**Полякова, О. А.** Разработка автоматизированной информационно-поисковой системы медиаконтента на естественном языке / О. А. Полякова, К. П. Кузнецов // Прикладная математика и вопросы управления. – 2025. – № 2. – С. 83–98. DOI 10.15593/2499-9873/2025.2.06

**Библиографическое описание согласно ГОСТ Р 7.0.100–2018**
Полякова, О. А. Разработка автоматизированной информационно-поисковой системы медиаконтента на естественном языке / О. А. Полякова, К. П. Кузнецов. – Текст : непосредственный. – DOI 10.15593/2499-9873/2025.2.06 // Прикладная математика и вопросы управления / Applied Mathematics and Control Sciences. – 2025. – № 2. – С. 83–98.

пермский политех

**ПРИКЛАДНАЯ МАТЕМАТИКА И ВОПРОСЫ УПРАВЛЕНИЯ**
**№ 2, 2025**
https://ered.pstu.ru/index.php/amcs


Научная статья
DOI: 10.15593/2499-9873/2025.2.06
УДК 004.41

# Разработка автоматизированной информационно-поисковой системы медиаконтента на естественном языке

**О.А. Полякова, К.П. Кузнецов**

Пермский национальный исследовательский политехнический университет,
Пермь, Российская Федерация

О СТАТЬЕ

Получена: 30 июня 2025
Одобрена: 15 июля 2025
Принята к публикации:
08 августа 2025

**Финансирование**
Исследование не имело спонсорской поддержки.
**Конфликт интересов**
Авторы заявляют об отсутствии конфликта интересов.
**Вклад авторов**
равноценен.

**Ключевые слова:**
система, естественный язык, поисковый запрос, медиаконтент, нейросетевые модели, Qwen3, Vikhr, Saiga, YandexGPTLite 5, трансформер, NLP

АННОТАЦИЯ

Представлена разработка автоматизированной информационно-поисковой системы медиаконтента по запросу пользователя на естественном языке. Описаны современные проблемы поиска в условиях стремительного роста объемов и разнообразия медиаконтента, а также выявлены ограничения традиционных методов поиска, основанных на точном совпадении ключевых слов и метаданных. Особое внимание уделено анализу методов обработки естественного языка (NLP), таких как токенизация, лемматизация, векторизация и вычисление семантической близости, а также использованию нейросетевых моделей на основе архитектуры «трансформер». Проведен сравнительный анализ современных открытых языковых моделей, включая Qwen3, Vikhr, Saiga и YandexGPTLite 5, с точки зрения их применимости для многоязычных и мультимодальных задач поиска и генерации текстов. В работе предложены решения по интеграции современных NLP-методов и нейросетевых алгоритмов в серверную часть поисковой системы, что позволяет повысить релевантность, точность и удобство поиска медиаконтента по неструктурированным и неточным запросам. Представленные подходы обеспечивают баланс между качеством поиска, производительностью и универсальностью системы, а также открывают перспективы для дальнейшего развития интеллектуальных поисковых сервисов.

© **Полякова Ольга Андреевна** - кандидат технических наук, доцент кафедры «Информационные технологии и автоматизированные системы» e-mail: olgastratum@mail.ru
**Кузнецов Кирилл Петрович** – бакалавр кафедры «Информационные технологии и автоматизированные системы», ORCID 0009-0007-6854-111X

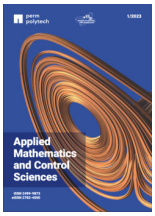
**Perm Polytech Style:** Poliakova O.A., Kuznetsov K.P. Development of an automated information retrieval system for media content in natural language. *Applied Mathematics and Control Sciences.* 2025, no. 2, pp. 83–98. DOI: 10.15593/2499-9873/2025.2.06

**MDPI and ACS Style:** Poliakova, O. A.; Kuznetsov, K.P. Development of an automated information retrieval system for media content in natural language. *Appl. Math. Control Sci.* **2025, 2**, 83–98. https://doi.org/10.15593/2499-9873/2025.2.06

**Chicago/Turabian Style:** Poliakova, Olga A., and Kirill P. Kuznetsov. 2025. "Development of an automated information retrieval system for media content in natural language". *Appl. Math. Control Sci.* no. 2: 83–98. https://doi.org/10.15593/2499-9873/2025.2.06

perm polytech

APPLIED MATHEMATICS AND CONTROL SCIENCES
№ 2, 2025
https://ered.pstu.ru/index.php/amcs




Article
DOI: 10.15593/2499-9873/2025.2.06
UDC 004.41

# Development of an automated information retrieval system for media content in natural language

**O.A. Poliakova, K.P. Kuznetsov**

Perm National Research Polytechnic University, Perm, Russian Federation

ARTICLE INFO

Received: 30 June 2025
Approved: 15 July 2025
Accepted for publication:
11 August 2025

**Funding**
This research received no external funding.
**Conflicts of Interest**
The authors declare no conflict of interest.
**Author Contributions**
equivalent.

**Keywords:**
relational attention module; pre- trained model; Transformer Seq2Seq; text to SQL conversion; Spider dataset

ABSTRACT

The paper presents the development of an automated information retrieval system of media content on user's request in natural language. It describes the current problems of search in the context of rapid growth of media content volume and diversity, and also reveals the limitations of traditional search methods based on the exact match of keywords and metadata. Special attention is paid to the analysis of natural language processing (NLP) methods, such as tokenization, lemmatization, vectorization and semantic proximity computation, as well as the use of neural network models based on the "transformer" architecture. A comparative analysis of modern open language models, including Qwen3, Vikhr, Saiga and YandexGPTLite 5, in terms of their applicability for multilingual and multimodal text retrieval and generation tasks is carried out. The paper proposes solutions for integrating modern NLP-methods and neural network algorithms into the server side of the search system, which allows to improve the relevance, accuracy and convenience of media content search for unstructured and imprecise user queries. The presented approaches provide a balance between search quality, performance and versatility of the system, and open up prospects for further development of intelligent search services.

© **Olga A. Poliakova** – CSc, Associate Professor of the Department of Information Technologies and Automated Systems, e-mail: olgastratum@mail.ru
**Kirill P. Kuznetsov** – Bachelor of the Department of Information Technologies and Automated Systems, ORCID 0009-0007-6854-111X

## Введение

Медиаконтент – ключевая часть цифровой среды, характеризуется непрерывным и стремительным ростом объемов данных [1]. Ежедневно создается и публикуется значительное количество новых цифровых объектов, включая фильмы, музыкальные произведения, изображения и интерактивные игры. Такой экспоненциальный рост информационного массива порождает серьезные вызовы в области его эффективной обработки, систематизации и поиска.

Традиционные методы поиска, основанные на точном совпадении по названию или заданным тегам, оказываются недостаточно эффективными в условиях масштабных и разнородных медиабиблиотек. Основная проблема заключается в том, что пользователи часто не располагают точной информацией о названии искомого медиаконтента, что существенно затрудняет процесс поиска. В таких случаях актуальным становится использование описательных запросов на естественном языке, которые содержат приблизительное или контекстуальное описание искомого материала.

Для обработки подобных запросов применяются методы обработки естественного языка (Natural Language Processing, NLP) [2], которые позволяют извлекать семантическую информацию из текстовых описаний, анализировать синтаксические и смысловые структуры, а также учитывать контекст запроса. Это открывает новые перспективы для разработки интеллектуальных систем поиска, способных интерпретировать нечеткие или неполные запросы и обеспечивать релевантный результат.

В современных способах поиска значительную роль играют нейросетевые технологии и методы глубокого обучения [3], которые позволяют создавать модели для работы с семантикой и контекстом запросов, а не только с ключевыми словами. Такие модели обучаются на больших корпусах данных и способны выявлять скрытые зависимости и смысловые связи, что существенно повышает качество поиска. В частности, методы нечеткого поиска (fuzzy search) [4] и семантического сопоставления обеспечивают гибкость и адаптивность системы к разнообразным формулировкам запросов.

Целью настоящей работы заключается в создании высокоэффективного инструмента поиска медиаконтента, который позволит пользователям быстро и с минимальными усилиями находить необходимую информацию. Система должна обеспечивать возможность ввода поисковых запросов на естественном языке, содержащих семантические особенности и приблизительное описание искомого контента, без необходимости использования дополнительных параметров (например, выбор жанров и страны) или точных ключевых слов. Для этого планируется интеграция современных методов NLP и нейросетевых алгоритмов, которые обеспечивают глубокий семантический анализ и интерпретацию запросов пользователей [5].

Таким образом, внедрение актуальных технологий обработки естественного языка и искусственного интеллекта в системы поиска медиаконтента позволит значительно повысить качество и удобство поиска [1; 3].

## Описание модели поиска медиаконтента по запросу на естественном языке с использованием стандартных методов

Краткое описание модели поиска медиаконтента по запросу на естественном языке с использованием стандартных методов выглядит следующим образом (рис. 1):

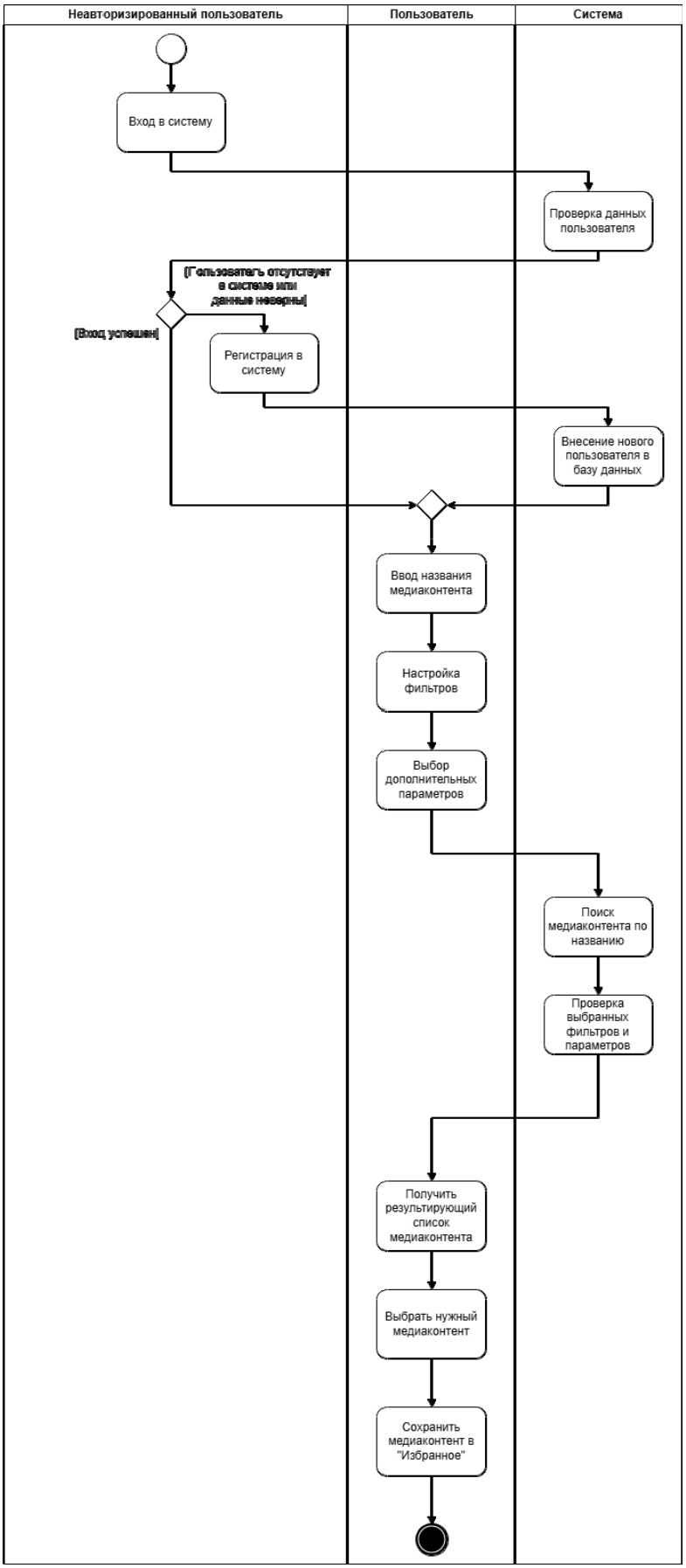



Рис. 1. Модель поиска медиаконтента по запросу с использованием стандартных методов

1. Пользователь входит в систему, чтобы иметь возможность добавить киноленту в «Избранное».

2. Если нужный медиаконтент не найден на главной странице, он использует поиск, вводя название или примерное описание киноленты.

3. Для повышения точности настраиваются фильтры или дополнительные параметры.

4. Система обрабатывает запрос, сравнивая его с названиями и фильтрами.

5. Если название медиаконтента обладает высокой степенью соответствия запросу, то кинолента добавляется в результирующую выборку.

6. Пользователь выбирает нужный медиаконтент и добавляет его в «Избранное» для быстрого доступа в будущем.

Основными причинами низкой эффективности существующих поисковых систем являются неполное использование всей доступной информации о медиаконтенте при поиске, а также избыточное количество дополнительных параметров по метаданным для сужения области поиска. Вышеописанное требует от пользователя дополнительного времени, что и приводит к неэффективному процессу поиска.

Для устранения недостатков предлагаются следующие решения:

– применение методов обработки естественного языка для анализа запросов и метаданных медиаконтента [2];

– формирование наборов ключевых слов для каждого медиаконтента, что позволит хранить более полную и структурированную информацию о каждой киноленте;

– автоматизация процесса выбора и настройки фильтров в поисковых запросах.

Для разрабатываемой системы используются следующие методы NLP:

1. Токенизация – процесс сегментации текстового набора на отдельные слова и символы для представления текста в понятном для системы виде без утраты смысловой нагрузки [1; 6].

2. Лемматизация – процесс приведения словоформ к их базовой форме, называемой леммой. Лемма представляет собой стандартную форму слова, которая может быть найдена в словаре. Например, для слов «играю», «играли» и «играют» леммой будет «играть». Основной принцип лемматизации заключается в выделении минимальных значимых единиц языка и определения роли слова в предложении [6].

3. Мешок слов – упрощенная форма отображения текста, которая фиксирует присутствие конкретных лексических единиц в тексте, игнорируя их расположение относительно друг друга [5].

4. Term Frequency-Inverse Document Frequency (TF-IDF) – метод оценки значимости каждого слова в конкретном документе с учетом частоты его появления и распространенности этого слова во всей совокупности текстов, а также представление набора документов и запросов пользователей на естественном языке в векторной форме. Метод выделяет ключевые термины и определяет, какие слова имеют наибольший вес для отдельного документа в контексте всей коллекции [7].

Формула TF-IDF выглядит следующим образом:

$$TfIdf\left(d\right)=\frac{f\left(d\right)}{k}\cdot\ln\left(\left(\frac{1+n}{1+df\left(d\right)}\right)+1\right), \tag{1}$$

где $f$ – количество вхождений слов $d$ в документе; $k$ – общее количество слов в документе, $n$ – общее число документов в коллекции; $df$ – количество документов в коллекции, в которых слово $d$ встречается минимум один раз; $t_i$ – документ, который идет $i$-м в коллекции.

Результатом выполнения методов является список медиаконтента, который представлен в векторной форме и наиболее близок к пользовательскому запросу по косинусной близости.

Но такого результата обработки запроса на естественном языке может оказаться недостаточно. Классические подходы в обработке естественного языка сталкиваются с рядом ограничений [8]:

– ограниченный набор признаков и отсутствие глубокого семантического анализа. Традиционные методы обработки естественного языка основаны на ограниченном наборе признаков и заранее заданных правилах, словарях и простых статистических характеристиках, таких как частоты слов. При этом статические модели (например, скрытая марковская модель [9]) часто игнорируют контекстуальные связи и многозначность слов, что приводит к неэффективной обработке омонимов, синонимов и сложных синтаксических конструкций [10]. Особенно снижена эффективность таких методов при работе с короткими запросами и мультиязычными данными, где отсутствие глубокого семантического анализа и контекстуального понимания существенно ограничивает качество результатов поиска [1];

– проблемы масштабирования. С ростом объемов данных классические алгоритмы становятся менее эффективными, так как они не способны адекватно обрабатывать длинные тексты и учитывать глобальные зависимости между словами.

Перспективным подходом для устранения недостатков классических подходов является использование нейросетевой модели на основе архитектуры «трансформер» [11], ключевым преимуществом в данном случае является механизм самовнимания, позволяющий анализировать глобальные зависимости в тексте без последовательной обработки.

Основные компоненты трансформера [11]:

1. Кодировщик. Он принимает на вход последовательность элементов (например, слов), преобразует их в векторные представления (эмбеддинги или свертки) и добавляет к ним информацию о позиции каждого элемента. Для учета взаимосвязей между элементами используется механизм внимания, который вычисляет взвешенное представление значений на основе сходства запросов и ключей.

2. Декодировщик. Также состоит из нескольких слоев, каждый из которых содержит два механизма внимания. Первый механизм – многоголовое самовнимание с маскированием, предотвращающее использование информации о будущих позициях. Второй механизм – кросс-внимание, которое связывает выходы кодировщика с текущим состоянием декодировщика, обеспечивая взаимодействие между входной и выходной последовательностями.

Основной механизм внимания описывается следующей формулой [11]:

$$\text{Attention}\left(Q,K,V\right)=\text{softmax}\left(\frac{QK^{T}}{\sqrt{d_{x}}}\right)V, \tag{2}$$

где $Q,K,V$ – матрицы запросов, ключей и значений, а $d_x$ – размерность ключей, которая используется для масштабирования.

В декодировщике для самовнимания применяется маска $M$, которая запрещает доступ к будущим позициям [11]:

$$\text{MaskedAttention}\left(Q,K,V\right)=\text{softmax}\left(\frac{QK^{T}}{\sqrt{d_{x}}}+M\right)V. \tag{3}$$

Для более глубокого анализа взаимосвязей применяется многоголовое внимание (Multi-Head Attention), которое объединяет результаты нескольких параллельных механизмов внимания [11]:

$$\text{MultiHead}\left(Q,K,V\right)=\text{Concat}\left(\text{head}_1,\ldots,\text{head}_h\right)W^O, \tag{4}$$

$$\text{head}_i=\text{Attention}\left(QW_i^Q,KW_i^K,VW_i^V\right), \tag{5}$$

здесь $h$ – число «голов» внимания $W_i^Q,W_i^K,W_i^V$ – обучаемые матрицы весов для каждой головы, $W^O$ – матрица объединения результатов.

После механизмов внимания в обоих компонентах применяется сеть прямой связи (Feed-Forward Network) с нелинейной активацией [11]:

$$FFN\left(x\right)=\max\left(0,xW_1+b_1\right)W_2+b_2, \tag{6}$$

где $W_1,W_2,b_1,b_2$ – параметры сети.

Для устойчивости обучения и улучшения сходимости применяется слой нормализации с остаточной связью [11]:

$$\text{LayerNorm}\left(x+\text{Sublayer}\left(x\right)\right), \tag{7}$$

где $\text{Sublayer}\left(x\right)$ – выход механизма внимания или FFN.

В основе современных способов поиска информации лежат открытые нейросетевые модели, которые демонстрируют высокую эффективность в задачах понимания и генерации текста. Для повышения качества обработки на русском языке используются модели, которые обучены на больших корпусах русскоязычных данных.

1. Модель Qwen. Разработана компанией Alibaba Cloud [12], представляет собой семейство открытых мультимодальных языковых моделей нового поколения, отличающихся поддержкой работы не только с текстом, но и с изображениями и аудио. Архитектурно Qwen построен на трансформере с оптимизациями Mixture of Experts (MoE), что обеспечивает высокую вычислительную эффективность и улучшенную производительность на различных задачах. Qwen поддерживает работу с длинным контекстом (до 128 000 токенов), обладает расширенной мультиязычной поддержкой и демонстрирует выдающиеся результаты в бенчмарках по генерации и пониманию текста, превосходя многие западные аналоги по ряду показателей [13].

2. Модель Vikhr. Отечественная языковая модель, которая ориентирована на работу с русскоязычными данными и задачами обработки естественного языка. Она создана с учетом особенностей морфологии и синтаксиса русского языка, что позволяет ей более точно интерпретировать сложные запросы, обеспечивать релевантность поиска и поддерживать высокое качество генерации текста в различных предметных областях [14].

3. Модель Saiga. Отечественная модель, которая выделяется оптимизацией под задачи поиска, диалога и генерации текстов на русском языке. Saiga прошла дообучение на больших объемах отечественных текстовых данных, что обеспечивает ей конкурентоспособность при решении задач смыслового поиска, классификации и автоматического ответа на вопросы, связанные с русскоязычным контентом [15].

4. Yandex GPTLite 5. Облегченная версия крупной языковой модели от «Яндекса» [16], которая разработана для эффективной работы с русскоязычными и англоязычными тек-

стами в условиях ограниченных вычислительных ресурсов. Модель ориентирована на задачи генерации, поиска, диалога и понимания текста, а также на интеграцию в корпоративные и пользовательские сервисы [17; 18].

В табл. 1 представлен анализ рассмотренных нейросетевых параметров.

Таблица 1

Анализ нейросетевых моделей

| Параметр | Qwen3 | YandexGPTLite 5 | Vikhr | Saiga |
|---|---|---|---|---|
| Размер модели | 8 млрд параметров | ~3 млрд параметров | 1 млрд параметров | ~8 млрд параметров |
| Языковая поддержка | Более 110 языков, включая русский – универсальная многоязычная модель | Русский, английский | Русский, английский (билингвальная модель) | Русский (основной), английский (ограниченно) |
| Особенности | Гибридная архитектура MoE, высокая эффективность, поддержка длинного контекста, мультимодальность, улучшенная точность в математике и программировании | Компактная, поддержка длинного контекста (до 32 тыс. токенов), интеграция с сервисами «Яндекса» | Специализация на русском языке, глубокое дообучение, поддержка инструкций, открытый исходный код | Оптимизирована под диалоговые задачи, быстрый отклик, высокая релевантность на русском языке |
| Область применения | Многоязычные задачи генерации и понимания текста, семантический поиск, мультимодальные задачи | Поиск, генерация текстов, чат-боты, корпоративные решения, обработка больших текстовых массивов | Генерация и понимание текста на русском, смысловой поиск, диалоговые системы, классификация | Диалоговые ассистенты, смысловой поиск, генерация и обработка текстов на русском языке |

Чем выше размер и универсальность модели, тем шире спектр ее применения и выше требования к вычислительным ресурсам. Наиболее универсальным и масштабируемым решением из рассмотренных в табл. 1 является Qwen3, которая поддерживает работу более чем с 110 языками, включая русский, и способна эффективно обрабатывать большие объемы данных благодаря гибридной архитектуре Mixture-of-Experts (MoE) [13] и поддержке длинного контекста. Это делает Qwen3 оптимальным выбором для многоязычных и мультимодальных задач генерации и поиска информации, а также для проектов, где важны высокая точность, масштабируемость и эффективность использования ресурсов.

Использование Qwen3 в связке с языком программирования JavaScript [19] позволяет интегрировать современные методы поиска и генерации текста непосредственно в серверную часть, которая разработана на основе платформы Node.js. Запросы пользователей формируются на клиентской части, которая реализована с использованием библиотеки React [20]. Такой подход обеспечивает не только высокую релевантность результатов для русскоязычных и международных пользователей, но и гибкость масштабирования системы под задачи различной сложности, что особенно важно для поиска медиаконтента по неструктурированным и нечетким пользовательским запросам.

Выбор Qwen3 для реализации автоматизированной информационно-поисковой системы медиаконтента по запросу на естественном языке позволяет достичь баланса между качеством поиска, производительностью и универсальностью решения.

Для повышения качества поиска медиаконтента в информационно-поисковой системе применяется комбинированный подход, который объединяет классический частотный анализ и возможности нейросетевой модели. В основе алгоритма лежит обработка текстовых данных методами NLP с использованием методов лемматизации и фильтрации стоп-слов, что позволяет выделить ключевые слова и нормализовать запрос пользователя.

1. Пользовательский запрос и набор ключевых слов преобразуются в токены, которые разбиваются на леммы, из которых удаляются служебные слова.

2. Формируется векторное представление текстов с помощью мешка слов и TF-IDF. Полученные числовые векторы нормализуются по длине для точного сравнения запроса и набора ключевых слов.

3. Сравнение векторов запроса и наборов ключевых слов по косинусной близости. Наборы ключевых слов, которые обладают наибольшей степенью соответствия, добавляются в предварительный результат, что позволяет ранжировать медиаконтент по степени релевантности.

4. После получения предварительного списка релевантных результатов система обращается к нейросетевой модели (например, к Qwen3), которая через API принимает сформированный промпт, содержащий пользовательский запрос и ключевые слова из базы.

5. Нейросетевая модель анализирует полученную информацию и возвращает уточненный ответ, который помогает выбрать наиболее подходящие варианты медиаконтента.

Таким образом, интеграция нейросетевой модели в процесс поиска позволяет существенно повысить качество выдачи результата за счет глубокого анализа смысловых связей между запросом пользователя и медиаконтентом. Для реализации подобного подхода в рамках автоматизированной информационно-поисковой системы был разработан соответствующий программный модуль.

## Описание модели поиска медиаконтента по запросу на естественном языке с использованием разрабатываемой системы

На основе модели, показанной на рис. 1, а также выявленных недостатков и предложенных способов их устранения была создана новая модель поиска медиаконтента с использованием разрабатываемой информационно-поисковой системы. Данная модель отображена на рис. 2 в виде UML-диаграммы.

Ключевое отличие модели с использованием разработанной системы от модели, которая использует стандартные методы поиска, заключается в том, что пользователю достаточно ввести только свой запрос, где дается описание и различная информация о медиаконтенте, который необходимо найти, поэтому в модели, приведенной на рис. 2, отсутствует настройка фильтров и дополнительных параметров, а система обрабатывает запрос с помощью методов NLP и набора ключевых слов.

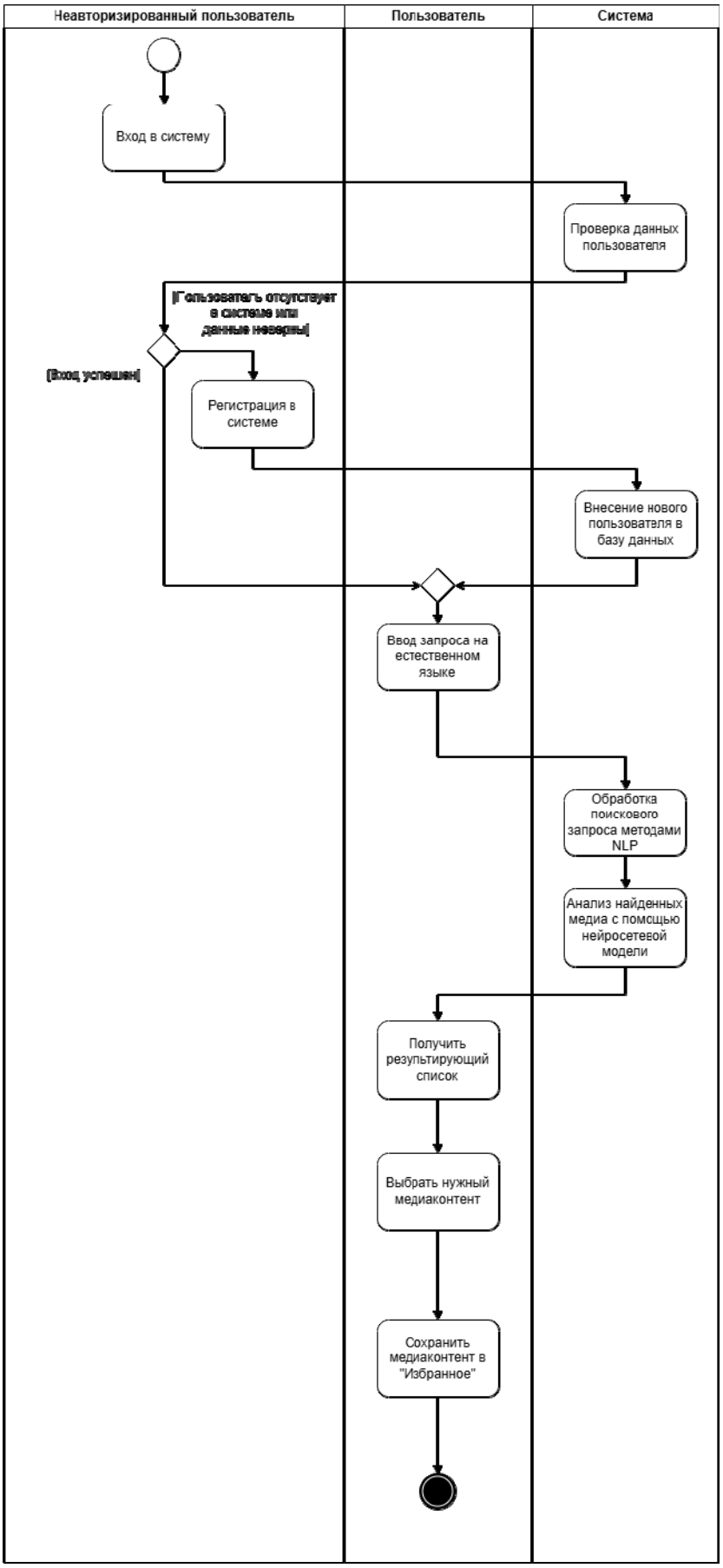



Рис. 2. Модель поиска медиаконтента по запросу с использованием разрабатываемой системы

## Апробация результатов

Для оценки эффективности разработанной системы медиаконтента проведен эксперимент, имитирующий реальную пользовательскую ситуацию поиска. В качестве тестового задания был сформулирован запрос: «сказочный российский фильм про самозванца и любовь». Данный запрос вводится в поисковую строку системы. С помощью методов NLP обрабатываются все доступные киноленты в базе данных. На рис. 3 представлен результат обработки запроса с использованием NLP.

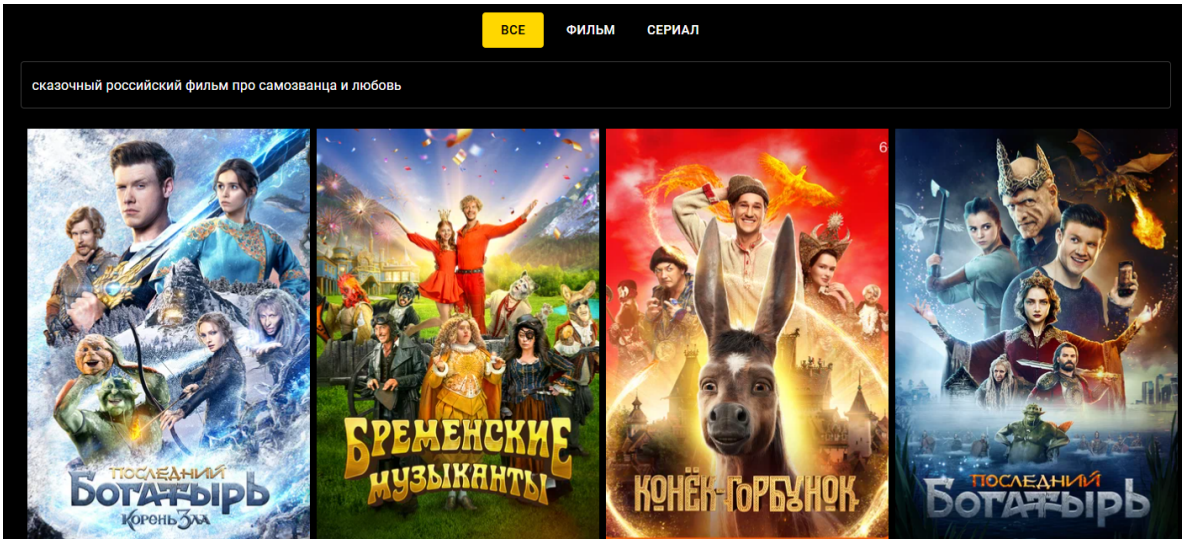



Рис. 3. Пример работы поискового запроса с использованием разработанной системы

Результатом обработки является выборка из четырех кинолент, процент соответствия запросу которых выше заданной. На следующем этапе система использует нейросетевую модель для дополнительной интеллектуальной фильтрации и ранжирования результатов. После обработки модель дает следующий ответ: «Наилучшим образом отвечают на ваш запрос фильмы “Последний Богатырь” и “Последний Богатырь: Корень Зла”». Таким образом, итоговая выборка была сокращена с четырех до двух позиций, что существенно облегчило пользователю процесс выбора и повысило точность поиска.

Таблица 2

Результаты поисковых запросов

| № запроса | Содержание запроса | Предварительный результат с использованием NLP | Результат после обработки нейросетевой моделью | Доля сокращения, % |
|---|---|---|---|---|
| 1 | «Компьютерный зверек, который любит апельсины. Кроме этого, у него есть друг по имени Гена. Это фильм с живыми актерами» | 7 кинолент:<br>1. Чебурашка (1971).<br>2. Чебурашка (2022).<br>3. Чебурашка. Выходной (2023).<br>4. Крокодил Гена (1983).<br>5. Шапокляк (1974).<br>6. Чебурашка и крокодил Гена (1971).<br>7. Чебурашка идет в школу (1983) | 2 киноленты:<br>1. Чебурашка (2022).<br>2. Чебурашка. Выходной (2023) | 71 |

Окончание табл. 2

| № запроса | Содержание запроса | Предварительный результат с использованием NLP | Результат после обработки нейросетевой моделью | Доля сокращения, % |
|---|---|---|---|---|
| 2 | «Фильм про советского инженера, который изобрел уникальную машину времени» | 5 кинолент:<br>1. Малыш (2019).<br>2. Гостья из будущего (1984).<br>3. Человек с бульвара Капуцинов (1987).<br>4. Время вперед! (1965).<br>5. Кин-дза-дза! (1986) | 1 кинолента:<br>Гостья из будущего (1984) | 80 |
| 3 | «Российский фильм про человека, который вынужден жить в деревне, которая искусственно построена другими людьми.<br>В фильме сделан акцент на отношения между людьми» | 7 кинолент:<br>1. Дети понедельника (1997).<br>2. Холоп (2019).<br>3. Холоп 2 (2024).<br>4. Жила-была одна баба (2011).<br>5. Суходол (2011).<br>6. Любовь земная (1974).<br>7. Светлая речка Вздвиженка (1971) | 2 киноленты:<br>1. Холоп (2019).<br>2. Холоп 2(2024) | 71 |
| 4 | «Интересуют российские фильмы про волшебство и с элементами приключений и семейной драмы» | 5 кинолент:<br>1. Волшебник Изумрудного города. Дорога из желтого кирпича (2025).<br>2. Мажор в Сочи (2023).<br>3. Финист. Первый богатырь (2025).<br>4. Отпуск в октябре (2021).<br>5. Свет (2020) | 2 киноленты:<br>1. Волшебник Изумрудного города (2025).<br>2. Финист. Первый богатырь (2025) | 60 |
| 5 | «Российский сериал с сильным актерским составом и хорошей режиссурой. В истории что-то про оператора и режиссера, а также там есть любовная драмма» | 6 кинолент:<br>1. Метод (2015).<br>2. Ликвидация (2007).<br>3. Мажор (2014).<br>4. Оттепель (2013).<br>5. Хрустальный (2024).<br>6. Перевал Дятлова (2020) | 2 сериала:<br>1. Оттепель (2013).<br>2. Ликвидация (2007) | 67 |
| **Средний результат** | **-** | **-** | **-** | **69,8** |

Результаты поисковых запросов показали, что применение разработанной системы позволяет существенно сократить объем предварительных результатов, полученных с использованием NLP, при этом сохраняя релевантность и точность рекомендаций по российским фильмам и сериалам. Средняя доля сокращения составляет 69,8 %, что подтверждает эффективность модели в фильтрации и уточнении информации для облегчения пользователями поиска наиболее подходящих кинолент по сложным и развернутым запросам.

Таким образом, разработанная система повышает качество выдачи и снижает информационную нагрузку, что свидетельствует о ее высокой практической ценности.

## Валидация результатов

Для проверки качества результатов разработанной системы в сравнении с существующими поисковыми системами киносервисов проведен сравнительный эксперимент.

Составлен список из 20 неформализованных поисковых запросов, для каждого из которых заранее определен искомый медиаконтент, который необходимо найти. Каждый запрос вводится в разработанную систему, а также в существующие поисковые системы по 3 раза. Для сравнительного анализа берутся первые 25 кинолент из результата. В итоге общее число медиаконтента на каждую поисковую систему составляет 1500 кинолент. Медиаконтент, подходящий по смыслу к запросу, считается релевантным. Медиаконтент, который нужно найти, считается искомым.

Критерии оценки:

1. Релевантность выдачи. Для каждой системы высчитывается количество релевантных кинолент из общего числа кинолент. Процент релевантности рассчитывается как отношение количества релевантных кинолент к общему их числу. Критерий отражает, насколько качественно система формирует выдачу по запросу, исключая нерелевантный медиаконтент из выборки 25 кинолент.

2. Точность попадания искомого медиаконтента. Для каждого поискового запроса фиксируется, на какой позиции в выдаче находится искомый медиаконтент. Особое внимание уделяется случаям, когда искомый медиаконтент занимает первое или второе места. Процент таких попаданий рассчитывается как отношение количества запросов к искомым кинолентам на первом или втором месте к общему числу поисковых запросов. Критерий отражает качество ранжирования поисковой системы, показывая, насколько часто искомый медиаконтент занимает одну из двух верхних позиций в выдаче.

На рис. 4 и 5 представлены диаграммы с расчетом критериев оценки систем.

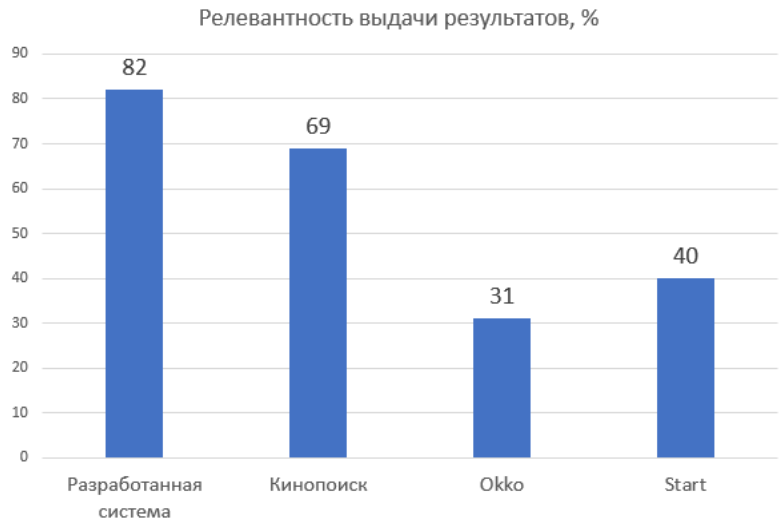



Рис. 4. Диаграмма релевантности результатов

Сравнительный анализ двух показателей позволяет сделать вывод, что разработанная система превосходит существующие поисковые системы по общему уровню релевантности выдачи, что обеспечивает получение более качественного набора кинолент из неформализованных списков. В то же время «Кинопоиск» демонстрирует наивысшую точность ранжирования, чаще других размещая искомый медиаконтент на первых двух позициях. Это может быть связано с особенностями алгоритмов ранжирования релевантных результатов.

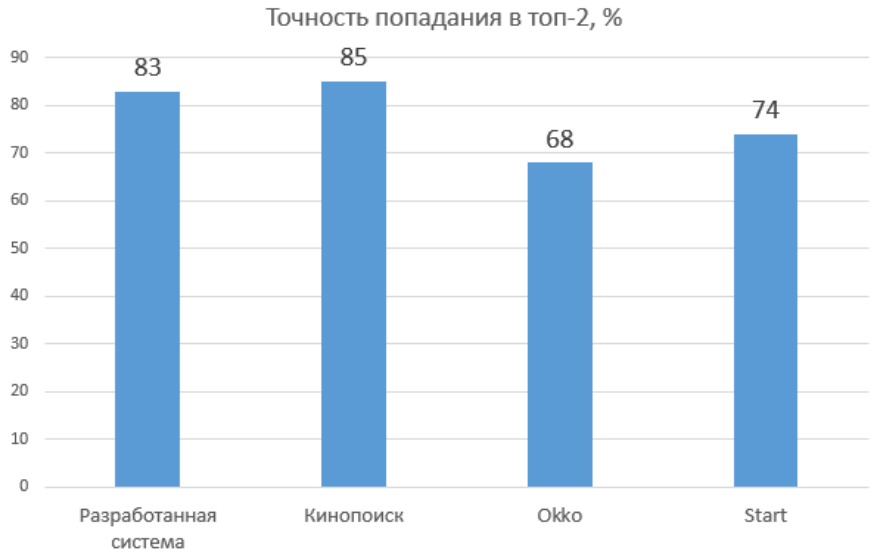



Рис. 5. Диаграмма точности попадания в топ-2

Таким образом, через сравнительный анализ разработанной и существующих поисковых систем проведена валидация результатов.

## Заключение

Разработанная система обеспечивает высокую релевантность и точность поиска даже при отсутствии у пользователя точной информации о названии или тегах медиаконтента с помощью интеграции современных методов NLP и нейросетевых моделей на архитектуре «трансформер», что позволяет анализировать смысловые и контекстуальные особенности пользовательских запросов. Система выделяется универсальностью и масштабируемостью: она способна эффективно работать с большими объемами разнородных данных, поддерживает мультиязычные и мультимодальные сценарии поиска, а также быстро адаптируется под различные предметные области и типы медиаконтента.

Система реализована с использованием языка программирования JavaScript [19] и нейросетевой модели Qwen3 [13], с помощью которой реализуется поддержка мультиязычных сценариев поиска, а также повышается качество обработки русскоязычных запросов пользователей. Серверная часть построена с использованием платформы Node.js, которая обеспечивает высокую производительность, масштабируемость и интеграцию с современными нейросетевыми моделями и алгоритмами NLP на основе нейросетевых алгоритмов и методов NLP, что позволяет эффективно анализировать и интерпретировать сложные поисковые запросы на естественном языке. Клиентская часть системы представляет собой WEB-приложение, которое состоит из JSX-элементов, построенных на основе библиотеки React [20].

Новизна системы заключается в возможности обработки неструктурированных и описательных запросов. В отличие от традиционных поисковых решений, система реализует интеллектуальный поиск медиаконтента по смыслу, а не только по ключевым словам, что минимизирует влияние человеческого фактора и позволяет получать релевантные результаты даже при неполной или неточной исходной информации. Внедрение системы позволяет существенно повысить эффективность поиска, снизить временные затраты пользователей, а также расширить возможности интеллектуальных поисковых сервисов для раз-

личных категорий пользователей, включая специалистов по медиаконтенту, системных аналитиков и конечных потребителей цифровых продуктов.

Система находится в стадии опытной эксплуатации.

## Список литературы

1. Афанасьева, Е.А. Роль автора в создании медиаконтента: трансформация профессиональных практик / Е.А. Афанасьева // Журналистский ежегодник. – 2015. – С. 151–154.

2. Хобсон, Лейн. Обработка естественного языка в действии / Лейн Хобсон, Ханнес Хапке, Коул Ховард. – СПб.: Питер, 2020. – 576 с.

3. Гольдберг, Й. Нейросетевые методы в обработке естественного языка / Й. Гольдберг; пер. с англ. А.А. Слинкина. – М.: ДМК Пресс, 2019. – 282 с.

4. Мосалев, П.М. Обзор методов нечеткого поиска текстовой информации / П.М. Мосалев // Вестник МГУП имени Ивана Федорова. – 2013. – № 2. – С. 87–91.

5. Прошина М. Современные методы обработки естественного языка: нейронные сети / М.В. Прошина // Экономика строительства. – 2022. – № 5. – С. 27–42.

6. Jurafsky, D. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition / D. Jurafsky, J.H. Martin. – 3rd ed. – Pearson, 2021.

7. Яруллин, Д.В. Интеллектуальная система управления подготовкой ИТспециалистов на основе денотативной аналитики / Д.В. Яруллин. – DOI 10.15593/2499-9873/2022.3.08 // Прикладная математика и вопросы управления / Applied Mathematics and Control Sciences. – 2022. – № 3. – С. 141–164.

8. LLM Vs Traditional NLP models: A Comparative Analysis – goML [Электронный ресурс]. – URL: https://www.goml.io/llm-vs-traditional-nlp-models/ (дата обращения: 19.05.2025).

9. Что такое скрытые модели Маркова [Электронный ресурс]. – URL: https://habr.com/ru/articles/135281/ (дата обращения: 19.05.2025).

10. The 10 Biggest Issues Facing Natural Language Processing – i2 Group [Электронный ресурс]. – URL: https://i2group.com/articles/the-10-biggest-issues-facing-natural-language-processing (дата обращения: 19.05.2025).

11. Attention Is All You Need [Электронный ресурс]. – URL: https://arxiv.org/html/1706.03762 (дата обращения: 19.05.2025).

12. Alibaba Group. Alibaba Group [Электронный ресурс]. – URL: https://www.alibaba-group.com/ (дата обращения: 19.05.2025).

13. Qwen Team. Qwen3 [Электронный ресурс]. – URL: https://qwenlm.github.io/blog/qwen3/ (дата обращения: 19.05.2025).

14. Vikhr: The Family of Open-Source Instruction-Tuned Large Language Models for Russian / А. Николич, К. Королев, С. Братчиков, Н. Компанець, А. Шельманов // arXiv preprint arXiv:2405. – 2024. 13929. https://arxiv.org/pdf/2405.13929

15. Ilya Gusev. Saiga: Russian Instruction-following Large Language Models [Электронный ресурс]. – URL: https://huggingface.co/IlyaGusev/saiga_7b_lora (дата обращения: 19.05.2025).

16. Яндекс. Официальный сайт компании [Электронный ресурс]. – URL: https://yandex.ru/company/ (дата обращения: 19.05.2025).

17. Яндекс. YandexGPT-5-Lite-8B-instruct [Электронный ресурс]. – URL: https://huggingface.co/yandex/YandexGPT-5-Lite-8B-instruct (дата обращения: 19.05.2025).

18. Yandex Cloud. Foundation Models – Документация [Электронный ресурс]. – URL: https://yandex.cloud/ru/docs/foundation-models/ (дата обращения: 19.05.2025).

19. Документация по языку программирования JavaScript [Электронный ресурс]. – URL: https://metanit.com/web/javascript/ (дата обращения: 19.05.2025).

20. Документация по React [Электронный ресурс]. – URL: https://ru.react.js.org (дата обращения: 19.05.2025).

## References

1. Afanasyeva E.A. The Role of the Author in Media Content Creation: Transformation of Professional Practices. *Journalist Yearbook*, 2015, pp. 151–154.

2. Hobson L., Khapke Kh., Khovard K. Natural Language Processing in Action. Saint Petersburg, Piter, 2020, 576 p.

3. Goldberg Y. Neural Network Methods in Natural Language Processing; translated from English by A.A. Slinkin. Moscow, DMK Press, 2019, 282 p.

4. Mosalev P.M. Review of Fuzzy Text Search Methods. *Ivan Fedorov Moscow State University of Printing Arts Herald*, 2013, no. 2, pp. 87–91.

5. Proshina M.V. Modern Methods of Natural Language Processing: Neural Networks. *Construction Economics*, 2022, no. 5, pp. 27–42.

6. Jurafsky D., Martin J.H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. 3rd ed., Pearson, 2021.

7. Iarullin D.V. Intelligent Management System for IT Specialists Training Based on Denotative Analytics. *Applied Mathematics and Control Sciences*, 2022, no. 3, pp. 141–164. DOI: 10.15593/2499-9873/2022.3.08.

8. LLM Vs Traditional NLP models: A Comparative Analysis, available at: https://www.goml.io/llm-vs-traditional-nlp-models/ (accessed: 2025-05-19).

9. What are Hidden Markov Models, available at: https://habr.com/ru/articles/135281/ (accessed: 2025-05-19).

10. The 10 Biggest Issues Facing Natural Language Processing, available at: https://i2group.com/articles/the-10-biggest-issues-facing-natural-language-processing (accessed: 2025-05-19).

11. Attention Is All You Need, available at: https://arxiv.org/html/1706.03762 (accessed: 2025-05-19).

12. Alibaba Group: official website, available at: https://www.alibabagroup.com/ (accessed: 2025-05-19).

13. Qwen Team. Qwen3, available at: https://qwenlm.github.io/blog/qwen3/ (accessed: 2025-05-19).

14. Nikolic A., Korolev K., Bratchikov S., Kompanets N., Shelmanov A. Vikhr: The Family of Open-Source Instruction-Tuned Large Language Models for Russian. *arXiv preprint arXiv:2405.13929*, 2024. , available at: https://arxiv.org/pdf/2405.13929.

15. Gusev I. Saiga: Russian Instruction-following Large Language Models, available at: https://huggingface.co/IlyaGusev/saiga_7b_lora (accessed: 2025-05-19).

16. Yandex, available at: https://yandex.ru/company/ (accessed: 2025-05-19).

17. Yandex. YandexGPT-5-Lite-8B-instruct, available at: https://huggingface.co/yandex/YandexGPT-5-Lite-8B-instruct (accessed: 2025-05-19).

18. Yandex Cloud. Foundation Models: documentation, available at: https://yandex.cloud/ru/docs/foundation-models/ (accessed: 2025-05-19).

19. JavaScript programming language documentation, available at: https://metanit.com/web/javascript/ (accessed: 2025-05-19).

20. React documentation, available at: https://ru.react.js.org (accessed: 2025-05-19).