

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

На правах рукописи



Мохаммад Жуман Хуссайд

**МОДЕЛЬ, МЕТОД И АЛГОРИТМЫ ВЕБ-ИНЖЕНЕРИИ
ДЛЯ ИНТЕЛЛЕКТУАЛЬНОЙ ПЕРСОНАЛИЗАЦИИ
КОНТЕНТА ВЕБ-САЙТОВ**

Специальность – 1.2.1. Искусственный интеллект и машинное обучение
(технические науки)

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель –
Кравченко Юрий Алексеевич
доктор технических наук, доцент

Таганрог – 2026

Содержание

ВВЕДЕНИЕ.....	5
ГЛАВА 1. АНАЛИТИЧЕСКИЙ ОБЗОР СОВРЕМЕННЫХ ПОДХОДОВ К ПЕРСОНАЛИЗАЦИИ КОНТЕНТА ВЕБ-САЙТОВ НА ОСНОВЕ МЕТОДОВ ОБРАБОТКИ И АНАЛИЗА ТЕКСТОВ.....	17
1.1. Персонализация веб-контента в составе технологии Веб-Инженерии	17
1.1.1. Понятие Веб-инженерии и персонализации	18
1.1.2. Этапы процесса персонализации веб-контента	22
1.2. Обзор методов построения профиля пользователя.....	30
1.2.1. Методы, основанные на концептах	30
1.2.2. Методы, основанные на онтологических моделях.....	35
1.3. Методы извлечения ключевых фраз, используемые в задачах персонализации	38
1.4. Методы устранения неоднозначности смысла слов	43
1.5. Постановка задачи персонализации контента.....	47
1.6. Выводы по разделу	50
ГЛАВА 2. МОДИФИЦИРОВАННЫЙ МЕТОД ИЗВЛЕЧЕНИЯ КЛЮЧЕВЫХ ФРАЗ И АЛГОРИТМ УСТРАНЕНИЯ НЕОДНОЗНАЧНОСТИ СМЫСЛА СЛОВ.....	53
2.1. Разработка модифицированного метода извлечения ключевых фраз	55
2.1.1. Постановка задачи извлечения ключевых фраз	56
2.1.2. Функциональная схема метода извлечения ключевых фраз	56

2.1.3. Применение FBKE в задачах персонализации контента веб-сайтов	63
2.2. Разработка модифицированного алгоритма устранения неоднозначности смысла слов	64
2.2.1. Постановка задачи устранения неоднозначности слов	69
2.2.2. Разработка алгоритма устранения неоднозначности смысла слов на основе контекстуального векторного представления	70
2.3. Выводы по разделу	74
ГЛАВА 3. ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ И АЛГОРИТМ ПЕРСОНАЛИЗИРОВАННОГО ПОИСКА ВЕБ-САЙТОВ	76
3.1. Онтологическая модель представления интересов пользователей	76
3.2. Алгоритм персонализированного поиска веб-сайтов	84
3.2.1. Алгоритм создания профиля пользователя	85
3.2.2. Процедура ранжирования документов при персонализации интересов пользователя	91
3.2.3. Процедура обновления профиля пользователя	93
3.3. Выводы по разделу	96
ГЛАВА 4. РАЗРАБОТКА ПРОГРАММНОГО ПРИЛОЖЕНИЯ И ПРОВЕДЕНИЕ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА	97
4.1. Модульная архитектура программного приложения	97
4.2. Результаты эксперимента по анализу эффективности модифицированного метода <i>FBKE</i> и настройка его параметров	100
4.3. Результаты эксперимента по анализу эффективности модифицированного алгоритма устранения неоднозначности слов	110

4.4. Результаты эксперимента по анализу эффективности алгоритма персонализации результатов поиска веб-сайтов	116
4.5. Результаты эксперимента по анализу эффективности онтологии	121
4.6. Выводы по разделу	136
ЗАКЛЮЧЕНИЕ	138
СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ	142
СПИСОК ЛИТЕРАТУРЫ	143
ПРИЛОЖЕНИЕ №1	160
ПРИЛОЖЕНИЕ №2	164

ВВЕДЕНИЕ

Актуальность темы исследования и степень её разработанности. Технологии искусственного интеллекта (ИИ) развиваются стремительно, объёмы данных растут экспоненциально, а цифровые сервисы – социальные сети, Интернет вещей, большие данные, облачные вычисления – проникают во все сферы жизни. В этих условиях персонализация представления веб-контента под индивидуальные потребности пользователей приобретает ключевое значение. Найти релевантную информацию всё труднее: доступного контента стало слишком много, а по форматам и источникам он крайне разнороден. Отсюда возникает феномен информационной перегрузки – состояние, при котором пользователь получает чрезмерный объём данных, значительная часть которых не отвечает его актуальным потребностям и интересам [1, 2]. В результате эффективность информационного поиска заметно падает, а когнитивная нагрузка, напротив, растёт: пользователю приходится отфильтровывать нерелевантный контент, ориентироваться в информационном шуме и принимать решения в условиях семантической неоднозначности.

Всё это делает востребованной разработку более совершенных технологий фильтрации и персонализации контента. Здесь на первый план выходит веб-инженерия, задача которой – создание веб-систем, отвечающих потребностям пользователей [3, 4]. Персонализация относится к её ключевым направлениям. Под ней понимается совокупность информационных процессов и методов, направленных на формирование индивидуализированного пользовательского взаимодействия на основе анализа предпочтений, поведения и других релевантных данных пользователя [5–7]. На практике это означает динамическую адаптацию веб-контента, рекомендаций и пользовательских интерфейсов под конкретные потребности и интересы каждого пользователя.

Основой персонализации веб-контента служит профиль пользователя. Его построение предполагает сбор и анализ данных об отдельных пользователях, позволяющих понять их предпочтения, интересы и поведение [7–10]. Современные подходы к построению пользовательских профилей опираются на две парадигмы: методы на основе концептов и методы на основе онтологических моделей. Первые сводятся главным образом к извлечению ключевых слов или фраз из пользовательских данных, анализу их частоты и выделению значимых терминов [11]. Такие методы просты и вычислительно эффективны, однако базовую семантику текстового контента они учитывают слабо: слова рассматриваются как изолированные единицы, а контекстные связи между ними игнорируются. Особенно остро это ограничение проявляется при неоднозначности на уровне предметной области, когда один и тот же термин интерпретируется по-разному в разных областях. Так, термин «cell» (клетка) может обозначать биологическую клетку в медицинском контексте и мобильный телефон – в контексте телекоммуникаций. В итоге профили оказываются перегружены шумом и двусмысленностью, что повышает процент ошибок алгоритмов функционирования систем персонализации.

Чтобы преодолеть ограничения, связанные с использованием одних лишь ключевых слов, авторы работ [12–15] привлекают внешние классификационные источники. С их помощью термины соотносятся с более общими концептами, а пользовательские интересы представляются в виде взвешенных тематических категорий. Это уже шаг в сторону подходов, основанных на онтологиях, однако семантических связей между концептами таким методам по-прежнему не хватает.

Методы онтологического подхода, напротив, используют формальные онтологии для построения пользовательских профилей и структурируют интересы пользователей в виде онтологических концептов используя формальные семантические отношения. Однако такие методы часто опираются на статические,

предопределённые онтологии, которые не способны адаптироваться к динамичному характеру предпочтений пользователей или новым предметным областям. В [13] пользовательские профили создаются на основе URL-адресов и сопоставляются с категориями из онтологий *OpenDNS* и *DBpedia* [16, 17]. Это обогащает пользовательские профили семантическими связями, но может не обеспечивать точности в захвате конкретных интересов.

Соответственно, методы, основанные на концептах, делают акцент на простоту, но им не хватает семантической глубины, в то время как методы, использующие онтологический подход, предлагают более расширенную семантику, но вместе с тем сопряжены со снижением вычислительной эффективности недостаточной адаптивностью к динамически изменяющимся пользовательским предпочтениям и новым предметным областям.

В результате, существующие методы по-прежнему недостаточно эффективны для обработки текстовых данных и извлечения знаний, что не позволяет достичь необходимого уровня эффективности систем веб-персонализации и делает их неспособными снизить когнитивную нагрузку на пользователя. Это связано с неспособностью этих методов совместить семантическую точность, адаптивность и масштабируемость при построении профилей пользователей.

В данном исследовании предложено комплексное решение задачи персонализации контента веб-сайтов на основе применения методов искусственного интеллекта, в том числе, алгоритмов автоматизации построения онтологий. Эти алгоритмы выполняют три ключевые функции: во-первых, осуществляют извлечение ключевых фраз и понятий; во-вторых, автоматически устраняют возникающую семантическую неоднозначность; и, в-третьих, интегрируются в единый конвейер обработки текстовых данных. Основу решения составляют методы анализа *контекстуальной семантической близости*, которые

принципиально отличаются от статических или синтаксических тем, что учитывают конкретный контекст употребления языковых единиц, а не только их абстрактное значение или формальные связи [18, 19].

Следовательно, развитие моделей, методов и алгоритмов для обработки и анализа текстов на естественном языке и извлечения знаний из них, позволяющих повысить эффективность систем персонализации веб-контента в условиях *семантической неоднозначности*, является **актуальной научной задачей**.

Ряд работ российских и зарубежных учёных посвящен разработке методов персонализации контента веб-сайтов с использованием методов интеллектуального анализа текста, поиска информации и обработки естественных языков. Источниками исследования диссертации послужили следующие труды известных ученых по основам построения профиля пользователя и персонализации контента веб-сайтов: М. Гонтек [20], Б. Мобашер [8], А. Хавала [21], С. Гауч [9, 22], Д. Уэстон и Д. Юрафски [23]. Работы Х. Шютце и К. Д. Мэннинга [24], К. С. Джонса, М. А. Хёрста [25], В.М. Курейчика [26, 27], А. Панченко [28], И. Д. Игнатов [29] и А. Кутузова [30] вносят значительный вклад в разработку методов ИИ, таких как построение онтологий и извлечение знаний из текста.

Целью диссертационной работы является повышение эффективности систем персонализации контента веб-сайтов. Под эффективностью понимается минимизация процента ошибок алгоритмов функционирования систем персонализации.

Для достижения поставленной цели были решены следующие основные **задачи**:

1. Проведён аналитический обзор современных подходов к персонализации контента веб-сайтов на основе методов обработки и анализа текстов.

2. Разработан модифицированный метод извлечения ключевых фраз из отдельных документов, основанный на новой весовой функции, которая учитывает частоту встречаемости фраз и их сходство с документом.

3. Разработан модифицированный алгоритм устранения неоднозначности смысла слов, основанный на методах контекстуального представления и расширенных аннотаций синсетов.

4. Построена онтологическая модель, предназначенная для автоматизации процесса выявления связей между понятиями, с целью отражения индивидуальных особенностей интересов пользователя в его профиле.

5. Разработан алгоритм персонализации интересов пользователя на основе онтологической модели.

6. Разработано программное приложение для проведения вычислительного эксперимента и подтверждение достоверности и эффективности полученных основных результатов.

Объект исследования – тексты на естественном языке.

Предмет исследования – методы и алгоритмы обработки и анализа текстов для решения задачи персонализации контента веб-сайтов.

Методология и методы диссертационного исследования. При выполнении диссертационной работы использовались методы интеллектуального анализа данных, методы обработки и анализа текстов на естественном языке, теории информационных систем, методы искусственного интеллекта и машинного обучения, а также методы объектно-ориентированного программирования.

Степень достоверности результатов. Достоверность научных результатов работы подтверждается непротиворечивостью и согласованностью с известными фактами и исследованиями в рассматриваемой области, высокой степенью сходимости теоретических результатов с данными экспериментов, и определяется применением теоретических и методологических основ разработок ведущих

ученых в области создания интеллектуальных систем, корректным и обоснованным использованием математического аппарата, экспериментальными исследованиями разработанных моделей и методов.

Тематика работы соответствует п. 4 «Разработка методов, алгоритмов и создание систем искусственного интеллекта и машинного обучения для обработки и анализа текстов на естественном языке, для изображений, речи, биомедицины и других специальных видов данных», п. 5 «Методы и технологии поиска, приобретения и использования знаний и закономерностей, в том числе – эмпирических, в системах искусственного интеллекта. Исследования в области совместного применения методов машинного обучения и классического математического моделирования. Методы и средства использования экспертных знаний» паспорта специальности 1.2.1. Искусственный интеллект и машинное обучение (технические науки).

Научная новизна и соответствие научной специальности:

1. Разработан модифицированный метод извлечения ключевых фраз из отдельных документов, **отличающийся** применением новых правил для определения весов ключевых фраз на основе частоты их встречаемости и контекстуальной семантической близости, **позволяющий** извлекать наиболее релевантные ключевые фразы (пункт 5 паспорта специальности 1.2.1, страницы 52-63 диссертации);

2. Разработан модифицированный алгоритм устранения неоднозначности смысла слов, **отличающийся** применением контекстных векторных представлений и расширенных аннотаций синсетов, **что позволяет** снизить процент ошибки при определении правильного значения неоднозначного слова (пункт 5 паспорта специальности 1.2.1, страницы 63-74 диссертации);

3. Построена онтологическая модель для представления индивидуальных особенностей интересов пользователя, **отличающаяся** применением новой

функции расчета весовых коэффициентов, а также контекстно-зависимых отношений, **что позволяет** найти более релевантные темы (пункт 5 паспорта специальности 1.2.1, страницы 75-83 диссертации);

4. Разработан алгоритм персонализированного поиска веб-сайтов, **отличающийся** применением весовых коэффициентов в формуле ранжирования документов, учитывающей иерархические уровни онтологического профиля пользователя, **позволяющий** снизить процент ошибки при решении задачи персонализации веб-контента (пункт 4 паспорта специальности 1.2.1, страницы 83-95 диссертации).

Теоретическая значимость работы заключается в разработке модели, метода и алгоритмов обработки естественно-языковых текстов, которые обеспечивают: (1) эффективную обработку и анализ текстов на естественном языке; (2) извлечение и фильтрацию ключевых фраз, а также устранение их неоднозначности; (3) исследование и разработку средств персонализации контента веб-сайтов. Полученные научные результаты существенно развивают методологический аппарат в области искусственного интеллекта и машинного обучения, предлагая новые решения важной научной задачи увеличения объема релевантной текстовой информации в условиях семантической неоднозначности.

Практическая значимость заключается в разработке и реализации модели и алгоритмов в виде программного приложения, позволяющего использовать разработанные модель, метод и алгоритмы обработки и анализа текстов на естественном языке для минимизации процента ошибки при решении задачи персонализации веб-контента.

Положения, выносимые на защиту:

1. Модифицированный метод извлечения ключевых фраз из отдельных документов, на основе частоты их встречаемости и контекстуальной

семантической близости позволяет извлекать наиболее релевантные ключевые фразы;

2. Модифицированный алгоритм устранения неоднозначности смысла слов на основе контекстных векторных представлений и расширенных аннотаций синсетов позволяет снизить процент ошибки при определении правильного значения неоднозначного слова;

3. Онтологическая модель для представления индивидуальных особенностей интересов пользователя позволяет найти более релевантные темы;

4. Алгоритм персонализированного поиска веб-сайтов позволяет снизить процент ошибки при решении задачи персонализации веб-контента.

Реализация и внедрение результатов работы. Теоретические и практические результаты работы внедрены в информационные процессы ИТ-компании ООО «Ит-Эффект». Полученные в работе научные результаты позволили повысить эффективность решения задачи персонализации контента веб-сайтов в условиях семантической неоднозначности, включающей разработку комплекса алгоритмов извлечения ключевых фраз, устранения их неоднозначности и извлечения семантических отношений между ними. Результаты работы также используются в учебном процессе института компьютерных технологий и информационной безопасности Южного федерального университета.

Апробация результатов диссертации. Основные положения и отдельные результаты исследования докладывались и обсуждались на следующих конференциях: XVIII и XX Всероссийская конференция молодых ученых аспирантов и студентов «Информационные технологии, системный анализ и управление ИТСАУ» (Россия, Таганрог, 2019–2022); VI Всероссийская научно-техническая конференция «Фундаментальные и прикладные аспекты компьютерных технологий и информационной безопасности» (Россия, Таганрог,

2020); II Научно-методическая конференция НПР «Современные компьютерные технологии» (Россия, Таганрог, 2021 и 2022 гг.); XII Международная научно-техническая конференция «Технологии разработки информационных систем (ТРИС–2022)» (Россия, Таганрог, 2022); International Russian Automation Conference RusAutoCon, (Россия, Сочи, 2023); (Россия, Москва, 2022); «5th International Scientific Convention UCIENCIA» (Куба, Гавана, 2023); «6th International Scientific Convention UCIENCIA» (Куба, Гавана, 2025).

Публикации. По теме диссертации опубликовано **17** научных работ, из которых: **4** статьи опубликованы в издании из перечня рекомендованных ВАК (К2), в т.ч. 2 статьи опубликованы без соавторов; **3** статьи – в изданиях из международных баз данных Scopus и/или Web of Science. Получены **2** свидетельства о государственной регистрации программ для ЭВМ. В трудах всероссийских и международных конгрессов и конференций опубликовано **10** работ.

Личное участие соискателя в получении результатов. Все выносимые на защиту результаты и положения, составляющие основное содержание диссертационной работы, разработаны и получены лично автором или при его непосредственном участии. В работах, опубликованных в соавторстве, соискателю принадлежит определяющая роль в развитии аппарата ИИ моделей и методов обработки и анализа текстов на естественном языке.

Структура и объем работы. Диссертация состоит из введения, **4** разделов, заключения, списка литературы, содержащего **139** наименований, и **2** приложений. Основная часть работы содержит **142** страниц, включая **24** рисунка и **11** таблиц.

Во введении сформулирована цель работы, обоснована актуальность темы диссертации, описаны основные научные положения, выносимые на защиту, научная новизна, теоретическая и практическая ценность, апробация диссертационной работы, реализация и внедрение, а также структура диссертации.

Первая глава диссертационной работы посвящена анализу исследуемой предметной области. Проведен аналитический обзор научных исследований в области обработки и анализа текстов на основе методов искусственного интеллекта и машинного обучения. Также рассмотрены существующие методы персонализации контента веб-сайтов. Основное внимание уделяется методам извлечения ключевых фраз и устранения их неоднозначности. Также проведен анализ публикаций, тесно связанных с темой диссертации. Даны формализованные постановки основных задач исследования и сделаны выводы по разделу в целом.

Во второй главе описана разработка модифицированного метода извлечения ключевых фраз, а также модифицированного алгоритма устранения неоднозначности слов. Разработан модифицированный метод извлечения ключевых фраз, который использует новую функцию взвешивания, учитывающую частоту появления фраз в тексте и их контекстуальную семантическую близость к контексту документа. Это позволяет извлекать наиболее релевантные ключевые фразы. Разработан модифицированный алгоритм устранения неоднозначности смысла слов, который отличается применением функции определения контекстуальной семантической близости и расширенных аннотаций синсетов, что позволяет снизить погрешность определения правильного значения неоднозначного слова в данном контексте. Использование одного и того же метода контекстуального представления текстов как для извлечения ключевых фраз, так и для устранения неоднозначности слов не только снижает вычислительные затраты, но и гарантирует, что процесс устранения неоднозначности основан на том же контекстуальном понимании, которое использовалось для извлечения ключевых фраз, что приводит к более точной интерпретации интересов пользователя.

Третья глава посвящена разработке онтологической модели и нового алгоритма персонализации, построенных на основе ранее предложенных автором модифицированного метода извлечения ключевых фраз и модифицированного алгоритма устранения неоднозначности слов. Онтологическая модель обеспечивает представление индивидуальных характеристик пользовательских интересов. Она характеризуется введением новой функции для расчёта весовых коэффициентов, а также способностью извлекать контекстно-зависимые отношения, что позволяет выявлять наиболее релевантные темы. Алгоритм персонализации применяет весовые коэффициенты в формуле ранжирования документов, учитывая иерархические уровни онтологического профиля пользователя, что позволяет снизить процент ошибки при решении задачи персонализации веб-контента. Алгоритм описывает всеобъемлющий рабочий процесс, который начинается с извлечения ключевых фраз, за которым следует устранение неоднозначности значения слов в зависимости от контекста, и завершается объединением взвешенных ключевых фраз в отдельные "темы интересов".

Четвертая глава диссертации посвящена описанию разработки программного приложения и проведения вычислительного эксперимента для сравнения эффективности разработанных модели, метода и алгоритмов персонализации контента веб-сайтов. Проведенные эксперименты подтвердили эффективность разработанных модели и алгоритмов построения пользовательского профиля и персонализации результатов поиска веб-страниц. Результаты также подтверждают эффективность алгоритма персонализации в задаче адаптации контента к индивидуальным предпочтениям пользователя. В среднем предложенные автором решения снижают процент ошибок функционирования систем персонализации на 5 – 8 %.

В заключении изложены итоги выполненного исследования, рекомендации, перспективы дальнейшей разработки темы.

В приложениях приведены свидетельства об официальной регистрации программ для ЭВМ и копии актов внедрения.

ГЛАВА 1. АНАЛИТИЧЕСКИЙ ОБЗОР СОВРЕМЕННЫХ ПОДХОДОВ К ПЕРСОНАЛИЗАЦИИ КОНТЕНТА ВЕБ-САЙТОВ НА ОСНОВЕ МЕТОДОВ ОБРАБОТКИ И АНАЛИЗА ТЕКСТОВ

В данной главе проводится аналитический обзор современных исследований в области веб-инженерии, уделяется особое внимание методам искусственного интеллекта для персонализации контента веб-сайтов под индивидуальные предпочтения пользователей. Глава начинается с описания базовых концепций и основных этапов реализации процессов персонализации с учетом их взаимосвязи с веб-инженерией. Далее рассматривается возможность применения технологий искусственного интеллекта, в том числе методов обработки естественного языка (англ. Natural Language Processing, NLP) для персонализации контента веб-сайтов. Эти методы, включая извлечение и ранжирование ключевых фраз, устранение лексической неоднозначности (WSD), а также автоматизацию построения онтологий из текста, формируют основу для дальнейших исследований. В конце главы дана постановка задачи персонализации контента веб-сайтов.

1.1. Персонализация веб-контента в составе технологии Веб-Инженерии

Веб-инженерия как научно-практическая дисциплина прошла эволюцию от обеспечения надежности статических гипертекстовых документов (Web 1.0) к проектированию сложных интерактивных платформ, управляемых данными (Web 2.0). Накопление беспрецедентных объемов поведенческих данных создало техническую основу для перехода к качественно новому этапу – интеллектуальной персонализации, которая становится не дополнительной функцией, а системным свойством современных веб-приложений. Данная парадигма знаменует сдвиг в целеполагании веб-инженерии: от построения

платформ, реагирующих на явные действия пользователя, к созданию проактивных систем, способных на основе анализа поведенческих паттернов неявно выводить его интересы и прогнозировать информационные потребности. В этом контексте персонализация выступает интегратором двух фундаментальных для веб-инженерии дисциплин – информационного поиска (IR), смещающего фокус с релевантности документа на релевантность для конкретного пользователя, и информационной архитектуры (IA), обеспечивающей переход от универсальных навигационных схем к адаптивным структурам, подстраивающимся под индивидуальную когнитивную модель. Методологический аппарат веб-инженерии позволяет органично встроить в жизненный цикл разработки ключевые механизмы семантического анализа: извлечение ключевых фраз, разрешение лексической неоднозначности (WSD) и автоматическое построение динамических онтологий. Последние функционируют как «семантический мост», преобразующий неструктурированный текст в концептуальную сеть, насыщенную иерархическими и ассоциативными связями, что создает основу для глубокого понимания контекста интересов пользователя.

1.1.1. Понятие Веб-инженерии и персонализации

Веб-инженерия (англ. Web-engineering) – это дисциплина, которая фокусируется на разработке, внедрении и поддержке высококачественных веб-систем. Она включает в себя применение принципов, методологий и техник инженерии для проектирования, создания и управления веб-приложениями, веб-сайтами и другими веб-решениями [4, 31].

Веб-инженерия охватывает различные аспекты, включая разработку программного обеспечения, веб-дизайн, удобство использования, безопасность, оптимизацию производительности и управление проектами. Она также включает использование языков программирования, фреймворков, баз данных и других технологий для создания динамических и интерактивных веб-приложений.

Целью веб-инженерии является создание веб-систем, которые удовлетворяют потребностям пользователей и обеспечивают бесперебойное применение. Это включает понимание требований пользователей, проектирование пользовательских интерфейсов, разработку функциональности бэкэнда и обеспечение надежности, масштабируемости и поддерживаемости веб-систем. Веб-инженерия также включает такие аспекты, как оптимизация поисковых систем (англ. Search Engine Optimization, SEO), доступность, совместимость с различными браузерами и адаптивность для мобильных устройств, чтобы обеспечить эффективный доступ и использование веб-приложений широким кругом пользователей [4]. В целом веб-инженерия объединяет технические навыки, принципы дизайна и техники управления проектами для создания надежных, удобных для пользователя и эффективных веб-решений.

Веб-персонализация является одним из приложений веб-инженерии, включающим в себя набор информационных процессов, предназначенных для разработки, управления и доставки контента на основе известной, наблюдаемой и прогнозируемой информации [8, 32, 33]. **Персонализация** выступает в качестве ключевого элемента многих интеллектуальных информационных систем и их приложений, включая социальные сети и рекомендательные системы. Она направлена на формирование индивидуализированного пользовательского взаимодействия на основе анализа предпочтений, поведения и других релевантных данных пользователя. Она включает динамическую адаптацию веб-контента, рекомендаций и пользовательских интерфейсов для соответствия конкретным потребностям и интересам каждого пользователя [34, 32]. Следовательно, персонализация – важная часть технологий поиска и обработки информации в рамках междисциплинарной веб-инженерии (рис. 1.1).

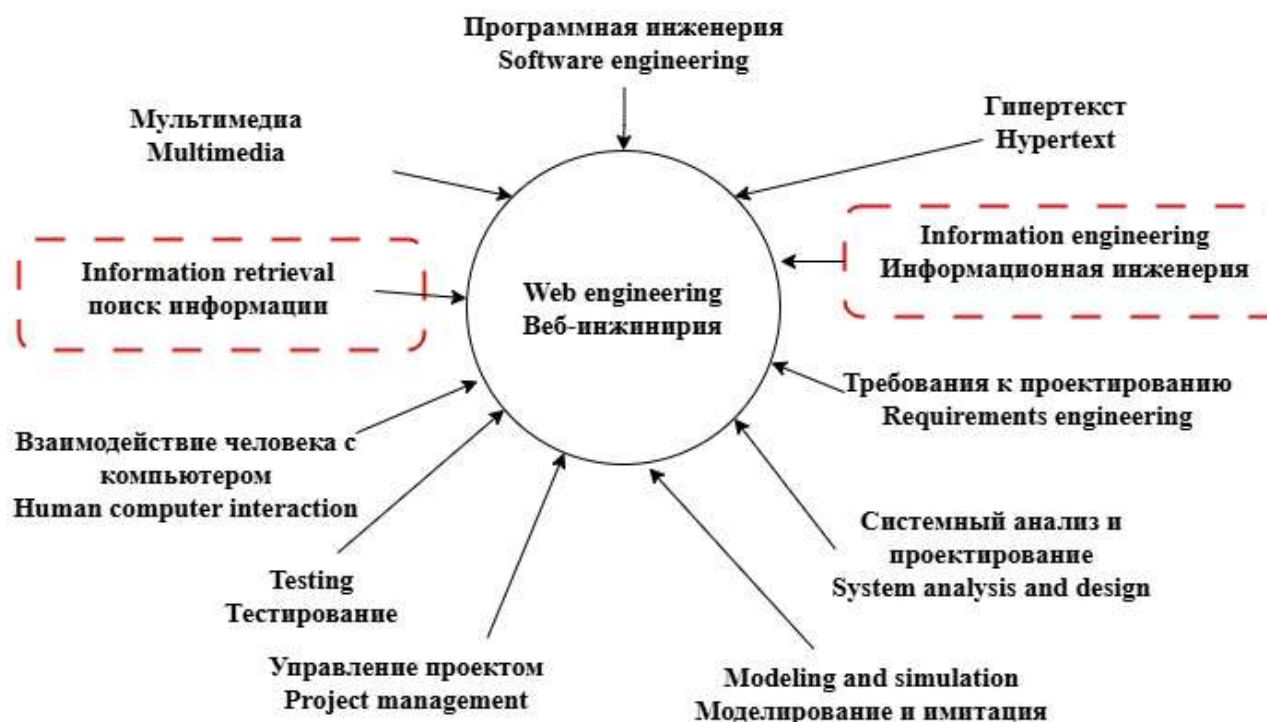


Рисунок 1.1 – Взаимосвязь между персонализацией и веб-инженерии

Современные системы веб-персонализации могут быть условно разделены на две фундаментальные категории, различающиеся по типу используемых данных и методов обработки [7, 35]:

Подход, основанный на явных знаниях, предоставленных пользователем, опирается на явно выраженные знания о предметной области и атрибутивные данные пользователей, такие как демографическая информация или результаты анкетирования. Эти системы используют жесткие правила логического вывода, заданные экспертами, например: "если профессия пользователя – врач, то рекомендовать медицинские статьи". Основными преимуществами данного подхода являются прозрачность принимаемых решений и возможность строгого контроля качества рекомендаций. Однако такие системы обладают существенным ограничением - они статичны и не способны адаптироваться к изменениям в

поведении пользователей. Типичным примером применения может служить рекомендация финансовых продуктов на основе возраста и уровня дохода клиента.

Второй класс образует подход, основанный на поведении (Behavior-Based Personalization). Данный подход анализирует неявные данные о поведении пользователей, включая клики, время просмотра и навигационные паттерны. В отличие от систем первого типа, здесь применяются методы машинного обучения для выявления сложных зависимостей в данных. Среди используемых моделей можно выделить анализ частотных и последовательных паттернов, применение глубоких нейросетевых архитектур и построение графовых моделей взаимодействий. Источниками данных служат как неявные показатели (история просмотров, текстовый контент и д.р.), так и редкие случаи явных оценок, таких как «лайки» или рейтинги. Практическим примером реализации этого подхода является динамическая адаптация новостной ленты на основе анализа времени, которое пользователь уделяет просмотру статей различных тематик.

В рамках данной диссертации особое внимание уделяется созданию методов обработки и анализа текстов на естественном языке для повышения эффективности систем поведенческой персонализации. Ключевое отличие предлагаемых решений заключается в глубокой семантической интерпретации потребляемого пользователем контента, что позволяет выявлять содержательные закономерности, а не только статистические зависимости. Такой анализ включает извлечение тематических профилей из текстов, а также построение динамических онтологий интересов.

Критерии оценки персонализации. Задача определения *релевантности контента* для пользователя является крайне сложной, поскольку его удовлетворенность не может быть измерена без обратной связи. Ученые выделяют два типа релевантности в информационном поиске [36, 37]. *Содержательная релевантность* отражает степень удовлетворения пользователя найденной

информацией и оценивается через поведенческие факторы и обратную связь [38]. *Формальная релевантность*, используемая в моделях информационного поиска, определяется на основе близости между формальными представлениями запроса (или профиля) и документа, либо через вероятность появления запроса в тексте. Она рассчитывается с использованием формул, основанных на алгоритмах машинного обучения или инженерии знаний. Одной из наиболее распространенных формул является косинусная мера сходства, которая вычисляется следующим образом [39]:

$$s = \text{similarity}(U, V) = \cos(\theta) = \frac{U \cdot V}{\|U\| \cdot \|V\|} = \frac{\sum_{i=1}^n U_i V_i}{\sqrt{\sum_{i=1}^n U_i^2} \sqrt{\sum_{i=1}^n V_i^2}}, \quad (1)$$

где U_i и V_i – компоненты вектора X (вектор запроса) и Y (вектор документа), соответственно.

В алгоритмах информационного поиска применяется именно формальная релевантность, что обусловлено несколькими причинами. Во-первых, содержательная релевантность субъективна и непостоянна [40]: ее оценка варьируется между разными пользователями и даже у одного пользователя в разное время [41]. Во-вторых, пользователи часто не могут заранее сформулировать ожидаемый результат, а их информационная потребность выражается только через запрос, без последующей оценки [42, 43].

В данном пункте автор привел определения основных терминов необходимых для понимания предметной области диссертационного исследования. В следующем пункте описываются этапы процесса персонализации веб-контента, с указанием методов и алгоритмов обработки и анализа текстов, применяемых на каждом этапе.

1.1.2. Этапы процесса персонализации веб-контента

Процесс персонализации веб-контента включает в себя ряд шагов, которые позволяют предоставлять пользователям индивидуализированные впечатления на

основе их предпочтений и поведения [44]. На рисунке 1.2 представлена последовательность этапов процесса в системе персонализации веб-контента на основе методов искусственного интеллекта и машинного обучения, которые подробно описаны ниже.

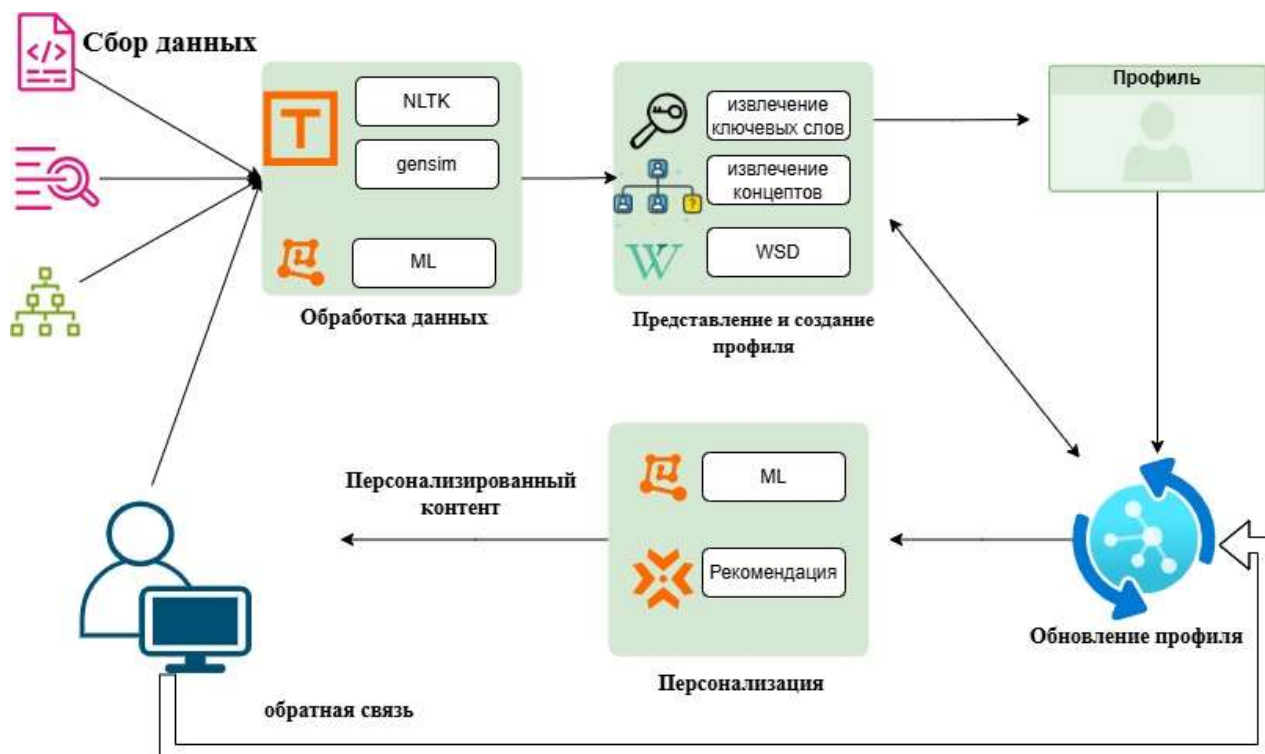


Рисунок 1.2 – Этапы процесса персонализации веб-контента

1. Сбор данных. Первый этап включает в себя сбор необходимых данных о пользователе. Это может включать как явные данные, предоставленные пользователем при регистрации или в ходе опросов, так и неявные данные, собранные в результате взаимодействия пользователя с веб-сайтом или приложением. Неявные данные могут включать историю просмотров, поисковые запросы, клики, историю покупок и активность в социальных сетях.

Этот этап не требуется реализовывать, поскольку сведения, используемые в диссертации, представляют собой стандартный набор AOL4PS [6], который является коллекцией данных о взаимодействии пользователей с веб-сайтами и специально создан для исследовательских целей.

2. Обработка данных: после сбора данные должны пройти обработку для удаления шума, заполнения отсутствующих значений и приведения в необходимый формат для анализа. Этот этап включает такие процессы, как очистка данных, нормализация и создание признаков. Для реализации этих операций необходим набор инструментов, в который входят следующие:

- *NLTK (Natural Language Toolkit)* [45] представляет собой библиотеку Python, предназначенную для обработки естественного языка (NLP). Она предоставляет разработчикам широкий набор инструментов и ресурсов для выполнения различных задач, связанных с анализом текстовых данных. *NLTK* включает в себя функции для токенизации (tokenization), стемминга (stemming), лемматизации (lemmatization), морфологического и синтаксического анализа, а также анализа настроений.

1. Токенизации. Разделение текста на отдельные слова или фразы.

2. Стемминг. Этот процесс сводит слова к их базовой форме, удаляя аффиксы (например, окончания множественного числа у существительных или -ing у глаголов). После стемминга каждое слово представлено своей основой [46, 47].

3. Лемматизация. В отличие от стемминга, лемматизация преобразует слова в их словарную форму, учитывая морфологический анализ. Это делает её более точной, но менее быстрой, чем стемминг. Оба метода играют ключевую роль в предобработке текста, уменьшая размер словаря и упрощая дальнейшую обработку. Стемминг часто применяется для больших текстовых объемов и малых словарей, тогда как лемматизация предпочтительней для небольших документов с большим словарным пространством. Выбор метода зависит от задачи и характеристик данных [48].

4. Удаление стоп-слов: исключение распространенных слов, которые не несут значимой информации.

Библиотека NLTK предлагает доступ к более чем 50 корпусам текстов и лексическим ресурсам, таким как *WordNet*, что позволяет работать с большими объемами данных на разных языках;

- **WordNet** – это лексическая база данных английского языка, разработанная в Принстонском университете [49]. Она структурирована как семантическая сеть, где слова группируются в наборы синсетов (синонимов), а между этими наборами устанавливаются семантические отношения. Для каждого концепта WordNet предоставляет список различных синсетов, каждый из которых имеет определение (англ. gloss), объясняющее его значение, а также пример использования этого конкретного смысла концепта в языке.

WordNet широко применяется в задачах обработки естественного языка: при анализе текста, поиске синонимов, лемматизации, разрешении неоднозначностей и в ряде других задач. Его основные особенности таковы:

1. синсеты (synsets) – группы слов, выступающих синонимами в определённом контексте. Так, синсет для слова "car" может включать automobile, motorcar, auto;
2. семантические отношения – WordNet задаёт связи между синсетами: гиперонимы (hypernyms), гипонимы (hyponyms), меронимы (meronyms), холонимы (holonyms), антонимы (antonyms) и синонимы (synonyms);
3. части речи – слова разделены на четыре основные категории: существительные, глаголы, прилагательные и наречия. У каждой категории своя структура синсетов и отношений.

При всех этих достоинствах у WordNet есть и слабые стороны. Во-первых, он ограничен английским языком, хотя для других языков существуют аналоги, например RussNet для русского [50]. Во-вторых, современная и разговорная лексика в нём представлена не всегда. В данной диссертации WordNet

используется для разработки модифицированного алгоритма решения задачи устранения неоднозначности слов (Word Sense Disambiguation, WSD);

- WikiData [16] – структурированная база знаний, в которой информация представлена в виде сущностей (элементов) и их свойств (утверждений). У каждой сущности есть уникальный идентификатор (QID), и она связана с другими сущностями через отношения, что и делает WikiData семантической сетью. База охватывает широкий круг тем – людей, места, события, концепции – и поддерживает многоязычность.

Все перечисленные инструменты задействованы в данной диссертации на этапе обработки данных при реализации предложенных метода и алгоритмов. WordNet применяется главным образом в алгоритме устранения неоднозначности слов как источник толкований смыслов, тогда как Wikidata служит источником семантических связей между сущностями на этапе представления и построения профиля пользователя. Этот этап подробно рассматривается в следующем подпункте.

3. Представление и создание профиля

Без профиля пользователя система персонализации работать не может: именно в нём фиксируются информационные потребности и предпочтения, и от его качества во многом зависит эффективность рекомендаций. Представлять такие профили можно по-разному, и у каждого способа есть свои преимущества и свои сложности [51].

Как правило, извлечённые для построения профиля характеристики делятся на концепты и/или ключевые слова. Далее они представляются одним из трёх способов: как вектор в пространстве, как семантическая сеть или как граф.

Векторная модель – наиболее распространённый способ представления интересов пользователя [52, 53, 39]. В ней используются взвешенные векторы ключевых слов (или концептов), которые чаще всего извлекаются из документов,

просмотренных пользователем. Эти ключевые слова или концепты отражают его интересы и могут напрямую применяться для расширения запросов. Интересы при этом представляются либо единым вектором, включающим все интересы сразу, либо набором векторов, каждый из которых отвечает за отдельную область. Модель проста в реализации и вычислительно эффективна. Её слабое место – полисемия (одно слово имеет несколько значений) и синонимия (разные слова имеют схожие значения), из-за которых точность может снижаться.

Семантическая сеть. Подход на основе семантических сетей снимает ограничения профилей, построенных на ключевых словах: интересы пользователя представляются в виде взвешенных ключевых слов или концептов, встроенных в семантическую сеть. Каждый узел такой сети соответствует концепту, а связи между узлами отражают семантические отношения. Чтобы построить подобную модель, слова, описывающие интересы, нужно сопоставить с концептами – с помощью существующих онтологий вроде WordNet, средствами машинного обучения или вручную. Проблемы полисемии и синонимии этот метод решает хорошо, но строить и поддерживать такую модель заметно сложнее. К тому же она зависит от наличия хорошо определённых онтологий либо требует значительных ручных усилий.

Иерархическая модель. Здесь интересы пользователя представлены узлами концептов, а связи между ними отражают иерархические отношения – например, более широкий или более узкий концепт. Уровень интереса пользователя к каждой теме определяется с помощью различных механизмов взвешивания. Такое представление получается структурированным и наглядным, однако требует заранее заданной чёткой иерархии, а это ограничивает гибкость по сравнению с семантическими сетями.

Граф знаний (Knowledge graph, KG) – ещё один способ представления профиля пользователя, объединяющий преимущества семантических сетей и

иерархических моделей. Графы знаний существуют в двух основных формах: без атрибутов и с атрибутами. В простейшем случае граф состоит только из сущностей (узлов), обозначающих объекты, понятия или события (например, "Альберт Эйнштейн"), и связей (рёбер), отражающих отношения между ними (например, "разработал"). Такие графы легко обрабатывать, и они хорошо подходят для задач, где важна одна лишь топология связей, как в анализе социальных сетей. Однако выразительность у них ограничена: дополнительные свойства элементов хранить негде.

Более развитая версия – графы знаний с атрибутами – расширяет базовую структуру метаданными, описывающими свойства как сущностей, так и отношений. Этот подход позволяет не только учитывать семантические связи, но и обогащать профиль дополнительными атрибутами и контекстом. Графы знаний особенно удобны для хранения сложных взаимосвязей между концептами и могут автоматически пополняться по мере поступления новых данных. Платой за это становятся значительные вычислительные ресурсы и потребность в качественных алгоритмах извлечения информации.

В данной диссертации профиль пользователя строится на основе онтологической модели (онтологии). Онтология представляет собой формальную, явную спецификацию разделяемой концептуализации [54], которая определяет классы, свойства, отношения и ограничения в рамках конкретной предметной области. Стандарты вроде OWL (Web Ontology Language) служат базовыми примерами для семантических веб-приложений. Граф знаний, в отличие от онтологии, – это уже практическая реализация: сеть реальных сущностей (узлов), связанных отношениями (рёбрами). Известные примеры – Google Knowledge Graph и Wikidata [16].

Иными словами, онтология задаёт структурную схему через определённые классы и свойства, а граф знаний наполняет эту схему конкретными данными.

Хорошая иллюстрация такого взаимодействия – онтология Schema.org [55], лежащая в основе структуры Wikidata: последняя использует RDF-подобную онтологическую структуру и обогащает её фактическими данными.

4. Обновление профиля. Профиль пользователя не статичен: его нужно постоянно обновлять, чтобы он отражал изменения в предпочтениях и поведении. На этом этапе отслеживается взаимодействие пользователя с системой, собираются новые данные и профиль корректируется. Используются те же методы, что и на этапе сбора данных и построения профиля: анализ и обработка описаний объектов, извлечение характеризующих их ключевых фраз и устранение их неоднозначности.

5. Персонализация. Последний этап – применение профиля для персонализации веб-контента или рекомендаций. Рекомендации, подстроенные под потребности пользователя, формируются на основе информации, полученной при анализе профиля, с помощью нескольких техник. Прежде всего это методы оценки близости или семантической схожести, определяющие актуальность новых объектов для конкретного пользователя. Кроме того, применяются алгоритмы машинного обучения, которые позволяют строить автоматизированные модели персонализации и выявлять скрытые паттерны как в данных самого пользователя, так и в данных других пользователей. Опираясь на эти паттерны, система выдаёт рекомендации, соответствующие предпочтениям и требованиям пользователя.

Из сказанного видно, что без методов обработки и анализа текстов на естественном языке персонализация контента невозможна. Что бы ни выступало объектом персонализации – тексты, видео, аудиофайлы или любые другие продукты, – у всех них есть описания, либо их можно привести к текстовому представлению. Именно эти текстовые описания затем анализируются и используются в процессе персонализации.

В данном пункте были даны определения основных понятий предметной области. Изучены этапы процесса персонализации веб-контента. Раскрыта основная информация для дальнейшего выбора направлений исследования. В следующем пункте проводится аналитический обзор основных подходов к созданию и представлению профиля пользователя.

1.2. Обзор методов построения профиля пользователя

В данном пункте описываются основные методы построения профиля пользователя, которые включают две основные категории: методы, основанные на концептах, и методы, основанные на онтологических моделях.

1.2.1. Методы, основанные на концептах

В контексте методов, основанных на концептах, *концепт* относится к семантически значимой единице, извлеченной из данных, созданных пользователем (например, ключевых слов, фраз или терминов), которая представляет конкретный интерес, тему или область. Концепты обычно извлекаются из неструктурированного текста (например, поисковых запросов, документов, постов в социальных сетях) и рассматриваются как индикаторы интересов пользователя. Например, термин "нейронные сети" может быть извлечен как концепт, представляющий интерес пользователя к машинному обучению.

В литературе существуют две основные категории работ, затрагивающих применение концепций при построении пользовательских профилей. Далее следует разбор каждой из них с примерами соответствующих исследований.

Методы, основанные только на ключевых словах. Эти методы полагаются исключительно на ключевые слова, извлеченные из пользовательских данных (например, поисковые запросы, документы), не используя никаких внешних

источников знаний. Они извлекают ключевые слова и выполняют анализ частотности терминов для выявления важных терминов из пользовательских данных [11]. Хотя такие подходы просты и понятны, они часто испытывают трудности с неоднозначностью (например, полисемией) и не обладают необходимой семантической глубиной для точного представления интересов пользователей [7–9, 56].

Методы, усиленные внешними данными. Эти методы улучшают представления на основе ключевых слов, связывая ключевые слова с внешними ссылками, такими как категории, таксономии или базы знаний (например, WordNet, ODP). Отображая ключевые слова на структурированные источники, эти подходы могут разрешить неоднозначности и обогатить семантическое значение концепций. Например, термин "Java" может быть уточнён до "языка программирования", а не "острова" или "кофе", ссылаясь на внешнюю таксономию ODP.

Наиболее распространенным представлением является набор взвешенных ключевых слов или терминов, которые могут быть извлечены из описания продуктов или предоставлены непосредственно пользователями [7, 8]. Веса, которые обычно ассоциируются с ключевыми словами, представляют собой численные значения интересов пользователя. Каждое ключевое слово может представлять тему интереса, или ключевые слова могут быть сгруппированы в категории, чтобы отразить более стандартное представление интересов пользователя.

Степень интереса пользователя к этим ключевым словам часто взвешивается с использованием моделей, таких как TF-IDF (Frequency-Inverse Document Frequency) в векторном подходе [57, 9]. TF-IDF широко используется, благодаря своей простоте и эффективности. Алгоритм сосредоточен на определении важности слов в документах относительно коллекции документов (корпуса) и

может быть адаптирован для представления пользовательских предпочтений на основе их взаимодействия с контентом. TF-IDF состоит из двух основных компонентов, первый – это частота появления термина (TF), которая указывает на то, как часто термин появляется в документе. Чем чаще термин встречается в документе, тем более важным он считается для этого документа. Обратная частота документа (IDF), показывает, насколько важен термин в рамках всего корпуса. Данный показатель снижает вес общих терминов, которые встречаются во многих документах. Оценка $TF-IDF$ для термина t в документе d затем вычисляется следующим образом:

$$tf-idf(t, d, D) = tf(t, d) \cdot idf(t, D) = \frac{n_t}{\sum_k n_k} \cdot \log \frac{N}{|\{D_i \in D | t \in D_i\}|}, \quad (2)$$

где N – общее количество документов в корпусе.

Таким образом, вектор пользовательского профиля создается путем агрегирования значений TF-IDF документов (D), с которыми пользователь взаимодействовал. Это можно сделать, суммируя значения TF-IDF для каждого термина по всем документам, которые посетил пользователь:

$$User Profile(t) = \sum_{d_i \in D} tf-idf(t, d, D) \quad (3)$$

Документы, которые система извлекает в ответ на поисковый запрос, преобразуются в векторы ключевых слов с соответствующими весами. Эти векторы затем сравниваются с профилем пользователя с применением косинусной меры сходства (1). Только те документы, векторы которых наиболее близки к профилю, передаются пользователю.

Алгоритм персонализации на основе TF-IDF широко известен своей простотой, интерпретируемостью и масштабируемостью, а также эффективностью определения отличительных терминов для захвата предпочтений пользователя. Однако этот алгоритм имеет несколько ограничений, включая неспособность учитывать контекстное значение и ограниченное семантическое понимание по сравнению с более продвинутыми моделями [58]. Данный алгоритм использовался

в представленном исследовании в качестве аналога для сравнения с предложенным автором решением.

В [59] авторы усовершенствовали первоначальную терминологическую модель, разработав семантически улучшенную модель пользователя. В этой модели термины связываются с соответствующими высокоуровневыми концепциями. Для классификации обучающихся по их интересам и в соответствии с моделью Фелдера-Сильвермана применяется метод нечеткой логики. Авторы в [60] представили новый алгоритм извлечения данных, который сочетает в себе Word2Vec – инструмент для создания векторных представлений слов, выпущенный Google в 2013 году, и TF-IDF для формирования весового вектора. Этот подход позволяет преодолеть ограничения традиционных методов векторного представления.

В отличие от простых ключевых слов, концепты в работах [61, 62] призваны отражать более высокий уровень семантического значения. Для их верификации и обогащения обычно используется сопоставление с внешними источниками знаний, в частности – с Open Directory Project (ODP, также известным как DMOZ). ODP представлял собой многоязычный каталог веб-ссылок, который до своего закрытия в 2017 году широко применялся для персонализации поиска благодаря своей иерархической структуре категорий. Эта структура послужила основой для построения онтологий и классификации контента, что позволяло создавать более точные концептуальные профили пользователей.

Например, в [12], пользовательские интересы извлекаются из веб-документов и представляются в виде взвешенных категорий из ODP. В моделях, построенных с применением данных методов, отсутствуют семантические связи между концептами. Принцип работы методов на основе ODP следующий:

1. Документ представляется в виде векторной модели на основе терминов с использованием TF-IDF;

2. Документы сопоставляются с концептами из ODP с использованием метрик подобия, таких как косинусная мера сходства [39]. Пусть $C = \{c_1, c_2, \dots, c_N\}$ – это множество концептов из ODP. Каждый термин t в документе D сопоставляется с концептами c_i в C с помощью формулы (1);

3. Выбираются N наиболее значимых концептов, которые лучше всего отражают содержание документа;

$$C_{selected} = Top-N_c S(t, c_i) \quad (4)$$

4. Вес каждого концепта определяется его значимостью и релевантностью, рассчитанными на основе косинусной меры сходства;

В области семантического анализа структура ODP использовалась для обогащения контента, что улучшало понимание тематики веб-страниц и интересов пользователей. Это, в свою очередь, способствовало созданию более точных рекомендательных систем, основанных на семантических связях между интересами пользователя и содержанием веб-страниц. Хотя ODP больше не функционирует, его структура и принципы продолжают оказывать влияние на современные системы персонализации, использующие онтологическое моделирование и семантический анализ.

Несмотря на интерпретируемость и вычислительную эффективность, эти методы ограничены в своей способности учитывать базовую семантику контента. Они рассматривают слова как изолированные единицы и игнорируют контекстные связи между концептами. Данное ограничение становится особенно проблематичным при наличии в профиле терминов с междисциплинарной полисемией, когда одно слово приобретает различные значения в разных предметных областях.

Для преодоления ограничений существующих подходов, используется онтология для обогащения профиля пользователя семантической информацией. В

следующем разделе будут рассмотрены методы, основанные на онтологических моделях.

1.2.2. Методы, основанные на онтологических моделях

Методы, основанные на онтологических моделях, используют формальные онтологии для структурирования пользовательских профилей и структурируют интересы пользователей в виде онтологических концептов (графов знаний), используя формальные семантические отношения. Концепты формально определяются как узлы в графе знаний с явными отношениями (например, "is-a", "part-of") и атрибутами. Например, онтологический концепт "глубокое обучение" – это подкласс "машинного обучения" [63, 64].

Было представлено несколько работ, в которых используется этот подход. В [35, 7, 22] пользовательский профиль представлен ориентированным графом, где узлы (V) соответствуют концептам, взвешенным по их значимости для пользователя. Ребра (E), соединяющие эти узлы, делятся на три категории:

1. Связи типа *"is-a"*, отражающие иерархические отношения;
2. Перекрёстные символические связи, представляющие специфические ассоциации;
3. Связи, обозначающие *общие ассоциативные отношения*.

Для построения профиля пользователей документы, полученные в результате поиска, также преобразуются в графы. Содержимое документа анализируется для извлечения терминов, которые затем сопоставляются с концептами из онтологии. На основе этого сопоставления формируется граф документа с узлами-концептами и ребрами, отражающими их взаимосвязи.

Эффективность этих алгоритмов напрямую зависит от качества и полноты онтологии. Если онтология недостаточно «глубокая» или не охватывает интересы пользователя, это может привести к неточным профилям и снижению качества рекомендаций. Кроме того, построение и поддержание графовых представлений

для большого объема данных требует значительных вычислительных ресурсов, что может негативно сказаться на производительности системы в реальном масштабе времени [65].

В работах [66, 64] для персонализации контента применялась справочная онтология (таксономия), разработанная экспертами. Она использовалась в качестве источника описания элементов для формирования профиля пользователя.

В [67] были созданы семантически улучшенные профили пользователей, которые отражают интересы и намерения пользователей в рамках набора определенных категорий. В [68] предложена онтологическая модель пользователя, которая позволяет большому количеству различных приложений взаимодействовать с одной и той же базой знаний на семантическом уровне, а также представляет интересы в виде онтологических концепций, связанных с предопределенной исходной онтологией с использованием концепций из Wikipedia и WordNet. В работе [69] описано улучшение онтологического профиля пользователя в мобильных средах для контекстно-ориентированных систем. В [70] предложен метод улучшения представления данных в социальных веб-платформах с использованием онтологического представления знаний, что позволяет осуществлять семантическое профилирование на социальных сайтах.

Однако такие модели часто опираются на статические, предопределённые онтологии, которые невозможно адаптировать к динамичному характеру предпочтений пользователей или к новым предметным областям. В [13] пользовательские профили создаются на основе URL-адресов и сопоставляются с категориями из онтологий *OpenDNS* и *DBpedia* [16, 17]. Этот метод обогащает пользовательские профили семантическими связями, но может не обеспечивать точности в захвате конкретных интересов. Кроме того, гибридные методы интегрируют предметно-ориентированные онтологии с методами коллаборативной фильтрации для повышения точности рекомендаций.

Важно отметить, что, несмотря на использование онтологии для построения концептов из текста, процесс идентификации ключевых терминов по-прежнему опирается на TF-IDF. Эта модель учитывает только частотность употребления слов, игнорируя их семантическое значение и контекстуальные связи. Это ограничение снижает эффективность последующего семантического анализа и точность построения концептуальной модели текста.

В результате, *современные методы построения профилей пользователей по-прежнему недостаточно эффективны для обработки больших объемов текстовой информации и извлечения знаний из них*, что не позволяет эффективно представлять интересы пользователей в системах веб-персонализации и делает их неспособными снизить *когнитивную нагрузку* на пользователя.

Кроме того, независимо от метода представления пользовательского профиля, необходимо выполнить этап извлечения ключевых слов или фраз, описывающих элементы, с которыми взаимодействует пользователь. Эти ключевые слова затем обрабатываются и взвешиваются для формирования взвешенного профиля на основе ключевых слов или сопоставляются с концептами или референтной онтологией для получения профиля на основе концептов. Кроме того, крайне важно применить этап устранения неоднозначности значений слов, чтобы определить наиболее подходящее значение для контекста, в котором появляется слово или концепт. Этот процесс позволяет точнее интерпретировать интересов пользователя и улучшить качество персонализации.

В данном пункте описаны основные подходы к персонализации контента веб-сайтов с использованием технологий искусственного интеллекта. Рассмотрены ограничения текущих подходов и определено направление исследований, позволяющее эти ограничения устранить.

В следующем пункте описаны методы обработки и анализа текстов на естественном языке, используемые при разработке алгоритмов и методов,

предложенных в этом исследовании для повышения эффективности системы веб-персонализации на основе искусственного интеллекта.

1.3. Методы извлечения ключевых фраз, используемые в задачах персонализации

Задача извлечения релевантных ключевых слов (фраз) из документов существует давно [71], и решения её оказались очень важными для множества задач и сценариев, включая обобщение текста, рекомендацию, кластеризацию [72], построение тезауруса, обогащение предметных метаданных, категоризацию [73], поиск, индексацию [74] и расширение запросов [75].

Извлечение ключевых слов (фраз) – это автоматическое определение терминов, которые лучше всего описывают содержание документа [76]. Ключевые фразы¹ – это термины (т.е. выражения), которые лучше всего описывают предмет документа [77] [78].

Первые методы извлечения ключевых слов в основном следовали контролируемому подходу [79, 80]. При таком подходе сначала извлекаются характеристики ключевых слов, а затем классификатор обучается на эталонном наборе данных. Например, КЕА [80], контролируемый современный алгоритм извлечения ключевых слов, основанный на наивном байесовском алгоритме. Хотя эти методы имеют неплохую производительность [81, 82], они полагаются на эталонный набор данных и требуют (относительно) длительного процесса обучения. Кроме того, модель, обученная в определенной предметной области, не всегда дает хорошие результаты в других областях. Это контрастирует с общими неконтролируемыми методами, которые могут быть быстро применены к документу на разных языках или доменах с меньшими усилиями, поскольку они

¹ Ключевая фраза обычно содержит от одного до трех слов и в оставшейся части статьи, используя этот термин, также имеется в виду ключевое слово

не зависят от предметной области и не требуют размеченных обучающих данных. Их можно разделить на статистические методы и методы на основе графов.

Первые, такие как TF-IDF [83], YAKE [84], KP-Miner [85] и RAKE [86], используют статистические характеристики текстов для захвата ключевых слов. Методы, основанные на обработке графов, были представлены как альтернатива чисто статистическим методам. Наиболее известными примерами таких методов являются TextRank и его версии TopicRank, и SingleRank [87], которые строят графы для ранжирования слов в зависимости от их положения в графе.

Ниже приводится описание некоторых методов, использованных в данной диссертации для сравнения, а также тех, которые составляют основу для разработки. Учитывая ограничения, связанные с доступностью размеченных данных, методы, основанные на обучении с учителем, не являются предметом данной диссертации.

TF-IDF – это статистическая мера, используемая для оценки важности слов в документе относительно коллекции документов (корпуса) [83]. Для извлечения ключевых слов с использованием TF-IDF сначала вычисляются веса для каждого слова в документе. Это делается путем расчета значения TF-IDF (2), которое учитывает частоту термина в документе (TF) и его редкость во всех документах корпуса (IDF). После расчета весов слова ранжируются по их значениям TF-IDF. Слова с наибольшими значениями считаются ключевыми, так как они часто встречаются в текущем документе, но редко в других документах корпуса. На заключительном этапе выбираются топ-N слов с наивысшими весами, которые и представляют собой ключевые слова.

Метод YAKE (Yet Another Keyword Extractor) [84]. Помимо позиции/частоты термина, YAKE также использует новые статистические показатели, которые фиксируют контекстную информацию и распространение терминов в документе. Во-первых, YAKE предварительно обрабатывает текст,

разбивая его на отдельные термины. Во-вторых, для каждого термина рассчитывается набор из пяти признаков:

- W_{case} учитывает регистр написания слова;
- $W_{Position}$ оценивает позиционную значимость (слова в начале документа получают больший вес);
- W_{Freq} анализирует частоту встречаемости слова;
- W_{Rel} измеряет контекстную значимость (количество уникальных терминов в окружении слова);
- $W_{DifSentence}$ определяет распределение слова по предложениям документа

На основе этих показателей вычисляется интегральная оценка $S(w)$ для каждого термина, позволяющая ранжировать слова по их значимости в тексте.

$$S(w) = \frac{W_{Rel} \cdot W_{Position}}{W_{case} + \frac{W_{Freq}}{W_{Rel}} + \frac{W_{DifSentence}}{W_{Rel}}}. \quad (5)$$

Наконец, непрерывная последовательность ключевых слов-кандидатов размером 1, 2 и 3-граммы генерируется с использованием скользящего окна 3-граммы. Каждому ключевому слову-кандидату k_w присваивается следующая оценка:

$$S(kw) = \frac{\prod_{w \in kw} S(w)}{Tf(kw) \cdot (1 + \sum_{w \in kw} S(w))}. \quad (6)$$

Чем меньше оценка S , тем более значимым будет ключевое слово. В данном исследовании YAKE не используется для разработки, а только для сравнения с разрабатываемым алгоритмом.

RAKE (Rapid Automatic Keyword Extraction) [88] определяет ключевые слова, сегментируя текст на ключевые слова-кандидаты на основе разделителей, таких как стоп-слова и знаки препинания. Затем **RAKE** анализирует совместное появление слов в этих кандидатах, чтобы выявить их связи. Например, если "Natural" часто встречается с "Language", их сильная ассоциация может быть использована для оценки релевантности. **RAKE** присваивает каждому ключевому

слову-кандидату оценку на основе его частоты и совместного появления с другими словами:

$$Score(k) = \sum_{i=1}^n frequency(w_i) \cdot co - occurrence(k, w_i) \quad (7)$$

где k – это ключевое слово-кандидат, w_i – совместно появляющиеся слова, а n – общее количество рассматриваемых совместно появляющихся слов. Более высокие оценки указывают на более релевантные ключевые слова. Наконец, *RAKE* ранжирует ключевые слова по их оценкам. *RAKE* не используется для разработки, а только для сравнения с разрабатываемым алгоритмом.

TextRank [76] – Прежде всего, текст токенизируется и аннотируется тегами *POS*. Затем к текстовым единицам применяются синтаксические фильтры, т.е. сохраняются только существительные и прилагательные. Затем лексические единицы, которые проходят упомянутые выше фильтры, то есть кандидаты, добавляются к графу как узлы, и ребро добавляется между узлами, которые одновременно встречаются в окне из M слов. Граф неориентированный и невзвешенный. Первоначальная оценка, присвоенная каждому узлу, равна 1, а затем алгоритм PageRank выполняется до тех пор, пока он не сойдется. В частности, для узла V_i соответствующая функция оценки, которая повторно вычисляется, имеет следующий вид:

$$S(V_i) = (1 - \lambda) + \lambda \cdot \sum_{j \in N(V_i)} \frac{1}{N(V_j)} S(V_j), \quad (8)$$

где $N(V_i)$ – множество соседей V_i , $N(V_j)$ – множество соседей V_j , а λ – вероятность перехода от одного узла к другому. Как только алгоритм сходится, узлы сортируются по убыванию оценки. Улучшения *TextRank* включают *SingleRank* и *ExpandRank* [89] и более поздние версии *TopicRank*, которые извлекают релевантные ключевые слова для основных тем документов.

Извлечение ключевых слов на основе методов контекстного представления

В последние годы наблюдается значительный прогресс в области представления текста, особенно с появлением методов контекстного представления, таких как Elmo [90], представление двунаправленного кодировщика от трансформаторов (англ. Bidirectional Encoder Representations from Transformers, BERT) [82], Sentence-BERT (S-BERT) [91] и другие. Эти разработки привели к существенным улучшениям в различных задачах обработки естественного языка, и область извлечения ключевых фраз не стала исключением. Был разработан ряд инновационных методов, основанных на технологиях контекстного представления, которые показали многообещающие результаты. Методы на основе модели контекстного встраивания BERT использовались во многих работах для извлечения ключевых слов при контролируемом подходе [82].

В [92] авторы предложили метод извлечения ключевых слов с использованием контекстного векторного представления на основе BERT. Текст предварительно обрабатывается для удаления ненужных элементов, и извлекаются кандидаты на ключевые слова. Затем BERT используется для генерации контекстных представлений текста, и применяется косинусная мера сходства для определения наиболее релевантных ключевых слов.

В [93] авторы представили метод извлечения ключевых слов, который сочетает BERT с многофункциональным объединением признаков. Процесс включает два ключевых шага. Во-первых, BERT используется для извлечения ключевых предложений из основного текста научных статей, которые выбираются на основе их семантического сходства с аннотацией. Эти предложения предоставляют дополнительный контекст для аннотации. Во-вторых, комбинация TF-IDF, TextRank и LDA используется для извлечения ключевых слов, как из аннотации, так и из набора ключевых предложений. Лучшие ключевые слова из каждого метода объединяются для формирования окончательного набора ключевых слов.

В работах [94, 95] предложен упрощённый метод KeyBERT, который использует предобученную модель S-BERT без необходимости её дополнительного обучения. KeyBERT кодирует документ и его слова с помощью BERT [91], после чего вычисляет косинусную меру сходства между каждым словом и документом. Слова ранжируются по степени сходства, а для выбора наиболее релевантных фраз применяются дополнительные операции.

На основе проведенного в данном пункте анализа был выбран метод KeyBERT в качестве основы для разработки модифицированного метода извлечения ключевых фраз в настоящей работе. Подробное описание его недостатков и механизмов их устранения будет представлено во второй главе.

В следующем пункте рассматриваются методы устранения неоднозначности смысла слов и определяются ключевые из них для разработки алгоритмов в данной диссертации.

1.4. Методы устранения неоднозначности смысла слов

Устранение неоднозначности смысла слов WSD (англ. Word Sense Disambiguation) является открытым вопросом в компьютерной лингвистике, его задача состоит в том, чтобы автоматически определить конкретное значение многозначного слова (смысл) в данном контексте [96, 97]. Методы WSD используются практически во всех приложениях языковых технологий, таких, как персонализация, машинный перевод, поиск информации, интеллектуальный анализ текста и извлечение информации (IE, information extraction). При этом применяется в качестве неотъемлемой части извлечения концептов, что считается базовым этапом в таких задачах, как поиск информации, построение профилей пользователей при персонализации веб-контента и рекомендации на основе контента, в дополнение к его приложениям в задачах классификации и кластеризации текстов.

В зависимости от используемых источников знаний, для устранения лексической неоднозначности, методы WSD делятся на три категории: основанные на знаниях, с учителем и без учителя.

Методы WSD с учителем применяются для получения объективных оценок [98], однако их ключевым ограничением является необходимость в больших объемах размеченных данных, создание которых требует значительных временных и трудовых затрат.

Методы WSD без учителя (unsupervised WSD) не зависят от размеченных данных. Вместо этого они опираются на статистические закономерности, кластеризацию или анализ распределения значений слов в крупных текстовых корпусах. Эти подходы выявляют контекстуальные паттерны для автоматического определения смысла слов, что делает их более масштабируемыми, но менее точными по сравнению с методами с учителем.

Подходы, основанные на знаниях (knowledge-based WSD), используют информацию из лексических баз знаний (LKB, от англ. Lexical Knowledge Base), таких как WordNet, что позволяет выполнять WSD без привлечения дополнительных корпусных данных. Среди них выделяются две ключевые категории:

Подходы, основанные на графах. Эти подходы применяют хорошо известные графовые методы для анализа и использования структурных свойств графа, лежащего в основе конкретной лексической базы знаний (LKB, от англ. Lexical Knowledge Base). Их способность разрешать лексическую неоднозначность сразу для всех слов в последовательности, используя существующие семантические связи (рёбра) между понятиями (узлами), делает эти алгоритмы более фундаментальными по сравнению с методами, основанными на частичных пересечениях определений значений.

Однако, поскольку анализ проводится над всем графом целиком, такие методы требуют значительных вычислительных ресурсов, особенно при работе с крупными графами типа WordNet. Кроме того, они полностью игнорируют роль определений слов в процессе устранения лексической неоднозначности. Учитывая указанные недостатки, эти методы не подходят для предложенных решений в данной диссертации и будут использоваться только в качестве аналогов для сравнения с разработанными методами.

Методы, основанные на принципе перекрытия слов (word overlap) исторически восходят к классическому алгоритму *Леска* (Lesk, 1986) [99], который заложил основы целого семейства алгоритмов анализа семантических пересечений.

Алгоритм *Леска* использует *WordNet* как источник семантической информации. Для каждого понятия WordNet предоставляет список различных синсетов, каждое с определением (англ. gloss), объясняющим его значение. Каждый синсет содержит список синонимов или синонимичных словосочетаний и указатели, описывающие отношения между ним и другими синсетами.

Суть метода заключается в следующем: для каждого возможного значения многозначного слова (синсета в WordNet) система анализирует его словарное определение и подсчитывает количество общих слов между этим глоссом и контекстом анализируемого слова в тексте. Значение с максимальным перекрытием (совпадением терминов) выбирается как правильное.

Алгоритм *Леска* использует только два типа информации: 1) набор синсетов, по одной для каждого возможного значения слова, и 2) информацию о контексте, в котором встречается слово. Для двух слов алгоритм выбирает те смыслы (синсеты), определения которых имеют максимальное перекрытие, то есть наибольшее количество общих слов.

В данной работе автор концентрируется на подходах, основанных на знаниях, и, в частности, на подходе, реализованном в методе Леска.

Данный подход имеет ряд известных недостатков. Во-первых, вычислительную сложность, поскольку количество сравнений комбинаторно возрастает с увеличением числа слов в тексте. Во-вторых, ограниченную выразительность, так как перекрытие определяется преимущественно на основе точного совпадения слов в глоссах синсетов, без учета более глубоких семантических связей.

Первый недостаток был частично устранен с помощью «упрощенной» версии алгоритма Леска [100], в которой неоднозначность каждого слова разрешается независимо. Для анализируемого слова выбирается значение, глосс которого демонстрирует максимальное совпадение с текущим контекстом, представленным окружающими словами. Как показано в работе [101], упрощенный алгоритм Леска существенно превосходит исходную версию.

Второй недостаток был смягчен в «адаптированном» алгоритме Леска, предложенном Банерджи и Педерсеном [102], где каждый глосс дополняется определениями семантически связанных значений, что позволяет учитывать отношения между синсетами.

Дополнительное развитие метода было представлено в работе [103], где алгоритм Леска комбинируется с векторными представлениями слов (embeddings). В этом случае перекрытие между глоссом и контекстом вычисляется с использованием мер семантического сходства в распределенном семантическом пространстве. Несмотря на повышение точности, данный подход характеризуется высокой вычислительной сложностью, связанной с парным сопоставлением слов, которая возрастает с увеличением объема текста.

Несмотря на перечисленные улучшения, в существующих вариациях алгоритма Леска сохраняются принципиальные ограничения. В частности,

недостаточно учитывается контекстуальное семантическое представление слов, а также не используется в полном объеме потенциал лексических ресурсов, таких как WordNet, включая примеры употребления и другие доступные семантические данные.

В данной диссертации алгоритм Леска используется в качестве базового метода благодаря простоте его реализации и доказанной эффективности. Глава 2 посвящена анализу указанных ограничений и описанию возможных направлений их устранения, а также представлению модифицированного алгоритма, основанного на принципах метода Леска.

На основе анализа, проведенного в данном и предыдущих разделах, далее формулируется постановка задачи исследования и определяются основные направления дальнейшей работы.

1.5. Постановка задачи персонализации контента

Задача персонализации контента веб-сайта в соответствии с предпочтениями пользователя является актуальной и чрезвычайно важной, поскольку она напрямую влияет на удовлетворенность пользователя, вовлеченность и общее качество взаимодействия с веб-сайтами. Решение данной задачи требует разработки и внедрения моделей и алгоритмов обработки и анализа текстов на естественном языке, которые способны эффективно анализировать поведение пользователей и извлекать ключевые темы интересов.

Таким образом, для достижения цели диссертационного исследования, направленной на повышение эффективности моделей, методов и алгоритмов персонализации контента веб-сайтов, применено комплексное решение. Это решение используется в задаче персонализации на основе применения методов искусственного интеллекта, в том числе, алгоритмов автоматизации построения онтологий. Для этого разработаны модель, метод и алгоритмы, предназначенные

для обработки и анализа текстов на естественном языке, что позволит снизить процента ошибок алгоритмов функционирования систем персонализации.

Описанная и проанализированная научная задача в области искусственного интеллекта требует решения **ряда подзадач**.

Для представления интересов пользователя необходимо разработать онтологическую модель, которая отличается отличающаяся применением новой функции расчета весовых коэффициентов, а также контекстно-зависимых отношений, что позволяет найти более релевантные темы. Это позволит повысить эффективность поиска элементов знаний (концептов и ключевых фраз), соответствующих интересам пользователя.

Для реализации онтологической модели необходимо разработать модифицированный метод автоматического извлечения ключевых фраз из текста, который позволит преодолеть недостатки существующих аналогов и будет совместим с другими алгоритмами для решения задачи персонализации веб-контента.

Для построения тем интересов в профиле пользователя и сопоставления с ними ключевых слов (униграмм) необходимо разработать и применить алгоритм устранения неоднозначностей слов, преодолевающий ограничения существующих методов. Предлагаемый алгоритм учитывает контекстное представление слов, что облегчает его интеграцию с методом извлечения ключевых фраз.

Онтологическая модель и вышеупомянутые алгоритмы должны быть применены вместе в едином конвейере для построения профиля пользователя. Построенный профиль используется в различных задачах. В данной диссертации профиль применяется в задаче персонализации контента веб-сайта и результатов поиска веб-страниц.

Для проведения исследования эффективности предложенных модели, метода и алгоритмов обработки и анализа текстов на естественном языке в системах

искусственного интеллекта и машинного обучения разработан программное приложение, позволяющее проводить сравнительный анализ предложенных алгоритмов с существующими аналогами.

Дадим **формализованную постановку задачи**.

Персонализация веб-контента *формализуется* как задача ранжирования текстовых элементов на основе прогноза интереса пользователя.

Пусть: $U = \{u_1, u_2, \dots, u_{|U|}\}$ – множество пользователей, $I = \{i_1, i_2, \dots, i_{|I|}\}$ – множество текстовых элементов, с которыми взаимодействовал пользователь $J = \{j_1, j_2, \dots, j_{|J|}\}$ – множество новых текстовых элементов, предлагаемых для рекомендации.

Для каждого пользователя $u \in U$ необходимо сформировать упорядоченный список $R_u = \{j_k\}_{k=1}^N \subseteq J$ из N элементов, максимизирующих семантическое сходство с историей взаимодействий $I_u \subseteq I$. Для этого определяется функция $\rho: I \times J \rightarrow [0,1]$, оценивающая семантическую близость текстовых элементов, например, косинусная мера сходства векторных представлений. Тогда элемент $j \in J$ включается в кандидаты $\check{R}_u$, если:

$$\max_{i \in I_u} \rho(i, j) \geq \Delta, \quad (9)$$

где $\Delta \in [0,1]$ – это порог включения, а агрегация через *max* означает, что элемент считается релевантным, если он имеет высокое сходство хотя бы с одним элементом из истории пользователя. Это позволяет включать в рекомендации узкоспециализированные или редкие интересы, даже если они встречаются в истории всего один раз.

Затем для каждого $j \in \check{R}_u$ вычисляется интегральная релевантность:

$$r(j) = \frac{1}{|I_u|} \sum_{i \in I_u} \rho(i, j). \quad (10)$$

Для получения финального списка $\check{R}_u$ ранжируется по убыванию $r(j)$ и выбираются топ-N элементов. Предложенные автором решения должны

минимизировать ошибку ранжирования (процент ошибки функционирования систем персонализации):

$$\mathcal{L} = \sum_{u \in U} \sum_{j \in J} (\hat{y}_{uj} - y_{uj})^2 \rightarrow \min, \quad (11)$$

где $\hat{y}_{uj}$ – предсказанная релевантность, y_{uj} – фактическое взаимодействие (бинарный индикатор).

Реализация данных решений включает разработку и использование набора методов и алгоритмов обработки и анализа текстов на естественном языке, таких как извлечение ключевых слов из контента, устранение неоднозначности значений извлеченных ключевых фраз, извлечение связей между ними и создание профиля пользователя, повышая тем самым эффективность инструментов веб-инженерии для задач персонализации веб-контента.

1.6. Выводы по разделу

Первый раздел посвящен анализу исследуемой предметной области и постановке основных задач исследования.

Проведен аналитический обзор особенностей создания систем искусственного интеллекта и машинного обучения для персонализации контента веб-сайтов при обработке и анализе текстов на естественном языке. Особое внимание уделяется интеграции методов искусственного интеллекта и машинного обучения в процессы веб-инженерии, что позволяет более точно определять пользовательские предпочтения при создании интеллектуальных систем обработки естественного языка. Исследованы этапы процесса персонализации контента веб-сайтов и пересечение этих этапов с разработкой и использованием инструментов искусственного интеллекта, таких как обработка и анализ текстов на естественном языке.

По результатам проведённого анализа были получены следующие промежуточные результаты:

1. Описана задача персонализации контента веб-сайтов и проанализированы методы её решения;

2. Выявлено противоречие в теории, которое связано с неэффективным представлением интересов пользователей в системах веб-персонализации, обусловленным невозможностью существующих методов одновременно обеспечить семантическую точность и масштабируемость при построении пользовательских профилей. Концептуальные подходы характеризуются недостаточной семантической глубиной и подвержены семантической неоднозначности, тогда как онтологические методы обеспечивают более точное учёт контекста, но требуют значительных вычислительных ресурсов и обладают ограниченной адаптивностью к динамическим изменениям пользовательских интересов;

3. Обоснована необходимость использования комплексного решения задачи персонализации контента веб-сайтов на основе применения методов искусственного интеллекта, в том числе, алгоритмов автоматизации построения онтологий. Эти алгоритмы выполняют ключевые функции: извлекают ключевые фразы и понятия, автоматически устраняют семантическую неоднозначность, выявляют онтологические отношения в тексте и сопоставляют концепты с онтологией, а также интегрируются в единый конвейер обработки текстовых данных;

4. Проанализированы методы извлечения ключевых фраз и устранения неоднозначности слов, используемые в контексте задачи персонализации веб-контента, и выявлены недостатки, препятствующие достижению цели диссертации;

5. Сформулирована постановка задачи исследования, а именно задача разработки модели, методов и алгоритмов обработки и анализа текстов на естественном языке, позволяющих повысить эффективность систем

персонализации контента веб-сайтов при условии снижения семантической неоднозначности;

6. Выбраны методология и методы диссертационного исследования, которые автор использовал для преодоления указанных недостатков и разработки предложенных решений. Эти методы находятся на пересечении трех направлений: методов обработки естественного языка, методов автоматизации построения онтологий и методов оценки семантической близости. Описаны методы, которые будут использоваться в разработке, а также те, которые применяются в вычислительных экспериментах для оценки эффективности полученных решений. В качестве основы для разработки метода извлечения ключевых фраз был выбран метод KeyBert, а для разработки алгоритма устранения неоднозначности слов – алгоритм *Леск*. Во всех этих алгоритмах для получения векторного представления фраз и текстов используется модель векторизации S-BERT. Для определения семантической близости между векторами используется косинусная мера сходства.

Последующие главы диссертации описывают получение запланированных результатов, направленных на устранение ограничений существующих методов персонализации контента веб-сайтов. Разработанные метод и набор алгоритмов позволяют снизить процент ошибок алгоритмов функционирования систем персонализации контента веб-сайтов.

ГЛАВА 2. МОДИФИЦИРОВАННЫЙ МЕТОД ИЗВЛЕЧЕНИЯ КЛЮЧЕВЫХ ФРАЗ И АЛГОРИТМ УСТРАНЕНИЯ НЕОДНОЗНАЧНОСТИ СМЫСЛА СЛОВ

На основе сравнительного анализа, проведенного в пункте 1.3, был выбран метод KeyBERT в качестве основы для разработки модифицированного метода извлечения ключевых фраз, учитывая его способность учитывать контекстное представление текста, что обеспечивает более точное выделение семантически значимых фраз.

На следующем рисунке 2.1 представлена схема работы метода KeyBERT. Вначале, документ и его слова кодируются с помощью BERT, что позволяет получить векторные представления (встраивания) как для документа, так и для отдельных фраз. Затем вычисляется косинусная мера сходства между вектором документа и векторами всех фраз. Фразы ранжируются по степени сходства с документом, и наиболее близкие по значению выбираются в качестве ключевых.

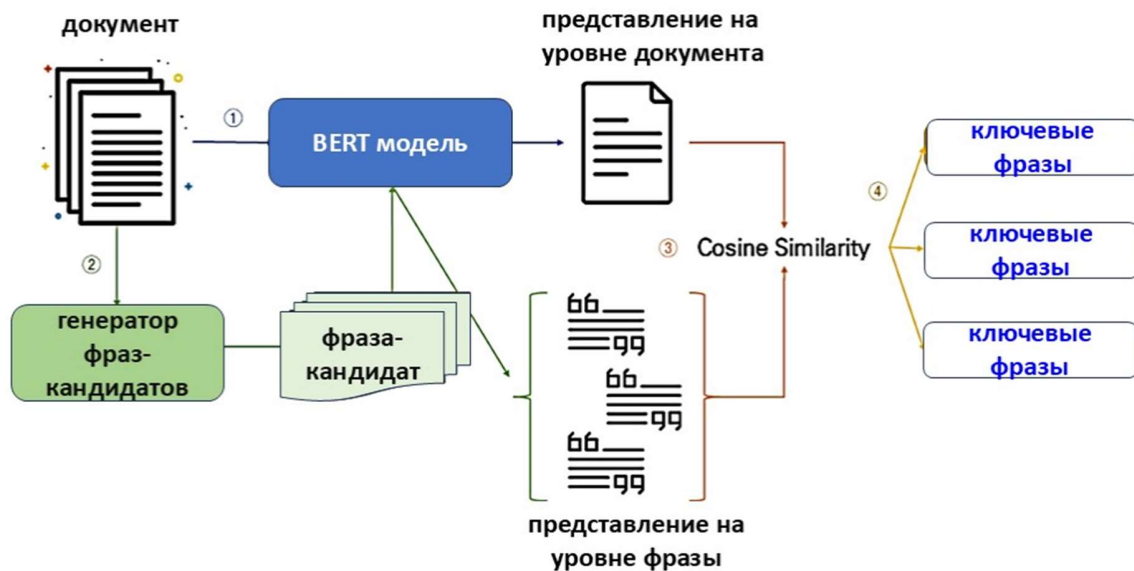


Рисунок 2.1 – Схема работы метода извлечения ключевых фраз KeyBERT

Этот метод позволяет находить фразы, которые лучше всего описывают содержание документа, без необходимости дополнительного обучения модели. Однако, поскольку KeyBERT определяет сходство *части текста* (фразы-кандидата) со всем текстом, логично, что чем длиннее часть, тем выше его сходство с документом. На рисунке (2.2) представлен пример результатов работы KeyBERT по извлечению ключевых фраз из текста.

Machine learning is a transformative technology that powers many modern applications. **Machine learning algorithms** enable systems to learn patterns from data and make predictions or decisions. **Supervised machine learning**, a popular approach, relies on labeled datasets to train models effectively. **Supervised machine learning algorithms**, such as linear regression and decision trees, are widely used for tasks like classification and regression. By leveraging **supervised machine learning**, **businesses can** automate processes and improve decision-making. The versatility of **machine learning algorithms allows** them to be applied across industries, from healthcare to finance. As machine learning continues to evolve, **supervised machine learning algorithms** remain a cornerstone of AI innovation.

Рисунок 2.2 – Пример результатов работы KeyBERT по извлечению ключевых фраз

Как видно из результатов, KeyBERT не учитывает знаки препинания в первом предложении, объединяя «machine learning» и «businesses can» в одну фразу. Кроме того, метод не разделяет глаголы и существительные, что приводит к формированию некорректных или избыточных фраз, таких как «machine learning algorithms allows». Еще одним недостатком KeyBERT является склонность к выделению длинных фраз, даже если они не несут значимой смысловой нагрузки в контексте. Это может снижать точность и релевантность извлеченных ключевых фраз.

Для устранения указанных недостатков разработан модифицированный метод извлечения ключевых фраз, направленный на улучшение эффективности

процесса персонализации контента веб-сайтов. Также описана разработка алгоритма устранения неоднозначности слов, позволяющего снизить погрешность определения правильного значения неоднозначного слова в контексте задачи персонализации контента веб-сайтов.

2.1. Разработка модифицированного метода извлечения ключевых фраз

Разработанный метод извлечения ключевых фраз определяет наиболее релевантные ключевые фразы для документа. Метод отличается применением новых правил для определения весов ключевых фраз на основе частоты их встречаемости и контекстуальной семантической близости.

В качестве основы для разработки метода извлечения ключевых фраз был выбран метод KeyBERT, который использует контекстно-зависимые векторные представления для оценки важности ключевых слов-кандидатов в документе. При этом оценивается семантическая близость между самим документом и его потенциальными ключевыми фразами. Предложенный автором модифицированный метод отличается от KeyBERT тем, что он устраняет смещение в значениях семантического сходства для длинных ключевых слов. Для этого предложена функция, которая нормализует оценки семантической близости в зависимости от длины ключевой фразы-кандидата. Итоговые оценки важности ключевых фраз рассчитываются с использованием функции оценки, которая возвращает среднее гармоническое частоты и близости.

Для реализации предложенного метода применяется модель векторизации фраз S-BERT, которая создает векторные представления фраз на основе контекста. Эта особенность учтена в названии предложенного метода «извлечение ключевых фраз на основе частоты и Bert» (англ. Frequency and Bert based Key phrase Extraction, FBKE) [104–106].

Для более полного представления исследуемых процессов, в следующем пункте дана постановка задачи извлечения ключевых фраз. Данная постановка задачи раскрывает особенности извлечения ключевых фраз в условиях семантической неоднозначности.

2.1.1. Постановка задачи извлечения ключевых фраз

Традиционно извлечение ключевых фраз представляет собой задачу ранжирования, которая описывается следующим образом:

Формально, рассматриваются N объектов $D = \{d_i\}_{i=1}^N$ и M элементов $E = \{e_j\}_{j=1}^M$ где объект – это документ, а элемент – это ключевая фраза. Результат работы алгоритма ранжирования элементов E для объекта $d \in D$ – это отображение $r: E \rightarrow R$, которое сопоставляет каждому элементу $e \in E$ вес $r(e)$, характеризующей степень релевантности элемента объекту (чем больше вес, тем релевантнее элемент).

При этом, набор весов $\{r(e)\}_{e \in E}$ задает перестановку $\pi: [1 \dots M] \rightarrow [1 \dots M]$ на наборе элементов E (считается, что множество элементов упорядоченное) исходя из их сортировки по убыванию веса $r(e)$.

Для решения текущей задачи и определения показателя релевантности, используемого в ранжировании фраз, вводится весовая функция, учитывающая частоту появления фраз в тексте и их семантическую близость к контексту документа (объекту). Для детализации механизма взвешивания фраз и решения задачи извлечения и ранжирования релевантных ключевых фраз далее описываются этапы разработки метода.

2.1.2. Функциональная схема метода извлечения ключевых фраз

Предлагаемый метод состоит из четырех этапов: предварительная обработка текста и формирование n -грамм, выбор фраз-кандидатов, оценка фраз-кандидатов и выбор окончательных ключевых фраз.

N-грамма (или n-gram) в контексте ключевых фраз – это последовательность из N слов или терминов, встречающихся вместе в тексте.

Этап 1. Предварительная обработка текста. На этом этапе документ предварительно преобразуется в машиночитаемый формат для определения потенциальных фраз-кандидатов. Типичная процедура предварительной обработки включает в себя очистку текста, разделение предложений, токенизацию, перевод в нижний регистр и удаление стоп-слов.

После обработки и очистки текста формируются и фильтруются *n*-граммы. Для сканирования текста и построения *n*-грамм используется окно переменной длины.

Основываясь на фундаментальном предположении, что ключевое слово или ключевая фраза не могут охватывать несколько предложений или содержать внутренние знаки препинания (например, запятые, точки с запятой), автор предлагает следующие шаги:

1. Сегментация текста: текст разбивается на предложения с помощью токенизатора NLTK *Punkt*², каждое предложение дополнительно делится на части по знакам препинания (запятые, тире и т. д.), чтобы выделить связные фрагменты для поиска ключевых фраз.

2. Построение *n*-грамм: *n*-граммы формируются только в пределах одного сегмента (части предложения между знаками препинания), чтобы избежать некорректных сочетаний слов из разных частей текста. Это гарантирует семантическую связность, отсекая фрагментированные или пересекающиеся границы последовательности. Полный процесс предобработки текста и генерации *n*-грамм представлен на рисунке 2.3, который визуализирует алгоритмические шаги.

² Документацию можно найти по ссылке https://www.nltk.org/_modules/nltk/tokenize/punkt.html

Этап 2. Извлеченные ключевые фразы фильтруются на основе определенных шаблонов частей речи (таблица 2.1) с использованием любого инструмента для их определения, без необходимости применения полноценного парсера. Это снижает зависимость от языково-специфичных парсеров.

В разработанном алгоритме инструмент NLTK используется для определения частей речи, которые затем применяются для удаления лишних слов, таких как артикли [107].

Таблица 2.1 – POS-шаблоны для извлечения ключевых слов

Паттерн (англ.)	Описание
(ADJ)* + NOUN+	Один или несколько существительных, которым могут предшествовать прилагательные (опционально).
NOUN + NOUN	Сочетание двух существительных, часто образующих сложный термин.
NOUN + ADP + NOUN	Существительное с предлогом и другим существительным – выражает контекстные отношения.
PROPN + PROPN	Комбинация собственных имён (названия, имена, бренды).

Этап 3. Выбор фраз-кандидатов. Отбор ключевых фраз-кандидатов выполняется на основе частоты их встречаемости. Однако, в отличие от предыдущих исследований, автор предлагает рассчитывать частоту для каждого типа n-грамм (униграмм, биграмм и т.д.) независимо. Это связано с тем, что слова в униграммах (отдельные слова) встречаются значительно чаще, чем в биграмах или триграммах (составные фразы). Для устранения этого дисбаланса частота для каждого типа n-грамм рассчитывается по следующей формуле:

$$TF_{n-gram} = \frac{n_t^i}{\sum_k n_k^i}, \quad (12)$$

где $\sum_k n_k^i$ – общее количество n -грамм того же типа (с i элементами) в документе, а n_t^i – количество вхождений конкретного токена t (n -граммы из i элементов) в документе.

Этап 4. Оценка релевантности фразы-кандидата. На этом этапе степень важности термина в документе измеряется с помощью эвристической функции, которая вычисляет сходство между документом и всеми его фразами-кандидатами. Для этого и кандидат, и документ представлены в виде векторов, которые позволяют измерить расстояние между данными векторами с помощью косинусной меры сходства.

Кандидаты ранжируются по частоте, а затем выбираются n лучших кандидатов для окончательной оценки на следующем этапе.

Для этого используется модель векторизации фраз BERT с двунаправленным представлением энкодера на основе трансформеров (англ. Sentences Bidirectional Encoder Representations from Transformers, S-BERT³) [91]. Данная модель использует новую нейросетевую архитектуру на основе «трансформеров». Хотя внутренне S-BERT использует токенизацию на под-слова (кусочки слов), его ключевая особенность – генерация векторов встраивания (эмбеддингов) на уровне предложений (через агрегацию векторов слов, например, mean/max pooling), что отличает его от классического BERT, работающего с координатными выходами токенов. Это послужило мотивацией для использования S-BERT в данной работе, поскольку он способен создавать единое контекстно-зависимое векторное представление для всей фразы (в отличие от отдельных векторных представлений слов), что позволяет более точно определять важность многословных ключевых фраз по отношению к документу.

³ https://www.sbert.net/docs/pretrained_models.html

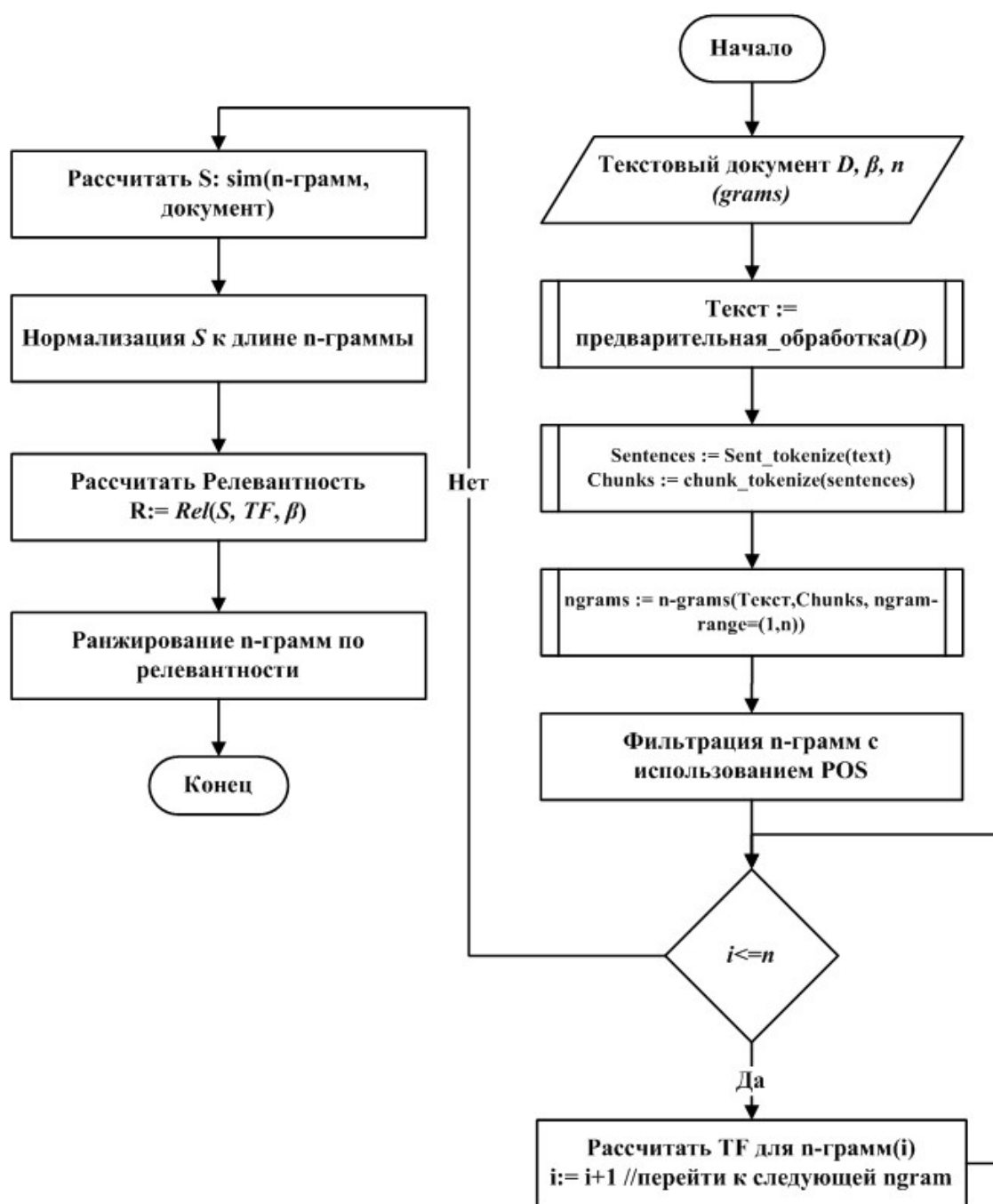


Рисунок 2.3 – Алгоритм FBKE

Однако, поскольку здесь определяется сходство *части текста* (фразы-кандидата) со всем текстом, логично, что чем длиннее эта часть, тем выше сходство с документом. Таким образом, чтобы отрегулировать влияние длины фразы на сходство при его сопоставлении с документом, оценка сходства нормализуется к длине слова-кандидата по следующей формуле:

$$S_{norm}^j = S^j \cdot e^{S^j/|c^j|}, \quad (13)$$

где S^j – сходства j -ого кандидата с документом. $|c^j|$ это длина j -ого фразы.

Как показано на рисунке 2.4, влияние этой функции на степень сходства будет значительным только для больших значений (выше 0,5). Можно сказать, что это функция штрафует высокие значения сходства для «низкочастотных длинных фраз-кандидатов», таким образом, уравнивая длину ключевой фразы.

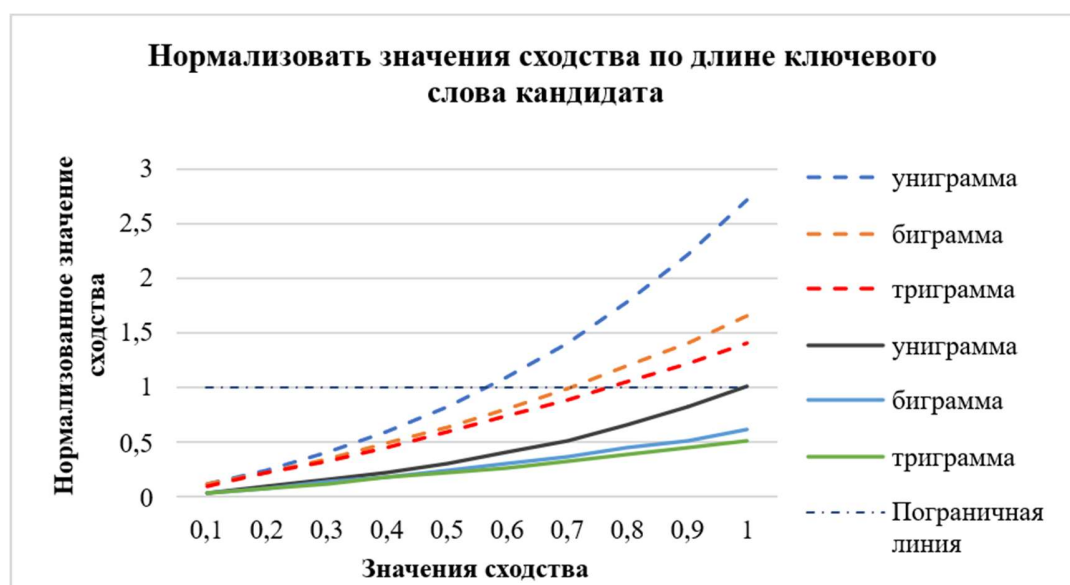


Рисунок 2.4 – Нормализация значений меры сходства ключевых фраз по их длине

Например, предположим, что имеются три ключевых фразы-кандидата, униграмма, биграмма и триграмма, степень сходства каждой из которых с документом составляет 0,8, тогда после нормализации с применением предложенной ранее функцией получим следующие значения сходства: 1,7; 1,19; 1,04 соответственно для каждого типа n -граммы. Чтобы сделать разницу более выразительной, значения сходства нормализуются в интервале $[0-1]$, используя общую формулу *min-max*, которая задается как:

$$x' = \frac{x - \text{Min}(x)}{\text{Max}(x) - \text{Min}(x)}, \quad (14)$$

где x – исходное значение, x' – нормализованное значение. В итоге получается результат 0,65 для униграмм, 0,43 для биграмм и 0,38 для триграмм.

Окончательная оценка релевантности полученных ключевых фраз рассчитывается как среднее гармоническое значение сходства ($S_{norm}^{j,d}$) и частоты встречаемости (tf_{cj}) на основе использования F-меры расчета точности классификатора:

$$rel(d, c^j) = (1 + \beta^2) \cdot \frac{tf_{cj} \cdot S_{norm}^{j,d}}{\beta^2 \cdot tf_{cj} + S_{norm}^{j,d}}, \quad (15)$$

где β принимает значения в диапазоне $0 < \beta < 1$, для выбора приоритета частоты, в противном случае при $\beta > 1$, приоритет отдается сходству. При $\beta = 1$ частота и сходство имеют одинаковый приоритет. Данная формула позволяет управлять выходным значением функции с помощью параметра бета. Это очень важно для оценки влияния каждого из этих параметров на конечный результат. После определения релевантности для каждой ключевой фразы-кандидата, они ранжируются в порядке убывания, после чего выбираются N -первых, значение N является свободным параметром метода.

На рисунке 2.5 представлены результаты, полученные с использованием метода *FBKE* для извлечения ключевых фраз из текста, который также был обработан с помощью *KeyBERT* (рис. 2.2).

Machine learning is a transformative technology that powers many modern applications. **Machine learning algorithms** enable systems to learn patterns from data and make predictions or decisions. **Supervised machine learning**, a popular approach, relies on labeled datasets to train models effectively. **Supervised machine learning algorithms**, such as linear regression and decision trees, are widely used for tasks like classification and regression. By leveraging supervised machine learning, businesses can automate processes and improve decision-making. The versatility of **machine learning algorithms** allows them to be applied across industries, from healthcare to finance. As machine learning continues to evolve, **supervised machine learning algorithms** remain a cornerstone of AI innovation.

Рисунок 2.5 – Пример результата работы FBKE по извлечению ключевых фраз

Можно заметить, что ключевые фразы, выделенные модифицированным методом FBKE, не содержат лишних слов, таких как «киберт». Это можно объяснить тем, что процесс чанкинга (Chunking) позволяет корректно определить границы фраз, а фильтрация по шаблонам частей речи эффективно отсеивает шум.

Пример демонстрирует важность учета текстовых блоков и пунктуации, а также показывает необходимость рассмотрения частоты встречаемости, длины фразы и семантической схожести при выявлении значимых фраз. Данные результаты демонстрируют превосходство предложенного метода по сравнению с методом KeyBERT. В главе 4 автором представлены результаты вычислительных экспериментов, подтверждающие данное утверждение.

2.1.3. Применение FBKE в задачах персонализации контента веб-сайтов

Предложенный метод FBKE демонстрирует несколько преимуществ, которые подтверждают его эффективность для решения задач персонализации контента веб-сайтов, особенно при создании пользовательских профилей [108]. Ключевые преимущества включают:

1. Метод использует модель векторного представления S-BERT, которая позволяет определять значение фраз с учётом контекста. Это особенно важно для персонализации, где важно учитывать намерения и контекст взаимодействий пользователей;

2. Метод генерирует векторы, представляющие извлеченные ключевые фразы для каждого документа. Эти векторы используются на последующих этапах для сопоставления ключевых фраз профиля, что значительно снижает вычислительные затраты, связанные с этим процессом;

3. Метод работает на уровне документа, что делает его независимым от общего корпуса текста. Это снижает вычислительные затраты при обновлении профилей пользователей.

Хотя контекстные модели векторизации, такие как S-BERT, эффективно улавливают смысловые особенности ключевых фраз при анализе документов, они имеют ограничения в устранении неоднозначности униграмм (отдельных слов). В частности, S-BERT формирует единое обобщенное векторное представление для отдельных слов (например, "банк"), если соседние слова не принимаются во внимание. Без окружающих слов предложения модель не может различить разные значения слова. Следовательно, S-BERT лучше всего работает с полными предложениями или фразами, где контекст помогает получить точные и осмысленные векторные представления.

Поэтому для устранения данного недостатка в пункте 2.2 диссертации описана разработка алгоритма устранения неоднозначности смысла слов, который отличается применением функции определения контекстуальной семантической близости и расширенных аннотаций синсетов, что позволяет снизить погрешность определения правильного значения неоднозначного слова в данном контексте.

2.2. Разработка модифицированного алгоритма устранения неоднозначности смысла слов

В данном пункте описывается разработка модифицированного алгоритма устранения неоднозначности смысла слов, который построен на основе алгоритма Леска (с встраиванием). Для понимания необходимости предлагаемых модификаций в работе рассматриваются недостатки классических и существующих подходов.

Алгоритм Леска в различных версиях, включая модификации с использованием векторных представлений, опирается в основном на определения (glosses) и контекст слова (рисунок 2.6). Однако определение синсета не обязательно должно включать слова, которые являются общими для контекста

слова в анализируемом тексте. Поэтому оно не всегда является достаточным представлением для устранения неоднозначности.

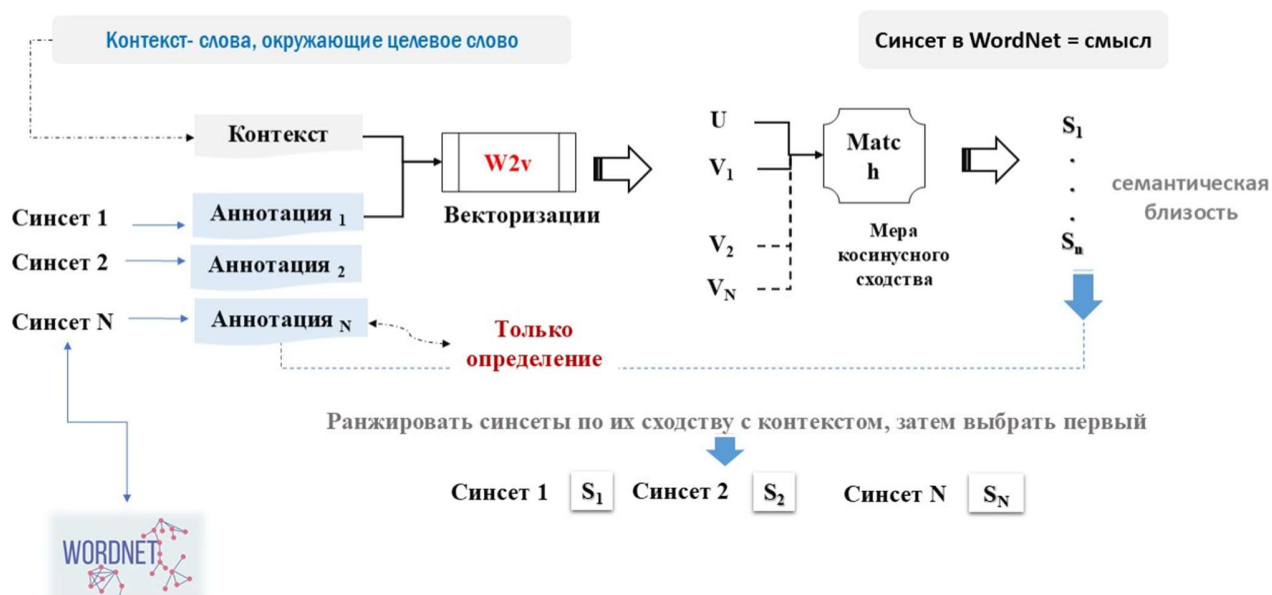


Рисунок 2.6 – Недостатки алгоритма LSK для устранения неоднозначности слов

Для иллюстрации этого ограничения рассмотрим устранение неоднозначности слова «ключ» в предложении: «мы пили воду из холодного ключа». WordNet (или его русскоязычный аналог) обеспечивает несколько наборов синонимов для слова "ключ". Правильное значение в данном контексте – это «родник» или «источник воды». Однако определение этого смысла может не содержать слов «пили» или «воды», что создает трудности для алгоритмов, основанных исключительно на пересечении слов. В то же время, примеры использования, такие как «чистый ключ бьет из-под земли», предоставляют дополнительные контекстуальные подсказки.

Это демонстрирует, что одних только определений часто недостаточно, и методы, основанные исключительно на принципе перекрытия слов, не являются эффективными. Более того, даже в модифицированном алгоритме Леска, использующем векторные представления слов (например, Word2Vec), существует существенный недостаток. Word2Vec предоставляют единый статический вектор

для слова независимо от его контекста. Например, вектор для слова «ключ» усредняет смыслы «дверной ключ», «музыкальный ключ» и «родник», что не позволяет учитывать тонкие контекстуальные связи и различать значения в зависимости от ситуации.

В качестве лексической основы для работы с русским языком в исследовании используется Russian WordNet (RuWordNet)⁴ – большая семантическая сеть для русского языка, разработанная по образцу Princeton WordNet. Она организует слова в синсеты (наборы синонимов) и связывает их семантическими отношениями. Для корректной обработки данных алгоритмом необходимо понимать структуру идентификаторов в этом ресурсе.

Дальше представлен формат синсетов WordNet, используемый в наборах данных. Чтобы понять формат, можно использовать следующий пример, содержащий слово мышь "mouse", упомянутое дважды, каждый раз с разным значением.

Для предложения: «Ученый использовал мышь для управления компьютером, пока кот гонялся за маленькой мышкой в саду» (*The scientist used a mouse to control the computer, while the cat chased a small mouse in the garden*), WordNet предоставляет несколько значений для слова «мышкой» (*mouse*), которые описываются следующим образом:

значение 1: маленький грызун", синсет WordNet: «mouse%1:05:00::»;

значение 2: ручное устройство для управления компьютером, синсет WordNet: «mouse%1:06:00::».

Обозначения, используемые в WordNet для идентификации смыслов слов, такие как «mouse%1:05:00::» и «mouse%1:06:00::», следуют определенному формату, чтобы уникально идентифицировать каждый смысл слова. Расшифровка каждой части обозначения следующая.

⁴ <https://ruwordnet.ru/ru>

«слово%часть_речи:номер_лексического_файла:лексический_ID::»:

- `mouse`: это лемма (базовая форма) слова. В данном случае слово – "mouse" (мышь);
- `%`: этот символ разделяет лемму и идентификатор смысла;
- `1`: указывает на часть речи (POS) слова. Число `1` соответствует существительным в WordNet. Другие числа обозначают разные части речи:
 - `1`: Существительное;
 - `2`: Глагол;
 - `3`: Прилагательное;
 - `4`: Наречие;
 - `5`: Прилагательное-сателлит;
- `05` и `06`: эти числа представляют номер лексического файла. WordNet организует слова в 45 лексических файлов (категорий) на основе их семантических групп. Например, `05` относится к категории животных (например, "мышь" как маленький грызун), а `06` относится к категории артефактов (например, "мышь" как компьютерное устройство);
- `00`: это лексический ID, который различает разные смыслы внутри одного лексического файла. Например, если в одной категории есть несколько смыслов слова "мышь", это число помогает их различить;
- `::`: это разделитель, обозначающий конец идентификатора смысла. Часто за ним следует дополнительная информация (например, определения или примеры), но в данном случае он пустой.

Такой структурированный формат позволяет системам, основанным на WordNet, точно устранять неоднозначность слов в зависимости от контекста.

Учитывая эти ограничения, разработан модифицированный алгоритм устранения неоднозначности слов, который устраняет недостатки существующих методов. Предлагаемый модифицированный алгоритм применяет расширенную

аннотацию синсетов в WordNet, включив в нее не только определения, но и примеры использования слов. Это обусловлено тем, что примеры лучше отражают контекстуальное употребление слова, что является ключевым для понимания его семантики в конкретной ситуации [109, 110].

Определения синсетов, хотя и важны, могут не содержать общих слов с контекстом анализируемого текста, что ограничивает их полезность при устранении неоднозначностей. Примеры демонстрируют реальное использование слова в различных контекстах, что позволяет более точно определить его значение в конкретном случае. Однако существенным ограничением является отсутствие примеров для некоторых синсетов. Для решения этой проблемы предлагается использовать примеры из семантически связанных синсетов (таких как синонимы, гипонимы, гиперонимы и другие отношения, предусмотренные WordNet).

Предложенный модифицированный алгоритм отличается от оригинального алгоритма Леска также тем, что применяет контекстное векторное представление S-BERT вместо статических Word2Vec. Это приводит к более тонкому пониманию предпочтений пользователя, поскольку значения ключевых фраз уточняются в соответствии с их использованием в контексте. В отличие от word embeddings, S-BERT учитывает контекст, что позволяет различать значения многозначных слов (как в примере со словом «ключ»). Кроме того, использование S-BERT как для извлечения ключевых фраз, так и для устранения неоднозначности слов позволяет системе поддерживать согласованное семантическое представление на протяжении всего процесса персонализации, что приводит к более точной интерпретации интересов пользователя. Данное решение также позволяет избежать дополнительной процедуры векторизации контекстов для униграмм, являющихся частью ключевых фраз (биграмм, триграмм и выше), извлеченных на предыдущем этапе персонализации, что уменьшает общие вычислительные затраты.

На рисунке 2.7 зелёным цветом указаны области, в которых предлагается внести изменения в базовый алгоритм. Внесённые модификации включают использование векторного представления на основе S-BERT, а также расширение описания каждого синсета за счёт примеров с определениями.

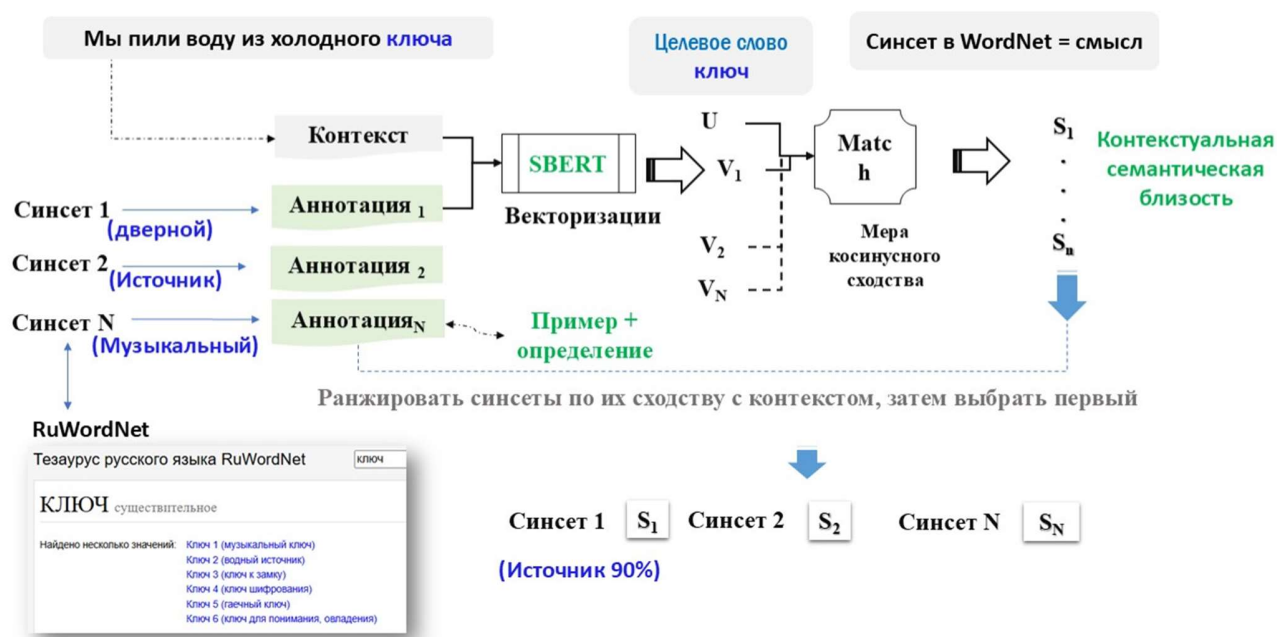


Рисунок 2.7 – Механизм устранения неоднозначности смысла слова LESK-SBERT

С целью всестороннего анализа рассматриваемых процессов далее приводится формулировка задачи устранения неоднозначности слов.

2.2.1. Постановка задачи устранения неоднозначности слов

Пусть w - некоторое слово, а C - контекст (текстовый фрагмент), в котором это слово встречается. Также, пусть KB – это лексическая база знаний (WordNet), которая предоставляет для каждого слова список синсетов, аннотаций, объясняющих множественные значения слова, причём каждый синсет имеет определение и примеры того, как он используется в языке.

Задача состоит в том, чтобы определить правильное значение слова w в данном контексте C , используя информацию лексической базы данных KB . Задача

решается путём измерения сходства между контекстом и синсетами целевого слова, а затем выбирается синсет, который достигает наивысшей степени сходства с контекстом. Формально задача заключается в максимизации сходства синсетов и представляется следующим выражением:

$$f(S_i) = \max_{s \in S(w)} S(w, C, s), \quad (16)$$

$$s^* = \arg \max_{s_i} f(s_i), \quad (17)$$

где $S(w)$ – это множество всех синсетов слова w , s – конкретный синсет из этого множества, а $S(w, C, s)$ – функция определения сходства между C и s . Синсет s^* , достигающий максимального сходства, представляет правильное значение слова.

Для решения данной задачи предлагается модифицированный алгоритм Леска для устранения неоднозначности слов, включающий следующие усовершенствования: (1) адаптивный метод построения векторных представлений для учета контекстуальной зависимости, (2) интеграцию расширенных аннотации из *WordNet*. В целях детализации механизма устранения неоднозначности смысла слов на основе данных *WordNet* ниже последовательно излагаются ключевые этапы работы алгоритма.

2.2.2. Разработка алгоритма устранения неоднозначности смысла слов на основе контекстуального векторного представления

Модифицированный алгоритм устранения неоднозначности слов описывается следующими шагами.

Шаг 1. Определяется контекст, в котором появилось ключевое слово, то есть его окружающие слова. Если контекст слова включает одну или несколько из ранее извлеченных ключевых фраз, то эти фразы рассматриваются в качестве контекста текущего слова. Цель заключается в использовании векторного представления, сгенерированного по этим фразам на этапе извлечения ключевых фраз, для снижения вычислительных затрат.

Шаг 2. Для контекста C и целевого слова w (неоднозначное слово) извлекается набор синсетов $S = (S_0, S_1, \dots, S_N)$ с помощью WordNet. Затем для каждого синсета извлекается определение $O = (o_1, o_2, \dots, o_N)$ и набор примеров $\Pi = (\pi_1, \pi_2, \dots, \pi_N)$. Если в WordNet нет примеров использования текущего синсета, функция *get_related_examples* «поиск релевантных примеров» ищет ближайшие примеры в связанных синсетах (синонимах, гипонимах или гиперонимах). Таким образом, при отсутствии прямых примеров, система автоматически обращается к родственным понятиям, чтобы предоставить релевантные варианты.

На рисунке 2.8 показан процесс поиска примеров для расширения аннотации синсета в WordNet, у которого отсутствуют примеры употребления.

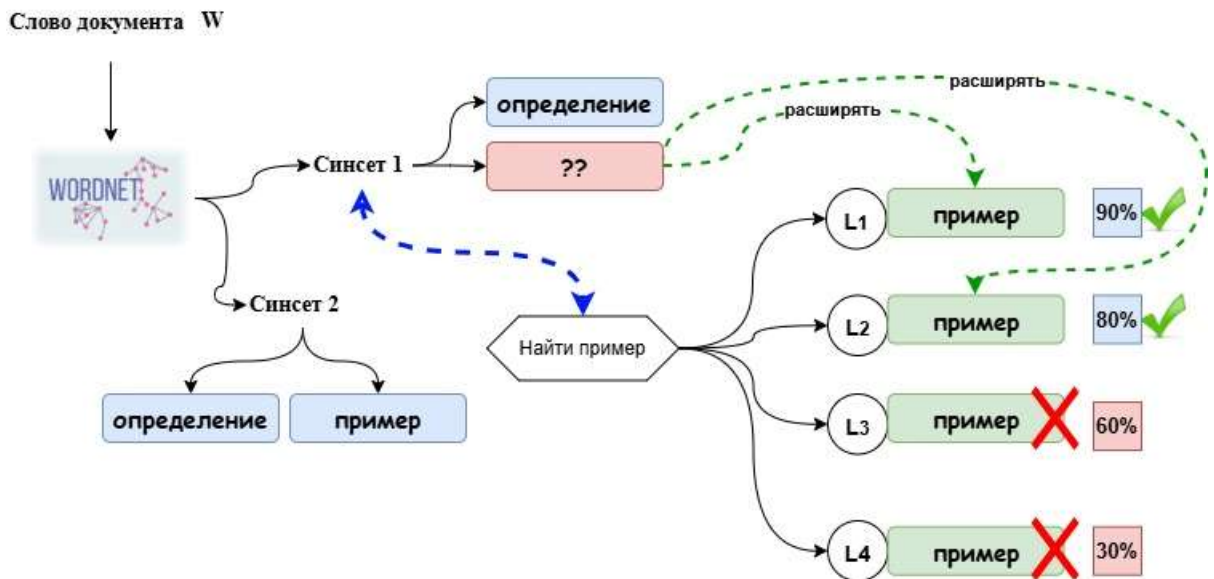


Рисунок 2.8 – Схема расширения аннотации синсета

Процесс аннотации начинается с целевого синсета S , для которого производится сбор примеров-кандидатов из связанных синсетов в соответствии с иерархией приоритетов:

- Уровень L1: Прямые гиперонимы (наиболее общие и семантически согласованные примеры).
- Уровень L2: Прямые гипонимы (конкретные экземпляры).

- Уровень L3: Сестринские синсеты (контекстуальная схожесть).
- Уровень L4: Холонимы (отношения «целое – часть»).

Алгоритм 2. Алгоритм устранения неоднозначности слов (*Леск-S-BERT*)

Вход - Целевое слово w
 - Текст документа D
 - Предварительно извлеченные ключевые фразы KP
 - Лексическая база WordNet

Выход S_{best} Конкретное значение слова w в данном контексте

```

//определить контекст  $K$  слова.
1:  if  $w$  является частью  $n$ -граммы в  $KP$ :
2:      |      вектора контекста  $K^B = \text{существующий вектор из } KP$ 
   else:
3:      контекст  $K = \text{извлечь\_окружающие\_слова}(w, D, win = 5)$ 
4:      вектора контекста  $K^B = S\text{-Bert}(K)$ 

```

```

// обработка синсетов
5:   $S = \text{WordNet.get\_synsets}(w)$  //Список возможных синсетов слова
    $S = (S_0, S_1, \dots, S_N)$ 
6:  for each synset in  $S$  do:
7:       $O = \text{synset.definitions}$  //извлекается определение каждого
   синсета  $O = (o_1, o_2, \dots, o_N)$ 
8:       $\Pi = \text{synset.examples}$  //  $\Pi = (\pi_1, \pi_2, \dots, \pi_N)$  //извлекается набор
   примеров каждого синсета
9:      if not  $\Pi$ :  $\Pi = \text{get\_related\_examples}()$ 
    $T_{\text{аннотации}} = \text{concatenate}(\Pi, O)$  //объединять тексты
   End
10:  $N = \text{len}(S)$  //количество синсетов
11:  $\text{Similarity\_scores} = []$ 
12: for  $i = 0$  to  $N$  do:
13:      $t_i^B = S\text{-Bert}(t_i)$  // Представление синсета  $S[i]$  в виде вектора
    $\text{Sim}_{S\text{-Bert}} = \cos(\vec{t_i^B}, \vec{K^B})$ 
14:      $\text{Similarity\_scores}[i] \leftarrow \text{sim}$ 
15: end
16:  $S_{best} = \text{max}(\text{Sort}(S, \text{Similarity\_scores}))$  // Ранжирование синсетов

```

На следующем этапе для каждого примера-кандидата вычисляется косинусное сходство между его векторным представлением и вектором определения исходного синсета. Полученные кандидаты ранжируются по убыванию вычисленного показателя релевантности.

Финальный шаг включает в себя отбор всех кандидатов, значение косинусного сходства которых превышает установленный порог θ , и их последующее объединение для формирования итоговой аннотации целевого синсета.

Полученные примеры и определение объединяются в один текст аннотации синсета:

$$T_{\text{аннотации}} = \text{concatenate}(\Pi, O) = (t_1, t_2, \dots, t_N). \quad (18)$$

Шаг 3. Для контекста и синсета целевого слова создаются векторы с помощью S-BERT [111, 91]. Если контекст присутствует среди ранее извлеченных ключевых фраз, используется соответствующий вектор.

$$t_i^{SBERT} = \text{Встраивания}(t_i), \quad (19)$$

где, t_i^* – это вектор встраивания аннотации i -го синсета.

Шаг 4. Определить сходство между контекстом K и аннотацией t_i^{S-Ber} с помощью косинусной меры сходства, задаваемой формулой (1):

$$\text{Sim}_{S-Ber} = \cos(\vec{t_i^B}, \vec{K^B}). \quad (20)$$

Синсеты ранжируются по степени сходства, затем выбирается синсет с наивысшим показателем схожести.

Разработанный модифицированный алгоритм является универсальным и может применяться независимо от задачи персонализации. Его эффективность проверяется при описании вычислительных экспериментов в разделе 4, где продемонстрировано решение задачи устранения неоднозначности смысла слов, показанное на стандартных наборах данных. Предложенный автором

модифицированный алгоритм позволяет получить конкурентоспособные результаты по точности и скорости в сравнении с известными алгоритмами.

2.3. Выводы по разделу

В данной главе описана разработка модифицированного метода извлечения ключевых фраз, который использует функцию взвешивания, учитывающую частоту появления фраз в тексте и их семантическую близость к контексту документа. Метод отличается тем, что частота каждого типа n -грамм рассчитывается независимо для обеспечения честной конкуренции между ними на этапе ранжирования. Кроме того, метод использует модель контекстно-зависимого векторного представления S-BERT для оценки сходства между фразами-кандидатами и текстом документа, после чего эти значения нормализуются с учетом длины фраз, чтобы отрегулировать влияние длины фразы на сходство при его сопоставлении с документом. Применение S-BERT позволяет раскрыть значение фраз в зависимости от их контекста, что особенно важно для персонализации.

Для устранения неоднозначности значений извлеченных слов (униграмм), автор разработал модифицированный алгоритм *Lesk-S-BERT*, который использует векторы S-BERT вместо встраивания слов и дополняет аннотацию синсетов из WordNet примерами. Использование расширенной аннотации синсетов повышает точность определения правильного значения слова, поскольку эти примеры отражают контекстуальное использование слова, что является ключевым для понимания его значения в конкретном контексте.

Использование S-BERT как при извлечении ключевых фраз, так и при устранении неоднозначности слов снижает вычислительные затраты алгоритма. Также, это гарантирует, что процесс устранения неоднозначности слов основан на том же контекстуальном понимании, которое использовалось для извлечения

ключевых фраз, что приводит к более точной интерпретации интересов пользователя.

Разработанные модифицированные метод и алгоритм интегрированы в создание и расширение пользовательского профиля с целью улучшения процесса персонализации веб-контента. Выводы подтверждаются результатами вычислительных экспериментов, описанными в четвертой главе. В следующей главе описаны процесс построения пользовательского профиля пользователя на основе онтологической модели и алгоритм персонализации результатов поиска веб-страниц, которые разработаны с применением полученных ранее в работе результатов.

ГЛАВА 3. ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ И АЛГОРИТМ ПЕРСОНАЛИЗИРОВАННОГО ПОИСКА ВЕБ-САЙТОВ

В этом разделе описывается применение разработанного метода извлечения ключевых фраз и алгоритма устранения их неоднозначности в ходе процесса построения профиля для решения задачи персонализации интересов пользователя. Описано построение онтологической модели профиля пользователя (интересов пользователя), а также алгоритм персонализированного поиска веб-сайтов на основе предложенной онтологии.

3.1. Онтологическая модель представления интересов пользователей

Построена модель организации интересов пользователя в онтологической структуре с учетом информации, изложенной в пунктах 1.2-1.4, посвященных сравнительному анализу технологий обработки и анализа текстов на естественном языке при решении задачи персонализации контента веб-сайтов. Извлечение ключевых фраз и концептов является основой для определения интересов пользователя в системах искусственного интеллекта и машинного обучения. Кроме того, использование онтологической модели позволяет повысить эффективность методов извлечения ключевых фраз и алгоритмов устранения неоднозначности концептов в контексте обновления профиля пользователя, поскольку она способствует улучшению поиска концептов, связанных с интересами пользователя, и оценке их семантической близости [112–116].

Предложенная онтологическая модель существенно отличается от традиционных, динамическим созданием онтологии "на лету" из текстов. Кроме того, она основана на двухуровневой иерархии, которая сочетает тематические концепты ($\mathcal{C}$) и ключевые фразы ($\mathcal{I}$), связанные семантическими отношениями ($\mathcal{R}$).

Рассмотрим формальное описание предложенной онтологической модели O , которое определяется как кортеж:

$$O = \langle I, C, R, W, \chi, D, A \rangle, \quad (21)$$

где I – множество экземпляров, представляющих конкретные сущности или ключевые фразы; C – множество концептов (классов), представляющих общие категории или темы интересов; R – множество семантических отношений (является «is-a», часть-целое «part-of» и др.), импортированных из эталонных онтологий, определяющих связи между концептами и/или экземплярами; R включает как иерархические, так и неиерархические отношения. Предлагаемая онтология характеризуется дополнительными элементами, включающими:

- D – множество контекстно-зависимых извлекаемых отношения (automatic relations), не зафиксированные в R , полученных из текста, где $D \cap R = \emptyset$;
- Весовые функции:

$W: C \cup I \rightarrow [0,1]$, присваивает весовые коэффициенты концептам (C) и экземплярам (I), количественно отражающие уровень интереса пользователя к соответствующим темам. Данные веса рассчитываются на основе анализа поведения пользователя и его взаимодействия с текстовыми данными;

$\chi: D \rightarrow [0,1]$ присваивает весовые коэффициенты, определяющие степень достоверности извлекаемых отношений.

- A – множество аксиом (логических ограничений), обеспечивающих правила для определения и фильтрации связей.

Далее, приводится подробное объяснение онтологии и того, как она реализована.

1. Уровень концептов (C) – Темы и подтемы интересов. Это обобщённые концепты, представляющие темы в какой-то области (например, "машинное

обучение", "обработка естественного языка"). Эти темы получаются путём кластеризации семантически близких ключевых фраз. Каждый концепт имеет:

- векторное представление (усреднённое от входящих в него ключевых фраз);
- вес (W) – степень важности для пользователя;
- метка или название темы.

Название темы основано на названиях фраз, из которых она состоит. Для этого используется простой алгоритм для поиска самой длинной последовательности элементов, которая встречается в одинаковом порядке в двух или более последовательностях. В контексте кластеризации семантически схожих ключевых фраз последовательностями обычно являются слова, из которых состоят фразы. Эта общая подпоследовательность затем используется в качестве репрезентативного названия для кластера. Если кластер содержит только одно слово, оно считается названием кластера. Если кластер не имеет общих последовательностей, в качестве его метки выбирается ключевая фраза, ближайшая к центроиду кластера.

2. *Уровень экземпляров (I) – Ключевые фразы.* Это конкретные термины, извлечённые из текстов (например, *"нейронные сети"*, *"Scikit-Learn"*). Фразы извлекаются автоматически (например, с помощью FBKE или RAKE). Каждая фраза имеет:

- собственный вектор (например, из S-BERT);
- вес (W) – значимость в документе;
- связи с концептами (например, *"PyTorch"* → является → *"библиотеки машинного обучения"*).

3. *Отношения D.* Это автоматически извлекаемые отношения между ключевыми фразами, полученные из текста с использованием алгоритма извлечения семантических отношений. Особенность данного алгоритма

заключается в автоматическом формировании этих связей из текста с помощью методов оценки семантической близости, применяемых к векторным представлениям фраз и концептов.

Для оценки семантической близости на уровне фраз (между фразами) с использованием моделей BERT или S-BERT обычно применяется косинусная мера сходства с порогом в диапазоне $[0,6-0,9]$: значения выше 0,9 часто указывают на почти идеальные синонимы, результаты в промежутке 0,6–0,9 указывают на высокую семантическую близость, но не строгую синонимичность, а показатели ниже 0,6, как правило, не считаются показателем сильной семантической связи.

Алгоритм извлечения семантических отношений

Вход: *KWS* //Список ключевых фраз

Выход: *Relation* // Список извлекаемых отношений

```

1:  for i in range(0, length(KWS)) do:
2:      for j in range(i + 1, length(KWS) – 1) do:
3:          kw1, kw2 = KWS [i], KWS [j]
4:          s = cos_sim(kw1, kw2)
5:          if s > 0.9: Relation(kw1, kw2) = synonym // синонимичность
6:          if s < 0.6: Relation(kw1, kw2) =  $\emptyset$  // отсутствие связи
7:          if 0.6 < s < 0.9:
8:              Relation(kw1, kw2) = search_wikiData(kw1, kw2)
9:              if Relation(kw1, kw2) ==  $\emptyset$ :
10:                 Relation(kw1, kw2) = has-relation
11:      end
12:  end
13:  Return Relation

```

Для ключевых фраз с оценкой близости в диапазоне от 0.6 до 0.9 используется Wikidata для проверки определенных отношений (например, синонимия, часть-целое, подкласс) между парами фраз. Пары, не имеющие определенных отношений в Wikidata, маркируются обобщенной меткой "has-relation", обозначающей наличие тематической связи подчиненного характера.

Таким образом, данный алгоритм извлекает следующие типы отношений:

- синонимия (сходства >90% или 60-90% при подтверждении через Wikidata);

- отношения: часть-целое (part of); более широкое понятие (Broader); более узкое понятие (Narrower); subClassOf верифицируются через Wikidata при сходстве 60-90%;
- общая (нетипизированная) связь (с доверительным уровнем 60-90%);
- отсутствие связи $\emptyset$.

Например, косинусное сходство между векторами S-BERT для фразы «машинного обучения⁵» и «искусственного интеллекта» составляет 0.72. Это указывает на то, что требуется дополнительный шаг для определения конкретного типа отношения. Для уточнения взаимосвязи используется WikiData (рис. 3.1), которая подтверждает, что «машинное обучение» является подклассом «искусственного интеллекта».

Использование *Wikidata* в качестве источника для выявления связей между концептами профиля пользователя предоставляет значительное преимущество благодаря своей динамической и расширяемой природе. В отличие от статических онтологий, *Wikidata*, основанный на структуре описания ресурсов RDF (Resource Description Framework), позволяет непрерывно дополнять и уточнять связи между сущностями без жестких структурных ограничений. Это обеспечивает возможность выявления неочевидных и междисциплинарных взаимосвязей, что особенно ценно в задачах семантического анализа и рекомендательных систем.

Кроме того, интеграция *Wikidata* позволяет преодолеть ограничения традиционных баз знаний, поскольку она поддерживает связанные открытые данные (Linked Open Data, LOD), что обеспечивает совместимость с внешними источниками, такими как DBpedia или GeoNames. Это особенно важно для задач, требующих контекстуального анализа, например, при кластеризации научных публикаций или персонализации контента [28].

⁵ [machine learning - Wikidata](#)

subclass of		computer science	edit
		reason for deprecated rank	concerns a superclass of this item
		▸ 1 reference	
		artificial intelligence	edit
		▸ 1 reference	
	+ add value		

part of		artificial intelligence	edit
		▸ 1 reference	
		mathematics	edit
		▸ 2 references	
+ add value			

Рисунок 3.1 – Получены отношения для «машинного обучения» из WikiData

На рисунке 3.2 представлен пример онтологии, иллюстрирующий отношения между ее элементами. Серые узлы представляют метки для основных категорий онтологии: ключевые слова и темы. Синие узлы соответствуют ключевым словам, каждый из которых имеет атрибуты (зелёные узлы): вес (weight), который выражает степень интерес пользователя к этому элементу; вектор (vector) и метка (label).

Красные узлы обозначают тематические категории, которые обладают теми же атрибутами, что и ключевые слова. Из рисунка 3.2 видно, что понятие «машинное обучение» связано с «искусственным интеллектом» отношением (broader / narrower), т.е. подклассом. Обе ключевые фразы связаны с темой «AI Topic» отношением «isRelatedTo», при этом оба понятия относятся к категории ключевых фраз, а сама категория «ключевая фраза» является подклассом (типа Class).

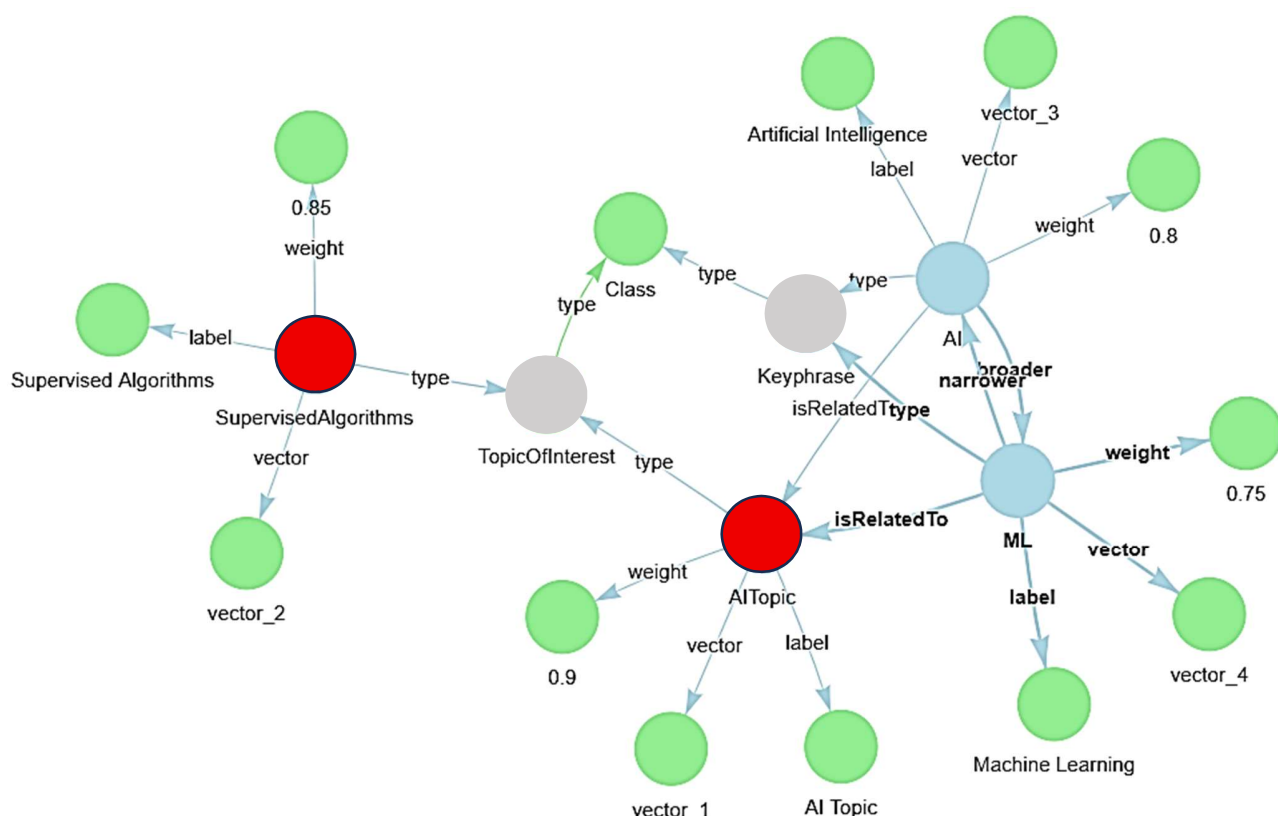


Рисунок 3.2 – Пример онтологии, иерархия концептов (C) и их экземпляров (I), связанных отношениями (R)

4. *Динамическое взвешивание концептов (W)*. Пусть K – множество ключевых фраз (экземпляров). Каждой ключевой фразе $k_i \in K$ присваивается вес $w(k_i)$, определяющий её значимость в документе и среди ключевых фраз, присутствующих в профиле.

Ключевые фразы группируются в темы интересов $T = \{t_1, t_2, \dots, t_m\}$, где каждая тема t_i представляет собой кластер семантически близких ключевых фраз. Каждой теме t_i присваивается общий вес $w(t_i)$ на основе агрегированных весов её ключевых фраз:

$$w(t_i) = \frac{1}{J} \sum_{k_j \in t_i} w(k_j). \quad (22)$$

Формулы, используемые для расчёта веса, подробно описаны в алгоритме построения пользовательского профиля в разделе 3.2.

4. *Примеры использования аксиом (A).* Для повышения логической строгости, динамичности и персонализации онтологии используются следующие типы аксиом:

- разделение экземпляров и концептов (Если x – ключевая фраза, он не может быть концептом):

$$\forall x (InstanceOf(x, I) \rightarrow \neg Concept(x, C)); \quad (23)$$

- антисимметричность отношений (Если "X" шире "Y", то "Y" не может быть шире "X"):

$$\forall x, y (Broader(x, y) \rightarrow \neg Broader(y, x)); \quad (24)$$

- обратные отношения:

$$\forall x, y (Broader(x, y) \leftrightarrow Narrower(y, x)). \quad (25)$$

- верификация типов отношений через Wikidata:

$$\forall d (data(d) \wedge HasRelationType(d, k, c, subclassOf) \wedge WikidataRelation(k, "subclassOf", c) \rightarrow () \exists r (SubClassOf(r, k, c) \wedge AddToOntology(r))), \quad (26)$$

где $data(d)$ – входные данные (например, ключевые фразы и концепты); $HasRelationType(d, k, c, w)$ – проверит отношение типа `subclassOf` между k (ключевая фраза) и c (концепт) с весом w ; $WikidataRelation(k, "subclassOf", c)$ проверка, что в *Wikidata* есть утверждение k *subclassOf* c .

Предложенная онтологическая модель предоставляет формальную структуру для решения задачи персонализации на основе интересов пользователя в системах искусственного интеллекта и машинного обучения. Ключевыми особенностями предложенной онтологической структуры являются следующие:

- *гибкость*, так как новые ключевые фразы автоматически включаются в существующие концепты. Например, можно добавлять любые типы отношений (например, *"Python"* $\rightarrow$ использует-для $\rightarrow$ *"анализ данных"*);

- *персонализация*. В отличие от традиционных онтологий, которые часто используются для обобщённого представления знаний в домене, данная онтология ориентирована на персонализацию. Веса (W) обновляются на основе поведения пользователя (например, частота просмотра статей по теме). Она строится на основе данных, связанных с конкретным пользователем и его взаимодействиями с контентом, что означает поддержку обновления в реальном масштабе времени на основе новых данных. Это позволяет повысить точность поиска концептов, релевантных интересам пользователя.

- *автоматизированный поиск связей*. Система самостоятельно находит связи через: текстовую аналитику (векторное сходство); внешние онтологии (Wikidata, DBpedia).

- *поддержка многозначности*. Одна ключевая фраза может относиться к разным концептам в зависимости от контекста. В этом случае используется разработанный модифицированный алгоритм устранения неоднозначности смысла слов.

Чтобы подчеркнуть эффективность предложенной онтологической структуры и детализировать семантику её компонентов, в следующем пункте описывается разработка алгоритма персонализации интересов пользователя, который использует созданную онтологическую модель при последующем применении для решения задачи персонализации результатов поиска веб-сайтов.

3.2. Алгоритм персонализированного поиска веб-сайтов

В данном пункте представлен алгоритм персонализации результатов поиска веб-сайтов, учитывающий интересы пользователя для выдачи релевантных и адаптированных результатов [117]. Общий процесс алгоритма включает в себя создание профиля пользователя, сортировку документов в соответствии с этим

профилем, а также обновление профиля пользователя. Подробное объяснение этих шагов представлено ниже.

3.2.1. Алгоритм создания профиля пользователя

Построение профиля пользователя включает реализацию многоэтапного процесса извлечения ключевых фраз и концептов, устранения их неоднозначности, а также выявления отношений между ними в разработанной онтологической модели (рис. 3.3).

Предположим, что K – это набор веб-страниц, посещенных пользователем U .

Входные данные: Текстовое содержимое веб-страниц, посещенных пользователем.

Выходные данные: Профиль пользователя, структурированное представление интересов пользователя, включая выявленные ключевые фразы и их связанные веса.

Для создания профиля пользователя выполняется ряд шагов, которые включают в себя:

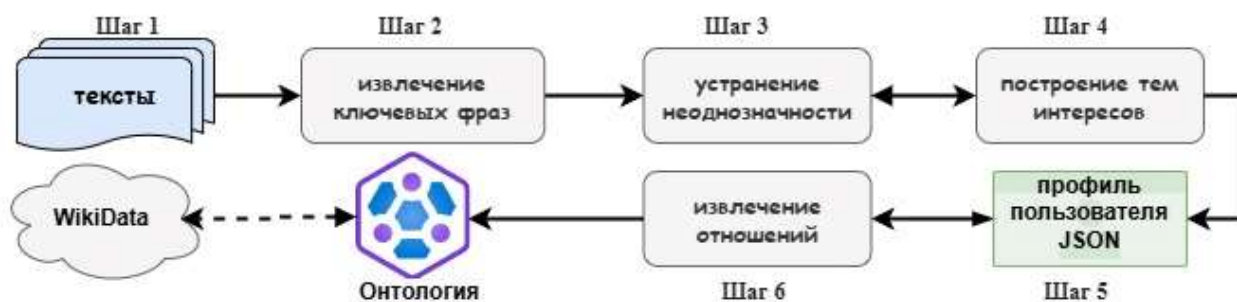


Рисунок 3.3 – Шаги алгоритма построения профиля пользователя

Шаг 1. Сбор содержимого: сбор содержимого каждой веб-страницы, посещенной пользователем. Основные заголовки и подзаголовки сохраняются в отдельном списке для дальнейшего использования, поскольку они считаются показателем важности ключевых фраз, которые могут в них появиться.

Процесс сбора текстовых данных с веб-страниц реализуется с помощью метода веб-скрапинга, который позволяет автоматически извлекать информацию из HTML-документов. Для решения этой задачи используется библиотека BeautifulSoup⁶, обеспечивающая эффективный парсинг и обработку веб-контента.

Шаг 2. Извлечение ключевых фраз. Применить разработанный модифицированный метод *FBKE* (п. 2.1) для извлечения семантически релевантных ключевых фраз. Для структурированного контента, такого как веб-страницы и научные статьи, функция взвешивания ключевых фраз учитывает позиционную информацию. Более высокий вес придается ключевым фразам, которые появляются в видных позициях, таких как:

- тег заголовка (Title): получает наивысший вес, указывающий на видимость и влияние заголовка;
- вес заголовков (H1, H2, H3 и т.д.): вес уменьшается постепенно от H1 к H3 и далее к менее значимым заголовкам;
- основное содержание (Body): ключевые слова, упомянутые в основном тексте, которые вносят меньший вклад в вес по сравнению с теми, что находятся в заголовках или тегах.

Формально, можно определить функцию W , которая вычисляет вес ключевой фразы k на основе её *FBKE* веса f_k и её позиции p_k :

$$W(k) = f_k \cdot p_k, \quad (27)$$

$$p_k = \begin{cases} \alpha & \text{если } k \text{ в заголовке} \\ \frac{\alpha}{i} & \text{если } k \text{ в подзаголовке } H_i, \\ 1 & \text{если } k \text{ в основном тексте} \end{cases} \quad (28)$$

где α – экспериментально установленная константа со значением 4 (обычно $\alpha \in [2,5]$).

⁶ [Beautiful Soup Documentation — BeautifulSoup 4.4.0 documentation](#)

На этом этапе создаются ключевые фразы (n -граммы), где n находится в диапазоне от 1 до 4. Для каждой фразы существует *контекстный вектор* (S-BERT) и вес, определенный в соответствии с функцией ранжирования алгоритма *FBKE* с учетом веса позиции.

Шаг 3. Устранение неоднозначности слов. Векторы, построенные для каждого из ключевых слов (униграмм), извлеченных на предыдущем этапе, являются обобщенными, в отличие от векторов биграмм и триграмм. Чтобы улучшить векторизацию неоднозначных ключевых слов, выполняется этап устранения неоднозначности с использованием модифицированного алгоритма, разработанного в пункте 2.2 (Lesk-S-BERT). Алгоритм выбирает синсет, наиболее близкий к данной фразе, из всех доступных в WordNet. Затем вектор соответствующего синсета V_s , определенный алгоритмом, объединяется с вектором фразы V_p для формирования контекстного вектора V_c .

Оба вектора – оригинальный вектор фразы и контекстный вектор, отражающий ее значение – сохраняются. Контекстный вектор рассчитывается следующим образом:

$$\text{контекстный вектор } V_c = \text{average}(V_p, V_s). \quad (29)$$

Следующим шагом является группировка ключевых фраз и слов в кластеры, представляющие темы интересов пользователя. Фразы с наибольшим весом будут использоваться для построения профиля.

Шаг 4. Создание тем интересов пользователя. Для создания профиля пользователя применяется иерархический алгоритм кластеризации векторов ключевых фраз. Это позволяет выявить тематические интересы пользователя, где каждый кластер соответствует определенной теме. Алгоритм осуществляет многоуровневое разделение фраз на кластеры и подкластеры на основе вычисления семантической близости между векторными представлениями.

Критически важным параметром является порог сходства, определяющий объединение фраз в единую тему. Экспериментальным путем установлено, что оптимальное значение данного порога составляет 60%. Что касается внутренней структуры темы (подкластеров) и взаимосвязей между ее элементами, их определение осуществляется с помощью алгоритма, подробно описанного в пункте 3.1.

На рисунке 3.4 представлен механизм создания тем интересов с использованием алгоритма иерархической кластеризации.

Как показано на диаграмме, при пороге сходства 68% данные разделяются на 5 тематических кластеров интересов, каждый из которых содержит соответствующие ключевые фразы. Внутри каждого кластера наблюдаются подгруппы, которые согласно алгоритму выявления связей (п. 3.1) могут формировать узкоспециализированные тематические подкатегории (подтемы), а также группы семантически эквивалентных терминов (синонимичные ряды).

После формирования кластеров интересов они взвешиваются в соответствии с весами фраз, составляющих каждый из них, где вес кластера интересов t_i представляет собой среднее значение весов фраз в нем, и он определяет степень интереса пользователя к теме в целом. Вес темы интересов определяется следующей формулой:

$$w_{t_i} = \frac{1}{|P_{t_i}|} \sum_{p \in P_{t_i}} w_p, \quad (30)$$

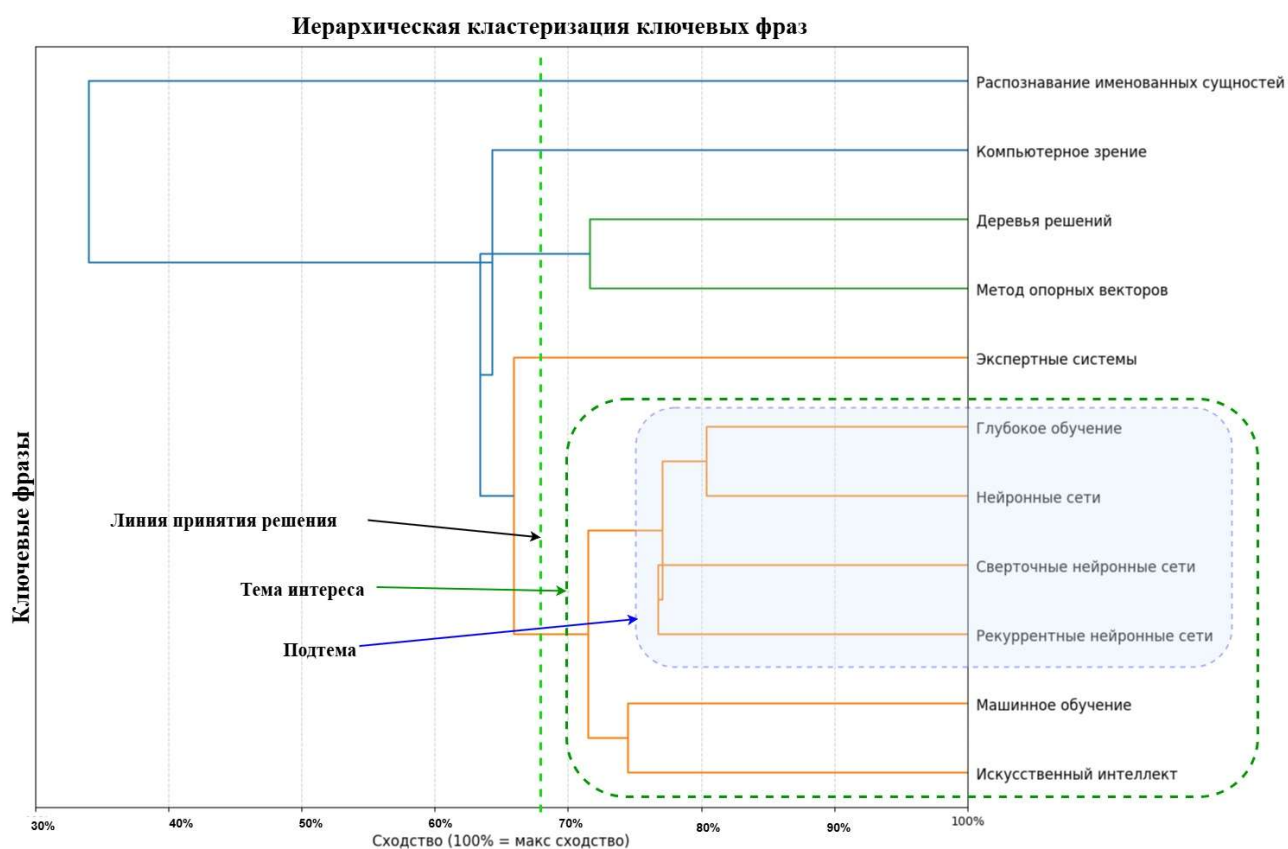


Рисунок 3.4 – Создание тем интересов с использованием иерархической кластеризации ключевых фраз

где P_{ti} – количество фраз в кластере t_i . А итоговый вес ключевой фразы w_p на уровне темы интересов t_i рассчитывается как среднее значение её весов во всех соответствующих документах (веб-страницах):

$$w_p = \frac{1}{|D_p|} \sum_{d \in D_p} w_{p,d}, \quad (31)$$

где p – ключевая фраза. D_p – множество документов (веб-страниц), в которых встречается фраза p . $w_{p,d}$ – вес фразы p в документе d , рассчитан по формуле (29).

Шаг 5. Хранение профиля пользователя. Для хранения профиля пользователя используется документ-ориентированная структура данных *JSON*. Данная структура позволяет эффективно хранить и извлекать данные о профиле пользователя. У каждого пользователя есть массив тем интересов. Эти темы интересов содержат собственные векторные представления и веса, что позволяет

создать детальное представление предпочтений пользователя. Внутри каждой темы находится список ключевых фраз, каждая из которых включает свое собственное векторное представление и индивидуальный вес.

Сгенерированная структура тем и ключевых фраз может служить в качестве пользовательского профиля и использоваться в задачах персонализации контента. Однако ее семантическая глубина остается ограниченной, так как нет формального описания связей отношений между фразами и темами. Основная примка предложенного алгоритма построения профиля заключается в интеграции сильных сторон концептуально-ориентированных и онтологически-ориентированных подходов. Для этого в JSON-файл профиля добавляется дополнительное поле *Relations*, описывающее связи между ключевыми фразами, а также между ними и темами. Поиск этих отношений осуществляется в описанной в пункте 3.1 онтологической модели. Данный поиск происходит на основе применения описанного в следующем 6-ом шаге алгоритма обновления профиля пользователя с отношениями.

Шаг 6. Извлечение семантических отношений. Данный процесс предназначен для оптимальной организации внутренней структуры тематических кластеров путем их разделения на специализированные подкатегории и группы синонимичных ключевых фраз. Основная задача заключается в рекурсивном разбиении кластеров на семантически однородные группы, что позволяет создать иерархически организованное пространство пользовательских интересов. Данный алгоритм выполняет проверку внутренних взаимосвязей тематических кластеров, выявленных на этапе построения тем интересов, с использованием семантической меры сходства между ключевыми фразами.

Входные данные алгоритма – профиль в формате JSON (J), содержащий темы интересов (Topics) и связанные ключевые фразы (KWS) с их весами

(Weights), с пустыми полями *Relations* (для будущих отношений). Результат работы алгоритма – профиль пользователя с построенными связями.

Для обеспечения вычислительной эффективности был установлен минимальный и максимальный предел количества ключевых фраз, представляющих каждую тему, в диапазоне от 1 до 30. Ключевые фразы, которые соответствуют конкретной теме удаляются и заменяются с течением времени с использованием функции временного уменьшения, на этапе обновления профиля пользователя, как описано в подпункте 3.2.3.

Результатом данного процесса является семантически насыщенный онтологический профиль пользователя. Далее, в следующем подпункте, описано как строится профиль, который используется для персонализации контента под пользователя.

3.2.2. Процедура ранжирования документов при персонализации интересов пользователя

Для заданного набора текстовых документов D , требующих персонализированного ранжирования на основе профиля пользователя, где документы могут представлять как результаты поискового запроса, так и контент веб-сайта [117], предлагаемый алгоритм осуществляет сортировку материалов в соответствии с их релевантностью к конкретному пользователю. Далее описывается механизм сопоставления документов с онтологическим профилем пользователя для генерации персонализированных результатов, соответствующих интересам, заданным в профиле. Предполагается, что исходный профиль создан в соответствии с шагами, описанными в предыдущем подпункте и структурирован в виде трехуровневой иерархии:

- уровень 1: основные темы интересов (T_1-T_k);
- уровень 2: подтемы внутри каждой темы (S_1-S_m);
- уровень 3: индивидуальные ключевые фразы (K_1-K_p).

1. Инициализация и предварительная обработка: все полученные документы преобразуются в векторные представления с использованием того же метода векторизации тем интересов и ключевых фраз в профиле (S-BERT).

Вводятся следующие переменные:

- весовые коэффициенты: $\alpha = 0.5$, $\beta = 0.3$, $\gamma = 0.2$;
- пороговые значения: $\theta_{topic} = 0.4$ (минимальное сходство с темой);
 $\theta_{sub-topic} = 0.6$ (минимальное сходство с подтемой).

2. Иерархическая оценка релевантности. Для каждого документа $d \in D$:

а) оценка на уровне тем:

$$R_{topic}(d) = \max_{t \in T} [\cos(v_d, v_t) \cdot w_t], \quad (32)$$

если $R_{topic}(d) < \theta_{topic} \rightarrow$ пропустить документ;

б) оценка на уровне подтем (для $t^* = \operatorname{argmax}(R_{topic}(d))$):

$$R_{sub}(d) = \max_{s \in S_{t^*}} [\cos(v_d, v_s) \cdot w_s], \quad (33)$$

$$\text{если } R_{sub}(d) < \theta_{sub-topic} \rightarrow R(d) = \alpha \cdot R_{topic}(d), \quad (34)$$

иначе перейти к следующему шагу;

в) детальная оценка на уровне фраз (для $s^* = \operatorname{argmax}(R_{sub}(d))$):

$$R_{phrase}(d) = \frac{\sum_{k \in K_{s^*}} \cos(v_d, v_k) \cdot w_k}{|K_{s^*}|}. \quad (35)$$

Итоговый балл:

$$R(d) = \alpha \cdot R_{topic}(d) + \beta \cdot R_{sub}(d) + \gamma \cdot R_{phrase}(d). \quad (36)$$

3. Ранжирование и выдача результатов:

- сортировка документов по убыванию $R(d)$;
- отсортированный список топ-N документов предоставляется пользователю в качестве персонализированных результатов поиска.

Следует отметить, что сначала система оценивает семантическую близость документов с текущим профилем пользователя и ранжирует их по степени релевантности к профилю, после чего извлекает ключевые слова из документов, к

которым пользователь проявил интерес, используя метод *FBKE*. Затем выполняется устранение неоднозначности смысла ключевых слов (униграмм) для точного определения их смысла в контексте интересов пользователя. На завершающем этапе профиль пользователя обновляется в автономном режиме, включая новые ключевые слова, что позволяет в дальнейшем предоставлять более персонализированные и точные результаты поиска. Полный процесс обновления описан на следующем подпункте.

3.2.3. Процедура обновления профиля пользователя

Ключевые фразы на данном этапе имеют рассчитанные веса, основанные на их частоте и семантической близости к содержанию страницы, из которой они извлечены. Процедура обновления профиля пользователя новыми ключевыми фразами включает следующие шаги:

1. Если ключевая фраза уже присутствует в профиле:
 - a. Корректируется вес самой фразы. В данной работе вводится новая метрика (41);
 - b. Обновляется вес темы интересов, к которой фраза относится (формула 32);
2. Если фраза новая:
 - a. Определяется её сходство с существующими темами интересов через косинусную меру сходства (1). Вес фразы определяется как значение, полученное по формуле (33) и нормализованное относительно частотного распределения всех новых фраз (41);
 - b. Фраза добавляется к наиболее близкой теме. Затем вес данной темы интересов обновляется по (32);
 - c. Если сходство ниже порогового значения (экспериментально установлено 30%), фраза считается принадлежащей к новой теме, которая создаётся автоматически;

3. Обработка ненаблюдаемых фраз: если ключевая фраза не встречается в последующих результатах дальнейшего поиска, ее вес линейно уменьшается со временем с использованием фактора распада (34):

$$w_{(t)} = w_0 \cdot (1 - \lambda t), \quad (37)$$

где $w_{(t)}$ – вес в момент времени t , w_0 – начальный вес, λ – скорость распада (0.1 для 10% в день), t – время в днях, прошедшее с момента последнего наблюдения ключевого слова.

Данная мера подходит для временных трендов (мемы, новости), где подойдёт линейное уменьшение. Для долгосрочных интересов (например, профессиональных навыков) лучше всего использовать экспоненциальный спад:

$$w_{(t)} = w_0 \cdot e^{-\lambda \cdot t}. \quad (38)$$

В данной диссертации предлагается использовать новую метрику для обновления весов ключевых фраз в профиле пользователя, которая учитывает временную динамику и частотное распределение новых фраз. Веса динамически перераспределяется между всеми фразами: устаревающие фразы теряют часть своего веса по (39, 40), который переходит к новым, более релевантным (41). Предлагаемая метрика описывается следующей формулой:

$$w_{(t)} = w_{(t-1)} + \beta \cdot \frac{f_i}{\sum f_j}, \quad (39)$$

где $w_i(t)$ – вес фразы i в момент времени t , t – время с последнего обновления (в днях), β – коэффициент усиления новых фраз (например, 1.0), f_i – частота появления фразы i в новых данных, $\sum f_j$ – сумма частот всех новых фраз.

Параметр $w_{(t-1)}$ для новых фраз это w_p , который обозначает вес, рассчитанный согласно формуле (33).

Для описания применения предложенной метрики рассмотрим следующий пример. Исходный профиль пользователя содержит три ключевые фразы с весами $w_{(t-1)}$: "машинное обучение", "Python" и "анализ данных". В процессе работы

системы зафиксированы новые вхождения: фраза "нейросети" (частота=2) как новая и "Python" (частота=1) как существующая. Параметры алгоритма: коэффициент затухания $\lambda=0.05$, коэффициент усиления $\beta=1.0$. Исходный вес w_p каждой фразы рассчитывается при извлечении по формуле 33. Полный процесс обновления весов представлен в таблице 1.

Таблица 3.1 – процесс обновления весов фраз в профили пользователя

Фраза	$w_{(t-1)}$	w_p	частота	Вес	Δ
машинное обучение	1.5	–	0	$1.5 \cdot e^{(-0.05)} \approx 1.42$	- 0.07
анализ данных	0.8	–	0	$0.8 \cdot e^{(-0.05)} \approx 0.76$	- 0.04
нейросети	0	0.7	2	$0.7 \cdot (2/3) \approx 0.46$	–
Python	0.8	0.5	1	$0.8 + 0.5 \cdot (1/3) \approx 0.96$	0.16

Как видно из представленных результатов, фразы, отсутствующие в новых данных ("машинное обучение", "анализ данных"), демонстрируют ожидаемое экспоненциальное затухание весов (примерно на 5%), в то время как повторяющиеся фразы (например, "Python") получают увеличение веса пропорционально их частоте встречаемости. Особый интерес представляет добавление новой фразы «нейросети», вес которой определялся на основе её значимости в тексте и относительной частоты встречаемости в новых данных.

При этом сохраняется баланс общего весового бюджета системы. Таким образом, пример наглядно иллюстрирует ключевые функциональные возможности алгоритма: плавное забывание устаревающих понятий, динамическую адаптацию к изменяющимся интересам пользователя и правильную интеграцию новых терминов в профиль.

В данном пункте подробно описаны шаги алгоритма построения профиля пользователя на основе разработанных ранее модифицированных метода извлечения ключевых фраз и алгоритма устранения неоднозначности смысла слов,

а также описана метрика обновления весов ключевых фраз в профиле. В следующем разделе представлено описание результатов вычислительных экспериментов, подтверждающих эффективность предложенных автором решений.

3.3. Выводы по разделу

В данной главе описана разработка онтологической модели, предназначенной для динамического обновления профиля пользователя на основе его взаимодействия с текстовыми данными. Предложенная онтологическая модель, характеризуется введением двух новых компонентов: набор контекстно-зависимых связей, автоматически извлекаемых из текстовых данных, и весовые функции для определения уровня интереса пользователя к релевантным темам, что позволяет повысить адаптивность онтологии к изменяющимся контекстам.

Также описана разработка алгоритма персонализации интересов пользователя на основе онтологической модели. Алгоритм включает реализацию многоэтапного процесса, начиная с извлечения ключевых фраз, за которым следует устранение неоднозначности значений слов и завершается группировкой взвешенных ключевых фраз в отдельные "темы интересов". В процессе работы алгоритм определяет веса ключевых фраз, учитывая частоту их встречаемости, контекстуальную релевантность контенту и их позиционную значимость в тексте. Кроме того, применяется процедура устранения неоднозначности для униграмм, которые появляются в различных контекстах, что значительно повышает эффективность результатов персонализации в условиях семиотической неоднозначности. Результаты вычислительных экспериментов, описанные в четвертой главе, подтверждают эффективность предложенных алгоритмов и их вклад в повышение точности решений задачи персонализации.

ГЛАВА 4. РАЗРАБОТКА ПРОГРАММНОГО ПРИЛОЖЕНИЯ И ПРОВЕДЕНИЕ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

В данной главе диссертации представлено описание разработки программного приложения, а также подготовки и проведения вычислительных экспериментов, направленных на оценку производительности предложенных модели, метода и алгоритмов в сравнении с аналогами. Оценка каждого результата выполняется в рамках индивидуальной задачи на специализированных эталонных данных для этой задачи, а затем анализируется в контексте решения задачи персонализации интересов пользователя при поиске веб-страниц и адаптации контента веб-сайтов.

4.1. Модульная архитектура программного приложения

Предложена модульная архитектура программного приложения (рис. 4.1), реализующего разработанные модели и алгоритмы для обработки и анализа текстов на естественном языке, включая извлечение ключевых фраз и концептов, устранение неоднозначности слов, извлечение отношений между понятиями и поиск веб-страниц [118, 119].

Программная система реализована на Python версии 3.12 с использованием ключевых библиотек, таких как *Beautifulsoup*, *NLTK*, *scikit-learn* и *FastAPI* [120, 121]. Для работы использовался ноутбук *HP* с процессором Intel Core i5 с тактовой частотой 1,99 ГГц.

Пользовательский интерфейс написан на HTML и JavaScript [122] и позволяет проводить вычислительные эксперименты для оценки производительности разработанных алгоритмов. Для этой цели интерфейс предоставляет возможность загрузки файлов и выбора типа операции, которую

необходимо выполнить. Кроме того, приложение предоставляет инструменты для визуализации результатов, проведения оценок и сравнения различных моделей.

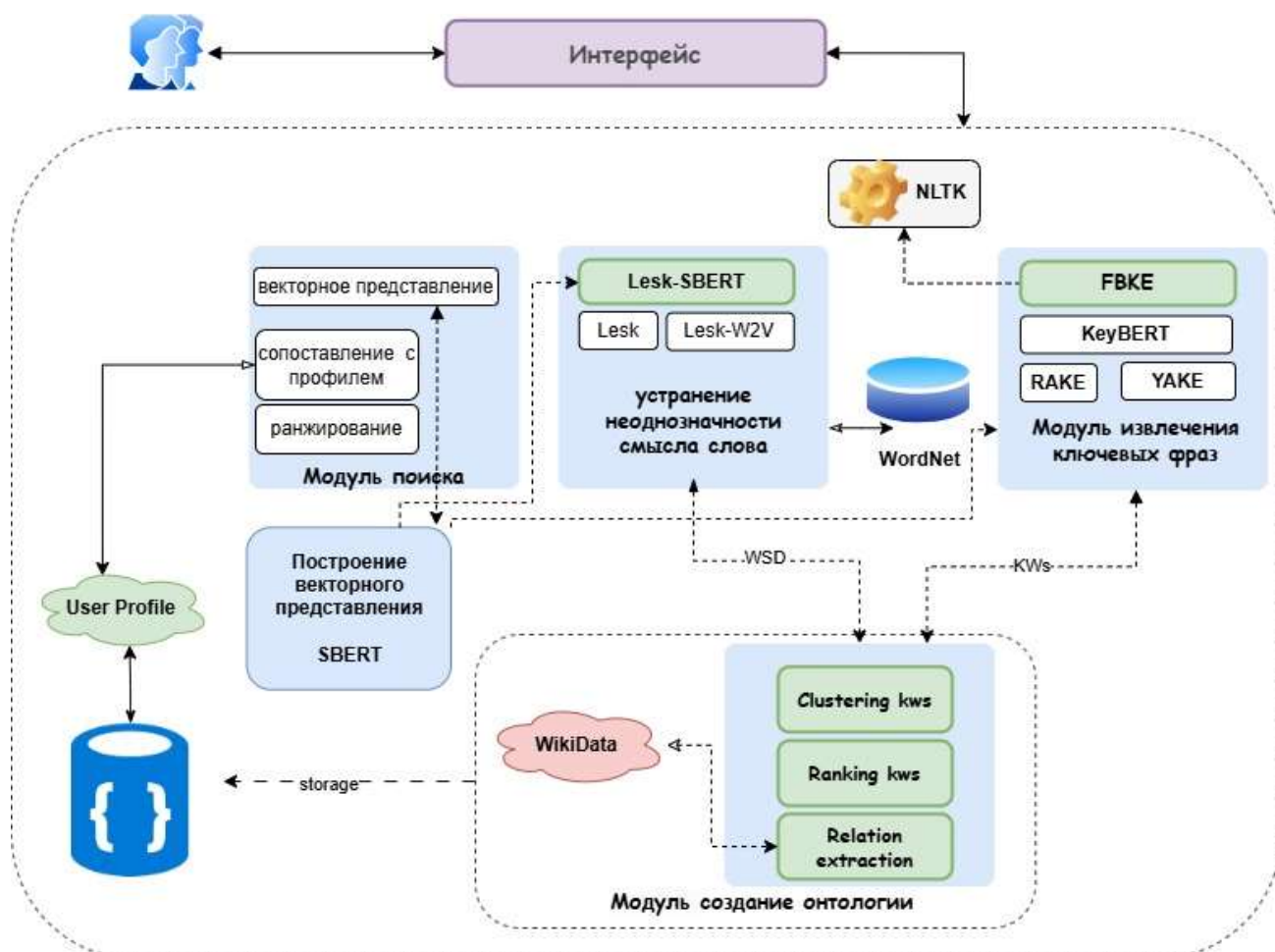


Рисунок 4.1 – модульная архитектура программного приложения системы персонализации веб-контента

Серверное веб-приложение было разработано с использованием *FastAPI* – одного из самых быстрых и популярных фреймворков для создания API на Python. Благодаря асинхронной обработке запросов, *FastAPI* обеспечивает высокую производительность, что особенно важно при работе с большими объемами данных, а также при выполнении ресурсоемких операций, таких как извлечение, сортировка и анализ документов. На рисунке (4.1) представлены ключевые модули, среди которых наиболее значимыми являются:

Модуль поиска отвечает за выполнение задачи ранжирования результатов поиска с использованием БД и модуля векторизации. Этот модуль принимает на вход поисковый запрос и список релевантных результатов. Тексты кодируются с использованием S-BERT, после чего результаты сортируются в соответствии с их соответствием профилю пользователя, хранящемуся в базе данных.

Модуль устранения неоднозначности слов отвечает за выполнение процесса устранения неоднозначности слов, полученных в результате этапа извлечения ключевых фраз. Он ожидает входные данные, такие как целевое слово и его контекст (текстовое предложение). Извлекаются возможные значения слова (синсеты) из WordNet, а затем выполняется устранение неоднозначности с использованием одного из алгоритмов: LESK, модифицированный LESK или алгоритм, разработанный в данной работе, который требует кодирования текстов с использованием S-BERT.

Модуль извлечения ключевых фраз ожидает текст на естественном языке и тип алгоритма, который будет использоваться. Для алгоритма FBKE требуется кодирование текста с использованием S-BERT.

Приложение использует базу данных MongoDB, которая является NoSQL-базой данных, известной своей масштабируемостью и возможностью управления неструктурированными данными [123]. MongoDB используется для хранения таких данных, как загруженные текстовые документы, профили пользователей, онтологии в формате JSON и результаты вычислительных экспериментов.

Модуль построения онтологии вызывается автоматически после ранжирования результатов поиска для конкретного пользователя. Модули вызываются в следующем порядке: извлечение ключевых фраз, устранение неоднозначности слов, группировка ключевых фраз и извлечение отношений. Полученные концепты, фразы и отношения автоматически связываются с

текущим профилем, хранящимся в базе данных, с помощью процедуры обновления профиля, описанного в подпункте 3.1.3.

В этом пункте описана разработка модульной архитектуры программного приложения, которое имитирует работу части системы искусственного интеллекта и машинного обучения для обработки и анализа текстов на естественном языке, а также выполнения персонализированного поиска веб-страниц.

В следующем пункте описываются вычислительные эксперименты, проведенные с использованием разработанного программного приложения. Представлены независимые эксперименты для каждой из задач, выполняемых приложением, а затем проводится комплексная оценка производительности этих алгоритмов в контексте задачи персонализации результатов поиска веб-страниц.

4.2. Результаты эксперимента по анализу эффективности модифицированного метода *FBKE* и настройка его параметров

Эффективность модифицированного метода FBKE оценивается по сравнению с несколькими надежными методами путем их оценки при выполнении задачи извлечения ключевых фраз. Приведем описание использованных наборов данных и метрик оценки:

Наборы данных. В таблице 4.1 представлены наборы данных, применённые для оценки методов извлечения ключевых слов/фраз. Набор данных включает коллекцию документов и набор эталонных ключевых фраз, созданные экспертами для каждого документа.

SemEval2010 (SE10) [124] состоит из 244 полных научных статей, извлеченных из цифровой библиотеки ACM (Association for Computing Machinery)⁷. Статьи относятся к четырем различным областям исследований в

⁷ Международная научная и профессиональная организация, основанная в 1947 году, которая занимается развитием информатики, компьютерных технологий и смежных областей.

области информатики: распределенные системы; поиск и извлечение информации; распределенный искусственный интеллект; социальные и поведенческие науки (экономика). Каждая статья имеет набор ключевых слов, назначенный автором, и набор ключевых фраз, назначенных профессиональными редакторами.

Таблица 4.1 – Наборы использованных данных

Набор данных	Wiki20	Theses100	Fao30	SE10	Sh08
Количество документов	20	100	30	244	1231
Количество слов на документ	6177.65	4728.8	4777.7	8332,3	3901,31
Среднее количество ключевых слов	36.5	7.67	33.23	16.47	44.69
Неявные ключевые слова	51.8%	47.6%	41.7%	11.3%	13.6%
n-грамм (n>3)	5.48%	6.39%	0.8%	9.2%	2.58%

FAO(Food and Agriculture Organization dataset) [125] состоит из 30 полнотекстовых документов, случайно выбранных из репозитория FAO (Продовольственная и сельскохозяйственная организация Объединенных Наций), где эталонные ключевые слова были вручную назначены шестью профессиональными аннотаторами FAO.

Wiki20 [126] состоит из 20 отчетов о технических исследованиях на английском языке, охватывающих различные аспекты информатики. Каждый отчет содержит в среднем 35 эталонных ключевых слов, назначенных специалистами в предметной области.

Набор данных *Theses100* [84] состоит из 100 полных магистерских и докторских диссертаций. Тематика диссертаций весьма разнообразна и включают химию, информатику, экономику, психологию, философию, историю и многое другое.

Schutz2008 (Sh08) [71] – это набор данных, основанный на полнотекстовых статьях, собранных из *PubMed Central*, который включает более 26 миллионов

ссылок на биомедицинскую литературу из *MEDLINE*, журналов по естественным наукам и онлайн-книг. Использованная выборка состоит из 1,231 статьи, отобранной из *PubMed Central*. Ключевые слова, определенные авторами, скрыты в статьях и используются в качестве эталонных (золотых) ключевых слов.

Для обработки исходных данных тексты были разделены на токены, все слова приведены к нижнему регистру и разбиты на небольшие сегменты, а также удалены стоп-слова. На этапе извлечения возможных ключевых слов/ключевых фраз пунктуация была сохранена, чтобы отфильтровать n -граммы, которые не принадлежат одной и той же части предложения.

Метрики оценки качества. Для оценки результатов используются средняя точность на K элементах «*Mean average precision at K, MAP@K*», которая вычисляет среднюю точность по множеству запросов, учитывая порядок релевантных элементов в результатах поиска:

$$MAP@K = \frac{1}{Q} \sum_{q=1}^Q \frac{1}{R} \sum_{k=1}^K Precision@k \cdot rel(k), \quad (40)$$

где Q – множество ключевых фраз, определенных экспертами; R – количество ключевых фраз в документе, которые соответствуют ключевым фразам; $rel(k)$ – бинарная функция сопоставления, равная 1 если ключевая фраза на позиции k соответствует ключевой фразе q , и 0 иначе.

Precision@K (Точность на позиции k): точность на позиции k в ранжированном списке ключевых фраз документа:

$$Precision@K = \frac{\text{релевантные элементы}}{K}. \quad (41)$$

Таким образом, данная метрика позволяет оценить, насколько хорошо система упорядочивает ключевые фразы в документе в соответствии с ключевыми фразами, определенными экспертами.

Диапазон значений $MAP@K$ находится между 0 и 1, где, 0 означает, что ни один из релевантных элементов не был найден или правильно ранжирован. 1

означает, что все релевантные элементы были найдены и расположены в верхней части списка.

Реализация модифицированного метода и анализ результатов. Эксперименты проводились на 5 различных наборах данных, и производительность предложенного модифицированного метода *FBKE* сравнивалась с четырьмя проанализированными ранее методами, а именно, мера *TF-IDF*, *TextRank*, *YAKE* и *KeyBert*⁸.

Первым шагом в реализации каждого метода является формирование n-грамм ключевых слов / фраз-кандидатов. Согласно статистике набора данных (табл. 4.1), количество n-граммов, состоящих из более чем трех слов, очень мало, поэтому n был выбран от 1 до 3 для всех методов.

Для предложенного метода «FBKE» после формирования n-граммов частота каждого типа рассчитывалась по формуле (14), а затем все типы объединялись в один словарь, содержащий (ключевые слова / фразы) и их частоту. В качестве кандидатов на следующий этап были отобраны 300 ключевых фраз с наибольшей частотой.

S-BERT⁹ использовалась для кодирования документа и его фраз-кандидатов. S-BERT допускает последовательность максимальной длины 384 слов и отображает их в 768-мерное плотное векторное пространство. Косинусная мера сходства использовалась для определения близости между документом и его словами-кандидатами.

Для реализации TF-IDF использовалась функция, предоставленная библиотекой *sikit-learn*. Для методов *TextRank*¹⁰ и *YAKE*¹¹ использовалась

⁸ <https://pypi.org/project/keybert/>

⁹ Использовалась версия SBERT all-mpnet-base-v2 https://www.sbert.net/docs/pretrained_models.html

¹⁰ Реализация TextRank для извлечения ключевых слов в Python <https://summanlp.github.io/textrank/>

¹¹ Реализация предоставлена автором <https://github.com/LIAAD/yake>

реализация, предоставленная авторами. Все методы были реализованы на языке *Python* в среде *Google Collab*.

KeyBERT возвращает ранжированный список ключевых фраз, в котором более длинные фразы обычно занимают более высокие позиции. Это означает, что прямое сравнение его результатов с FBKE при совместной генерации n -грамм разной длины (от 1 до 3 слов) может привести к смещенным результатам. Для обеспечения объективности сравнения оценивалась производительность KeyBERT и FBKE отдельно для каждого типа n -грамм: отдельные слова (униграммы); словосочетания из двух слов (биграммы); словосочетания из трех слов (триграммы). В качестве тестовых данных достаточно было использовать наборы Wiki20 и Theses100, так как они имеют общий характер.

Для исследования влияния коэффициента β в формуле (17) на точность ранжирования ключевых фраз был проведен эксперимент с последовательным увеличением значения β от 0 до 1000. Результаты представлены в таблицах 4.2 и 4.3. Как следует из формулы (17):

- при $\beta=1$ оба фактора (сходство и частота) вносят равный вклад в расчет итоговой релевантности;
- при $\beta=0$ показатель MAP@K рассчитывается исключительно на основе частотных характеристик;
- увеличение $\beta>1$ приводит к доминированию фактора семантического сходства в оценке релевантности.

Анализ данных в таблице 4.2 показывает, что метод FBKE с параметром $\beta=1$ демонстрирует статистически значимое превосходство над базовыми методами с приростом производительности по метрике MAP@K в диапазоне от 1% (Theses100) до 6% (semEval2010, Schutz2008). При этом TF-IDF показывает ограниченное преимущество лишь в двух случаях: на наборе wiki20 при $k=5$

($\Delta=+0.01$) и на Theses100 при $k = [5, 10]$ ($\Delta=+0.03$), что свидетельствует о существенно более низкой эффективности по сравнению с FBKE.

Предложенный автором модифицированный метод продемонстрировал наивысшую точность ранжирования (FBKE (BEST)) при значениях параметра β в диапазоне 0-20, что отражено в таблице 4.2. Однако дальнейшее увеличение β свыше 50 приводило к постепенному снижению точности на всех исследуемых наборах данных. Максимальные значения метрики MAP@K были достигнуты на наборе Schutz2008, тогда как минимальные показатели зафиксированы для наборов semEval2010, Wiki20 и Theses100.

Таблица 4.2 – Точность ранжирования, измеренная с помощью MAP@K

	@k	TF-DF	YAKE	TEXTRANK	FBKE	FBKE (best)
Wiki20	@5	20,4	07,2	5,6	20,3	23,7
	@10	13,8	07,2	3,5	14,2	16
	@40	5,5	02,8	1,4	06,1	07
Theses100	@5	8,4	05,2	4,4	08,1	10,2
	@10	7,0	3,9	3,6	07,0	08,7
	@40	7,6	3,8	3,6	08	09,6
Fao30	@5	24,0	7,8	14,9	26	29,3
	@10	15,4	4,9	09,1	17	20,0
	@40	06,7	1,9	03,9	08,3	09,9
semEval2010	@5	13,9	10,3	03,1	16,8	19,8
	@10	09,5	06,9	01,9	10,8	12,4
	@40	09,3	05,1	01,4	09,9	10,7
Schutz2008	@5	44,8	11,4	12,4	50,8	50,9
	@10	34,0	07,8	13,9	41,4	41,6
	@40	16,7	03,9	06,6	23,1	23,1

Наблюдаемые низкие показатели эффективности метода могут объясняться двумя взаимосвязанными факторами. Во-первых, существенное влияние оказывает высокая доля неявных слов (40-50% в наборах данных wiki20 и Theses100), поскольку метод извлекает исключительно ключевые слова, фактически присутствующие в тексте. Во-вторых, критическую роль играет количество эталонных ключевых фраз, предоставленных экспертами: как

показывает пример SemEval2010, даже при низком проценте неявных слов (11.3%) ограниченное среднее количество эталонных ключевых фраз (16.47) снижает эффективность метода. Наиболее показательным является случай с набором данных Theses100, где сочетание обоих негативных факторов – высокой доли неявных слов (47.6%) и минимального количества эталонных ключевых фраз (7.67) – приводит к наихудшим результатам, что подтверждает комплексное влияние этих параметров на качество работы метода.

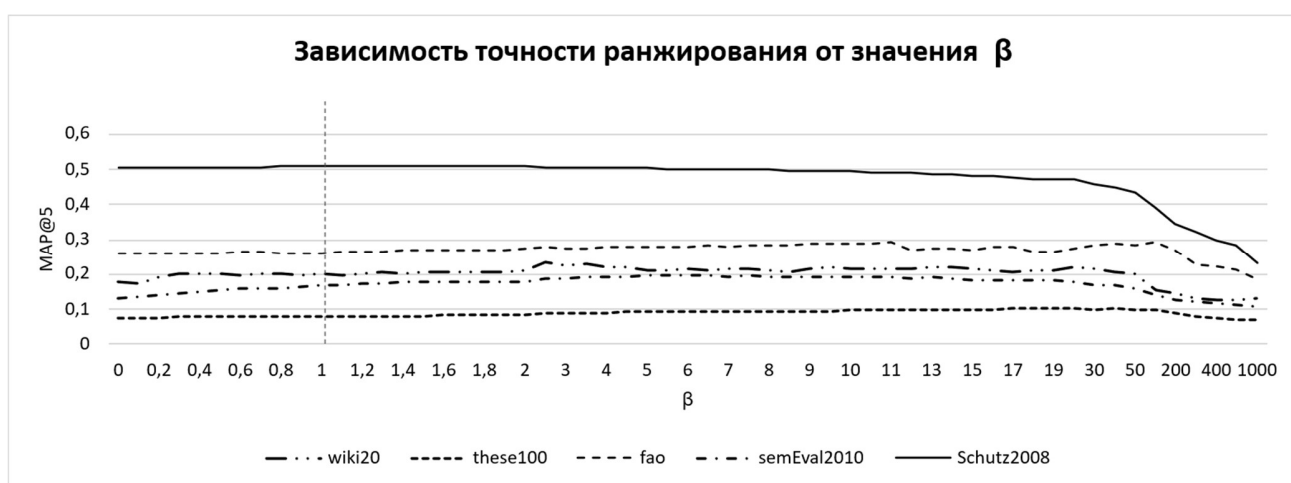


Рисунок 4.2 – Влияние параметров сходства и частоты встречаемости на итоговую точность ранжирования (метрика MAP@K)

Предложенный модифицированный метод FBKE демонстрирует существенное превосходство над KeyBert (таблица 4.3). Результаты эксперимента свидетельствуют, что FBKE превосходит не только статистические подходы (YAKE, TF-IDF), но и графовые методы (TextRank) на различных наборах данных.

Особого внимания заслуживает устойчивое преимущество FBKE при обработке объемных текстов. Как показывают результаты для наборов SemEval2010 и Wiki20 (таблица 4.1), содержащих относительно длинные тексты, метод сохраняет высокую эффективность, в отличие от современных аналогов.

Кроме того, можно также отметить, что FBKE в целом хорошо работает для различных полей и типов документов, таких как сельскохозяйственный Fao30;

Информатика SemEval2010 и Wiki20; медицинские науки Schutz2008 и набор данных Theses100, который включает различные области, такие как химия, компьютер и другие. Это важный момент, отмеченный в [78], а именно необходимость тестирования метода на различных наборах данных. Эта особенность очень важна в контексте построения пользовательского профиля и персонализации контента, поскольку она подтверждает способность метода работать с различными областями, что является типичным сценарием при создании разнообразного профиля.

Таблица 4.3 – Результаты экспериментов по сравнению *KeyBert* с *FBKE*

	@K	Wiki20		Theses100	
		Keybert	FBKE	Keybert	FBKE
Униграммы	5	10.00	14.7	6.3	7.75
	10	6.5	10.8	4.9	6.7
	40	2	04.2	4.9	7.8
Биграммы	5	0.8	17.6	0.6	04.4
	10	0.5	10.2	0.4	03.3
	40	0.2	04.1	0.4	03.5
Триграммы	5	0.5	1	0	00.5
	10	0.3	0.6	0	00.4
	40	0.1	0.2	0	00.4

Для подтверждения выводов автора был проведен вычислительный эксперимент, сравнивающий производительность разработанного модифицированного метода и аналогичных алгоритмов с большой языковой модели GPT-3, которая превзошел все исследуемые методы [106, 127]. Однако точность GPT-3 не идеальна, и цель эксперимента – продемонстрировать, что снижение точности разработанных и сравниваемых методов обусловлено не только их способностью определять правильные фразы, но и их способностью генерировать предложения и фразы, отсутствующие в исходном тексте. Эксперимент проводился на наборе данных *Inspec*.

Набор данных *Inspec* состоит из 2000 рефератов статей из научных журналов по компьютерным наукам. Набор данных *Inspec* состоит из 2000 аннотаций статей

из журналов по компьютерным наукам. Каждый документ имеет набор ключевых слов, назначенных вручную, некоторые из которых присутствуют в словаре Inspec, а некоторые отсутствуют в нем.

Для реализации алгоритма на основе GPT-3 была использована бесплатная версия модели GPT-3, позволяющая отправлять 60 запросов каждый час¹². Следующий фрагмент кода (рис. 4.3) иллюстрирует функции взаимодействия с языковой моделью GPT-3.

```

1  import pandas as pd
2  from processing import *
3  from freeGPT import Client
4  model = "gpt3"
5  import json
6
7  my_prompt = """Extract the top 20 key phrases from the following text,
8                ranking them according to their importance,
9                and return the response in JSON format: {key phrases: list} """
10
11 def get_kws(my_prompt, doc):
12     prompt = my_prompt + doc # doc: документ в обработке
13
14     try:
15         response = Client.create_completion(model, prompt) # Sending request
16         return response
17     except Exception as e:
18         print(e)
19         return [] # return empty list
20

```

Рисунок 4.3 – Листинг кода для извлечения ключевых фраз с помощью GPT-3

GPT-3 показывает высокие результаты в извлечении ключевых фраз, превосходя другие методы даже при $K=10-20$. Это подтверждает его точность в выделении важных фрагментов (таблица 4.4).

Высокая производительность GPT-3 объясняется архитектурой трансформера и механизмом внимания, которые позволяют параллельно обрабатывать последовательности данных, фокусируясь на важных частях входных данных и игнорируя менее значимые детали. Именно это и требуется для задачи извлечения

¹² freeGPT 1.3.5 <https://pypi.org/project/freeGPT/>

ключевых фраз, поскольку модель может концентрироваться на наиболее релевантных словах или фразах.

Таблица 4.4 – Точность алгоритмов, измеренная с помощью MAP@K

Методы	MAP@K			
	@1	@5	@10	@20
TF	0,18	0,096	0,058	0,046
YAKE	0,25	0,115	0,078	0,075
RAKE	0,23	0,107	0,087	0,079
GPT-3	0,445	0,313	0,23	0,198
Разработанный метод FBKE	0,29	0,18	0,12	0,08

Однако стоит отметить, что использование GPT-3 ресурсоёмко, особенно при работе с большими наборами данных или в реальных приложениях с ограниченными ресурсами. В таких случаях применение предложенного автором модифицированного метода или других алгоритмов, таких как YAKE или RAKE, может оказаться лучшим выбором, чем GPT-3, в сценариях, где важна скорость выполнения в ущерб незначительному снижению качества результатов.

В дополнение к предыдущим экспериментам предложенный модифицированный метод FBKE был применен в контексте задачи построения векторного представления текстов на основе концептов, что привело к значительному улучшению производительности метода векторизации при решении задач классификации и кластеризации [128–131].

В итоге, предложенный модифицированный метод успешно обрабатывает тексты из различных областей, включая сельское хозяйство, компьютерные науки, медицину и химию, что подтверждает его универсальность и надежность. Важным преимуществом метода является его независимость от корпусов данных: он способен извлекать ключевые слова из отдельных документов без необходимости использования внешних коллекций, что особенно полезно в условиях отсутствия обучающих данных. Эти особенности делают метод практичным и применимым для задач персонализации контента и построения пользовательских профилей.

В этом пункте описано проведение вычислительного эксперимента по оценке эффектности модифицированного метода извлечения ключевых фраз в задаче их выделения с учётом эталонных данных, определённых экспертами. Результаты эксперимента демонстрируют, что модифицированный метод превышает базовые аналоги по точности MAP@K на 1–3% для всех тестовых областей (сельское хозяйство, компьютерные науки, медицина, химия).

В следующем пункте анализируется работа модифицированного алгоритма устранения неоднозначности смысла слов, а затем разработанные метод и алгоритм оцениваются совместно в рамках единого конвейера при решении задач персонализации результатов поиска и контента веб-сайта.

4.3. Результаты эксперимента по анализу эффективности модифицированного алгоритма устранения неоднозначности слов

Эффективность алгоритма устранения неоднозначности слов оценивалась путем его сравнения с несколькими эталонными методами на стандартных наборах данных, разработанных для задач WSD. Среди методов, с которыми проводилось сравнение, можно выделить:

UKB В качестве примера можно рассмотреть метод UKB, который использует алгоритмы случайного блуждания по семантическим сетям, в частности, по структуре WordNet. Данный метод применяет алгоритм Personalized PageRank [132], инициализируемый на основе контекста целевого слова. Аналогичный графовый подход реализован в системе *Babelfy*, предназначенной для устранения лексической неоднозначности. *Babelfy* использует алгоритм случайного блуждания с перезапуском по семантической сети *BabelNet* [133], объединяющей WordNet с другими лексическими ресурсами, включая Wikipedia и Wiktionary. В основе метода лежит эвристика сверхплотного подграфа, позволяющая отбирать семантические интерпретации с высокой степенью

когерентности входного текста. При этом оптимальная конфигурация *Babelfy* учитывает не только локальный контекст в пределах предложения, но и глобальную семантическую структуру всего документа [134].

Леск (простой) [99], представляет собой простой алгоритм WSD, основанный на знаниях, который основывает свои вычисления на расчете перекрытия между контекстом целевого слова и его определениями, которые даны в WordNet.

Расширенный Леск. Основываясь на том же принципе, в различных работах был адаптирован исходный алгоритм с учетом определений из связанных слов (*расширенный Леск*) [135] или путем вычисления распределительного сходства между определениями и контекстом целевого слова [103].

Леск (с встраиванием) [103]. В основе этого метода лежит упрощенный алгоритм *Леска*, однако он использует функцию сходства слов, определенную в распределенном семантическом пространстве, для вычисления перекрытия определения с контекстом. Учитывая текст $w_1, w_2, \dots, w_n$ из n слов, он устраняет неоднозначность по одному, принимая во внимание сходство между глоссом, связанным с каждым смыслом целевого слова w_i , и контекстом. Выбирается значение, глосса которое имеет наибольшее сходство. Контекст может быть представлен подмножеством окружающих слов или всего текста, в котором это слово встречается. *BabelNet* используется как реестр смыслов, представляющая собой большую многоязычную семантическую сеть, построенную на базе WordNet и Wikipedia. В *BabelNet* лингвистические знания обогащены энциклопедическими концепциями, взятыми из Википедии.

Кроме того, проводилось сравнение с алгоритмом WN1'st sense [96], который выбирает кандидата (слово), считающегося первым синсетом в WordNet 3.0. Этот алгоритм служит базовым ориентиром для оценки эффективности других аналогов.

Наборы данных и метрика оценки. Для оценки предложенного алгоритма и его аналогов использовались пять эталонных наборов данных, представленных в таблице 4.5:

- Senseval-2 (Se2) [136]. Этот набор данных изначально был аннотирован WordNet 1.7. После стандартизации он состоит из 2282 смысловых аннотаций, включая существительные, глаголы, наречия и прилагательные;
- Senseval-3 (Se3) [137]. Этот набор данных состоит из трех документов, относящихся к трем различным областям (редакционная, новостная и художественная), всего 1850 смысловых аннотаций;
- SemEval-07 (Se7) [98]. Этот набор данных содержит 455 смысловых аннотаций только для существительных и глаголов. Изначально он был аннотирован с помощью инвентаризации WordNet 2.1;
- SemEval-13 (Se13) [138]. Этот набор данных включает тринадцать документов из разных доменов. Исходной инвентаризацией смысла был WordNet 3.0. Количество смысловых аннотаций – 1644, хотя учитываются только существительные;
- SemEval-15 (Se15) [134]. Это самый последний доступный на сегодняшний день набор данных WSD, аннотированный с помощью WordNet 3.0. Набор состоит из 1022 смысловых аннотаций в четырех документах из трех разнородных областей, биомедицины, математики вычислений и социальных вопросов.

Согласно [96], уровень неоднозначности каждого набора данных вычисляется как общее количество возможных смыслов разделенное на количество смысловых аннотаций. Значение неоднозначности может указывать на то, насколько сложным может быть данный набор данных. В этом случае SemEval-07 отображает самый высокий уровень неоднозначности среди всех наборов оценочных данных.

Таблица 4.5 – Наборы использованных данных

Набор данных	Кол-во документов	Кол-во предложения	Кол-во аннотаций	Неоднозначность
SensEval-2	3	242	2282	5.4
SensEval-3	3	352	1850	6.8
SemEval-7	3	135	455	8.5
SemEval-13	13	306	1644	4.9
SemEval-15	4	138	1022	5.5

Метрика оценки. В задачах устранения неоднозначности смысла слов для измерения соответствия между предсказанным сходством и принятым стандартом, т.е. среднего сходства по оценке экспертов-людей, используется известная F -мера.

F -мера – представляет собой гармоническое среднее между точностью и полнотой, где точность – это количество истинно положительных результатов, деленное на количество всех положительных результатов, включая те, которые не были идентифицированы правильно, а полнота – это количество истинно положительных результатов, разделенное на количество всех выборок, которые должны были быть идентифицированы как положительные:

$$F - \text{мера} = \frac{2 \times \text{точность} \times \text{Полнота}}{\text{точность} + \text{Полнота}}, \quad (42)$$

$$\text{Точность} = \frac{TP}{TP + FP}, \quad (43)$$

$$\text{Полнота} = \frac{TP}{\text{количество экземпляров истинности}}, \quad (44)$$

где TP – это сумма долей правильных прогнозов, FP – сумма долей неверных прогнозов. В частности, доля неверных прогнозов вычисляется из ложноположительных прогнозов системы.

В ходе вычислительного эксперимента были протестированы несколько вариантов реализации предложенного автором модифицированного алгоритма Lesk-S-BERT, с использованием различных аннотаций синсетов WordNet, только

определения синсетов, только примеры из синсетов, а затем с комбинацией определений и примеров. Для построения векторов встраивания слов используются S-BERT [19].

Для метода *Леска* (расширенного и с встраиванием) использовалась та же реализация, предложенная в [96], а для реализации простого *Леска* использовалась функция, предоставляемая библиотекой *NLTK* в *Python*. Для *UKB* использовалась конфигурация по умолчанию: с использованием графа *WordNet (UKB)*.

Результаты эксперимента приведены в таблице 4.6. Наблюдается разница в работе алгоритмов на разных наборах данных, между лучшими и худшими результатами есть значительный разрыв. Это указывает на то, что уровень неоднозначности данных влияет на сложность задачи. Это подтверждается низкими результатами алгоритмов устранения неоднозначности смысла слов на наборе *SemEval-07* – самом неоднозначном из всех (см. таблицу 4.5).

Эксперименты подтвердили, что комбинированное использование примеров и определений синсетов значительно повышает эффективность алгоритма по сравнению с их отдельным применением. При этом определения демонстрируют более стабильные результаты, чем примеры. Такое различие объясняется неполнотой данных: поскольку не все синсеты содержат примеры, их отсутствие приводит к ложноположительным срабатываниям.

Также, следует отметить, что предложенный алгоритм превзошел *Леск* (простой) и *Леск* (расширенный) по всем наборам данных. На наборе данных *SemEval-07* предложенный алгоритм превзошел все аналоги кроме *Леск* (с встраиванием).

Кроме того, экспериментальные результаты показали, что предложенный алгоритм продемонстрировал более высокую точность по сравнению с *UKB* на всех тестовых наборах данных.

Таблица 4.6 – Результаты устранения неоднозначности смысла слов,
рассчитанные с использованием F-меры

	F1-Мера (%)				
Методы	Se2	se3	se7	se13	se15
WN 1 st sense	66.8	66.2	55.2	62.95	67.8
<i>Леск</i> (простой)	40.29	35.94	22.63	38.27	41.21
<i>Леск</i> (расширенный)	50.6	44.5	32.0	53.6	51.0
<i>Леск</i> (с встраиванием)	63.0	63.7	56.7	66.2	64.6
UKB	56	51.7	39.0	53.6	55.2
Разработанный Алгоритм (LESK-BERT)					
<i>С расширенными аннотациями</i>	66.96	67.78	56.16	66.27	68.38
С определениями	53.06	48.48	34.28	56.2	59.98
С примерами	51.84	48.81	35.6	46.3	54.4

В данном пункте представлены результаты вычислительного эксперимента по оценке эффективности модифицированного алгоритма устранения неоднозначности слов, который по своему механизму имитирует хорошо известный вариант алгоритма *Леска* «с встраиванием». Разработанный алгоритм Lesk-S-BERT демонстрирует линейную зависимость от количества синсетов ($O(k)$) благодаря векторным операциям с эмбедингами S-BERT, тогда как классический алгоритм Леска требует квадратичных вычислений ($O(k \cdot m)$) для попарного сравнения слов. При этом сохраняется конкурентоспособная точность разрешения неоднозначностей, близкая к графовым методам, но с существенно меньшими вычислительными затратами.

В данном эксперименте предложенный модифицированный алгоритм был протестирован на стандартных наборах данных, которые включают таблицу истинности целевого слова в контексте конкретного предложения. Однако его

производительность может варьироваться при применении к другим наборам данных или в сочетании с различными алгоритмами для решения различных практических задач. Поэтому, для подтверждения этих результатов в следующем пункте эффективность данного алгоритма проверена в контексте задачи персонализации контента веб-сайтов путем его использования для построения онтологического профиля пользователя.

4.4. Результаты эксперимента по анализу эффективности алгоритма персонализации результатов поиска веб-сайтов

В данном пункте проводится вычислительный эксперимент для оценки эффективности алгоритма построения профиля пользователя и онтологии путем их применения в задаче персонализации результатов поиска веб-страниц.

Набор данных AOL4PS (AOL for Personalized Search) [6] представляет собой крупномасштабный ресурс, созданный для исследований в области персонализированного поиска. Он основан на реальных поисковых запросах и кликах пользователей AOL (America Online) и включает анонимизированные идентификаторы пользователей, временные метки (timestamps) и информацию о кликах (click data). Этот набор данных широко используется для анализа поведения пользователей, разработки персонализированных алгоритмов поиска и оценки их эффективности.

Данные состоят из трёх основных файлов:

1. Основные данные взаимодействий (data), содержат информацию о действиях пользователей, включая:

- AnonID – анонимный идентификатор пользователя;
- QueryIndex – индекс запроса, уникальный для каждого пользователя;
- QueryTime – временная метка запроса;
- SessionNo – номер сессии (сегмента взаимодействия);

- `DataType` – тип данных (0: исторические, 1: обучающие, 2: валидационные, 3: тестовые);
 - `DocIndex` – индекс документа, на который кликнул пользователь;
 - `CandiList` – список кандидатов (документов, предложенных в результатах поиска);
 - `ClickPos` – позиция кликнутого документа в списке кандидатов;
2. Поисковые запросы (`query`), содержит текст запросов:
- `Query` – текст запроса;
 - `QueryIndex` – индекс запроса (соответствует `QueryIndex` в файле `data`);
3. Документы (`doc`), содержит информацию о документах (`URL`, заголовки):
- `Url` – веб-адрес документа;
 - `DocIndex` – индекс документа (соответствует `DocIndex` в файле `data`);
 - `Title` – заголовок страницы.

Предварительная обработка данных. При использовании AOL4PS важно учитывать, что некоторые `URL`-адреса в наборе данных могут быть недоступны из-за истечения срока их действия или географических ограничений. Это может затруднить проведение экспериментов, особенно если требуется доступ к конкретным веб-страницам. Поэтому используется выборка данных, где все `URL`-адреса удовлетворяют требованиям доступности и актуальности. Отфильтрованная выборка набора данных включает 1000 выбранных пользователей. Анализ их журналов дал 243,148 уникальных `URL`-адресов.

Поскольку в AOL4PS отсутствуют метки релевантности, для оценки связи между запросом и документами используется модель S-BERT. Если оценка релевантности превышает 0.5, метка релевантности устанавливается равной 1, в противном случае – 0.

Для каждого пользователя (`AnonID`) данные разбиваются на сессии (`SessionNo`) – периоды активности с временными интервалами. Внутри сессии

анализируются история запросов (QueryIndex, QueryTime), клики (DocIndex, ClickPos) и списки кандидатов (CandiList). На основе исторических данных (DataType = 0) строится профиль пользователя. Основная задача системы заключается в построении персонализированного профиля пользователя на основе его исторических данных и последующем адаптивном ранжировании результатов поиска для новых запросов.

На первом этапе система анализирует исторические взаимодействия пользователя (запросы, клики, сессии) и формирует его профиль. Этот профиль служит основой для персонализации выдачи.

Для новых запросов система использует профиль пользователя, чтобы:

- определить релевантность документов в соответствии с его интересами;
- изменить порядок результатов поиска, подняв наиболее релевантные документы на верхние позиции;
- обеспечить соответствие стандартному набору данных, где релевантность определена на основе модели S-BERT (с порогом > 0.5).

Метрики оценки качества персонализации. Для оценки результатов используются *Mean Average Precision (MAP@K)*, которая вычисляет среднюю точность по множеству запросов, учитывая порядок релевантных элементов в результатах поиска (формула 42). Однако переменные в формуле 42 описаны в соответствии с задачей следующим образом:

Q – общее количество запросов; Rq – общее количество релевантных элементов для запроса q ; $rel(k) = 1$, если элемент на позиции k релевантен, иначе 0. *Precision@K (Точность на K элементах)* – доля релевантных результатов среди топ-K предложенных. Показывает, насколько точно система возвращает релевантные результаты в пределах первых K элементов.

Для оценки эффективности предложенного алгоритма персонализации результатов поиска в сравнении с другими методами были разработаны серии

экспериментальных пайплайнов (конвейеров обработки данных, Pipelines). Эти пайплайны направлены на демонстрацию индивидуального и совокупного влияния ключевых компонентов – извлечения ключевых фраз, устранения неоднозначности слов и извлечения семантических отношений (на основе онтологии) – на общую производительность процесса персонализации результатов поиска. Путем изоляции и анализа каждого этапа пайплайны показывают, как улучшения в этих компонентах способствуют более точным и контекстуально релевантным результатам поиска. В таблице 4.7 представлены реализованные варианты обработки данных (пайплайны), тогда как в таблице 4.8 приведены полученные результаты их применения.

Пайплайн 1 служит базовым уровнем. Низкие значения MAP@K и Precision@K в первом пайплайне обусловлены использованием базовых методов, таких как TF-IDF и *Леск*. TF-IDF не учитывает контекст, а *Леск* часто ошибается в случаях неоднозначности из-за зависимости от принципа пересечения, который не улавливает семантическое сходство и контекст. Этот пайплайн служит базовым уровнем.

Во втором пайплайне наблюдается улучшение за счет использования FBKE, который более эффективно извлекает ключевые слова, специфичные для предметной области. Однако ограничения *Леск* по-прежнему сдерживают дальнейшее повышение точности.

Третий пайплайн демонстрирует снижение производительности по сравнению со вторым, даже несмотря на внедрение предложенного алгоритма WSD, который более эффективно справляется с неоднозначностью. Это подчеркивает важность точного извлечения ключевых фраз, особенно принимая во внимание, что TF-IDF не учитывает семантические нюансы и не интегрирован с алгоритмом BERT для устранения неоднозначности.

Таблица 4.7 – Экспериментальные пайплайны для оценки персонализации

	Извлечение ключевых фраз	WSD	Онтология	Цель
Пайплайн 1	TF-IDF	<i>Леск</i>	нет	Базовый конвейер для демонстрации влияния традиционных методов извлечения ключевых фраз и WSD без учета семантических отношений
Пайплайн 2	FBKE	<i>Леск</i>	нет	Демонстрирует улучшение при использовании FBKE вместо TF-IDF для извлечения ключевых фраз при сохранении методов WSD
Пайплайн 3	TF-IDF	ЛЕСК _{S-BERT}	нет	Демонстрирует улучшение при использовании ЛЕСК _{S-BERT} вместо <i>Леск</i>
Пайплайн 4	FBKE	ЛЕСК _{S-BERT}	нет	Демонстрирует комбинированный эффект FBKE и ЛЕСК _{S-BERT} на точность профиля
Пайплайн 5	FBKE	ЛЕСК _{S-BERT}	Есть	Его основная цель – оценка эффективности онтологической модели. Показывает эффект интеграции семантических отношений на основе онтологии в процесс построения профиля.

Четвертый пайплайн объединяет *FBKE* и *Леск_{BERT}*, что обеспечивает высокую точность за счет использования контекстных встраивания слов алгоритмом *Леск_{BERT}* для более точного устранения неоднозначности. Это также подтверждает преимущества современных методов на основе трансформаторов.

Наконец, в пятом пайплайне достигаются наивысшие значения MAP@K и Precision@K благодаря интеграции онтологических отношений.

Таблица 4.8 – Результаты персонализации, измеренные с помощью $MAP@K$

	$MAP@5$	$Precision @5$	$MAP @10$	$Precision @10$
Пайплайн 1	42.02	48.33	37.13	41.4
Пайплайн 2	52.7	58.6	47.32	52.8
Пайплайн 3	49.6	52.9	41.42	47.88
Пайплайн 4	64.2	67.91	56.88	63.01
Пайплайн 5	68.05	70.95	59.83	67.92

Благодаря использованию онтологии система не только отображает ключевые слова на концепты, но и учитывает их семантические связи. Это приводит к созданию более богатого и контекстуально точного профиля пользователя, значительно повышая релевантность результатов поиска.

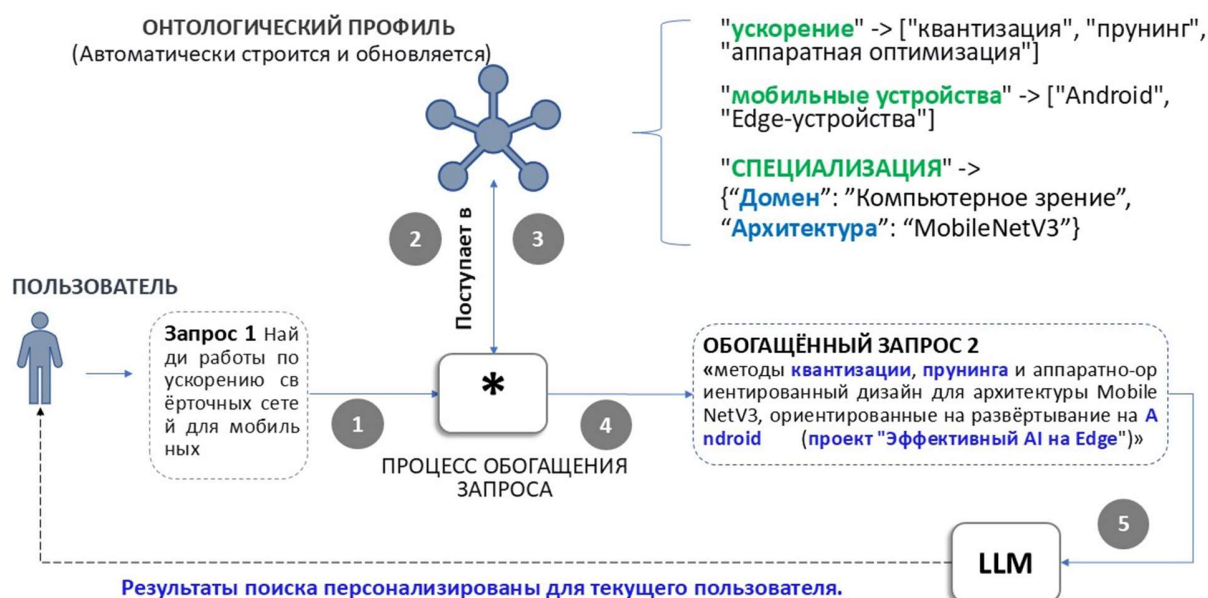
Систематическая оценка этих пайплайнов подтверждает эффективность предложенного алгоритма, демонстрируя его способность улучшать релевантность результатов поиска за счет продвинутого извлечения ключевых фраз, точного разрешения неоднозначности и семантического анализа на основе онтологии. В среднем предложенный алгоритм персонализации результатов поиска веб-страниц снижает частоту ошибок функционирования систем персонализации на 5 – 8 %.

Результаты, представленные в данном пункте, подтверждают, что качество онтологии напрямую влияет на эффективность систем персонализации результатов поиска веб-сайтов. Поэтому в следующем пункте проводится эксперимент по анализу эффективности процесса построения онтологии.

4.5. Результаты эксперимента по анализу эффективности онтологии

Для оценки эффективности онтологии, построенной из текста, применяются разработанные в данной работе алгоритмы обработки и анализа текстов на естественном языке в т.ч. модифицированный метод извлечения ключевых фраз,

модифицированный алгоритм устранения неоднозначности смысла слов и алгоритм построения профиля пользователя.



Полный процесс реализации онтологии описан ниже. На рисунках 4.4 и 4.5 представлен пользовательский интерфейс для построения и оценки онтологий, который позволяет выбрать подходящие методы и функции для выполнения каждого из описанных этапов.

Загрузка документов

Перетащите файлы или кликните для загрузки
Поддерживаемые форматы: TXT, PDF, DOCX

Язык текста:
Русский

Извлечение ключевых фраз

Метод извлечения:
YAKE (Unsupervised)

Диапазон N-грамм:
Unigrams, Bigrams и Trigrams

Максимальное количество фраз:
100 фраз

Word Sense Disambiguation

Метод WSD:
Алгоритм Lesk

Порог уверенности WSD:
0.7

Векторизация и кластеризация

Метод векторизации:
BERT Embeddings

Порог кластеризации (сходства):
0.6

Минимальный размер кластера:
3 фразы

Максимальное количество тем:
20 тем

Построить онтологию

Рисунок 4.4 – Интерфейс приложения для построения и оценки онтологии

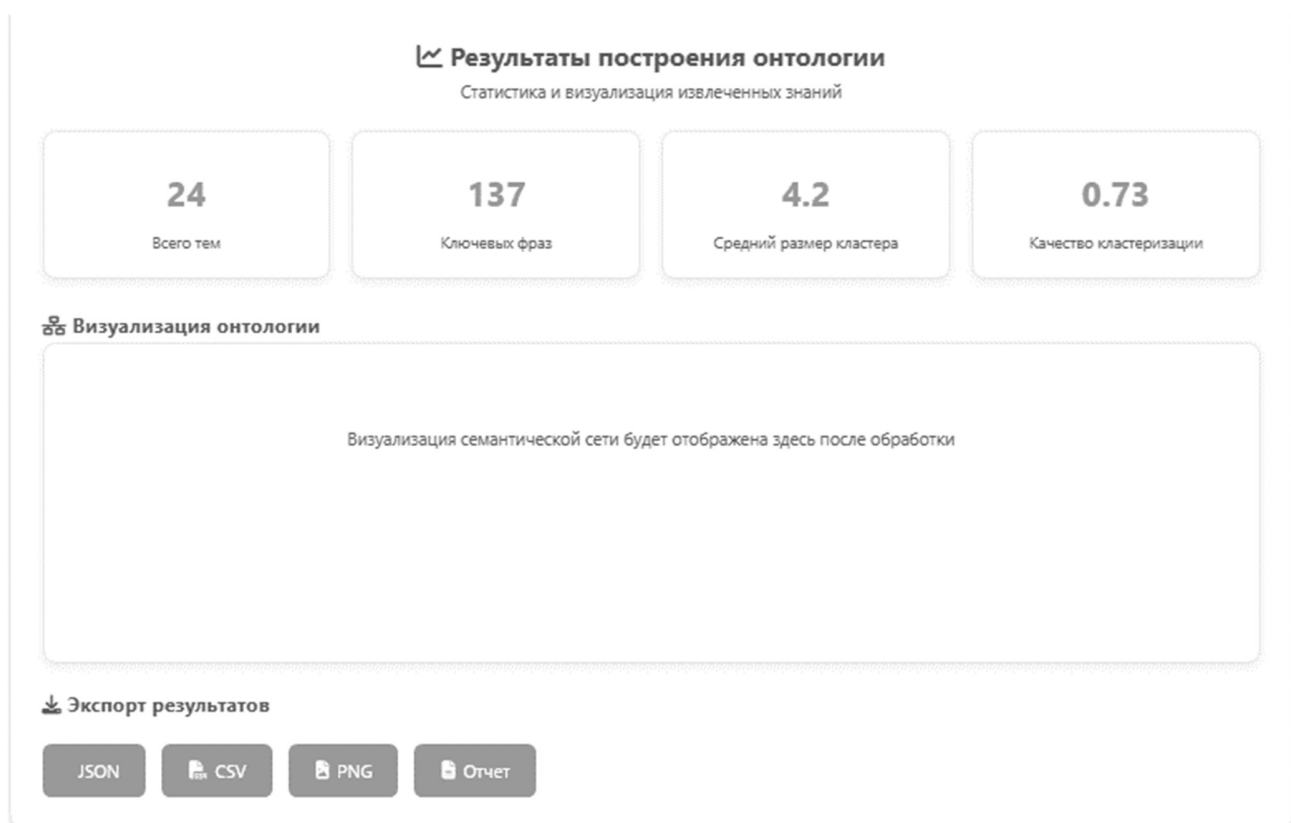


Рисунок 4.5 – Интерфейс приложения для построения и оценки онтологии

Процесс начинается с этапа предварительной обработки текстовых документов, которые представляют собой результаты поиска или любой другой текст, введенный пользователем для последующей обработки и преобразования в онтологию. Для извлечения ключевых фраз можно выбрать любой метод, например, Yake, Rake, TF-IDF, keyBERT или FBKE. В текущем эксперименте для извлечения ключевых фраз используется разработанный автором метод FBKE, который использует S-BERT для векторного представления предложений и возвращает для каждого документа список ключевых фраз вместе с их векторами. Данный этап может быть расширен за счет проведения семантического анализа с использованием библиотек типа SpaCy или других.

Устранение неоднозначности смысла слов. В данной реализации был выбран модифицированный алгоритм LESK-BERT, предложенный автором в диссертации, для определения подходящих синсетов для слов-униграмм и

корректировки их векторных представлений в соответствии с их значениями для лучшего соответствия контексту, в котором они используются.

Для формирования тем и подтем применяется агломеративная кластеризация со следующими параметрами:

- Метрика расстояния: косинусное расстояние
- Метод связи: используется average-linkage, где расстояние между двумя кластерами вычисляется как среднее арифметическое расстояний между всеми парами объектов в этих кластерах.

При этом интерфейс позволяет управлять следующими параметрами:

- Максимальное количество тем
- Пороговое расстояние (порог кластеризация): $\text{distance_threshold} = 0.4$
- Минимальный размер кластера: $\text{min_cluster_size} = 3$

Построение отношений. Отношения устанавливаются на основе семантического сходства с пороговой фильтрацией в соответствии с алгоритмом извлечения семантических отношений, описанным в подпункте 3.2.1, при этом в данном эксперименты рассматриваются только два типа отношений: синонимы и общее отношение (related), указывающее на наличие семантической связи без использования Викиданных.

$$\begin{aligned} \text{relation_type} = \{ \\ \text{синонимы (synonymous) если } \text{sim}(v_i, v_j) \geq \theta_{\text{syn}}, \\ \text{общее отношение (related) если } \theta_{\text{rel}} \leq \text{sim}(v_i, v_j) < \theta_{\text{syn}} \}, \end{aligned} \quad (45)$$

где порог синонимии: $\theta_{\text{syn}} = 0.9$, порог связанности: $\theta_{\text{rel}} = 0.6$.

После настройки параметров и выбора подходящих методов для построения онтологии из текста можно нажать кнопку «Построить онтологию», показанную на рисунке 4.5 интерфейса. Затем приложение выполняет построение онтологии и визуализирует её. Одновременно с этим вычисляется набор метрик,

демонстрирующих степень эффективности полученной онтологии. Далее приводится подробное описание этих метрик и результатов.

Кластерный анализ. Иерархическая кластеризация (Agglomerative Clustering) применяется для группировки семантически близких фраз. Алгоритм работает по принципу "снизу вверх", объединяя наиболее близкие кластеры на каждом шаге. Мера расстояния между кластерами вычисляется по формуле:

$$D(C_i, C_j) = 1 - \text{sim}(\mu_i, \mu_j), \quad (46)$$

где μ_i и μ_j - центроиды кластеров.

Для оценки качества разделения кластеров используются две метрики, Силуэтный коэффициент и Индекс Калински-Харабаза [139].

Силуэтный коэффициент (Silhouette coefficient) – метрика для оценки качества кластеризации, которая показывает, насколько хорошо объекты внутри кластеров отделены от объектов других кластеров. Она измеряет, насколько похож объект на свой собственный кластер (сплоченность) по сравнению с другими кластерами (разделимость).

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \quad (47)$$

где $a(i)$ - среднее расстояние внутри кластера, $b(i)$ - расстояние до ближайшего соседнего кластера.

Индекс Калински-Харабаза (Calinski-Harabasz Index) – метрика оценки качества кластеризации, основанная на соотношении межкластерной дисперсии (разделимость) и внутрикластерной дисперсии (компактность).

$$CH = \frac{\text{Межкластерная дисперсия}/(k-1)}{\text{Внутрикластерная дисперсия}/(n-k)}, \quad (48)$$

где k – число кластеров, n – число объектов.

Кроме того, используются следующие структурные метрики:

- Средний размер кластера, который показывает степень детализации онтологии:

$$ACS = \frac{1}{n} \sum_{i=1}^n |C_i|; \quad (49)$$

- Плотность отношений измеряет связанность онтологии:

$$RD = \frac{2 \cdot |E|}{|V| \cdot (|V| - 1)}, \quad (50)$$

Коэффициент когерентности показывает отношение внутрикластерной к межкластерной близости:

$$\text{Coherence_Ratio} = \text{IntraSim} / \text{InterSim}, \quad (51)$$

где IntraSim (внутрикластерное сходство) оценивает семантическую однородность:

$$\text{IntraSim}(C_i) = \frac{2}{|C_i| \cdot (|C_i| - 1)} \sum_{x, y \in C_i} \text{sim}(x, y), \quad (52)$$

а InterSim (межкластерное сходство) измеряет дискриминативность кластеров:

$$\text{InterSim} = \frac{1}{k(k-1)} \sum_{i \neq j} \text{sim}(\mu_i, \mu_j), \quad (53)$$

Сетевой анализ. Онтология представляется как взвешенный граф $G(V, E, w)$, где:

- $V = \{v_1, v_2, \dots, v_n\}$ - множество концептов (вершин);
- $E \subseteq V \times V$ - семантические отношения (ребра);
- $w: E \rightarrow [0,1]$ - функция весов, определяющая силу отношений.

Коэффициент кластеризации измеряет локальную плотность:

$$C_i = \frac{2 \cdot e_i}{k_i \cdot (k_i - 1)}, \quad (54)$$

где e_i – число связей между соседями вершины i , k_i – степень вершины, которая показывает количество связей.

Связные компоненты показывают структурную целостность онтологии. Компонента связности – это максимальное подмножество вершин, где между любой парой вершин существует путь.

Временная сложность алгоритма построения онтологии:

- Извлечение ключевых фраз: $O(n \cdot m)$ где n – количество документов, m – средняя длина документа
- Кластеризация: $O(n^2 \log(n))$ для иерархической кластеризации
- Построение отношений: $O(k^2)$ где k – количество кластеров

Экспериментальные результаты

В результате проведенного эксперимента была сформирована оптимальная кластерная структура, состоящая из 18 семантических кластеров, которые объединили 70 ключевых фраз. Средний размер кластера, составивший 3.89 элемента, свидетельствует о достижении баланса между излишней дробностью и чрезмерным обобщением, так называемой «зоны Златовласки». Данная структура демонстрирует репрезентативную группировку концептов, полноценное покрытие предметной области и смысловую целостность каждого кластера.

Важным результатом является высокая семантическая когерентность модели. Показатель внутрикластерного сходства, достигший значения 0.841, указывает на исключительное качество кластеризации. Анализ выявил наличие сильных семантических связей внутри кластеров, их четкую тематическую согласованность и формирование осмысленных семантических концептов в каждой группе.

Качество семантических отношений в построенной сети также оценивается как высокое. Средняя сила установленных связей находится на уровне 0.803, что подчеркивает их значимость и информационную ценность. Сеть характеризуется низким уровнем шума и подтверждает эффективность примененного принципа «качество над количеством».

Оценка метрических показателей подтверждает устойчивость модели. Значение силуэтного коэффициента (0.444) указывает на удовлетворительное

разделение кластеров с минимальным перекрытием и четкими границами, что оставляет потенциал для дальнейшей оптимизации до уровня 0.5+ (рис. 4.6).

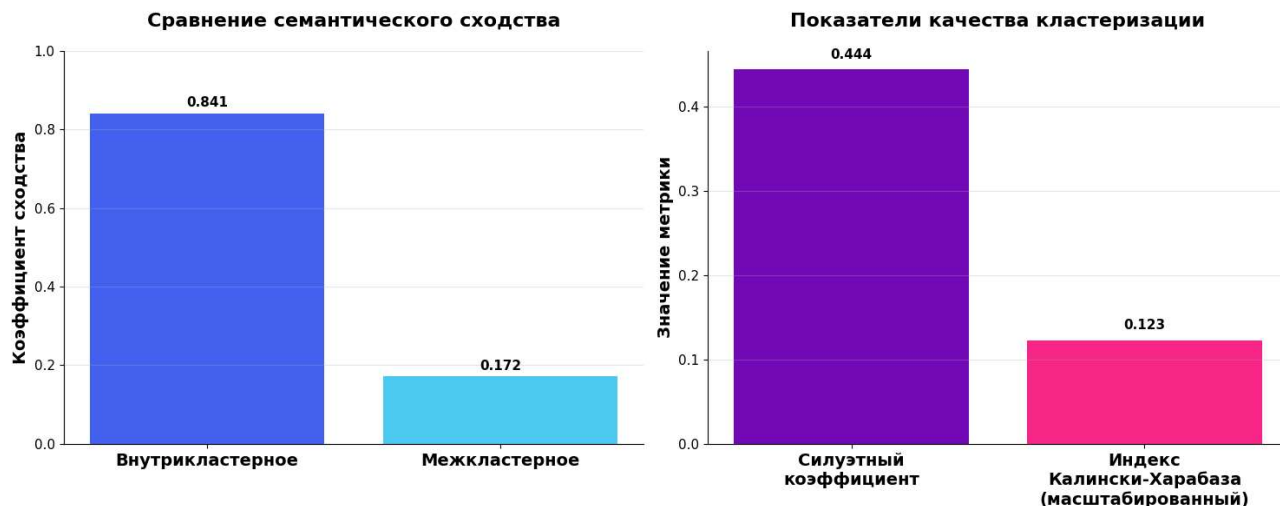


Рисунок 4.6 – Показатели качества кластеризации

Показатель плотности отношений (5.77) отражает оптимальную связность сети и сбалансированность ее структуры, обеспечивающую эффективные паттерны информационных потоков. Само собой разумеется, что количество общих связей (related) превышает количество связей-синонимов (synonyms), поскольку данная категория включает различные типы отношений, которые могут быть точно определены с использованием внешних источников знаний, таких как Wikidata (рис. 4.7). Однако в рамках данного эксперимента было решено ограничиться общей классификацией, поскольку основной целью исследования является оценка качества онтологии с точки зрения её структуры и согласованности, а также для снижения вычислительной сложности эксперимента.

Коэффициент ретенции, составивший 37% (сохранение 70 из 188 исходных фраз), свидетельствует о высокой точности фильтрации при сохранении полноты покрытия предметной области.

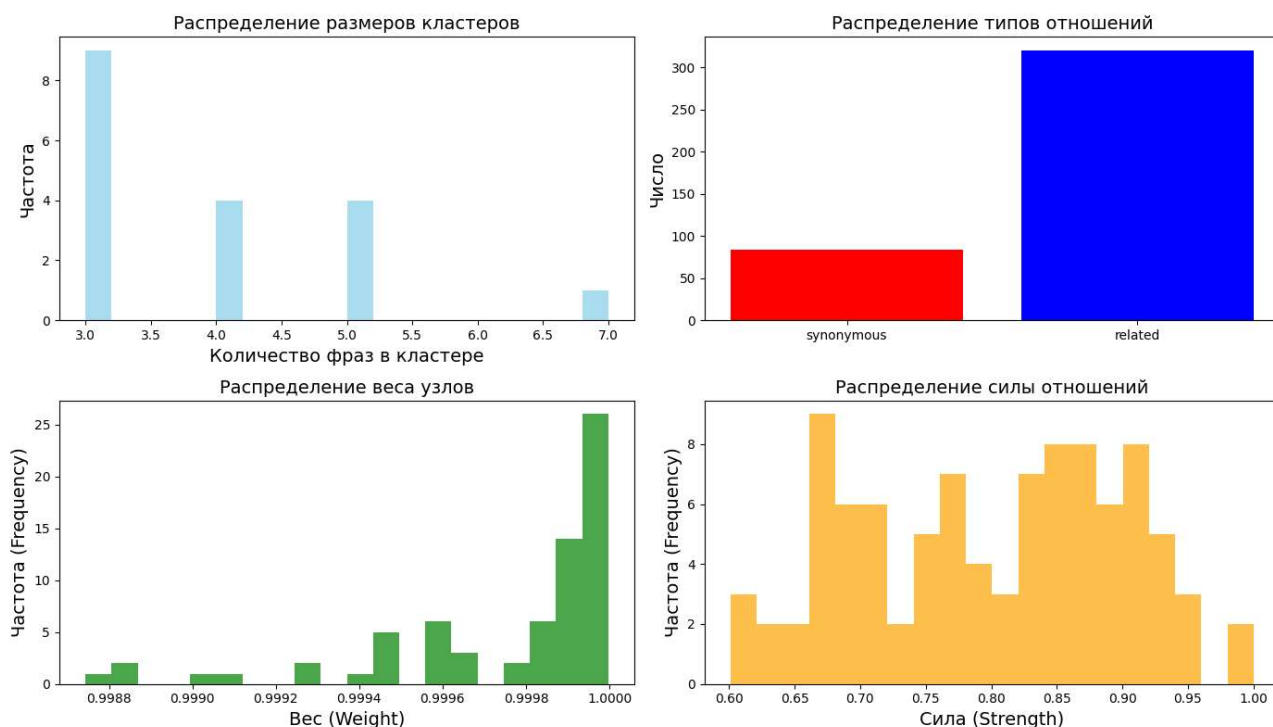


Рисунок 4.7 – Показатели качества построения отношений

Практическая значимость работы заключается в высоком потенциале применения полученной онтологической модели в таких областях, как рекомендательные системы, семантический поиск, управление знаниями и категоризация контента. Модель обеспечивает пользователям четкую тематическую структуру, осмысленные семантические связи и надежную организацию знаний.

Чтобы продемонстрировать значимость разработанных алгоритмов, таких как извлечение ключевых фраз, устранение неоднозначности смысла слов и автоматическое построение онтологического профиля пользователя, реализован пример с использованием программного приложения, описанного в пункте 4.1.

Представленный онтологический профиль используется для решения множества задач, не ограничиваясь только персонализацией поиска. Например, его можно применять для ранжирования содержимого веб-страниц в соответствии

с предпочтениями пользователя, например, в электронных библиотеках для индексации, классификации книг и их поиска.

Для примера, рассмотрим цифровую библиотеку, которая содержит разнообразную коллекцию книг из различных областей, включая политику, экономику и машинное обучение. Основная цель – упорядочить и организовать отображаемые элементы в соответствии с их соответствием предпочтениям пользователя. Предпочтения фиксируются и представляются через заранее созданный профиль пользователя, который строится на основе истории его просмотров и обучающих данных из предыдущих экспериментов. Эти алгоритмы позволяют системе динамически организовывать и отображать контент в соответствии с предпочтениями пользователя.

Чтобы продемонстрировать, как содержимое веб-сайта электронной библиотеки может быть персонализировано на основе профиля пользователя, мы можем создать список из 15 книг, охватывающих различные темы. Каждая книга будет иметь 5 ключевых фраз, извлеченных из ее описания и названия. Эти ключевые слова помогут определить релевантность книги интересам пользователя, которые представлены в онтологии (например, Machine Learning, Artificial Intelligence, Python и т.д.).

Предположим, у пользователя есть две основные темы интересов (рис. 4.8): Artificial Intelligence (Искусственный интеллект) и Programming (Программирование). Для Artificial Intelligence ключевые фразы включают "Artificial Intelligence" (Bec: 0.5), "AI" (Bec: 0.45), "ChatGPT" (Bec: 0.52), "Large Language Model" (Bec: 0.18) и "LLM" (Bec: 0.3). Для Programming ключевые фразы включают "Programming" (Bec: 0.66), "Python for Beginners" (Bec: 0.8), "Applied Python" (Bec: 0.9), "Data Structures" (Bec: 0.6), "FastAPI" (Bec: 0.5) и "Flask" (Bec: 0.62).

На рисунке 4.9 показан порядок книг в соответствии с предыдущим профилем. Можно заметить, что книги, которые включают ключевые фразы из обеих тем, занимают более высокие позиции, такие как «Python for Data Analysis» и «Data Science from Scratch: First Principles with Python». Стоит отметить, что термин "искусственный интеллект", присутствующий в профиле, не упоминается напрямую в названии или описании книги, а связь определяется на основе связанных ключевых фраз, таких как анализ данных и машинное обучение.

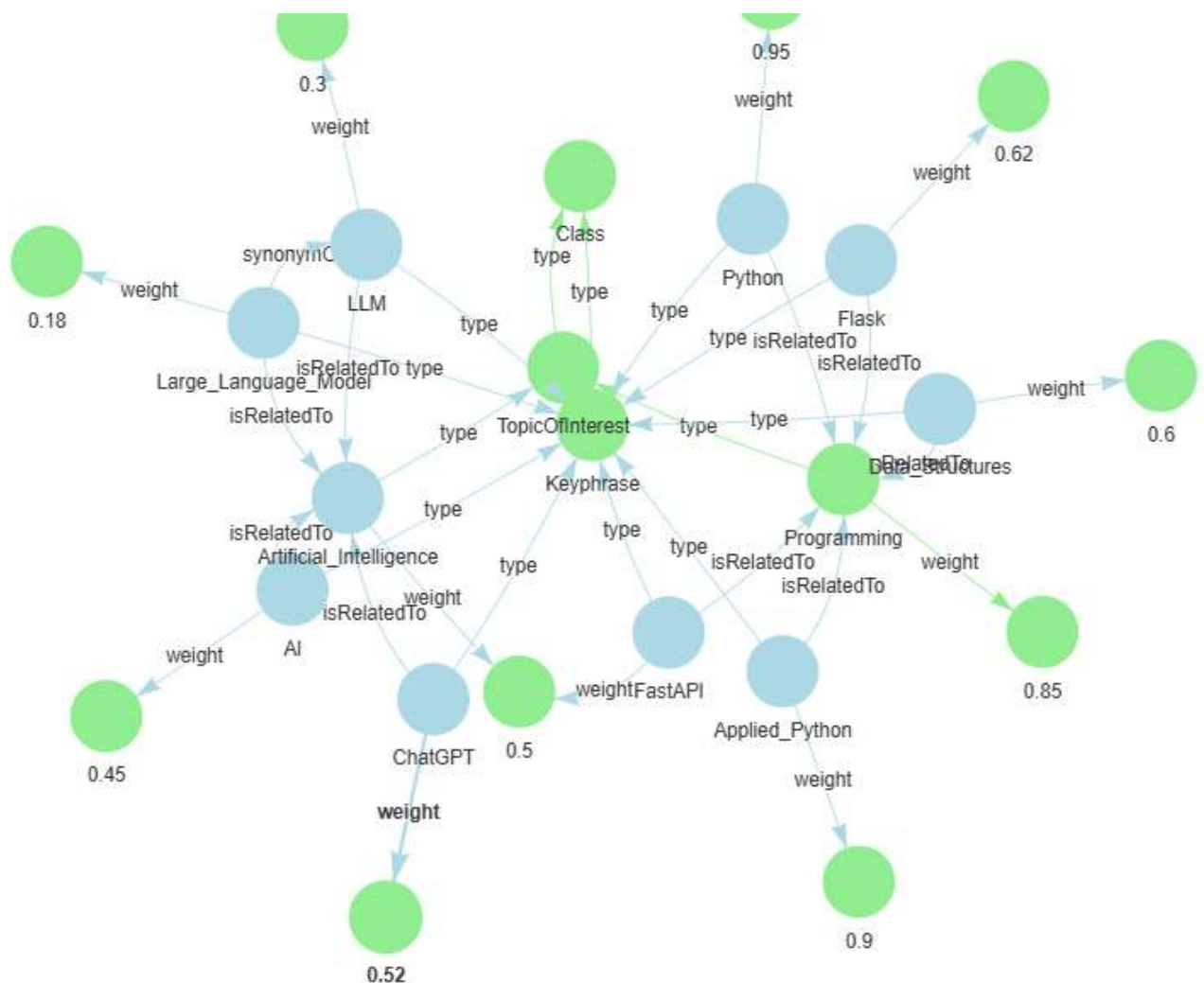


Рисунок 4.8 – Визуальное представление части онтологического профиля пользователя электронной библиотеки

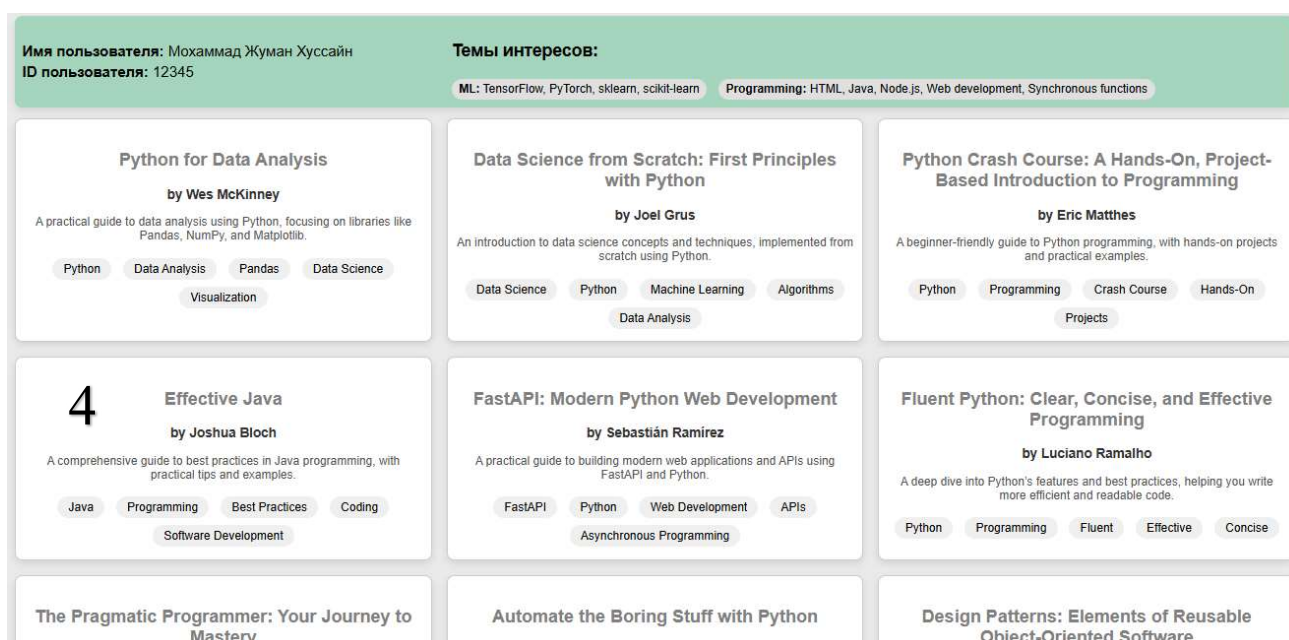


Рисунок 4.9 – Пример персонализации контента сайта для электронной библиотеки (случай 1)

В таблице 4.9 показано, как профиль пользователя влияет на порядок книг на веб-странице библиотеки путем добавления и удаления или изменения весов ключевых фраз.

Таблица 4.9 – Результаты эксперимента по ранжированию списка книг, в соответствии с профилем пользователя

Случай 1	1. Python for Data Analysis (0.2477) 2. Data Science from Scratch: First Principles with Python (0.2467) 3. Python Crash Course: A Hands-On, Project-Based Introduction to Programming (0.2388) 4. Effective Java (0.2254) 5. FastAPI: Modern Python Web Development (0.2220) 6. Fluent Python: Clear, Concise, and Effective Programming (0.2135) 7. The Pragmatic Programmer: Your Journey to Mastery (0.2126) 8. Automate the Boring Stuff with Python (: 0.2123) 9. Design Patterns: Elements of Reusable Object-Oriented Software (0.1903) 10. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (0.1837) 11. You Don't Know JS: Up & Going (0.1824) 12. Deep Learning by Ian Goodfellow (0.1812) 13. Artificial Intelligence: A Modern Approach (0.1786) 14. Eloquent JavaScript: A Modern Introduction to Programming (0.1762) 15. Clean Code: A Handbook of Agile Software Craftsmanship (0.1655)
----------	---

	16. HTML and CSS: Design and Build Websites (0.1424) 17. The Republic by Plato (0.0922)
Случай 2 Java (Вес = 0.2) без изменений в теме ИИ	Ranked Books Based on Weighted User Profile: 1. Python for Data Analysis (0.2237) 2. Data Science from Scratch: First Principles with Python (0.2230) 3. Python Crash Course: A Hands-On, Project-Based Introduction to Programming (0.2166) 4. Effective Java (0.2110) 5. FastAPI: Modern Python Web Development (0.2020) 6. The Pragmatic Programmer: Your Journey to Mastery (0.1950) 7. Fluent Python: Clear, Concise, and Effective Programming (0.1938) 8. Automate the Boring Stuff with Python (0.1932) 9. Design Patterns: Elements of Reusable Object-Oriented Software (0.1752) 10. You Don't Know JS: Up & Going (0.1688) 11. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (0.1653) 12. Deep Learning by Ian Goodfellow (0.1643) 13. Eloquent JavaScript: A Modern Introduction to Programming (0.1634) 14. Artificial Intelligence: A Modern Approach (0.1627) 15. Clean Code: A Handbook of Agile Software Craftsmanship (0.1521) 16. HTML and CSS: Design and Build Websites (0.1313) 17. The Republic by Plato (0.0847)
Случай 3 Java (Вес = 0.5) без изменений в теме ИИ	1. Python for Data Analysis (0.2407) 2. Data Science from Scratch: First Principles with Python (0.2404) 3. Effective Java (0.2377) 4. Python Crash Course: A Hands-On, Project-Based Introduction to Programming (0.2345) 5. FastAPI: Modern Python Web Development (0.2195) 6. The Pragmatic Programmer: Your Journey to Mastery (0.2141) 7. Fluent Python: Clear, Concise, and Effective Programming (0.2101) 8. Automate the Boring Stuff with Python (0.2100) 9. Design Patterns: Elements of Reusable Object-Oriented Software (0.1935) 10. You Don't Know JS: Up & Going (0.1876) 11. Eloquent JavaScript: A Modern Introduction to Programming (0.1820) 12. Deep Learning by Ian Goodfellow (0.1779) 13. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (0.1772) 14. Artificial Intelligence: A Modern Approach (0.1770) 15. Clean Code: A Handbook of Agile Software Craftsmanship (0.1674) 16. HTML and CSS: Design and Build Websites (0.1451) 17. The Republic by Plato (0.0931)

В случае 2 можно заметить, что появление новой ключевой фразы "Java" в профиле приводит к изменению веса книги «Effective Java» без изменения её позиции, в то время как книги «You Don't Know JS» и «Eloquent JavaScript» перемещаются с 14 и 11 на 13 и 10 места соответственно.

В случае 3, с увеличением веса ключевой фразы "Java" до 0.5, можно заметить, что книга «Effective Java» поднимается на третье место, а книга «Eloquent JavaScript» перемещается с 13 на 11 место (рис. 4.10).

Вычислительный эксперимент, проведенный в рамках данного исследования, подтверждает эффективность разработанного пользовательского профиля и демонстрирует его практическую пользу в персонализации контента веб-сайта в соответствии с индивидуальными предпочтениями пользователей. Этот адаптивный подход может быть интегрирован в поисковые системы двумя способами: во-первых, путем реагирования на конкретные пользовательские запросы и уточнения результатов поиска для лучшего соответствия намерениям пользователя, и, во-вторых, путем работы в режиме, независимом от запросов, где сам профиль служит основой для организации и представления контента веб-сайта в персонализированной и контекстуально релевантной форме.

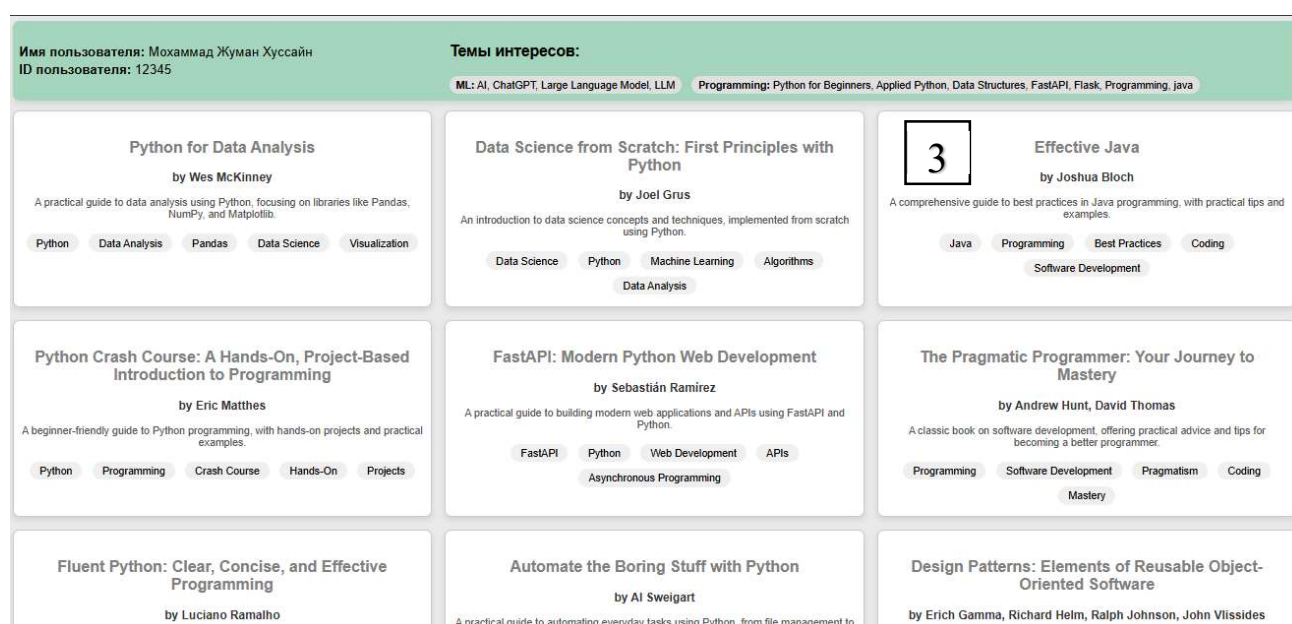


Рисунок 4.10 – Пример персонализации контента сайта для электронной библиотеки (случай 3)

В перспективе будущие исследования будут сосредоточены на совершенствовании технических возможностей данной функции. Ключевые

направления разработки включают внедрение динамических изменений разделов веб-сайта в реальном времени, что позволяет обеспечить более гибкий и отзывчивый пользовательский опыт. Кроме того, усилия будут направлены на создание автоматизированной системы генерации страниц, основанной на алгоритме персонализации, для формирования индивидуальных макетов контента, которые будут адаптироваться к изменяющимся предпочтениям и поведению пользователей. Эти улучшения направлены на дальнейшую оптимизацию расширения практического применения персонализированной доставки контента в цифровой среде.

4.6. Выводы по разделу

В данной главе представлены результаты вычислительных экспериментов, целью которых была проверка эффективности предложенных модели, метода и алгоритмов для персонализации поиска и контента веб-сайтов. Основное внимание уделялось повышению точности анализа текстового контента, с которым взаимодействовал пользователь, а также созданию онтологического профиля пользователя, который максимально точно отражает его интересы.

Проведенные эксперименты подтвердили эффективность разработанного алгоритма персонализации интересов пользователя, а также индивидуального вклада предложенных метода и алгоритмов обработки и анализа текстов. Эти результаты демонстрируют, что алгоритм персонализации успешно справляется с задачей адаптации контента под индивидуальные предпочтения пользователя даже в условиях семантической неоднозначности.

Разработанный модифицированный метод извлечения ключевых фраз продемонстрировал свою эффективность в двух ключевых аспектах. Во-первых, он существенно повышает эффективность персонализации при применении его для извлечения ключевых фраз на этапе создания профиля пользователя. Во-

вторых, он доказал свою высокую эффективность в задаче извлечения ключевых фраз автономно, вне контекста алгоритма персонализации.

Предложенный модифицированный алгоритм устранения неоднозначности смысла слов продемонстрировал высокую производительность, превзойдя оригинальный алгоритм *Леск* в различных сценариях. Хотя он не превзошел алгоритмы, построенные на основе графовых моделей, но показал весьма конкурентоспособные результаты, близкие к их уровню. Особенно стоит отметить, что алгоритм сохраняет простоту, характерную для алгоритма *Леск*, что делает его как эффективным, так и легким в реализации.

Результаты экспериментов подтвердили, что комбинированное применение модифицированного алгоритма устранения неоднозначности смысла слов и модифицированного метода извлечения ключевых фраз FBKE значительно повышает эффективность алгоритма персонализации результатов поиска веб-страниц. Устранение неоднозначности слов позволяет точнее интерпретировать запросы пользователей, учитывая контекст, в то время как извлечение ключевых фраз помогает выделить наиболее релевантные элементы контента. Это также обеспечивает более точное ранжирование и персонализацию поисковой выдачи, что в итоге повышает удовлетворенность пользователя при взаимодействии с системой. Данный результат подчеркивает важность интеграции различных методов и алгоритмов обработки и анализа текстов на естественном языке для достижения более качественных результатов в задачах поиска и персонализации контента веб-сайтов.

ЗАКЛЮЧЕНИЕ

Основной целью выполненной диссертационной работы являлось повышение эффективности моделей, методов и алгоритмов персонализации контента веб-сайтов. Под эффективностью понимается минимизация частоты ошибок функционирования систем персонализации, при условии снижения семантической неоднозначности.

Разработанные в рамках данного исследования модель, метод и алгоритмы демонстрируют перспективы развития методов искусственного интеллекта и машинного обучения и их интеграции в процессы веб-инженерии, что позволяет более точно определять пользовательские предпочтения при создании интеллектуальных систем обработки естественного языка.

Основными результатами проведенных исследований являются следующие:

1. Проведен аналитический обзор современных подходов к персонализации контента веб-сайтов на основе методов обработки и анализа текстов. Сформулирована постановка задачи исследования;

2. Разработан модифицированный метод извлечения ключевых фраз из одного документа, отличающийся применением новых правил для определения весов ключевых фраз на основе частоты их встречаемости и контекстуальной семантической близости, позволяющий извлекать наиболее релевантные ключевые фразы;

3. Разработан модифицированный алгоритм устранения неоднозначности смысла слов, отличающийся применением контекстных векторных представлений и расширенных аннотаций синсетов, что позволяет снизить процент ошибки при определении правильного значения неоднозначного слова;

4. Построена онтологическая модель для представления индивидуальных особенностей интересов пользователя, отличающаяся применением новой

функции расчета весовых коэффициентов, а также контекстно-зависимых отношений, что позволяет найти более релевантные темы;

5. Разработан алгоритм персонализированного поиска веб-сайтов, отличающийся применением весовых коэффициентов в формуле ранжирования документов, учитывающей иерархические уровни онтологического профиля пользователя, позволяющий снизить процент ошибки при решении задачи персонализации веб-контента;

6. Разработано программное приложение, реализующее разработанные модель, метод и алгоритмы обработки и анализа текстов на естественном языке в системах искусственного интеллекта и машинного обучения, позволяющее проводить сравнительный анализ предложенных алгоритмов с существующими аналогами;

7. Выполнены вычислительные эксперименты, которые подтвердили эффективность полученных решений, превзошедших результаты известных алгоритмов. В среднем предложенные автором решения снижают частоту ошибок функционирования систем персонализации на 5 – 8 %.

В целом, предложенные в работе модель, модифицированные метод и алгоритмы имеют важное значение для развития технологий искусственного интеллекта, получили высокую оценку научного сообщества при апробации и положительные рекомендации для внедрения в информационные процессы предприятий, учреждений и организаций различного профиля деятельности.

Достоверность научных результатов работы подтверждается непротиворечивостью и согласованностью с известными фактами и исследованиями в рассматриваемой области, высокой степенью сходимости теоретических результатов с данными экспериментов и определяется применением теоретических и методологических основ разработок ведущих ученых.

Теоретические и практические результаты работы внедрены в информационные процессы ИТ-компании ООО «Ит-Эффект» (г. Москва). Полученные в работе научные результаты позволили повысить эффективность решения задачи персонализации контента веб-сайтов в условиях семантической неоднозначности.

Во время выполнения работы зарегистрированы результаты интеллектуальной деятельности (два свидетельства о регистрации программ для ЭВМ). Результаты диссертации применяются в процессе подготовки студентов Института компьютерных технологий и информационной безопасности Южного федерального университета, что подтверждено актом об использовании в учебном процессе.

Таким образом, при выполнении диссертационного исследования решены все поставленные задачи, цель выполнения данной работы достигнута. Построена модель и разработаны метод и алгоритмы персонализации контента веб-сайтов при обработке и анализе текстов, которые направлены на решение актуальной научной задачи, связанной с необходимостью повышения эффективности *моделей, методов и алгоритмов* персонализации контента веб-сайтов. Под эффективностью понимается минимизация частоты ошибок функционирования систем персонализации, при условии снижения *семантической неоднозначности*.

Дальнейшее развитие полученных результатов диссертационного исследования может быть сосредоточено на объединении рекомендаций на основе контента, использующих разработанную онтологическую модель, с методами коллаборативной фильтрации или обучения с подкреплением для повышения как разнообразия, так и точности рекомендаций. Кроме того, оптимизация алгоритмов для распределенных вычислительных сред будет иметь ключевое значение для эффективной обработки крупномасштабных наборов данных и удовлетворения требований к обработке в реальном времени. Дальнейшее повышение

эффективности может быть достигнуто за счет изучения методов снижения вычислительной сложности, таких как использование приближенного поиска ближайших соседей или методов снижения размерности, особенно для приложений, работающих в реальном времени. Кроме того, онтологические отношения могут быть расширены для включения более сложных связей при сохранении вычислительной эффективности, что позволит обеспечить более богатое представление интересов пользователей и повысить общую производительность системы.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

ИИ – Искусственный интеллект

NLP – Natural language processing

WSD – Word Sense Disambiguation

TF-IDF – Term Frequency–Inverted Document Frequency;

NLTK – Natural Language Toolkit

BERT – Bidirectional encoder representations from transformers

S-BERT – Sentence BERT

ODP – Open Directory Project

WN-

Wordnet

СПИСОК ЛИТЕРАТУРЫ

1. Gorgoglione, M. Recommendation strategies in personalization applications / M. Gorgoglione, U. Panniello, A. Tuzhilin // Information & Management. – 2019. – Vol. 56. – No. 6. – P. 103143.
2. Tiihonen, J. An introduction to personalization and mass customization / J. Tiihonen, A. Felfernig // Journal of Intelligent Information Systems. – 2017. – Vol. 49. – No. 1. – P. 1-7.
3. Rossi, G. 25 years of model-driven web engineering. What we achieved, what is missing / G. Rossi, M. Urbiet, D. Distant, J.M. Rivero, S. Firmenich // CLEI Electronic Journal. – 2016. – Vol. 19. – No. 3. – P. 5-57.
4. Pressman, R. S. Web engineering / R.S. Pressman, D. Lowe // Software Engineering: A Practitioner's Perspective. – 2000. – P. 769-798.
5. Ibrahim, M.E. Ontology-based personalized course recommendation framework / M.E. Ibrahim, Y. Yang, D.L. Ndzi, G. Yang, M. Al-Maliki // IEEE Access. - 2018. - Vol. 7. - P. 5180-5199.
6. Guo, Q. AOL4PS: A large-scale data set for personalized search / Q. Guo, W. Chen, H. Wan // Data Intelligence. – 2021. – Vol. 3. – AOL4PS. – No. 4. – P. 548-567.
7. Farid, M. User profiling approaches, modeling, and personalization / M. Farid, R. Elgohary, I. Moawad, M. Roushdy // Proceedings of the 11th international conference on informatics & systems (INFOS 2018). – 2018.
8. Mobasher, B. Data mining for web personalization / B. Mobasher // The adaptive web. Springer. – 2007. – P. 90-135.
9. Gauch, S. User profiles for personalized information access / S. Gauch, M. Speretta, A. Chandramouli, A. Micarelli // The adaptive web. – 2007. – P. 54-89.
10. Grivokostopoulou, F. An ontology-based approach for user modelling and personalization in e-learning systems / F. Grivokostopoulou, I. Perikos, M. Paraskevas,

I. Hatzilygeroudis // 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS). - IEEE, 2019. - P. 1-6.

11. Leung, K. W.-T. Deriving concept-based user profiles from search engine logs / K.W.-T. Leung, D.L. Lee // IEEE Transactions on knowledge and data engineering. – 2009. – Vol. 22. – No. 7. – P. 969-982.

12. Liu, F. Personalized web search by mapping user queries to categories / F. Liu, C. Yu, W. Meng // Proceedings of the eleventh international conference on Information and knowledge management CIKM02: Eleventh ACM International Conference on Information and Knowledge Management. – McLean Virginia USA: ACM. – 2002. – P. 558-565.

13. Penas, P. Collective knowledge ontology user profiling for Twitter–automatic user profiling / P. Penas, R. Del Hoyo, J. Veja-Murguía, C. González, S. Mayo // 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). IEEE. – 2013. – Vol. 1. – P. 439-444.

14. Gauch, S. Ontology-Based Personalized Search and Browsing / S. Gauch, J. Chaffee, A. Pretschner // ACM Web Intelligence and Agent System. – 2003. – Vol. 1. – No. 3/4. – P. 219–234.

15. Xu, Y. Privacy-Enhancing Personalized Web Search / Y. Xu, K. Wang, B. Zhang, Z. Chen // Proceedings of World Wide Web (WWW)Conference, 2007.– P. 591–600.

16. Abián, D. Wikidata and DBpedia: A Comparative Study / D. Abián, F. Guerra, J. Martínez-Romanos, R. Trillo-Lado // Semantic Keyword-Based Search on Structured Data Sources: Lecture Notes in Computer Science / eds. J. Szymański, Y. Velegrakis. – Cham: Springer International Publishing. –2018. – Vol. 10546. – Wikidata and DBpedia. – P. 142-154. – ISBN 978-3-319-74496-4.

17. Lehmann, J. Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia / J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P.N.

Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer // Semantic web. – 2015. – Vol. 6. No. 2. – P. 167-195.

18. Kenter, T. Siamese cbow: Optimizing word embeddings for sentence representations / T. Kenter, A. Borisov, M. De Rijke // arXiv preprint arXiv:1604.04640. – 2016.

19. Pennington, J. Glove: Global vectors for word representation / J. Pennington, R. Socher, C.D. Manning // Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). – 2014. – P. 1532-1543.

20. Gontek, M. User modeling for exploratory search on the Social Web. Exploiting social bookmarking systems for user model extraction, evaluation and integration // Gontek Mirko; Universität zu Köln. – 2011.

21. Hawalah, A. Dynamic user profiles for web personalisation / A. Hawalah, M. Fasli// Expert Systems with Applications. – 2015. – Vol. 42. – No. 5. – P. 2547-2569.

22. Gauch, S. Ontology-Based User Profiles for Search and Browsing, User Modeling and User-Adapted Interaction: The Journal of Personalization Research / S. Gauch, J. Chaffee, A. Pretschner// Special Issue on User Modeling for Web Hypermedia Information Retrieval. – 2003.

23. Jurafsky, D. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition / D. Jurafsky, J. H. Martin. – 2000.

24. Manning, C. Foundations of statistical natural language processing / C. Manning, H. Schutze. – MIT press. – 1999.

25. Hearst, M. A. Text Tiling: Segmenting text into multi-paragraph subtopic passages / M.A. Hearst// Computational linguistics. – 1997. – Vol. 23. no 1. – P. 33-64.

26. Курейчик, В. М. Обработка информации на основе онтологий / В.М. Курейчик// Труды Конгресса по интеллектуальным системам и информационным технологиям" IS&IT'15". – 2015. – С. 62-75.
27. Курейчик, В. М. Онтологический подход к решению задач классификации документов / В.М. Курейчик, И.Б. Сафроненкова // Информационные технологии и математическое моделирование систем 2016. – 2016. – С. 32-35.
28. Panchenko, A. Russe: The first workshop on russian semantic similarity / A. Panchenko, N. Loukachevitch, D. Ustalov, D. Paperno, C. Meyer, N. Konstantinova// arXiv preprint arXiv:.05820. – 2018.
29. Игнатов, Д. И. Анализ формальных понятий: от теории к практике / Д.И. Игнатов, Р.Э. Яворский// Доклады всероссийской научной конференции АИСТ. – 2012. – Том. 12. – Р. 3-15.
30. Kutuzov, A. Word vectors, reuse, and replicability: Towards a community repository of large-text resources / A. Kutuzov, M. Fares, S. Oepen, E. Velldal// Proceedings of the 58th Conference on Simulation and Modelling. Linköping University Electronic Press. – 2017. – P. 271-276.
31. Ginige, A. Web Engineering: A Holistic, Disciplined Approach to Web-Based System Development / A. Ginige, S. Murugesan // 12th International World Wide Web Conference. – 2003. – Vol. 3. – P. [Web Engineering].
32. Eirinaki, M. Web mining for web personalization / M. Eirinaki, M. Vazirgiannis // ACM Transactions on Internet Technology. – 2003. – Vol. 3. – № 1. – P. 1–27.
33. Meister, F. “Getting to know you”: What’s new in personalization technologies / F. Meister, D. Shin, L. Andrews // E-Doc. – 2002. – Vol. 16. – № 2. – P. 8.

34. Bharambe, I. Design and implementation of search engine using vector space model for personalized search / I. Bharambe, R.K. Makhijani // *International Journal of Advanced Research in Computer Science Software Engineering*. – 2014. – Vol. 2277. – P. 1019–1023.
35. Purificato, E. User Modeling and User Profiling: A Comprehensive Survey / E. Purificato, L. Boratto, E.W. De Luca // *arXiv preprint arXiv:2402.09660*. – 2024.
36. Hjørland, B. The foundation of the concept of relevance / B. Hjørland // *J. Assoc. Inf. Sci. Technol.* – 2010. – Vol. 61. – P. 217–237.
37. Мохаммад, Ж. Релевантность контента как основной критерий ранжирования при поиске информации / Ж. Мохаммад, А. Мансур // *Труды XVIII всероссийской научной конференции молодых ученых, аспирантов и студентов «Информационные технологии, системный анализ и управление (ИТСАУ-2020)»*. – Таганрог. – 2020. – С. 118-121.
38. Сегалович, И. Как работают поисковые системы / И. Сегалович // *Мир Internet*. – 2002. – Vol. 10. – С. 24-32.
39. Маннинг, К. Д. Введение в информационный поиск / К.Д. Маннинг, П. Рагхаван, Х. Шютце. – . 2011.
40. Voorhees, E. M. Variations in relevance judgments and the measurement of retrieval effectiveness / E.M. Voorhees // *Information processing management*. – 2000. – Vol. 36. – No. 5. – P. 697-716.
41. Schamber, L. Relevance and information behavior / L. Schamber // *Annual review of information science technology*. – 1994. – Vol. 29. – P. 3-48.
42. Gelernter, J. Studies on relevance, ranking and results display / J. Gelernter, D. Cao, J. Carbonell // *arXiv preprint arXiv:.03705*. – 2010.
43. Westbrook, L. Faculty relevance criteria: Internalized user needs / L. Westbrook. –2001.

44. Eke, C. I. A survey of user profiling: State-of-the-art, challenges, and solutions / C.I. Eke, A.A. Norman, L. Shuib, H.F. Nweke // IEEE Access. – 2019. – Vol. 7. – P. 144907-144924.
45. Bird, S. NLTK: the natural language toolkit / S. Bird//Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions. – 2006. – P. 69-72.
46. Porter, M. F. An algorithm for suffix stripping / M.F. Porter//Program. – 1980. – Vol. 14. no 3. – P. 130-137.
47. Willett, P. The Porter stemming algorithm: then and now / P. Willett//Program. – 2006. – Vol. 40. no 3. – P. 219-223.
48. Balakrishnan, V. Stemming and lemmatization: A comparison of retrieval performances / V. Balakrishnan, E. Lloyd-Yemoh. – 2014.
49. Miller, G. A. WordNet: a lexical database for English / C.A. Miller. – 1995. – Vol. 38. – P. 39-41.
50. Azarova, I. Russnet: Building a lexical database for the russian language / I. Azarova, O. Mitrofanova, A. Sinopalnikova, M. Yavorskaya, I. Oparin // Proceedings of the Workshop on Wordnet Structures and Standardisation and How this affect Wordnet Applications and Evaluation. – 2002.
51. Lops, P. Content-based recommender systems: State of the art and trends / P. Lops, M. De Gemmis, G. Semeraro // Recommender systems handbook. – 2011. – P. 73-105.
52. Poelmans, J. Formal concept analysis in knowledge processing: A survey on applications / J. Poelmans, D.I. Ignatov, S.O. Kuznetsov, G. Dedene // Expert Systems with Applications. – 2013. – Vol. 40. – No. 16. – P. 6538–6560.
53. Poelmans, J. Text mining scientific papers: a survey on FCA-based information retrieval research / J. Poelmans, D.I. Ignatov, S. Viaene, G. Dedene, S.O. Kuznetsov // Advances in Data Mining. Applications and Theoretical Aspects: 12th

Industrial Conference, ICDM 2012, Berlin, Germany, July 13–20, 2012. Proceedings 12. – Berlin: Springer. – 2012. – P. 273–287.

54. Gruber, T. What is an Ontology / T. Gruber. – 1993.

55. Hernich, A. Schema. org as a description logic / A. Hernich, C. Lutz, A. Ozaki, F. Wolter // IJCAI'15 Proceedings of the 24th International Conference on Artificial Intelligence. AAAI Press. – 2015. – P. 3048-3054.

56. Cantador, I. Ontology-based personalised and context-aware recommendations of news items / I. Cantador, A. Bellogín, P. Castells. – Text : direct // 2008 IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology. IEEE, 2008. Vol. 1. – P. 562-565.

57. Abubakar, H. D. Sentiment classification: Review of text vectorization methods: Bag of words, Tf-Idf, Word2vec and Doc2vec / H.D. Abubakar, M. Umar, M.A. Bakale // SLU Journal of Science Technology. – 2022. – Vol. 4. – no 1 & 2. – P. 27-33.

58. Goossen, F. News personalization using the CF-IDF semantic recommender / F. Goossen, W. IJntema, F. Frasincar, F. Hogenboom, U. Kaymak // Proceedings of the International Conference on Web Intelligence, Mining and Semantics. – 2011. – P. 1-12.

59. Nada, Y. A. An approach to improve the representation of the user model in the web-based systems / Y.A. Nada, K.M. Fouad // International Journal of Advanced Computer Science and Applications. – 2011. – Vol. 2. – No. 12.

60. Hu, J. A user profile modeling method based on Word2Vec / J. Hu, F. Jin, G. Zhang, J. Wang, Y. Yang // 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE. – 2017. – P. 410-414.

61. Daoud, M. A personalized graph-based document ranking model using a semantic user profile / M. Daoud, L. Tamine, M. Boughanem // User Modeling,

Adaptation, and Personalization: 18th International Conference, UMAP 2010, Big Island, HI, USA, June 20-24, 2010. Proceedings 18. Springer. – 2010. – P. 171-182.

62. Daoud, M. A session based personalized search using an ontological user profile / M. Daoud, L. Tamine-Lechani, M. Boughanem, B. Chebaro // Proceedings of the 2009 ACM symposium on Applied Computing. – 2009. – P. 1732-1736.

63. Maedche, A. Ontology learning part one—on discovering taxonomic relations from the web / A. Maedche, V. Pekar, S. Staab // Web Intelligence. Springer. – 2003. – P. 301-319.

64. Liang, H. Personalized recommender system based on item taxonomy and folksonomy / H. Liang, Y. Xu, Y. Li, R. Nayak // Proceedings of the 19th ACM international conference on Information and knowledge management. – 2010. – P. 1641-1644.

65. Городецкий, В. И. Онтологии и персонификация профиля пользователя в рекомендующих системах третьего поколения / В.И. Городецкий, О.Н. Тушканова // Онтология проектирования. – 2014. – № 3 (13). – P. 7-31.

66. Han, K. Adaptive and multiple interest-aware user profiles for personalized search in folksonomy: A simple but effective graph-based profiling model / K. Han, J. Park, M.Y. Yi // 2015 International Conference on Big Data and Smart Computing (BIGCOMP). IEEE. – 2015. – P. 225-231.

67. Liu, X. Modeling Users' Dynamic Preference for Personalized Recommendation. / X. Liu // IJCAI. – 2015. – P. 1785-1791.

68. Sieg, A. Web search personalization with ontological user profiles / A. Sieg, B. Mobasher, R. Burke // Proceedings of the sixteenth ACM conference on Conference on information and knowledge management CIKM07: Conference on Information and Knowledge Management. – Lisbon Portugal: ACM. – 2007. – P. 525-534.

69. Skillen, K.-L. Ontological User Profile Modeling for Context-Aware Application Personalization / K.-L. Skillen, L. Chen, C.D. Nugent, M.P. Donnelly, W. Burns, I. Solheim // *Ubiquitous Computing and Ambient Intelligence: Lecture Notes in Computer Science* / coll. D. Hutchison [et al.]; eds. J. Bravo, D. López-de-Ipiña, F. Moya. – Berlin, Heidelberg: Springer Berlin Heidelberg. – 2012. – Vol. 7656. – P. 261-268. – ISBN 978-3-642-35376-5.
70. Sosnovsky, S. Ontological technologies for user modelling / S. Sosnovsky, D. Dicheva // *International Journal of Metadata, Semantics and Ontologies*. – 2010. – Vol. 5. – No. 1. – P. 32.
71. Schutz, A. T. Keyphrase extraction from single documents in the open domain exploiting linguistic and statistical methods / A.T. Schutz // *M. App. Sc Thesis*. – 2008.
72. Hammouda, K. M. Corephrase: Keyphrase extraction for document clustering / K.M. Hammouda, D.N. Matute, M.S. Kamel // *International workshop on machine learning and data mining in pattern recognition*. Springer. – 2005. – P. 265-274.
73. Karanikolas, N. N. Key-phrase extraction for classification / N.N. Karanikolas, C. Skourlas // *MEDICON and HEALTH TELEMATICS 2004: X Mediterranean Conference on Medical and Biological Engineering*. – 2004.
74. Wu, Y. B. Domain-specific keyphrase extraction / Y.B. Wu, Q. Li, R.S. Bot, X. Chen // *Proceedings of the 14th ACM international conference on Information and knowledge management*. – 2005. – P. 283-284.
75. Song, I. Y. Keyphrase extraction-based query expansion in digital libraries / I.Y. Song, R.B. Allen, Z. Obradovic, M. Song // *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'06)*. IEEE. – 2006. – P. 202-209.
76. Mihalcea, R. Corpus-based and knowledge-based measures of text semantic similarity / R. Mihalcea, C. Corley, C. Strapparava // *Aaai*. – 2006. – Vol. 6. – P. 775-780.

77. Beliga, S. An overview of graph-based keyword extraction methods and approaches / S. Beliga, A. Meštrović, S. Martinčić-Ipšić // *Journal of information organizational sciences*. – 2015. – Vol. 39. – No. 1. – P. 1-20.
78. Hasan, K. S. Automatic keyphrase extraction: A survey of the state of the art / K.S. Hasan, V. Ng // *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. – 2014. – P. 1262-1273.
79. Nguyen, T. D. WINGNUS: Keyphrase extraction utilizing document logical structure / T.D. Nguyen, M.-T. Luong // *Proceedings of the 5th international workshop on semantic evaluation*. – 2010. – P. 166-169.
80. Witten, I. H. Kea: Practical automated keyphrase extraction / I.H. Witten, G.W. Paynter, E. Frank, C. Gutwin, C.G. Nevill-Manning // *Design and Usability of Digital Libraries: Case Studies in the Asia Pacific*. IGI global. – 2005. – P. 129-152.
81. Meng, R. Deep keyphrase generation / R. Meng, S. Zhao, S. Han, D. He, P. Brusilovsky, Y. Chi // *arXiv preprint arXiv:1706.06879*. – 2017.
82. Sharma, P. Self-Supervised Contextual Keyword and Keyphrase Retrieval with Self-Labeling / P. Sharma, Y. Li. – 2019.
83. Jones, K. S. A statistical interpretation of term specificity and its application in retrieval / K.S. Jones // *Journal of documentation*. – 1972.
84. Campos, R. YAKE! Keyword extraction from single documents using multiple local features / R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, A. Jatowt//*Information Sciences*. – 2020. – Vol. 509. – P. 257-289.
85. El-Beltagy, S. R. KP-Miner: A keyphrase extraction system for English and Arabic documents / S.R. El-Beltagy, A. Rafea // *Information Systems*. – 2009. – Vol. 34. – No. 1. – P. 132-144.
86. Rose, S. Automatic keyword extraction from individual documents / S. Rose, D. Engel, N. Cramer, W. Cowley//*Text mining: applications theory*. – 2010. – Vol. 1. – P. 1-20.

87. Mihalcea, R. Textrank: Bringing order into text / R. Mihalcea, P. Tarau // Proceedings of the 2004 conference on empirical methods in natural language processing. – 2004. – P. 404-411.
88. Das, B. Automatic keyword extraction from any text document using N-gram rigid collocation / B. Das, S. Pal, S.K. Mondal, D. Dalui, S.K. Shome // Int. J. Soft Comput. Eng. (IJSCE). – 2013. – Vol. 3. – No. 2. – P. 238-242.
89. Wan, X. Single Document Keyphrase Extraction Using Neighborhood Knowledge / X. Wan, J. Xiao // AAAI. – 2008. – Vol. 8. – P. 855-860.
90. Hassan, H. A. M. Bert, elmo, use and infersent sentence encoders: The panacea for research-paper recommendation? / H.A.M. Hassan, G. Sansonetti, F. Gasparetti, A. Micarelli, J. Beel // RecSys (Late-Breaking Results). – 2019. – P. 6-10.
91. Reimers, N. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks / N. Reimers, I. Gurevych // arXiv preprint arXiv:. – 2019. – Vol. abs/1908.10084.
92. Khan, M. Q. Impact analysis of keyword extraction using contextual word embedding / M.Q. Khan, A. Shahid, M.I. Uddin, M. Roman, A. Alharbi, W. Alosaimi, J. Almalki, S.M. Alshahrani // PeerJ Computer Science. – 2022. – Vol. 8. – P. e967.
93. Qian, Y. BERT-based text keyword extraction / Y. Qian, C. Jia, Y. Liu // Journal of Physics: Conference Series. IOP Publishing. – 2021. – Vol. 1992. – P. 042077.
94. Giarelis, N. A Comparative Assessment of State-Of-The-Art Methods for Multilingual Unsupervised Keyphrase Extraction / N. Giarelis, N. Kanakaris, N. Karacapilidis // IFIP International Conference on Artificial Intelligence Applications and Innovations. Springer. – 2021. – P. 635-645.
95. Grootendorst, M. Keybert: Minimal keyword extraction with bert / M. Grootendorst. – 2020.

96. Das Dawn D. et al. A dataset for evaluating Bengali word sense disambiguation techniques //Journal of Ambient Intelligence and Humanized Computing. – 2023. – Vol. 14. – No. 4. – P. 4057-4086.
97. Raganato, A. Word sense disambiguation: A unified evaluation framework and empirical comparison / A. Raganato, J. Camacho-Collados, R. Navigli // Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers. – 2017. – P. 99-110.
98. Pradhan, S. Semeval-2007 task-17: English lexical sample, srl and all words / S. Pradhan, E. Loper, D. Dligach, M. Palmer // Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007). – 2007. – P. 87-92.
99. Lesk, M. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone / M. Lesk // Proceedings of the 5th annual international conference on Systems documentation. – 1986. – P. 24-26.
100. Banerjee, S. Extended gloss overlaps as a measure of semantic relatedness / S. Banerjee, T. Pedersen // Ijcai. Citeseer. – 2003. – Vol. 3. – P. 805-810.
101. Vasilescu, F. Evaluating Variants of the Lesk Approach for Disambiguating Words / F. Vasilescu, P. Langlais, G. Lapalme // Lrec. – 2004.
102. Banerjee, S. An adapted Lesk algorithm for word sense disambiguation using WordNet / S. Banerjee, T. Pedersen // International conference on intelligent text processing and computational linguistics. Springer. – 2002. – P. 136-145.
103. Basile, P. An enhanced lesk word sense disambiguation algorithm through a distributional semantic model / P. Basile, A. Caputo, G. Semeraro // Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. – 2014. – P. 1591-1600.
104. Мохаммад Ж. Х., Метод извлечения ключевых фраз на основе новой функции ранжирования / Ж.Х. Мохаммад, А.М. Мансур, Ю.А. Кравченко, В.В. Бова // Информационные технологии. – 2022. – Том. 28. № 9. – С. 465-474.

105. Мохаммад Ж. Х. Метод автоматического извлечения ключевых слов / Ж.Х. Мохаммад, А.М. Мансур, Д.Ю. Кравченко, Ю.А. Кравченко // Труды международного научно-технического конгресса «Интеллектуальные системы и информационные технологии – 2022» («ИС & ИТ-2022», «IS&IT'22»). Научное издание. – Таганрог: Изд-во Ступина С.А., Т.1. – 2022. – С. 90-97.

106. Мохаммад, Ж. Х. Извлечение ключевых фраз на основе больших языковых моделей / Ж.Х. Мохаммад // Известия ЮФУ. Технические науки. – 2024. – № 5.

107. Mohammad, J. Algorithm for Optimization of Keyword Extraction Based on the Application of a Linguistic Parser / A. Mansour, J. Mohammad, D. Kravchenko, Y. Kravchenko & N. Pavlov // Informatics and Automation. – Vol. 23. – 2024. – No. 2. – P. 467-494.

108. Мохаммад Ж. Х. Перспективы применения метода извлечения ключевых фраз FBKE в задачах персонализации веб-контента / Ж.Х. Мохаммад, А.М. Мансур, Ю.А. Кравченко// Труды XX всероссийской научной конференции молодых ученых, аспирантов и студентов «Информационные технологии, системный анализ и управление (ИТСАУ-2022)». – Таганрог. – 2022. – С. 206.

109. Мохаммад Ж. Х. Модифицированный метод устранения неоднозначности смысла слов, основанный на методах распределенного представления / Ж.Х. Мохаммад, А.М. Мансур, Ю.А. Кравченко //Известия Южного федерального университета. Технические науки. – 2021. № 3 (220). – С. 92-101.

110. Мохаммад Ж. Х. Модифицированный метод устранения семантической неоднозначности слов / Ж.Х. Мохаммад, А.М. Мансур // Труды XIX всероссийской научной конференции молодых ученых, аспирантов и студентов «Информационные технологии, системный анализ и управление» (ИТСАУ-2021). – 2021. – С. 204-207.

111. Iacobacci, I. Embeddings for word sense disambiguation: An evaluation study / I. Iacobacci, M.T. Pilehvar, R. Navigli // *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. – 2016. – Vol. 1. – P. 897-907.

112. Мохаммад Ж. Перспективы развития процессов автоматического построения онтологий на основе применения методов оценки семантической близости/ Ж. Мохаммад, А. Мансур // *Труды XVIII всероссийской научной конференции молодых ученых, аспирантов и студентов «Информационные технологии, системный анализ и управление (ИТСАУ-2020)*. – Таганрог. – 2020. – С. 121-124.

113. Мохаммад Ж. Использование методов веб-майнинга в автоматическом построении онтологий/ Ж. Мохаммад А. Мансур // *Труды VI всероссийской научно-технической конференции молодых ученых, аспирантов, магистрантов и студентов «Фундаментальные и прикладные аспекты компьютерных технологий и информационной безопасности»*. – Таганрог. – 2020. – С. 321-324.

114. Мохаммад Ж. Х. Онтологическая модель для персонализации контента веб-сайтов на основе методов обработки и анализа текстов / Ж.Х. Мохаммад // *Труды международного научно-технического конгресса «Интеллектуальные системы и информационные технологии – 2025» («ИС & ИТ-2025», «IS&IT'22»)*. Научное издание. – Таганрог: Изд-во Ступина С.А., Т.1. – 2025. – С. 324-329.

115. Мохаммад, Ж. Х. Методика построения и оценки онтологического профиля для систем персонализации контента / Ж.Х. Мохаммад // *Известия ЮФУ. Технические науки*. – 2025. – № 6 (248).

116. Mohammad, J. Automated Ontology Extraction from Text for Content-Based Web Personalization/ J. Mohammad, A. Mansour, Y. Kravchenko // *Lecture Notes in Computer Science*. – LNCS. – Vol. 16328. – 2026. – P. 139-151.

117. Мохаммад, Ж. Перспективы развития процессов информационного поиска на основе применения методов data mining / Ж. Мохаммад, А. Мансур // Труды VI всероссийской научно-технической конференции молодых ученых, аспирантов, магистрантов и студентов «Фундаментальные и прикладные аспекты компьютерных технологий и информационной безопасности». – Таганрог. – 2020. – С. 317-320.

118. Мохаммад Ж. Программный модуль оптимизации извлечения ключевых слов при обработке лингвистической экспертной информации / Ж. Мохаммад, А. Мансур, Ю.А. Кравченко, К. Н. Владимировна // Свидетельство регистрации программы для ЭВМ. – 14.12.2023. – № 2023687372.

119. Мохаммад Ж. Программный модуль оптимизации работы классификатора при векторизации текста на основе биоэвристик / Ж. Мохаммад, А. Мансур, Ю.А. Кравченко, Д. Ю. Кравченко // Свидетельство регистрации программы для ЭВМ. – 12.12.2023. – № 2023687185.

120. Voron, F. Building Data Science Applications with FastAPI: Develop, manage, and deploy efficient machine learning applications with Python. Building Data Science Applications with FastAPI // F. Voron // Packt Publishing Ltd, – 2023.

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

122. Flanagan, D. JavaScript / D. Flanagan, P. Matilainen // Anaya Multimedia, – 2007. p.– ISBN 84-415-2202-2.

123. Somasundar, A. MongoDB integration with Python and Node. js, Express. js / A. Somasundar, M. Chilakaraao, B.R.K. Raju, S.K. Behera, C.V. Ramana, P.K. Sethy// 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT). IEEE, – 2024. – P. 1-5.

124. Kim, S. N. Automatic keyphrase extraction from scientific articles / S.N. Kim, O. Medelyan, M.-Y. Kan, T. Baldwin // *Language resources evaluation*. – 2013. – Vol. 47. – No. 3. – P. 723-742.
125. Medelyan, O. Domain-independent automatic keyphrase indexing with small training sets / O. Medelyan, I.H. Witten // *Journal of the American Society for Information Science Technology*. – 2008. – Vol. 59. – No. 7. – P. 1026-1040.
126. Medelyan, O. Topic indexing with Wikipedia / O. Medelyan, I.H. Witten, D. Milne // *Proceedings of the AAAI WikiAI workshop*. – 2008. – Vol. 1. – P. 19-24.
127. Мохаммад Ж. Х. Разработка чат-бота для классификации и анализа текстов на естественном языке с использованием локальных больших языковых моделей / Ж.Х. Мохаммад, А.М. Мансур, Ю.А. Кравченко // *Известия ЮФУ. Технические науки*. – 2025. – № 3 (2025). – С. 159-171.
128. Mohammad, J. Harnessing Key Phrases in Constructing a Concept-Based Semantic Representation of Text Using Clustering Techniques / J. Mohammad, A. Mansour, Y. Kravchenko, D. Kravchenko, N. Silega // *Lecture Notes in Computer Science*. – LNCS. – Vol. 14335. – 2023. – P. 190-201.
129. Mohammad, J. Generating Conceptual Semantic Vectors Based on Key Phrase Extraction Techniques / J. Mohammad, A. Mansour, Y. Kravchenko // *2023 International Russian Automation Conference (RusAutoCon)*. IEEE. – 2023. – С. 374-379.
130. Mohammad, J. Text vectorization method based on concept mining using clustering techniques / J. Mohammad, A. Mansour, Y. Kravchenko // *2022 VI International Conference on Information Technologies in Engineering Education (Inforino)*. IEEE. – 2022. – P. 1-10. (Scopus)
131. Мохаммад, Ж. Х. Векторизация текста с использованием методов интеллектуального анализа данных / Ж.Х. Мохаммад, А.М. Мансур, Ю.А.

Кравченко // Известия ЮФУ. Технические науки. – 2021. – № 2 (219). – С. 154-167.

132. Haveliwala, T. H. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search / T.H. Haveliwala // IEEE transactions on knowledge data engineering. –2003. – Vol. 15. no 4. – P. 784-796.

133. Moro, A. Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking / A. Moro, R. Navigli // Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015). – 2015. – P. 288-297.

134. Navigli, R. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network / R. Navigli, S.P. Ponzetto // Artificial intelligence. – 2012. – Vol. 193. – P. 217-250.

135. Kilgarriff, A. Framework and results for English SENSEVAL / A. Kilgarriff, J. Rosenzweig // Computers the Humanities. – 2000. Vol. 34. – No. 1. – P. 15-48.

136. Edmonds, P. Senseval-2: overview / P. Edmonds, S. Cotton // Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems. – 2001. – P. 1-5.

137. Snyder, B. The English all-words task / B. Snyder, M. Palmer // Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text. – 2004. – P. 41-43.

138. Navigli, R. Semeval-2013 task 12: Multilingual word sense disambiguation / R. Navigli, D. Jurgens, D. Vannella // Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). – 2013. – P. 222-231.

139. Agirre, E. Random walks for knowledge-based word sense disambiguation / E. Agirre, O. Lopez de Lacalle, A. Soria // Computational linguistics. – 2014. – Vol. 40. – No. 1. – P. 57-84.

ПРИЛОЖЕНИЕ №1

АКТЫ О ВНЕДРЕНИИ РЕЗУЛЬТАТОВ РАБОТЫ

**ОБЩЕСТВО С ОГРАНИЧЕННОЙ
ОТВЕТСТВЕННОСТЬЮ ИТ-ЭФФЕКТ**

ИНН 7729594500, КПП 77290100, ОГРН 1087746189893
119517, Москва, Нежинская ул., д.8, корп.2, пом.6
(985) 997-5591

УТВЕРЖДАЮ
Генеральный директор
ООО «ИТ-ЭФФЕКТ»



С.А. Сафонов

05 мая 2025 г.

АКТ

о внедрении результатов диссертационной работы на соискание ученой степени кандидата технических наук Мохаммад Жуман Хуссайн в обществе с ограниченной ответственностью «ИТ-ЭФФЕКТ» (ООО «ИТ-ЭФФЕКТ»)

Комиссия в составе:

председатель комиссии

- *генеральный директор С.А. Сафонов;*

члены комиссии:

- Технический директор Ю.А.Ермилов;

- Начальник отдела маркетинга А.И.Бурдаев.

составили настоящий акт о том, что научные результаты диссертационной работы аспиранта Южного федерального университета Ж.Х. Мохаммад по теме «Модель, метод и алгоритмы веб-инженерии для интеллектуальной персонализации контента веб-сайтов», представленной на соискание ученой степени кандидата технических наук, использованы в ООО «ИТ-ЭФФЕКТ» при построении системы семантического поиска на основе правил, использующей комплекс методов и алгоритмов обработки и анализа текстов на естественном языке для автоматического извлечения ключевых фраз и устранения их неоднозначности, а также автоматического построения онтологий и графов знаний.

В частности, были использованы следующие конкретные научные результаты кандидатской диссертации Ж.Х. Мохаммад:

- Онтологическая модель, основанная на динамической графовой структуры знаний для представления профилей пользователей и улучшения персонализации результатов поиска;
- Алгоритм извлечения ключевых фраз, основанный на использовании функций контекстуальной семантической близости и шаблонов частей речи парсера;
- Алгоритм устранения лексической неоднозначности слов с использованием мер контекстуальной семантической близости.

Внедренные результаты диссертационного исследования позволили повысить

качество и унификацию проектных решений.

Председатель комиссии:
Генеральный директор



С.А. Сафонов

Члены комиссии:
Технический директор
Начальник отдела маркетинга




Ю.А.Ермилов;
А.И.Бурдаев.



АКТ

об использовании в учебном процессе
Института компьютерных технологий и информационной безопасности
Южного федерального университета
результатов кандидатской диссертации Ж.Х. Мохаммад «Модель, метод и алгоритмы веб-инженерии для интеллектуальной персонализации контента веб-сайтов»

Я, нижеподписавшийся, руководитель образовательной программы (ОП) «Разработка информационных систем и web-приложений» 09.04.01 Информатика и вычислительная техника Кулиев Э.В. к.т.н., доцент кафедры САПР имени В.М. Курейчика, составил настоящий акт о том, что в учебном процессе кафедры САПР имени В.М. Курейчика, Института компьютерных технологий и информационной безопасности ЮФУ используются следующие результаты, полученные в кандидатской диссертации Мохаммад Ж.Х.:

- Онтологическая модель, отличающаяся введением новых компонентов, включающих множество динамических контекстно-зависимых отношений, извлекаемых из текстовых данных, для определения уровня интереса пользователя в системах персонализации контента веб-сайтов;
- Модифицированный метод извлечения ключевых фраз, основанный на использовании функций контекстуальной семантической близости и применением новых правил для определения весов ключевых фраз;
- Модифицированный алгоритм устранения неоднозначности смысла слов с использованием мер контекстуальной семантической близости.
- Программное приложение для решения задач персонализации контента веб-сайтов с использованием инструментов обработки и анализа текстов.

Указанные результаты используются при проведении следующих курсов в ИКТИБ: «Онтологические модели в информационных системах», «Технологии Big Data», «Методы машинного обучения при построении информационных систем».

Внедрение в учебный процесс ряда теоретических и практических результатов диссертационной работы Мохаммад Ж.Х. позволило повысить качество подготовки магистров.

Руководитель ОП «Разработка информационных систем
и web-приложений»,
к.т.н., доцент

Э. В. Кулиев

ПРИЛОЖЕНИЕ №2

**СВИДЕТЕЛЬСТВА О ГОСУДАРСТВЕННОЙ РЕГИСТРАЦИИ ПРОГРАММ
ДЛЯ ЭВМ**

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2023687372

**Программный модуль оптимизации извлечения
ключевых слов при обработке лингвистической
экспертной информации**

Правообладатель: *федеральное государственное автономное
образовательное учреждение высшего образования
«Южный федеральный университет» (RU)*

Авторы: *Кравченко Юрий Алексеевич (RU), Мохаммад
Жуман (AB), Мансур Али (RU), Кулиева Нина
Владимировна (RU)*



Заявка № 2023686104

Дата поступления 30 ноября 2023 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 14 декабря 2023 г.

*Руководитель Федеральной службы
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 429b6a0fe3853164baf96f83b73b4aa7
Владелец **Зубов Юрий Сергеевич**
Действителен с 10.05.2023 по 02.08.2024

Ю.С. Зубов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2023687185

**Программный модуль оптимизации работы
классификатора при векторизации текста на основе
биоэвристик**

Правообладатель: *федеральное государственное автономное
образовательное учреждение высшего образования
«Южный федеральный университет» (RU)*

Авторы: *Кравченко Даниил Юрьевич (RU), Кравченко Юрий
Алексеевич (RU), Мансур Али (RU), Мохаммад Жуман
(RU)*



Заявка № 2023686057

Дата поступления 30 ноября 2023 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 12 декабря 2023 г.

*Руководитель Федеральной службы
по интеллектуальной собственности*

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 429b6a0fe3853164baf96f63b73b4aa7
Владелец **Зубов Юрий Сергеевич**
Действителен с 10.05.2023 по 02.08.2024

Ю.С. Зубов