Казахско-Британский Технический Университет

УДК 004.91

На правах рукописи

АХМЕТОВ ИСКАНДЕР

Разработка метода для информативного экстрактивного реферирования научных текстов на английском языке

8D06101 — Информатика, Вычислительная Техника и Управление

Диссертация на соискание ученой степени
доктора философии (PhD)

Научный консультант
Пак А.А.
кандидат технических
наук, ассоц. профессор KBTU
(Kazakhstan)

Зарубежный научный
консультант
Гельбух А.Ф.
профессор IPN (Mexico)

Республика Казахстан
Алматы, 2022

СОДЕРЖАНИЕ

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей диссертации использованы ссылки на следующие стандарты:

1 Закон Республики Казахстан «О науке» от 18.02.2011 г. N2 407-IV ЗРК.

2 Приказ и.о. Министра здравоохранения и социального развития Республики Казахстан от 31 июля 2015 г. № 647 утверждении государственных общеобязательных стандартов и типовых профессиональных учебных программ по медицинским и фармацевтическим специальностям» с внесенными изменениями в Приказе Министра здравоохранения Республики Казахстан от 21 февраля 2020 года N9 ҚР ДСМ-12/2020. Зарегистрирован в Министерстве юстиции Республики Казахстан 27 февраля 2020 года № 20071.

3 Правила присуждения ученых степеней, утвержденных приказом Министра образования и науки Республики Казахстан от 31 марта 2011 года 127 (зарегистрирован в Реестре государственной регистрации нормативных правовых актов под № 6951).

4 ГОСТ 7.32-2017 Межгосударственный Стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления.

ОПРЕДЕЛЕНИЯ

В настоящей диссертации применяют следующие термины с соответствующими определениями:

абстрактное резюмирование	Методы абстрактного резюмирования генерируют резюме путем построения новых коротких предложений, подобно человеку. Резюме может содержать фразы, которых нет в оригинальном тексте. Для создания языка абстрактных резюме необходимы методы генерации и сжатия.
матрица смежности	Матрица смежности - это квадратная матрица, представляющая конечный граф. Элементы матрицы показывают, являются ли пары вершин смежными или нет в графе.
автоматический анализ текста	Автоматический анализ текста - это метод машинного обучения, позволяющий автоматически извлекать ценные сведения из текстовых данных. Предприятия используют инструменты автоматического анализа текста для быстрого анализа данных и документов в Интернете и превращения их в полезную информацию.
автоматическое реферирование текста	Автоматическое реферирование текста - это процесс сжатия текстовых данных вычислительным путем для создания резюме, представляющего наиболее ценную информацию из оригинального текста.
big pharma	Компании Big Pharma - это шесть крупнейших фармацевтических компаний, расположенных в США, включая Pfizer, Johnson & Johnson, Amgen, Merck, Gilead и AbbVie.
bilingual evaluation understudy	BLEU - это метрика, разработанная для автоматизированной оценки машинного перевода, и ее поведение хорошо коррелирует с человеческой оценкой. Основная идея BLEU заключается в измерении степени близости между созданным переводом и набором золотых стандартов. Близость рассчитывается на основе средневзвешенного значения совпадений n-грамм переменной длины между сгенерированным и целевым человеческим переводом.
bleu score	см. “bilingual evaluation understudy”.
бизнес-аналитика	Бизнес-аналитика (BI), термин был впервые введен H. P. Luhn в 1958 году, означает стратегии и технологии для

	анализа данных и управления бизнес-информацией в компаниях. Функции технологий бизнес-аналитики включают аналитическую обработку данных в режиме онлайн, отчетность, разработку приборных панелей, аналитику, интеллектуальный анализ данных, интеллектуальный анализ процессов, управление эффективностью бизнеса, обработку сложных событий, бенчмаркинг, интеллектуальный анализ текста, предписывающую аналитику и предиктивную аналитику.
cheetsheet	Cheetsheet - это сверхсжатый конспект учебного материала, подготовленный для быстрого ознакомления и повторения.
методы сжатия	Методы сжатия - это процессы кодирования информации с использованием меньшего количества битов, чем исходное представление. Любая конкретная техника сжатия является либо сжатием с потерями (удаление ненужной или менее важной информации), либо сжатием без потерь, определяющим и устраняющим статистическую избыточность.)
вычислительные ресурсы	Вычислительные ресурсы - это ресурсы, используемые вычислительными моделями при решении вычислительных задач.
concept mining	Concept mining - это процесс извлечения понятий из текстов, включающий аспекты AI и статистики, такие как data mining и text mining. Задача осложняется неструктурированностью текстов, но ее результаты дают полезное представление о семантике и сходстве документов.
косинусное сходство	Косинусное сходство - это мера сходства между двумя векторами одинаковой длины, и определяется как косинус угла между ними в векторном пространстве.
критические части текста	Критические части текста - это наиболее ценные фрагменты текста, которые в совокупности меньше оригинального текста по размеру.
data mining	Data Mining - это междисциплинарная область компьютерных наук, изучающая процессы извлечения и обнаружения закономерностей в больших массивах данных с

	использованием методов на стыке статистики, машинного обучения и систем баз данных. Конечной целью Data Mining является преобразование информации в понятную структуру для дальнейшего использования при принятии решений.
выборка данных	Выборка данных - это набор точек данных, собранных или отобранных из исходного набора данных по определенной процедуре. Элементы выборки называются наблюдениями.
набор данных	Набор данных - это совокупность данных, поступающих в структурированной или неструктурированной форме и используемых для обучения ML-модели после предварительной обработки.
коллекция документов	Коллекция документов - это набор текстовых документов в электронной форме.
реферирование документов	см. в разделе “автоматическое реферирование текста”.
eigenvector centrality	Eigenvector Centrality - это мера влияния узлов в графе. Относительные баллы присваиваются всем узлам графа на основе концепции, что связи с узлами с высоким собственным вектором более ценны для оценки узла, чем связи с узлами с низким собственным вектором. Для узла высокий показатель собственного вектора означает, что он связан со многими узлами с высоким показателем.
эволюционные алгоритмы	Эволюционные алгоритмы - это метаэвристические методы, вдохновленные естественным процессом отбора, включая мутацию, кроссинговер и другие.
executive summary	Executive Summary - это краткий документ или раздел документа, резюмирующий более длинный деловой документ или серию документов, чтобы читатели могли быстро ознакомиться с большим объемом текстового материала без необходимости читать его целиком.
исчерпывающий поиск	Исчерпывающий поиск, или поиск "грубой силой это очень распространенная техника решения задач, состоящая в полной проверке того, удовлетворяет ли каждое решение-кандидат постановке задачи.

экстрактивное реферирование	Методы экстрактивного реферирования выбирают информативные предложения из исходного документа на основе заданных критериев для построения резюме.
общее резюме	Общее резюме не ориентировано на особые потребности какой-либо конкретной группы пользователей и в среднем подходит широкой публике. Универсальность общих резюме можно рассматривать и как преимущество, поскольку она удовлетворяет потребность в информации большинства пользователей в целом, и как недостаток, связанный с отсутствием персонализации в таких резюме. Девиз обобщенных резюме - “Один размер подходит всем!”.
среднее геометрическое	Среднее геометрическое - это мера, указывающая на центральную тенденцию набора чисел и определяется как n^{th} корень из произведения n чисел.
глобальный минимум	см. “глобальный оптимум”.
global optima	Глобальные оптимумы или экстремумы включают максимумы и минимумы функции, которые являются наибольшим и наименьшим значением функции для всей области.
представление графов	Графовое представление данных - это метод, реализующий концепции неориентированных и направленных графов из математической области теории графов для использования ее инструментов в работе с данными.
жадный алгоритм	Жадный алгоритм - это любой алгоритм, который следует эвристике решения проблемы, принимая из возможных вариантов лучшее локальное решение для задачи оптимизации. Для некоторых задач жадная эвристика может дать локально оптимальное решение, приближенное к глобально оптимальному решению за разумное время.
жадный метод	см. “жадный алгоритм”.
среднее гармоническое	Среднее гармоническое - одна из мер центральной тенденции, включая медиану, моду, арифметические и геометрические средние. Он определяется как обратное среднее арифметическое взаимно обратных чисел данного набора чисел.

эвристический поиск	Эвристический поиск - это техника быстрого нахождения приближенного решения задачи, когда классические методы не могут найти точного решения или им требуется очень много времени для сходимости. Быстрые результаты достигаются ценой точности.
ориентировочное резюме	Основная цель ориентировочного резюме - порекомендовать содержание статьи, не давая подробной информации о ней. Оно может служить в качестве тизера, чтобы побудить пользователя получить полный текст. Аннотации к книгам, фрагменты результатов веб-поиска и трейлеры к фильмам являются примерами ориентировочных резюме.
информационный поиск	Информационный поиск - это наука о поиске запрашиваемой информации путем поиска соответствующих документов (или изображений и звуков) и информации в них.
информативное реферирование	Информативное реферирование максимизирует полноту представления информации из исходного документа или документов. Поэтому оно содержит всю важную информацию, необходимую для передачи основного смысла исходного текста, и опускает вспомогательную информацию.
обратная частота документа	Обратная частота документа (IDF) - это обратная частота документа слова (в скольких документах появилось слово), используемая для снижения важности слова, подразумеваемой его частотой термина (TF).
обратная функция	Обратная функция функции f - это функция, которая инвертирует f.
извлечение ключевых слов	Извлечение ключевых слов - это автоматическое определение и извлечение терминов, которые наилучшим образом описывают тему текстового документа.
база знаний	База знаний (KB) - это система компьютерных технологий, используемая для хранения сложной структурированной и неструктурированной информации, относящейся к определенной области знаний.
языковая инженерия	Языковая инженерия - это отдельная область, противопоставленная вычислительной лингвистике и обработке естественного языка (NLP), и она включает в себя создание систем NLP, стоимость и результаты которых поддаются количественной оценке и предсказуемы.

порождение языка	см. “порождение естественного языка”.
языковая модель	Языковая модель - это вероятностное распределение по последовательностям слов из заданного языка и текстовых корпораций. Она оценивает вероятность появления слова, учитывая предшествующую последовательность слов.
latent dirichlet allocation	Latent Dirichlet Allocation - это генеративная статистическая тематическая модель, которая на основе количества вводимых тем и статистики совпадений слов относит документы в данной корпорации к ряду тем, полученных несамостоятельным способом [1].
lay summary	Lay Summary - это текстовое резюме научных или технических документов, предназначенных для использования нетехнической аудиторией.
лемматизация	- это процесс приведения слова к исходной грамматической форме: множественного числа к единственному, глаголов к неопределенной форме и так далее.
лексическая семантика	Лексическая семантика - это изучение значений слов, включая структуру значения, грамматику и композиционное поведение, а также отношения словоупотребления и смысла слов.
лексическая замена	Лексическая замена - это задача определения наилучшей альтернативы для слова, учитывая контекст фразы.
обзор литературы	Обзор литературы - это обзор ранее опубликованных работ по определенной теме, представленный в виде полного научного документа или раздела такого документа.
локальные минимумы	см. “локальные оптимумы”.
локальный оптимум	Локальный оптимум задачи оптимизации - это решение, которое является максимальным или минимальным в соседнем множестве решений-кандидатов. Это отличие от глобальной оптимы, которая является оптимальным решением из всех возможных решений.

longest common subsequence	Longest Common Subsequence (LCS) - это проблема нахождения самой длинной подпоследовательности, общей для всех членов набора последовательностей.
машинное обучение	Машинное обучение (ML) как часть искусственного интеллекта (AI) - это изучение компьютерных алгоритмов, которые могут улучшаться автоматически путем обучения на наборах данных.
машинный перевод	Машинный перевод (MT) изучает использование программного обеспечения как средства перевода текста или речи с одного языка на другой.
ручной труд	Ручной труд - это физический или умственный труд, выполняемый человеком, в отличие от труда машин или рабочих животных.
максимальное значение	см. “глобальный оптимум” и “локальный оптимум”.
средние значения	Средние значения - это меры центральной тенденции; см. “гармоническое среднее” и “геометрическое среднее”.
мета-данные	Мета-данные - это данные, предоставляющие информацию о других данных, но не являющиеся фактическим содержанием этих данных. Они могут включать информацию об источнике данных, сроках, авторах и т.д.
Mind-map	Mind-map - это очень популярная концепция диаграммы предметных связей, придуманная и активно рекламируемая Тони Бьюзаном в 1974 году. Диаграмма организована в виде верхней темы, помещенной в центр, к которой иерархически подключены связанные с ней идеи, категории, термины и представления [2].
протокол собрания	Протокол собрания (MOM) - это, как правило, официальный протокол, содержащий краткое изложение обсуждения тем повестки дня собрания. Документ также содержит резолюции и распределение обязанностей по задачам, подписанные всеми участниками собрания.
многодокументное резюме	Методы многодокументного резюме используют коллекцию документов, связанных с определенной темой или событием, и создают резюме по нескольким документам во временном порядке.

многоязычное обобщение	Иногда входные данные могут быть представлены в виде документа на смешанном языке. Например, стенограмма диалога между людьми в многоязычной и многокультурной среде, или документ, состоящий из множества иноязычных терминов. Также это может быть серия документов на разных языках, связанных общей темой, в случае многодокументного обобщения.
мультимодальное обобщение	Означает представление краткого изложения на носителях, безразличных к исходному документу. Например, обобщение аудиозаписи в текстовой форме или обобщение текста в виде изображения.
взаимная информация	Взаимная информация (ВИ) двух случайных величин определяет количество информации, полученной об одной случайной величине путем наблюдения за другой случайной величиной. Этот термин тесно связан с понятием энтропии или мерой информации, содержащейся в случайной переменной.
Двусмысленность естественного языка	Двусмысленность естественного языка - это тот факт, что один и тот же термин может иметь несколько значений.
генерация естественного языка	Генерация естественного языка (NLG) - это запрограммированный процесс производства вывода естественного языка в виде текста.
обработка естественного языка	Обработка естественного языка (NLP) - это междисциплинарное исследование в области информатики, лингвистики и искусственного интеллекта, связанное с взаимодействием компьютеров с использованием человеческого языка. Цель NLP - сделать компьютеры способными понимать тексты на человеческом языке без путаницы, вызванной двусмысленностью и сарказмом, и производить осмысленный и точный вывод в задачах классификации текстов, поиска информации и NLG.
модели нейронных сетей	Нейронные сети (НС) - это вычислительные системы, вдохновленные биологическими нейронными сетями, которые составляют нейронные системы животных.
серия чисел	Серия чисел - это упорядоченная коллекция чисел.

числовая мера	Числовая мера - это количественная характеристика объекта.
целевая функция	Целевая функция - это функция, результат которой должен быть максимизирован или минимизирован в задаче оптимизации.
матрица встречаемости	Матрица встречаемости описывает частоту встречаемости терминов в коллекции документов. В матрице встречаемости строки соответствуют документам в коллекции, столбцы - терминам, а значения - количеству раз, когда термин встречается в документе.
opinion mining	см. анализ настроений.
оптимальное решение	Оптимальное решение - это лучшее решение среди всех возможных решений, достигнутое с использованием оптимального времени и ресурсов.
алгоритмы оптимизации	Алгоритмы оптимизации - это последовательность операций, направленных на поиск глобального или локального оптимума для функции цели.
проблема оптимизации	Оптимизация Проблема поиска оптимального решения с учетом ограничений и критериев выбора.
перефразирование	Перефразирование - это ручной или автоматический процесс переписывания фрагмента текста другими словами или фразами, чтобы избежать плагиата, создавая уникальный текст.
случайная выборка	Случайная выборка - это случайный отбор точек данных из набора данных для формирования репрезентативного подмножества данных и оценки характеристик всего набора данных. Выборка позволяет сэкономить время и ресурсы, особенно если исходный набор данных очень большой или операции, на которые направлена выборка, сложны с вычислительной точки зрения.
количество ссылок	В этой книге количество ссылок означает количество проиндексированных ссылок на статью за определенный период времени.
относительная энтропия	Относительная энтропия или дивергенция Куллбэка-Лейблера (KL) измеряет, насколько два распределения вероятности отличаются друг от друга.

относительная частота	Относительная частота - это число определенных исходов, деленное на число всех исходов.
репрезентативная выборка	Репрезентативная выборка - это выборка данных, отобранная таким образом, что она сохраняет структуру и относительный состав исходного набора данных.
исследовательское сообщество	Исследовательское сообщество - это разнообразная сеть взаимодействующих ученых или ученых, работающих в одной области.
влияние исследований	Термин "влияние исследований разработанный Times Higher Education (THE), является показателем роли университета в распространении новых научных знаний и идей.
научные работы	Научные работы - это научные документы, описывающие уникальные исследования и результаты, достигнутые определенным ученым по определенной теме, и опубликованные для использования и пользы научного сообщества.
обзорная статья	Обзорная статья - это статья, обобщающая текущее состояние научных знаний по какой-либо теме, путем анализа и обсуждения методов и выводов в ранее опубликованных работах.
размер выборки	Размер выборки - это количество точек данных, включенных в выборку данных. Выбор размера выборки определяется требованием репрезентативности выборки и ресурсами, доступными для исследования.
sentiment analysis	Sentiment Analysis (SA) или Opinion Mining (OM), это задача обработки естественного языка (NLP) для обнаружения эмоционального отношения (например, положительного, отрицательного или нейтрального), выраженного автором текста относительно темы или предмета. SA или OM обычно решается как контролируемая классификация текста [3] или неконтролируемая задача кластеризации с использованием искусственного интеллекта (AI), машинного обучения (ML) и добычи данных.
научная литература	Научная литература - это научная работа, опубликованная в научных журналах, книгах или диссертациях. Большинство научных журналов и академических книг, как

	правило, основываются на форме экспертной оценки, чтобы квалифицировать тексты для публикации.
научные статьи	см. “исследовательские работы”.
научная литература	см. “научная литература”.
научные работы	см. в разделе “научные работы”.
научные публикации	см. в разделе “исследовательские работы”.
научно-исследовательское сообщество	см. “Научно-исследовательское сообщество”.
научная работа	см. “исследовательские работы”.
скоринговая модель	Скоринговая модель - это функция или правило, измеряющее точность вероятностных прогнозов.
эвристика поиска	см. “Эвристический поиск”.
результаты поиска	Результаты поиска - это страницы, отображаемые поисковыми системами в ответ на запрос пользователя и содержащие список найденных в процессе поиска элементов.
семантическая связанность	Семантическая связанность - это метрика того, насколько сильно слова, предложения и документы похожи или несхожи по смыслу.
семантическое сходство	см. “семантическое родство”.
семантическое пространство	Семантическое пространство - это представление естественного языка, которое способно передать смысл, решая проблемы неоднозначности естественного языка и несовпадения словарного запаса.
извлечение предложений	Извлечение предложений - это техника отбора предложений на основе некоторых критериев для автоматического резюмирования текста.
кратчайший путь	Проблема кратчайшего пути - это задача поиска пути между двумя вершинами графа или узлами, такого, чтобы сумма весов ребер пути была минимальной.

мера сходства	Мера сходства - это вещественная функция, определяющая сходство между двумя объектами путем инверсии меры расстояния (косинуса, евклидова, манхэттенского) между этими объектами. Мера принимает большие значения для схожих объектов и нулевое, малое или отрицательное значение для несхожих объектов.
Резюме одного документа	Методы резюмирования одного документа направлены на резюмирование одного единственного документа, без обращения к какому-либо внешнему источнику, кроме данного документа. Резюмируя один документ, мы сталкиваемся с меньшей избыточностью информации и неопределенностью оценки, чем в случае с методами резюмирования многодокументного текста.
одноязычное обобщение	Для этого типа ввода текст или тексты представлены на одном языке. Это позволяет использовать одну конкретную языковую модель, не сталкиваясь с проблемами отсутствия слов в словаре (OVW), межъязыковой омонимии и другими проблемами.
singular value decomposition	Singular Value Decomposition (SVD) - это факторизация вещественной или комплексной матрицы, используемая в задачах тематического моделирования и уменьшения размерности.
социальные медиа	Социальные медиа - это коммуникационные технологии и цифровые информационные каналы, которые облегчают генерацию и обмен идеями, информацией, интересами и другими формами выражения в виртуальном сообществе и сетевых СМИ.
исходный текст	Исходный текст - это оригинальный текст, который переводится на другой язык, цитируется или обобщается.
speech to text	Speech to Text (STT) - это технология производства текста с расшифровкой записи речи.
stemming	сокращение слов до их основной формы путем отсечения префиксов и аффиксов, которые являются общими для данного языка. Получившийся в результате стемминг может отличаться от грамматически правильного корня, но он может хорошо служить цели сокращения словарного запаса текста.

стандартное отклонение	Стандартное отклонение - это мера вариации или дисперсии набора значений. Низкое значение стандартного отклонения указывает на то, что значения близки к среднему или ожидаемому значению набора, в то время как высокое значение стандартного отклонения указывает на то, что значения разбросаны в более широком диапазоне.
статистический вывод	Статистический вывод - это процесс использования анализа данных для обобщения свойств базового распределения вероятности.
частота термина	Частота термина - это количество раз, когда термин встречается в документе. См. также “инверсная частота документа“.
Term Frequency-Inverse Document Frequency	TF-IDF - сокращение от term frequency-inverse document frequency, является числовой статистикой, направленной на отражение важности слова для документа в наборе данных.
тестовые данные	Тестовые данные - это данные, которые были специально отобраны для тестирования модели, обученной на отдельном наборе данных.
тестовый набор	см. “тестовые данные”.
анализ текста	смотрите “автоматический анализ текста”.
классификация текста	Классификация текста - это задача отнести документ к одному или нескольким классам или категориям, которая может быть выполнена вручную или алгоритмически.
текстовые корпорации	Текстовые корпорации - это языковые ресурсы, состоящие из больших и структурированных наборов текстовых данных, используемых для проведения статистического анализа и проверки гипотез, проверки встречаемости или утверждения лингвистических правил для конкретного языка.
текстовые данные	Текстовые данные - это обычно любой набор текстов, собранных для целей анализа текста.
текстовый файл	Текстовый файл - это тип компьютерного файла, структурированный как последовательность строк электронного текста.

генерация текста	см. “генерация языка”.
text mining	Text Mining, или добыча текстовых данных - это процесс извлечения ценной информации из текста. Он включает в себя алгоритмическое обнаружение новой, ранее неизвестной информации, путем автоматического извлечения информации из различных письменных ресурсов.
обработка текста	Обработка текста - это ручной или автоматизированный процесс создания или манипулирования электронным текстом.
поиск текста	Алгоритмы поиска текста пытаются найти место, где одна или несколько строк встречаются в более крупной строке или тексте.
размер текста	Размер текста - это мера объема текста, выраженная в байтах, символах, словах, предложениях или страницах.
резюмирование текста	см. “автоматическое резюмирование текста”.
text to speech	Text to Speech (TTS) - это технология создания речевой записи из текста, которая по сути является обратной Speech to Text (STT).
словарный запас текста	Список уникальных слов из текста. Может значительно отличаться по размеру в зависимости от метода, которым мы измеряем уникальность слова. Скажем, если мы используем формальный принцип идентичности последовательности символов, то словарь может содержать множество грамматических форм одного и того же слова. Но если мы используем методы стемминга или лемматизации, то словарный запас сократится и будет содержать только стеммы или неопределенные формы слов.
текстовые данные	см. “текстовые данные”.
временная сложность	Временная сложность - это мера вычислительной сложности, описывающая количество машинного времени, необходимого для выполнения алгоритма, которое обычно оценивается по количеству элементарных операций, выполняемых алгоритмом.
данные для обучения	Данные для обучения - это выборка данных, используемая для обучения ML-модели.

transformer model	Transformer Model - это модель нейронной сети, которая использует механизм самовнимания, дифференцированно взвешивая значимость для каждого фрагмента входных данных. Трансформеры превосходят популярные ранее рекуррентные нейронные сети (RNN) благодаря тому, что им не нужно обрабатывать данные по порядку, что позволяет работать с большим объемом текстов.
уникальное слово	Слово, которое не повторяется в списке или последовательности слов. Таким образом, список уникальных слов из текста образует словарь текста.
upper bound	Верхняя граница подмножества S некоторого предварительно упорядоченного множества K, это элемент K, который больше или равен каждому элементу S. Аналогично, нижняя граница S определяется как элемент K, который меньше или равен каждому элементу S.
резюме, ориентированное на пользователя	Резюме, ориентированные на пользователя, настраиваются для удовлетворения потребностей определенной группы или отдельного пользователя, настраивая резюме в соответствии с профессиональным и образовательным уровнем конечного пользователя или его индивидуальными предпочтениями.
поиск по переменным окрестностям	Поиск по переменным окрестностям (VNS), впервые предложенный Младеновичем & Хансеном в 1997 году, является метаэвристическим методом для решения множества задач оптимизации.
vocabulary mismatch	Vocabulary Mismatch - это тот факт, что одно и то же значение может быть выражено разными способами в естественных языках.
средневзвешенная средняя	Средневзвешенная средняя - это мера центральной тенденции, учитывающая различный вклад (вес) элементов в их сумму.
частота слов	Частота слов - это частота, с которой слова встречаются в данной текстовой корпорации.
список слов	Список слов - это список лексем в данном корпусе текстов, служащий для составления словаря.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящей диссертации использованы следующие обозначения и сокращения:

ABS	Attention-Based Summarization
ACL	Association for Computational Linguistics
ACS	Суммирование комментариев к статьям
AI	Искусственный интеллект
ALRG	Automatic Literature Review Generation
ANT	Автоматическое ведение записей
ARDA	Advanced Research and Development Activity
ATG	Автоматическая генерация тестов
ATS	Automatic Text Summarization
B	billion (1,000,000,000)
BART	Двунаправленный и авторегрессионный трансформатор
BERT	Представления двунаправленного кодера из трансформаторов
BLANC	Bacronymic Language model Approach for summary quality estimatioN. Круто?
BLEU	BiLingual Evaluation Understudy
BP	Бревити пенальти
BPE	Кодирование байтовых пар
CCIS	Communications In Computer And Information Science
CKB	Корпоративная база знаний
CLTS	Cross-Language Text Summarization
COLING	International Conference on Computational Linguistics
DARPA	Агентство передовых оборонных исследовательских проектов
DAC	Сборник статей Дайджеста
DUC	Конференция по пониманию документов

EMNLP	Empirical Methods in Natural Language Processing
EMR	Электронные медицинские записи
FRUMP	Программа понимания и запоминания быстрого чтения
GAN	Generative Adversarial Network
GiB	Gibibyte
GPO	US Government Publishing Office
GPT	Generative Pre-trained Transformer
GPU	Graphic Processing Unit
GRAD	GRAph Distance
GSG	Gap Sentences Generation
h-index	Hirsch-Index
HatBART	Hierarchical attention BART
HMM	Скрытые модели Маркова
K	thousands (от kilo по-гречески)
KB	База знаний
KiB	Kibibyte
KL	Kullback-Lieber
KWIC	Ключевые слова в контексте индексации
LCS	Longest Common Subsequence
LDA	Latent Dirichlet Allocation
LIT	Язык и информационные технологии
LNAI	Конспект лекций по искусственному интеллекту
LNBI	Конспект лекций по биоинформатике
LNCS	Lecture Notes in Computer Science
LongSumm	Генерирование длинных резюме для научных документов, общая задача SDP
LSA	Latent Semantic Analysis

M	million
MDS	Multi-Document Summarizationl
MLM	Маскированная языковая модель
MMS	Multi-Modal Summarizationl
MOM	Протокол собрания
MSLR	Multi-Document Summarization for Literature Reviews, SDP Shared task
MT	Машинный перевод
NIST	Национальный институт стандартов и технологий
NLG	Natural Language Generation
NLM	National Library of Medicine
NLP	Natural Language Processing
NLTK	Natural Language Tool Kit
NSG	Генерация новостных сниппетов
OTS	Open Text Summarizer
OVW	Вышедшее из словаря слово
PSG	Генерация презентационных слайдов
QA	Ответы на вопросы
RANLP	International Conference on Recent Advances In Natural Language Processing
RAS	Russian Academy of Science
RoBERTa	Robustly Optimized BERT Pretraining Approach
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
RSS	Общая информация о сайте
SE	Поисковая система
SCUs	Summary Content Units
SDI	Выборочное распространение информации

SDP	Scholarly Document Processing, COLING workshop
Seq2Seq	Архитектура нейронной сети "от последовательности к последовательности"
SERA	Оценка суммирования с помощью анализа релевантности
SIGDAT	ACL Special Interest Group for linguistic Data
SimCLS	Simple Framework for Contrastive Learning of Abstractive Summarization
STT	Speech-To-Text
SunSITE	Sun Software, Information & Technology Exchange
SVD	Singular Value Decomposition
T5	Text-to-Text-Transfer-Transformer
TAC	Конференция по анализу текста
TC	Total Citations
TF-IDF	Частота термина-Инверсная частота документа
THE	Times Higher Education
TIDES	Translingual Information Detection Extraction and Summarization
TP	Общие публикации
TS	Суммирование текста
TSC	Цитаты количество публикаций в области суммирования текста
TSP	Количество статей по обобщению текста
TTS	Text to Speech
UniLM	Unified pre-trained Language Model
VNS	Variable Neighborhood Search
WWI	World War I
X-Sum	Extreme Summarization
АРТ	Автоматическое Реферирование Текста, см. также ATS

ААР	Автоматическое Абстрактивное Реферирование
АЭР	Автоматическое Экстрактивное Реферирование
ОЕЯ	Обработка Естественных Языков, см. NLP
ПК	Персональный Компьютер
ПИО	Поиск по Изменяемым Окрестностям, см. также VNS
ПО	Программное Обеспечение

1 ВВЕДЕНИЕ

Быстрая обработка информации является жизненно важной функцией, необходимой в настоящее время каждому современному человеку. Процесс **Автоматического Реферирования Текста (АРТ)** сталкивается со множеством проблем, несмотря на то, что технологии в этой области постоянно развиваются, и эта проблема изучается с 1958 года [4]. Например, как оценивать качество полученных рефератов, и что должно служить эталоном для сравнения? Есть две основные задачи, которые решаются в процессе АРТ:

1 Выбор критически важной информации из заданного текста.
2 Представление этой информации в сжатом виде.

АРТ это сложная задача в области обработки естественного языка, поскольку она включает в себя тщательный семантический и лексический анализ текста для создания обоснованного сжатого представления исходных текстовых данных. Высококачественный автореферат должен содержать основную информацию, быть точным в отношении фактов, релевантным, читаемым и не избыточным [5]. Исследования в этой области начались в 1958 году [4] и новые многочисленные работы и методы появляются каждый год, начиная с 2003 года [6], когда стали доступны большие массивы данных для этой цели и необходимое вычислительное оборудование оживившие интерес к данной теме исследований.

С самого начала исследования проблем реферирования текста было разработано множество различных методов; подробнее см. Глава 3.2. Методы различаются по количеству документов, к которым они применяются; таким образом, существует одно-документное и много-документное авто-реферирование. [7] определил два класса методов обобщения текста:

1 **Экстрактивный автореферат** - включает в себя шаги по извлечению конкретных предложений из исходного текста, без каких либо изменений.
2 **Абстрактивный автореферат** - предполагает связное и сжатое изложение исходного текста в свободной форме.

Если сравнивать эти два метода, то второй тип больше похож на человеческое мышление, поскольку встает необходимость заменять слова синонимами и переставлять их местами. В отличие от этого, экстрактивный метод заключается в составлении резюме из исходного текста, находя самые важные предложения. Таким образом, экстрактивные резюме легче получить и ожидается, что они дадут лучшие результаты, чем абстрактные резюме [8]. Из этого следует, что второй класс сложнее, так как в нем задействованы такие сложные техники, как генерация естественного языка [9].

В настоящее время исследования во всем мире переориентировались на абстрактивное авто-реферирование [10]. Тем не менее, экстрактивное автоматическое реферирование все еще в тренде, как видно из научных работ

последних двух лет [11–13]. Помимо сложности формирования резюме, открытым вопросом в научном сообществе является его оценка. Метрика качества текстов должна учитывать неоднозначность естественного языка.

Актуальность работы

Работы в направлении разработки методов для автоматического информативного реферирования научных текстов сейчас являются как никогда актуальными в следствии экспоненциального роста информации в целом и научной информации в частности в нашем сегодняшнем Мире.

Самые современные методы автоматического реферирования текстов на текущий момент построены на основе сложных архитектур нейронных сетей с миллиардами параметров, и натренированные на огромных количествах данных и используют эмбеддинги из предтренированных языковых моделей таких как BERT, GPT-3 и другие. Это поднимает вопросы о возможной избыточной сложности таких моделей, их способности к обобщению, ведь миллиарды параметров нейросети позволяют моделям просто "зазубривать"правильные ответы, и в конечном счете вопросы об экономической эффективности и экологической безопасности этих моделей.

В настоящее время, каждому человеку, а научному работнику в первую очередь, остро необходимы инструменты для эффективной работы с информацией, одним из которых может быть система для автоматического реферирования текстов. Данная тема подробно исследована в классических работах [14–16].

Цели, задачи и объект исследования

В этой работе мы начали с непростого вопроса: есть ли место для развития методов *Автоматического Экстрактивного Реферирования Текстов (АЭРТ)*? Или они устарели и должны быть заменены более современными методами *Автоматического Абстрактивного Реферирования Текста (ААРТ)*? Кроме того, естественно возникает вопрос: какого максимального качества резюме мы можем достичь с помощью экстрактивных методов реферирования?

Цель диссертационной работы – Разработать метод для информативного реферирования научных текстов на английском языке, которая поможет экономить время исследователям сокращая объем информации для переработки, и одновременно сохраняя информационную ценность текста.

Задачи диссертационной работы:

1 Экспериментальная оценка наивысшего значения метрики ROUGE, который можно получить с помощью методов экстрактивного автоматического реферирования текстов.
2 Разработка метода автоматического реферирования на основе подхода

жадного алгоритма.

3 Подбор управляющих параметров разработанного метода.

4 Сравнительный анализ работы полученного метода с существующими на данный момент.

Объект исследований. Алгоритмы Автоматического Экстрактивного Реферирования научных текстов.

Научная новизна

Новизна предложенного метода заключается в уникальном применении подхода жадного алгоритма в методе экстрактивного, информативного реферирования текстов.

Также, метод демонстрирует производительность на уровне современных реферативных методов, в разработке которых использовались нейронные сети и колоссальный объем данных для обучения. При этом, предлагаемый метод, относительно прост, и требует гораздо меньше времени и данных для обучения.

Вклад нашего исследования в научные знания заключается в следующем: 1) выявление верхней границы для методов экстрактивного суммирования (VNS, жадный алгоритм, генетический алгоритм) и обнаружение того, что VNS, инициализированный жадным алгоритмом, работает даже лучше, чем любой из алгоритмов самостоятельно для данной задачи, 2) предложение метода экстрактивного суммирования, основанного на алгоритме жадности, который работает на высоком уровне, несмотря на свою относительную простоту, 3) очищенный набор данных с различными типами резюме с высоким ROUGE и полезной статистикой текста.

Помимо этого, мы поднимаем несколько важных вопросов для дальнейших исследований и идей для других исследователей; см. Раздел 3.5.3:

- Нахождение оптимального количества предложений для максимизации метрики ROUGE.

- Определение оптимального параметра `min_df` для каждой реферируемой статьи индивидуально, вместо использования среднего значения.

- Проведение дальнейших экспериментов по определению верхней границы качества авторефератов достижимых Экстрактивными методами Автореферирования Текстов.

Методика исследований

Решение поставленных в работе задач осуществлялось на основе применения общенаучных методов исследования в рамках проведения экспериментов с текстовыми данными и количественной оценки полученных результатов. Программирование и исходные коды были выполнены на языке Python.

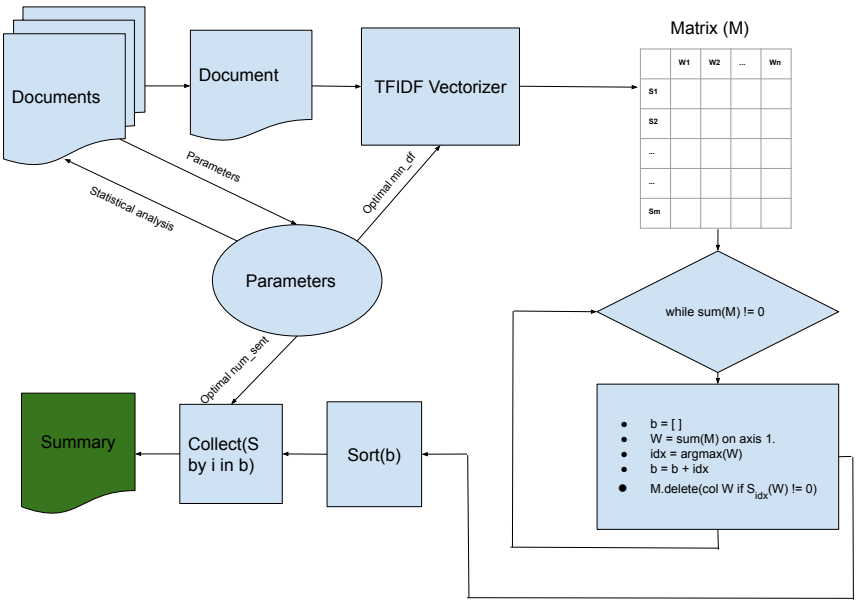


Рисунок 1.1 – GreedSum: метод автоматического реферирования текстов на основе жадного алгоритма.

Практическая значимость работы

Мы представляем подход экстрактивного суммирования[1], который использует простые и старые методы, но при этом работает на уровне современных методов, использующих сложные архитектуры нейронных сетей и огромные объемы данных для обучения; см. Рис. 1.1 для краткого описания нашего подхода. Набор данных arXiv extract [17] из 17 тысяч статей, которые мы использовали в наших экспериментах, доступен по адресу `https://data.mendeley.com/datasets/nvsxfcbzdk/1`. Некоторые из других преимуществ предложенного подхода включают:

- Вычислительная простота.

- Не требуется обучение модели машинного обучения, но используется статистический вывод.

- Рефераты, сгенерированные алгоритмом, богаты полезной информацией из текста.

Разработанный метод автоматического экстрактивного реферирования текста имеет самый широкий спектр практического применения в науке, образовании и бизнесе.

Наука:

- Автоматизация обзора литературы.
- Генерация аннотации статьи.
- Мультимодальная суммаризация.

[1]Исходный код доступен на GitHub по адресу `https://github.com/iskander-akhmetov/Greedy-Summarization`

- Популяризация науки.
- Обновление научной информации.
- Использование суммаризации в других задачах NLP.
- Тематическое моделирование.
- Анализ тональности.

Образование:

- Автоматическое создание заметок.
- Памятки.
- Интеллект-карты.
- Генерация слайдов презентации.
- Генерация тестовых вопросов.
- Написание эссе.

Бизнес:

- Резюме больших объемов текста (отчеты, исследования, бизнес-планы).
- Формирование протокола собрания.
- Реферирование, ориентированное на запрос.
- Оптимизация контекстной рекламы.

На защиту выносятся:

1 Метод эвристической оценки уровня качества реферата метрикой ROUGE-1, достижимого при помощи экстрактивных методов автоматического реферирования текстов дает результат в 0.59, что значимо выше текущего уровня в 0.46 у самых современных методов использующих нейронные сети.

2 Разработанный метод Автоматического Экстрактивного Реферирования (АЭР) GreedSum, который показывает результат 0.42 по метрике ROUGE-1 на датасете arXiv.

3 Вывод о значимости тонкой настройки гиперпараметра minimum document frequency (min_df) работы GreedSum , или минимальная частота вхождения слов в предложения реферируемого текста для их учета в создании словаря для построения TFIDF матрицы. На выборке в 376 текстов из датасета arXiv путем простого перебора оптимальное значение min_df было определено как 0.042 (т.е. слово должно появляться как минимум в 4.2% предложениях).

2 Обзор литературы

Цель данной главы - представить широкий обзор проблемы автоматического реферирования текста, включая существующие наборы данных, методы реферирования текста и методы оценки качества автоматического реферирования.

Другие обзоры в свободном доступе охватывают лишь отдельные аспекты проблемы автоматического реферирования текста. Например, они освещали подходы и методы [18, 19], методы [20], или методы оценки резюме [21]. Фрагментарность тем обзоров затрудняет работу исследователей, особенно тех, кто только начинает изучать эту область.

В следующих разделах этой главы мы представили библиометрический обзор области исследования, затем показали доступные данные, методы и метрики оценки, сравнили наиболее известные методы автоматического реферирования текста и перешли к результатам и выводам.

2.1 Библиометрия

Мы использовали базу данных Scopus и Google Scholar для библиометрического анализа публикаций по обобщению текста. Scopus компании Elsevier - это реферативная и индексирующая база данных с полнотекстовыми ссылками рецензируемой литературы. По состоянию на конец 2021 года коллекция содержит более 40 000 наименований из примерно 11 678 международных издательств, из которых почти 35 000 журналов являются рецензируемыми в ведущих предметных областях. Scopus охватывает широкий спектр форматов публикаций (книги, журналы, материалы конференций и другие) в области естественных, технических, медицинских, социальных наук, искусства и гуманитарных наук.

Google Scholar - это веб-поисковая система (SE), которая индексирует полный текст научной литературы или метаданные по широкому спектру областей исследований и форматов публикаций. Google Scholar содержит около 389 миллионов документов, включая книги, статьи и патенты, что делает его крупнейшим в мире академическим поисковиком [22]. Кроме того, Google Scholar включает контент с различных платформ, как бесплатных, так и требующих подписки, таких как Scopus.

Набор данных, полученный из базы данных Scopus, содержит метаданные о публикациях, такие как год публикации, название, авторы, журналы, страны публикаций, учреждения и предметная область. Таблица 2.1 показывает свойства набора данных.

Таблица 2.2 показывает самые продуктивные институты (период с 1958 по 2021 год), упорядоченные по количеству публикаций в области суммирования текста. Еще одна полезная информация - о мировом рейтинге университетов.

Таблица 2.1 – Свойства набора данных Scopus.

Статьи	57,255
Авторы	6,654
Учреждения	160
Источник	128
Конференции	29,107
Предметная область	28
Страны	160

Методология Times Higher Education (THE)[1] группирует 13 показателей рейтинга университетов в пять основных измерений: Исследования (30%), Цитирование (30%), Преподавание (30%), Международные перспективы (7,5%), Доход от промышленности (2,5%). THE описывает влияние исследований как показатель роли университета в распространении новых научных знаний и идей.

Таблица 2.3 показывает 10 наиболее продуктивных журналов по исследованиям в области Summarization. Заметное лидерство по количеству публикаций на 2020-2021 годы принадлежит журналу Lecture Notes in Computer Science (LNCS), включая его подсерии Lecture Notes in Artificial Intelligence (LNAI) и Lecture Notes in Bioinformatics (LNBI).

Информация о цитировании, библиографии и ключевых словах авторов 1 289 статей была экспортирована в VosViewer [23].

Анализ набора данных Scopus был проведен в контексте тематики публикаций и с использованием поисковых запросов для выявления наиболее часто встречающихся ключевых слов в статьях по обобщению текста. Иллюстрация в Рис. 2.1 показывает, что ключевое слово "Summarization"наиболее часто встречается в сочетании с ключевыми словами "Natural Language Processing Systems"и "Text Processing". Данные в Таблица 2.4 показывают количество статей, в которых ключевое слово "Summarization"используется в паре с другими ключевыми словами. Таблица 2.4 формируется из вершины наиболее часто встречающихся ключевых слов. Пересечение показывает количество статей, из которых видно, что ключевое слово 'Natural Language Processing systems' используется в сочетании с другими ключевыми словами наиболее часто.

[1] `https://www.timeshighereducation.com/world-university-rankings/world-university-rankings-2020-methodology`

[2] Всего публикаций (2017-2020)

[3] Всего цитат

[4] Cite Score 2020

[5] Publications on Text Summarization (2000-2021)

[6] In 2019.

Таблица 2.2 – Топ учреждений по количеству публикаций согласно базе данных Scopus.

Ранг	Учреждение	Всемирный рейтинг университетов 2021-2022	Всего публикаций	Страна
1	Китайская академия наук	73	110	Китай
2	Пекинский университет	59	79	Китай
3	Университет Карнеги-Меллон	85	62	Соединенные Штаты
4	Университет Китайской академии наук	73	61	Китай
5	Колумбийский университет	7	55	Соединенные Штаты
6	Университет д'Алакант	845	47	Испания
7	Гонконгский политехнический университет	267	44	Китай
8	Пекинский университет почт и телекоммуникаций	737	42	Китай
9	Университет Шеффилда	150	41	Великобритания
10	Университет Авиньона и горных районов Воклюза	1493	41	Франция

По данным Scopus можно выделить 10 самых плодовитых авторов. Данные в Таблица 2.6 показывают список авторов с наиболее значительным количеством статей по теме Резюме текста. Лидером является Lloret Elena, которую цитировали в 459 статьях, причем 160 цитирований приходится на обзорную статью [24].

Таблица 2.7 показывает наиболее ценных авторов для области Summarization. Основным признаком влияния является количество цитирований публикаций из области обобщения текста. По статистике, наиболее ценные публикации принадлежат Лю Бингу и Ху Минцину.

В анализ соавторства мы включили 113 стран, связанных с 2 967 авторами. Были отобраны авторы с наибольшей суммарной силой связей (53 из них соответствуют пороговым значениям), как показано на Рис. 2.2. Цвет авторов указывает на связи между авторами. Иллюстрация в Рис. 2.3 демонстрирует силу связей авторов по странам. Данные в Таблица 2.8 показывают матрицу соавторства, где - количество совместных публикаций на пересечении. В рам-

Таблица 2.3 – Топ-10 самых продуктивных журналов по количеству публикаций по обобщению текста.

	Журнал и публикация.[2]	Цитаты[3]	CS 2020[4]	TS Publ.[5]	
1	LNCS, LNAI, LNBI	82,766	141,179	1.8	139
2	Advances In Intelligent Systems And Computing	29,624	26,852	0.9[6]	42
3	CEUR Workshop Proceedings	18,904	15,553	0.8	42
4	Communications In Computer And Information Science	19,615	15,364	0.8	23
5	Экспертные системы с приложениями	2,710	34,460	12.7	21
6	ACM International Conference Proceeding Series	31,048	35,869	1.2	19
7	IEEE Access	41,670	201,619	4.8	19
8	Обработка информации и управление	541	4,676	8.6	16
9	Международная конференция по последним достижениям в обработке естественного языка (RANLP)	267	516	1.9	16
10	Procedia Computer Science	8,236	24,640	3.0	15

Таблица 2.4 – Топ 10 наиболее часто встречающихся ключевых слов при суммировании текста.

Ранг	Ключевое слово	Количество
1	Системы обработки естественного языка	1,887
2	обработка текста	1,871
3	Резюме текста	1,812
4	семантика	1,025
5	информационный поиск	708
6	вычислительная лингвистика	666
7	Data Mining	635
8	Автоматическое суммирование текста	499
9	обработка естественного языка	488
10	искусственный интеллект	474

ках общего распределения существует восемь основных сообществ, в которых авторы имеют совместные публикации.

Год публикации - еще один важный элемент информации для библиометрических исследований. Нашей первой задачей было изучить годы публика-

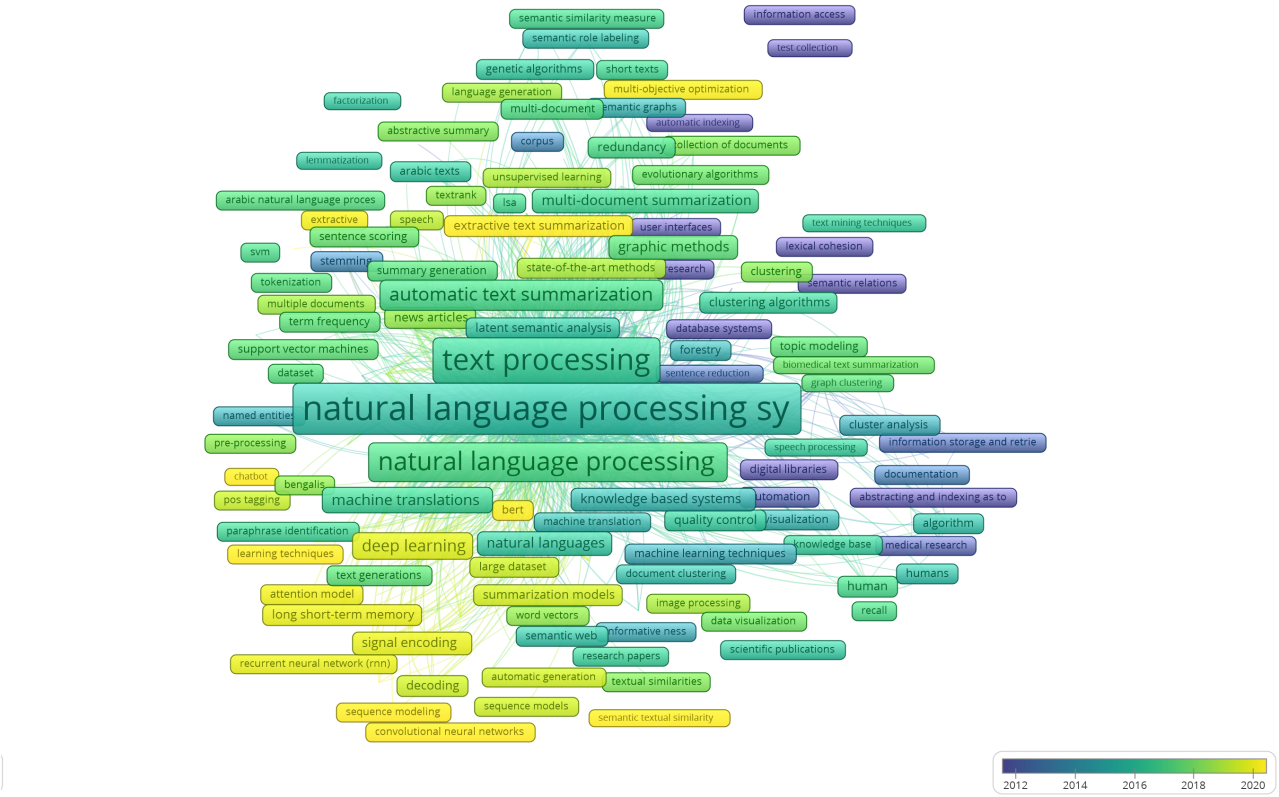


Рисунок 2.1 – Встречаемость ключевых слов в статьях по обобщению текста.

Таблица 2.5 – Совпадение ключевых слов.

	Системы обработки естественного языка	Суммирование текста	Текстовая обработка	Семантика	Natural Language Processing	Искусственный интеллект	Computational Linguistics	Information Retrieval	Data Mining
Системы обработки естественного языка	-	1,887	5,681	13,414	15,469	7,444	15,463	7,321	7,461
Резюме текста	1,887	-	1,812	1,025	750	474	666	708	635
Обработка текста	5,681	1,812	-	10,923	9,198	6,151	8,209	7,680	9,821
Семантика	14,600	1,025	10,923	-	3,753	11,852	11,884	10,867	8,878
Обработка естественного языка	15,469	488	9,198	3,753	-	9,006	15,867	7,901	8,175
Искусственный интеллект	7,444	474	6,251	11,852	9,006	-	4,212	7,275	16,245
Вычислительная лингвистика	15,463	666	8,209	11,884	15,867	4,212	-	3,567	2,487
Информационный поиск	7,321	708	7,680	10,867	7,901	7,275	3,567	-	12,125
Добыча данных	7,478	635	9,821	8,878	8,175	16,245	2,487	12,125	-

Таблица 2.6 – Десятка самых продуктивных авторов в области исследования суммирования текста. *Примечание: Количество статей по суммированию текста (TSP), общее количество публикаций (TP), h-индекс - индекс Хирша, общее количество цитирований (TC), код страны ISO 3166.*

	Автор	TSP	TP	h-index	TC	Current affiliation	Country
1	Lloret Elena	37	452	12	547	Universitat d'Alacant	ES
2	Салим Наоми	31	219	24	2,559	Universiti Teknologi Malaysia	MY
3	Saggion Horacio	30	153	23	2,055	Университет Помпеу Фабра Барселона,	ES
4	Lins Rafael Dueire	25	84	13	681	Федеральный сельский университет Пернамбуку,	BR
5	Palomar Manuel	21	867	17	1,013	Universitat d'Alacant	ES
6	Гупта Вишал	14	48	12	1,376	Университетский институт инженерии и технологии	IN
7	Абуджар Шейх	12	54	8	177	Независимый университет, Бангладеш	BD
8	Хоссейн Саид Ахтер	12	95	9	308	Университет либеральных искусств Бангладеш	BD
9	Каллимани Джагадиш С.	9	43	6	100	Visvesvaraya Technological University,	IN
10	Алами Набиль	7	8	4	69	Факультет наук Дар Эль Махраза, Университет Сиди Мохамед Бен Абделла,	MA

ции, указанные в поле резюме. Рис. 2.4 показывает гистограмму года публикации. Всего мы смогли идентифицировать 57 255 документов, начиная с 1958 года (год первой публикации Луна).

Мы показываем, что количество публикаций постоянно растет, и значительно увеличилось, начиная с 1995 по 2015 год, что объясняется тем, что публикации статей по методам машинного обучения и нейронных сетей впервые были применены к суммированию [33] и [34] соответственно. Наибольшее число публикаций по обобщению текстов - 6 608 - было опубликовано в 2019 году; см. Рис. 2.4.

Исследуя вопрос о том, сколько другие исследователи разных поколений ссылаются на методы, используемые для реферирования текста с 1958 года по настоящее время, мы с удивлением обнаруживаем, что метод, введенный Luhn [4] в начале этого периода, все еще ссылается. Более того, количество ссылок неуклонно растет с 1997 года по настоящее время с 42 до 217 в 2019 году, достигнув общего числа 3,681 ссылок за 62 года, что может означать, что метод взвешивания предложений/слов является фундаментальным и эффек-

Таблица 2.7 – Десять самых ценных авторов по теме "Суммирование текста"по количеству цитирований "Суммирование текста"

.

Author	The most cited article in Summarization research area	TSP	TSC	Current affiliation	Country
Liu Bing	Mining and summarizing customer reviews [25]	5	7,302	University of Illinois at Chicago	US
Hu Minqing	Mining and summarizing customer reviews [25]	3	5,565	MySpace Inc.	US
Radev Dragomir	LexRank: Graph-based lexical centrality as salience in text summarization [26]	18	4,899	Yale University	US
Erkan Gunes	LexRank: Graph-based lexical centrality as salience in text summarization [26]	3	1,918	University of Michigan	US
Zhai Chengxiang	Topic sentiment mixture: Modeling facets and opinions in weblogs [27]	11	1,789	University of Illinois Urbana-Champaign	US
Liu Pengfei	Searching for effective neural extractive summarization: What works and what's next [28]	3	1,264	Carnegie Mellon University	US
Lu Yue	Latent aspect rating analysis on review text data: A rating regression approach [29]	8	1,249	Nanjing University of Aeronautics and Astronautics	CHN
Li Wei	Pachinko allocation: DAG-structured mixture models of topic correlations [30]	30	1,209	Yahoo Research Labs	US
Liu Peter. J.	Get to the point: Summarization with pointer-generator networks [31]	4	1,203	Google LLC	US
McKeown Kathleen	Sentence fusion for multidocument news summarization [32]	9	1,084	Columbia University	US

тивным, см.Рис. 2.5. Однако после 2020 года тенденция снижается, достигая точки с показателем 61 и 81 в 2021 году.

Аналогичной популярностью пользуются и более поздние работы о методе машинного обучения, впервые примененном [33], [35] и [34] в 2015 году; см. Рис. 2.6. [34] разработал подход к реферированию на основе внимания (ABS) для создания резюме на уровне предложений; [33] классифицировал предложения как предложения резюме с помощью классификатора Наива-Бейеса.

Fast Reading Understanding and Memory Program (FRUMP) [36] и скрытые модели Маркова (HMM) [37] модели редко упоминаются в научных статьях;

Таблица 2.8 – Матрица соавторства

.

	Abujar S.	Hossain S.A.	Masum A.K.M.	Alami N.	Meknassi M.	Chen J.	Wang X.	Yu H.	Chen Q.	Chen X.	Li P.	Wang H.	Zhang C.	Ferreira R.	Freitas F.	Lins R. D.	Simske S. J.	Lloret E.	Palomar M.	Saggion H.	Vodolazova T.	Wang J.	Yang Z.	Zhang Y.	Zhang L.	Zhang X.	Wang Y.	Zhang H.
Abujar S.	-	8	6																									
Hossain S.A.	8	-	5																									
Masum A.K.M.	6	5	-																									
Alami N.				-	5																							
Meknassi M.				5	-																							
Chen J.						-	1	1																	1			
Wang X.						1	-		1																1			1	1
Yu H.						1		-																	1			
Chen Q.							1		-																			
Chen X.										-	1	1	1	1														
Li P.										1	-		1															
Liu X.										1		-		1														
Wang H.										1	1		-															
Zhang C.										1		1		-												1	1	
Ferreira R.															-	4	4	4										
Freitas F.															4	-	5	5										
Lins R. D.															4	5	-	5										
Simske S. J.															4	5	5	-										
Lloret E.																			-	11	1	4						
Palomar M.																			11	-	1	2						
Saggion H.																			1	1	-							
Vodolazova T.																			4	2		-						
Wang J.																							-	3				1
Yang Z.																							3	-				
Zhang Y.						1	1	1																	-	1		1	
Zhang L.														1											1	-			
Zhang X.														1													-		
Wang Y.							1																		1			-	
Zhang H.							1																1						-

см. Рис. 2.6.

Таблица 2.9 показывает наиболее цитируемые статьи в области исследования Extractive Text Summarization согласно базе данных Scopus. Заметно, что самые цитируемые публикации были до 2000 года, но LexRank вошел в топ-2 статей и является самым новым из предложенного списка.

Таблица 2.10 показывает аналогичную статистику, но только для абстрактного суммирования. В топ-10 самых цитируемых работ вошли статьи, опубликованные после 2015 года, а в топ-5 - статья от 2020 года с предложенным методом абстрактного суммирования Bottom-up.

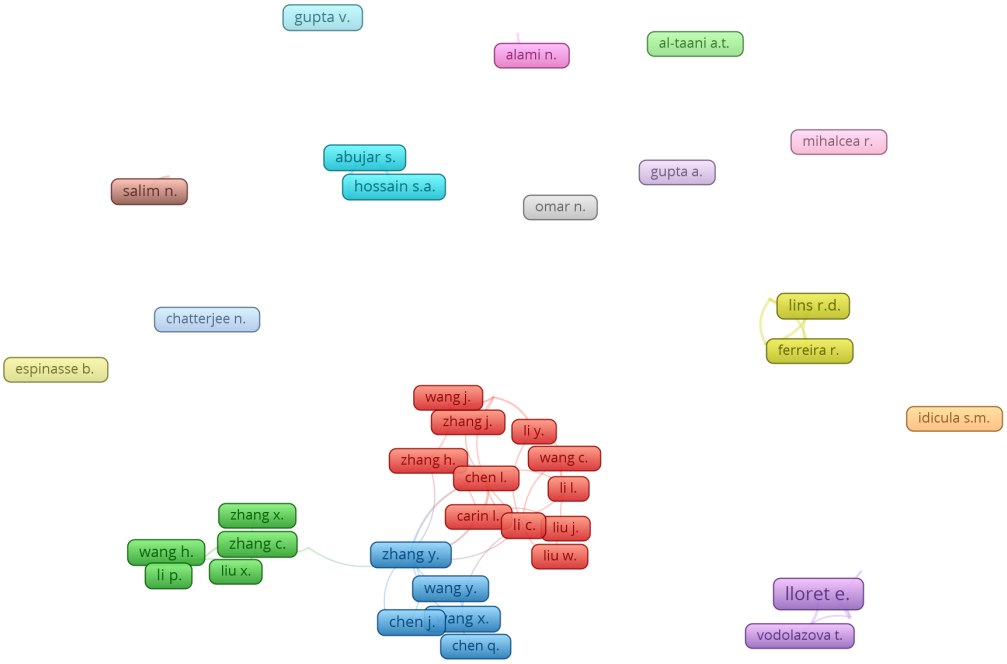


Рисунок 2.2 – Авторские отношения

.

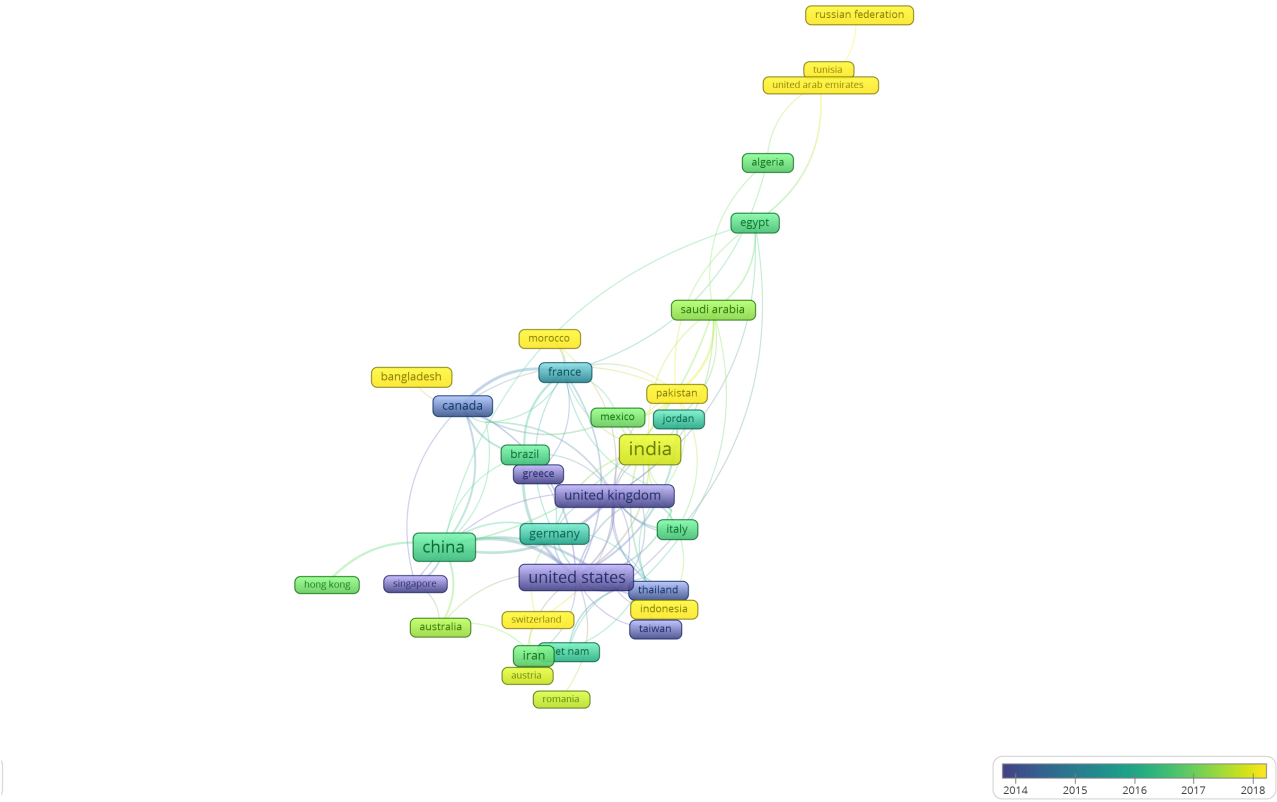


Рисунок 2.3 – Авторские отношения по странам.

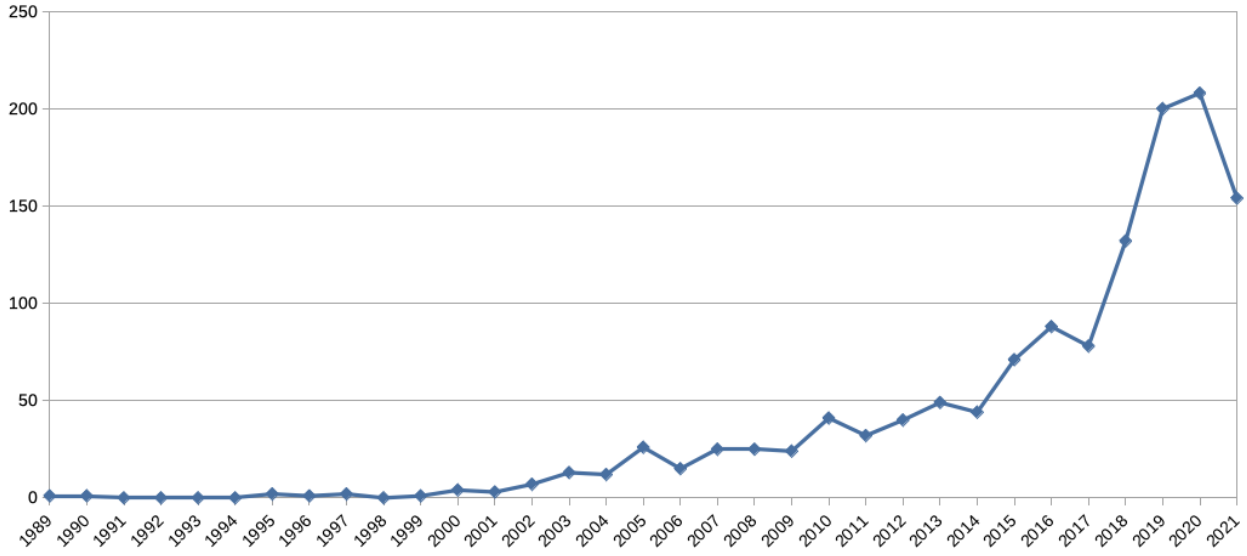


Рисунок 2.4 – Документы по годам.

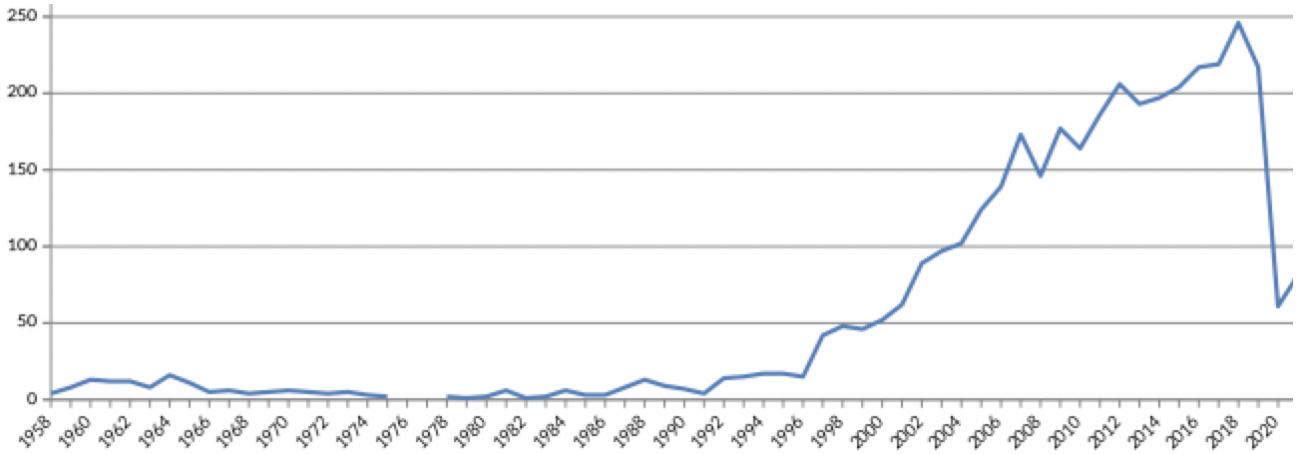


Рисунок 2.5 – Взвешивание слов/предложений (подсчет ссылок)

.

2.2 Методы автоматического реферирования текста

С тех пор, как в 1958 году Luhn опубликовал первую работу по теме обобщения текста [4], увидело свет множество научных работ, развивающихся от чисто статистических до более современных методов машинного обучения (ML) [33] и современных методов глубокого обучения [55–57].

Предложенный в 2004 году стохастический метод экстрактивного суммирования на основе графа, который вычисляет относительную важность текстовых единиц, подобно известному алгоритму PageRank [58], используемому раннее Google для ранжирования веб-страниц. Важность предложения в LexRank вычисляется на основе центральности собственных векторов в графовом представлении предложений. Кроме того, в качестве матрицы смежности используется мера косинусного сходства между предложениями.

Метод SumBasic [59], о котором сообщалось в 2005 году, использует исклю-

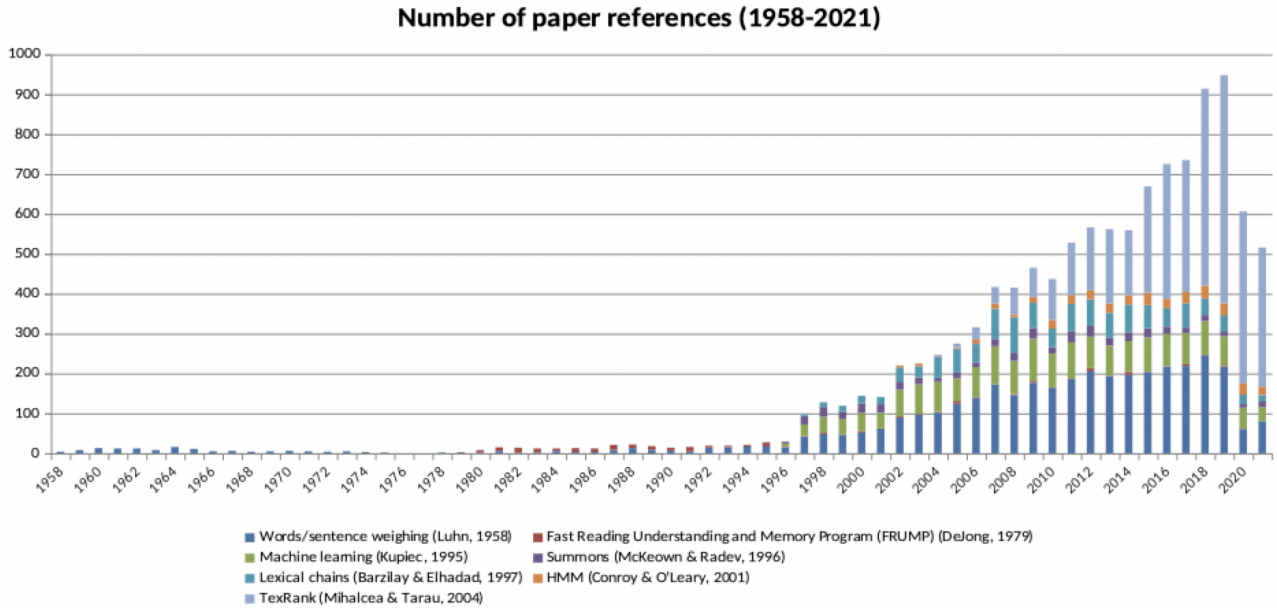


Рисунок 2.6 – Другие методы обобщения (количество ссылок)

.

чительно частоту для создания резюме. Он подтверждает, что только частота является мощной характеристикой при создании резюме, что очень похоже на то, что наш метод в значительной степени зависит от параметра минимальной частоты документа (описано в Раздел b)). Подход SumBasic также демонстрирует возможность включения корректировки контекста и удаления дубликатов в многодокументное обобщение.

Современные методы абстрактного суммирования начали появляться в последние десять лет. В этих методах используются сложные нейросетевые архитектуры: последовательность-последовательность, РНС с осознанием дискурса и трансформеры.

Рекуррентные нейронные сети Attentional Encoder-Decoder (Attn-Seq2Seq) моделируют абстрактное реферирование текста [47]. Метод позволяет моделировать ключевые слова, улавливать иерархию структуры от предложения к слову и редкие слова, не замеченные во время обучения. Несмотря на то, что модели последовательности к последовательности предоставили новый жизнеспособный подход для абстрактного обобщения текста, они имеют два существенных недостатка: 1) неспособность точно воспроизвести фактические детали и 2) привнесение избыточности в резюме [60]. Поэтому был предложен метод (Pntr-Gen-Seq2Seq), дополняющий стандартную модель внимания от последовательности к последовательности. Метод решает проблему фактической точности с помощью гибридной сети указатель-генератор, которая копирует слова из исходного текста с помощью указателя, сохраняя способность новых слов, созданных с помощью генератора. Кроме того, избыточность устраняется за счет использования охвата для отслеживания того, ка-

Таблица 2.9 – Самые цитируемые статьи в области исследований экстрактивного суммирования текста.

	Название	Авторы	Цитирование	Год
1	Использование MMR, рерайтинга на основе разнообразия для упорядочивания документов и составления резюме [38]	Carbonell J., Goldstein J.	1736	1998
2	LexRank: Основанная на графике лексическая центральность как значимость при реферировании текста [26]	Erkan G., Radev D.R.	1633	2004
3	Обучаемый обобщающий анализатор документов [33]	Kupiec J., Pedersen J., Chen F.	775	1995
4	TextTiling: Сегментирование текста на многопараграфные подтемы [39]	Hearst M.A.	774	1997
5	Поиск и обобщение рецензий на фильмы [40]	Zhuang L., Jing F., Zhu X.-Y.	635	2006
6	Алгоритмы обучения для извлечения ключевых фраз [41]	Turney P.D.	600	2000
7	Обобщение текста с использованием меры релевантности и латентного семантического анализа [42]	Gong Y., Liu X.	560	2001
8	Incorporating copying mechanism in sequence-to-sequence learning [43]	Gu J., Lu Z., Li H., Li V.O.K.	522	2016
9	Выведение иерархий понятий из текста [44]	Сандерсон М., Крофт Б.	431	1999
10	Резюме текстовых документов: Выбор предложений и метрики оценки [45]	Goldstein J., Kantrowitz M., Mittal V., Carbonell J.	323	1999

кая информация включена в резюме.

Модель Discourse-Aware Attention для абстрактного реферирования длинных документов (Discourse-att) была предложена Коханом [61], состоящая из иерархического кодера, моделирующего структуру дискурса документа, и внимательного декодера с учетом дискурса, генерирующего само резюме. В результате этой работы были получены два набора данных с большим количеством научных документов, которые мы используем в текущем исследовании; см. Раздел a).

Модель PEGASUS [55] появилась в конце 2019 года, используя метод предварительного обучения больших моделей кодеров-декодеров на основе трансформаторов на массивных текстовых корпорациях (C4 и HugeNews) с новой самоконтролируемой целью. В модели значимые предложения маскируются из входного документа и генерируются вместе как одна выходная последова-

Таблица 2.10 – Самые цитируемые статьи в области исследования абстрактивного реферирования текста.

	Название	Авторы	Цитирование	Год
1	Ближе к делу: Суммирование с помощью сетей генераторов указателей [31]	See A., Liu P.J., Manning C.D.	955	2017
2	Нейронная модель внимания для обобщения предложений [46]	Rush A.M., Chopra S., Weston J.	759	2015
3	Abstractive text summarization using sequence-to-sequence RNNs and beyond [47]	Nallapati R., Zhou B., dos Santos C., Gulçehre Ç., Xiang B.	597	2016
4	Абстрактное реферирование документов с помощью нейронной модели внимания на основе графов [48]	Tan J., Wan X., Xiao J.	158	2017
5	Bottom-up abstractive summarization [49]	Gehrmann S., Deng Y., Rush A.M.	146	2020
6	Глубокие коммуникативные агенты для абстрактного обобщения [50]	Celikyilmaz A., Bosselut A., He X., Choi Y.	114	2018
7	Toward abstractive summarization using semantic representations [51]	Liu F., Flanigan J., Thomson S., Sadeh N., Smith N.A.	106	2015
8	Deep recurrent generative decoder for abstractive text summarization [52]	Li P., Lam W., Bing L., Wang Z.	85	2017
9	Абстрактное обобщение текста с помощью глубокого обучения на основе LSTM-CNN [53]	Song S., Huang H., Ruan T.	76	2019
10	A framework for multi-document abstractive summarization based on semantic role labelling [54]	Khan A., Salim N., Jaya Kumar Y.	75	2015

тельность из остальных предложений, аналогично экстрактивному суммированию.

Что касается оценки составленного резюме, широко используется показатель ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [62]. Впервые она была предложена на конференции по пониманию документов (DUC) в 2004 году. Основная идея метрики основана на подсчете количества словесных и/или фразовых совпадений, также известных как n-граммы, между сгенерированным градуированным резюме и превосходным золотым стандартом, созданным человеком. Несмотря на то, что существует множество способов измерения сходства между резюме эталона и резюме кандидата, метрика ROUGE остается стандартной задачей внутритекстового реферирования.

Автоматическое обобщение текста позволяет уменьшить его объем, сохра-

няя при этом основную информацию. Модели реферирования текста обычно классифицируются как экстрактивные или абстрактные, однодокументные или многодокументные; см. Таблица 2.11. Важно отметить, что абстрактные модели обобщения могут формировать информативные, ориентировочные рефераты и смешанные - все зависит от набора данных, на котором обучалась модель.

Таблица 2.11 – Описание предлагаемых моделей.

Метод	Абстрактный	Экстрактный	Однодокументный	Многодокументный	
Luhn			✓	✓	✓
TextRank		✓	✓	✓	
LexRank		✓	✓	✓	
SumBasic		✓		✓	
LSA		✓		✓	
KL-sum		✓		✓	
PEGASUS	✓		✓	✓	
BigBird PEGASUS	✓		✓	✓	
T5	✓			✓	
BART	✓	✓	✓	✓	
HatBART	✓	✓	✓	✓	
GPT-2	✓		✓	✓	
GPT-3	✓	✓	✓		
SimCLS	✓	✓	✓	✓	
УниЛМ	✓	✓	✓		

2.2.1 Экстрактивные методы автоматического реферирования текста

а) Luhn

Один из самых ранних примеров такого типа методов автоматического реферирования текстов был представлен еще в 1958 году в работе [4]. Подход алгоритма суммирования Луна был основан на оценке терминов текста по частоте, отборе предложений с наиболее важными терминами для построения резюме:

1 Игнорировать стоп-слова: Игнорируются высокочастотные слова, такие

как артикли, предлоги, местоимения, которые не несут семантики, а выполняют служебную функцию в тексте.

2 Определить высокочастотные слова: Подсчитываются высокочастотные слова документа.

3 Выбор лучших слов: Для подсчета баллов отбирается относительно небольшое количество наиболее частотных слов.

4 Выбор лучших предложений: Оценка предложений в соответствии с их наибольшим содержанием слов. Четыре лучших предложения отбираются для резюме.

Он полезен, когда очень низкочастотные и высокочастотные слова (стоп-слова) не являются значимыми.

Этот метод можно считать первым открытием в области обобщения текстов. Например, в статье о методе жадной оптимизации для обобщения научных статей используется основная идея экстрактивного подхода Луна [63].

b) TextRank

[35] предложил основанный на теории графов алгоритм обобщения текста под названием TextRank [35], использующий концепцию ранее известного алгоритма PageRank от Google [64], который представляет предложения в тексте в виде вершин графа, а связи между предложениями в виде ребер. Каждая из вершин графа оценивается в соответствии с семантической связанностью предложения со всеми другими предложениями в тексте (аналогично количеству гиперссылок на страницу с других страниц в алгоритме PageRank), вычисляемой рекурсивно по всему графу (2.1):

$$S\left(V_i\right) = (1-d) + d \times \sum_{j \epsilon In(V_i)} \frac{1}{|Out\left(V_j\right)|} S\left(V_j\right), \tag{2.1}$$

где V_i - вершина, $In\left(V_i\right)$ - множество вершин, указывающих на нее, и $Out\left(V_i\right)$ - множество вершин, на которые указывает вершина, d - коэффициент затухания, принимающий значения 0 и 1, который играет интегрирующую роль для расчета вероятности перехода из данной вершины в любую другую случайную вершину графа.

После вычисления сходства между всеми предложениями строится граф, в котором каждая вершина может быть не связана ни с какой другой вершиной из-за отсутствия сходства между предложениями, представленными вершинами. Ребра, соединяющие две вершины, имеют вес, отражающий силу сходства. Наконец, алгоритм резюмирует документ, основываясь на наиболее значимых предложениях и ключевых фразах.

c) LexRank

Алгоритм LexRank был разработан в 2004 году [26] в Мичиганском университете. Алгоритм оценивает предложения по важности, используя концепцию центральности собственных векторов в графовом представлении предложений. Он использует внутрисентенционное косинусное сходство для матрицы смежности предложений.

Алгоритм основан на теории графов. Предложения с удаленными стопсловами в тексте становятся вершинами графа, а ребра строятся, сравнивая сходство предложений с помощью IDF-модифицированного косинуса в (2.2):

$$idf - modified - cosine\,(x,y) = \frac{\sum_{w \epsilon x,y} tf_{w,x} tf_{w,y}\,(idf_w)^2}{\sqrt{\sum_{x_i \epsilon x}\left(tf_{x_i,x} idf_{x_i}\right)^2}\sqrt{\sum_{x_i \epsilon x}\left(tf_{y_i,y} idf_{y_i}\right)^2}}, \quad (2.2)$$

где $tf_{w,s}$ - количество вхождений слова w в предложение s, а $idf_w = \log\left(\frac{N}{n_w}\right)$.

После построения графа к нему применяется алгоритм PageRank Google [64]. Таким образом, предложения с высоким рейтингом похожи на многие другие предложения в тексте. Результирующее резюме создается путем выбора предложений с наивысшим рейтингом x, где пользователь определяет x как желаемое количество предложений в резюме.

d) SumBasic

SumBasic - это алгоритм, который выбирает предложения на основе частоты с компонентом повторного веса для вероятностей слов, чтобы минимизировать избыточность [59].

В SumBasic каждому предложению S присваивается балл, основанный на содержании в нем высокочастотных слов (2.3):

$$Score\,(S) = \sum_{w \epsilon S} \frac{1}{|S|} P_D\,(w)\,, \quad (2.3)$$

где P_D - наблюдаемые вероятности униграмм, полученные из коллекции документов D. Резюме постепенно строится путем добавления предложения с наивысшей оценкой. Чтобы избежать избыточности, важность слов в выбранном предложении обновляется $P^D_{new}\,(w) = P^D_{old}\,(w)^2$. Предложения выбираются таким образом до тех пор, пока мы не достигнем предела суммарных слов.

e) Latent Semantic Analysis (LSA)

LSA - это математико-статистический метод, который извлекает скрытые семантические структуры слов и предложений без наблюдения [65]. LSA использует контекст входного документа и фиксирует совпадение слов и то, какие общие слова используются в различных предложениях.

Значительное количество слов, встречающихся в предложениях, указывает на то, что они семантически связаны. Это связано с тем, что смысл предложения определяется содержащимися в нем словами, а значения слов определяются другими словами в предложении, определяющими контекст.

Сингулярное разложение значений (SVD), алгебраический метод, используется для выявления взаимосвязей между предложениями и словами [66]. Помимо моделирования взаимосвязей между словами и предложениями, SVD также способен подавлять шумы для повышения точности.

f) алгоритм суммы Куллбэка-Либера (KL)

Алгоритм KL Sum выбирает предложения из исходного текста, где длина резюме фиксирована и составляет L слов. Он добавляет предложения в сводку с жадностью до тех пор, пока они уменьшают расхождение KL. Цель алгоритма KL Sum - найти набор предложений, длина которых меньше L слов, а распределение униграмм близко к распределению униграмм исходного документа [67].

В математической статистике дивергенция KL (или относительная энтропия) измеряет, насколько два распределения вероятностей отличаются друг от друга. Чем меньше расхождение, тем больше резюме похоже на документ с точки зрения удобочитаемости и смысла, который несет [68].

KL вводит критерий суммарного отбора предложений для включения в коллекцию предложений S в документе D, как показано в (2.4).

$$S^* = min(S : words(S) \leqslant L, KL(P_D||P_S)) \tag{2.4}$$

где P_S - эмпирическое распределение униграмм в резюме кандидата S, а $KL\,(P||Q)$ представляет собой расхождение Куллбэка-Либера (KL), определяемое следующим образом $\sum_w P\,(w) \log \frac{P(w)}{Q(w)}$. Эта величина представляет собой расхождение между истинным распределением P и приближенным распределением Q.

Этот критерий рассматривает реферирование текста как поиск набора предложений для реферата из исходного текста, которые близко соответствуют исходному распределению униграмм.

2.2.2 Абстрактивные методы автоматического реферирования текста

a) Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence (PEGASUS)

PEGASUS основан на архитектуре seq2seq, как и любая другая задача преобразования последовательности. Тем не менее, новизна этой архитектуры заключается в использовании самоконтролируемой цели для обучения модели трансформатора, называемой Gap Sentences Generation (GSG) [69].

Модель маскирует значимые предложения из входного документа и генерирует их обратно в виде выходной последовательности со всеми остальными предложениями. PEGASUS маскирует те предложения из текста, которые наиболее похожи на предложения эталонного резюме. Поэтому предсказание таких предложений максимизирует оценку ROUGE резюме-кандидата.

Хотя основным вкладом PEGASUS является GSG, он имеет трансформаторную архитектуру; таким образом, имеет смысл предварительно обучить кодировщик как модель языка с маской (MLM). MLM случайным образом маскирует слова последовательности и использует другие последовательности для предсказания этих замаскированных слов. Задача GSG может рассматриваться как MLM на уровне документа и вытекает из этой концепции.

b) BigBird PEGASUS

BigBird - это трансформатор на основе разреженного внимания, расширяющий модели на основе трансформаторов, такие как BERT, на гораздо более длинную последовательность. В дополнение к разреженному вниманию, BigBird также имеет глобальное внимание и случайное внимание к входной последовательности [70]. Теоретически было показано, что применение разреженного, глобального и случайного внимания примерно соответствует уровню полного внимания и при этом является вычислительно менее сложным для более длинных текстовых последовательностей. Благодаря способности обрабатывать более широкий контекст, BigBird показал более высокую производительность при реферировании по сравнению с BERT или RoBERTa.

c) Text-to-Text-Transfer-Transformer (T5)

Модель T5 предлагает решать все задачи NLP в едином текстовом формате, когда входные и выходные данные являются текстовыми строками [71]. Модель T5 является эквивалентом оригинального Transformer, предложенного [72]. Тонкое отличие модели T5 от ранее обученных моделей MLM заключается в замене нескольких последовательных лексем одним ключевым словом Mask. Во время предварительного обучения T5 преобразует исходный текст в пары "вход"и "выход"путем добавления шума.

d) Bidirectional and Auto-Regressive Transformer (BART)

Недавно представленный BART состоит из двух основных компонентов: двунаправленного кодера и авторегрессионного декодера, которые имеют архитектуру на основе трансформатора и реализованы в виде модели seq2seq [73].

Базовая модель BART использует шесть слоев как в кодере, так и в декодере, в то время как Большая модель имеет 12 слоев. При предварительном обучении режима BART применяются следующие методы:

1 Token Masking: случайное подмножество входных данных заменяется лек-

семами [MASK], как и в модели BERT.

2 Token Deletion: случайные лексемы удаляются из входных данных, и модель должна решить, чего не хватает.

3 Заполнение текста: несколько отрезков текста разной длины заменяются одним маркером [MASK].

4 Перестановка предложений: перестановка входных предложений.

5 Вращение документа: случайно выбирается маркер, и последовательность поворачивается так, чтобы начать с выбранного маркера.

e) Иерархическое внимание BART (HatBART)

HatBART - новая архитектура на основе иерархического трансформера внимания, превосходящая стандартные трансформеры в нескольких задачах seq2seq [74].

Авторы модифицировали стандартную последовательность в последовательность архитектура трансформатора [72]по добавление иерархического внимания для улучшения обработки длинных документов. Количество параметров для большой иерархической модели на суммарных задачах составляет 471M по сравнению с простым трансформатором 408M.

Использовались двенадцать слоев кодера и декодера, скрытый размер 1024, 4096 для размерности полносвязных сетей с фид-форвардом и 16 головок внимания как в кодере, так и в декодере. В отличие от оригинального Transformer, вместо ReLU используется активация GELU.

f) Generative Pre-trained Transformer (GPT)

GPT-2 GPT-2 - это огромная языковая модель на основе трансформатора, имеющая 1.5B параметров, обученная на наборе данных 8M вебстраниц [75]. GPT-2 обучается с целью предсказать следующее слово, принимая во внимание предыдущие слова в тексте. Он использует парное кодирование байтов (BPE) для построения лексических лексем, что означает, что они обычно являются частями слова, и выводит по одной лексеме за раз.

Модель получает только один входной токен, поэтому активным будет только один путь. Токен последовательно проходит через все слои, и на выходе получается вектор, который может быть оценен с помощью словаря модели. В этом случае выбирается лексема с наибольшей вероятностью. Кроме того, в GPT-2 есть параметр top-k, который можно использовать для того, чтобы модель рассматривала выборку слов, отличных от верхнего слова. Затем добавьте выход первого шага к входной последовательности и попросите модель сделать следующее предсказание.

Каждый уровень GPT-2 сохраняет интерпретацию первого маркера и использует ее при обработке второго маркера. Таким образом, GPT-2 не переосмысливает первый токен в свете второго токена.

GPT-3 По своей сути GPT-3 является трансформаторной моделью, которая представляет собой модель глубокого обучения seq2seq, создающую текстовую последовательность на основе входной последовательности. Модели этого типа предназначены для задач генерации текста, таких как ответы на вопросы (QA), обобщение текста (TS) и машинный перевод (MT). GPT-3 - это третье поколение языковой модели GPT от OpenAI. Основным отличием GPT-3 от предыдущих моделей является ее огромный размер. GPT-3 содержит 175B параметров, что делает ее в 17 раз больше предшественницы GPT-2 и примерно в десять раз больше модели Turing NLG от Microsoft [76].

Производительность GPT-3 на три порядка выше, чем у GPT-2, без существенных изменений в архитектуре модели, просто более многочисленные и широкие слои с большим количеством обучающих данных.

g) Simple Framework for Contrastive Learning of Abstractive Summarization (SimCLS)

SimCLS - это концептуально простая структура для абстрактного обобщения текста. Модель устраняет разрыв между целью обучения и метриками оценки, возникающий в результате доминирующей в настоящее время системы обучения seq2seq, рассматривая генерацию текста как проблему оценки без ссылок с помощью контрастного обучения [77].

Система SimCLS для двухэтапного абстрактного обобщения:

1 BART используется для составления резюме кандидатов.

2 Скоринговая модель RoBERTa используется для прогнозирования качества резюме кандидатов на основе содержания исходного документа.

h) Unified pre-trained Language Model (UniLM)

UniLM - это многослойная NN, состоящая из нескольких моделей ИИ Transformer, совместно предварительно обученных на больших объемах текстовых данных и оптимизированных для языкового моделирования. Модели построены таким образом, что каждый выходной элемент связан с каждым входным элементом, и в результате весовые коэффициенты между ними рассчитываются динамически.

Предварительно обученный UniLM похож на BERT, и по запросу он может быть тонко настроен для адаптации к различным последующим задачам NLP. В отличие от BERT, UniLM может быть настроен с использованием различных масок самовнимания для агрегирования контекста для различных языковых моделей [78]. Кроме того, благодаря унифицированному характеру предварительного обучения, сети-трансформеры могут совместно использовать параметры, что делает выученные представления текста более общими и, таким образом, уменьшает чрезмерную подгонку под какую-либо одну задачу.

2.3 Результаты сравнения существующих моделей

Все эти модели продемонстрировали значительные результаты при реферировании текста. В Таблица 2.12 и Таблица 2.13 мы представляем оценку шести описанных экстрактивных алгоритмов на наборах данных DUC2001, CNN/Daily Mail, XSum и BigPatent, описанных в Глава 3.1, на основе ROUGE-1 и ROUGE-2[7]. Согласно метрикам ROUGE-1 и ROUGE-2 на массивах данных DUC2001, CNN/Daily Mail, XSum и BigPatent можно выделить три наиболее успешные модели: Luhn, TexRank и LexRank. Интересно, что на наборе данных XSum модель SumBasic имеет максимальный показатель ROUGE-1 всего 0.19, а у модели LexRank максимальное значение ROUGE-2 равно 0.03.

Таблица 2.12 – Результат работы моделей экстрактивного суммирования на наборах данных DUC2001, CNN/Daily Mail, XSum. R1 и R2 означают ROUGE-1 и РОУГ-2 соответственно

	DUC2001		CNN/Daily Mail		XSum	
	R-1	R-2	R-1	R-2	R-1	R-2
Luhn	0.42	**0.17**	/	/	/	/
TextRank	0.40p	0.15	**0.40**	**0.18**	/	/
LexRank	**0.42**	0.16	0.35	0.13	0.18	0.03
LSA	0.35	0.12	/	/	/	/
SumBasic	0.36	0.11	0.34	0.11	**0.19**	0.02
KLSum	0.35	0.12	0.30	0.11	0.17	0.02

Таблица 2.13 – Результат работы моделей экстрактивного суммирования на наборах данных BigPatent, ArXiv, PubMed. R1 и R2 означают ROUGE-1 и ROUGE-2 соответственно

	BigPatent		ArXiv		PubMed	
	R-1	R-2	R-1	R-2	R-1	R-2
Luhn	/	/	/	/	/	/
TextRank	**0.36**	**0.11**	/	/	/	/
LexRank	0.35	0.10	**0.33**	**0.10**	**0.39**	**0.14**
LSA	/	/	0.29	0.07	0.34	0.10
SumBasic	0.27	0.07	0.29	0.06	0.37	0.11
KLSum	/	/	/	/	/	/

Оценка популярных абстрактивных алгоритмов реферирования представлена в Таблица 2.14 и в Таблица 2.15. Оценка качества моделей суммирования

[7]Мы опустили метрику ROUGE-L, поскольку она сильно коррелирует с метрикой ROUGE-1

проводится на самых больших наборах данных.

Таблица 2.14 – Результат работы моделей абстрактного обобщения на наборах данных CNN/Daily Mail, Gigaword, X-Sumdataset. R1 и R2 означают ROUGE-1 и ROUGE-2 соответственно

	CNN/Daily Mail		Gigaword		X-Sum	
	R-1	R-2	R-1	R-2	R-1	R-2
SimCLS	**0.47**	**0.22**	/	/	**0.48**	**0.25**
UniLM	0.43	0.20	0.39	**0.20**	0.43	0.20
T5	0.44	0.21	/	/	/	/
Bart	0.44	0.21	/	/	0.45	0.22
HAT-Bart	0.45	0.21	/	/	0.46	0.23
GPT-2	0.29	0.08	/	/	/	/
BigBird PEGASUS	0.44	0.21	/	/	0.47	0.24
PEGASUS	0.44	0.21	**0.399**	0.19	0.47	0.24

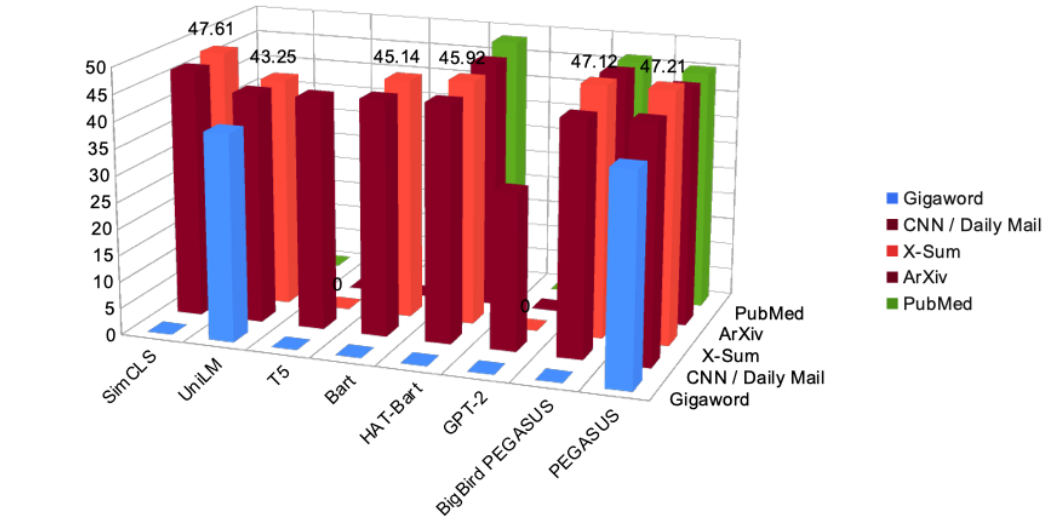
Таблица 2.15 – Результаты моделей абстрактного обобщения на наборах данных ArXiv и PubMed. R1 и R2 означают ROUGE-1 и ROUGE-2 соответственно

	ArXiv		PubMed	
	R-1	R-2	R-1	R-2
SimCLS	/	/	/	/
UniLM	/	/	/	/
T5	/	/	/	/
Bart	/	/	/	/
HAT-Bart	**0.47**	**0.20**	**0.48.25**	**0.21**
GPT-2	/	/	/	/
BigBird PEGASUS	0.46	0.19	0.46	0.20
PEGASUS	0.45	/	0.45	/

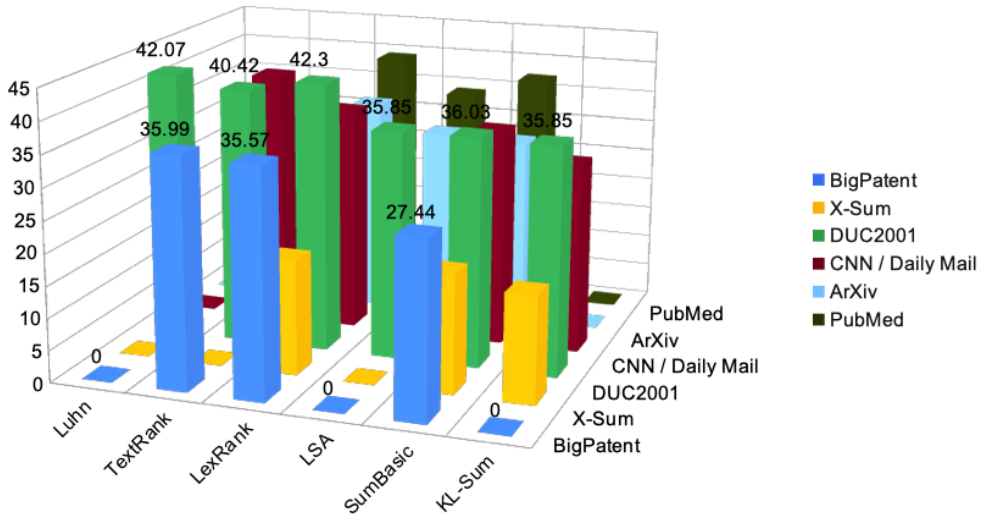
Можно отметить, что алгоритм SimCLS с показателями R-1 46,67 и 47,61 является лидером среди других моделей, включая экстрактивные алгоритмы. По метрикам ROUGE-1 и ROUGE-2 модели BigBird PEGASUS и PEGASUS также входят в тройку лидеров.

Остальные алгоритмы также демонстрируют хорошую способность к обобщению, отставая от лидеров на 1-2 пункта в метриках R-1 и R-2.

Проведя обзор современных моделей обобщения текстов, мы обнаружили, что в целом абстрактные модели превосходят экстрактивные, основанные на



(a) Абстрактные модели обобщения



(b) Модели экстрактивного суммирования

Рисунок 2.7 – ROUGE-1 для моделей абстрактного (a) и экстрактивного (b) суммирования на наборах данных CNN/Daily Mail, Gigaword, X-Sum, BigPatent, ArXiv и PubMed.

метрике ROUGE-1; см. Рис. 2.7.

2.4 Верхний предел качества АЭР

Большинство научных работ в области автоматического обобщения текстов (ATS) посвящено самим методам обобщения, но очень мало работ можно найти, исследующих верхнюю границу качества обобщений, которые могут быть сгенерированы.

Ceylan и др., работая над текстами в областях новостных, научных и юридических текстов, исследовали пространство резюме каждой области с помощью стратегии исчерпывающего поиска и нашли функцию плотности вероятности (pdf) распределения баллов ROUGE для каждой области. Затем, используя полученную pdf-функцию, они проранжировали существовавшие

на тот момент системы обобщения по процентилям [79].

Абстрактивные авторефераты, составленные людьми с использованием собственных слов, оставляют мало шансов для экстрактивного суммирования получить высокий балл ROUGE. В. М. Ванг и др. предлагают девять эвристических методов, генерирующих высококачественные резюме на основе предложений для длинных текстов из пяти различных корпораций. Они продемонстрировали, что результаты, достигнутые их эвристическими методами, близки к результатам алгоритмов Exhaustive (или Brute Force), но работают гораздо быстрее [80].

В данной работе мы использовали метод, представленный Н. Младеновичемć [81] алгоритм Variable Neighborhood Search (VNS) в качестве эвристики локального поиска для решения задачи максимума оценки по метрике ROUGE. Проще говоря, VNS берет начальное решение задачи и итеративно обновляет скорость изменения, когда не происходит улучшения в нахождении оптимума объективной функции, и фиксирует лучший результат.

2.5 Обзор существующих систем автоматического реферирования текстов

2.5.1 Приложения для автоматического реферирования текстов для Персональных Компьютеров (ПК))

a) Устаревшее и вышедшее из обращение Программное Обеспечение (ПО)

Microsoft Word 2007 В 2007 и более ранние версии продукта была включена функция Auto Summarize, позволяющая пользователям резюмировать длинные тексты до желаемой длины в % от размера исходного текста в формах:

- Выделение наиболее важных предложений в тексте.
- Составление резюме в начале документа.
- Замена текста на полученное резюме.

К сожалению, эта функция исчезла из последующих версий знаменитого текстового редактора.

Copernic Summarizer for Windows Разработанная *Copernic*.com[8] является еще одним примером заброшенного инструмента для подведения итогов. Сейчас он доступен для загрузки только на некоторых программных ресурсах protals[9] в режиме бесплатной пробной версии. Это простая программа для составления резюме, которая значительно повышает производительность и эффективность работы пользователя, создавая текстовые резюме из файлов или веб-ресурсов, экономя время чтения, не упуская ни кусочка важной информации.

b) Поддерживаемое ПО

Intellexer Summarizer *Intellexer Summarizer* - это программное обеспечение[10] от *EffectiveSoft Ltd.*[11] - приложение для автоматического реферирования текста, которое использует сложные алгоритмы ML и NLP для выполнения:

- Обобщение текста.
- Извлечение именованных сущностей.

Summarizer Разработана[12] компанией *Grupo Empresarial Multidisciplinario IASEC.* S.A. de C.V. (Мексика)[13] позволяет пользователям создавать индивидуальные резюме длины из различных форматов исходных документов, таких как PDF файл, документ Microsoft Word, коллекция изображений или веб-адрес. Дополнительные возможности

[8]Компания по разработке программного обеспечения, специализирующаяся на продуктах, помогающих людям искать информацию, `https://copernic.com/`

[9]`https://copernic-summarizer.en.softonic.com/`

[10]`https://summarizer.intellexer.com/`

[11]`https://www.effectivesoft.com/`

[12]`https://www.microsoft.com/en-us/p/summarizer/9wzdncrd2mvg`

[13]`https://iasec.com.mx/project/summarizer-ya-disponible/`

приложения включают:

- Извлечение ключевых слов.
- Пост-редактирование полученного резюме.
- Формирование речи из текста резюме.

2.5.2 Мобильные приложения

Linguakit *Linguakit*[14] - это приложение для Android от *Cilenis* Language Technology (Испания)[15], которое обладает функциональностью для обобщения, спряжения, перевода и анализа текстов [82]; см.Рис. 2.8. К прикладным инструментам относятся:

- Резюмируйте тексты.
- Проверка орфографии.
- Переводчик.
- Анализирует:
- Семантика.
- Синтаксис.
- Тональность.
- Извлечение именованных сущностей.

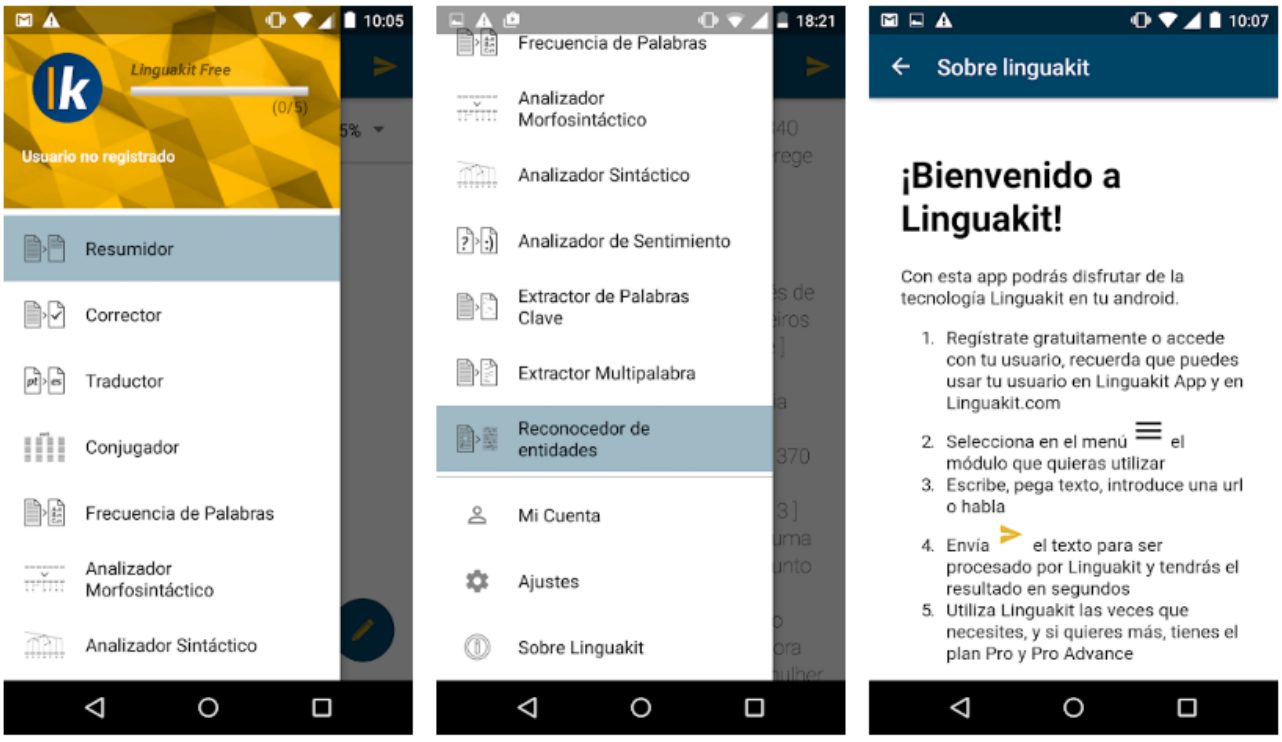


Рисунок 2.8 – Linguakit interface.

[14] https://play.google.com/store/apps/details?id=com.cilenis.linguakitandroid
[15] https://www.cilenis.com/

SumIt! SumIt![16] от Karim O.[17] использует простой и эффективный алгоритм извлечения ключевых предложений/ключевых фраз для автоматического составления резюме; см.Рис. 2.9. Приложение идеально подходит для занятых людей, помогая им анализировать и переваривать большие объемы информации из научных документов и любых других длинных текстов [82].

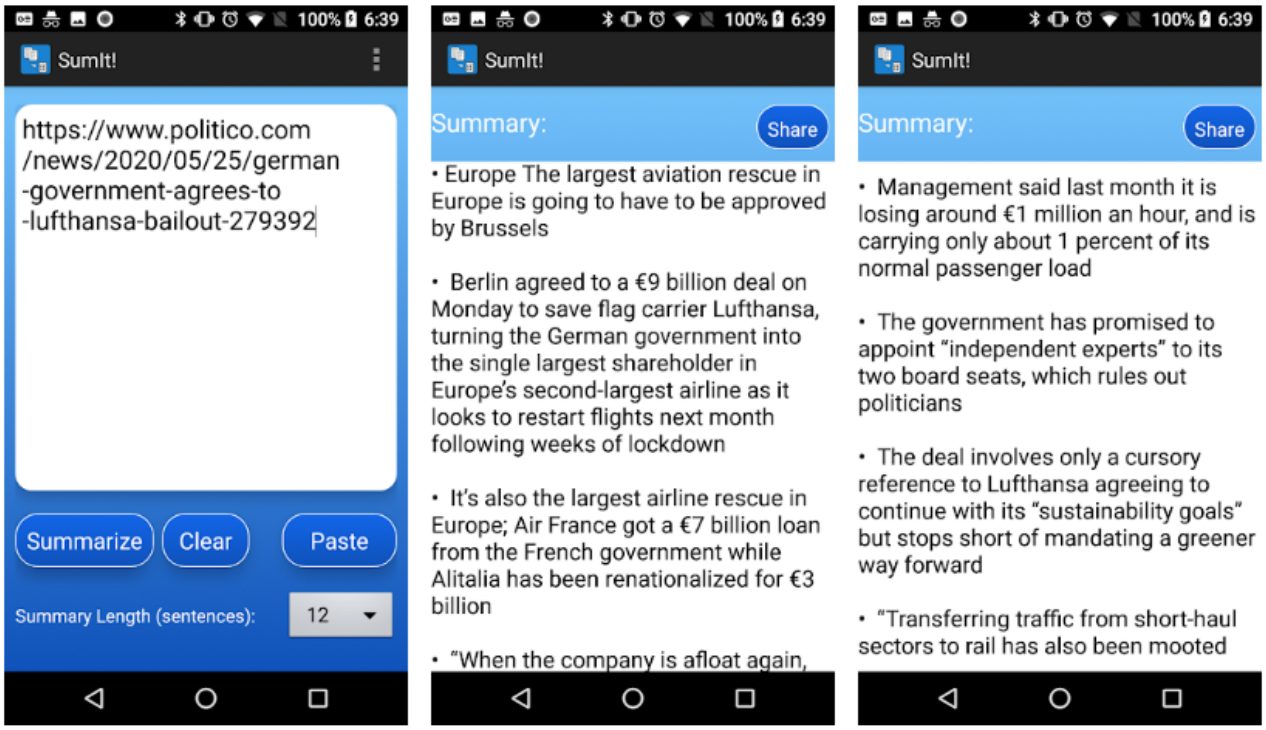


Рисунок 2.9 – SumIt! interface.

Text Summary *Text Summary*[18] Kreativity Apps[19] - это эффективный инструмент обобщения, позволяющий быстро извлечь важную информацию из любого текстового источника. Тексты могут быть предоставлены приложению путем вставки, указания URL или путем предоставления изображения отсканированного текста; см. Рис. 2.10.

Приложение позволяет сохранять полученный конспект в широком разнообразии форматов файлов (epub, pdf, docx, odt, pptx или обычный текст). Кроме того, приложение позволяет пользователю прослушать созданный конспект, что позволяет проверить, соответствует ли полученный конспект нашим требованиям, занимаясь другими делами [82].

Summarizer and Paraphraser *Summarizer and Paraphraser*[20] от Sudo AI[21] мгновенно извлекает ключевые моменты для реферирования текстов,

[16] https://play.google.com/store/apps/details?id=com.karimo.sumit_final
[17] https://play.google.com/store/apps/dev?id=5912932724675708500
[18] https://play.google.com/store/apps/details?id=com.aya.textsummarizer
[19] https://play.google.com/store/apps/developer?id=Kreativity+Apps
[20] https://play.google.com/store/apps/details?id=com.sudoai.sudo_summarizer
[21] https://play.google.com/store/apps/developer?id=sudo+ai

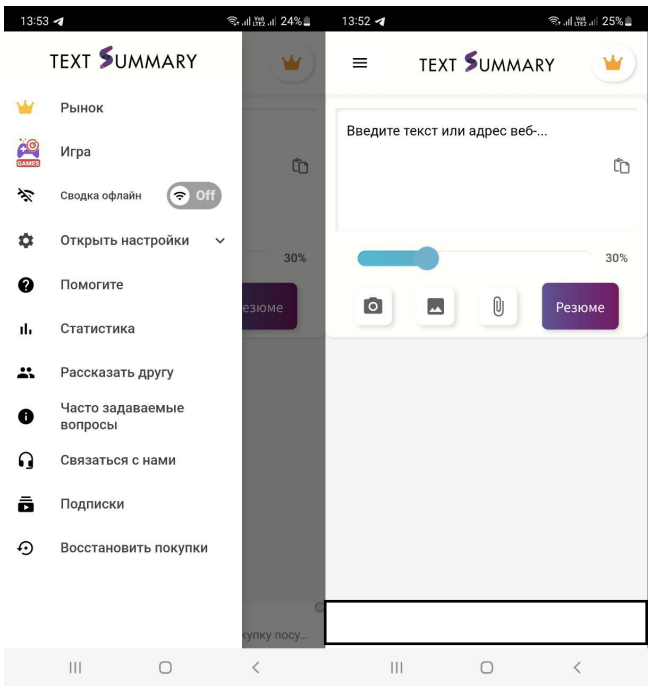


Рисунок 2.10 – Текст резюме интерфейса.

используя искусственный интеллект (ИИ), который способен обрабатывать естественный язык для определения наиболее важной информации из оригинального текста [82]; см. Рис. 2.11.

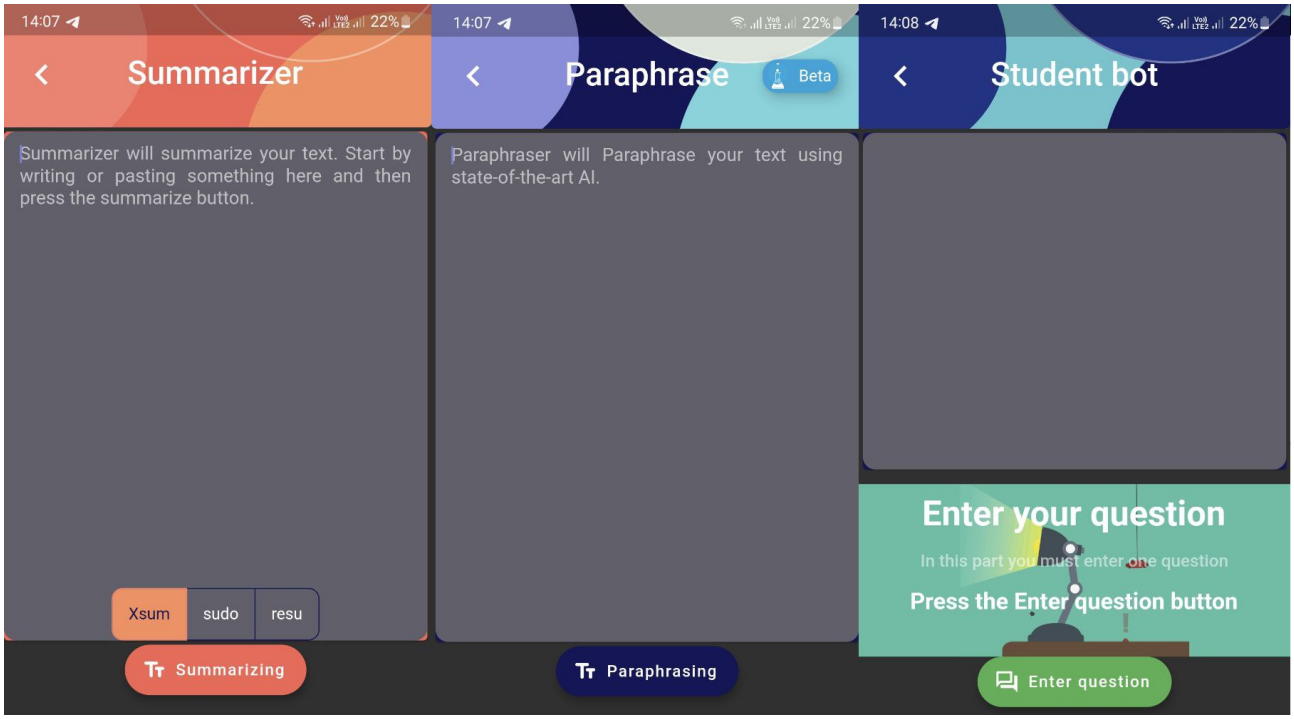


Рисунок 2.11 – Интерфейс суммирующего и перефразирующего устройства.

Встроенные функции:

- Paraphraser: позволяет сделать ваши резюме более уникальными и из-

бежать плагиата.

- Student-Bot: это модуль ответов на вопросы (QA), который по запросу может искать ответы в PDF, WikiPedia или книгах.

2.5.3 Веб-приложения

QuillBot *QuillBot*[22] резюмирующее устройство, включающее инструменты AI NLP, сжимает статьи, документы или бумаги, извлекая важную информацию, сохраняя при этом информационную ценность. Резюмирует тексты двумя способами:

1 Ключевые предложения: создает список наиболее значимых предложений.

2 Изложение: создает уникальную аннотацию, кратко описывающую содержание.

Для интерфейса QuillBot смотрите Рис. 2.12.

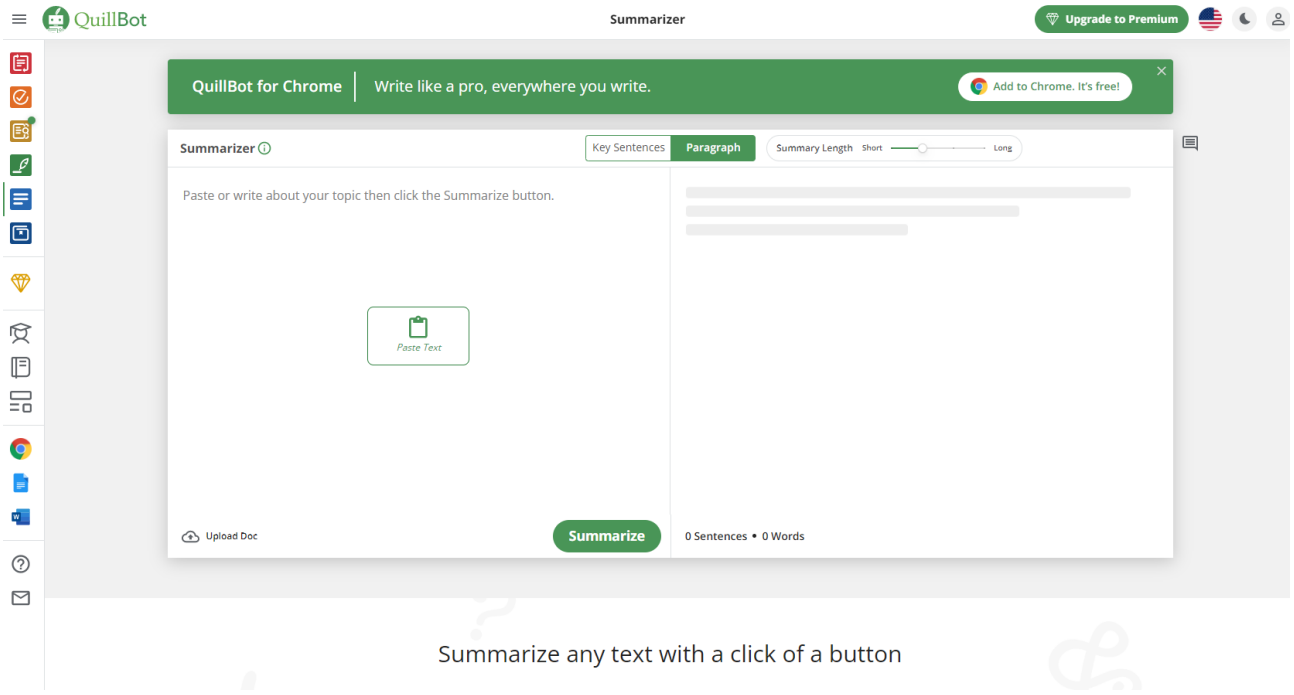


Рисунок 2.12 – QuillBot interface.

TLDRthis *TLDR*[23] помогает обобщить любой текст в сжатое, легко усваиваемое содержание и избежать информационной перегрузки; см. Рис. 2.13. Веб-приложение также автоматически извлекает метаданные текста, такие как:

- Информация об авторе и дате.
- Название и приблизительное время чтения.
- Похожие изображения.

Он также устраняет информационный шум, такой как реклама, графика и

[22] https://quillbot.com/summarize
[23] https://tldrthis.com/

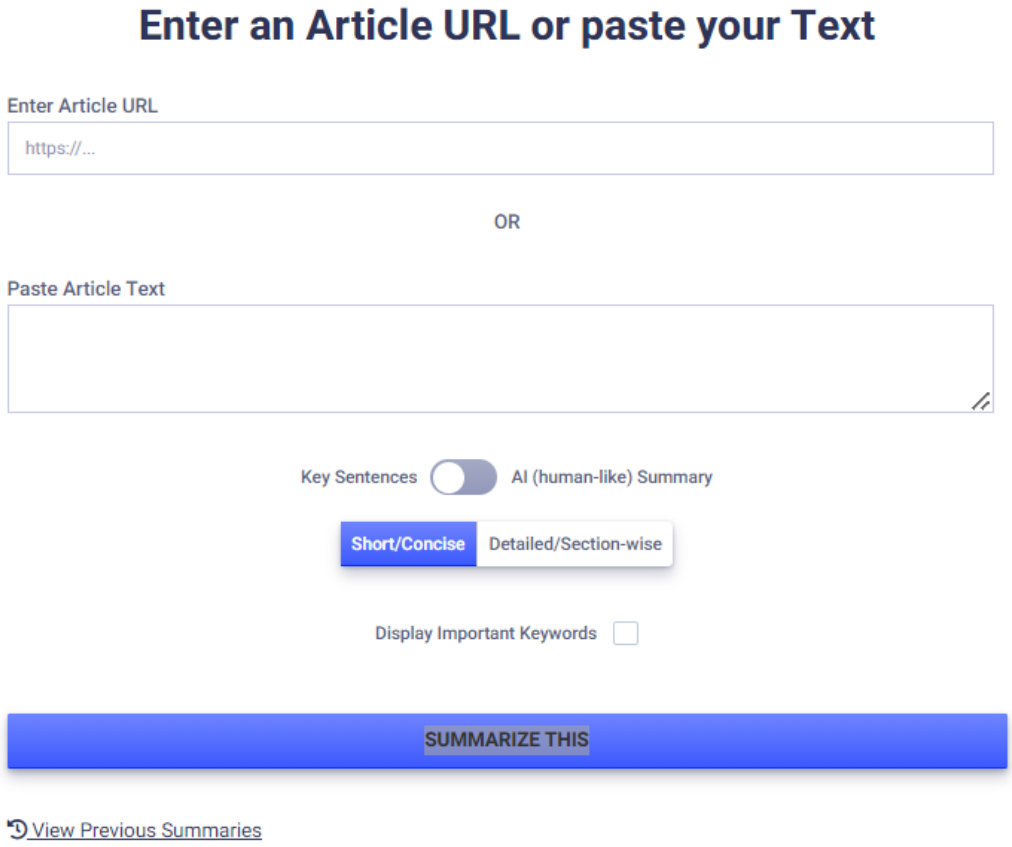


Рисунок 2.13 – Только интерфейс.

другие отвлекающие факторы в Интернете, обеспечивая четкое и сосредоточенное чтение.

Resoomer *Resoomer*[24] - это образовательный инструмент, обобщающий важную информацию из текстов. Он позволяет пользователю уловить главную идею или пробежаться по тексту, а также быстро интерпретировать тексты для разработки собственных синтезов; см. Рис. 2.14.

Splitbrain.org *Splitbrain*[25] - это веб-интерфейс (см. Рис. 2.15) к инструменту *Open Text Summarizer* (OTS)[26], который автоматически анализирует тексты на разных языках и пытается выявить наиболее важные части текста. Принцип OTS описывается идеей о том, что значимая для статьи информация содержится в ряде специфических терминов, в то время как избыточная информация использует менее технические термины и более распространенные высокочастотные слова.

Scholarcy *Scholarcy*[27] - это онлайн инструмент для реферирования статей

[24] https://resoomer.com/en/
[25] https://www.splitbrain.org/services/ots
[26] https://github.com/neopunisher/Open-Text-Summarizer/
[27] https://www.scholarcy.com/

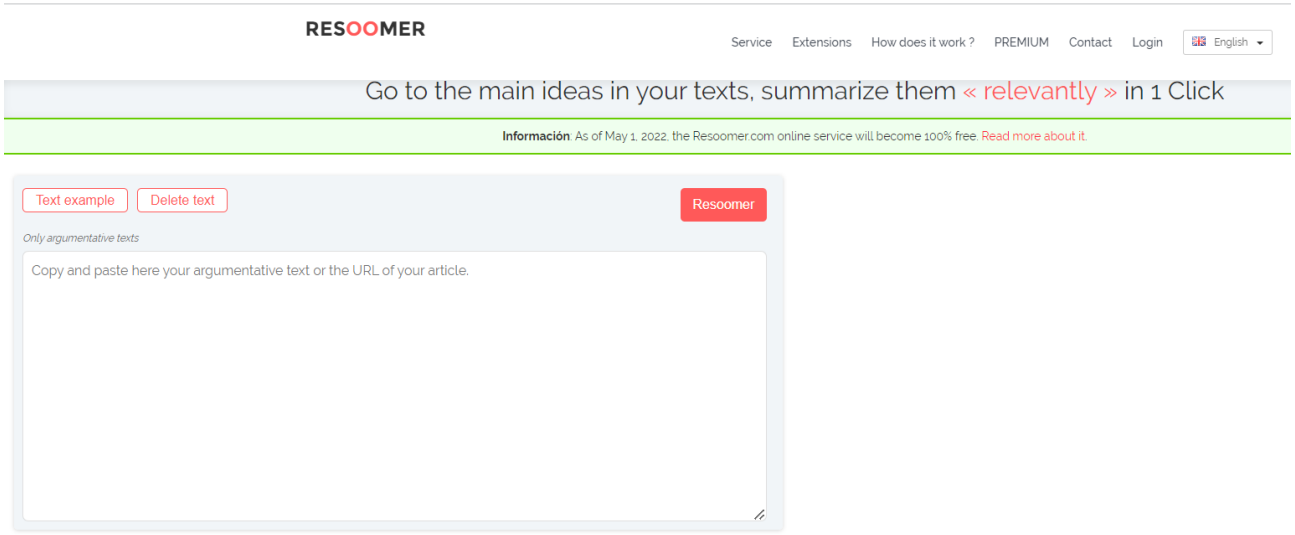


Рисунок 2.14 – Resoomer interface.

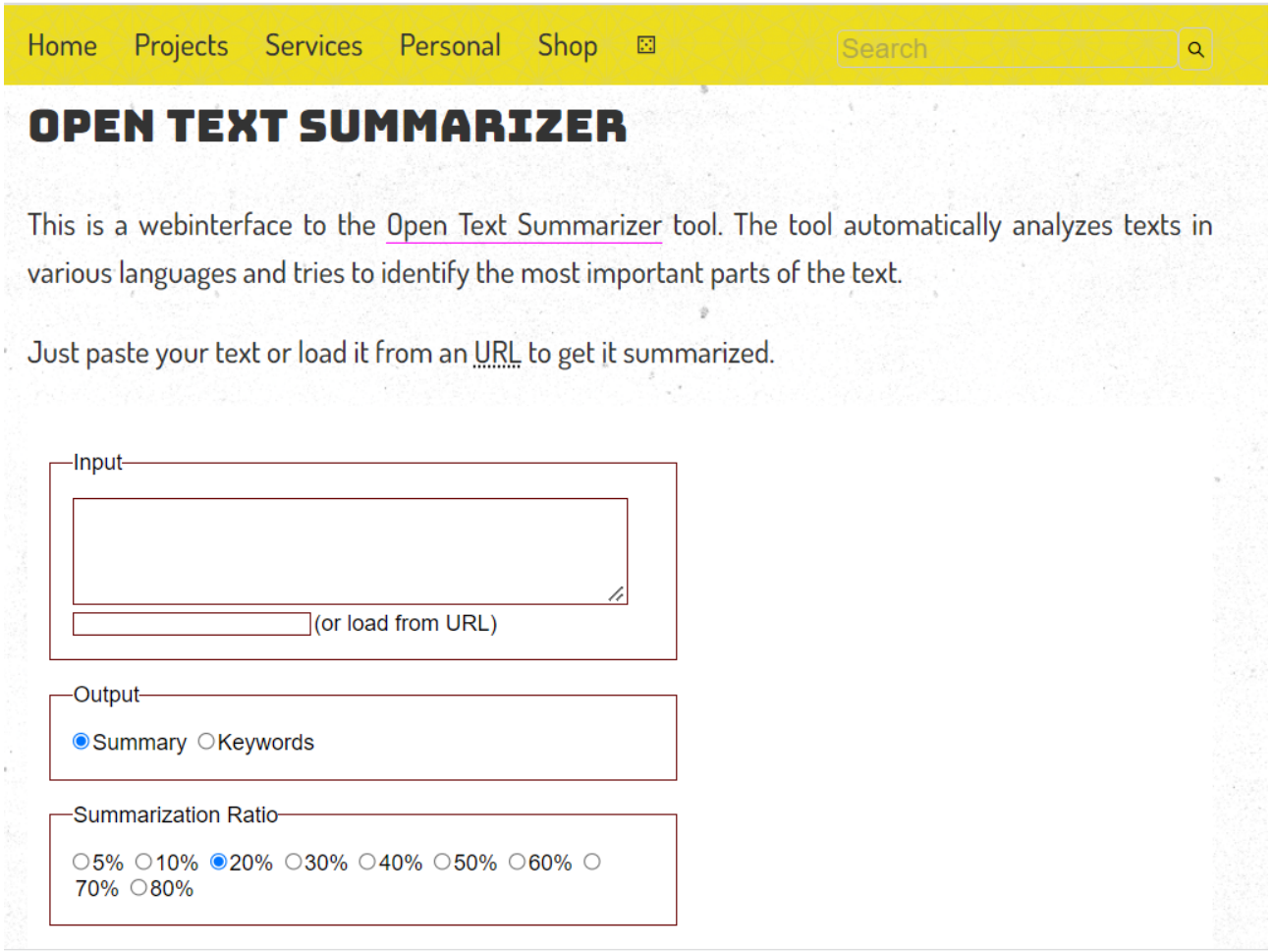


Рисунок 2.15 – Интерфейс Splitbrain.

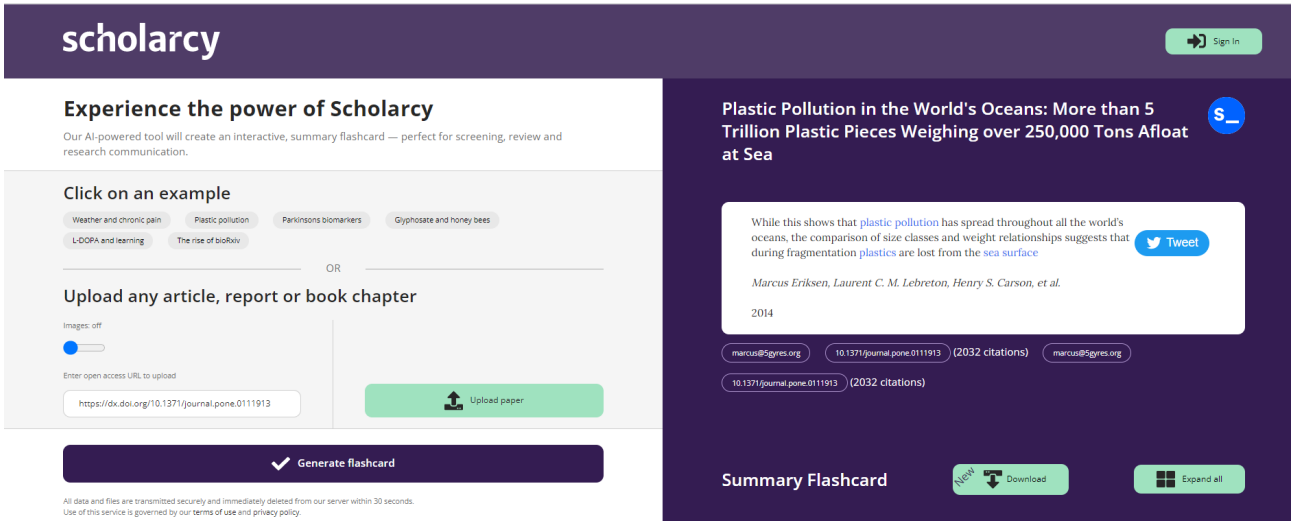


Рисунок 2.16 – Учебный интерфейс

.

(см. Рис. 2.16), который мгновенно обрабатывает отчеты, исследовательские статьи и главы книг и разбивает их на небольшие разделы, позволяя быстро оценить важность любого документа для работы, которой занят исследователь. Выделяя ключевую информацию, такую как анализ данных, участники исследования, основные выводы и ограничения, он экономит время на оценку текста более чем на 70%.

IvyPanda *IvyPanda*[28] - это инструмент для обобщения академического текста, который создает индивидуальное резюме из любой статьи, параграфа, эссе или любого другого текста; см. Рис. 2.17. Проект был запущен в 2015 году двумя энтузиастами академической науки как центр успеха, где студенты могут улучшить свои способности к обучению, подключившись к сети академических экспертов из 1230 человек по всему миру, а также используя онлайн-инструменты и ресурсы для самообучения.

GetDigest *GetDigest*[29] анализирует и обобщает веб-контент и текстовые документы, облегчая обработку информации и экономя время для повышения эффективности работы; см. Рис. 2.18. Технология веб-приложения основана на искусственном интеллекте, который способен дистиллировать информацию и обобщать ее соответствующим образом. Оно поддерживает более 33 языков мира.

SMMRY *SMMRY*[30] - это веб-приложение для обобщения статей и текстов (см. Рис. 2.19), сокращающее текст до нескольких наиболее важных предложений:

- Подсчет баллов и ранжирование предложений по важности.
- Сосредоточение резюме на теме, выбранной по ключевому слову.

[28] https://ivypanda.com/online-text-summarizer
[29] https://getdigest.com/en
[30] https://smmry.com/

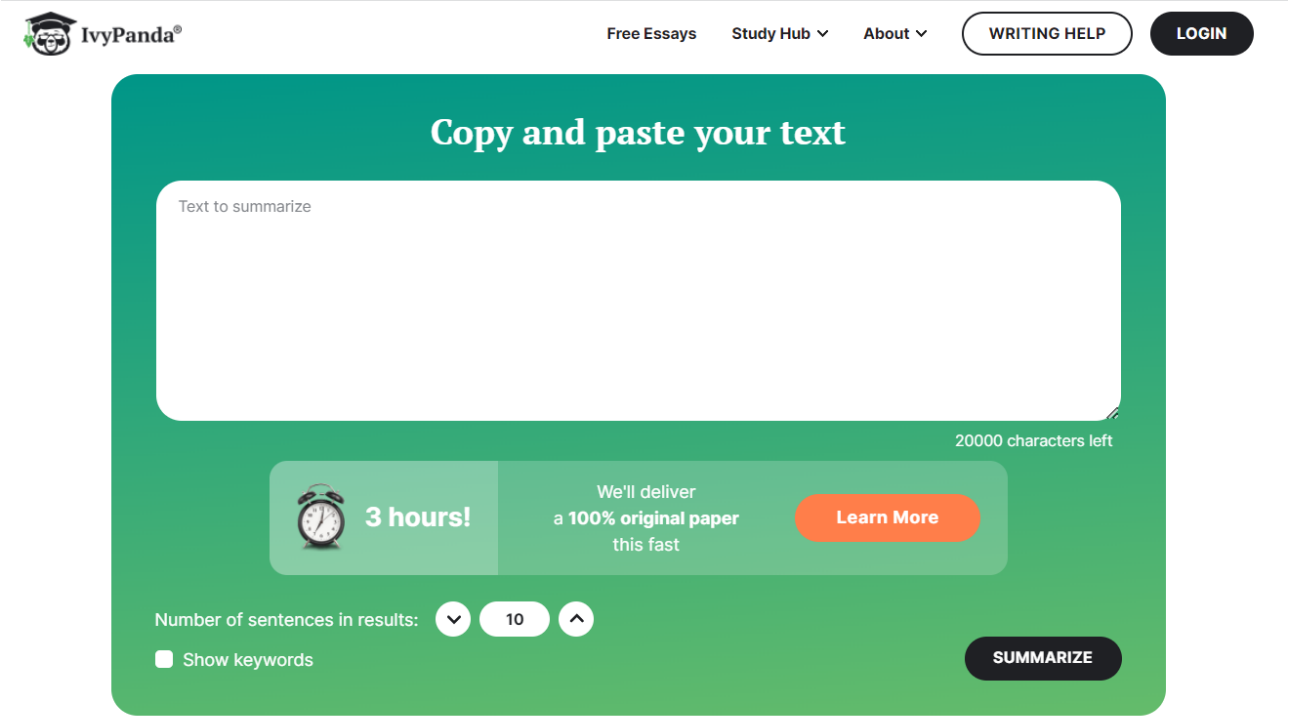


Рисунок 2.17 – Интерфейс IvyPanda.

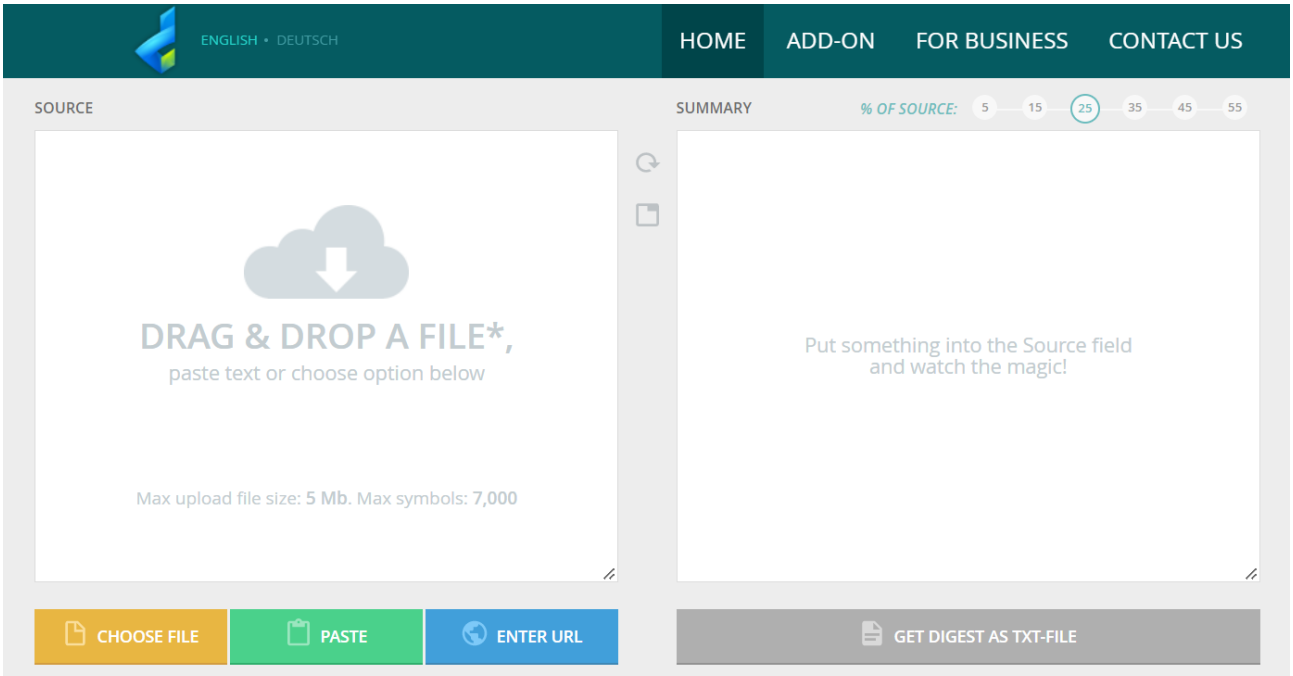


Рисунок 2.18 – Интерфейс GetDigest.

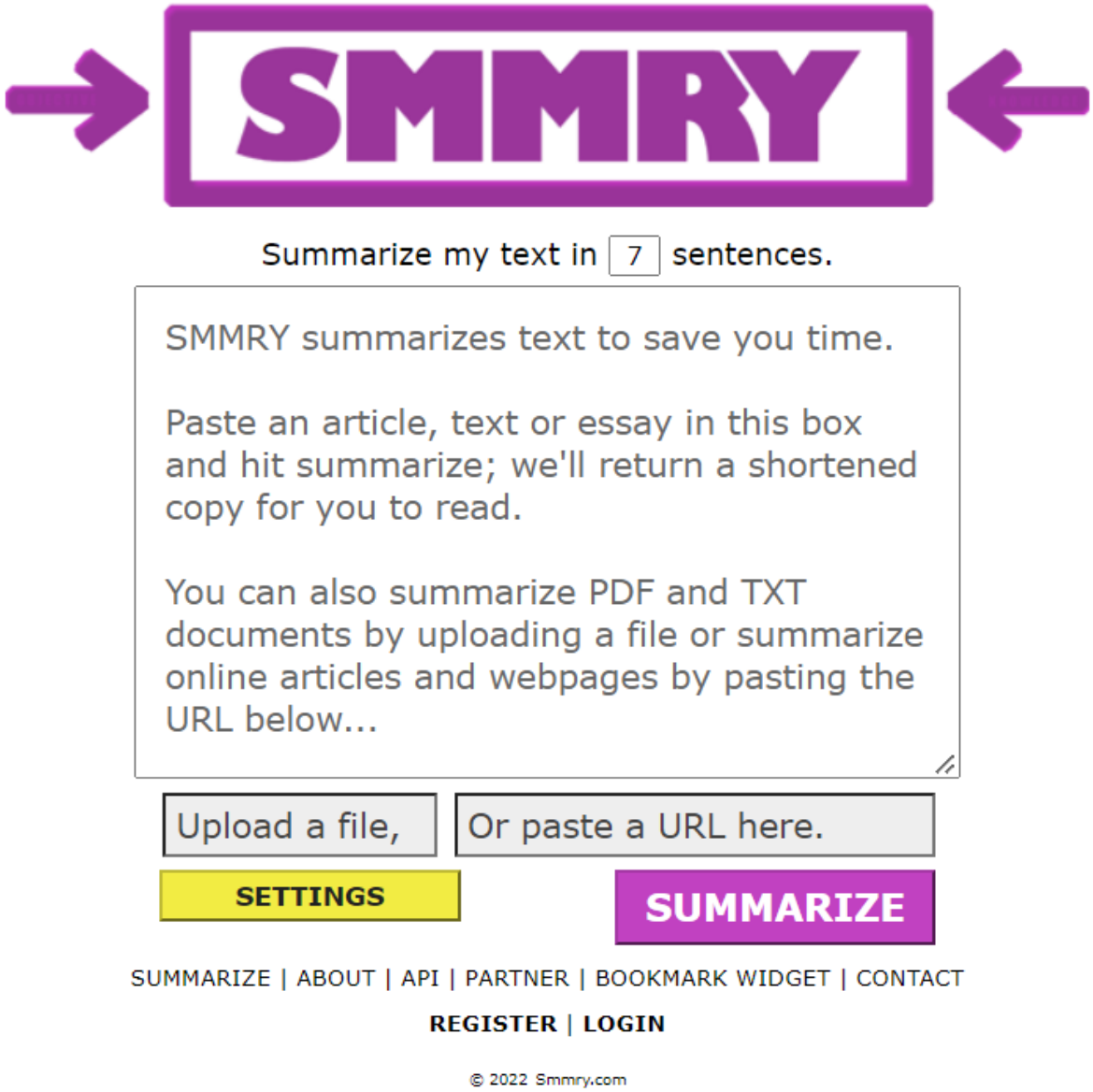


Рисунок 2.19 – См. интерфейс.

- Удаление:
- Переходные фразы.
- Ненужные пункты.
- Чрезмерные примеры.

Основной алгоритм SMMRY представлен в Рис. 2.20.

IntelliPPT *IntelliPPT*[31] компании Clourobo technologies LLP (Бангалор, Индия) резюмирует документы PDF и Microsoft Word и создает презентации Microsoft PowerPoint (PPT) из созданных резюме; см. Рис. 2.21. IntelliPPT просматривает текстовые файлы, сегментирует текст по разделам и параграфам для их индивидуального обобщения и создает PPT-файл, упорядочивая

[31] https://www.intellippt.com/

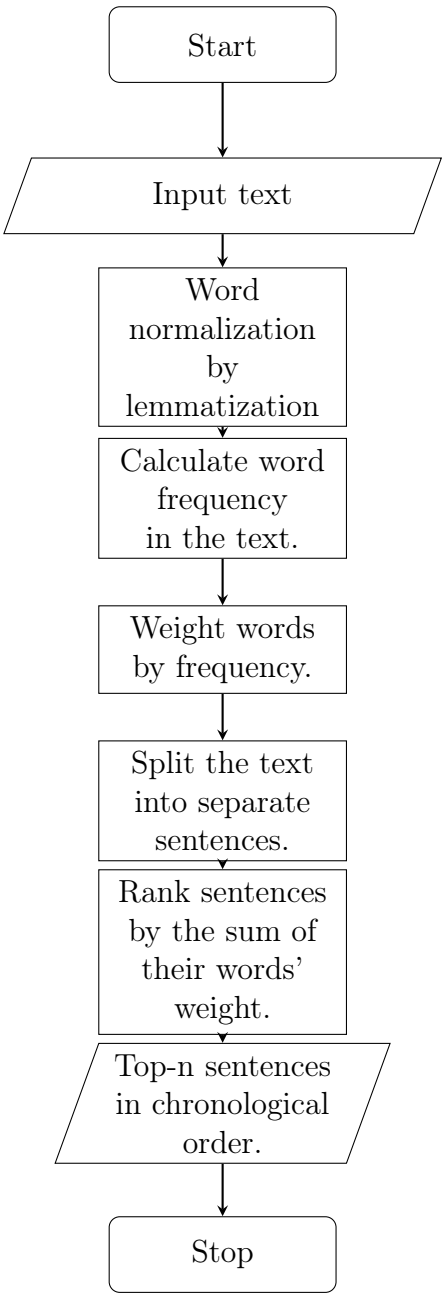


Рисунок 2.20 – SMMRY core algorithm flowchart.

слайды и сегменты резюме.

Приложение "Skimcast" Приложение[32] создает текстовые резюме из URL или PDF файла в диапазоне 1-99% от размера оригинального документа; см. Рис. 2.22. *Skimcast* также автоматически извлекает ключевые темы документа. Он может быть использован как расширение *Chrome*, и разработчик планирует создать мобильную версию.

[32] `https://www.skimcast.com/`

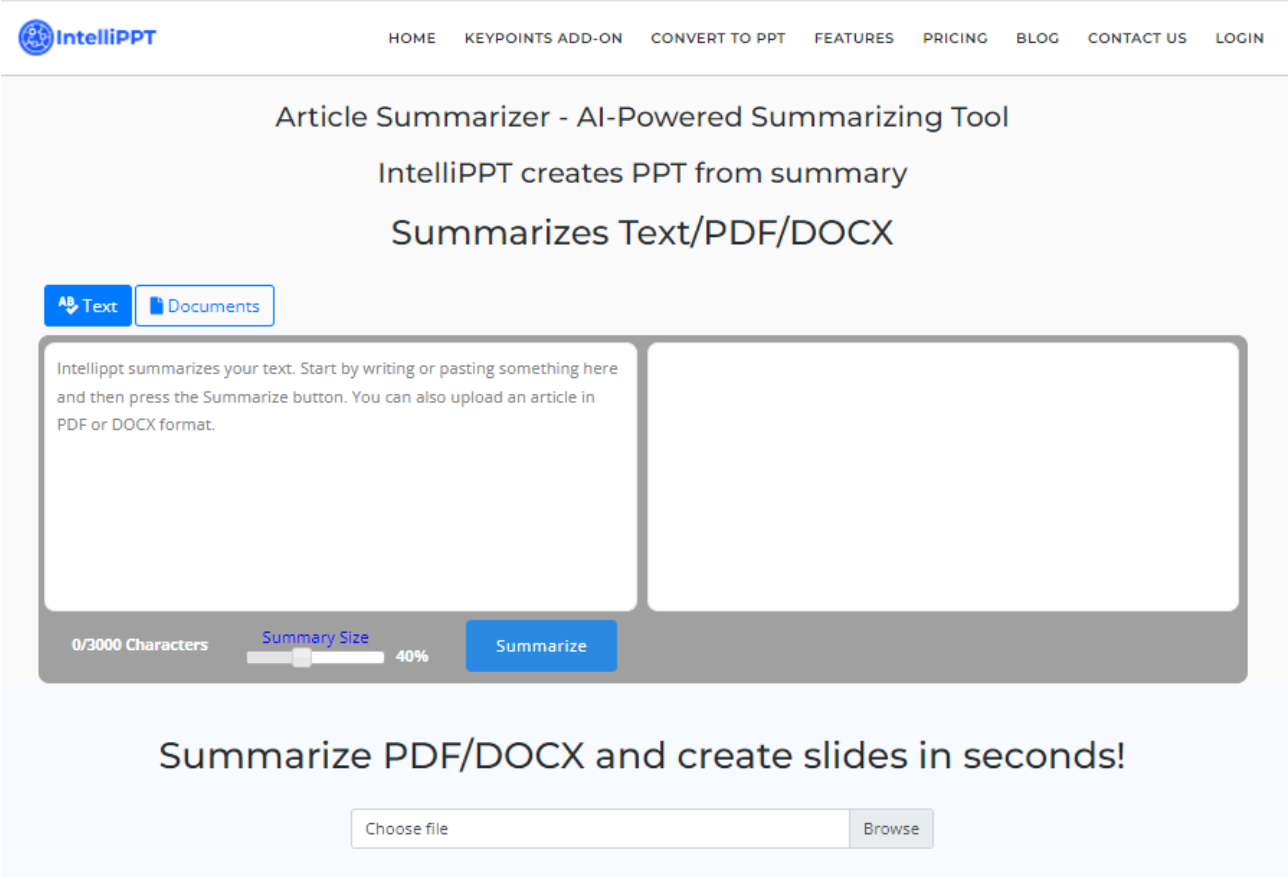


Рисунок 2.21 – Интерфейс IntelliPPT.

а) Некоммерческие научные разработки

VideoMash *VideoMash*[33] веб-приложение от [83] обобщает видео, помогая людям экономить время на просмотре только информационно насыщенных видеофрагментов. Приложение работает на сервере Django и может быть развернуто онлайн или запущено на локальной машине; см. Рис. 2.23.

Разработчик придерживался подхода суммирования текста субтитров с использованием 4 известных алгоритмов суммирования (*Luhn*, *LSA*, *TextRank*, *LexRank*) и их смеси голосования в следующих шагах:

1 Получить видеозапись для обобщения в цифровом формате.
2 Генерировать субтитры с временной меткой с использованием технологии STT, если субтитры отсутствуют.
3 Выберите желаемую степень сжатия в процентах от исходного видео.
4 Обобщите субтитры, используя один или несколько методов и моделей обобщения; см. Раздел 3.2.1 и главу 2.2.
5 Обрежьте видеозапись, используя временные метки субтитров, созданных на предыдущем шаге, чтобы в итоге получить видеорезюме.

Блок-схема алгоритма VideoMash показана в Рис. 2.24

[33] `https://github.com/aswanthkoleri/VideoMash`

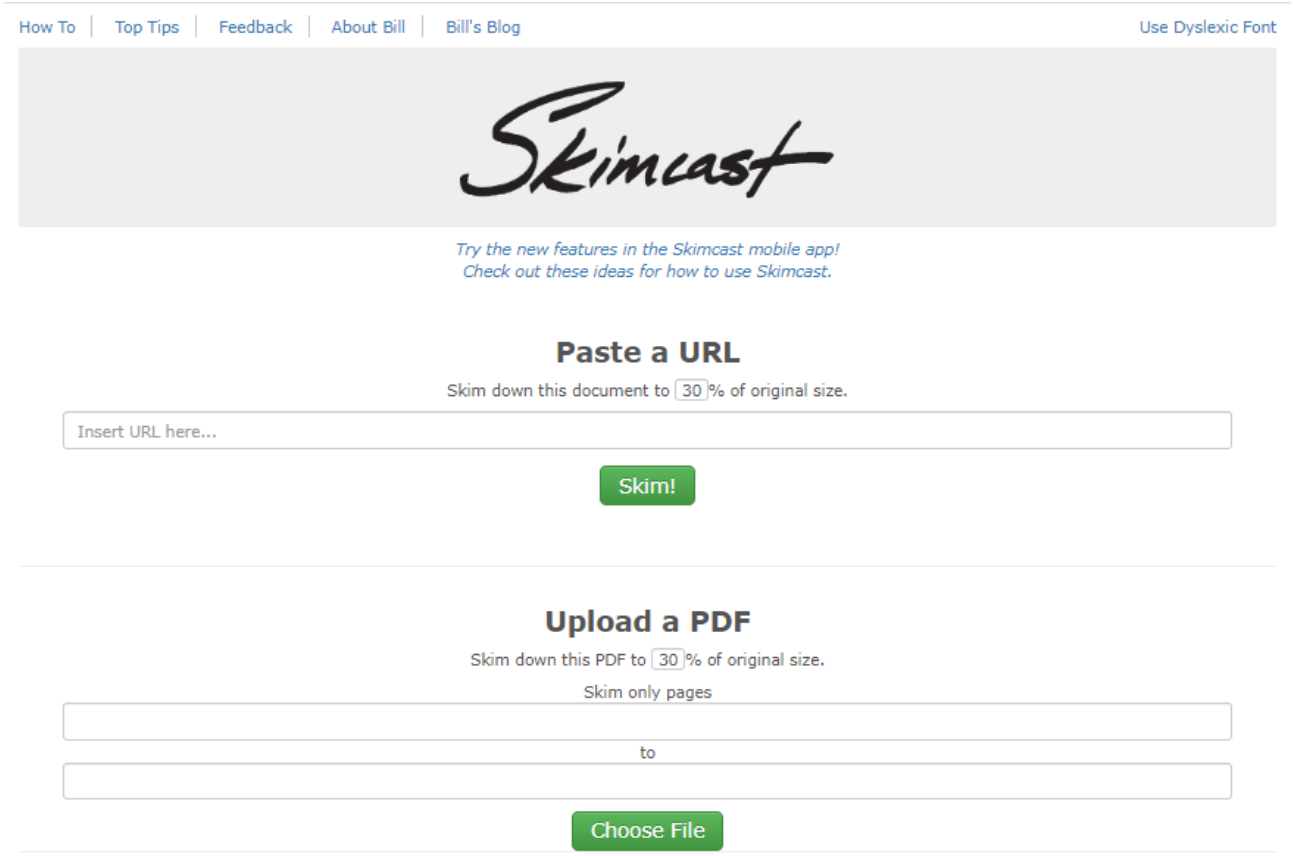


Рисунок 2.22 – Интерфейс Skimcast.

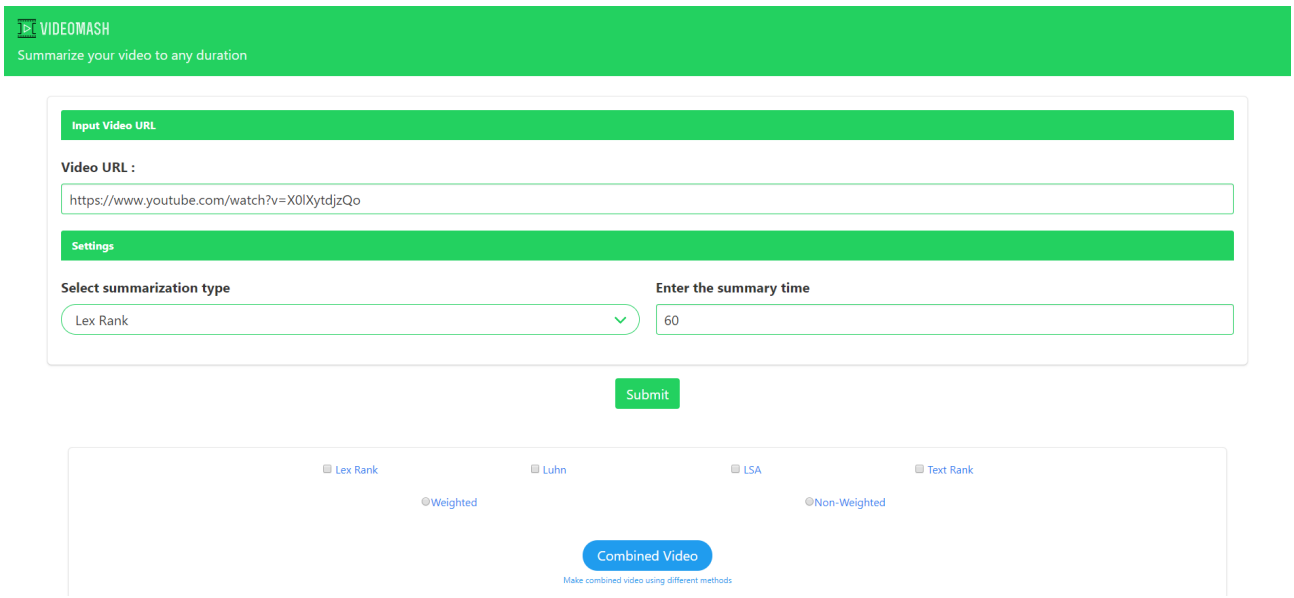


Рисунок 2.23 – VideoMash interface.

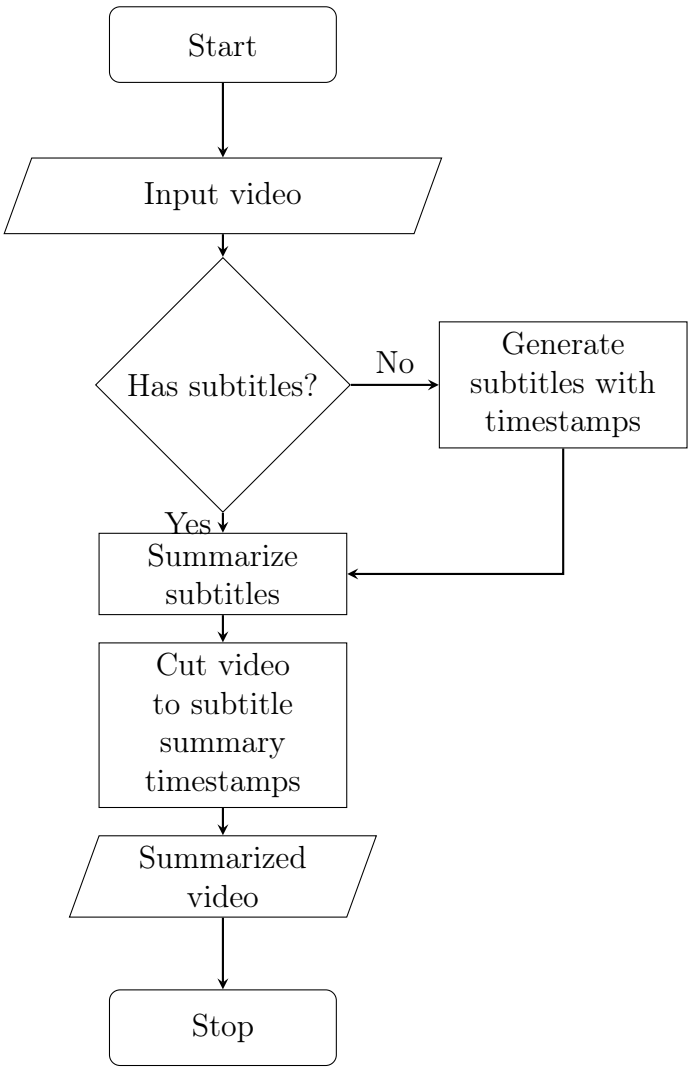


Рисунок 2.24 – VideoMash web-app algorithm flowchart.

2.5.4 Сравнение систем

Мы выбрали веб-приложения ATS, как наиболее многочисленную группу среди групп Mobile и Desktop, и сравнили их по следующим характеристикам:

- **No Character Limit**: имеет ли система ограничения по количеству символов в текстах для обобщения?

- **Многоязыковая поддержка**: есть ли в системе поддержка нескольких языков?

- **100% Plagiarism Free**: позволяет ли система подводить итоги без плагиата?

Как показано в Таблица 2.16, система Splitbrain имеет самый высокий рейтинг, поскольку не имеет ограничения по количеству символов и поддерживает несколько языков, но в ней отсутствует функция реферирования без плагиата. Поэтому мы можем использовать ее в сочетании с теми системами, которые имеют функцию без плагиата (TLDRthis, IvyPanda и IntelliPPT) или использовать дополнительный сервис перефразирования.

Таблица 2.16 – ATS Web-app comparison table.

Имя	No Char. Limit	Multi-lang. Sup.	100% Plag. Бесплатно	Rank
100% Plag. Free	Rank			
Splitbrain	1	1	0	2
TLDRthis	0	0	1	1
Scholarcy	1	0	0	1
TextSummarization	1	0	0	1
IvyPanda	0	0	1	1
GetDigest	0	1	0	1
SMMRY	1	0	0	1
IntelliPPT	0	0	1	1
QuillBot	0	0	0	0
Resoomer	0	0	0	0

3 Основная часть

В данном разделе мы поговорим о наборах данных из различных предметных областей (наука, новости, юридические документы, патенты, художественные книги) используемых для построения, тренировки и тестирования моделей автоматического реферирования текстов (см. Глава 3.1. Далее, в Глава 3.2, мы поговорим о методах используемых пр и построении моделей, и перейдем к метрикам оценки качества полученных авторефератов в Глава 3.3. Глава 3.4 посвящена экспериментам по оценке наивысшего уровня качества авторефератов достижимого при помощи Экстрактивных Методов Автоматического Реферирования Текстов. А разработанному нами методу Автореферирования текстов GreedSum мы посвятили Глава 3.5. В заключении разделы мы поговорим о практическом применении моделей автоматического реферирования текстов (см. Глава 3.6.

3.1 Данные

Количество данных, доступных для экспериментов по обобщению текстов, оставалось небольшим: несколько наборов данных, содержащих не более 1 000 статей и их резюме, до 2003 года, когда был представлен набор данных Gigaword с почти 4 миллионами пар "статья-резюме- [6].

Доступность больших наборов данных в середине 2000-х [84] открыла новые возможности для применения самых разных алгоритмов, методов и подходов, включая Deep Learning, которое стало новым брендом для алгоритмов нейронных сетей, известных с 1980-х годов.

Вышеперечисленные факторы также дали немедленный толчок к росту числа публикаций, посвященных обобщению текста, которое продолжало расти в геометрической прогрессии благодаря появлению новых больших наборов данных; см. Рис. 3.1.

Сравнивая общее количество научных публикаций с количеством публикаций, особенно по обобщению текстов, мы видим, что экспоненциальный рост обоих коррелирует, что означает, что технологический прогресс в мощностях компьютерной техники затронул узкую область NLP - обобщение текстов, охватившую все отрасли науки.

Хотя количество научных публикаций растет, наиболее распространенные данные - это News Datasets. Статистика по имеющимся категориям и данным представлена в Таблица 3.1, мы видим, что News Data делит лидерство с Scientific Papers. Уступающие документы составляют небольшую часть от общего количества и отображаются в отдельной категории под названием Other.

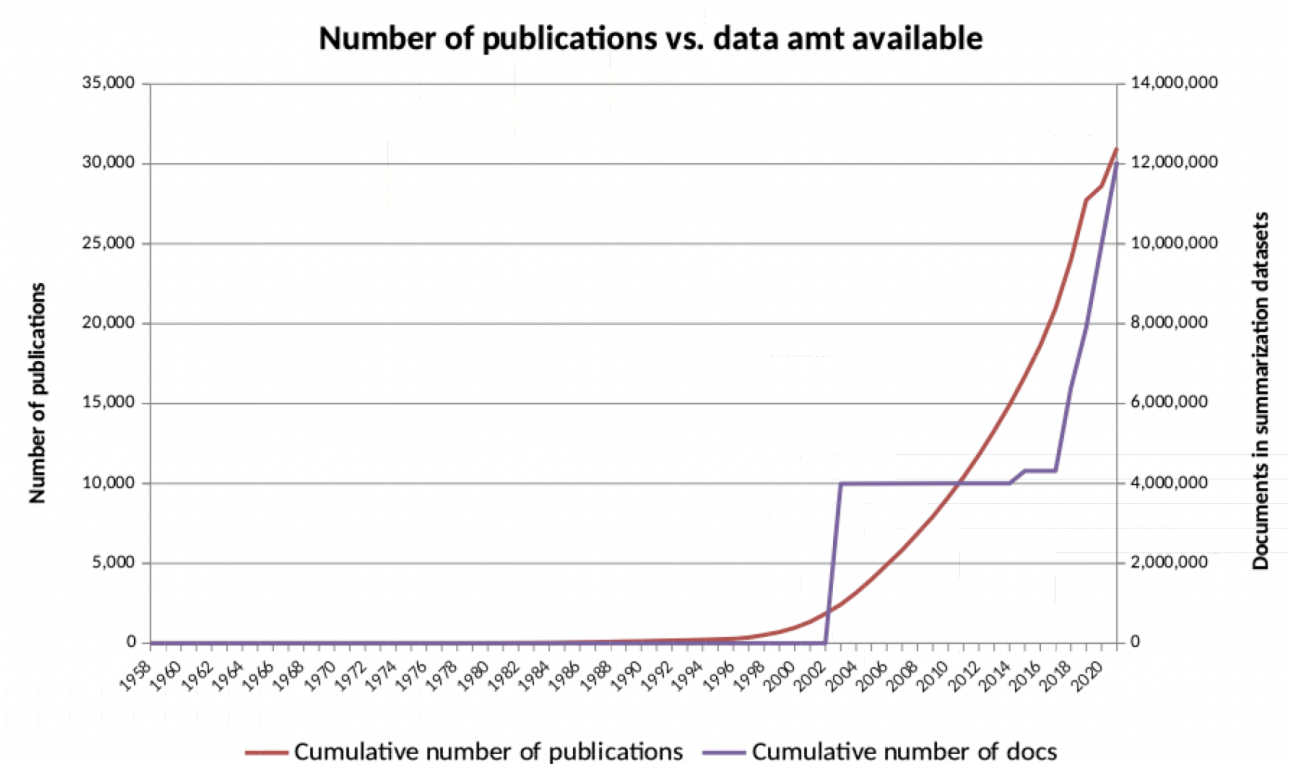


Рисунок 3.1 – Количество публикаций в сравнении с объемом имеющихся данных.

Таблица 3.1 – Суммирование наборов данных количества документов по доменным зонам.

Тема набора данных	количество документов
Электронные письма	18,000
Инструкции	200,000
Законодательство	23,000
Новости	5,897,122
Наука	1,648,250
Короткий рассказ	120,000
Итого	7,906,372

3.1.1 Научные наборы данных

Раздел научных наборов данных состоит из научных публикаций и статей. Структура датасета содержит имя автора, статью и аннотацию, что позволяет использовать данные в задаче Summarizing the text.

К научной категории можно отнести следующие наборы данных: arXiv, PubMed, BigPatent. Свойства наборов данных научного обобщения показаны в Таблица 3.2.

Таблица 3.2 – Свойства научного обобщения данных.

Набор данных	Количество документов	Среднее количество слов/статья	Среднее количес
arXiv	215,913	4,938.0	
PubMed	133,215	3,016.0	
BigPatent	1,341,362	116.5	

a) arXiv

Почти 30 лет arXiv[1] служит научному сообществу, предоставляя доступ к научным статьям из различных отраслей математики, физики, поддисциплин информатики и всего, что между ними и вокруг, включая статистику, электро- и машиностроение, биоинформатику и экономику.

Набор данных arXive, впервые представленный в 2018 году [61], содержит 215 тысяч `arXiv.org` репозитарных научных статей на английском языке из области физики, астрофизики, математики и других количественных областей. Статьи в наборе данных содержат аннотации, тексты статей, список разделов и тексты статей, разделенные по разделам; см. Глава 3.1, Раздел a).

b) PubMed

В 1996 году PubMed сделал возможным доступ к более чем 28 миллионам ссылок на биомедицинские и жизненные статьи из базы данных MEDLINE. Широкий свободный доступ к системе PubMed появился в июне 1997 года. PubMed содержит 119 тысяч статей из области медицины. Статьи в наборе данных содержат аннотации, тексты статей, список разделов и тексты статей, разделенные по разделам.

Мы обработали данные, чтобы очистить статьи с ошибочными аннотациями длиннее текста, слишком длинные и слишком короткие тексты статей и оставить только статьи с длиной аннотации от 10 до 20 предложений, чтобы создать набор данных из 17 038 статей. Описание полученного набора данных приведено в Таблица 3.3.

c) BigPatent

BigPatent - это набор данных, включающий 1,3 миллиона патентных документов США. Он содержит патенты, поданные после 1971 года в девяти различных технологических областях. Для решения задачи обобщения реферат патента рассматривается как золотое резюме, а его описание - как исходный текст [85].

BigPatent обладает следующими свойствами по сравнению с другими наборами данных для обобщения:

[1]arXiv - бесплатный архив с открытым доступом для 1 975 103 научных статей в области физики, информатики, математики, статистики, электротехники, количественной биологии, количественных финансов, системных наук и экономики (`https://arxiv.org/`).

Таблица 3.3 – Описание очищенных наборов данных.

	arXive		PubMed	
	Text length	Abstract length	Text length	Abstract length
count		17 038		
mean	263.44	11.75	88.89	6.85
std	102.57	2.13	60.56	2.88
min	100.00	10.00	3.00	1.00
25%	179.00	10.00	47.00	5.00
50%	252.00	11.00	77.00	7.00
75%	338.00	13.00	113.00	9.00
max	500.00	20.00	1887.00	23.00

- Резюме содержат более плотную структуру дискурса с большим количеством повторяющихся сущностей.
- Салиентная информация равномерно распределена во входном тексте.
- Более короткие фрагменты извлечений находятся в резюме.

3.1.2 Новостные наборы данных

Новости - самая большая категория по содержанию документов, как описано в Таблица 3.1.

Категория состоит из следующих наборов данных: CNN, Daily Mail, Gigaword, X-Sum, Newsroom, а также наборы данных с конференций DUC и TAC. Свойства наборов данных для обобщения новостей показаны в Таблица 3.4.

Таблица 3.4 – Свойства наборов данных для обобщения новостей.

Набор данных	Кол-во документов	Ср. кол-во слов/статья	Ср. кол-во слов/сумма.
CNN/Daily Mail	312,084	781.0	56.0
BBC News	2,225	-	-
Gigaword	3,990,951	31.4	8.3
X-Sum	226,711	431.0	23.0
Cornell Newsroom	1,321,995	658.6	26.7
NY Times Corpus	650,000	530.0	38.0
DUC-2001	309	100.0	-
DUC-2002	567	100.0	-
DUC-2004	500	-	-
TAC-2014	220	-	235.6

a) Конференция по пониманию документов (DUC)

DUC проводилась ежегодно с 2001 по 2008 год и была важным форумом для сравнения систем обобщения на общем тестовом наборе[2]. Наборы данных DUC 2001-2004 годов больше связаны с многодокументным обобщением, а наборы данных DUC 2005-2007 годов также связаны с этой темой, но ориентированы на запросы.

Наборы данных DUC - это новостные данные, содержащиеся в трех наборах данных, связанных с годом проведения конференции, разделенных на различные тематические кластеры, где каждый кластер содержит 2-4 резюме, составленных профессиональными экспертами.

b) Конференция по анализу текста (TAC)

Продолжением конференции DUC стал конкурс TAC, созданный для изучения области обработки естественного языка. Каждый из участников получил набор тестовых данных и оценку результатов [3].

TAC 2010 - это популярный набор данных суммирования, собранный из 440 документов. Набор данных можно разделить на пять основных категорий: Аварии и стихийные бедствия, Террористические атаки, Расследования и судебные разбирательства, Здоровье и безопасность, Исчезающие ресурсы.

c) Gigaword

Набор данных для обобщения Gigaword был представлен Граффом и др. в 2003 году [6] и состоял из 8,6 млн. коротких новостных статей для задачи генерации заголовков или составления резюме из одного предложения. Фактический набор данных Gigaword был представлен в работе [86]. Набор данных состоит из 3,8 млн обучающих, 189 тыс. разработок и 1951 тестового документа.

d) CNN и Daily Mail

Для оценки сводки использовался набор данных CNN / Daily Mail [47]. Она называется "анонимизированной поскольку в ней используются теги вместо названного объекта.

Сгенерированные человеком абстрактные резюме были составлены из новостных материалов на сайтах CNN и Daily Mail в качестве вопросов и историй в качестве соответствующих отрывков.

CNN: Набор данных абстрактного обобщения состоит из 92 000 документов, сгенерированных на сайте CNN. Впервые использовалась в 2016 году для абстрактного обобщения текста [47] .

Daily mail: набор данных для абстрактного обобщения состоит из 219 000 документов, полученных с сайта Daily Mail. Впервые использовалась в

[2]Онлайновые материалы конференций доступны на сайте `https://duc.nist.gov/data.html`
[3]Онлайновые материалы конференций доступны на сайте `http://tac.nist.gov/2010/`

2016 году для абстрактного обобщения текста [47].

e) Экстремальное обобщение (X-Sum)

X-Sum [87] - это набор данных, который не подходит для экстрактивного обобщения и поощряет подход абстрактного обобщения. Задача набора данных состоит в том, чтобы создать краткое, в одно предложение, резюме новости для предоставленной в качестве входных данных новостной заметки. Данные были собраны путем соскабливания страниц статей с сайта BBC. Набор данных содержит 204K обучающих, 11K проверочных и 11K тестовых наборов образцов. В среднем длина статьи составляет 431 слово (20 предложений), а длина резюме - 23 слова.

f) Cornell Newsroom

Cornell Newsroom [88] - это массивный набор данных для обучения и оценки систем обобщения. Набор данных состоит из 1,3 млн статей и резюме, составленных авторами и редакторами из 38 значимых новостных источников. Резюме собраны из поисковых и социальных метаданных в период с 1998 по 2017 год и используют различные стратегии обобщения, сочетающие извлечение и абстрагирование.

g) NY Times Corpus

Аннотированный корпус New York Times включает более 1,8 млн. статей, опубликованных в период с 01.01.1987 по 19.06.2007, дополненных метаданными статьи [89].

Набор данных состоит из 650 тысяч пар "статья-резюме большинство резюме статей были созданы вручную исследователями библиотек. Кроме того, более 1,5 млн. документов имеют хотя бы один тег, такой как темы, места, лица, организации и названия.

h) BBC News

Набор данных BBC News, включающий 2225 классифицированных статей, был получен из BBC News в 2004 году. и 2005 помечено как бизнес, развлечения, политика, спорт и технологии[4]. Набор данных находится в свободном доступе только для некоммерческих и исследовательских целей, и все данные предоставляются в предварительно обработанном формате [90].

3.1.3 Книги

Задача обобщения текстов книг не менее актуальна; для ее решения существует еще одна категория наборов данных - книги. Статистика по популярным наборам данных книг приведена в Таблица 3.5.

[4]Все права, включая авторские, на содержание оригинальных статей принадлежат BBC. `http://mlg.ucd.ie/datasets/bbc.html`

Таблица 3.5 – Свойства наборов данных книг.

Набор данных	Кол-во документов	Ср. размер документов (КиБ)	Размер (ГиБ)
Bookcorpus	11,038	419.37	4.63
BookCorpusOpen	17,868	369.87	6.30
Books3	197,000	538,36	100,96

a) Bookcorpus

Составители датасета собрали два датасета, один из которых состоит из фильмов и аннотаций, а второй - это датасет BookCorpus [91].

Набор данных BookCorpus состоит из 11K книг, взятых с сайта с электронными книгами. Важно отметить, что этот датасет не относится к авторскому праву, так как оно встречается только в бесплатных книгах неопубликованных авторов. Также, чтобы сохранить чистоту эксперимента, исследователи оставили в датасете только книги с более чем 20 тысячами слов в 16 различных жанрах.

b) Books1 или BookCorpusOpen

BookCorpusOpen - это расширенная версия BookCorpus [91]. Однако в связи с проблемой доступности набора данных BookCorpus и возможностью собрать более обширную версию, энтузиастами была собрана вторая версия BookCorpus. Эта версия содержит 17.9K книг, содержащих два поля: название и необработанный текст книги. Структура и доступный объем данных в корпусе аналогичны корпусу Books1, использованному при разработке GPT-3 [76]от OpenAI[5].

c) Books3 или Bibliotik

Books3 - это корпус книг, взятый из выборки Bibliotik. Этот набор данных является работой Шона Прессера и входит в набор данных The Pile [92, 93].

Bibliotik содержит художественную и нехудожественную литературу и является более обширным, чем BookCorpusOpen. Он содержит все документы в формате обычного текста, около 197 000 книг, которые были обработаны аналогично BookCorpus. Структура и доступный объем данных в корпусе аналогичны Books2.

3.1.4 Другие наборы данных

Еще одна отдельная категория состоит из наборов данных, которые содержат внутри себя неклассифицированную информацию - содержимое чеков, набор текстов из Вики. Статистика для наборов данных описана в Табли-

[5]OpenAI - компания по исследованию и разработке искусственного интеллекта (ИИ), поддерживаемая Элоном Маском. Миссия организации заключается в том, чтобы ИИ в целом приносил пользу всему человечеству

ца 3.6.

Таблица 3.6 – Другие свойства наборов данных для обобщения

.

Набор данных	Кол-во документов	Ср. кол-во слов/статья	Ср. кол-во слов/сумма.
Билсум	22,218	1,533.0	500.0
WikiHow	230,843	579.8	62.1
WikiLingua	42,783.	391.0	39.0

a) Billsum

Набор данных BillSum состоит из учебных и тестовых счетов США. Счета были собраны из сервиса Govinfo Правительственного издательства США (GPO) [94].

Всего в наборе данных 22,3K законопроектов сессий Конгресса США, которые были собраны за период с 1993 по 2018 год. Калифорнийский советник по законодательству участвует в подготовке резюме к законопроектам с 2015-2016 годов.

b) WikiLingua

WikiLingua - это крупномасштабный многоязычный набор данных для оценки моделей межъязыкового абстрактного обобщения. Он состоит из текстов на 18 языках, извлеченных из WikiHow (пары из статьи и ее резюме). Написанные человеком тексты WikiHow представляют собой высококачественные текстовые данные различной тематики. Золотым стандартом принципа обобщения является сопоставление текстов на разных языках по признаку встречаемости одних и тех же образов. Вкратце, набор данных состоит из 141.5K уникальных статей на английском языке. Каждый из остальных 17 языков имеет в среднем 42,8K статей, которые совпадают со статьей на английском языке.

c) WikiHow

WikiHow - это набор данных, состоящий из более чем 200 тысяч пар длинных последовательностей. Каждый из документов собран из одноименной онлайн-базы знаний путем объединения абзацев в статье и выявления обобщающих предложений [95].

В следующей главе мы поговорим о методах, используемых при построении моделей автоматического реферирования текстов.

3.2 Методы

3.2.1 Методы автореферирования

Чтобы с самого начала представить большую картину ландшафта методов, мы приводим классификацию методов обобщения текста; см. Рис. 3.2.

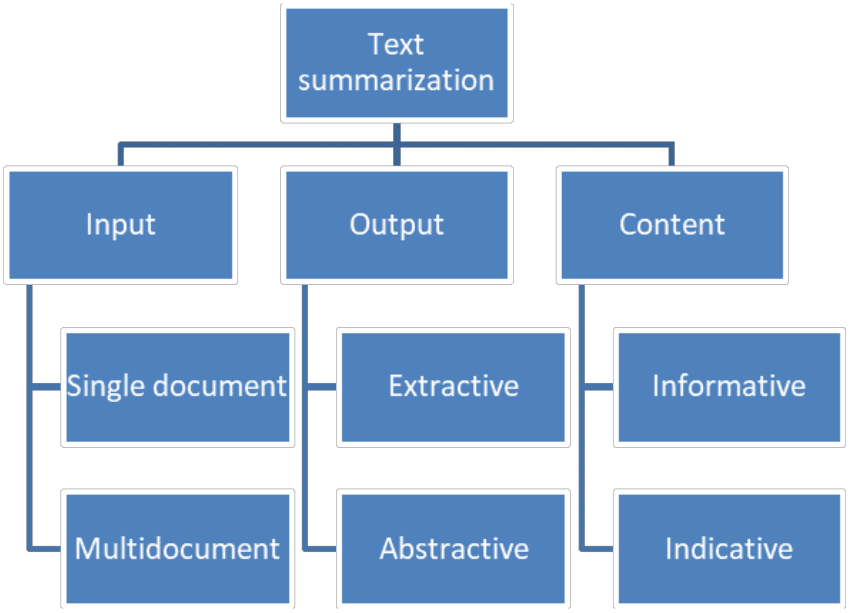


Рисунок 3.2 – Классификация методов обобщения текста [96, 97].

.

Во-первых, нам необходимо классифицировать подходы, используемые при резюмировании текста:

1 **Экстрактивное автореферирование** Методы экстрактивного суммирования выбирают информативные предложения из исходного документа на основе заданных критериев для построения резюме.
Основной проблемой экстрактивного суммирования является выбор значимых предложений из входного документа для включения в резюме. Таким образом, используется метод оценки предложений, основанный на их характеристиках [98]. Сначала предложения оцениваются по признакам, а затем ранжируются соответствующим образом. Таким образом, предложения, получившие наибольшее количество баллов, включаются в итоговое резюме.

2 **Абстрактивное автореферирование** Методы абстрактного суммирования генерируют резюме путем построения новых коротких предложений, подобно человеку. Резюме может содержать фразы, которых нет в оригинальном тексте. Для создания языка абстрактных резюме необходимы методы генерации и сжатия. Абстрактное обобщение текста можно разделить на подход, основанный на структуре, и подход, основанный на семантике.

3 **Информативное автореферирование** Информативное резюме представляет исходный документ в полном объеме. Поэтому оно содержит всю важную информацию, необходимую для передачи основного смысла исходного текста, и опускает вспомогательную информацию.

4 **Индикативное автореферирование** Основная цель ориентировочного резюме - порекомендовать содержание статьи, не давая подробной информации о ней. Оно может служить в качестве тизера, чтобы побудить пользователя получить полный текст. Аннотации к книгам, фрагменты результатов веб-поиска и трейлеры к фильмам являются примерами ориентировочных резюме.

5 **Однодокументное автореферирование** Резюмирующие устройства для одного документа нацелены на резюмирование одного единственного документа.

6 **Многодокументное автореферирование** Многодокументные обобщающие системы в качестве источника используют коллекцию документов, связанных общей темой или событием, и производят обобщение по нескольким документам во временном порядке. Это может быть использовано в процессе обзора литературы в научной работе или при составлении тематического отчета для получения краткой и сжатой информации по теме, сокращая избыточность [99]. Системы могут быть простыми, например, выбрать наиболее важный документ и использовать его для однодокументного обобщения [100]. В качестве альтернативы можно обобщить все документы по отдельности, объединить их, а затем обобщить объединенную сумму.

3.2.2 Методы оценки верхней границы качества автореферата

a) Variable Neighborhood Search (VNS)

VNS - это основа для построения эвристики (метаэвристики), которая использует идею систематического изменения окрестности начального решения для поиска оптимума объективной функции [101].

VNS систематически использует следующие факты наблюдений [101]:

1 Локальный минимум для одной структуры окрестности не обязательно является таковым для другой.

2 Глобальный минимум - это локальный минимум для всех возможных структур окрестностей.

3 Для многих задач локальные минимумы в одной или нескольких окрестностях находятся относительно близко друг к другу.

Initialization. Select the set of neighborhood structures $\mathcal{N}_k$, for $k = 1, \ldots, k_{\max}$, that will be used in the search; find an initial solution x; choose a stopping condition;
Repeat the following sequence until the stopping condition is met:
(1) Set $k \leftarrow 1$;
(2) *Repeat* the following steps until $k = k_{\max}$:
(a) Shaking. Generate a point x' at random from the kth neighborhood of x ($x' \in \mathcal{N}_k(x)$);
(b) Move or not. If this point is better than the incumbent, move there ($x \leftarrow x'$), and continue the search with $\mathcal{N}_1$ ($k \leftarrow 1$); otherwise, set $k \leftarrow k + 1$;

Рисунок 3.3 – Общий псевдокод алгоритма VNS.

b) Жадный алгоритм

Жадный алгоритм - это любой алгоритм, который следует эвристике решения проблемы, заключающейся в принятии наилучшего локального решения для задачи оптимизации [102]. Для некоторых задач жадная эвристика может дать локально оптимальное решение, приближенное к глобально оптимальному решению, за разумное время.

c) Генетический алгоритм

Генетический алгоритм - это метаэвристический метод, вдохновленный естественным процессом отбора, принадлежащий к большому классу эволюционных алгоритмов. Генетические алгоритмы широко используются для генерации решений задач оптимизации и поиска с помощью таких операторов, как кроссовер, мутация и отбор, которые встречаются в адаптационных и эволюционных процессах воспроизводства живых видов [103].

3.2.3 Методы используемые в разработанном алгоритме автореферирования

a) Векторизация частотно-инверсной частоты документов (TFIDF) и представление текстов в виде мешка слов

TFIDF - это числовая статистика, отражающая важность слова для документа в коллекции или корпусе данных [104]. Значение TFIDF прямо пропорционально количеству раз, когда слово встречается в документе [15] и корректируется количеством документов в коллекции или наборе данных, которые содержат это слово (обратная компонента частоты документов (IDF)). Это помогает учесть тот факт, что слова общего лексикона имеют высокую частоту почти в любом документе, поэтому мы можем отделить их от действительно важных слов, специфичных для документа.

Частота термина tf или частота термина t вычисляется в (3.1).

$$\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{\acute{t}\in d} f_{\acute{t},d}}, \tag{3.1}$$

где $f_{t,d}$ - необработанный подсчет термина t в документе d, а $\acute{t}$ - все остальные термины в документе.

Обратная частота документа (IDF) измеряет, насколько информативно слово и насколько часто или редко оно встречается среди других документов. Это логарифмически увеличенная обратная доля документов, содержащих слово, указанное в (3.2).

$$\mathrm{IDF}(t,D) = \log \frac{N}{\mid \{d \in D : t \in d\} \mid +1}, \tag{3.2}$$

где N - общее количество документов в корпусе $N =\mid D \mid$, а $\mid \{d \in D : t \in d\} \mid$ - это количество документов, содержащих термин t, и 1 добавляется к знаменателю, чтобы избежать деления на ноль, если термин отсутствует в корпусе.

TFIDF вычисляется с помощью перемножения результатов (3.1) и (3.2); см. (3.3).

$$\mathrm{TFIDF}(t,d,D) = \mathrm{tf}(t,d) \times \mathrm{IDF}(t,D) \tag{3.3}$$

b) Определение размера выборки

Цель метода - выбрать количество наблюдений, которые будут использоваться в выборке данных. Существует высокая вероятность того, что полученная выборка справедливо представляет данные популяции, и выводы, сделанные по выборке, могут быть обобщены на всю популяцию. Расчет размера выборки (SS) приведен в (3.4) [105].

$$\mathrm{SS} = \frac{Z^2 * (p) * (1-p)}{c^2}, \tag{3.4}$$

где Z - Z-score (например, 1,96 для 95% доверительной вероятности или точности), p - вероятность, с которой может быть выбран любой элемент из популяции (по умолчанию 0,5), c - доверительный интервал в десятичной форме.

Таким образом, мы рассчитываем размер выборки и проводим Случайную выборку с набором данных. Наконец, чтобы убедиться в репрезентативности нашей выборки, мы сравниваем распределения признаков в наборе данных и в полученной выборке.

Оценка качества выборки Методы проверки того, справедливо ли образец представляет свой источник:

1 **Коэффициент корреляции Пирсона**: показывает, как два ряда чисел соотносятся друг с другом. Следовательно, коррелируют или соотносятся

друг с другом, но это ничего не говорит нам о причинно-следственной связи. Коэффициент корреляции приведен в (3.5) [106].

$$\text{Correl}(X, Y) = \frac{\sum(x - \bar{x})(y - \bar{y})}{\sqrt{\sum(x - \bar{x})^2 \sum(y - \bar{y})}}, \tag{3.5}$$

где $\bar{x}$ и $\bar{y}$ - средние значения x и y. Чем выше коэффициент корреляции, тем сильнее связь между двумя рядами чисел. Как правило, сильной корреляцией считается все, что выше 0,70. Мы используем коэффициент корреляции, чтобы определить, коррелируют или нет такие статистические свойства, как среднее значение, стандартное отклонение, минимумы и максимумы, а также квартили выборки и популяции.

2 **Kolmogorov-Smirnov test (KS test)**: двухвыборочный KS тест показывает, следуют ли две выборки одинаковому распределению или нет, и он приведен в (3.6).

$$D_{n,m} = \sup_x \mid F_{1,n}(x) - F_{2,m}(x) \mid, \tag{3.6}$$

где $F_{1,n}$ и $F_{2,m}$ - эмпирические функции распределения, n и m - размеры первой и второй выборки соответственно, а sup - функция $supremum$ (коротко, число из множества, которое больше или равно любому числу из подмножества множества, если такое число существует [107]). Реализация теста Колмогорова-Смирнова в `scipy` [108] Модуль Python возвращает `statistics` и `p-value`, и если `p-value` больше 0.05, то мы не можем отвергнуть нулевую гипотезу (H_0) и должны предположить, что две протестированные выборки имеют одинаковое распределение вероятности.

В следующей главе мы поговорим о методах оценки качества авторефератов полученных при использовании различных моделей, для унифицированного подхода и объективной оценки результатов работы моделей.

3.3 Метрики оценки качества авторефератов

Балл оценки качества автореферата является критическим фактором, влияющим на успех задач реферирования. В настоящее время большинство существующих методов представляют собой меру сходства сгенерированного резюме с золотым стандартом, написанным людьми. В этой главе метрики расположены в порядке возрастания года публикации.

3.3.1 Экспертная оценка

Проблема оценки резюме текста не является тривиальной задачей, поскольку многомерность семантического пространства, в котором рассчитываются основные характеристики генерируемого текста, потенциально может иметь бесконечное количество оценок и их интерпретаций [109–111]. Тем не менее, на наш взгляд, можно выделить следующие ключевые моменты:

1 Ручная маркировка может быть, с одной стороны, избыточной, а с другой - недостаточной; поэтому ее повторное использование в смежных задачах требует дополнительного ручного труда.

2 Существуют различные обстоятельства ручной маркировки, а именно субъективность и конфликты между аннотаторами. В среднем, доля согласия между аннотаторами может достигать 70 процентов. Такая особенность затрудняет разработку на больших массивов данных [112].

3 Одной из причин использования ручного труда при оценке абстрактного обобщения является наличие единого золотого стандарта для целевой переменной в обучающем наборе данных. Это противоречит самой природе множественного представления смысла в естественных языках. Таким образом, метрики и оценки, основанные на совпадении слов, плохо подходят для абстрактного обобщения.

4 Основными характеристиками, измеряемыми с помощью автоматических метрик, являются точность и отзыв с точки зрения сходства с золотым стандартом; другими словами, можно назвать охват темы и избыточность текста. В академическом сообществе и в бизнесе существует интерес к множеству различных текстовых метрик, таких как читабельность, связность, информативность, краткость и другие. Следует отметить, что существует научный пробел в вычислительной лингвистике и обработке естественного языка, связанный в основном с психолингвистической природой восприятия кратких аннотаций. В зависимости от задачи, существуют также такие оценки, как артистизм, увлеченность, объективность и другие.

3.3.2 BiLingual Evaluation Understudy (BLEU)

Метрика BLEU предназначена для автоматизированной оценки машинного перевода, и ее поведение хорошо коррелирует с человеческой оценкой [113]. Поэтому она широко используется в системах машинного перевода. Более то-

го, она была адаптирована к проблеме оценки качества резюме текста [114].

Основная идея BLEU заключается в измерении степени близости между сгенерированным переводом и набором золотых стандартов. Близость рассчитывается на основе средневзвешенного значения совпадений n-грамм переменной длины между сгенерированным и целевым человеческим переводом.

Численные эксперименты показали, что средневзвешенное значение, то есть BLEU, высоко коррелирует с оценками, сделанными людьми. Аналогичным образом, авторы [114] использовали BLEU для оценки результатов автоматического реферирования текстов, руководствуясь соображением, что чем ближе сгенерированное резюме к золотому стандарту по количеству n-грамм, тем лучше работает генеративная модель языка.

Идея метрики очень близка к ROUGE, которая также оценивает близость между текстами с помощью n-грамм, разница между метриками заключается в нормализующем факторе. Позже обе метрики были объединены в одну метрику с помощью среднего геометрического.

BLUE - это метрика, основанная на точности, а для эмуляции recall вводится штраф за краткость (BP) для компенсации возможности слишком коротких переводов с высоким показателем точности.

Формула расчета BP приведена в (3.7):

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-\frac{r}{c})} & \text{if } c \leqslant r \end{cases}, \tag{3.7}$$

где c и r означают длину гипотезы и опорных переводов.

В результате расчет BLEU баллов выглядит следующим образом (3.8):

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \tag{3.8}$$

где n обозначает порядки n-грамм, рассматриваемых для p_n, а w_n обозначает веса, назначенные для точности n-грамм. Более подробно расчет p_n описан ниже в (3.9):

$$p_n = \frac{\sum_{C \epsilon \{\text{Candidates}\}} \sum_{n-gram \epsilon C} count_{clip}(n-gram)}{\sum_{C' \epsilon \{\text{Candidates}\}} \sum_{n-gram' \epsilon C'} count(n-gram')}, \tag{3.9}$$

Одним из наиболее важных ограничений метрики BLEU является то, что она основана на предположении, что она должна в среднем соответствовать человеческим оценкам на обширном тестовом корпусе, поскольку оценки отдельных предложений часто отличаются от человеческих оценок.

3.3.3 Recall-Oriented Understudy for Gisting Evaluation (ROUGE)

Одной из наиболее эффективных метрик оценки резюме является ROUGE [115]. Эта метрика была впервые предложена на конференции DUC в 2004 году. Основная идея оценки ROUGE основана на подсчете количества совпадений n-грамм между кандидатом и эталонным резюме. ROUGE является одной из самых популярных метрик и даже считается общепринятым стандартом в задачах обобщения.

В научной литературе существуют варианты этой метрики. Наиболее распространенные из них - ROUGE-N, ROUGE-L, ROUGE-W и ROUGE-S - являются частью общедоступного пакета обработки естественного языка NLTK для языка программирования Python [116].

Формально, ROUGE-N - это отзыв n-грамм между резюме кандидата и резюме ссылки, и он вычисляется следующим образом в (3.10):

$$ROUGE - N = \frac{\sum_{S \epsilon \{RS\}} \sum_{gram_n \epsilon S} count_{match}\left(gram_n\right)}{\sum_{S \epsilon \{RS\}} \sum_{gram_n \epsilon S} count\left(gram_n\right)}, \tag{3.10}$$

где RS - реферативные резюме, n - длина n-граммы, $gram_n$ и $Count_{match}(gram_n)$ - максимальное количество n-грамм, встречающихся как в кандидате, так и в реферативном резюме.

ROUGE-L - это мера слов Longest Common Subsequence (LCS). Преимущество LCS в том, что она требует не последовательных, а последовательно повторяющихся совпадений, отражающих порядок слов на уровне предложения. Кроме того, нет необходимости в предопределенной длине n-граммы, поскольку LCS автоматически включает самые длинные общие n-граммы в последовательности.

ROUGE-l основан на оценке LCS, которая вычисляет сходство между двумя рефератами, предполагая, что X - золотой реферат, а Y - реферат-кандидат. ROUGE-L вычисляется следующим образом в (3.13):

$$R_{lcs} = \frac{\mathrm{LCS}\left(X, Y\right)}{m}, \tag{3.11}$$

$$P_{lcs} = \frac{\mathrm{LCS}\left(X, Y\right)}{n}, \tag{3.12}$$

$$F_{lcs} = \frac{\left(1 + \beta^2\right) R_{\mathrm{lcs}} P_{\mathrm{lcs}}}{R_{lcs} + \beta^2 P_{\mathrm{lcs}}}, \tag{3.13}$$

где LCS(X, Y) - мера длины LCS из X и Y, а $\beta = P_{\mathrm{lcs}}/R_{\mathrm{lcs}}$.

К сожалению, базовый LCS не различает LCS различных промежутков между словами в LCS в пределах последовательностей их встраивания. Поэтому для улучшения базовой LCS был введен новый алгоритм WLCS, который запоминает длину последовательных совпадений, встречавшихся до сих

пор, в обычной двумерной динамической программной таблице, вычисляющей LCS, как в (3.16).

$$R_{wlcs} = f^{-1}\frac{\mathrm{WLCS}\,(X,Y)}{f\,(m)}, \tag{3.14}$$

$$P_{wlcs} = f^{-1}\frac{\mathrm{WLCS}\,(X,Y)}{f\,(n)}, \tag{3.15}$$

$$F_{wlcs} = \frac{\left(1+\beta^2\right)R_{\mathrm{wlcs}}P_{\mathrm{wlcs}}}{R_{\mathrm{wlcs}}+\beta^2 P_{\mathrm{wlcs}}}, \tag{3.16}$$

где f^{-1} - обратная функция весовой функции f, с помощью f можно параметризовать алгоритм WLCS для присвоения различных весов последовательным совпадениям в последовательности, так что последовательные совпадения получают больше баллов, чем непоследовательные.

ROUGE-S также известен как skip-gram co-occurrence, допускает любые пробелы между парами слов. Например, skip-bigram измеряет совпадение между двумя словами, которые находятся на расстоянии максимум двух пробелов друг от друга.

Учитывая резюме длины (X) m и n (Y), предполагая, что X является эталоном, а Y - резюме-кандидатом, F-мера на основе скип-биграмм может быть вычислена следующим образом в (3.19):

$$R_{skip2} = \frac{\mathrm{SKIP2}\,(X,Y)}{C\,(m,2)}, \tag{3.17}$$

$$P_{skip2} = \frac{\mathrm{SKIP2}\,(X,Y)}{C\,(n,2)}, \tag{3.18}$$

$$F_{skip2} = \frac{\left(1+\beta^2\right)R_{\mathrm{skip2}}P_{\mathrm{skip2}}}{R_{\mathrm{skip2}}+\beta^2 P_{\mathrm{skip2}}}, \tag{3.19}$$

где SKIP2(X,Y) - число совпадений между X и Y, β контролирует относительную важность P_{skip2} и R_{skip2}, а C - функция комбинации.

[115] сообщил о стабильности и надежности ROUGE при различных объемах выборки. Тем не менее, достижение высокой корреляции с человеческим суждением при реферировании нескольких документов, как это уже сделал ROUGE в задачах реферирования одного документа, все еще остается открытой темой исследования.

В Глава 3.4 мы расскажем о проведенных нами экспериментах по оценке верхнего уровня качества автореферата по метрике ROUGE достижимого при помощи Экстрактивных методов Автоматического Реферирования Текстов.

3.3.4 Пирамида

Пирамида - еще один вариант калибровки метрик под человеческие маневры реферирования [117].

Пирамидальный подход состоит из двух задач:

1 Человеческие аннотаторы определяют единицы краткого содержания (SCU), которые являются наборами текстовых фрагментов, выражающих одно и то же основное содержание, в резюме моделей и создают пирамиду (SCU взвешиваются в соответствии с количеством моделей, в которых они появляются).

2 Оценить новое резюме по пирамиде. Оценка пирамиды вычисляется путем деления общего веса всех SCU, присутствующих в кандидате, на общий вес SCU, возможный для резюме средней длины.

Пирамида - это надежная и предсказуемая метрика. Она помогает определить, какая важная часть отсутствует, а также честно сравнить оценки для различных входных наборов. Однако у нее есть два основных недостатка:

1 Метрика "Пирамида"игнорирует взаимозависимость между единицами контента.

2 Создание начальной пирамиды требует много работы, и для масштабного применения метода потребуется подход с достаточным уровнем автоматизации [117].

Полуавтоматический алгоритм "Пирамида"состоит из пяти шагов:

1 Создать начальную пирамиду.

2 Перечислите все кандидаты в участники (связные фразы) в каждом предложении резюме коллеги.

3 Найдите наиболее похожий SCU для каждого кандидата.

4 Найти расходящееся множество вкладчиков с максимальным общим сходством с пирамидой.

5 Рассчитать суммарный балл пирамиды, используя выбранные вкладчики и их веса SCU.

Предположим, что пирамида имеет n ярусов, T_n на вершине и T_1 внизу. Веса SCU в ярусе T_i будут равны i. Тогда, пусть $|T_i|$ обозначает количество SCU в ярусе $|T_i|$, а D_i - количество SCU в резюме, появляющихся в T_i. Сводным SCU, которые не появляются в пирамиде, присваивается нулевой вес. Общий вес SCUs D находится в (3.20):

$$D = \sum_{i=1}^{n} i \times D_i, \tag{3.20}$$

Резюме с оптимальной оценкой содержания X SCUs приведено в (3.21):

$$Max = \sum_{i=j+1}^{n} i \times |T_i| + j \times \left(X - \sum_{i=j+1}^{n} |T_i| \right), \tag{3.21}$$

где j определяется $j = \max \left(\sum_{t=i}^{n} |T_t| \geqslant X \right)$.

3.3.5 Summarization Evaluation by Relevance Analysis (SERA)

Автоматизированная метрика SERA на высоком уровне обобщения оценивает балл релевантности между сгенерированным резюме и резюме золотого стандарта; балл основан на подходах информационного поиска. В качестве входных данных алгоритм может использовать ключевые слова, фразы, состоящие из существительных, которые могут быть получены из текста сгенерированного резюме. Ключевые слова и фразы из существительных формируют запросы для полнотекстового поиска в базе данных резюме золотых стандартов. В результате поиска первые несколько документов из выдачи, ранжированные по релевантности, могут быть использованы для кодирования и вычисления человеческих оценок качества текста резюме. Такой подход позволяет использовать термины, не эквивалентные лексически, но семантически связанные.

Для обобщения научных текстов авторы SERA рассматривают научные статьи как контекст для слов, из которых состоят статьи. Таким образом, если два слова встречаются в похожих статьях, они семантически связаны. Аналогично, они считают два резюме похожими, если они ссылаются на один и тот же набор статей, даже если они не имеют много общего лексического содержания. Разработчики используют информационный поиск, чтобы определить, относится ли резюме к статье, рассматривая резюме как запросы, а статьи как текстовые документы. Затем они ранжируют статьи на основе их соотнесенности с данным резюме. Численно близкие рейтинги статей указывают на то, что резюме семантически связаны для данной пары резюме-кандидатов и резюме-ссылок [118].

SERA определяется следующим образом (3.22):

$$SERA = \frac{1}{M} \sum_{i=1}^{M} \frac{|R_C \cap R_{G_i}|}{|R_C|}, \tag{3.22}$$

Исходя из целевой области, первоначально строим индекс из набора связанных текстов статей. Задав резюме-кандидата C и набор ссылочных резюме G_i, запросите поисковую систему с текстами резюме-кандидата и золотого резюме и сравните их ранжированные результаты. R_C - ранжированный список найденных документов для резюме кандидата C, а R_G - ранжированный список результатов золотого резюме.

3.3.6 Graph Distance (GRAD)

Мотивацией для разработки другой метрики является устранение недостатков предыдущих подходов [119]. Идея метрики GRAD заключается в использовании семантического графа входного текста. Узлы семантического графа - это термины или слова, используемые в тексте, а вес ребер между узлами соответствует семантической связи соседних узлов слов. Таким образом, проверяемая гипотеза заключается в том, что справедливое резюме должно содержать слова, соответствующие узлы которых имеют максимальное количество соседей из исходного текста в семантическом графе. Более того, наоборот, если в тексте резюме много терминов с удаленными от исходного текста узлами, то такое резюме должно быть оценено ниже. Формально, мера качества резюме - это инвертированная сумма весов для каждого термина в тексте к его ближайшему термину в тексте резюме. Для расчета этой метрики необходимо, чтобы для каждого исходного текста имелось как минимум два резюме.

Авторы GRAD утверждают, что хорошее резюме состоит из терминов, относящихся к центральным вершинам семантического графа, то есть терминов, связанных с максимальным количеством других терминов в исходном тексте. Что касается метрики GRAD, балл резюме оценивается как нормализованная инвертированная сумма расстояний от каждого термина текста до его ближайшего термина резюме S, как показано в (3.23):

$$score(S) = \frac{1}{|S| \sum_{v_i} \min_{v_j \epsilon V \cap S} d\left(v_j, v_i\right)}, \tag{3.23}$$

где $d\left(v_j, v_i\right)$ - кратчайший путь между v_i и v_j. Нормализация выполняется путем деления оценки на количество обобщенных терминов. Нормализация необходима для того, чтобы метрика не отдавала предпочтение более длинным резюме.

Дополнительные результаты показали, что метрика GRAD не может отличить сгенерированный текст резюме от других резюме, созданных человеком. Тем не менее, она может оценить сходство между ними. Исследователи предлагают изучить различные дополнительные функции для улучшения работы метрики GRAD, такие как обратная документальная частота терминов или теги части речи.

3.3.7 Подход к модели бакронимического языка для оценки качества резюме (BLANC)

Метрика BLANC была предложена в качестве замены суммарных оценок качества семейства ROUGE [120].

BLANC можно определить как численную меру того, насколько резюме помогает независимой языковой модели выполнить задачу понимания исход-

ного документа. Авторы сосредоточились на задаче маскировки лексем, где перед моделью ставится задача восстановить замаскированные участки текста. Для предсказания маскированных текстовых лексем авторы BLANC использовали языковую модель BERT.

Существует две версии метрики BLANC:

1 **BLANC-help** непосредственно присоединяет текст резюме к каждому предложению.

2 **BLANC-tune** настраивает языковую модель и обрабатывает весь документ, используя текст резюме.

BLANC-помощь можно определить следующим образом в (3.24):

$$BLANC_{help} = A_s - A_f = \frac{S_{01} - S_{10}}{S_{\text{total}}}, \tag{3.24}$$

После перебора всех предложений в тексте и всех возможных комбинаций маскировки алгоритм получает четыре суммарных подсчета успешных и неуспешных маскировок $S_{ij}, i = 0, 1; j = 0, 1$. Здесь индекс i равен 0 (неуспешная маскировка) или 1 (успешная маскировка) - для заполняющего входа. Индекс j определяется аналогично для суммарного входа. Значения BLANC могут варьироваться от -1 до 1, но практически типичными являются значения от 0 до 0,3.

3.4 Оценка наивысшего качества автореферата достижимого экстрактивными методами автореферирования текста

Метод *Автоматического Экстрактивного Реферирования* (АЭР, ETS) для поиска важной информации из текста автоматически использует предложения из исходного текста. В этой главе мы ответим на вопрос, какого качества резюме мы можем достичь с помощью методов АЭР? Чтобы максимизировать оценку ROUGE-1, мы использовали пять подходов:

1 Адаптированный редуцированный Поиск по Изменяемым Окрестностям (ПИО, RVNS).
2 Жадный алгоритм.
3 ПИО инициализированный по результатам работы жадного алгоритма.
4 Генетический алгоритм.
5 Генетический алгоритм, инициализированный результатами жадного алгоритма.

Кроме того, мы провели эксперименты на статьях из набора данных arXive. В результате мы обнаружили 0,59 и 0,25 баллов для ROUGE-1 и ROUGE-2, соответственно, достижимых подходом, где *генетический алгоритм, инициализированный результатами жадного алгоритма*, что дает наилучшие результаты из всех протестированных подходов. Более того, эти оценки выше, чем оценки, полученные современными моделями обобщения текста: лучшая оценка в литературе для ROUGE-1 на том же наборе данных составляет 0,46. Таким образом, у нас есть место для развития методов ETS, которые сейчас незаслуженно забыты [121].

Результаты работы обобщенные в данной главе опубликованы в Akhmetov I, Mussabayev R, Gelbukh A. 2022. Reaching for upper bound ROUGE score of extractive summarization methods. PeerJ Comput. Sci. 8:e1103 DOI 10.7717/peerj-cs.1103.

3.4.1 Определение автоматического реферирования как оптимизационной задачи

Проблема оценки качества резюме, генерируемых моделями экстрактивного реферирования, рассматривается в Глава 3.3, где мы описали несколько методов, среди которых ROUGE scoring является наиболее распространенным и часто используемым. Таким образом, мы определяем ATS как задачу максимизации ROUGE-баллов сгенерированных резюме.

Мы используем оценки ROUGE-1 и ROUGE-2: которые определяют, в какой степени униграммы и биграммы в пересечении между резюме кандидата и эталона встречаются в них; см. Раздел 3.3.3. Под эталоном здесь подразумевается *эталонные пример* или так называемый *золотой стандарт* резюме для сравнения с произведенной оценкой резюме с помощью ROUGE score.

Поэтому, чтобы определить проблему оптимизации, мы должны найти та-

кой набор предложений в каждом тексте, который даст максимальную метрику ROUGE. Однако выполнение этой задачи с помощью алгоритма грубой силы может занять целую вечность, поэтому нам нужен более интеллектуальный метод.

Мы используем метод *Variable Neighborhood Search* (VNS) [122, 123], который использует эвристические методы поиска для получения оптимального решения за относительно короткое время. Более подробно о VNS можно прочитать в разделе Раздел a).

Для этой же задачи мы использовали *жадный алгоритм*. Вкратце, он заключается в том, чтобы взять для реферата предложения из текста, содержащие максимальное количество слов. Более подробно о методе в Раздел b). Мы также экспериментировали с VNS, инициализированной решением жадного алгоритма.

Наконец, мы экспериментировали с Генетическим алгоритмом и Генетическим алгоритмом, инициализированным результатами жадного алгоритма; см. Раздел d) и Раздел e).

Вклад нашего исследования в научные знания заключается в следующем:

1 Выявление верхней границы ROUGE оценки методов экстрактивного суммирования.

2 Очищенный набор данных с различными типами резюме высокого ROUGE и полезной текстовой статистикой. Наборы данных, созданные в ходе настоящего исследования и/или проанализированные в ходе него, доступны в хранилище данных Mendeley по адресу `https://data.mendeley.com/datasets/nvsxfcbzdk/1`.

3 Код для воспроизведения реализованного исследования[6].

3.4.2 Эксперименты

В нашей предыдущей статье [99] мы подошли к задаче поиска наилучшей оценки ROUGE-1, используя только эвристику VNS. Однако в этот раз для сравнения мы также использовали жадный алгоритм, генетический алгоритм и добавили оценку ROUGE-2.

Применение алгоритмов оптимизации здесь логически вытекает из того факта, что поиск наилучшего резюме из предложений текста путем перебора всех возможных комбинаций непрактичен из-за $O(n!)$ сложности такого подхода:

$$\binom{N_t}{N_a} = \frac{N_t!}{N_a!(N_t - N_a)!} \tag{3.25}$$

где N_a и N_t - количество предложений в резюме и тексте, соответственно. В то время как алгоритмы оптимизации дают довольно простую альтернативу,

[6] `https://github.com/iskander-akhmetov/Reaching-for-Upper-Bound-ROUGE-Score-of-Extractive-Summarization-Methods`

которая может обеспечить удовлетворительное решение за разумное время.

Поэтому мы используем VNS, жадный и генетический алгоритмы для поиска лучших комбинаций предложений из текстов статей, дающих наивысший балл ROUGE-1, с оригинальными аннотациями статей в качестве ссылки.

a) Поиск по Изменяемым Окрестностям (ПИО, VNS)

Используя терминологию VNS, для каждой из 17 038 статей в экстракте набора данных arXive (Таблица 3.3) мы выполнили следующие шаги в цикле:

1 **Инициальное решение** - Мы инициализируем наш поиск решения случайным набором предложений x в $\mathcal{N}_k = \binom{N_t}{N_a}$ пространстве возможных структур окрестностей, для которых мы получаем оценку ROUGE-1 [62].

2 **Встряска** - Вносить изменения, начиная с замены одного случайно выбранного предложения на новое из текста до замены k_{max} предложений, если не происходит улучшения оценки ROUGE-1. Максимальное количество изменений - параметр k_{max} (в нашем случае $k_{max} = 3$).

3 **Incumbent solution** - Пересчитайте оценку ROUGE-1 и зафиксируйте результат, если он лучше, чем начальная оценка, сбросьте k до одного предложения, или, если улучшения не происходит, постепенно увеличивайте k до k_{max}.

4 **Условие остановки** - цикл ограничен 5 000 итерациями или 60 секундами. Если после 700 последовательных итераций не происходит улучшения результата ROUGE-1, цикл прерывается.

b) Жадный алгоритм

Мы использовали следующую реализацию жадного алгоритма, основанную на общей идее алгоритма оптимизации данного класса, где мы пытаемся найти наиболее выполнимое мгновенное решение.

Дан полный текст статьи (T), разделенный на предложения (S), и ее аннотация (A):

1 Получите уникальный список слов из A в качестве словаря (V).

2 Создайте матрицу встречаемости слов (M), где для каждого элемента в V (столбцы) и каждого предложения в T (строки) мы имеем двоичное значение, указывающее на присутствие слова в предложении.

3 Пока M не станет пустым:

- Просуммируйте M вдоль оси 0 (строки) и получите индекс максимального значения, индекс предложения, содержащего максимум слов из A. Сохраните индекс в индексном списке (IL).

- Обновите M, удалив столбцы, соответствующие ненулевым значениям для предложения с максимальным количеством слов из A.

4 Для получения количества предложений в резюме, дающих максималь-

ный балл ROUGE:

- Рассчитайте баллы ROUGE для всех комбинаций предложений, начиная с 1 и до длины IL.

- Выберите количество предложений с максимальным баллом ROUGE.

- Обновить IL с наилучшей комбинацией предложений.

5 Отсортируйте индексы в IL в порядке возрастания и восстановите исходный порядок предложений в статье, и собрать резюме, взяв предложения из T по индексам в сортировке IL.

6 Вычислите ROUGE оценку между созданным резюме и A.

c) ПИО (VNS) инициализированный жадным алгоритмом

Мы работали над VNS, инициализированным лучшими результатами, достигнутыми жадным алгоритмом. Это просто модификация алгоритма, описанного в Раздел a), где мы вместо случайной инициализации используем предложения из лучших резюме, достигнутых жадным алгоритмом.

d) Генетический алгоритм

Вдохновленные результатами, которые эволюционные алгоритмы показывают в различных приложениях [103], мы разработали реализацию генетического алгоритма для нахождения верхней границы для оценки ROUGE.

Даны текст (T) и его реферат (A):

1 Вычислите длину T и A в количестве предложений (len_T и len_A).

2 Перемешайте предложения в T.

3 Генерировать начальную генерацию кандидатов на резюме, разрезая список предложений в T на фрагменты размером len_A.

4 Установите число потомков равным половине числа начальных кандидатов ($n\_offfsprings$).

5 Продолжение для 6 поколений:

   1 Пересечение всех кандидатов между собой путем смешивания предложений двух кандидатов, их перетасовки и случайного выбора len_A количества предложений.

   2 Рассчитайте оценку ROUGE-1 для всех потомков.

   3 Выберите лучшие $n\_offfsprings$ по баллу ROUGE-1 и повторите.

4 Выберите потомство из последнего поколения с наивысшим баллом ROUGE-1 и верните его в качестве сгенерированного итога.

e) Генетический алгоритм, инициализированный жадным алгоритмом

Этот алгоритм в основном такой же, как и случайно инициализированный генетический алгоритм (Раздел d)). Но на шаге 3 мы добавляем к начальным кандидатам резюме, сгенерированное жадным алгоритмом (Раздел b)).

3.4.3 Результаты

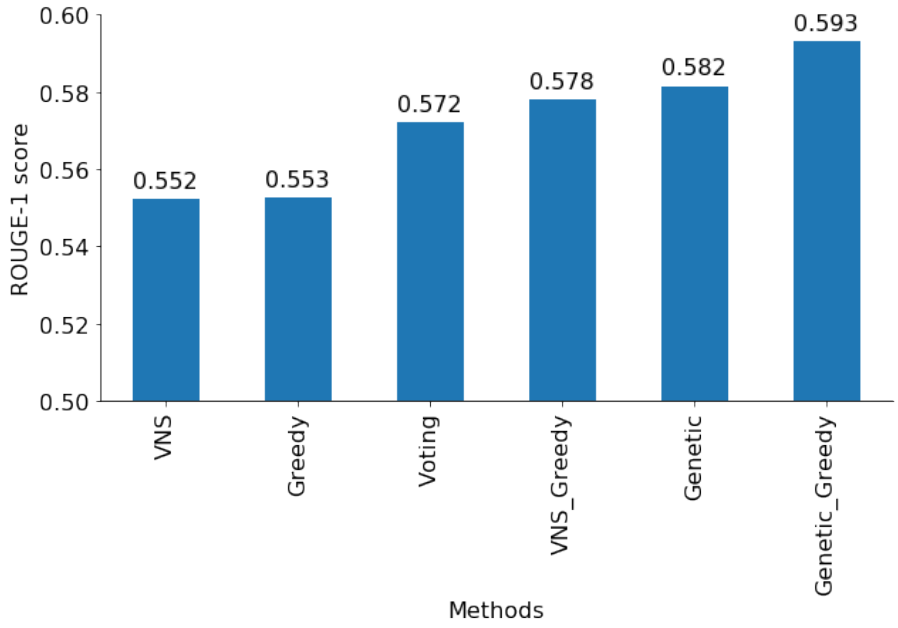
Применяя алгоритмы, описанные в Раздел 3.4.2, мы показали, что наилучших результатов достиг генетический алгоритм, инициализированный результатами жадного алгоритма 0.59/0.25 для оценок ROUGE-1/ROUGE-2; см. Таблица 3.7 и Рис. 3.4. В то время как лучшие современные методы, использующие сложные архитектуры нейронных сетей [55–57], могут достичь ROUGE-1 всего $\tilde{0}.45$ и ROUGE-2 0.17 на наборе данных arXive; см. Таблица 3.8. Таким образом, методы и технологии обобщения определенно нуждаются в улучшении.

Таблица 3.7 – Результаты расчета наилучших достижимых оценок ROUGE (R-1 и R-2) с использованием экстрактивных методов обобщения. Цифры, выделенные жирным шрифтом, указывают на самые высокие значения для ROUGE-1 (R-1) и ROUGE-2 (R-2) по ряду

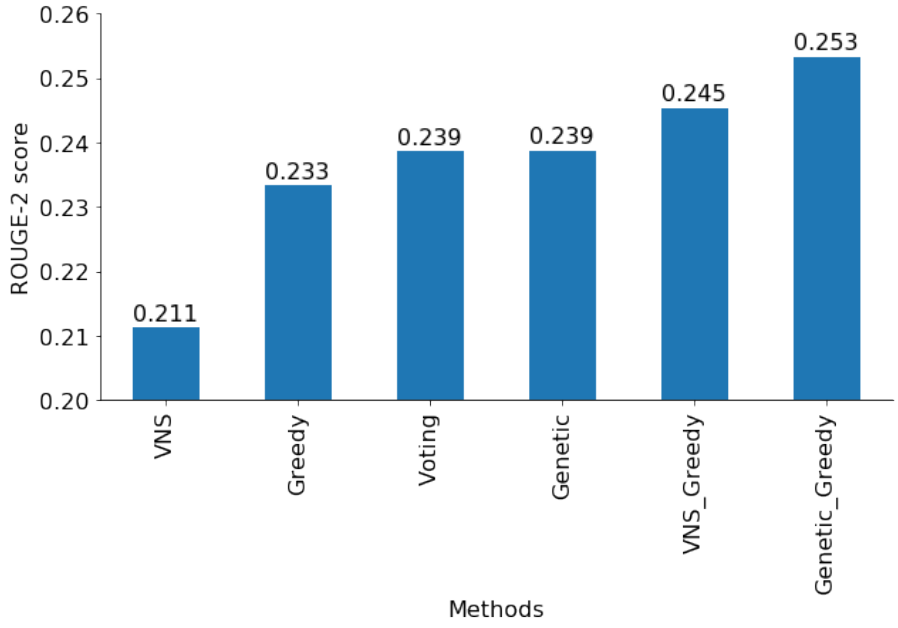
	VNS		Greedy		VNS_Greedy		Genetic		Genetic_Greedy	
	R-1	R-2	R-1	R-2	R-1	R-2	R-1	R-2	R-1	R-2
count	17,038									
mean	0.55	0.21	0.55	0.23	0.58	**0.25**	0.58	0.24	**0.59**	**0.25**
std	0.07	0.08	0.08	0.10	0.08	0.10	0.07	0.09	0.08	0.10
min	0.07	0.01	0.04	0.01	**0.09**	**0.02**	**0.09**	0.01	**0.09**	0.01
25%	0.52	0.16	0.51	0.16	0.54	0.18	0.55	0.18	**0.56**	**0.19**
50%	0.56	0.20	0.55	0.21	0.58	0.22	0.59	0.23	**0.60**	**0.24**
75%	0.59	0.25	0.60	0.28	0.62	0.29	0.63	0.29	**0.64**	**0.30**
max	0.84	0.78	**0.97**	0.93	**0.97**	**0.95**	0.86	0.84	0.92	0.88

Таблица 3.8 – Сравнение верхней границы, полученной с результатами современных моделей обобщения текстов лидеров на наборе данных arXive [55,61]. Цифры, выделенные жирным шрифтом, указывают на максимальные значения по столбцу

Class	Model	ROUGE-1	ROUGE-2
Genetic_Greedy upper bound		**0.59**	**0.25**
Extractive	SumBasic [62, 124]	0.30	0.07
	LexRank [26]	0.34	0.11
	LSA [125]	0.30	0.07
Abstractive	Attn-Seq2Seq [47]	0.29	0.06
	$\text{PEGASUS}_{\text{BASE}}$ [55]	0.35	0.10
	$\text{PEGASUS}_{\text{LARGE}}$ [55]	0.45	0.17
	Pntr-Gen-Seq2Seq [60]	0.32	0.09
	Discourse-att [61]	0.36	0.11



(a) ROUGE-1 балл



(b) ROUGE-2 балла

Рисунок 3.4 – Сравнение верхнего предела оценки ROUGE для различных методов на 17 038 статьях из набора данных arXive.

Интересно, что лучшие резюме, созданные пятью методами: VNS, жадный алгоритм, VNS инициализированный результатами жадного алгоритма, генетический алгоритм и генетический алгоритм инициализированный результатами жадного алгоритма, имеют разное количество предложений в среднем 15, 7, 10 и 12 для последних двух. Мы объясняем наличие семи предложений в автореферате сгенерированном жадным алгоритмом тем, что алгоритм намеренно выбирает наиболее лексически насыщенные предложения длиннее

среднего и, таким образом, максимизирует оценку ROUGE с меньшим количеством предложений. В отличие от этого, VNS пробует случайные комбинации предложений, не учитывая их свойств. Более того, определить оптимальное количество предложений, которые необходимо собрать для резюме, также непросто.

3.4.4 Обсуждение

Важным фактором в экстрактивном обобщении для максимизации оценки ROUGE является определение оптимального количества предложений, которые необходимо взять из исходного текста. Однако до сих пор мы не видели четкой корреляции между оптимальным количеством предложений, заданным VNS, жадным или генетическим алгоритмами, и какими-либо другими факторами, такими как длина текста в символах, словах и предложениях, и другими характеристиками.

Стейнбергер и Йезек [125] изучали важность длины резюме, но они подразумевают с помощью оценки Latent Semantic Analysis (LSA), что чем длиннее резюме, тем лучше. Статья была опубликована в том же году, когда был введен показатель ROUGE score [62], который сейчас является промышленным стандартом для оценки резюме. И если мы используем показатель ROUGE, то более длинные резюме увеличивают отзыв, но снижают точность. Поэтому необходимо дальнейшее исследование и обсуждение оптимального количества предложений для оптимального значения оценки ROUGE.

С другой стороны, скоринг ROUGE предполагает, что ссылочное резюме является базовой истиной, и избавляет от необходимости проверять само ссылочное резюме относительно текста статьи. В то же время, это может быть ориентировочное резюме в стиле тизера. Или же резюме, которое мы используем в качестве эталона в ROUGE scoring, может быть очень абстрактным, содержащим небольшое количество слов и предложений из самого текста, и в этом случае метод экстрактивного суммирования даст низкие результаты.

3.5 Модель Автоматического Экстрактивного Реферирования научных текстов на основе жадного алгоритма

В этой главе представлен метод обобщения научных статей из наборов данных arXive и PubMed с помощью жадного алгоритма экстрактивного реферирования текстов. Мы использовали этот подход вместе с Variable Neighborhood Search (VNS) для изучения того, что существует в области качества экстрактивного суммирования текста с точки зрения оценок ROUGE; подробнее об этом в Глава 3.4. Алгоритм основан на первоначальном отборе для резюме предложений из текста, содержащих максимальное количество слов с более высокими значениями TFIDF наряду с настройкой параметра минимальной частоты документа для векторизации TFIDF. В результате метод достигает 0,43/0,12 и 0,40/0,13 для оценок ROUGE-1/ROUGE-2 на наборах данных arXive и PubMed соответственно. Эти результаты сопоставимы с современными моделями, обученными на больших объемах текстовых данных, использующими сложные архитектуры нейронных сетей и серьезные вычислительные ресурсы. В отличие от этого, наш метод использует простую методологию статистического вывода [63].

Результаты работы обобщенные в данной главе опубликованы в I. Akhmetov, A. Gelbukh and R. Mussabayev, "Greedy Optimization Method for Extractive Summarization of Scientific Articles,"in IEEE Access, vol. 9, pp. 168141-168153, 2021, doi: 10.1109/ACCESS.2021.3136302.

3.5.1 Эксперименты

а) Метод автоматического реферирования

В работе [99] мы экспериментировали с использованием метода обучения с учителем для построения модели бинарной классификации предложений, что похоже на то, что было предложено Купицем в 1995 году [33], поскольку мы получили метки, используя метод VNS. Однако в этот раз мы также получили метки, полученные жадным алгоритмом, но не смогли построить хорошую модель классификации.

Для методов векторизации мы попытались использовать Fasttext [126], BERT [127], Electra [128] и Elmo [129]. Мы также попытались дополнить полученные векторы различными пользовательскими признаками, такими как порядковый номер предложения с начала. Конечные признаки, такие как количество существительных, глаголов и прилагательных в предложении, косинусное расстояние предложения от двух соседних предложений, взаимная информация предложения с соседними слева и справа. Более того, ничто не помогло нам достичь более 0,71 балла точности для сбалансированного набора данных.

Поэтому мы решили исследовать подход, основанный на жадном алгоритме, когда мы пытались узнать наилучшую возможную оценку ROUGE

с помощью техники экстрактивного обобщения. Однако на этот раз мы не хотим рассматривать предоставленный реферат статьи A и модифицируем алгоритм следующим образом:

Дан полный текст статьи (T), разделенный на предложения (S):

1 TfIdfVectorize T in c min_df=0.042 (найдено эмпирически), возвращающая матрицу (M). Мы используем здесь `TfidfVectorizer` [130] потому что на этот раз мы должны учитывать важность слов в тексте.

2 Пока Mx не станет пустым:

- Просуммируйте M вдоль оси 0 (или строк) и получите индекс максимального значения, индекс предложения, содержащего максимальное количество слов со значением TFIDF из T. Сохраните индекс в индексном списке (IL).

- Обновите M, удалив столбцы, соответствующие ненулевым значениям для предложения с максимальной суммой значений TFIDF слов из T.

3 Мы берем 8 верхних индексов предложений из IL с максимальной суммой значений TFIDF слов из T. В среднем, Жадный получает примерно восемь предложений (или 7,7 предложений, если быть более точным) для достижения максимальной оценки ROUGE.

4 Отсортируйте индексы в IL в порядке возрастания, чтобы восстановить исходный порядок следования предложений в статье.

5 Соберите резюме, взяв предложения из T по индексам в сортировке IL.

6 Вычислите ROUGE оценку между созданным резюме и A.

Блок-схема алгоритма GreedSum показана в Рис. 3.5, а его принципиальная схема представлена в Рис. 1.1.

b) Оптимизация параметров min_df и max_df в `TfidfVectorizer`

Еще одна проблема, которую нам предстоит решить - оптимизация параметров min_df и max_df в `TfidfVectorizer`, поскольку мы не видим аннотацию статьи заранее для целей реферирования и должны учитывать не только важность слов, но и отфильтровывать редко встречающиеся слова.

Чтобы решить эту проблему, мы получаем репрезентативную выборку из набора данных из 17 038 статей, оставшихся после очистки данных исходного набора данных arXive. Затем, для каждой статьи, получаем резюме, используя метод суммирования, описанный в Раздел a) и значение ROUGE-1, используя аннотацию статьи в диапазоне от 0.001 до 1.0 для min_df и max_df, чтобы получить средние значения в качестве оптимальных значений параметров.

Расчет репрезентативного размера выборки приведен в (3.4).

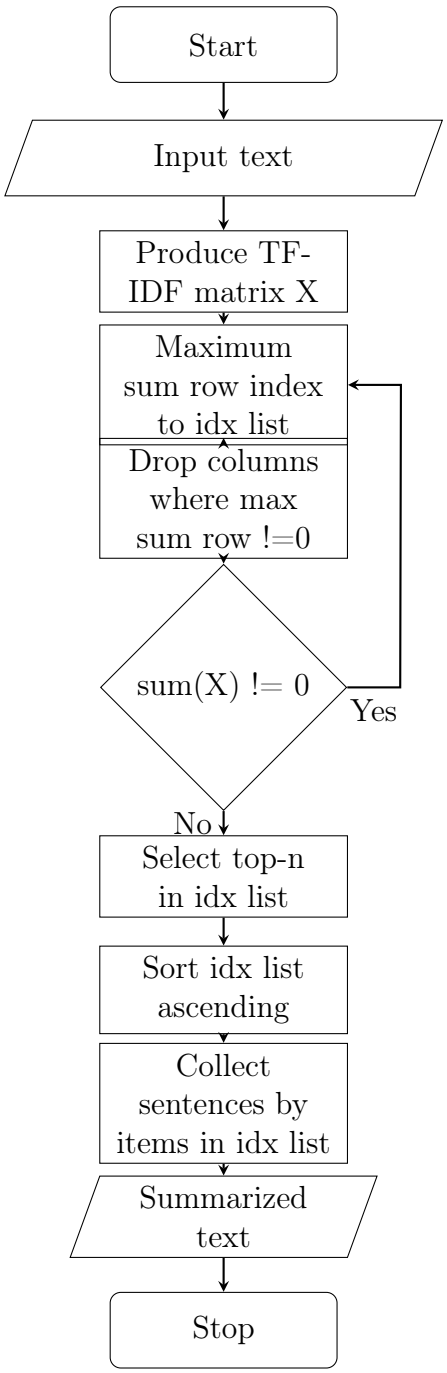


Рисунок 3.5 – GreedSum web-app algorithm flowchart.

3.5.2 Результаты

а) Результаты реферирования жадным алгоритмом

Чтобы жадный алгоритм автореферирования работал, нам необходимо собрать статистику о текстах, с которыми мы хотим работать для извлечения резюме. Некоторые статистические данные можно собрать быстро и легко (например, длину текста), но другие требуют утомительных тестов и пробных процедур, требующих вычислительных ресурсов и времени (например, поиск оптимальных значений параметра минимальной частоты документов для векторизатора). Поэтому мы решили собирать статистику не по всему

набору данных, а по его репрезентативной выборке.

b) Получение выборки

Расчет объема выборки и случайная выборка Используя (3.4), мы получаем объем выборки 376 статей из набора данных 17 038 статей. После случайной выборки мы проверяем, представляет ли выборка совокупность; под совокупностью здесь понимается набор данных, из которого была взята выборка.

Оценка качества образца Визуализация: Чтобы проверить качество нашей выборки, мы сначала посмотрим на графические представления распределений длины текста и длины абстракции в предложениях как для популяции, так и для выборки; см. Рис. 3.6 и Рис. 3.7.

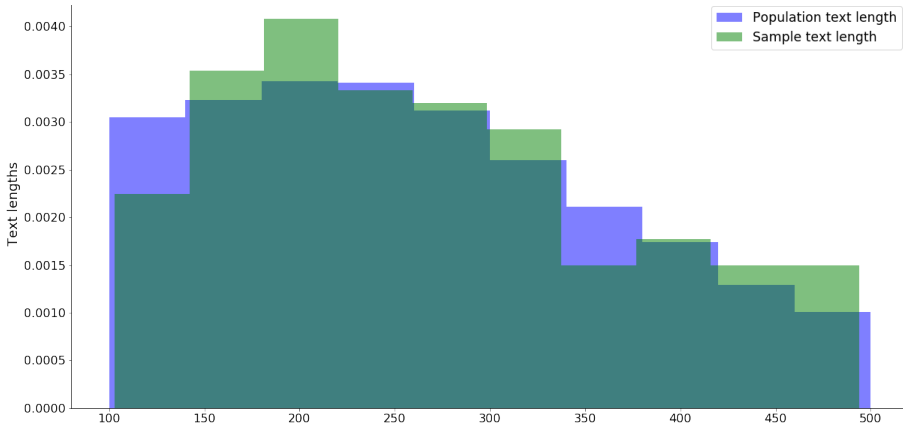


Рисунок 3.6 – Относительные частотные распределения длины текста популяции и выборки. Темно-зеленый цвет указывает на пересечение между популяцией и выборкой

.

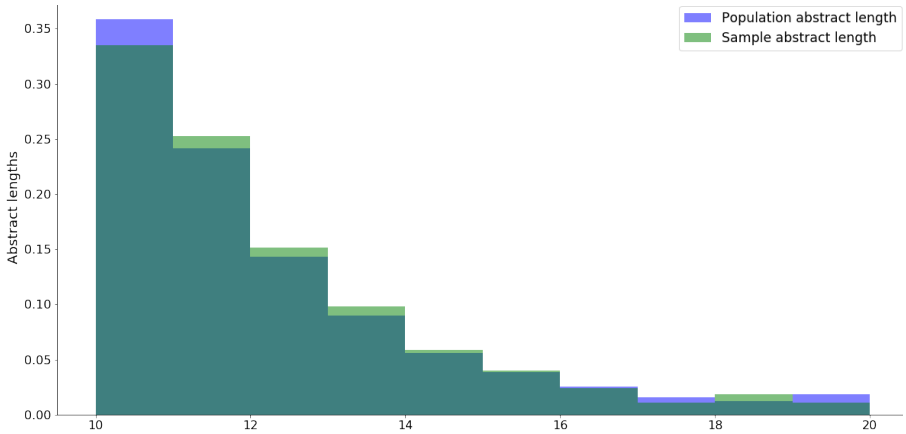


Рисунок 3.7 – Относительные частотные распределения длин рефератов популяции и выборки. Темно-зеленый цвет указывает на пересечение популяции и выборки

.

Визуально совокупность и выборка хорошо согласуются. Более того, теперь мы ищем фактические цифры в Таблица 3.9, которые дают нам такие статистические свойства, как среднее, стандартное отклонение, минимумы и максимумы, а также квартильные значения.

Таблица 3.9 – Длина текста и реферата в предложениях статистические свойства для выборки и популяции.

Properties	Sample		Population		Absolute differences	
	len_text	len_abstract	len_text	len_abstract	len_text	len_abstract
count	376		17 038		-	
mean	269.16	11.75	263.44	11.75	5.72	0.00
std	102.31	2.03	102.57	2.13	0.27	0.11
min	103.00	10.00	100.00	10.00	3.00	0.00
25% .	186.75	10.00	179.00	10.00	7.75	0.00
50%	253.50	11.00	252.00	11.00	1.50	0.00
75%	335.00	13.00	338.00	13.00	3.00	0.00
max	494.00	20.00	500.00	20.00	6.00	0.00
Total					27.24	0.11

Разница в статистических свойствах: Различия между статистическими свойствами популяции и выборки, такими как признак длины реферата, кажется незначительными. Также и для признака длины текста цифры не очень большие (не превышают 3% разницы); см. Таблица 3.9.

Корреляция: Тот факт, что `len_text` и `len_abstract` колонки выборки и популяции коррелируют с коэффициентами 0.99965 и 0.99999 соответственно, хорошо подтверждает предположение о хорошем качестве выборки.

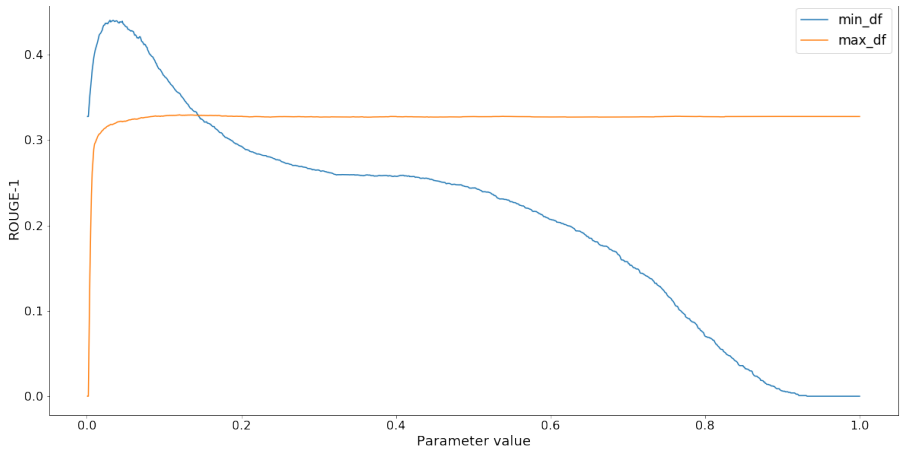
Колмогоров-Смирнова тест: Мы провели двухвыборочный тест Колмогорова-Смирнова для выборки и ее источника по признакам длины текстов и рефератов. Полученные p-значения для длин текстов и рефератов составляют 0,33 и 0,98, соответственно. Таким образом, поскольку значения p-значений намного больше, чем 0,05, мы не можем отвергнуть нулевую гипотезу (H_0) о том, что обе выборки имеют одинаковое распределение вероятности.

Следовательно, выборка разумно представляет исходные данные и может делать статистические выводы для всей совокупности.

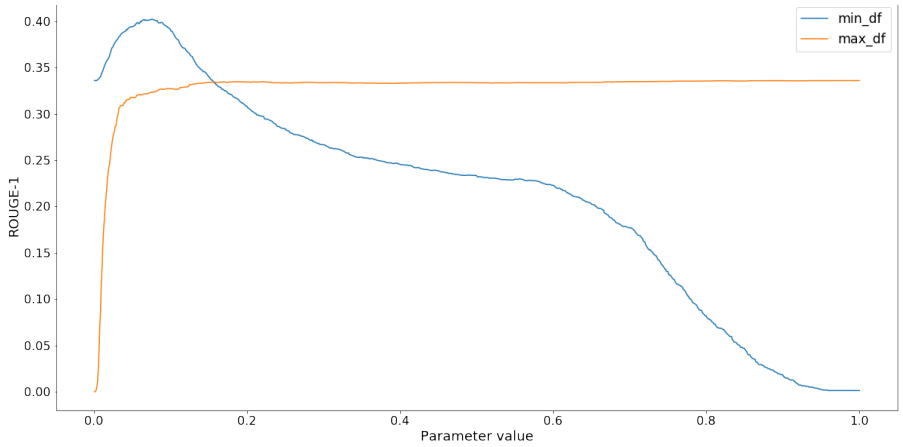
c) Оптимальные параметры минимальной и максимальной частоты документов

Мы запустили алгоритм Brute Force на выборке данных, которая оказалась качественной в представлении источника данных, чтобы найти оп-

тимальные значения параметров минимальной частоты документа и максимальной частоты документа TfidfVectorizer для каждой из 376 точек данных; см. Рис. 3.8.



(a) arXive



(b) PubMed

Рисунок 3.8 – ROUGE-1 score as function of minimum and maximum document frequency parameters.

Как видно на Рис. 3.8a (набор данных arXive), изменение параметра максимальной частоты документов (`max_df`) влияет на ROUGE-1 очень слабо, и начиная со значения 0,12 эффект исчезает. Напротив, параметр минимальной частоты документов (`min_df`) существенно влияет на ROUGE-1, достигая максимума 0,44 при значении параметра 0,042 и неуклонно снижая оценку ROUGE-1 до нуля по мере приближения значения параметра к 1,0. Мы интерпретируем это как то, что `min_df`значение параметра 0.042 (с вариацией 0.022) позволяет наиболее значимым словам попасть в словарный запас для `TfidfVectorizer`, отфильтровывая слишком редкие слова.

Мы наблюдаем аналогичную ситуацию на наборе данных Рис. 3.8b (PubMed), где максимальное значение ROUGE-1 0,40 достигается при оп-

тимальном параметре минимальной частоты документов (`min_df`) 0,076.

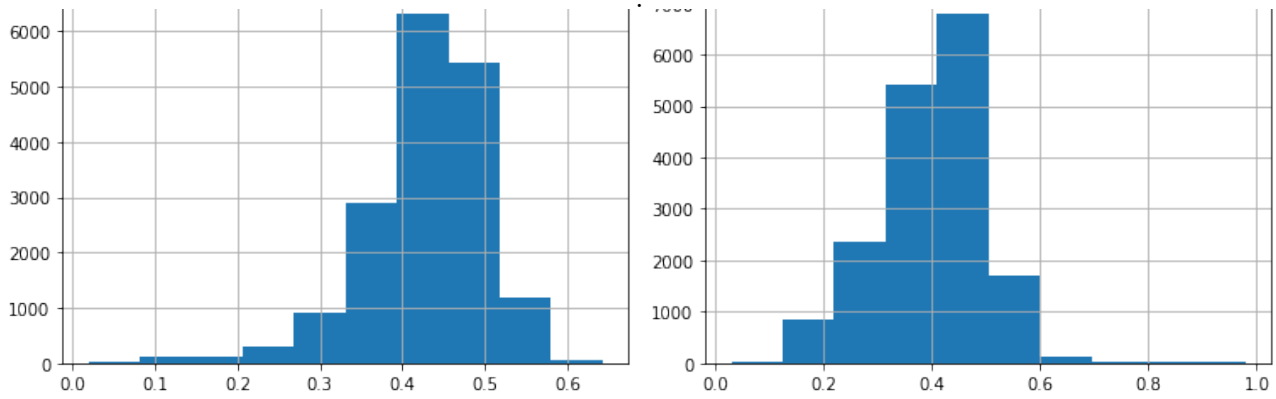
d) Качество реферирования текстов жадным алгоритмом

Мы протестировали наш подход жадного алгоритма на обоих наборах данных arXive и PubMed, по 17 038 статей в каждом, и получили в среднем 0,43/0,12 и 0,40/0,13 баллов ROUGE-1/ROUGE-2 соответственно; см. Таблица 3.10.

Таблица 3.10 – Статистика результатов подхода жадного суммирования для наборов данных arXive и PubMed.

	arXive		PubMed	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
подсчет	17 038			
среднее	0.43	0.12	0.40	0.13
std	0.07	0.05	0.10	0.08
min	0.02	0.00	0.02	0.00
25%	0.39	0.09	0.34	0.08
50%	0.44	0.12	0.41	0.12
75%	0.48	0.15	0.47	0.17
max	0.64	0.49	0.98	0.97

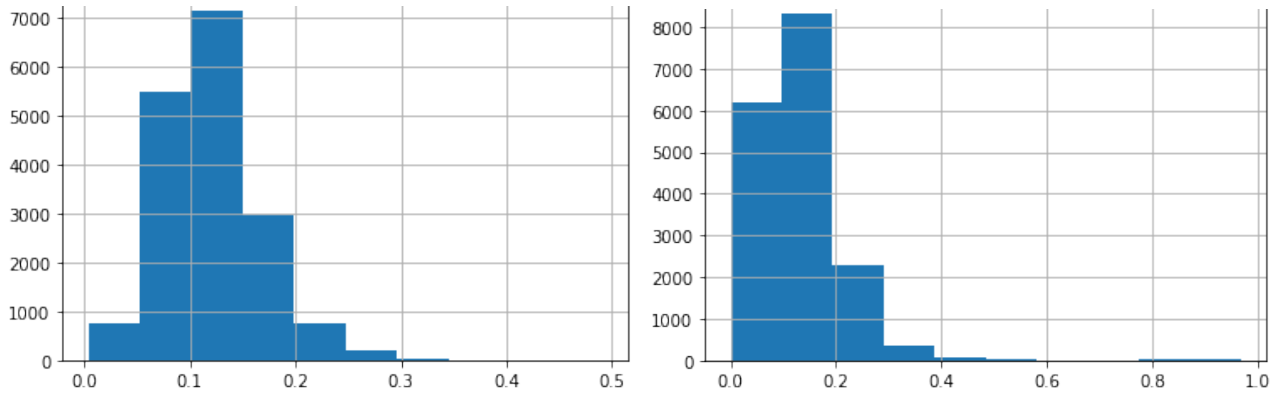
Распределения результатов ROUGE score, показанные в Рис. 3.9 и Рис. 3.10, являются плотными, и позволяют предположить, что подход работает стабильно, как и ожидалось. Примеры резюме, созданных GreedSum на наборе данных arXive, можно найти в Приложение В или по ссылке[7].



(a) arXive (b) PubMed

Рисунок 3.9 – GreedSum: ROUGE-1 распределение баллов.

[7] `https://github.com/iskander-akhmetov/Greedy-Summarization/blob/main/Greedy\%20Summary\%20examples\%20for\%20arXive\%20dataset.md`



(a) arXive (b) PubMed

Рисунок 3.10 – GreedSum: ROUGE-2 распределение баллов.

Таблица 3.11 – Сравнение предложенного подхода с современными моделями реферирования текстов лидеров. Цифры, выделенные жирным шрифтом, указывают на максимальные значения колонок по классам.

		arXive		PubMed	
Class	Model	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
Extractive	SumBasic [61, 62, 124]	0.30	0.07	0.37	0.11
	LexRank [26, 61]	0.34	0.11	0.39	**0.14**
	LSA [61, 125]	0.30	0.07	0.34	0.10
	GreedSum (this work)	**0.43**	**0.12**	**0.40**	0.13
Abstractive	Attn-Seq2Seq [47, 61]	0.29	0.06	0.32	0.09
	$\text{PEGASUS}_{\text{BASE}}$ [55]	0.35	0.10	0.40	0.15
	$\text{PEGASUS}_{\text{LARGE}}$ [55]	**0.45**	**0.17**	**0.46**	**0.20**
	Pntr-Gen-Seq2Seq [60, 61]	0.32	0.09	0.36	0.10
	Discourse-att [61]	0.36	0.11	0.39	0.15

Сравнение подхода GreedSum в Таблица 3.11 показывает, что он превосходит все модели, кроме $\boldsymbol{PEGASUS_{LARGE}}$, который был обучен на корпусе C4 [131], содержащем более 1000M документов, а наш метод использовал всего 376 документов для обучения. Таким образом, предложенный подход уверенно превосходит все известные модели экстрактивного суммирования со значительным отрывом примерно в 1/4. Что касается современной модели Transformer PEGASUS, то можно сказать, что мы достигли такого же уровня производительности.

3.5.3 Обсуждение

В данной работе был предложен метод жадной оптимизации для экстрактивного суммирования научных статей. К сожалению, несмотря на хорошие результаты среди современных моделей, заняв второе место после модели PEGASUS при обобщении статей в наборах данных arXive и PubMed, мы столкнулись с рядом проблем, которые стоит обсудить.

Важным фактором в экстрактивном обобщении для максимизации оценки ROUGE является нахождение оптимального количества предложений, которые должны быть взяты из исходного текста. Однако до сих пор мы не видели четкой корреляции между оптимальным количеством предложений, заданным VNS, жадным алгоритмом, голосованием и VNS, инициализированным жадным алгоритмом, и любым другим фактором, таким как длина текста в символах, словах и предложениях, и другими характеристиками. Важность длины резюме изучали Стейнбергер и Йезек [125], но они подразумевают, что по оценке LSA, чем длиннее резюме, тем лучше. Их статья была опубликована в том же году, когда была введена оценка ROUGE [62], которая сейчас является ожидаемым стандартом для оценки резюме, и более длинные резюме увеличивают recall, но снижают precision. С другой стороны, оценка по ROUGE предполагает, что реферативное резюме является истиной и не требует проверки самого реферативного резюме относительно текста статьи. В то же время, это может быть ориентировочное резюме в стиле тизера. Поэтому необходимо дальнейшее исследование и обсуждение оптимального количества предложений для максимизации ROUGE.

Идея использования частот слов для реферирования текста насчитывает более 60 лет, впервые она была рассмотрена Луном [4] и вновь рассмотрена Ненковой в 2005 году [59] с ее методом SumBasic. Более того, теперь мы возвращаемся к вопросу с разных сторон оптимизации параметра минимальной частоты документа (`min_df`) для максимизации оценки ROUGE. Лун предполагает, что лучшие слова для резюме лежат между самыми частыми словами (отсечка C) и самыми редкими словами (отсечка D); см. Рис. 3.11. Однако наши результаты показывают (Рис. 3.8), что отсечение только самых редких слов имеет смысл для максимизации оценки ROUGE. Более того, возникает вопрос, как определить оптимальный параметр `min_df` для каждой статьи вместо использования среднего значения. В 2006 году на этот вопрос попытались ответить с помощью метода переходной точки (ТП) [132]. Техника TP взвешивает термины в зависимости от их расстояния до средней точки терминов (точка где-то между C и D на Рис. 3.11), которая несет основную информацию.

Говоря о результатах, достигнутых методом, предложенным в данной работе (Таблица 3.11), мы предлагаем пересмотреть методы экстрактивного

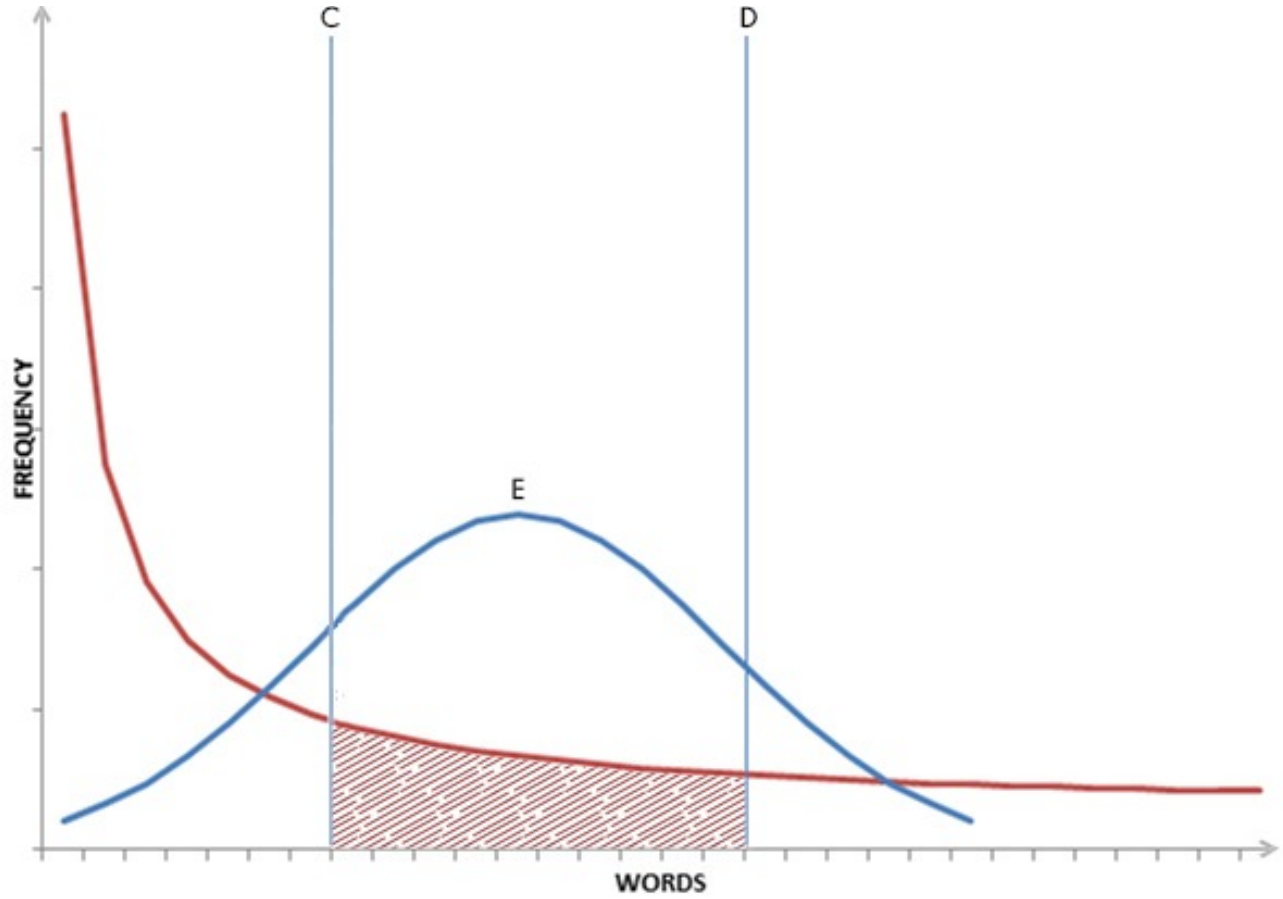


Рисунок 3.11 – Идея слов с максимальной разрешающей способностью, или слов, несущих наиболее ценную информацию (E), расположенных между словами с самой высокой (C) и самой низкой частотой (D).

суммирования. Поскольку верхняя граница в экстрактивном суммировании еще не достигнута, мы также считаем, что верхняя граница, которую мы обнаружили в нашей работе с помощью жадного алгоритма и VNS, не окончательная, а лишь первое приближение.

3.5.4 Вывод

В данной работе были продемонстрированы два метода определения наилучшей оценки ROUGE для экстрактивного суммирования на наборе данных arXive. В первом случае использовалась техника VNS, которая применялась в нашей предыдущей публикации [99], а во втором - жадный алгоритм. Оба подхода показали схожую производительность, но жадный подход занял значительно меньше времени. Исследование показало, что еще есть возможности для развития методов экстрактивного суммирования, так как наилучшая возможная оценка ROUGE-1 в среднем составляет 0,55, в то время как современные модели не превышают уровень 0,50.

Использование подхода жадного алгоритма побудило нас использовать модифицированный алгоритм экстрактивного суммирования, и он показал результаты, сравнимые с результатами современных моделей в наших тестах.

Таким образом, в целом можно сказать, что экстрактивные методы суммирования все еще имеют потенциал для развития, поддерживая мнение Себастьяна Рудера, когда он говорит, что важную роль играет не сложность метода, а правильная настройка гиперпараметров и предварительная обработка данных [133].

В рамках будущей работы мы планируем провести исследования в следующих областях:

1 Определение оптимального количества предложений, дающих наивысший балл ROUGE для каждого случая.
2 Продолжить исследования по оптимизации параметра `min_df`, чтобы определять его на ходу для каждой статьи для достижения лучших результатов вместо использования статистически наведенных средних значений.
3 Разработать метод улучшения жадного алгоритма в режиме поиска, чтобы избежать локальных минимумов, в которые склонны попадать жадные алгоритмы.
4 Использование стохастического алгоритма вместо перебора для поиска оптимального `min_df`.

3.6 Практическое применение

Алгоритмы суммирования могут применяться в каждом случае, когда нам нужен *короткий рассказ* в таких областях, как наука, бизнес и новостные СМИ.

3.6.1 Образование

В сфере образования применение алгоритмов обобщения довольно простое для сжатия учебного материала.

a) Автоматическое ведение записей

Автоматическое конспектирование (ANT) материала учебника или стенограммы лекции может быть сделано с использованием алгоритмов обобщения текста и поможет студентам сэкономить много времени. Современные платформы для онлайн обучения, такие как Microsoft Teams, например, предлагают функцию автоматической расшифровки записи лекции в сочетании с технологией *Speech-To-Text* (STT), так что студенты могут найти текст того, что было сказано профессором во время лекции, и посмотреть видео на этом месте еще раз. Также было бы здорово, если бы у Teams была возможность резюмировать стенограммы лекций.

Студенческие задания по чтению также требуют времени для конспектирования и автоматической обработки текста, чтобы представить студенту краткое содержание, удобное для чтения и запоминания того, что было написано в тексте. Но, конечно, технология может и, безусловно, будет злоупотребляться студентами, как всегда, когда они могут пропустить реальное задание по чтению и использовать только конспект текста, упуская глубокие детали учебного материала.

b) Mind-map generation

Кроме того, освободившись от ручного конспектирования, студенты могут больше сосредоточиться на усвоении самого учебного материала. Сами конспекты не являются целью процесса обучения, вместо этого студент должен понять, как связаны между собой изучаемые понятия, какие отношения существуют между изученными терминами, какие следствия вытекают из вновь приобретенных знаний и как связать их с уже известными. Именно здесь студент может сконцентрировать свои умственные силы, используя хороший вспомогательный инструмент для преобразования автоматически созданных заметок в *mind-map*s.

Определение 3.1. *Mind-map это очень популярная концепция диаграммы предметных связей, придуманная и активно рекламируемая Тони Бьюзаном в 1974 году. Диаграмма организована в виде верхней темы, помещенной в центр, к которой по иерархической схеме подключаются связанные с ней идеи, категории, термины и представления [2]. Диаграмма может*

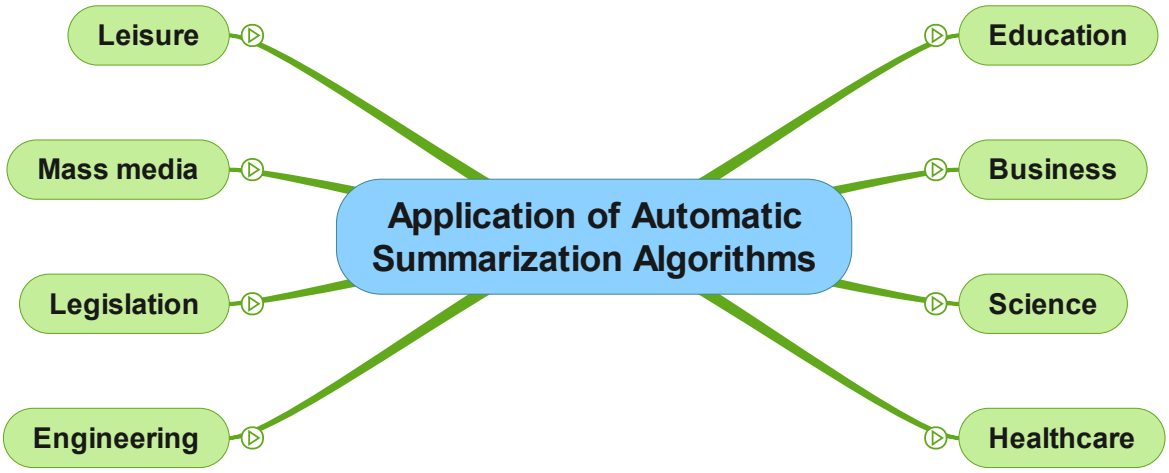


Рисунок 3.12 – Пример ментальной карты .

быть дополнена цитатами, изображениями, иконками и использованием цветов, чтобы помочь усвоению и запоминанию материала как взаимосвязанной части целого знания. Пример mind-map, нарисованной вручную в SimpleMind Application[8] *смотрите Рис. 3.12.*

c) Генерация шпаргалок

По аналогии с автоматическим конспектированием мы можем применить обобщение текста для получения сверхсжатого конспекта любого учебного материала для еще более быстрого повторения и повторения пройденных на уроках тем. Чистые таблицы обычно содержат уравнения, определения и теоремы, что делает их специальным назначением *экстрактное резюме*.

Disclaimer Здесь мы ни в коем случае не поощряем жульничество в процессе обучения, а используем термин *cheetsheet* исключительно для выявления краткого изложения текста с чрезвычайно высоким уровнем компрессии. Вместо этого мы призываем студентов строго следовать принципам академической этики и честности.

d) Генерация слайдов для презентаций (PSG)

Другим перспективным применением автоматического обобщения текста является создание презентационных слайдов из научной статьи (см. также Раздел 3.6.2 о применении обобщения текста в науке) или текста главы учебника. Процесс обобщения текста здесь расширен за счет создания структуры для текста резюме, чтобы разделить его для отдельных слайдов и генерировать заголовки слайдов. Более продвинутый подход может также включать такие функции, как *поиск изображений* и *перефразирование* для создания

[8]SimpleMind от ModelMaker Tools BV (Нидерланды)`https://simplemind.eu/`

слайдов с богатым содержанием [134].

Приложение может реально сэкономить время при подготовке к презентации научной статьи на конференции или научном семинаре. Оно также может быть полезно при подготовке к урокам для преподавателей для создания слайдов на основе материала учебника.

Набор данных, используемый [135], доступен на сайте Kaggle[9].

Аналогичным образом презентационные слайды могут быть созданы из широкого спектра деловых документов, таких как отчеты, бизнес-планы и инвестиционные предложения; см. Раздел 3.6.5для дальнейшего чтения о применении автоматического обобщения текста в бизнесе.

e) Генерация тестов

Другим полезным применением автоматического обобщения текста является *Automatic Test Generation* (*ATG*), где оно применяется для извлечения наиболее информативных предложений из текста. На основе извлеченных предложений существуют различные подходы для создания объективных и субъективных вопросов, таких как "*Вставьте пропущенное слово*" или "*Дайте определение ... своими словами*" [136].

Генерация ответов и оценка тестов также является предметом автоматизации в такого рода приложениях. Использование обобщения текста при составлении тестов помогает достичь большего соответствия тестовых вопросов изучаемому предмету и, таким образом, позволяет получить более четкое представление об успеваемости студентов.

f) Написание эссе

Утомительная задача *написание эссе* в старших классах и колледжах включает в себя чтение большого количества исходного материала в виде книг романов, философских работ, политических исследований, деловых или юридических дел, а затем выражение собственного мнения и уровня понимания предмета. Несмотря на то, что фактический поиск и чтение исходного материала имеют решающее значение в процессе обучения, часто случается так, что студенты упускают время и вынуждены писать эссе в очень сжатые сроки.

Поэтому приемлемым решением в такой ситуации может стать беглое прочтение автоматически созданных резюме исходных текстов и улавливание основных моментов, необходимых для написания эссе. Существует множество онлайн-ресурсов с аннотациями классической литературы, составленными людьми, и вот некоторые из них:

- **Gradesaver.com**[10], где представлены краткие изложения по английской и американской классической литературе, чтобы помочь студентам в их

[9] `https://www.kaggle.com/`

[10] `https://www.gradesaver.com/`

нелегкой работе. Gradesaver предоставляет комплексный материал для изучения каждой книги, включая краткое изложение, подробное изложение и анализ по главам, эссе, список тем, список персонажей, исторический контекст, биографию автора и тест.

- **Briefly.ru**[11], где представлены аннотации со степенью сжатия около 30% для всех основных переводов классической литературы и мировой классики на русский язык, а также их кинематографические версии. Отдельной интересной особенностью там является микрорезюме, выражающее основное содержание книги всего в паре предложений.

3.6.2 Наука

Модели автоматического суммирования текста могут применяться в сфере науки различными способами, как похожими на те, что используются в сферах образования и бизнеса, так и уникальными.

a) Подготовка литературного обзора

Любое научное исследование обычно начинается с обзора литературы по теме, чтобы увидеть, что уже было сделано другими исследователями в прошлом и какие результаты были достигнуты. Далее, на основе этого анализа исследователь может разработать свой собственный путь исследования по данной теме.

Автоматическое составление обзора литературы ($ALRG$) может быть выполнено с помощью:

- *Методы обобщения отдельных документов* для создания резюме для каждой отдельной научной статьи, выбранной исследователем.

- *Методы обобщения многодокументных текстов* для обобщения набора статей одним движением.

- *Многодокументное обобщение текста для методов ответа на вопросы*, чтобы автоматизировать поиск релевантных научных статей и их извлечение.

Также стоит отметить, что вывод ALRG требует *перефразирования* или *абстрактных методов обобщения*, чтобы избежать *плагиата*.

b) Создание реферата статьи

Автоматическое суммирование текста или ATS может быть применено для создания аннотации к только что написанной научной статье. Впервые это было сделано [33], когда авторы предоставили экстрактивные предложения в виде абстрактов для своей статьи под названием “A Trainable Document Summarizer”, демонстрируя работу своего ML-подхода в резюмировании.

[11] `https://briefly.ru/`

c) Мультимодальное обобщение

Multi-modal summarization (*MMS*) означает резюмирование источника в различных медиа, например, резюмирование текста в виде изображения или видео (см. Раздел a)), и наоборот, генерирование подписей к изображениям или текстовых резюме диаграмм. В науке MMS может служить следующим целям:

- Обобщить текст в виде mind-map (см. Раздел b)), диаграммы, иерархии (онтология, организационная карта и т.д.).

- Создание текстового резюме для таблиц, диаграмм, графиков и других структурированных данных.

- Создание резюме для аудио- и видеозаписей, которое может сопровождаться запросом интересующей информации, что требует решения задач распознавания, идентификации и классификации аудио/визуальных сигналов.

d) Популяризация науки

Популяризация научных концепций для неподготовленной или *lay*аудитории - это задача перевода сложного научного языка на обычный, понятный среднему человеку. Эта задача важна для повышения общего уровня образования и осведомленности людей о современных научных достижениях, которые могут быть распространены через СМИ и научно-популярные книги.

В 2020 году *LaySumm*[12] конкурс проводился в рамках *EMNLP* Конференции *SDP* Workshop. Задание по резюмированию LaySumm рассматривает автоматизацию генерации популярного изложения научных документов. Задача заключалась в автоматической генерации информативных резюме, понятных и интересных неспециалистам.

Определение 3.2. *Популярное изложение - это текстовое резюме научной статьи, предназначенное для нетехнической или светской аудитории. Обычно оно составляется либо автором, либо журналистом. Точнее, в кратком изложении в 70-100 словах объясняется цель, объем и общее воздействие научной статьи, избегая использования технического жаргона. Пример популярного изложения приведен в Таблица 3.12.*

:

Подведение итогов по сравнению с оригиналом

e) Поддержание исследователей в курсе их тем

Российский институт научной и технической информации при Российской академии наук (*RAS*) с 1952 года издает журнал "Реферативный журнал"[15],

[12] https://ornlcda.github.io/SDProc/sharedtasks.html\#laysumm

[15] http://www.viniti.ru/products/abstract-journal

Заглавие статьи:	Оптимальное распределение пропускной способности при случайном спросе пассажиров в сети высокоскоростных железных дорог[13]
Article id:	10.1016/j.engappai.2019.103363[14]
Оригинальная аннотация:	Распределение пропускной способности является практически значимым фактором, влияющим на качество расписания движения поездов в железнодорожных перевозках, особенно в условиях колебания спроса пассажиров. Целью данной работы является детальное описание структуры и характеристик проблемы распределения пропускной способности при случайном спросе в сети высокоскоростных железных дорог. Предлагается двухэтапная модель стохастического целочисленного программирования для получения решений по распределению пропускной способности для удовлетворения случайных колебаний спроса пассажиров в ежедневной работе, которая учитывает неопределенность спроса и не делает никаких предположений о структуре железнодорожной сети и распределении спроса пассажиров. Учитывая сложность решения этой задачи, мы предлагаем схему решения, включающую эвристический алгоритм, основанный на поиске табу, для получения почти оптимального решения и стратегии для получения эффективного расписания и корректировки формирования поездов. Наконец, два набора примеров, в которых образец железнодорожной сети с 5 станциями и данные сети высокоскоростных железных дорог Пекин-Шанхай, используются в качестве экспериментальной среды, чтобы проиллюстрировать производительность и эффективность предложенных методов.
Популярное изложение:	Как тактические планы сложных железнодорожных операций, расписание поездов программируется и обновляется каждый год или каждый сезон из-за значительного изменения спроса пассажиров. С точки зрения оптимизации, целью данной работы является исследование подробного описания и оптимальных методов (двухэтапная модель стохастического целочисленного программирования и соответствующий эвристический алгоритм) для эффективного получения почти оптимального расписания и распределения пассажиров по вместимости поездов в условиях колебаний ежедневного спроса пассажиров. С помощью реализации на высокоскоростной железнодорожной сети Пекин-Шанхай в Китае мы проверяем производительность и эффективность предложенных методов.

Таблица 3.12 – Пример популярного изложения.

который является периодическим научно-информационным изданием, публикующим рефераты, аннотации и библиографические описания российских

и мировых публикаций в области естественных, точных и технических наук, экономики и медицины. Аудитория журнала - ученые, которые хотят быть в курсе достижений в интересующих их научных темах.

Редакторы регулярно собирают, при необходимости переводят и обобщают статьи из:

- Серийные публикации.
- Книги и их главы.
- Материалы научных конференций.
- Картографические и корпоративные издания.
- Зарубежные диссертации.
- Патентные и нормативно-технические документы.
- Депонированные научные работы.

И отражает ежегодно более 800К документов, 40% из которых происходят из российских источников. Журнал состоит из 24 сводных томов, включающих 182 номера, на каждый из которых можно подписаться отдельно, и 39 отдельных номеров.

Опубликованные рефераты классифицируются по каталогу РАН и обычно содержат следующие поля:

- Abstract серийный номер.
- Название на русском языке.
- Название на языке оригинала.
- Авторы.
- Сокращенное название издания.
- Год публикации.
- Объем.
- Release.
- Страницы статьи.
- Язык источника.
- Abstract.
- Адрес первого автора.
- Библиография.

Пример публикации реферата в журнале 97.03-04М5.961. **Гепатит С как опасность для работников здравоохранения. Инфекция гепатита С как профессиональная опасность для работников здравоохранения** / Prakash Charu, Bhatia Rajesh, Kumari S., Verghese T., Datta KK// J. Commun. diseases. - 1995.-27, No. 4. - С. 272-274. -Английский Были исследованы сыворотки 57 мед. работников из больниц Дели, которые не контактировали непосредственно с лицами из групп высокого риска по гепатиту С (подвергались диализу, трансплантации органов, многократным переливаниям крови). Маркеры гепатита В не были обнаружены, антитела к

вирусу гепатита С были обнаружены методом ИФА в 4 (7%) образцах. Этот показатель был выше, чем при заражении от укола иглой или при семейном контакте с пациентами с хроническим гепатитом С. Индия, Nat. Inst. Of Communicable diseases 22, Sham Nath Marg, Delhi-110054. Библ. 2

f) Суммаризация как инструмент исследования в ОЕЯ

Любопытно, что автоматическое резюмирование текста, будучи само по себе задачей НЛП, также используется как инструмент в исследовании других задач НЛП. Например, в задаче *кластеризации* документов мы можем использовать резюме документов для интерпретации кластеров, почему документы сгруппированы вместе, путем поиска общего содержания в их резюме.

Или в случае работы с *Recurrent Neural Networks* (*RNN*), которые, как известно, имеют проблемы с использованием памяти и поэтому с трудом обрабатывают длинные документы. Мы можем подавать им резюме, а не целые документы.

Другой пример использования ATS в качестве инструмента для решения другой задачи NLP - резюмирование комментариев для *opinion mining* и *sentiment analysis* [137]; см. Раздел 3.6.6.

3.6.3 Инженерия

В инженерном деле автоматическое резюмирование текста может быть использовано для сокращения объема требуемого чтения в технической документации, патентной документации [138], системных журналах, тикетах системы поддержки и информации об отзывах пользователей.

3.6.4 Здравоохранение

Существует большой потенциал для применения моделей ATS в здравоохранении, начиная с обобщения истории болезни пациента и заканчивая помощью в проведении исследований для фармацевтических компаний.

a) Краткое изложение истории болезни пациента

Личные медицинские карты содержат временную информацию о прививках пациентов, заболеваниях, травмах, информацию и контактные данные лечащего врача, названия медицинских учреждений, результаты анализов и обследований. Поэтому анализ такой информации может сыграть решающую роль в эффективности и успешности назначенного лечения.

Была разработана конвейерная система автоматического обобщения информации о пациенте для использования с *Электронные медицинские карты* (*EMR*) на португальском языке [139], а также предприняты некоторые попытки создания естественно-языковых резюме по сигналам медицинского оборудования в отделениях интенсивной терапии [140].

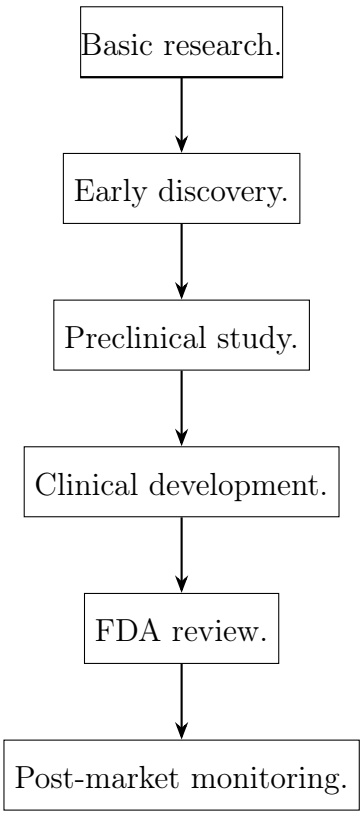


Рисунок 3.13 – Commercial drug development process [141].

b) Фармацевтические исследования

На самом деле деятельность фармацевтических компаний очень строго регулируется, поскольку их влияние на общество через поставку лекарств является критическим, и регулирующие органы хотят любыми средствами избежать потенциального злоупотребления и вреда для людей. Таким образом, при коммерческой разработке лекарств любое новое лекарство должно пройти через строгий цикл фармацевтических исследований и испытаний, прежде чем попасть на полки местных аптек. Процесс коммерческой разработки лекарств показан на рисунке Рис. 3.13.

На этапе фундаментальных исследований в процессе разработки коммерческих лекарств необходимо просмотреть большое количество медицинской литературы, и ручной процесс может занять несколько лет. Поэтому модели ATS могут помочь здесь ускорить процесс рецензирования, как это сделала компания *SemanticHub*[16], которая предоставляет такие услуги для компаний *Big Pharma*.

SemanticHub SemanticHub[17] основана в 2015 году Ефименко Ириной[18], Хорошевским Владимиром[19] и Недельским Виталием. Компания предоставляет

[16] https://www.semantic-hub.com/

[17] Semantic Hub LLC (Россия), Semantic Hub SARL (Лозанна), Wuxi Shengmanhe Information Technology LLC

[18] https://scholar.google.com/citations?user=BQd630AAAAAJ

[19] https://scholar.google.com/citations?user=g7kYtbUAAAAJ

услуги по идентификации кандидатов в лекарственные препараты с помощью искусственного интеллекта, собирая и обрабатывая до миллиона документов по нужной теме (научные статьи, патенты, отчеты о клинических исследованиях, пресс-релизы и другие) и проводя их глубокий семантический анализ, чтобы составить отчет для фармацевтической компании, показывающий оценку кандидатов в лекарственные препараты с учетом факторов риска и конкурентных преимуществ [142].

SemanticHub ускоряет оценку прототипов перспективных лекарств и снижает риск неудачи из-за отсутствия важной информации на 30%.

3.6.5 Бизнес

a) Подведение итогов деловых лонгридов

В бизнесе, чтобы сэкономить время менеджеров высшего и среднего звена, позволив им пропустить чтение большого количества документов, мы можем применить ATS для создания исполнительных резюме таких давно прочитанных документов, как:

- Годовые отчеты.
- Бизнес-планы.
- Инвестиционные предложения.
- Отраслевые отчеты.
- Маркетинговые отчеты.
- Юридические документы:
- Законы.
- Контракты.
- Счета.
- Решения суда.
- Изменения в юридических документах.

b) Генерация протокола собрания

Аналогично применению *ANT* в образовании, рассмотренному в Раздел a), мы также можем автоматически генерировать *Minutes of Meeting* (*MOM*) для ведения протокола деловой встречи и отслеживания отчетности участников.

c) Авторефераты ориентированные на запрос

Если у компании есть собственная *Корпоративная база знаний* (*КБЗ*) и альтернативный или одновременный доступ к любым внешним базам знаний, можно построить систему для создания ориентированных на запрос многодокументных резюме. Например, у нас есть юридический вопрос, касающийся нашего бизнеса или его отдельных аспектов, тогда мы можем запросить базу юридических знаний, содержащую правовые акты, комментарии и другие полезные документы, и обобщить наиболее релевантные документы для со-

здания отчета.

Этот же подход может быть использован для применения *многодокументного обобщения для ответов на вопросы* с помощью любой *поисковой системы.*

Определение 3.3. *База знаний (KB) - это система компьютерных технологий, используемая для хранения сложной структурированной и неструктурированной информации, относящейся к определенной области знаний.*

d) Оптимизация контекстной рекламы

В веб-рекламе важными целями являются, с одной стороны, увеличение доходов коммерческих компаний, а с другой - повышение удобства пользования сайтом. Эта цель достигается путем размещения рекламы на веб-странице относительно ее содержания и называется *Контекстная реклама* (*CA*). Таким образом, реклама лучше нацелена на целевую аудиторию, а пользователи меньше раздражаются от рекламы.

Автоматическое обобщение текста используется в CA для обобщения содержания веб-страницы и использования полученного обобщения для классификации содержания, чтобы сопоставить его с подходящей рекламой для размещения [143].

3.6.6 Масс-медиа и социальные сети

В СМИ применение CAP происходит в таких формах, как:

- *News Snippets Generation* (*NSG*) - это автоматическое создание короткого *показательного резюме*, чтобы мотивировать человека прочитать всю статью.

- *Digest Article Compilation* (*DAC*), являющийся продуктом *multi-document summarization* статей по выбранной теме.

- *Article Comments Summarization* (*ACS*) для публичного *opinion mining* и *sentiment analysis* для интересующей вас темы события или новостей. Здесь ATS снова используется как инструмент для работы над другими задачами НЛП; см. Раздел f).

- *Social Network Post Contextualization*, (*SNPC*) которая заключается в объяснении смысла сообщения с учетом предшествующих или последующих связанных сообщений с помощью методов многодокументного обобщения. Эта проблема особенно важна для постов Twitter, поскольку он предназначен для передачи коротких сообщений, и зависимость от информации в предыдущих твитах высока.

Определение 3.4. *Sentiment Analysis (SA) или Opinion Mining (OM), это Natural Language Processing (NLP) задача по выявлению эмоционального отношения (например, positive, negative или neutral), выраженного автором текста относительно темы или предмета. SA или OM обычно решает-*

ся как контролируемая классификация или неконтролируемая задача кластеризации с использованием искусственного интеллекта (AI), машинного обучения (ML) и добыча данных.

3.6.7 Развлечения

a) Суммаризирование видеозаписей

Кадры фильма - это хороший маркетинговый инструмент и пример *показательное резюме видео*, подстрекающий людей заплатить за просмотр всего фильма. Некоторые люди любят смотреть кадры из фильмов больше, чем сам фильм, и действительно, кадры часто содержат самые сочные сцены из фильма, чтобы вызвать желание посмотреть его полностью[20].

Более того, иногда человек смотрит фильм на высокой скорости, потому что он скучный и он хочет сэкономить время. Или даже если фильм действительно интересный, но длинный, и у вас нет свободных 3 часов для его просмотра, тогда возникает необходимость в коротком *видео резюме*, который длится всего 15-20 минут.

Видеоконспекты также могут быть адаптированы к информационным потребностям и предпочтениям пользователя или целевой аудитории. Например, мы можем создать видеорезюме, содержащее только батальные сцены из исторического фильма или нескольких фильмов.

Одним из нескольких способов создания видео резюме является подход к обобщению текста субтитров, реализованный в веб-приложении *VideoMash*; см. Раздел 2.5.3.Существует также множество различных и более сложных подходов к обобщению видео, использующих фреймворки DeepLearning и Reinforcement Learning[21].

Недавней разработкой в этой области является *VidPress* компании *Baidu*, которая генерирует фильм из короткого сюжетного текста, когда *Generative Adversarial Network* (*GAN*) генерирует фильм по короткому текстовому вводу [144].Разработка такой инновационной модели GAN стала возможной благодаря обилию ресурсов образцов видео, которыми располагает компания Baidu. И это действительно первый шаг к будущему автоматическому созданию фильмов на основе ввода текстовых сценариев, который полностью разрывает воображение.

b) Суммаризация аудиозаписей

Аналогично видеоконспектам, мы можем создавать конспекты аудиокниг, подкастов, интервью и всех других видов аудиозаписей. Это приложение переводит резюме текстов в аудиоформат, поэтому мы можем слушать резюме книг, например, за рулем, на улице или в спортзале.

[20] `https://www.frameaddict.com/`

[21] `https://paperswithcode.com/task/video-summarization`

Хорошим примером сбора резюме аудиокниг является BestBookBits.com[22], где вы можете прослушать сгенерированные человеком резюме книг, так что вы можете притвориться, что прочитали их целиком.

ATS может автоматизировать процесс создания аудио резюме либо путем резюмирования транскрипта аудиозаписи (для этого используются технологии *STT*) и последующего сокращения записи в соответствии с временными метками резюмированного транскрипта, либо путем резюмирования исходного текста и последующего применения технологии *Text to Speech* (*TTS*) для создания выходного результата.

[22] `https://bestbookbits.com/`

4 ЗАКЛЮЧЕНИЕ

В рамках данной работы была произведена оценка верхнего уровня качества авторефератов достижимого при помощи Экстрактивных методов автореферирования текстов (см. Глава 3.4) и выявлено достаточное пространство для развития методов автоматического реферирования в сравнением с их текущим уровнем (см. Глава 2.3).

В результате, нами был разработан метод экстрактивного автоматического реферирования текстов GreedSum, на которую было получено авторское свидетельство № 22446 от 15.12.2021. В тестах на текстах научных статей из корпуса данных arXive и PubMed, метод показал качество рефератов на уровне современных моделей использующих нейронные сети и проигрывая сравнения только модели $PEGASUS_large$ (см. Таблица 3.11).

Метрика	Ср. знач.	Ниж.	Верх.
ROUGE-1	0.429332	0.428249	0.430414
ROUGE-2	0.120340	0.119653	0.121027

Таблица 4.1 – Средние значения метрик ROUGE-1/2 для авторефератов полученных при помощи метода GreedSum на корпусе статей arXive, нижнее и верхнее значения 95% доверительного интервала.

Резюме:

- Произведена оценка наилучшего возможного балла ROUGE для методов Extractive Text Summarization (ETS).

- Применены подходы:

1 Подход с переменным поиском соседства (VNS).
2 Жадный алгоритм.
3 Генетический алгоритм.

- Полученные результаты:

1 Подход с использованием жадного алгоритма — для сходимости требуется меньше времени, чем для VNS.
2 Наилучшая возможная оценка ROUGE составляет 0,59, а самые современные модели дают всего 0,46.
3 Следовательно, есть еще место для развития ETS .

- GreedSum : жадный алгоритм автоматического реферирования текста (ATS).

1 Результаты сопоставимы с результатами современных моделей.
2 Методы ETS все еще имеют потенциал для развития.
3 Ключевым фактором является настройка гиперпараметров.

Основные результаты работы опубликованы в работах [63].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 *Blei, David M.* Latent Dirichlet allocation / David M. Blei, Andrew Y. Ng, Michael I. Jordan // *Journal of Machine Learning Research.* — 2003. — Vol. 3.

2 *Buzan, Tony.* Mind map mastery the complete guide to learning and using the most powerful thinking tool in the universe. — 2018.

3 Topic-Aware Sentiment Analysis of News Articles / Iskander Akhmetov, Iskander Akhmetov, Alexander Gelbukh, Rustam Mussabayev // *Computación y Sistemas.* — 2022. — 3. — Vol. 26. https://cys.cic.ipn.mx/ojs/index.php/CyS/article/view/4179.

4 *Luhn, H. P.* The Automatic Creation of Literature Abstracts / H. P. Luhn // *IBM Journal of Research and Development.* — 1958. — Vol. 2, no. 2. — Pp. 159–165.

5 *Verma, Pradeepika.* A Comparative Analysis on Hindi and English Extractive Text Summarization / Pradeepika Verma, Sukomal Pal, Hari Om // *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* — 2019. — May. — Vol. 18, no. 3. — 39 pp. https://doi.org/10.1145/3308754.

6 *Parker, Robert.* English gigaword fifth edition, linguistic data consortium. — 2011.

7 *Bhatia, Neelima.* Automatic text summarization and it's methods - A review / Neelima Bhatia, Arunima Jaiswal // Proceedings of the 2016 6th International Conference - Cloud System and Big Data Engineering, Confluence 2016. — 2016.

8 *El-Refaiy, Ahmed.* Review of recent techniques for extractive text summarization / Ahmed El-Refaiy, A.R. Abas, I. Elhenawy // *Journal of Theoretical and Applied Information Technology.* — 2018. — 12. — Vol. 96. — Pp. 7739–7759.

9 Multi-Task Learning for Abstractive and Extractive Summarization / Yangbin Chen, Yun Ma, Xudong Mao, Qing Li // *Data Science and Engineering.* — 2019.

10 *Gupta, Som.* Abstractive Summarization: An Overview of the State of the Art / Som Gupta, S.K Gupta // *Expert Systems with Applications.* — 2018. — 12. — Vol. 121.

11 *Khan, Rahim.* Extractive based Text Summarization Using KMeans and TF-IDF / Rahim Khan, Yurong Qian, Sajid Naeem // *International Journal of Information Engineering and Electronic Business.* — 2019. — 05. — Vol. 11. — Pp. 33–44.

12 Sentence Relations for Extractive Summarization with Deep Neural Networks / Pengjie Ren, Zhumin Chen, Zhaochun Ren et al. // *ACM Trans. Inf. Syst.* — 2018. — Apr.. — Vol. 36, no. 4. — 32 pp. https://doi.org/10.1145/3200864.

13 *Van Lierde, Hadrien.* Learning with fuzzy hypergraphs: A topical approach to query-oriented text summarization / Hadrien Van Lierde, Tommy W.S. Chow // *Information Sciences.* — 2019. — Sep. — Vol. 496. — P. 212–224. http://dx.doi.org/10.1016/j.ins.2019.05.020.

14 *Carbonell, Jaime G.* Vision statement to guide research in Question & Answering (Q&A) and Text Summarization. — 2000. https://www.researchgate.net/publication/228600187.

15 *Luhn, H. P.* A Statistical Approach to Mechanized Encoding and Searching of Literary Information / H. P. Luhn // *IBM Journal of Research and Development.* — 1957. — Vol. 1, no. 4. — Pp. 309–317.

16 *Luhn, Hans Peter.* COMPUTER FOR VERIFYING NUMBERS. — 1960. — Aug. https://patents.google.com/patent/US2950048?oq=US2950048A.

17 *Akhmetov, Iskander.* The arXive dataset extract with high ROUGE score summaries generated by 5 different methods V1. — 2021.

18 *Allahyari, Mehdi.* Text Summarization Techniques: A Brief Survey. — 2017.

19 *Nazari, Narges.* A survey on Automatic Text Summarization / Narges Nazari, Mohammad Amin Mahdavi // *Journal of AI and Data Mining.* — 2019. — Vol. 7. — Pp. 121–135.

20 *Rajasekaran, Abirami.* Review on automatic text summarization / Abirami Rajasekaran, R. Varalakshmi // *International Journal of Engineering and Technology(UAE).* — 2018. — 06. — Vol. 7. — Pp. 456–460.

21 *Saziyabegum, Saiyed.* REVIEW ON TEXT SUMMARIZATION EVALUATION METHODS / Saiyed Saziyabegum // *Indian Journal of Computer Science and Engineering.* — 2017. — Vol. 8.

22 *Gusenbauer, Michael.* Google Scholar to Overshadow Them All? Comparing the Sizes of 12 Academic Search Engines and Bibliographic Databases / Michael Gusenbauer // *Scientometrics.* — 2019. — 01. — Vol. 118. — P. 177–214.

23 *van Eck, Nees Jan.* Software survey: VOSviewer, a computer program for bibliometric mapping / Nees Jan van Eck, Ludo Waltman // *Scientometrics.* — 2010. — Vol. 84, no. 2.

24 *Lloret, Elena.* Text summarisation in progress: A literature review / Elena Lloret, Manuel Sanz // *Artif. Intell. Rev.* — 2012. — 04. — Vol. 37. — Pp. 1–41.

25 *Hu, Minqing.* Mining and Summarizing Customer Reviews / Minqing Hu, Bing Liu // Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — KDD '04. — New York, NY, USA: Association for Computing Machinery, 2004. — P. 168–177. https://doi.org/10.1145/1014052.1014073.

26 *Erkan, G.* LexRank: Graph-based Lexical Centrality as Salience in Text

Summarization / G. Erkan, D. R. Radev // *Journal of Artificial Intelligence Research*. — 2004. — Dec. — Vol. 22. — P. 457–479. http://dx.doi.org/10.1613/jair.1523.

27 Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs / Qiaozhu Mei, Xu Ling, Matthew Wondra et al. // Proceedings of the 16th International Conference on World Wide Web. — WWW '07. — New York, NY, USA: Association for Computing Machinery, 2007. — P. 171–180. https://doi.org/10.1145/1242572.1242596.

28 Searching for Effective Neural Extractive Summarization: What Works and What's Next / Ming Zhong, Pengfei Liu, Danqing Wang et al. // *CoRR*. — 2019. — Vol. abs/1907.03491. http://arxiv.org/abs/1907.03491.

29 *Wang, Hongning*. Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach / Hongning Wang, Yue Lu, Chengxiang Zhai // Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. — KDD '10. — New York, NY, USA: Association for Computing Machinery, 2010. — P. 783–792. https://doi.org/10.1145/1835804.1835903.

30 *Li, Wei*. Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations / Wei Li, Andrew McCallum // Proceedings of the 23rd International Conference on Machine Learning. — ICML '06. — New York, NY, USA: Association for Computing Machinery, 2006. — P. 577–584. https://doi.org/10.1145/1143844.1143917.

31 *See, Abigail*. Get To The Point: Summarization with Pointer-Generator Networks. — 2017.

32 *Barzilay, Regina*. Sentence Fusion for Multidocument News Summarization / Regina Barzilay, Kathleen R. McKeown // *Comput. Linguist.* — 2005. — sep. — Vol. 31, no. 3. — P. 297–328.

33 *Kupiec, Julian*. Trainable document summarizer / Julian Kupiec, Jan Pedersen, Francine Chen // SIGIR Forum (ACM Special Interest Group on Information Retrieval). — 1995.

34 *Rush, Alexander M.* A Neural Attention Model for Abstractive Sentence Summarization / Alexander M. Rush, Sumit Chopra, Jason Weston // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. — Lisbon, Portugal: Association for Computational Linguistics, 2015. — Sep.. — Pp. 379–389. https://www.aclweb.org/anthology/D15-1044.

35 *Mihalcea, Rada*. TextRank: Bringing Order into Text / Rada Mihalcea, Paul Tarau // Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004. — Barcelona, Spain: ACL, 2004. — Pp. 404–411. https://aclanthology.org/W04-

3252/.

36 *DeJong, Gerald.* Prediction and substantiation: A new approach to natural language processing / Gerald DeJong // *Cognitive Science.* — 1979.

37 *Conroy, John M.* Text summarization via hidden Markov models / John M. Conroy, Dianne P. O'Leary // SIGIR '01. — 2001.

38 *Carbonell, Jaime.* The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries / Jaime Carbonell, Jade Stewart // *SIGIR Forum (ACM Special Interest Group on Information Retrieval).* — 1999. — 06.

39 *Hearst, Marti A.* TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages / Marti A. Hearst // *Comput. Linguist.* — 1997. — mar. — Vol. 23, no. 1. — P. 33–64.

40 *Zhuang, Li.* Movie Review Mining and Summarization / Li Zhuang, Feng Jing, Xiao-Yan Zhu // Proceedings of the 15th ACM International Conference on Information and Knowledge Management. — CIKM '06. — New York, NY, USA: Association for Computing Machinery, 2006. — P. 43–50. https://doi.org/10.1145/1183614.1183625.

41 *Turney, Peter.* Learning Algorithms for Keyphrase Extraction / Peter Turney // *Inf. Retr.* — 2000. — 05. — Vol. 2. — Pp. 303–336.

42 *Gong, Yihong.* Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis / Yihong Gong, Xin Liu // Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — SIGIR '01. — New York, NY, USA: Association for Computing Machinery, 2001. — P. 19–25. https://doi.org/10.1145/383952.383955.

43 *Gu, Jiatao.* Incorporating Copying Mechanism in Sequence-to-Sequence Learning. — 2016.

44 *Sanderson, Mark.* Deriving Concept Hierarchies from Text / Mark Sanderson, Bruce Croft // Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — SIGIR '99. — New York, NY, USA: Association for Computing Machinery, 1999. — P. 206–213. https://doi.org/10.1145/312624.312679.

45 Summarizing Text Documents: Sentence Selection and Evaluation Metrics / Jade Goldstein, Mark Kantrowitz, Vibhu Mittal, Jaime Carbonell // Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. — SIGIR '99. — New York, NY, USA: Association for Computing Machinery, 1999. — P. 121–128. https://doi.org/10.1145/312624.312665.

46 *Rush, Alexander M.* A Neural Attention Model for Abstractive Sentence Summarization. — 2015.

47 Abstractive text summarization using sequence-to-sequence RNNs and beyond / Ramesh Nallapati, Bowen Zhou, Cicero dos Santos et al. // CoNLL 2016 - 20th SIGNLL Conference on Computational Natural Language Learning, Proceedings. — 2016.

48 *Tan, Jiwei.* Abstractive Document Summarization with a Graph-Based Attentional Neural Model / Jiwei Tan, Xiaojun Wan, Jianguo Xiao // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Vancouver, Canada: Association for Computational Linguistics, 2017. — Jul.. — Pp. 1171–1181. https://aclanthology.org/P17-1108.

49 *Gehrmann, Sebastian.* Bottom-up abstractive summarization / Sebastian Gehrmann, Yuntian Deng, Alexander M. Rush // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018. — 2020.

50 *Celikyilmaz, Asli.* Deep Communicating Agents for Abstractive Summarization. — 2018.

51 *Liu, Fei.* Toward Abstractive Summarization Using Semantic Representations. — 2018.

52 *Li, Piji.* Deep Recurrent Generative Decoder for Abstractive Text Summarization. — 2017.

53 *Song, Shengli.* Abstractive text summarization using LSTM-CNN based deep learning / Shengli Song, Haitao Huang, Tongxiao Ruan // *Multimedia Tools and Applications.* — 2018. — Vol. 78. — Pp. 857–875.

54 *Khan, Atif.* A framework for multi-document abstractive summarization based on semantic role labelling / Atif Khan, Naomie Salim, Yogan Jaya Kumar // *Applied Soft Computing.* — 2015. — 02. — Vol. 30.

55 *Zhang, Jingqing.* PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. — 2019.

56 *Liu, Yang.* Text summarization with pretrained encoders / Yang Liu, Mirella Lapata // EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference. — 2020.

57 *Lloret, Elena.* The challenging task of summary evaluation: an overview / Elena Lloret, Laura Plaza, Ahmet Aker // *Language Resources and Evaluation.* — 2018. — Vol. 52, no. 1. — Pp. 101–148. http://arxiv.org/abs/2002.07767.

58 The PageRank Citation Ranking: Bringing Order to the Web.: Technical Report 1999-66 / Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd: Stanford InfoLab, 1999. — November. — Previous number = SIDL-WP-1999-0120. http://ilpubs.stanford.edu:8090/422/.

59 *Nenkova, Ani.* The impact of frequency on summarization / Ani Nenkova,

Lucy Vanderwende // *Msr-Tr-2005*. — 2005. — 01.

60 *See, Abigail*. Get to the point: Summarization with pointer-generator networks / Abigail See, Peter J. Liu, Christopher D. Manning // ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers). — Vol. 1. — 2017. — Pp. 1073–1083.

61 A discourse-aware attention model for abstractive summarization of long documents / Arman Cohan, Franck Dernoncourt, Doo Soon Kim et al. // NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference. — 2018.

62 *Lin, Chin-Yew*. ROUGE: A Package for Automatic Evaluation of Summaries / Chin-Yew Lin // Text Summarization Branches Out. — Barcelona, Spain: Association for Computational Linguistics, 2004. — Jul.. — Pp. 74–81. https://aclanthology.org/W04-1013.

63 *Akhmetov, Iskander*. Greedy Optimization Method for Extractive Summarization of Scientific Articles / Iskander Akhmetov, Alexander Gelbukh, Rustam Mussabayev // *IEEE Access*. — 2021. — Pp. 1–1.

64 The PageRank Citation Ranking: Bringing Order to the Web.: Technical Report 1999-66 / Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd: Stanford InfoLab, 1999. — November. — Previous number = SIDL-WP-1999-0120. http://ilpubs.stanford.edu:8090/422/.

65 A revised algorithm for latent semantic analysis / Xiangen Hu, Zhiqiang Cai, M Louwerse at al. // IJCAI'03 Proceedings of the 18th International Joint Conference on Artificial Intelligence. — Acapulco, Mexico: Morgan Kaufman Publishers, 2003. — Pp. 1489–1491. — 18th International Joint Conference of Artificial Intelligence , IJCAI'03 ; Conference date: 09-08-2003 Through 15-08-2003.

66 *Gliozzo, Alfio*. Domain Kernels for Word Sense Disambiguation / Alfio Gliozzo, Claudio Giuliano, Carlo Strapparava // Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05). — Ann Arbor, Michigan: Association for Computational Linguistics, 2005. — Jun.. — Pp. 403–410. https://aclanthology.org/P05-1050.

67 *Haghighi, Aria*. Exploring Content Models for Multi-Document Summarization / Aria Haghighi, Lucy Vanderwende // Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. — Boulder, Colorado: Association for Computational Linguistics, 2009. — Jun.. — Pp. 362–370. https://aclanthology.org/N09-1041.

68 *Kullback, S.* On Information and Sufficiency / S. Kullback, R. A. Leibler // *The Annals of Mathematical Statistics*. — 1951. — Vol. 22, no. 1. — Pp. 79–86.

http://www.jstor.org/stable/2236703.

69 *Zhang, Jingqing.* PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. — 2020.

70 Big bird: Transformers for longer sequences / Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey et al. // *Advances in Neural Information Processing Systems.* — 2020. — Vol. 33.

71 *Raffel, Colin.* Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. — 2020.

72 *Vaswani, Ashish.* Attention Is All You Need. — 2017.

73 BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension / Mike Lewis, Yinhan Liu, Naman Goyal et al. // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. — Online: Association for Computational Linguistics, 2020. — Jul.. — Pp. 7871–7880. https://aclanthology.org/2020.acl-main.703.

74 *Rohde, Tobias.* Hierarchical Learning for Generation with Long Source Sequences. — 2021.

75 Language Models are Unsupervised Multitask Learners, Enhanced Reader / Radford Alec, Wu Jeffrey, Child Rewon et al. // *OpenAI Blog.* — 2019. — Vol. 1.

76 *Brown, Tom B.* Language Models are Few-Shot Learners. — 2020.

77 *Liu, Yixin.* SimCLS: A Simple Framework for Contrastive Learning of Abstractive Summarization. — 2021.

78 *Dong, Li.* Unified Language Model Pre-training for Natural Language Understanding and Generation. — 2019.

79 Quantifying the limits and success of extractive summarization systems across domains / H. Ceylan, R. Mihalcea, U. Özertem at al. // NAACL HLT 2010 - Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Main Conference. — 2010. — Pp. 903–911. www.scopus.com.

80 How far we can go with extractive text summarization? Heuristic methods to obtain near upper bounds / W.M. Wang, Z. Li, J.W. Wang, Z.H. Zheng // *Expert Systems with Applications.* — 2017. — Vol. 90. — Pp. 439–463. https://www.sciencedirect.com/science/article/pii/S0957417417305869.

81 *Hansen, Pierre.* J-Means: a new local search heuristic for minimum sum of squares clustering / Pierre Hansen, Nenad Mladenović // *Pattern Recognition.* — 2001. — Vol. 34, no. 2. — Pp. 405–413.

82 *Sala, Ignacio.* The best applications to summarize texts with your mobile. — 2021. — Jul. https://androidguias.com/en/summarize-texts/.

83 *Aswanth, Anil.* Video summarizer made easy using NLP. — 2019.

— Jan. https://medium.com/@aswanthkanil/video-summarizer-made-easy-using-nlp-af0afdea49b5.

84 *Hinton, Geoffrey E.* A Fast Learning Algorithm for Deep Belief Nets / Geoffrey E. Hinton, Simon Osindero, Yee-Whye Teh // *Neural Computation.* — 2006. — 07. — Vol. 18, no. 7. — Pp. 1527–1554. https://doi.org/10.1162/neco.2006.18.7.1527.

85 *Sharma, Eva.* BigPatent: A large-scale dataset for abstractive and coherent summarization / Eva Sharma, Chen Li, Lu Wang // ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference. — 2020.

86 *Rush, Alexander M.* A neural attention model for sentence summarization / Alexander M. Rush, Sumit Chopra, Jason Weston // Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing. — 2015.

87 *Narayan, Shashi.* Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization / Shashi Narayan, Shay B. Cohen, Mirella Lapata // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018. — 2020.

88 *Grusky, Max.* NEWSROOM: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies / Max Grusky, Mor Naaman, Yoav Artzi // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. — New Orleans, Louisiana: Association for Computational Linguistics, 2018. — June. — Pp. 708–719. http://aclweb.org/anthology/N18-1065.

89 *Sandhaus, Evan.* The new york times annotated corpus / Evan Sandhaus // *Linguistic Data Consortium, Philadelphia.* — 2008. — Vol. 6, no. 12. — P. e26752.

90 *Greene, Derek.* Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering / Derek Greene, Pádraig Cunningham // Proceedings of the 23rd International Conference on Machine Learning. — ICML '06. — New York, NY, USA: Association for Computing Machinery, 2006. — P. 377–384. https://doi.org/10.1145/1143844.1143892.

91 Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books / Yukun Zhu, Ryan Kiros, Rich Zemel et al. // The IEEE International Conference on Computer Vision (ICCV). — 2015. — December.

92 The Pile: An 800GB Dataset of Diverse Text for Language Modeling / Leo Gao, Stella Biderman, Sid Black et al. // *arXiv preprint arXiv:2101.00027.* — 2020.

93 *Presser, Shawn.* Books3. 2020. — https://twitter.com/theshawwn/status/1320282149329784833. — 2020.

94 *Kornilova, Anastassia.* BillSum: A Corpus for Automatic Summarization of US Legislation. — 2019.

95 *Koupaee, Mahnaz.* WikiHow: A Large Scale Text Summarization Dataset. — 2018.

96 *Radev, Dragomir R.* Introduction to the Special Issue on Summarization / Dragomir R. Radev, Eduard Hovy, Kathleen McKeown // *Computational Linguistics.* — 2002.

97 Text Summarization: A Brief Review / Laith Abualigah, Mohammad Qassem Bashabsheh, Hamzeh Alabool, Mohammad Shehab // *Studies in Computational Intelligence.* — 2020. — Vol. 874, no. January. — Pp. 1–15.

98 Assessing sentence scoring techniques for extractive text summarization / Rafael Ferreira, Luciano Cabral, Rafael Lins et al. // *Expert Systems with Applications.* — 2013. — 10. — Vol. 40. — P. 5755–5764.

99 *Akhmetov, I.* Using K-Means and Variable Neighborhood Search for Automatic Summarization of Scientific Articles / I. Akhmetov, N. Mladenovic, R. Mussabayev // Variable Neighborhood Search / Ed. by Nenad Mladenovic, Andrei Sleptchenko, Angelo Sifaleras, Mohammed Omar. — Cham: Springer International Publishing, 2021. — Pp. 166–175.

100 *Zhang, Jianmin.* Towards a Neural Network Approach to Abstractive Multi-Document Summarization / Jianmin Zhang, Jiwei Tan, Xiaojun Wan // *arXiv preprint arXiv:1804.09010.* — 2018.

101 *Burke, Edmund K.* Search methodologies: Introductory tutorials in optimization and decision support techniques, second edition / Edmund K. Burke, Kendall Graham. — Switzerland: Springer, 2014.

102 *Black, Paul E.* Dictionary of Algorithms and Data Structures. — 2005. `https://www.nist.gov/dads/HTML/greedyalgo.html`.

103 *Mitchell, Melanie.* An introduction to genetic algorithms / Melanie Mitchell. — MIT Press, US, 1996. — feb.

104 *Rajaraman, Anand.* Data Mining / Anand Rajaraman, Jeffrey David Ullman // Mining of Massive Datasets. — Cambridge University Press, 2011. — P. 1–17.

105 *Guthrie, William F.* NIST/SEMATECH e-Handbook of Statistical Methods (NIST Handbook 151). — 2020. `https://www.itl.nist.gov/div898/handbook/`.

106 *Pearson, Karl.* Note on Regression and Inheritance in the Case of Two Parents / Karl Pearson // *Proceedings of the Royal Society of London Series I.* — 1895. — jan. — Vol. 58. — Pp. 240–242.

107 *Rudin, Walter.* Principles of mathematical analysis, 3rd edition / Walter Rudin. — McGraw-Hill New York, 1976. — Pp. 1–342.

108 SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python / Pauli Virtanen, Ralf Gommers, Travis E. Oliphant et al. // *Nature Methods*. — 2020. — Vol. 17. — Pp. 261–272.

109 *Paice, Chris D.* Review of "Automatic Summarization"by Inderjeet Mani, Amsterdam: John Benjamins (Natural Language Processing Series, Edited by Ruslan Mitkov, Volume 3), 2001 / Chris D. Paice. — Cambridge, MA, USA: MIT Press, 2002. — Jun.. — Vol. 28. — P. 221–223.

110 Summarizing text documents: Sentence selection and evaluation metrics / Jade Goldstein, Mark Kantrowitz, Vibhu Mittal, Jaime Carbonell // Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. — 1999. — Pp. 121–128.

111 *Giannakopoulos, George*. Summary evaluation: Together we stand npower-ed / George Giannakopoulos, Vangelis Karkaletsis // International Conference on Intelligent Text Processing and Computational Linguistics / Springer. — 2013. — Pp. 436–450.

112 *Gholamrezazadeh, Saeedeh*. A comprehensive survey on text summarization systems / Saeedeh Gholamrezazadeh, Mohsen Amini Salehi, Bahareh Gholamzadeh // 2009 2nd International Conference on Computer Science and its Applications / IEEE. — 2009. — Pp. 1–6.

113 BLEU: a method for automatic evaluation of machine translation / Kishore Papineni, Salim Roukos, Todd Ward, Wei-Jing Zhu // Proceedings of the 40th annual meeting of the Association for Computational Linguistics. — 2002. — Pp. 311–318.

114 *Lin, Chin-Yew*. Automatic evaluation of summaries using n-gram co-occurrence statistics / Chin-Yew Lin, Eduard Hovy // Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. — 2003. — Pp. 150–157.

115 *Lin, Chin-Yew*. Rouge: A package for automatic evaluation of summaries / Chin-Yew Lin // Text summarization branches out. — 2004. — Pp. 74–81.

116 *Bird, Steven*. Natural language processing with Python: analyzing text with the natural language toolkit / Steven Bird, Ewan Klein, Edward Loper. — "O'Reilly Media, Inc. 2009.

117 *Nenkova, Ani*. Evaluating content selection in summarization: The pyramid method / Ani Nenkova, Rebecca J Passonneau // Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004. — 2004. — Pp. 145–152.

118 *Cohan, Arman*. Revisiting Summarization Evaluation for Scientific Articles. — 2016.

119 *Ermakova, Liana*. GRAD: A Metric for Evaluating Summaries. / Liana Ermakova, Anton Firsov // Coria. — 2018.

120 *Vasilyev, Oleg.* Fill in the BLANC: Human-free quality estimation of document summaries / Oleg Vasilyev, Vedant Dharnidharka, John Bohannon // Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems. — Online: Association for Computational Linguistics, 2020. — Nov.. — Pp. 11–20. https://aclanthology.org/2020.eval4nlp-1.2.

121 Reaching for upper bound ROUGE score of extractive summarization methods / Iskander Akhmetov, Iskander Akhmetov, Alexander Gelbukh, Rustam Mussabayev // *PeerJ Computer Science.* — 2022. — Vol. 8:e1103.

122 *Hansen, Pierre.* Variable neighborhood search. — 2018.

123 Variable neighbourhood search: Methods and applications / Pierre Hansen, Nenad Mladenović, José A. Moreno Pérez, José A. Moreno Pérez // *Annals of Operations Research.* — 2010.

124 Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion / Lucy Vanderwende, Hisami Suzuki, Chris Brockett, Ani Nenkova // *Information Processing and Management.* — 2007.

125 *Jezek, Karel.* Using latent semantic analysis in text summarization and summary evaluation / Karel Jezek, Josef Steinberger, Karel Ježek // Proceedings of the 7th International Conference ISIM. — 2004.

126 Enriching Word Vectors with Subword Information / Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov // *arXiv preprint arXiv:1607.04606.* — 2016.

127 BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers) / Ed. by Jill Burstein, Christy Doran, Thamar Solorio. — Association for Computational Linguistics, 2019. — Pp. 4171–4186. https://doi.org/10.18653/v1/n19-1423.

128 *Clark, Kevin.* ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. — 2020.

129 Deep contextualized word representations / Matthew E. Peters, Mark Neumann, Mohit Iyyer et al. // Proc. of NAACL. — 2018.

130 Scikit-learn: Machine Learning in Python / F. Pedregosa, G. Varoquaux, A. Gramfort et al. // *Journal of Machine Learning Research.* — 2011. — Vol. 12. — Pp. 2825–2830.

131 *Dodge, Jesse.* Documenting the English Colossal Clean Crawled Corpus. — 2021.

132 *Pinto, David.* Clustering Abstracts of Scientific Texts Using the Transition

Point Technique / David Pinto, Héctor Jiménez-Salazar, Paolo Rosso // Computational Linguistics and Intelligent Text Processing / Ed. by Alexander Gelbukh. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. — Pp. 536–546.

133 *Ruder, Sebastian.* On word embeddings - Part 3: The secret ingredients of word2vec. — http://ruder.io/secret-word2vec/. — 2016.

134 *Fu, Tsu-Jui.* DOC2PPT: Automatic Presentation Slides Generation from Scientific Documents. — 2021. https://arxiv.org/abs/2101.11796.

135 Extractive Research Slide Generation Using Windowed Labeling Ranking / Athar Sefid, Prasenjit Mitra, Jian Wu, C Lee Giles // Proceedings of the Second Workshop on Scholarly Document Processing. — Online: Association for Computational Linguistics, 2021. — Jun.. — Pp. 91–96. https://aclanthology.org/2021.sdp-1.11.

136 Automatic question generation and answer assessment: a survey / Bidyut Das, Mukta Majumder, Santanu Phadikar, Arif Ahmed Sekh // *Research and Practice in Technology Enhanced Learning.* — 2021. — Vol. 16.

137 Feature and Opinion Mining for Customer Review Summarization / Muhammad Abulaish, Jahiruddin, Mohammad Najmud Doja, Tanvir Ahmad // Pattern Recognition and Machine Intelligence / Ed. by Santanu Chaudhury, Sushmita Mitra, C. A. Murthy et al. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. — Pp. 219–224.

138 Report on the CLEF-IP 2011 Experiments: Exploring Patent Summarization / Parvaz Mahdabi, Linda Andersson, Allan Hanbury, Fabio Crestani // Proceedings of the Conference on Multilingual and Multimodal Information Access Evaluation, CLEF 2011. — Vol. 1177. — 2011.

139 Extracting clinical information from electronic medical records / Manuel Lamy, Rúben Pereira, João C. Ferreira et al. // *Advances in Intelligent Systems and Computing.* — 2019. — Vol. 806.

140 Automatic Generation of Textual Summaries from Neonatal Intensive Care Data / François Portet, Ehud Reiter, Jim Hunter, Somayajulu Sripada // Artificial Intelligence in Medicine / Ed. by Riccardo Bellazzi, Ameen Abu-Hanna, Jim Hunter. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. — Pp. 227–236.

141 *Pandey, Abhay.* Phases of drug development process, Drug Discovery Process. — 2022. — Apr. https://www.nebiolab.com/drug-discovery-and-development-process/.

142 *Anonymous.* IIDF sold its share in SemanticHub: Russian AI developer for pharmaceutical industry. — 2021. https://zdrav.expert/a/375803.

143 *Armano, Giuliano.* Experimenting text summarization techniques for contextual advertising / Giuliano Armano, Alessandro Giulian, Eloisa Vargiu //

Proceedings of the 2nd Italian Information Retrieval (IIR) Workshop. — Vol. 704. — 2011.

144 *Zhang, Cici*. Baidu's AI produces short videos in one click. — 2021. — Jun. https://spectrum.ieee.org/baidus-ai-produces-short-videos-in-one-click.

Приложение А Листинг кода GreedSum

Листинг кода GreedSum на языке python[1]:

```
import argparse
from sklearn.feature_extraction.text import TfidfVectorizer
import numpy as np

def greed_sum(text, num_sent, min_df=1, max_df=1.0):

    #fit a TFIDF vectorizer
    vectorizer = TfidfVectorizer(min_df=min_df, max_df=max_df)
    vectorizer.fit(text)

    #get the matrix
    X = vectorizer.transform(text).toarray()

    #get the sentence indices
    idx = []
    while sum(sum(X)) != 0:
        ind = np.argmax(X.sum(axis=1))
        idx.append(ind)

        #update the matrix deleting the columns corresponding to the words found in previous step
        cols = X[ind]
        col_idx = [i for i in range(len(cols)) if cols[i] > 0]
        X = np.delete(X, col_idx, 1)


    idx = idx[:num_sent]
    idx.sort()

    summary = [text[i] for i in idx]
    return summary

def main():
    #parse arguments
    parser = argparse.ArgumentParser()
    parser.add_argument('--input_fname', help="Summarization input file name")
    parser.add_argument('--output_fname', help="Summarization output file name")
    parser.add_argument('--min_df', help="Minimum document frequency word threshold")
    parser.add_argument('--max_df', help="Maximum document frequency word threshold")
    parser.add_argument('--num_sent', help="Number of sentences for summary")

    args = parser.parse_args()

    if not args.input_fname:
        print("No output file name was provided, quitting")
        quit
    else:
        INPUT_FILENAME = args.input_fname
        print("Input file name was provided:", INPUT_FILENAME)

```

[1]https://github.com/iskander-akhmetov/Greedy-Summarization

```
    OUTPUT_FILENAME = 'summary_output.txt' #the default file name
    if not args.output_fname:
        print("No output file name was provided, using the default", OUTPUT_FILENAME)
    else:
        OUTPUT_FILENAME = args.output_fname
        print("Output file name was provided:", OUTPUT_FILENAME)

    MIN_DF = 1.0 #the default min_df
    if not args.min_df:
        print("No minimum document frequency parameter was provided, using the default", MIN_DF)
    else:
        MIN_DF = float(args.min_df)
        print("Minimum document frequency parameter was provided:", MIN_DF)

    MAX_DF = 1.0 #the default max_df
    if not args.max_df:
        print("No maximum document frequency parameter was provided, using the default", MAX_DF)
    else:
        MAX_DF = float(args.max_df)
        print("Maximum document frequency parameter was provided:", MAX_DF)

    NUM_SENT = 10 #the default max_df
    if not args.num_sent:
        print("No number of sentences was provided, using the default", NUM_SENT)
    else:
        NUM_SENT = int(args.num_sent)
        print("Number of sentences was provided:", NUM_SENT)


    # 1. Get the text from the file provided
    #f = open(INPUT_FILENAME, 'r', encoding='cp1251')
    with open(INPUT_FILENAME, "r", encoding="UTF-8") as f:
        text = ' '.join([l.strip() for l in f.readlines()]).replace('\n',' ').replace('  ',' ').split('. ')
    f.close()

    # 2. Summarize
    summary = greed_sum(text, NUM_SENT, min_df=MIN_DF, max_df=MAX_DF)

    # 3. Save summary to a file
    f = open(OUTPUT_FILENAME, 'w', encoding="UTF-8")
    #f.writelines(summary)
    for s in summary:
        print(s)
        print()
        f.write(s+'\n')

    f.close()

if __name__ == "__main__":
    main()
```

Приложение В Примеры авторефератов сгенерированных GreedSum

2.1 Пример 1

- **Title:** The spin and orientation of dark matter halos within cosmic filaments
- **Article_ID:** 0906.1654[1]
- **ROUGE-1:** 0.49
- **ROUGE-2:** 0.12

2.1.1 Сгенерированный автореферат

we first identify and classify the large - scale environment , then we determine whether a given halo resides in a filament or sheet , and finally we try to find correlations between the halo spin and shape orientations and their large - scale environment . up to the present day , a number of different approaches have been suggested to find filaments (and/or sheets) in simulations as well as in observations . among these methods. pimbblet (2005) searched the 2df galaxy redshift survey catalog for filamentary structures using the orientations of inter - cluster galaxies . a related approach based on the inter - cluster dark matter distribution derived from n - body simulations. the cosmological parameters used in the simulation are @xmath8 , @xmath9 , @xmath10 , @xmath11 , @xmath12 , and @xmath13 . in the upper - left panel of fig .. associated with each segment , there are two node halos , one of which is the most massive one among all the associated halos .. [major_masstop] , we show the alignment signals for halos in segments with most massive halos in four mass bins (note that this can only be done for method slowromancap2@) .. this alignment strength is in very good agreement with that obtained by aragn - calvo et al .. we first rank all the (member) halos within the filaments according to the cosine of their angles between the major axes and the filament directions , @xmath128 , here we use @xmath129 to represent the angle between the major axis and the filament directions .] .. hahn et al . (2007b) applied a hessian matrix approach to the gravitational potential field (instead of the density field) , and also found an opposite mass dependence for the alignment strengths of spin- and shape - filament alignments .

[1] https://arxiv.org/abs/0906.1654

2.1.2 Оригинальная аннотация

clusters , filaments , sheets and voids are the building blocks of the cosmic web .. forming dark matter halos respond to these different large - scale environments , and this in turn affects the properties of galaxies hosted by the halos .. it is therefore important to understand the systematic correlations of halo properties with the morphology of the cosmic web , as this informs both about galaxy formation physics and possible systematics of weak lensing studies . in this study. , we present and compare two distinct algorithms for finding cosmic filaments and sheets , a task which is far less well established than the identification of dark matter halos or voids .. one method is based on the smoothed dark matter density field , the other uses the halo distributions directly .. we apply both techniques to one high resolution n - body simulation and reconstruct the filamentary / sheet like network of the dark matter density field .. we focus on investigating the properties of the dark matter halos inside these structures , in particular on the directions of their spins and the orientation of their shapes with respect to the directions of the filaments and sheets .. we find that both the spin and the major axes of filament - halos with masses @xmath0 are preferentially aligned with the direction of the filaments .. the spins and major axes of halos in sheets tend to lie parallel to the sheets .. there is an opposite mass dependence of the alignment strengths for the spin (negative) and major (positive) axes , i.e. with increasing halo mass the major axis tends to be more strongly aligned with the direction of the filament whereas the alignment between halo spin and filament becomes weaker with increasing halo mass .. the alignment strengths as a function of distance to the most massive node halo indicate that there is a transit large scale environment impact : from the 2-d collapse phase of the filament to the 3-d collapse phase of the cluster / node halo at small separation .. overall , the two algorithms for filament / sheet identification investigated here agree well with each other .. the method based on halos alone can be easily adapted for use with observational data sets .

2.2 Пример 2

- **Title:** First passage times and asymmetry of DNA translocation
- **Article_ID:** q-bio0508010[2]

[2] https://arxiv.org/abs/q-bio/0508010

- **ROUGE-1:** 0.43
- **ROUGE-2:** 0.13

2.2.1 Сгенерированный автореферат

authors of the work @xcite phenomenologically interpret their data by assigning two different diffusion constants for the two separate experiments in which the same dna is placed in the channel in two possible orientations .. indeed , for free diffusion , the friction coefficient (averaged over the scale well exceeding a single base) moving in one direction and in the opposite direction must be the same , as follows from the onsager symmetry relation , and the assumption of asymmetric friction would be a grave mistake . although no one actually made this mistake , including @xcite , it is worth emphasizing why an assumption of asymmetric friction would be a mistake . indeed , if we only imagine that dna (not driven by any applied voltage !) diffuses in one direction faster than in the other , then we can easily build a _. (note that since we know very little about the details of the interactions between the dna bases and the pore , we can not determine if case a in figure [fig : sawtooth] corresponds to case 1a in figure [fig : dnaporecases] , and case b in figure [fig : sawtooth] corresponds to case 1b in figure [fig : dnaporecases] , or if it is the other way around .) since dna translocation is ultimately not classical diffusion , but rather subdiffusion @xcite , we consider also the first passage times for the subdiffusion in the presence of an asymmetric potential . in general , the first passage time for subdiffusion was recently a matter of considerable interest and dispute in the literature @xcite .. the solution of this differential equation for a sawtooth potential @xmath15 is outlined in appendix [sec : solution] . for the particle initially located at the origin (@xmath34) , the mean first passage time to reach @xmath22. conditional _ probability distribution of the first passage events that get counted under such a protocol is then given by @xmath64 for such an experiment , there exists a perfectly defined and finite average first passage time .. , we expect a long time interval in which the behaviour of the mittag - leffler function @xmath115 behaves like an exponential ; at much longer times the behaviour changes to a power law . the crossover is expected to happen when @xmath116 , or at about @xmath117 .. after some algebra one obtains @xmath121\ ]] or @xmath122\ ]] for @xmath57

, corresponding to classical diffusion , the survival probability @xmath123 decays exponentially and the term with the integral goes to zero , yielding the familiar result of @xmath124 for the right - hand - side @xcite . for @xmath125 , @xmath123 goes like @xmath126 (see ([s - series])) and the term with the integral goes like @xmath127 .. limit , using ([eigensol]) and ([mittaglimit]) , @xmath149 @xmath150 @xmath151 again , these results indicate that the mfpt diverges for @xmath125 .

2.2.2 Оригинальная аннотация

motivated by experiments in which single - stranded dna with a short hairpin loop at one end undergoes unforced diffusion through a narrow pore , we study the first passage times for a particle , executing one - dimensional brownian motion in an asymmetric sawtooth potential , to exit one of the boundaries .. we consider the first passage times for the case of classical diffusion , characterized by a mean - square displacement of the form @xmath0 , and for the case of anomalous diffusion or subdiffusion , characterized by a mean - square displacement of the form @xmath1 with @xmath2 . in the context of classical diffusion ,. we obtain an expression for the mean first passage time and show that this quantity changes when the direction of the sawtooth is reversed or , equivalently , when the reflecting and absorbing boundaries are exchanged .. we discuss at which numbers of ` teeth ' @xmath3 (or number of dna nucleotides) and at which heights of the sawtooth potential this difference becomes significant . for large @xmath3. , it is well known that the mean first passage time scales as @xmath4 . in the context of subdiffusion , the mean first passage time. does not exist .. therefore we obtain instead the distribution of first passage times in the limit of long times .. we show that the prefactor in the power relation for this distribution is simply the expression for the mean first passage time in classical diffusion .. we also describe a hypothetical experiment to calculate the average of the first passage times for a fraction of passage events that each end within some time @xmath5 .. we show that this average first passage time scales as @xmath6 in subdiffusion .

2.3 Пример 3

- **Title:** Mid Infrared Spectral Energy Distribution of NGC 1068 with 0.1 arcsec Spatial Resolution

- **Article_ID:** astro-ph/0108413[3]
- **ROUGE-1:** 0.37
- **ROUGE-2:** 0.1

2.3.1 Сгенерированный автореферат

mir spectra of ngc 1068 and other agns have shown the silicate absorption feature at 9.7 @xmath5 m .. instead , we performed nodding observations .. data with lower time resolution are also used along with those used for the saa method .. number of iteration was limited to 110 . again , resultant images for different nights are compared and confirm that they have essentially the same morphology .. excess flux in our 4 aperture is l@xmath37 @xmath6 2 @xmath1 10@xmath36 w. this is , for example , 7 times larger than 3.0 @xmath1 10@xmath38 w that is measured by @xcite from the spectrum taken by the short wavelength spectrometer on board _ infrared space observatory_. the single temperature continuum model adopted in this work may be too simple , and uncertainty in relative calibration between the filters may be too large to measure the excess luminosity .. in higher surface brightness , figure [fig : r.contour]@xmath21 shows an elongation toward a p.a . of @xmath010 from the central peak .. consequently , the disklike structures can not be explained as emission sources absorbed by dust .. has been observed to be in the north - south direction , the plausible torus has a diameter comparable or less than 13 pc .

2.3.2 Оригинальная аннотация

the central region of the seyfert 2 galaxy ngc 1068 is imaged in the mid infrared (mir) using the mid - infrared test observation system on the 8.2 m subaru telescope .. the oversampling pixel scale associated with shift - and - add method shows 0.1 resolution images with a high dynamic range after deconvolution . along with an extended structure at a position angle (p.a .) of @xmath010 with higher surface brightness , another structure extends wider with lower surface brightness at a p.a .. of 20 .. the central peak elongates north - south with fwhm of 0.3 @xmath1 0.2 .. spectral energy distribution (sed) of the central peak is fitted to have the silicate absorption feature of @xmath2 = 0.9 @xmath3 0.3 .. this is half of the absorption expected from the near - infrared (nir) feature of carbonaceous dust

[3] https://arxiv.org/abs/astro-ph/0108413

.. this suggests a temperature gradient of the absorbing dust along the line of sight .. another possibility , which is not distinguishable here , is the size distribution of dust different from our galaxy .. intrinsic luminosity of emission from the central peak is 3 @xmath1 10@xmath4 w. the sed shows a hint of the poly aromatic hydrocarbon (pah) emission features .. although a high spatial resolution mir spectrum is required , it suggests that the pah carriers near the active galactic nuclei (agns) are sheltered from the high - energy emission from the agns and the agns have nuclear starbursts . for the nir disklike structures ,. no counterparts are detected in the mir .. the nature of the structures remains unclear .

2.4 Пример 4

- **Title:** Kinetics of a Network of Vortex Loops in He II and a Theory of Superfluid Turbulence
- **Article_ID:** 0802.0651[4]
- **ROUGE-1:** 0.47
- **ROUGE-2:** 0.12

2.4.1 Сгенерированный автореферат

on the right picture we depicted the self - intersection and break down of loop of the length @xmath0 into two daughter loops with lengths @xmath10 and @xmath11 the rates of these processes are characterized by the rate coefficients @xmath12 and @xmath6 correspondingly.it is widely appreciated that the `` recombination '' processes greatly influence both the structure and dynamics of the vortex tangle .. the feynman s idea was confirmed in various numerical calculations , where the procedure of artificial elimination of small loops had been used @xcite-@xcite . to clarify the role of recombination , let us perform the following numerical estimation .. there are two mechanisms for change of @xmath39 the first one is the mentioned above deterministic process of evolution of elements of the individual loops , during which they move undergo the stretching or shrinking .. let us consider function @xmath47which is the vector connecting points @xmath48 and @xmath49 (see fig .. this enables us to express the net flux @xmath158 ([flux]) via quantity @xmath14 . substituting (vld_vs_curvature) into relation for the net flux ([flux]) we arrive at conclusion that both the

[4] https://arxiv.org/abs/0802.0651

positive constituent @xmath142 and the negative one @xmath159 are proportional to the squared vortex line density @xmath160 in unsteady case at finite temperature quantities @xmath142 and @xmath143 do not compensate each other , so the net flux @xmath141 does not vanish and it is also proportional to the squared vortex line density @xmath160 that means that the rate of decay of quantity @xmath125 due to fluxes carrying away the length from the system can be written as @xmath161relation ([ve]) is the particular case of the so - called vinen equation discussed in detail in the next subsection .. if we take all terms in collision integral with the plus sign and use for estimation our solution for @xmath1 we obtain the total number of reconnections .. the distant parts @xmath219 are separated in @xmath217 space by the distance @xmath220 , i.e. the vortex loop has the typical random walk structure . the scale @xmath221 is depicted here in the left upper corner.,width=264] the average loop can be imagined as consisting of many arches with the mean radius of curvature equal @xmath35 randomly (but smoothly) connected to each other .. the presence of counterflow velocity violates an assumptions of the isotrpic wiener distribution used in previous sections .

2.4.2 Оригинальная аннотация

a theory is developed to describe the superfluid turbulence on the base of kinetics of the merging and splitting vortex loops . because of very frequent reconnections the vortex loops (as a whole) do not live long enough to perform any essential evolution due to the deterministic motion . on the contrary , they rapidly merge and split , and these random recombination processes prevail over other slower dynamic processes . to develop quantitative description we take the vortex loops to have a brownian structure with the only degree of freedom , which is the length @xmath0 of the loop .. we perform investigation on the base of the boltzmann type kinetic equation for the distribution function @xmath1 of number of loops with length @xmath0 .. this equation describes a slow change of the density of loops (in space of their lengths @xmath0) due to the deterministic equation of motion and due to fast random change because of the frequent reconnections . by use of the special ansatz in the collision . integral we have found the exact power - like solution @xmath2 to kinetic equation in the stationary case .. this solution is not (thermodynamically) equilibrium , but on the contrary ,

it describes the state with two mutual fluxes of the length (or energy) in space of sizes of the vortex loops .. the term flux means just redistribution of length (or energy) among the loops of different sizes due to reconnections .. analyzing this solution we drew several results on the structure and dynamics of the vortex tangle in the turbulent superfluid helium .. in particular , we obtained that the mean radius of the curvature is of the order of interline space .. we also evaluated the full rate of the reconnection events . assuming , further , that the processes of random collisions are the fastest ones , we studied the evolution of full length of vortex loops per unit volume - the so - called vortex line density @xmath3 .. it is shown this evolution to obey the famous vinen equation .. the properties of the vinen equation from the point of view of the developed approach had been discussed .. thus , depending on the temperature (and independently on velocity) vortices either develop into the highly chaotic turbulent state (low temperature) , or degenerate into few smooth lines (high temperature) .. this observation can be an alternative explanation for the phenomenon discovered in helsinki group (nature 424 , 10221025 (2003)).pacs - numbers : 67.25.dk , 47.37.+q , 05.20.-y

2.5 Пример 5

- **Title:** The ACS Survey of Galactic Globular Clusters. VII. Relative Ages
Article_ID: 0812.4541[5]
- **ROUGE-1:** 0.4
- **ROUGE-2:** 0.08

2.5.1 Сгенерированный автореферат

using the derived mean ridge lines , we performed ms@xmath4 and rgb@xmath4fitting between each cluster in each metallicity group and the corresponding reference cluster . figure [msfexample] shows examples of the fitting , in which the mean ridge lines of clusters with @xmath23_{\rm cg } < -1.1 \$] (solid lines) have been shifted in both magnitude and color to fit the reference cluster , ngc 6981 (dashed line) .. in particular , d07 isochrones with similar metallicity and different ages were superimposed on the same cmd . based on visual inspection , these two particular regions were found to have little dependence upon cluster age , and were adopted as the optimum intervals for the

[5] https://arxiv.org/abs/0812.4541

ms fitting procedure .. typical values of @xmath48 are in the interval @xmath50.01 - 0.06 mag . to estimate the uncertainties induced by differential reddening , binary star population , and total number of cluster stars , as well as by differences between the measured msto and the actual ggc s msto , we generated over 250 synthetic cmds using the iac@xmath4star @xcite synthetic cmd program .. the effect that this @xmath65 0.05 dex uncertainty has on the final relative ages will be discussed at the end of this section .. is measured for each ggcs in our database , and then obtained magnitudes are transformed into ages using a set of theoretical stellar evolution models . during this transformation ,. it is worth mentioning that while the age@xmath4metallicity relation may be model dependent (though it is somehow reassuring that the four most recent theoretical libraries provide consistent results on this respect) the age dispersion@xmath4metallicity relation is not model dependent . from now on. a more detailed analysis is necessary , but it is important to note that any successful galaxy formation scenario must account for the existence of a large nubmer of old , coeval globular clusters in the milky way .. on the other hand if the young group is taken into account , it can be seen that the age s variance increases with galactocentric distance . in order to determine if this variance increase depends either on the galactocentric distance or the metallicity , a three dimensional principal@xmath4component (pc)

2.5.2 Оригинальная аннотация

the acs survey of galactic globular clusters is a \verbatim{@xmath}0 @xmath1 @xmath2 @xmath3 treasury program designed to provide a new large , deep and homogeneous photometric database .. based on observations from this program , we have measured precise relative ages for a sample of 64 galactic globular clusters by comparing the relative position of the clusters main sequence turn offs , using main@xmath4sequence fitting to cross@xmath4compare clusters within the sample .. this method provides relative ages to a formal precision of 2 - 7\% .. we demonstrate that the calculated relative ages are independent of the choice of theoretical model .. we find that the galactic globular cluster sample can be divided into two groups a population of old clusters with an age dispersion of @xmath55\% and no age@xmath4metallicity relation , and a group

of younger clusters with an age@xmath4metallicity relation similar to that of the globular clusters associated with the sagittarius dwarf galaxy . . these results are consistent with the milky way halo having formed in two phases or processes. . the first one would be compatible with a rapid (@xmath60.8 gyr) assembling process of the halo , in which the clusters in the old group were formed .. also these clusters could have been formed before reionization in dwarf galaxies that would later merge to build the milky way halo as predicted by @xmath7cdm cosmology .. however , the galactocentric metallicity gradient shown by these clusters seems difficult to reconcile with the latter . as for the younger clusters ,. it is very tempting to argue that their origin is related to their formation within milky way satellite galaxies that were later accreted , but the origin of the age@xmath4metallicity relation remains unclear .