Таким образом, были получены следующие результаты:

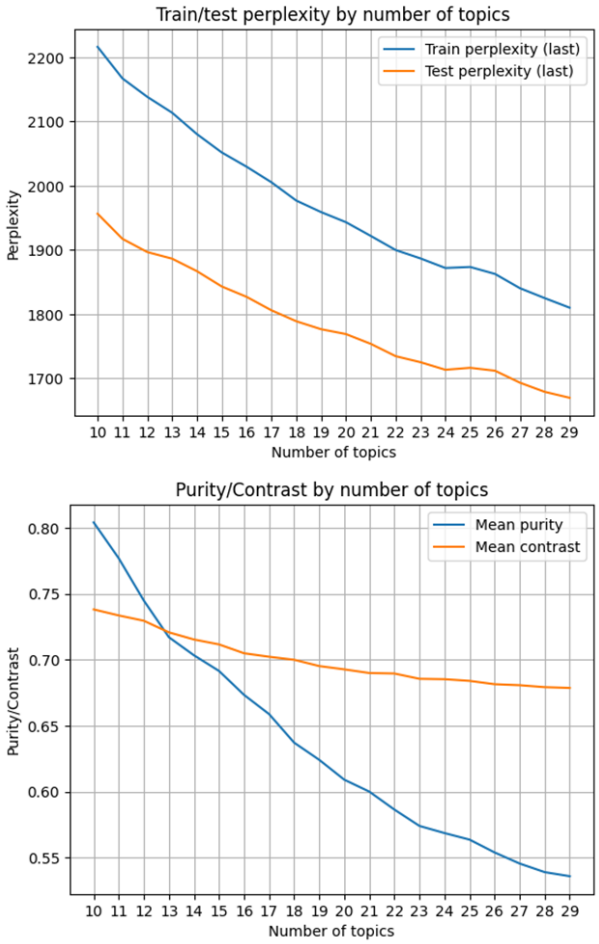


Рисунок 4.3 Результаты построений PLSA

Источник: скриншот авторской программы

Из первого графика видно, что при увеличении количества тем перплексия уменьшалась. Для анализа необходимо обратиться к графику чистоты и контрастности, для уточнения результатов.

Из второго графика видно, что метрики, которые должны быть как можно больше, убывают с увеличением количества тем. При этом, чистота убывает стремительнее чем контрастность, и на последней итерации принимает значения примерно 0.53, что свидетельствует об ухудшении качества модели при увеличении числа тем.

Хороший компромисс достигается при k=15: значения чистоты и контрастности находятся в районе 0.7, перплексия заметно ниже, чем при 10 темах.

Таким образом, в дальнейших исследованиях, все тематические модели строились на 15 темах.

4.4 Выбор тематических моделей.

Когда количество тем определено, следующий важный шаг по достижению поставленных целей – это построение тематических моделей, благодаря которым в последствии были получены семантические ядра тем.

Были построены два вида тематических моделей: на исходной коллекции текстов и на коллекции, где некоторые термы были заменены коллокациями. Коллокация – это словосочетания, имеющее признаки целостной семантической и синтаксической единицы.

Для построения тематических моделей использовался подход с добавлением аддитивной регуляризацией (ARTM), с использованием трех регуляризаторов: разреживание Φ, разреживание Θ, декоррелирование тем. Также, были исследованы различные стратегии применения регуляризаторов.

1 стратегия:

1 – 5 итераций: без регуляризаторов.

6 – 10 итераций: Разреживание Θ.

11 – 15 итераций: Разреживание Φ.

16 – 40 итераций: Декоррелирование тем.

2 стратегия:

1 – 5 итераций: без регуляризаторов.

6 – 20 итераций: разреживание Θ + разреживание Φ.

21 – 40 итераций: Декоррелирование тем (с изменением параметра τ).

3 стратегия:

1 – 10 итераций: декоррелирование тем.

11 – 30 итераций: разреживание Θ.

31 – 40 итераций: разреживание Φ.

Таблица 4.1

Параметры регуляризаторов

	A (1)	B (1)	C (1)	D (2)	E (2)	F (3)	G (3)
Разр. Ф	-0.5	-0.4	-0.4	-0.4	-0.4	-0.4	-0.4
Разр. Θ	-0.5	-0.7	-1.5	-0.5	-1.5	-0.5	-1.5
Декорр. Тем	10^5	10^5	10^5	50^5, 10^5	50^5, 10^5	10^5	10^5

На представленной таблице указаны значения параметра τ для разных регуляризаторов, при построении разных моделей (A – G), в скобках у названия модели указана стратегия регуляризации.

Рассмотрим полученные результаты качества моделей по основным метрикам:

Таблица 4.2

Метрики построенных моделей ARTM

	Perp. Train	Perp.Test	Purity	Contrast	Sparsity Φ	Sparsity Θ
A	2202	1960	0.70	0.78	0.89	0.45
B	2198	1959	0.70	0.77	0.89	0.50
C	2197	1951	0.69	0.76	0.89	0.59
D	2231	1974	0.71	0.79	0.90	0.44
E	2228	1965	0.70	0.78	0.89	0.58
F	2169	1948	0.70	0.78	0.89	0.45
G	**2167**	**1941**	**0.70**	**0.77**	**0.89**	**0.62**

Была выбрана модель G, показавшая хорошие показатели по выбранным внутренним метрикам оценивания тематических моделей.

При построении тематической модели на коллокациях использовались те же стратегии регуляризации и те же значения параметра τ.

Рассмотрим полученные результаты качества моделей по основным метрикам:

Таблица 4.3

Метрики построенных ARTM на коллокациях

	Perp. Train	Perp.Test	Purity	Contrast	Sparsity Φ	Sparsity Θ
A	3129	2769	0.69	0.76	0.89	0.49
B	3124	2767	0.69	0.75	0.89	0.54
C	3121	2753	0.68	0.75	0.89	0.63
D	3182	2794	0.70	0.77	0.90	0.48
E	**3178**	**2780**	**0.69**	**0.76**	**0.89**	**0.62**
F	3082	2747	0.68	0.73	0.89	0.51
G	3078	2735	0.67	0.73	0.88	0.64

Была выбрана модель E, показавшая хорошие показатели по внутренним метрикам. По перплексии модель занимает не первое место, но в контексте построения моделей для автоматической суммаризации тем, перплексия уступает другим метрикам по важности.

После выбора моделей были получены семантические ядра от моделей G, E для всех 15 тем. Так как каждому документу соответствует строка в матрице Θ, которая является распределением тем для данного документа, то для каждого документа была выбрана метка той темы, вероятность принадлежности к которой была максимальна. Таким образом, каждому документу мы сопоставили две метки L_G, L_E – метки тем, к которым, по мнению тематических моделей G, E принадлежал данный документ.

4.5 Выбор ключевых тематических предложений.

Для получения названий и суммаризаций тем от больших языковых моделей, необходимо было получить не только семантические ядра, но также ключевые предложения для каждой темы.

Алгоритм нахождения ключевых тематических предложений для некоторой темы t_i и меток одной из моделей G, E:

1. Отобрать документы, где хотя бы одна из меток L_i равняется t_i

2. Для каждого предложения очередного документа рассчитать модифицированную меру Жаккара относительно ключевых слов темы t_i.

3. Выбрать первые k предложений, чье значение модифицированной меры Жаккара выше, чем у других.

Для модифицированной меры Жаккара необходимо было выбрать значение параметра α. Для этого были произведены запуски алгоритма при разных значениях α = 0.3, 0.5, 0.7 и при k = 30.

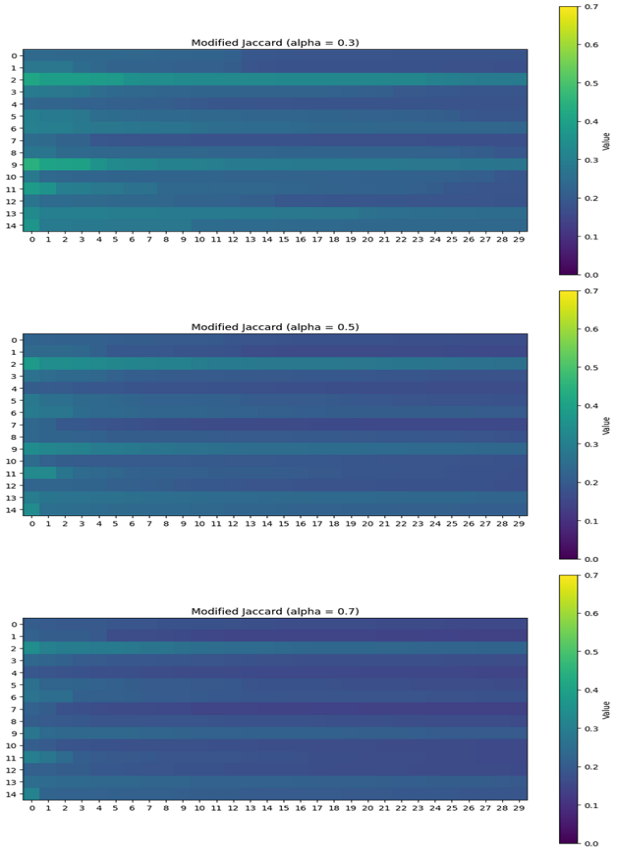


Рисунок 4.4 Тепловые карты распределения меры Жаккара

Источник: скриншот авторской программы

Таким образом, на тепловых картах представлено распределение значений меры Жаккара по темам и ключевым предложениям, при разных значениях альфа. Самые высокие значения достигались при α = 0.3, поэтому для дальнейших извлечений ключевых предложений было выбрано именно это значение альфа.

Для каждой темы и двух моделей G, E было отобрано 30 ключевых предложений. Таким образом, для каждой темы было составлено два набора: (ключевые слова, ключевые предложения).

4.6 Получение и оценка названий и описаний тем

Для каждой темы было сформировано 6 вариантов объединения тематических моделей и стратегий формирования запросов:

- Ez: модель E и стратегия zero–shot.
- Ef: модель E и стратегия few–shot.
- Ei: модель E и стратегия iterative–improving.
- Gz: модель G и стратегия zero–shot.
- Gf: модель G и стратегия few–shot.
- Gi: модель G и стратегия iterative–improving.

Отметим, что в данном случае, под моделью мы подразумеваем пару значений, которые от нее зависят – (семантическое ядро, ключевые предложения) для каждой темы (так как, в зависимости от модели мы получаем разные семантические ядра, и как следствие, разные ключевые предложения).

Для генерации названий и суммаризаций использовалась модель deepseek-r1-distill-llama-70b: дистилированный deepseek-r1-145b в llama2-70b, то есть, модель llama2-70b обучили на ответах более крупной модели deepseek-r1-145b.

Таким образом, для каждой темы было сформировано по 6 вариантов названий и кратких описаний.

Далее, необходимо было провести оценку полученных вариантов без привлечения экспертов. Был выбран вариант проведения парного сравнения: относительно каждой темы, мы создаем пары ответов и спрашиваем у моделей–оценщиков, какой вариант оказался лучше.

Для большей независимости, для оценивания было использовано 5 различных моделей:

– nvidia/llama-3.1-nemotron-ultra-253b-v1.
– google/gemma-3-27b.
– qwen/qwen-2.5-72b.
– deepseek/deepseek-chat-v3-0324.
– mistralai/mistral-small-3.1-24b-instruct.

Таким образом, относительно каждой темы было сформировано 15 пар для оценивания. Всего было получено 1125 пар.

Была сформирована матрица результатов:

Таблица 4.4

Матрица результатов парного сравнения

	Ez	**Ef**	**Ei**	**Gz**	**Gf**	**Gi**
Ez	-	42	32	49	43	47
Ef	33	-	54	61	56	63
Ei	38	21	-	44	45	45
Gz	26	14	31	-	48	33
Gf	32	19	30	27	-	50
Gi	28	12	30	42	25	-

Для определения самого наилучшего варианта была построена статистическая модель Брэдли–Терри. Значения матрицы результатов использовались как коэффициенты wij метода простых итераций по формуле (3.24), где новое значение вектора сил получался через правую часть формулы, где использовалось значение вектора сил с предыдущей итерации.

В качестве начального приближения был выбран вектор (0.5, 0.5, 0.5, 0.5, 0.5, 0.5) и алгоритм работал до момента достижения точности ∞-нормы разности двух векторов сил, полученных, на текущей и предыдущей итерациях, значения 10^{-8}. Алгоритм сошелся за 12 итераций.

Был получен следующий вектор сил:

Таблица 4.5

Вектор сил

Ez	**Ef**	**Ei**	**Gz**	**Gf**	**Gi**
1.344	2.154	1.048	0.715	0.756	0.619

Таким образом, можно сделать следующие выводы:

– Самым лучшим вариантом оказался Ef.

– Значения сил группы, где была задействована модель E, выше, чем значения сил группы модели G.

Следовательно, можно предположить, что семантические ядра модели E оказались лучше, чем у модели G, для данной коллекции текстов, что позволило в дальнейшем большой языковой модели лучше понять смысл тем и сформировать релевантные названия и суммаризации. Одним из ключевых различий моделей заключалось в том, что E была обучена на коллекции научных текстов, где некоторые парны термов были заменены одним (коллокации). Вероятно, это позволило модели лучше выделить темы и их ключевые слова.

Варианты названий:

Таблица 4.6

Название тем (Ef)

Topic 0	Algorithmic Problem Solving and Computational Methods
Topic 1	Health Research and Outcomes
Topic 2	Quantum Mechanics
Topic 3	Theoretical Physics

Продолжение таб. 4.6

Topic 4	Materials Analysis
Topic 5	Galaxy Clusters
Topic 6	Astrometry
Topic 7	Particle Interactions
Topic 8	Research Methodologies
Topic 9	Treatment Evaluation and Comparison
Topic 10	Computational Physics
Topic 11	Protein Signaling and Expression
Topic 12	Phase Transitions
Topic 13	Surgery
Topic 14	Electron Interactions and Material Properties

Таблица 6. Названия тем (Ef)

Варианты кратких описаний тем:

Topic 0:

Algorithmic problem solving and computational methods focus on the development and application of algorithms to tackle complex mathematical and real-world problems. These methods often involve graph theory, matrix operations, and optimization techniques to analyze and model systems. Key concepts include problem decomposition, network analysis, and the use of theorems to establish solutions. Applications range from scientific computing to machine learning, emphasizing efficiency, scalability, and precision in addressing challenges across diverse domains.

Topic 1:

Health Research and Outcomes examines the study and management of various health conditions across diverse populations. This field encompasses both public health aspects and individual patient care, with a focus on research methodologies, data collection, and the analysis of health metrics. It includes investigations into reproductive tract infections, mental health, physical therapy, and surgical procedures,

while addressing specific demographics such as children, adolescents, and particular patient groups. The topic emphasizes the use of scoring systems, research findings, and study outcomes to enhance our understanding and improvement of health conditions.

Topic 2:

Quantum mechanics explores the behavior of matter and energy at the smallest scales, focusing on the principles governing quantum systems. It delves into the mathematical framework describing states, operators, and interactions, including concepts like wave functions, Hamiltonians, and density matrices. The field examines phenomena such as superposition, entanglement, and decoherence, which distinguish quantum systems from classical physics. Techniques like the interaction picture and master equations are used to study the dynamics of these systems, including their transition to classical behavior in specific limits. Applications range from fundamental theories to emerging technologies in quantum computing and information science.

Topic 3:

Theoretical physics explores the mathematical and conceptual frameworks that describe physical phenomena, focusing on the development and solution of equations governing fundamental laws. Central to this field are the study of spacetime, cosmological models, and the behavior of energy and matter. Key aspects include analyzing solutions to equations of motion, understanding curvature in spacetime, and examining the implications of these solutions for our comprehension of the universe. Theoretical physics often combines principles from relativity, quantum mechanics, and thermodynamics to address questions about the nature of reality, from the smallest particles to the entire cosmos.

Topic 4:

Materials analysis involves the evaluation and characterization of materials to determine their properties and performance under various conditions. This field encompasses a wide range of techniques and methods to assess the structural, chemical, and physical characteristics of materials, often involving comparisons between

different samples and compounds. Whether through laboratory experiments, computational modeling, or advanced imaging, the goal is to understand material behavior, optimize their use, and develop new applications across industries.

Topic 5:

Galaxy clusters are studied through observations and emissions to understand their structure and galaxy distribution. Methods like spectrum analysis and luminosity measurements are used, focusing on density, radius, and other factors. This helps in understanding how galaxies are arranged within clusters and their properties.

Topic 6:

Astrometry involves the precise measurement of celestial objects' positions, distances, and movements. It employs various techniques to handle errors and enhance accuracy, including Bayesian methods and least squares estimates. This field addresses challenges in determining the positions of sources like stars and moons, often using sophisticated statistical approaches to manage measurement uncertainties. Key aspects include signal processing, imaging optics, and the application of diffraction formulas to improve data analysis. Astrometry is crucial for understanding the dynamics and structures of the universe through accurate and reliable positional data.

Topic 7:

Particle interactions involve the study of processes where subatomic particles, such as neutrinos, quarks, protons, and photons, engage in fundamental forces and exchanges. These processes include production mechanisms, decay pathways, and energy transfer dynamics, which are crucial for understanding the behavior of matter and energy at microscopic scales. Key aspects explored include cross-sections, which measure the likelihood of interactions, and the role of mediators like photons in facilitating these exchanges. Calculations and modeling play a significant role in predicting and analyzing these phenomena, enabling insights into the underlying physics of particle physics and its applications in astrophysics and beyond. By examining these interactions, researchers can elucidate the mechanisms governing particle production, decay, and their contributions to broader physical systems.

Topic 8:

Research methodologies encompass the systematic approaches and techniques used across various scientific disciplines to design, conduct, and analyze experiments. These methods focus on sample selection, control groups, and performance evaluation to ensure reliable and generalizable results. Applications range from medical research, where glycemic control and genetic analysis are critical, to engineering, where signal processing and network performance are optimized. The integration of quantitative analysis, statistical tools, and advanced computational methods underpins the development of robust research frameworks, enabling accurate conclusions and informed decision-making across diverse fields.

Topic 9:

Treatment evaluation and comparison involve the assessment of various therapies across different diseases to determine their effectiveness and safety. This process includes analyzing patient responses, comparing treatment outcomes, and identifying optimal strategies for managing conditions such as glaucoma, COPD, and other chronic diseases. Through randomized controlled trials, meta-analyses, and clinical observations, researchers aim to provide evidence-based recommendations to improve patient care. The focus is on understanding how different therapies impact disease progression, symptom management, and quality of life, ultimately guiding healthcare providers in selecting the most appropriate treatments for their patients.

Topic 10:

Computational physics focuses on the numerical simulation and analysis of physical systems to understand their behavior under various forces, densities, and dynamic conditions. This field employs mathematical models and computational techniques to study particle motion, fluid dynamics, and electromagnetic interactions. By solving equations of motion and using methods like Monte Carlo simulations, researchers explore phenomena such as phase transitions, critical points, and particle trajectories, providing insights into broader applications in materials science, astrophysics, and beyond. The integration of these computational tools allows

for the analysis of complex systems that are challenging to study through experimental means alone.

Topic 11:

Protein signaling and expression involve the study of how proteins are produced, modified, and regulated within cells to mediate various biological processes. These mechanisms are crucial for cellular communication, metabolism, and responses to external and internal stimuli. Research in this field explores how proteins interact with receptors, signaling pathways, and other molecules to trigger specific cellular activities. Through experiments in models such as mouse studies and human tissues, scientists investigate the effects of protein activation and expression on health and disease, including inflammation, cancer, and immune responses. Understanding these processes is essential for developing therapeutic interventions to modulate protein function and improve treatment outcomes.

Topic 12:

Phase transitions represent the study of changes in the state of matter and energy under varying conditions such as temperature, density, and pressure. These transitions are explored through theoretical and computational methods to understand the behavior of systems across different regions of phase space. Key aspects include the analysis of isospin distributions, energy density, and the role of simulations in predicting system behavior. The field examines how systems evolve between states, such as liquid, gas, and coexistence phases, and how these transitions are influenced by factors like binding energy and particle interactions. Computational tools, including exact diagonalization and density matrix methods, are employed to model and predict these phenomena. The study of phase transitions is critical for understanding physical systems at various scales, from nuclear matter to astrophysical objects.

Topic 13:

Surgery encompasses a broad range of medical procedures aimed at diagnosing, treating, or managing various conditions, including tumors, injuries, and dis-

eases. Surgical interventions often involve complex decision-making regarding techniques, patient preparation, and postoperative care to optimize outcomes. Key considerations include minimizing complications, managing tissue health, and ensuring effective treatment plans tailored to individual patient needs. The evaluation of surgical success is crucial, focusing on recovery, recurrence prevention, and enhancing the quality of life for patients across diverse clinical scenarios.

Topic 14:

This topic explores the behavior of electrons in various material systems, focusing on their interactions, energy states, and the influence of external factors such as magnetic fields and charge distribution. It delves into the analysis of electronic structures, phase transitions, and the effects of interactions like Van der Waals and Coulombic forces. The subject encompasses both theoretical and computational approaches to understand phenomena such as conduction, magnetic properties, and quantum effects, ultimately aiming to advance the understanding of material properties at a microscopic level.

ЗАКЛЮЧЕНИЕ

В данной работе была рассмотрена задача тематического моделирования коллекции научных статей с последующим использованием тематических представлений для генерации названий и описаний тем с помощью больших языковых моделей.

В ходе работы была использована коллекция “ccdv/arxiv-summarization”, прошедшая этап тщательной предобработки, включающей очистку от разметки, удаление пунктуации, лемматизацию и фильтрацию по стоп-словам. Проведён анализ структуры документов, выявивший особенности распределения предложений и длины текстов.

Особое внимание было уделено выбору числа скрытых тем. Первоначально протестированный метод K-Means не показал удовлетворительных результатов из-за слабой разделимости в эмбеддинговом пространстве. Более адекватный результат был достигнут путём перебора числа тем в модели PLSA и анализа метрик перплексии, чистоты и контрастности, в результате чего оптимальным числом тем было выбрано 15.

Для дальнейшего анализа были построены тематические модели ARTM с различными стратегиями регуляризации. Наилучшими по совокупности метрик качества (перплексия, чистота, контрастность, разреженность) стали модели G (на исходной коллекции) и E (на коллекции с коллокациями). Эти модели были использованы для выделения семантических ядер тем и соответствующих ключевых тематических предложений, подобранных с помощью модифицированной меры Жаккара.

На основе семантических ядер и ключевых предложений были сгенерированы названия и описания тем с применением различных стратегий взаимодействия с LLM (zero – shot, few – shot, iterative – improving). Для оценки качества полученных результатов была проведена масштабная процедура парного сравнения с использованием пяти различных моделей-оценщиков.

Результаты сравнений были агрегированы с помощью статистической модели Брэдли–Терри, что позволило определить наиболее эффективную стратегию генерации.

Наиболее эффективным вариантом оказалась тематическая модель ARTM, построенная на коллекции с коллокациями, и взаимодействие с большой языковой моделью при помощи стратегии few – shot, когда в запрос добавляются положительные и отрицательные примеры.

Значимость данной работы заключается в том, что был разработан новый метод абстрактивной суммаризации тем, с использованием тематического моделирования и больших языковых моделей, повышающий интерпретируемость тем. Также, был предложен метод автоматического оценивания получившихся названий и кратких описаний, при помощи парного сравнения и построения статистической модели Брэдли – Терри, позволяющий масштабировать исследования по разработке систем автоматической суммаризации и именования тем, без использования экспертных оценок.

СПИСОК ИСТОЧНИКОВ

1. Воронцов К., Потапенко А. Аддитивная регуляризация тематических моделей // Машинное обучение. — 2015. — Т. 101. — С. 303-323.

2. Vaswani A. Attention is all you need //Advances in Neural Information Processing Systems. – 2017.

3. Han K. et al. Transformer in transformer //Advances in neural information processing systems. – 2021. – Т. 34. – С. 15908-15919.

4. Han K. et al. A survey on vision transformer //IEEE transactions on pattern analysis and machine intelligence. – 2022. – Т. 45. – №. 1. – С. 87-110.

5. Mei Q., Shen X., Zhai C. Automatic labeling of multinomial topic models // Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. – 2007. – С. 490–499.

6. Gourru A., Velcin J., Roche M., Gravier C., Poncelet P. United we stand: Using multiple strategies for topic labeling // Natural Language Processing and Information Systems: 23rd International Conference, NLDB 2018, Paris, France. – 2018. – С. 352–363.

7. Truică C.-O., Apostol E.-S. TLATR: Automatic topic labeling using automatic (domain-specific) term recognition // IEEE Access. – 2021. – Т. 9. – С. 76624–76641.

8. Kinariwala S. A., Deshmukh S. Onto_TML: Auto-labeling of topic models // Journal of Integrated Science and Technology. – 2021. – Т. 9. – № 2. – С. 85–91.

9. Luhn H. P. The automatic creation of literature abstracts // IBM Journal of Research and Development. – 1958. – Т. 2. – № 2. – С. 159–165.

10. Torres-Moreno J.-M. Automatic text summarization.–Wiley, 2014.–320 с.

11. Zhang Y., Jin H., Meng D., Wang J., Tan J. A comprehensive survey on process-oriented automatic text summarization with exploration of LLM-based methods. – 2025.

12. Subramanian S., Li R., Pilault J., Pal C. On extractive and abstractive neural document summarization with transformer language models. – 2019. – URL: https://arxiv.org/abs/1909.03186

13. Williams L., Anthi E., Arman L., Burnap P. Topic modelling: Going beyond token outputs // Big Data and Cognitive Computing. – 2024. – Т. 8. – № 5. – С. 44.

14. Wang D., Zhu S., Li T., Gong Y. Multi-document summarization using sentence-based topic models // Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. – 2009. – С. 297–300.

15. Litvak M., Vanetik N., Liu C., Xiao L., Savas O. Improving summarization quality with topic modeling // Proceedings of the 2015 Workshop on Topic Models: Post-Processing and Applications. – 2015. – С. 39–47.

16. Lin C.-Y. ROUGE: A package for automatic evaluation of summaries // Text Summarization Branches Out: Proceedings of the ACL Workshop. – 2004. – С. 74–81.

17. Lin C.-Y., Cao G., Gao J., Nie J.-Y. An information-theoretic approach to automatic evaluation of summaries // Proceedings of the NAACL 2006. – 2006. – С. 463–470.

18. Sahoo P., Singh A. K., Saha S., Jain V., Mondal S., Chadha A. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications // arXiv preprint arXiv:2402.07927. – 2024. – С. 1–15. arXiv

19. Vatsal S., Dubey H. A Survey of Prompt Engineering Methods in Large Language Models for Different NLP Tasks // arXiv preprint arXiv:2407.12994. – 2024. – С. 1–23. PromptLayer+3arXiv+3Papers with Code+3

20. White J., Fu Q., Hays S., Sandborn M., Olea C., Gilbert H., Elnashar A., Spencer-Smith J., Schmidt D. C. A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT // arXiv preprint arXiv:2302.11382. – 2023. – С. 1–18.

21. Klein C., Maier C., Sunyaev A. Prompt Engineering in Higher Education: A Systematic Review to Help Inform Curricula // International Journal of Educational Technology in Higher Education. – 2025. – Vol. 22, Article 9. – C. 1–38. – DOI: 10.1186/s41239-025-00503-7.

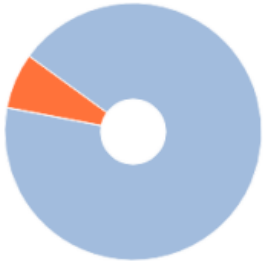
22. Newman M. E. J. Efficient computation of rankings from pairwise comparisons // Journal of Machine Learning Research. – 2023. – T. 24. – C. 1–25.

ПРИЛОЖЕНИЕ А Отчет системы «Антиплагиат»

АНТИПЛАГИАТ
ОБНАРУЖЕНИЕ ЗАИМСТВОВАНИЙ

Отчет о проверке

РЕЗУЛЬТАТЫ ПРОВЕРКИ



Совпадения:
7,44%

Оригинальность:
92,56%

Цитирования:
0%

Самоцитирования:
0%

«Совпадения», «Цитирования», «Самоцитирования», «Оригинальность» являются отдельными показателями, отображаются в процентах и в сумме дают 100%, что соответствует проверенному тексту документа.

Проверено: 94,86% текста документа, исключено из проверки: 5,14% текста документа. Разделы, отключенные пользователем: Библиография