УДК 004

СЕМАНТИЧЕСКИЙ АНАЛИЗ КОРПУСА НАУЧНЫХ СТАТЕЙ НА ОСНОВЕ ГРАФОВОГО ПРЕДСТАВЛЕНИЯ

В. А. Чунихин[1] [0009-0007-0845-1374], С. А. Зайцев[2] [0000-0002-7902-9695], О. М. Атаева[3] [0000-0003-0367-5575]

[1–3]*Московский университет имени С. Ю. Витте, г. Москва, Россия*

[3]*Федеральный исследовательский центр «Информатика и управление» РАН, г. Москва, Россия*

[1]supermen.tut.103103@gmail.com, [2]szaytsev@muiv.ru, [3]OAtaeva@frccsc.ru

Аннотация

Проблема эффективной навигации и поиска релевантной информации в постоянно растущем объеме научных публикаций требует перехода от классических методов полнотекстового поиска к семантическим моделям. В работе предложен подход к структурированию гетерогенного корпуса научных текстов путем построения графа знаний. Разработан конвейер обработки данных, включающий извлечение метаданных, ключевых слов и структурных элементов статей, а затем их интеграцию в единый граф. На основе построенного графа знаний реализованы методы анализа явных и извлечения неявных связей между публикациями. Результаты исследования демонстрируют эффективность графового представления научной информации для выявления скрытых закономерностей в предметных областях и поддержки интеллектуальной навигации.

Ключевые слова*: семантический анализ, корпус научных статей, граф знаний, онтология, RDF, SPARQL, большие языковые модели, извлечение информации, графовые базы данных.*

ВВЕДЕНИЕ

Рост объема научных публикаций и их предметного разнообразия существенно усложняет поиск, анализ и систематизацию научной информации. Традиционные методы полнотекстового поиска позволяют находить документы по совпадению ключевых слов, однако недостаточно учитывают смысловые связи между публикациями, авторами, научными направлениями и используемыми

понятиями. В связи с этим все большее распространение получают методы семантического анализа, онтологического моделирования и графового представления знаний, обеспечивающие более содержательную обработку научных текстов и поддержку интеллектуального поиска.

В области семантического анализа научных текстов можно выделить ряд исследований, близких по постановке задач и используемым подходам. В работе [1] представлен практический опыт формирования корпуса русскоязычных научных публикаций и его применения для извлечения сущностей и отношений. Авторы отмечают дефицит открытых размеченных данных и показывают, что дообучение моделей позволяет существенно повысить качество извлечения информации даже на относительно небольших выборках. Кроме того, продемонстрировано, что предварительное структурирование научных текстов упрощает последующие этапы семантического анализа и автоматического извлечения знаний.

В работе [2] семантический анализ и онтологическое моделирование рассматриваются как средства интеграции разнородных источников данных. Использование онтологий обеспечивает единое описание предметной области, позволяет формализовать типы сущностей и отношений между ними, а также выполнять поиск на основе смысловых связей вместо простого совпадения текстовых фрагментов.

Несмотря на значительные результаты существующих исследований, вопросы интеграции извлеченной семантической информации в единую графовую модель научных публикаций и использования больших языковых моделей (БЯМ) для интеллектуального взаимодействия с графом знаний (ГЗ) остаются недостаточно проработанными.

Настоящая работа направлена на решение данной задачи. Предлагаемый подход объединяет методы семантического анализа, технологии ГЗ и БЯМ для построения графовой базы научных публикаций, автоматизации извлечения информации и поддержки выполнения запросов на естественном языке посредством их преобразования в SPARQL-запросы. Все это позволяет анализировать как явные, так и неявные связи между объектами научного корпуса и расширяет возможности интеллектуальной навигации по научным данным.

СЕМАНТИЧЕСКИЙ АНАЛИЗ КОРПУСА НАУЧНЫХ СТАТЕЙ

Семантический анализ реализован в виде процесса (рис. 1): сбор статистики по корпусу, извлечение текстового содержимого и семантических характеристик, построение онтологической модели, наполнение ее экземплярами, формирование графа знаний и выполнение запросов на естественном языке.

Каждый этап выполнялся последовательно и фиксировал промежуточный результат в виде структурированных записей. Это позволило отделить извлечение текста и семантических характеристик от последующего построения онтологии и загрузки данных в GraphDB, а также повторять отдельные шаги без повторной обработки всего корпуса.

На этапе первичного анализа корпуса было принято решение обрабатывать PDF-тексты, так как этот формат доминирует в данных. Было отобрано 33396 документов, собранных в рамках библиотеки LibMeta [3]. Корпус включает научные математические статьи 1958–2024 гг., оформленные в различных стилях, что потребовало унификации входного формата. Неоднородность данных проявлялась в структуре документов (одна или две колонки текста, различная компоновка заголовка и списка авторов, отличия в оформлении ссылок и идентификаторов), а также в способе представления текста в PDF. Для части файлов текстовый слой извлекается корректно, для части — содержит искажения, связанные со шрифтами и кодировками. Поэтому дальнейшая обработка строилась на едином варианте входа и наборе проверок качества извлеченного текста.

Рис. 1. Процесс обработки документов.

Представленный процесс отражает общий порядок работ: от оценки состава корпуса до получения графового представления и выполнения запросов. При этом для части документов возможен «возврат» на шаг контроля качества: если после извлечения текста выявлялись искажения или неполнота ключевых полей, такие документы исключались из набора, используемого при построении графа знаний.

На следующем этапе производится извлечение семантических данных из текстов документов [4, 5]. На этапе извлечения для каждого PDF-файла выделялся текст, который ограничивался первыми 15000 символами для предотвращения переполнения контекстного окна языковых моделей при сохранении достаточности данных для анализа. При увеличении фрагмента текста возрастала доля ответов, в которых модель переставала следовать заданному формату и возвращала данные с нарушением структуры. При выбранном ограничении сохранялись элементы, необходимые для извлечения основных характеристик (это название, авторы, год, издательство, тематическое описание), и одновременно снижалась доля ответов с нарушением формата. Далее сформировался структурированный набор характеристик: тип документа, важность (субъективная оценка БЯМ), название, авторы, год, издательство (русскоязычное и англоязычное названия), краткое тематическое описание, предметная область, ключевые слова, язык и идентификационные коды (при наличии). Извлеченные характеристики сохранялись в структурированном виде по фиксированной схеме, что упрощало последующую проверку заполненности полей и загрузку в онтологическую модель. Для полей со множественными значениями (авторы, ключевые слова, идентификаторы) сохранялись списки значений, а для полей-меток (предметная область, язык, тип документа) — одно значение, согласованное по формату.

Из-за временных ограничений семантические данные были получены для 8114 файлов. Обработка выполнялась последовательно для каждого файла и включала извлечение текста, формирование запроса к модели и проверку результата на корректность формата. Поэтому суммарное число обработанных документов определялось доступным вычислительным временем и скоростью выполнения отдельных шагов. Существенной сложностью стала специфика PDF-формата: в части документов извлекались некорректные Unicode-символы, так как текст хранится как набор глифов и ссылок на карты отображения. OCR не применялся из-за высокой вычислительной стоимости, поэтому документы с искаженным содержимым исключались на последующих шагах обработки. Под «некорректным» Unicode в данном случае понимаются ситуации, когда извлеченная строка содержит большое число нечитаемых символов, управляющих кодов или последовательностей, не соответствующих ожидаемому тексту статьи. В таких

файлах нарушается связность фрагмента текста, что приводит к ошибкам извлечения полей и снижает корректность результата даже при исправной работе модели.

На этапе валидации выполнялась проверка на «битый» Unicode и заполненность ключевых полей. Ключевыми считались поля, необходимые для построения графа знаний и последующих запросов: название, год, список авторов, а также хотя бы один признак тематической характеристики (предметная область, ключевые слова или краткое тематическое описание). Документы, где эти поля отсутствовали или имели явно некорректный вид, исключались из набора, используемого на этапе построения онтологии и загрузки экземпляров. В результате было отобрано 4250 корректных документов; файлы с некорректным текстом исключались и помечались для последующей OCR-обработки.

На этапе извлечения сравнивались две БЯМ [6]. Модель gemma3:27b давала высокое качество, но требовалось 1–1.5 мин на документ, и иногда происходило зацикливание (например, генерация повторяющихся списков авторов); при этом поддерживалось большое контекстное окно (~256k токенов). Модель qwen3:8b обрабатывала файл за 8–14 с при сопоставимом качестве и более устойчивом следовании промпту, но на части документов наблюдались «зависания»; увеличение длины обрезаемого текста до 20000 символов иногда приводило к нарушению формата ответа (уход от JSON к свободному тексту).

Сопоставление показало, что при работе с корпусом важны два параметра: соблюдение формата ответа и стабильность обработки большого числа документов в одном потоке. Поэтому при последующей обработке выбор модели определялся не только качеством извлечения отдельных полей, но и предсказуемостью результата при массовой обработке.

После завершения этапов извлечения и разметки семантической информации выполнялся переход к графовому представлению данных. В онтологии (рис. 2) задавались набор базовых классов сущностей, их атрибуты и типы семантических связей между ними, а также формальная основа для построения ГЗ.

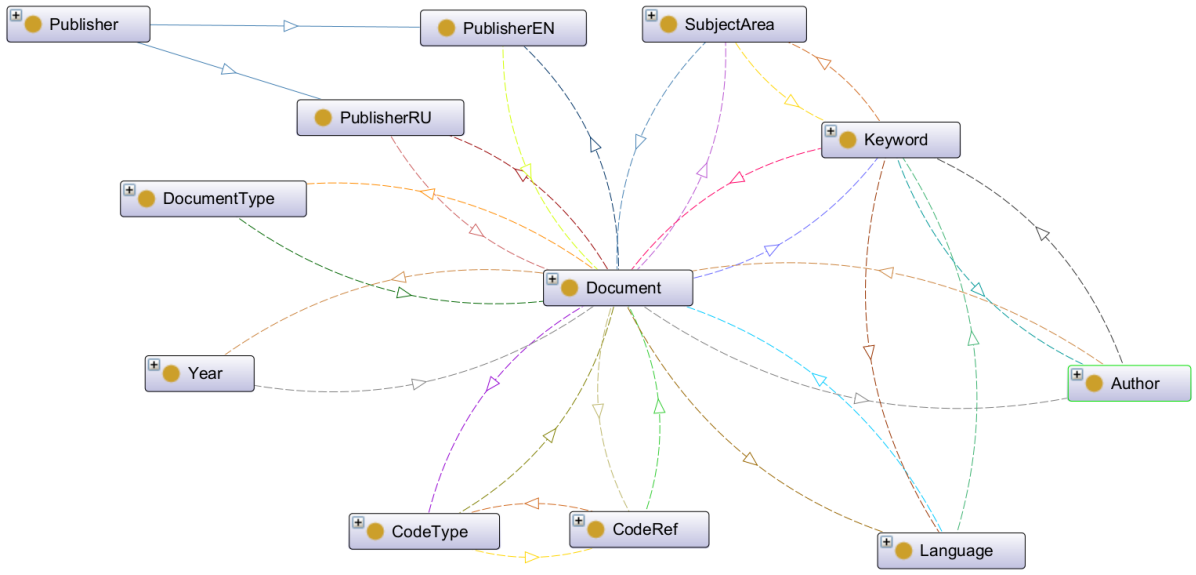

Рис. 2. TBox (схема) используемой онтологии.

Для представления RDF-графа была проведена сериализация в формате Turtle (Terse RDF Triple Language). Разбиение данных по отдельным файлам по классам было использовано как технический прием, упрощающий контроль целостности и повторную загрузку: при необходимости можно было пересоздавать и импортировать только часть экземпляров, не затрагивая базовую онтологию. Для экземпляров были использованы устойчивые идентификаторы, что позволяло повторно связывать одни и те же сущности (например, автор, год, издательство) с различными документами. Для каждого класса формировался файл, содержащий все триплеты-экземпляры данного класса. В базовой онтологии были явно определены классы Author, Document, DocumentType, Keyword, Language, PublisherEN, PublisherRU, SubjectArea, Year, CodeType и CodeRef, представляющие основные типы сущностей корпуса. Между ними были заданы семантические связи, отражающие авторство, тематическую принадлежность, тип публикации, языковую характеристику и временной контекст.

Под предметной областью (SubjectArea) понималась обобщенная тематическая принадлежность научного текста, отражающая основное направление исследования. Значение предметной области извлекалось автоматически с использованием БЯМ на основе анализа фрагмента текста и формулировалось в виде одной категориальной метки (mathematics, physics, chemistry, biology, engineering и т. д.).

Извлечение предметной области было задано как выбор одной метки на документ, что упрощает последующие запросы и группировку материалов. При этом такая схема не отражает междисциплинарные работы и случаи, когда документ содержит несколько направлений. В тексте это проявлялось как неизбежное упрощение: метка использовалась как ориентир для навигации и фильтрации, а не как точная классификация.

Загрузка в GraphDB выполнялась в два этапа: сначала импортировалась базовая онтология, затем последовательно загружались RDF-триплеты экземпляров. Итоговый граф содержит 199490 утверждений: 137252 явных и 62238 полученных выводом на основе аксиом; коэффициент расширения составил 1.45. Явные утверждения соответствуют загруженным триплетам экземпляров и базовой онтологии. Выведенные утверждения формируются механизмом вывода GraphDB на основе аксиом онтологии (например, иерархий классов и ограничений на свойства). Это увеличивает число фактов, доступных для запросов, без ручного дублирования связей в данных.

ПРЕОБРАЗОВАНИЕ ЗАПРОСОВ С ЕСТЕСТВЕННОГО ЯЗЫКА НА SPARQL

Контрольная группа «эталонных» запросов представляет собой вручную собранный и проверенный набор заданий для преобразования NL-запросов в SPARQL. Датасет оформлен как список записей, где каждой записи соответствуют текст NL-запроса на русском языке, эталонный SPARQL-запрос и ожидаемый ответ. Последний фиксировался по результатам выполнения эталонного запроса в GraphDB. После расширения контрольной группы для всех записей был добавлен английский вариант NL-запросов, используемый для отдельного эксперимента с англоязычной инструкцией генерации.

В рамках данного этапа было выполнено тестирование 39 БЯМ на задаче преобразования NL-запросов в SPARQL. В исследование были включены преимущественно компактные модели различных семейств: Qwen, Gemma, Mistral, Llama, CodeLlama, StarCoder, DeepSeek и др. – с количеством параметров до 30 млрд. Дополнительно для части моделей было оценено влияние различных вариантов квантования на качество преобразования NL-запросов в SPARQL.

Для всех моделей были использованы одинаковая контрольная группа NL-запросов, единая инструкция генерации и одна среда выполнения (одинаковое

аппаратное обеспечение). Для каждого NL-запроса модель формировала SPARQL-запрос; далее запрос выполнялся в GraphDB. Корректность оценивалась по совпадению результата выполнения с эталонным запросом: для агрегатных запросов сравнивались числовые значения, для выборок — состав возвращаемых значений. Запросы, не выполнявшиеся из-за синтаксических ошибок или несоответствия схеме данных, учитывались как некорректные [7–9]. Подобный подход к оценке качества преобразования запросов соответствует исследованиям в области семантического парсинга и генерации структурированных запросов [10].

По результатам первичного тестирования для детального сравнения были отобраны четыре модели, показавшие наибольшее число корректно выполненных запросов на контрольной группе: gemma3:4b, qwen3:8b, Qwen3-4B-Instruct и deepseek-coder-7b-instruct.

Табл. 1. Результаты тестирования БЯМ.

Модель	Верно выполненные запросы
gemma3:4b	4
qwen3:8b	39
Qwen3-4B-Instruct	34
deepseek-coder-7b-instruct	12
Общее количество эталонных запросов	50

В табл. 1 представлены результаты тестирования БЯМ, указаны количества корректных преобразований NL-запросов в SPARQL. Лучший результат показала модель qwen3:8b (39/50 за 2 мин 15 с), далее Qwen3-4B-Instruct (34/50 за 1 мин 32 с). Модель deepseek-coder-7b-instruct преобразовала 12 запросов за 4 мин 21 с, а gemma3:4b – 4 запроса за 2 мин 25 с. Таким образом, по совокупности корректности и стабильности генерации модель qwen3:8b оказалась наиболее эффективной для задачи трансляции запросов с естественного языка в SPARQL в рамках корпуса данных [11]. Использование подобных моделей представляет интерес и для задач автоматизированного пополнения и сопровождения графов знаний [12].

Количественные показатели в табл. 1 не описывают характер ошибок. В табл. 2 приведены примеры корректно сгенерированных SPARQL-запросов для

некоторых из БЯМ. По этим примерам и журналам выполнения в GraphDB выделены типовые ограничения генерации.

Табл. 2. Образцы корректно сгенерированных SPARQL-запросов.

Имя БЯМ	NL-запрос	SPARQL-эталон	Результат генерации

gemma3:4b	Сколько всего авторов у документов в 2019 году?	PREFIX : <https://scilib.ai/ontology/semantic-graph/>
		
		SELECT (COUNT(DISTINCT ?author) as ?totalAuthors2019)
		WHERE {
		?document a :Document ;
		:hasYear ?yearInd ;
		:hasAuthor ?author .
		
		?yearInd :year_value 2019 .
		}
			PREFIX : <https://scilib.ai/ontology/semantic-graph/>
			
			SELECT (COUNT(?author) AS ?authorCount)
			WHERE {
			?document a :Document ;
			:hasAuthor ?author ;
			:hasYear ?year .
			?year :year_value 2019 .
			}

qwen3:8b	У скольких авторов есть публикации с важностью 6+ которые содержат в заголовках термин "синтез"?	PREFIX : <https://scilib.ai/ontology/semantic-graph/>
		
		SELECT (COUNT(DISTINCT ?author) as ?authorsCount)
		WHERE {
		?document a :Document ;
		:title ?title ;
		:importance ?importance ;
		:hasAuthor ?author .
		
		FILTER (?importance >= 6)
		FILTER (CONTAINS(LCASE(?title), LCASE("синтез")))
		}
			PREFIX : <https://scilib.ai/ontology/semantic-graph/>
			
			SELECT (COUNT(DISTINCT ?author) AS ?count)
			WHERE {
			?document a :Document ;
			:hasAuthor ?author ;
			:hasYear ?yearInd ;
			:title ?title .
			
			?yearInd :year_value ?year .
			
			FILTER (xsd:integer(?year) >= 6) .
			
			FILTER (CONTAINS(LCASE(?title), LCASE("синтез"))) .
			
			?author a :Author .
			}

Наиболее частыми являются синтаксические ошибки (неверное оформление FILTER и префиксов), а также структурные ошибки, когда запрос формально выполняется, но возвращает результат, не соответствующий эталону. Отдельный

класс ошибок связан с выбором условий: модель может добавлять лишние связи. Кроме того, встречаются ошибки согласования со схемой данных: использование отсутствующих свойств и классов или смешение типов значений в фильтрах. Эти случаи требуют проверки не только текста SPARQL, но и результата выполнения.

Анализ успешных преобразований показал, что модели устойчивее работают с агрегатными запросами, основанными на операциях подсчета (COUNT) и простой фильтрации. При переходе к запросам на извлечение списков сущностей (например, ключевых слов [13] или связанных объектов) чаще возникают ошибки структуры запроса: некорректные переменные в SELECT, пропуски обязательных связей, смешение уровней сущностей и неверные ограничения по условиям отбора.

Дополнительно был проведен эксперимент с англоязычной формулировкой задания: инструкция генерации переведена на английский язык, а в контрольную группу добавлен английский вариант NL-запросов. Ожидание общего роста числа корректных преобразований не подтвердилось. Модели, показавшие высокие результаты на русском языке, сохранили их и на английском; отклонение составило 1–2 запроса в лучшую или худшую сторону. При этом часть моделей с результатом ниже половины контрольной группы улучшила показатель примерно в два раза (например, с 12 до 25 корректных запросов для модели deepseek-coder-7b-instruct). Некоторые модели с нулевым результатом сохранили тот же результат.

Был также проведен эксперимент с квантованием моделей. Для ряда БЯМ были протестированы варианты весов различной точности (например, Q8 и Q4). В пределах рассмотренных малых и средних моделей, с числом параметров от 256M (256 миллионов) до 30B (30 миллиардов), наблюдалась зависимость качества преобразования NL-запросов в SPARQL от степени квантования: при более высокой точности (варианты ближе к исходным весам) число корректных преобразований было выше, при более низкой точности – ниже. Наблюдение получено в рамках указанного диапазона размеров и не переносится автоматически на модели других архитектур и классов без отдельной проверки.

Полученные результаты показывают, что БЯМ могут использоваться как средство выполнения запросов к графовой базе знаний по NL-запросам, если

структура запроса остается простой и проверяется выполнением в GraphDB. Вместе с тем при переходе от агрегатных запросов к выборкам, требующим точного управления структурой графовых объектов, возрастает доля ошибок синтаксиса и смысла запроса, что задает ограничения на класс задач, решаемых без дополнительной валидации и корректировки.

ЗАКЛЮЧЕНИЕ

Разработан процесс семантического анализа корпуса научных статей: статистическая оценка, извлечение семантики, построение онтологии и представление данных в виде RDF-триплетов с загрузкой в GraphDB. Показано, что ограничение длины текста позволяет устойчиво извлекать библиографические и тематические характеристики, достаточные для графового представления. Сравнение языковых моделей выявило различия в эффективности извлечения семантики и преобразования естественных запросов в SPARQL; интеграция БЯМ и графовой модели упрощает доступ к данным через естественно-языковой интерфейс, но генерация сложных структур и запросов требует дальнейшей оптимизации.

Сделанные выводы могут послужить отправными точками для расширения онтологии, повышения устойчивости обработки неструктурированных источников и улучшения механизмов преобразования NL-запросов в формальные графовые запросы.

СПИСОК ЛИТЕРАТУРЫ

1. *Бручес Е.П. и др*. Семантический анализ научных текстов: опыт создания корпуса и построения языковых моделей // Программные продукты и системы. 2021. Т. 34, №. 1. С. 132–144.

2. *Захарова О.И.* Семантический анализ и синтез текстовых данных // Вестник ВГУ. Серия: Системный анализ и информационные технологии. 2023. №. 4. С. 182–208.

3. *Атаева О.М., Серебряков В.А.* Персональная открытая семантическая цифровая библиотека LibMeta. Конструирование контента. Интеграция с источниками LOD // Информатика и ее применения. 2017. Т. 11, № 2. С. 85–100. https://doi.org/10.14357/19922264170210. EDN YTYGAL

4. *Гаянова М.М., Вульфин А.М.* Структурно-семантический анализ научных публикаций выделенной предметной области // Системная инженерия и информационные технологии. 2022. Т. 4, №. 1 (8). С. 37–43.

5. *Гавенко О.Ю., Шашок Н.А.* О повышении качества выходных данных в вопросно-ответной системе, обрабатывающей климатическую информацию // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2024. Т. 22, №. 4. С. 5–16.

6. *Сидорова Е.А., Иванов А.И., Овчинникова К.А.* Извлечение информации из текстов на основе онтологии и больших языковых моделей // Онтология проектирования. 2025. Т. 15, №. 1 (55). С. 114–129.

7. *Литвин А.А., Величко В.Ю., Каверинский В.В.* Метод получения информации из онтологии на основе анализа фразы на естественном языке // Проблеми програмування. 2020.

8. *Ковалев М.* Семантический анализ в подготовке обучающих выборок // Открытые системы. СУБД. 2018. №. 3. С. 25–25.

9. *Shaik S. et al.* Transforming natural language query to SPARQL for semantic information retrieval // International Journal of Engineering Trends and Technology. 2016. Vol. 7. P. 347–350.

10. *Сомов О.Д.* Многозадачное обучение для улучшения генерализации в задаче генерации структурированных запросов // Труды Московского физико-технического института. 2024. Т. 16, №. 2 (62). С. 25–31.

11. *Тунян Э.Г., Сазиков Р.С., Харламов С.А.* Обогащение базы знаний с помощью автоматического извлечения ключевых слов и аннотаций // Успехи кибернетики. 2025. Т. 6, №. 2. С. 108–113.

12. *Босенко Т.М.* Разработка и реализация методов оптимизации малых языковых моделей для эффективной генерации SQL-запросов в образовательных информационных системах // Образовательные ресурсы и технологии. 2025. № 1 (50). С. 54–67. https://doi.org/10.21777/2500-2112-2025-1-54-67

13. *Хуссайн М.Ж.* Извлечение ключевых фраз на основе больших языковых моделей // Известия ЮФУ. Технические науки. 2024. № 5.

GRAPH-BASED SEMANTIC ANALYSIS OF A SCIENTIFIC ARTICLE CORPUS

V. A. Chunikhin[1] [0009-0007-0845-1374], S. A. Zaitsev[2] [0000-0002-7902-9695],

O. M. Ataeva[3] [0000-0003-0367-5575]

[1–3]*Moscow Witte University, Moscow, Russia*

[3]*Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (FRC CSC RAS), Moscow, Russia*

[1]supermen.tut.103103@gmail.com, [2]szaytsev@muiv.ru, [3]OAtaeva@frccsc.ru

Abstract

The problem of effective navigation and search for relevant information within growing volumes of scientific publications necessitates a shift from classical full-text search methods to semantic models. This work proposes an approach to structuring a heterogeneous corpus of scientific texts by constructing a Knowledge Graph (KG). A data processing pipeline is developed, encompassing the extraction of metadata, keywords, and structural elements of articles, followed by their integration into a unified graph. Based on the constructed Knowledge Graph, methods for analyzing explicit and extracting implicit connections between publications are implemented. The research results demonstrate the effectiveness of the graph-based representation of scientific information for uncovering hidden patterns within subject domains and supporting intelligent navigation.

Keywords*: semantic analysis, corpus of scientific articles, knowledge graph, ontology, RDF, SPARQL, large language models, information extraction, graph databases.*

REFERENCES

1. *Bruches E.P. et al.* Semantic analysis of scientific texts: experience in corpus creation and language model building // Software & Systems. 2021. Vol. 34, No. 1. P. 132–144.

2. *Zakharova O.I.* Semantic analysis and synthesis of textual data // Vestnik of Voronezh State University. Series: Systems Analysis and Information Technologies. 2023. No. 4. P. 182–208.

3. *Ataeva O.M., Serebryakov V.A.* LibMeta Personal Open Semantic Digital

Library: Content Construction and Integration with LOD Sources // Informatics and Its Applications. 2017. Vol. 11, No. 2. P. 85–100.
https://doi.org/10.14357/19922264170210. EDN: YTYGAL

4. *Gayanova M.M., Vul'fin A.M.* Structural and semantic analysis of scientific publications in a selected subject area // Systems Engineering and Information Technologies. 2022. Vol. 4, No. 1 (8). P. 37–43.

5. *Gavenko O.Yu., Shashok N.A.* On improving the quality of output data in a question-answering system processing climate information // Bulletin of Novosibirsk State University. Series: Information Technologies. 2024. Vol. 22, No. 4. P. 5–16.

6. *Sidorova E.A., Ivanov A.I., Ovchinnikova K.A.* Information extraction from texts based on ontology and large language models // Ontology of Designing. 2025. Vol. 15, No. 1 (55). P. 114–129.

7. *Litvin A.A., Velichko V.Yu., Kaverinsky V.V*. A method for obtaining information from an ontology based on natural language phrase analysis // Programming Problems. 2020.

8. *Kovalev M.* Semantic analysis in the preparation of training datasets // Open Systems. DBMS. 2018. No. 3. P. 25–25.

9. *Shaik S. et al.* Transforming natural language query to SPARQL for semantic information retrieval // International Journal of Engineering Trends and Technology. 2016. Vol. 7. P. 347–350.

10. *Somov O.D.* Multi-task learning to improve generalization in structured query generation // Proceedings of the Moscow Institute of Physics and Technology. 2024. Vol. 16, No. 2 (62). P. 25–31.

11. *Bosenko T.M.* Development and implementation of optimization methods for small language models for efficient SQL query generation in educational information systems // Educational Resources and Technologies. 2025. No. 1 (50). P. 54–67. https://doi.org/10.21777/2500-2112-2025-1-54-67. EDN JNZQLM

12. *Tunyan E.G., Sazikov R.S., Kharlamov S.A.* Enrichment of a knowledge base by automatic keyword and abstract extraction // Advances in Cybernetics. 2025. Vol. 6, No. 2. P. 108–113.

13. *Khusainov M.Zh.* Keyphrase extraction based on large language models // News of the Southern Federal University. Technical Sciences. 2024. No. 5.

СВЕДЕНИЯ ОБ АВТОРАХ

ЧУНИХИН Вадим Андреевич – аспирант 1 курса Московского университета имени С. Ю. Витте, г. Москва, Россия, по направлению «Системный анализ, управление и обработка информации, статистика», специализируется на разработке и исследовании рекомендательных систем. Научные интересы включают семантический анализ, извлечение информации, обработку естественного языка, а также применение больших языковых моделей для анализа научных данных.

Vadim Andreevich CHUNIKHIN – first year postgraduate student at Moscow Witte University, Moscow, Russia, specializing in Systems Analysis, Control and Information Processing, Statistics, focuses on the development and research of recommender systems. His research interests include semantic analysis, information extraction, natural language processing, as well as the application of large language models for scientific data analysis.

email: supermen.tut.103103@gmail.com

ORCID: 0009-0007-0845-1374

ЗАЙЦЕВ Сергей Александрович – к. т. н., доцент, декан факультета информационных технологий Московского университета имени С.Ю. Витте, г. Москва, Россия. Окончил Курский государственный технический университет по специальности инженер-технолог. Является автором более 45 научных работ, включая 4 патента на изобретения и полезные модели. Научные интересы включают анализ данных, программирование и имитационное моделирование.

Sergey Aleksandrovich ZAITSEV – PhD in Engineering Sciences, Associate Professor, dean of the Faculty of Information Technologies at Moscow Witte University, Moscow, Russia. He graduated from Kursk State Technical University with a degree in Process Engineering. He is the author of more than 45 scientific publications, including 4 patents for inventions and utility models. His research interests include data analysis, programming, and simulation modeling.

email: szaytsev@muiv.ru

ORCID: 0000-0002-7902-9695

АТАЕВА Ольга Муратовна – старший научный сотрудник Вычислительного центра им. А. А. Дородницына ФИЦ ИУ РАН, к. т. н. Область научных интересов: системное программирование, базы данных, инженерия знаний и онтологии.

Olga Muratovna ATAEVA – senior researcher at the Dorodnicyn Computing Center, Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences; Candidate of Technical Sciences (PhD). Research interests: systems programming, databases, knowledge engineering, ontologies. Number of scientific publications – 80.

email: oataeva@frccsc.ru

ORCID: 0000-0003-0367-5575

Материал поступил в редакцию 18 марта 2026 года