Вестник Астраханского государственного технического университета.
Серия: Управление, вычислительная техника и информатика. 2026. № 2
Vestnik of Astrakhan State Technical University.
Series: Management, computer science and informatics. 2026. N. 2
ISSN 2072-9502 (Print), ISSN 2224-9761 (Online)

КОМПЬЮТЕРНОЕ ОБЕСПЕЧЕНИЕ И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

COMPUTER ENGINEERING AND SOFTWARE

Научная статья
УДК 347.77
https://doi.org/10.24143/2072-9502-2026-2-41-52
EDN BRECYR

Полнотекстовый русскоязычный поиск в распределенном хранилище патентной информации

***Дмитрий Михайлович Коробкин*✉,**
Сергей Алексеевич Фоменков, Артем Владимирович Бобунов

Волгоградский государственный технический университет,
Волгоград, Россия, dkorobkin80@mail.ru✉

Аннотация. Значительное увеличение количества патентных публикаций в последние годы создает сложности при проведении классического ручного анализа и поиска патентов-аналогов. Автоматизация поиска патентов-аналогов выступает ключевым инструментом для сокращения временных и финансовых издержек на этапах формирования патентной заявки и проведения патентной экспертизы. Использование технологий Big Data и распределенных систем позволяет построить эффективную архитектуру системы патентов-аналогов и повысить качество результатов патентного поиска. Теоретическая значимость работы заключается в разработке архитектуры и концепции системы полнотекстового патентного поиска на основе сравнительного анализа эффективности различных распределенных систем поиска и обработки текстовой русскоязычной информации с учетом ее морфологических и синтаксических особенностей. Практическая значимость работы состоит в реализованном программном обеспечении, включающем средства парсинга патентных документов в распределенную файловую систему, поиска с учетом особенностей естественного русского языка, а также веб-интерфейс для визуализации результатов поиска. В процессе работы использованы современные фреймворки и технологии: Apache Hadoop, Spark, Hive, Elasticsearch, PostgreSQL, ClickHouse. Elasticsearch показал наилучшие результаты и по времени отклика, и по качеству поиска (точность – 0,87, полнота – 0,82, F-мера – 0,84) для сложных запросов, отражающих специфику русского языка.

Ключевые слова: патентный поиск, распределенная система, Big Data, Elasticsearch

Благодарности: исследование выполнено при поддержке Центра цифровых научно-образовательных проектов и разработок в сфере промышленного искусственного интеллекта Ц2RED-ИИ ВолгГТУ, созданного в рамках реализации образовательных программ топ-уровня в сфере искусственного интеллекта (Соглашение № 70-2025-000756).

Для цитирования: *Коробкин Д. М., Фоменков С. А., Бобунов А. В.* Полнотекстовый русскоязычный поиск в распределенном хранилище патентной информации // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. 2026. № 2. С. 41–52. https://doi.org/10.24143/2072-9502-2026-2-41-52. EDN BRECYR.

Вестник Астраханского государственного технического университета.
Серия: Управление, вычислительная техника и информатика. 2026. № 2
ISSN 2072-9502 (Print), ISSN 2224-9761 (Online)
Компьютерное обеспечение и вычислительная техника

Original article

Full-text Russian-language search in a distributed repository of patent information

Dmitry M. Korobkin*✉, *Sergey A. Fomenkov, Artyom V. Bobunov

Volgograd State Technical University,
Volgograd, Russia, dkorobkin80@mail.ru✉

Abstract. A significant increase in the number of patent publications in recent years has created difficulties in conducting classical manual analysis and searching for patent analogues. Automation of the search for patent analogues is a key tool for reducing time and financial costs at the stages of patent application formation and patent examination. The use of Big Data technologies and distributed systems makes it possible to build an effective architecture of a system of patent analogues and improve the quality of patent search results. The theoretical significance of the work lies in the development of the architecture and concept of a full-text patent search system based on a comparative analysis of the effectiveness of various distributed systems for searching and processing textual Russian-language information, taking into account its morphological and syntactic features. The practical significance of the work lies in the implemented software, which includes tools for parsing patent documents into a distributed file system, searching taking into account the features of the natural Russian language, as well as a web interface for visualizing search results. Modern frameworks and technologies are used in the process of work: Apache Hadoop, Spark, Hive, Elasticsearch, PostgreSQL, ClickHouse. Elasticsearch showed the best results in both response time and search quality (accuracy – 0.87, completeness – 0.82, F-measure – 0.84) for complex queries reflecting the specifics of the Russian language.

Keywords: patent search, distributed system, Big Data, Elasticsearch

Acknowledgments: the study was carried out with the support of the Center for Digital Scientific and Educational Projects and Developments in the Field of Industrial Artificial Intelligence (C2RED-AI) of Volgograd State Technical University, created as part of the implementation of top-level educational programs in the field of artificial intelligence (Agreement No. 70-2025-000756).

For citation: Korobkin D. M., Fomenkov S. A., Bobunov A. V. Full-text Russian-language search in a distributed repository of patent information. *Vestnik of Astrakhan State Technical University. Series: Management, computer science and informatics*. 2026;2:41-52. (In Russ.). https://doi.org/10.24143/2072-9502-2026-2-41-52. EDN BRECYR.

Введение

В условиях стремительного технологического развития и роста объемов патентной информации, в том числе в России (рис. 1), поиск релевантных патентов [1, 2] становится важной задачей для исследовательских организаций, инженеров и специалистов по интеллектуальной собственности.

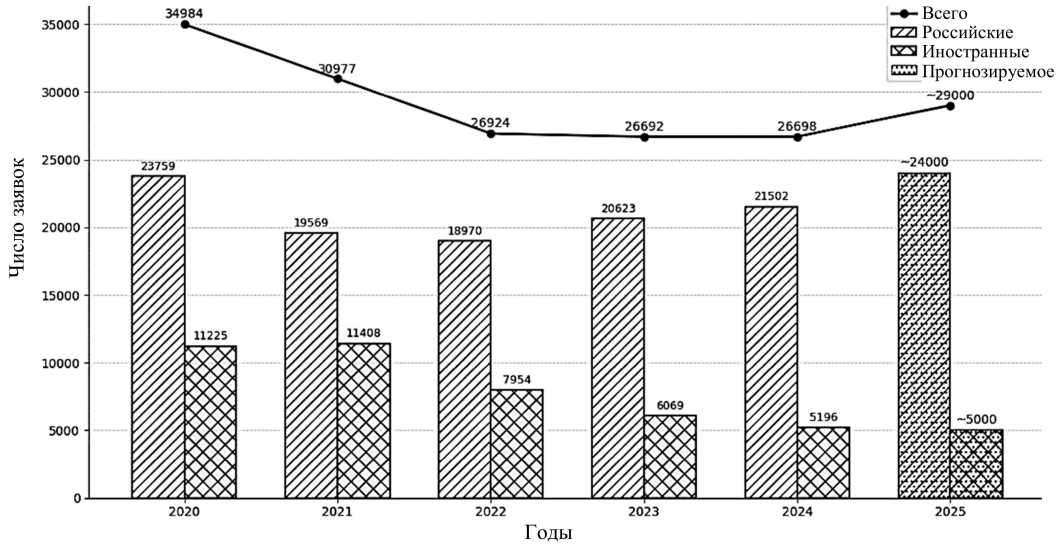


Рис. 1. Рост числа патентных заявок на изобретения в России

Fig. 1. Growth in the number of patent applications for inventions in Russia

Возможность быстро находить патенты-аналоги позволяет сократить издержки на дублирование исследований, повысить эффективность разработки новых решений и обеспечить соблюдение патентного законодательства.

Патентные данные, как правило, представлены в виде больших текстовых массивов, содержащих структурированную и неструктурированную информацию: заголовки, описания, формулы изобретений, списки авторов и владельцев. Обработка такого объема данных требует использования современных технологий распределенного хранения и обработки информации, а также специализированных поисковых систем, способных учитывать лингвистические особенности текста.

Анализ предметной области

Эффективный поиск в патентных базах данных (ФИПС Роспатента, Espacenet, USPTO, PATENTSCOPE), объем которых исчисляется сотнями миллионов документов, является критически важной задачей. Традиционные методы (булевский, TF-IDF [3], BM25 [4]) сталкиваются с проблемами лингвистической сложности патентных текстов и их огромного объема.

Рассмотрим современные подходы, объединяющие лингвистический анализ (морфологию и синтаксис) с кластерными распределенными вычислениями для обеспечения масштабируемости и точности поиска.

Классические лексические методы (BM25) обеспечивают высокую скорость, но страдают от словарного разрыва и игнорирования морфологии и синтаксиса. Современные подходы делают акцент на работе с ключевыми фразами (keyphrases), которые точнее отражают технические концепции. Эффективность демонстрируют методы, сочетающие синтаксические шаблоны (например, «прилагательное + существительное») и статистические метрики, что повышает точность извлечения по сравнению с чисто статистическими методами [5]. Парсинг зависимостей используется для построения семантических профилей патентов [6]. Для преодоления лексического разрыва применяется расширение запросов на основе языковых моделей [7] и семантические модели (Word2Vec, FastText [8]). Также современные подходы используют нейросетевые модели, такие как BERT и SciBERT, для вычисления контекстуального сходства между запросом и документом [9, 10].

Для морфологической нормализации используется стемминг [11] и, что особенно важно для языков с богатой морфологией, лемматизация с использованием анализаторов (TreeTagger, spaCy), которые позволяют объединять разные словоформы, повышая полноту поиска [12].

Синтаксический анализ применяется для выявления отношений между терминами, что критично для точного поиска. Гибридные модели, сочетающие синтаксические шаблоны и «плотный поиск» на основе Sentence-BERT, показывают значительное улучшение метрик эффективности [13].

Обработка больших объемов патентных данных требует распределенных архитектур. Классическая модель MapReduce и ее современные аналоги лежат в основе параллельной индексации и вычисления статистик [14]. Платформы Elasticsearch [15] и Apache Solr [16] обеспечивают горизонтальное масштабирование, репликацию и поддержку пользовательских анализаторов для лингвистической обработки, что делает их основой современных систем (PATENTSCOPE, PatSnap).

Фреймворк Apache Spark [17] используется для распределенного выполнения NLP-конвейеров (токенизация, лемматизация, извлечение ключевых фраз) и обучения моделей (например, распределенный Word2Vec) [18]. Архитектуры на основе Spark и библиотек векторного поиска (FAISS) обеспечивают быстрый семантический поиск по миллионам патентов [19].

Таким образом, современный поиск по патентной информации представляет собой синтез лингвистики, машинного обучения и распределенных вычислений. Перспективные направления включают интеграцию больших языковых моделей (LLM), использование мультимодальных данных и активное обучение на основе обратной связи пользователя. Кластерные вычисления являются не опцией, а обязательным фундаментом для работы с данными патентного масштаба.

Цель работы заключается разработке программного модуля распределенного поиска русскоязычных патентов-аналогов с учетом морфологических особенностей языка.

Проведен анализ возможностей различных систем управления базами данных (СУБД) и поисковых платформ в части полнотекстового поиска и поддержки русской морфологии.

На основании сравнительного анализа можно сделать вывод, что для задач полнотекстового поиска с учетом морфологических особенностей русского языка наибольшее соответствие функциональным требованиям демонстрируют Elasticsearch, Apache Solr и PostgreSQL. Все три системы поддерживают работу с русскоязычной морфологией и реализуют механизмы полнотекстового поиска (табл. 1).

Таблица 1

Table 1

Анализ систем полнотекстового поиска

Analysis of full-text search systems

Характеристика	Elasticsearch	Apache Solr	Apache Hive	PostgreSQL	ClickHouse
Морфология русского языка	Да (анализатор морфологии)	Да (стемминг, словари)	Нет	Да	Нет
Полнотекстовый поиск	Да (Lucene, распределенный)		Нет	Да (tsvector/tsquery)	Нет
Распределенная архитектура	Да (кластер с шардами/репликами)	Да (SolrCloud: шарды и реплики)	Да	Нет (только репликация или шарды на уровне приложений)	Да (кластер с шардированием)
Совместимость с HDFS	Есть (ES-Hadoop коннектор для обмена данными)	Есть (поддержка работы индексов в HDFS)	Есть	Есть (FDW для Hadoop)	Есть (движок для чтения из HDFS)

Elasticsearch и Apache Solr обладают распределенной архитектурой, что делает их максимально подходящими для обработки больших объемов патентных данных. Elasticsearch выделяется простотой настройки и встроенной поддержкой морфологии, в то время как Apache Solr требует более сложной конфигурации. PostgreSQL, несмотря на отсутствие нативной поддержки распределенности, имеет совместимость с HDFS, хорошо знакомую SQL-парадигму и встроенные средства обработки текстов.

Разработка алгоритмов

Алгоритм парсинга патентов представлен на рис. 2.

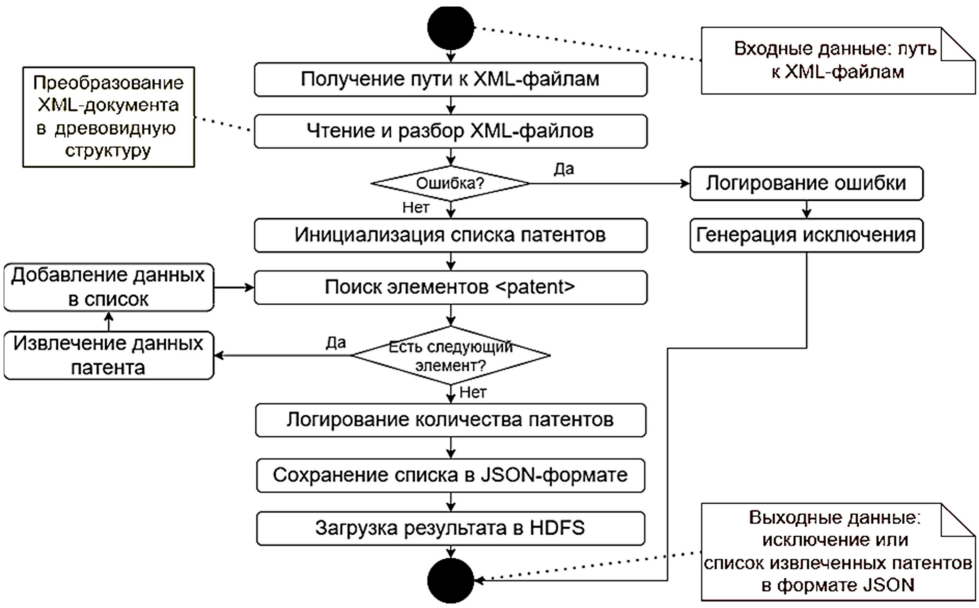


Рис. 2. Алгоритм парсинга патентов

Fig. 2. Patent parsing algorithm

Опишем последовательность действий:

1. XML-файл, выгруженный из системы Google Patents и содержащий данные о патентах, преобразуется в дерево элементов, из которого извлекается корневой элемент.

2. В дереве XML выполняется поиск всех элементов, соответствующих тегу <patent>, представляющему отдельный патент.

3. В цикле для каждого найденного элемента <patent> извлекаются значения всех необходимых

полей, таких как идентификатор, название, страна, даты, авторы, аннотация, описание и другие метаданные, которые собираются в структурированный объект.

4. Полученные структурированные объекты, содержащие патентную информацию, загружаются в распределенную файловую систему HDFS.

Данный алгоритм обеспечивает надежное извлечение данных из XML-файлов, поддерживая обработку больших объемов патентной информации.

Алгоритм тестирования (рис. 3) предназначен для оценки производительности и функциональных возможностей различных систем управления данными (Elasticsearch, Apache Solr, PostgreSQL) при полнотекстовом поиске патентов-аналогов с учетом морфологических особенностей русского языка.

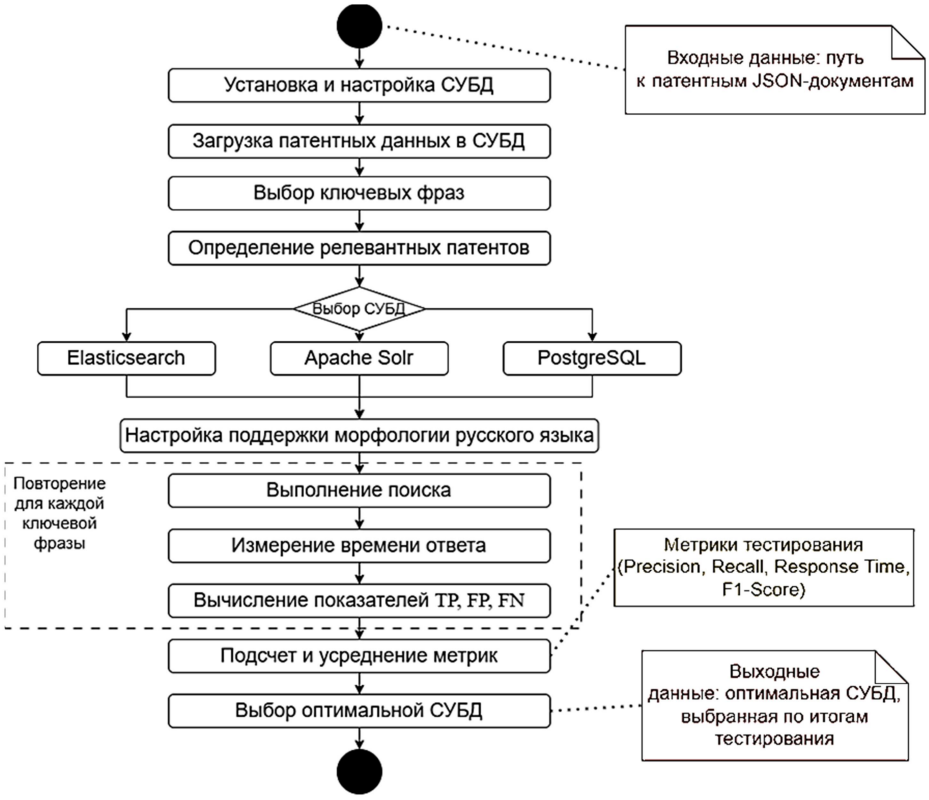


Рис. 3. Алгоритм тестирования систем

Fig. 3. System testing algorithm

Алгоритм тестирования систем включает в себя следующие шаги:

1. XML-документы патентов из системы Google Patents обрабатываются парсером, сохраняются в формате JSON и загружаются в HDFS. Далее данные импортируются в каждую систему управления данными, проводится проверка корректности загрузки и доступности записей для поиска.

2. Из массива патентов выбираются 50 ключевых фраз (из поля Concepts), автоматически сгенерированных системой Google Patents. Эти фразы используются без модификации в качестве тестовых запросов.

3. Для каждой системы выполняется настройка обработки естественного языка. Для Elasticsearch это настройка морфологического анализатора, обеспечивающего лемматизацию, для Apache Solr – добавление морфологических словарей, для PostgreSQL – настройка конфигурации russian на основе алгоритма стемминга Snowball.

4. Каждый из 50 запросов по ключевым фразам выполняется трижды на каждой системе, чтобы устранить флуктуации времени выполнения.

5. Осуществляется полнотекстовый поиск по всем текстовым полям патента (title, abstract, claims, description). Измеряется время ответа (Response Time).

6. Результаты поиска сравниваются с релевантными патентами, в которых содержалась данная ключевая фраза (в поле Concepts), что позволяет вычислить метрики эффективности поиска.

7. По совокупности метрик (эффективность и время) производится сравнительная оценка 3-х систем (на основе наилучшего баланса между полнотой, точностью и временем ответа).

Данный алгоритм обеспечивает систематическую оценку систем управления данными и поиска и поз-

воляет сделать обоснованный выбор архитектуры программного модуля поиска патентов-аналогов.

Алгоритм индексации данных (рис. 4) предназначен для создания индекса patents с заданной структурой и наполнения его данными, что обеспечивает поддержку полнотекстового поиска с учетом морфологии русского языка.

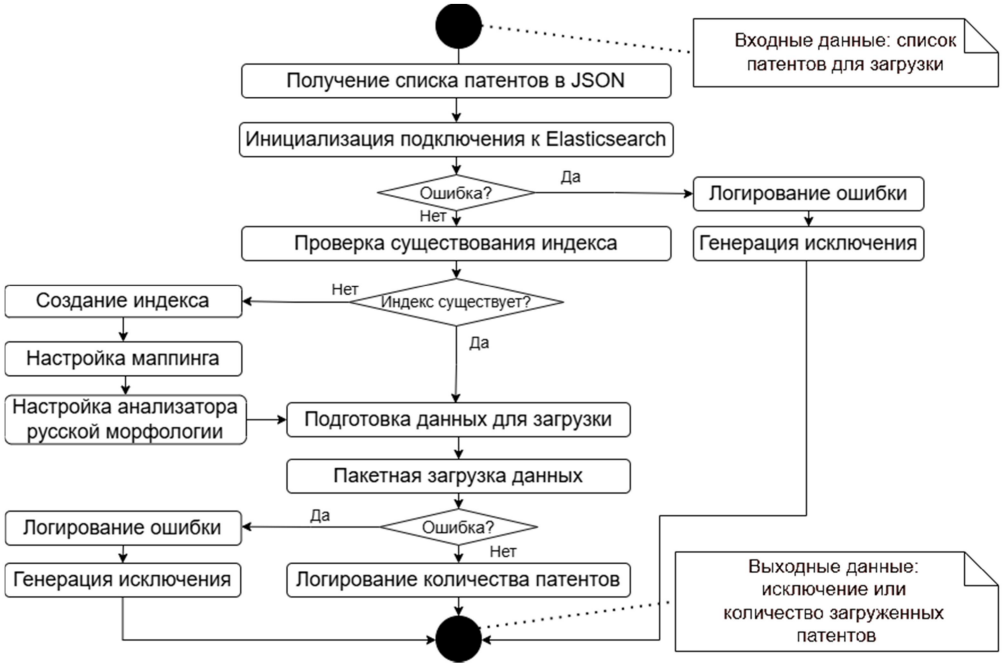


Рис. 4. Алгоритм индексации данных

Fig. 4. Data indexing algorithm

Алгоритм индексации данных включает в себя следующие шаги:

1. Устанавливается соединение с кластером Elasticsearch, создается индекс (табл. 2) с заданным маппингом, включая анализатор russian для текстовых полей.

Таблица 2

Table 2

Структура индекса Elasticsearch

Elasticsearch index structure

Патентное поле	**Тип данных**	**Описание**
id	keyword	ID патента
title	text	Название патента
country	keyword	Страна подачи патента
publication_number		Регистрационный номер патента
application_number		Номер заявки на патент
application_date	date	Дата подачи заявки
publication_date		Дата публикации
inventors	text	Авторы
assignees		Патентообладатели
abstract		Реферат
claims		Формула изобретения
description		Описание
drawings		Рисунки
similar_documents		Похожие документы
related_documents		Связанные документы
concepts		Ключевые понятия

2. Патентные данные преобразуются в формат, пригодный для пакетной загрузки в Elasticsearch, с указанием индекса и идентификатора для каждого документа.

Разработанный алгоритм обеспечивает эффективную индексацию больших объемов данных, подготавливая их для быстрого поиска.

Алгоритм поиска патентов-аналогов (рис. 5) осуществляет обработку пользовательских запросов, поиск релевантных патентов в распределенной системе управления данными Elasticsearch и визуализацию результатов в интерфейсе.

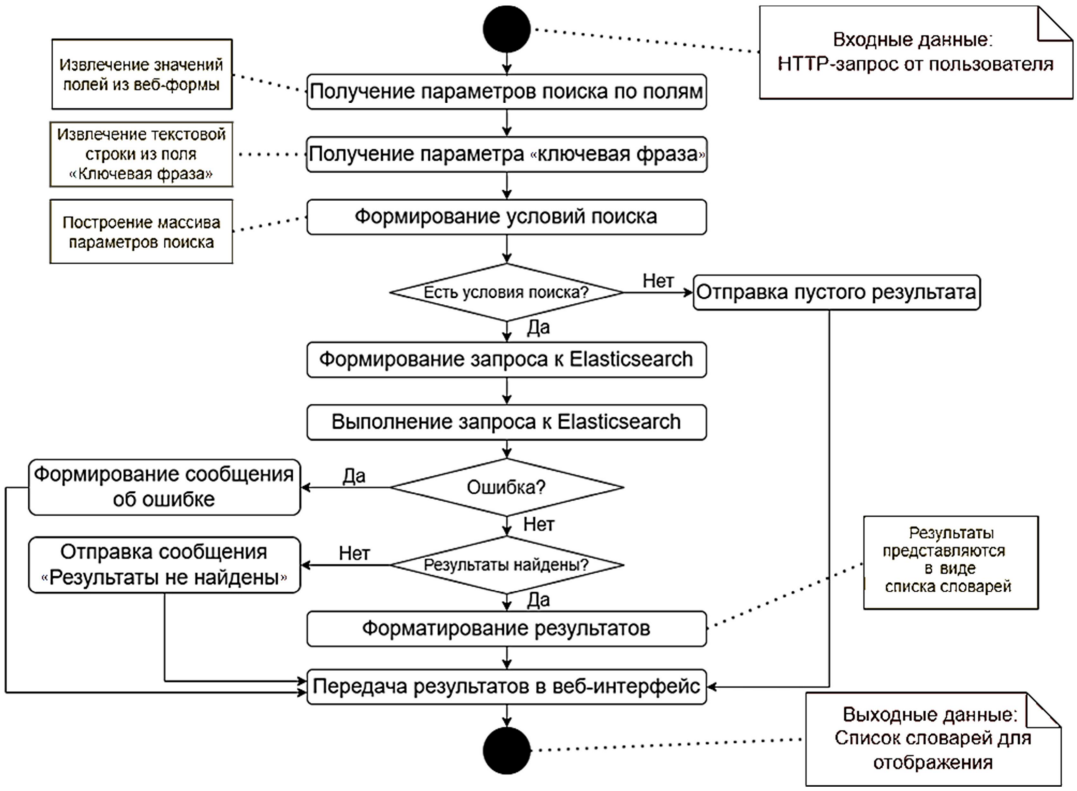


Рис. 5. Алгоритм поиска патентов-аналогов

Fig. 5. Patent analogues searching algorithm

В результате реализации алгоритма поиска патентов-аналогов была обеспечена эффективная обработка пользовательских запросов к распределенному хранилищу информации. Алгоритм учитывает особенности структуры патентных данных и специфики русского языка, что повышает точность и полноту поиска.

Сравнительное тестирование рассматриваемых систем

Для оценки производительности и функциональной пригодности различных систем управления данными к задаче полнотекстового поиска по ключевым фразам с учетом морфологических особенностей русского языка были проведены сравнительные тестирования.

Тестирование проводилось для систем Elasticsearch, Apache Solr и PostgreSQL. Перед проведением тестирования для каждой системы управления данными и поиска выполнена настройка соответствующих лингвистических компонентов.

В Elasticsearch создан индекс с заданной конфигурацией анализатора russian_analyzer, основанного на токенизации и фильтре russian_morphology, который обеспечивает лемматизацию и приведение слов к начальной форме.

В Apache Solr подключена поддержка русского языка путем добавления словаря стемминга в конфигурацию системы. Это обеспечило приведение слов к базовой форме на этапе индексирования и поиска.

В PostgreSQL использовалась встроенная конфигурация russian, основанная на алгоритме Snowball. Для каждого текстового поля сформирован индекс типа tsvector, а запросы строились с применением to_tsquery с использованием настроенного русскоязычного словаря, обеспечивающего стемминг.

Размер корпуса для тестирования составлял 10 000 патентов, полученных из поисковой системы Google Patents и загруженных в рассматриваемые системы. Все системы использовали механиз-

мы индексации и предварительную настройку анализаторов для русского языка.

Для оценки качества поиска использовались 50 ключевых фраз, полученных из патентного поля Concepts системы Google Patents. Ключевые понятия в поле Concepts формируются с использованием автоматических методов обработки естественного языка (NLP) и технологий извлечения информации из текстовых патентных данных. Выбранные фразы обеспечивают объективность и воспроизводимость тестирования, а также позволяют использовать формализованный набор ключевых понятий в качестве эталонного при определении качества поиска.

Каждый из 50 запросов выполнялся по 3 раза для каждой СУБД, после чего усреднялись показатели времени отклика (Response Time) и рассчитывались метрики Precision, Recall и F1-score. Перед расчетом метрик были установлены основные характеристики поиска:

– TP – количество релевантных патентов, которые были корректно найдены системой;

– FP – количество нерелевантных патентов, ошибочно включенных системой в результаты поиска;

– FN – количество релевантных патентов, которые не были найдены системой.

На основе указанных характеристик рассчитываются следующие ключевые метрики:

– Precision (точность поиска) показывает долю релевантных документов среди всех, найденных системой:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}};$$

– Recall (полнота поиска) отражает способность системы находить все релевантные документы:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}};$$

– F1-score (F-мера) – гармоническое среднее между точностью и полнотой, полезна при наличии дисбаланса между количеством релевантных и нерелевантных документов:

$$\text{F1-score} = 2\,\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

Результаты тестирования представлены в табл. 3.

Таблица 3

Table 3

Результаты тестирования полнотекстового поиска по ключевым фразам

Results of testing full-text search by keyphrases

Система	Response Time, мс	Precision	Recall	F1-score
Elasticsearch	131	0,87	0,82	0,84
Apache Solr	184	0,75	0,68	0,71
PostgreSQL	227	0,68	0,55	0,61

Elasticsearch показал наилучшие результаты как по времени отклика, так и по качеству поиска. Это объясняется встроенной поддержкой морфологических анализаторов для русского языка, использованием *n*-грамм и мощной архитектурой полнотекстового поиска.

Проектирование программного модуля

Программный модуль состоит из четырех основных подмодулей (рис. 6):

1. Подмодуль парсинга отвечает за обработку исходных XML-файлов с патентами. На этом этапе происходит извлечение структурированной информации, пригодной для дальнейшего хранения и индексации.

2. Подмодуль экспорта и лингвистической обработки данных сохраняет обработанные патенты в распределенной файловой системе HDFS в формате JSON, отправляет патентные данные в систему управления данными (Elasticsearch, Apache Solr, PostgreSQL) и выполняет лингвистическую обработку текстовых данных (токенизацию, стемминг, лемматизацию).

3. Подмодуль поиска патентов-аналогов реализует поиск патентов-аналогов на основе введенного пользователем запроса, содержащего ключевые фразы.

4. Подмодуль графического интерфейса предоставляет пользователю возможность ввести поисковый запрос, просматривать результаты и взаимодействовать с программой.

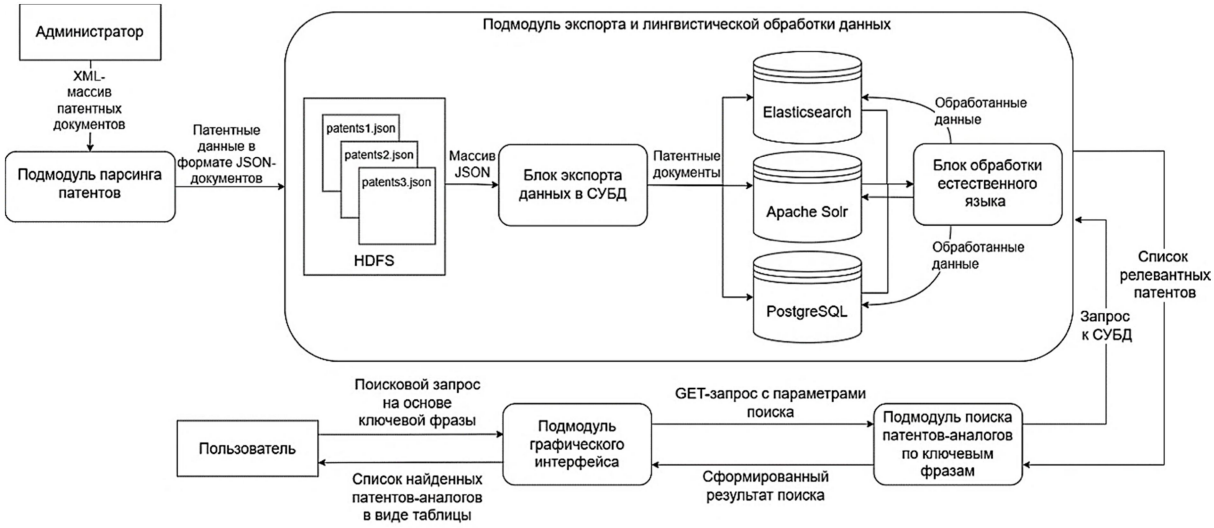


Рис. 6. Архитектура программного модуля

Fig. 6. Software architecture

Программная реализация модуля

Модуль поддерживает следующие ключевые функции:

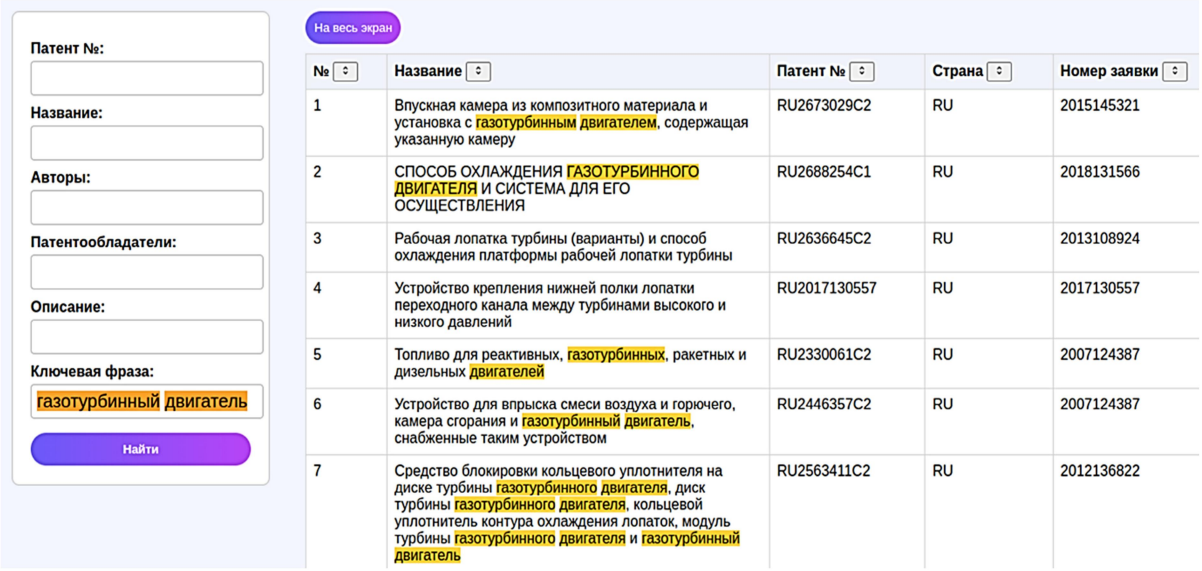
– позволяет загружать данные из XML-файлов, содержащих информацию о патентах системы Google Patents, и преобразовывать их в структурированный формат для дальнейшей работы;

– хранение данных: обеспечивает сохранение обработанных данных в распределенной файловой системе HDFS в виде JSON-файлов, обеспечивая масштабируемость и отказоустойчивость;

– индексация данных: поддерживает создание индекса в Elasticsearch с маппингом, оптимизированным для полнотекстового поиска, включая анализатор русского языка для текстовых полей, таких как название, реферат и описание;

– поиск патентов: предоставляет возможность поиска по множеству параметров, включая номер публикации, название, авторов, патентообладателей, описание и ключевые слова, реализующие возможность полнотекстового поиска;

– визуализация результатов (рис. 7) отображает найденные патенты в таблице с поддержкой сортировки по различным полям.



№	Название	Патент №	Страна	Номер заявки
1	Впускная камера из композитного материала и установка с газотурбинным двигателем, содержащая указанную камеру	RU2673029C2	RU	2015145321
2	СПОСОБ ОХЛАЖДЕНИЯ ГАЗОТУРБИННОГО ДВИГАТЕЛЯ И СИСТЕМА ДЛЯ ЕГО ОСУЩЕСТВЛЕНИЯ	RU2688254C1	RU	2018131566
3	Рабочая лопатка турбины (варианты) и способ охлаждения платформы рабочей лопатки турбины	RU2636645C2	RU	2013108924
4	Устройство крепления нижней полки лопатки переходного канала между турбинами высокого и низкого давлений	RU2017130557	RU	2017130557
5	Топливо для реактивных, газотурбинных, ракетных и дизельных двигателей	RU2330061C2	RU	2007124387
6	Устройство для впрыска смеси воздуха и горючего, камера сгорания и газотурбинный двигатель, снабженные таким устройством	RU2446357C2	RU	2007124387
7	Средство блокировки кольцевого уплотнителя на диске турбины газотурбинного двигателя, диск турбины газотурбинного двигателя, кольцевой уплотнитель контура охлаждения лопаток, модуль турбины газотурбинного двигателя и газотурбинный двигатель	RU2563411C2	RU	2012136822

Рис. 7. Интерфейс системы патентного поиска

Fig. 7. Patent search interface

Адаптивная структура таблицы с горизонтальной прокруткой позволяет эффективно работать с большими наборами результатов, а поддержка полноэкранного режима расширяет возможности анализа данных.

Заключение

Проанализированы существующие решения в области патентного поиска, такие как Google Patents, Яндекс.Патенты, ФИПС и Espacenet.

Изучены технологии распределенного хранения больших текстовых данных. Выполнена настройка кластера Hadoop и распределенной файловой системы HDFS, предназначенной для хранения исходных XML-документов патентов.

Проведен сравнительный анализ и тестирование систем распределенного хранения больших текстовых данных (Elasticsearch, Apache Solr, Apache Hive, ClickHouse и PostgreSQL) применительно к задаче полнотекстового русскоязычного поиска.

Разработан алгоритм парсинга патентных документов.

Сформирован алгоритм индексации патентных документов в системах распределенного хранения больших текстовых данных.

Разработан алгоритм поиска патентов-аналогов.

Спроектирован и программно реализован модуль поиска патентов-аналогов, интегрирующий распределенную файловую систему HDFS, XML-парсер, системы управления данными Elasticsearch, Apache Solr, PostgreSQL. Проверена его эффективность в решении задач полнотекстового поиска на основе русскоязычных ключевых фраз.

Тестирование качества полнотекстового поиска по ключевым фразам с учетом особенностей русского языка показало, что выбранный стек технологий обеспечивает высокие характеристики точности, полноты поиска, скорости отклика.

Список источников

1. Бобунов А. В., Коробкин Д. М., Фоменков С. А., Васильев С. С. Разработка программного модуля поиска патентов-аналогов // Инженер. вестн. Дона. 2022. № 11. 14 с. URL: http://www.ivdon.ru/ru/magazine/archive/n11y2022/8018 (дата обращения: 01.09. 2025).

2. Bobunov A. V., Korobkin D. M., Fomenkov S. A. Prior Art Candidate Search on Base of Semantic Tree Analysis // 2025 International Russian Smart Industry Conference (SmartIndustryCon): Proceedings. Sochi, 2025. P. 889–894. DOI 10.1109/SmartIndustryCon65166.2025.10986020.

3. Manning C. D., Raghavan P., Schütze H. Introduction to Information Retrieval. Cambridge: Cambridge University Press, 2008. 482 p.

4. Rossi J., Kanoulas E. Query Generation for Patent Retrieval with Keyword Extraction Based on Syntactic Features // Legal Knowledge and Information Systems: JURIX 2018. Amsterdam: IOS Press, 2018. P. 210–214.

5. Yang S. Y., Soo V. W. Extract conceptual graphs from plain texts in patent claims // Engineering Applications of Artificial Intelligence. 2012. V. 25. N. 4. P. 874–887. DOI 10.1016/j.engappai.2011.11.006.

6. Bai J., Song D., Bruza P., Nie J.-Y., Cao G. Query expansion using term relationships in language models for information retrieval // Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM 2005). NY: ACM, 2005. P. 688–695.

7. Robertson S. E., Zaragoza H. The Probabilistic Relevance Framework: BM25 and Beyond // Foundations and Trends in Information Retrieval. 2009. V. 3. N. 4. P. 333–389. DOI 10.1561/1500000019.

8. Bojanowski P., Grave E., Joulin A., Mikolov T. Enriching Word Vectors with Subword Information // Transactions of the Association for Computational Linguistics. 2017. V. 5. P. 135–146. DOI 10.1162/tacl_a_00051.

9. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis, 2019. P. 4171–4186. DOI 10.18653/v1/N19-1423.

10. Beltagy I., Lo K., Cohan A. SciBERT: A Pretrained Language Model for Scientific Text // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, 2019. P. 3615–3620. DOI 10.18653/v1/D19-1371.

11. Porter M. F. An algorithm for suffix stripping // Program. 1980. V. 14. N. 3. P. 130–137. DOI 10.1108/eb046814.

12. Schmid H. Probabilistic Part-of-Speech Tagging Using Decision Trees // Proceedings of the International Conference on New Methods in Language Processing. Manchester: University of Manchester, 1994. P. 44–49.

13. Stamatis V. End to End Neural Retrieval for Patent Prior Art Search // Advances in Information Retrieval. ECIR 2022. Lecture Notes in Computer Science. Cham: Springer, 2022. V. 13186. P. 537–544. DOI 10.1007/978-3-030-99739-7_66.

14. Dean J., Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters // Communications of the ACM. 2008. V. 51. N. 1. P. 107–113. DOI 10.1145/1327452.1327492.

15. Elasticsearch. Reference documentation. URL: https://www.elastic.co/guide/ (дата обращения: 24.09.2025).

16. Apache Solr Reference Guide. URL: https://solr.apache.org/guide/solr/latest/index.html (дата обращения: 24.09.2025).

17. Zaharia M., Xin R. S., Wendell P., Das T., Armbrust M., Dave A., Meng X., Rosen J., Venkataraman S., Franklin M. J., Ghodsi A., Gonzalez J., Shenker S., Stoica I. Apache Spark: A Unified Engine for Big Data Processing // Communications of the ACM. 2016. V. 59. N. 11. P. 56–65. DOI 10.1145/2934664.

18. Tang J., Wang B., Yang Y., Hu P., Zhao Y., Yan X., Gao B., Huang M., Xu P., Li W., Usadi A. K. PatentMiner: Topic-driven Patent Analysis and Mining // Proceedings of

the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2012). NY: ACM, 2012. P. 1366–1374. DOI 10.1145/2339530.2339741.

19. Johnson J., Douze M., Jégou H. Billion-scale similarity search with GPUs // IEEE Transactions on Big Data. 2019. V. 7. N. 3. P. 535–547.

References

1. Bobunov A. V., Korobkin D. M., Fomenkov S. A., Vasil'ev S. S. Razrabotka programmnogo modulya poiska patentov-analogov [Development of a software module for patent search]. *Inzhenernyj vestnik Dona*, 2022, no. 11, 14 p. Available at: http://www.ivdon.ru/ru/magazine/archive/n11y2022/8018 (accessed: 01.09. 2025).

2. Bobunov A. V., Korobkin D. M., Fomenkov S. A. Prior Art Candidate Search on Base of Semantic Tree Analysis. *2025 International Russian Smart Industry Conference (SmartIndustryCon): Proceedings.* Sochi, 2025. Pp. 889-894. DOI 10.1109/SmartIndustryCon65166.2025.10986020.

3. Manning C. D., Raghavan P., Schütze H. *Introduction to Information Retrieval*. Cambridge, Cambridge University Press, 2008. 482 p.

4. Rossi J., Kanoulas E. Query Generation for Patent Retrieval with Keyword Extraction Based on Syntactic Features. *Legal Knowledge and Information Systems: JURIX 2018*. Amsterdam, IOS Press, 2018. Pp. 210-214.

5. Yang S. Y., Soo V. W. Extract conceptual graphs from plain texts in patent claims. *Engineering Applications of Artificial Intelligence*, 2012, vol. 25, no. 4, pp. 874-887. DOI 10.1016/j.engappai.2011.11.006.

6. Bai J., Song D., Bruza P., Nie J.-Y., Cao G. Query expansion using term relationships in language models for information retrieval. *Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM 2005).* New York, ACM, 2005. Pp. 688-695.

7. Robertson S. E., Zaragoza H. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 2009, vol. 3, no. 4, pp. 333-389. DOI 10.1561/1500000019.

8. Bojanowski P., Grave E., Joulin A., Mikolov T. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 2017, vol. 5, pp. 135-146. DOI 10.1162/tacl_a_00051.

9. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Minneapolis, 2019. Pp. 4171-4186. DOI 10.18653/v1/N19-1423.

10. Beltagy I., Lo K., Cohan A. SciBERT: A Pretrained Language Model for Scientific Text. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).* Hong Kong, 2019. Pp. 3615-3620. DOI 10.18653/v1/D19-1371.

11. Porter M. F. An algorithm for suffix stripping. *Program*, 1980, vol. 14, no. 3, pp. 130-137. DOI 10.1108/eb046814.

12. Schmid H. Probabilistic Part-of-Speech Tagging Using Decision Trees. *Proceedings of the International Conference on New Methods in Language Processing*. Manchester, University of Manchester, 1994. Pp. 44-49.

13. Stamatis V. End to End Neural Retrieval for Patent Prior Art Search. *Advances in Information Retrieval. ECIR 2022. Lecture Notes in Computer Science*. Cham, Springer, 2022. Vol. 13186. Pp. 537-544. DOI 10.1007/978-3-030-99739-7_66.

14. Dean J., Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, 2008, vol. 51, no. 1, pp. 107-113. DOI 10.1145/1327452.1327492.

15. *Elasticsearch. Reference documentation*. Available at: https://www.elastic.co/guide/ (accessed: 24.09.2025).

16. *Apache Solr Reference Guide*. Available at: https://solr.apache.org/guide/solr/latest/index.html (accessed: 24.09.2025).

17. Zaharia M., Xin R. S., Wendell P., Das T., Armbrust M., Dave A., Meng X., Rosen J., Venkataraman S., Franklin M. J., Ghodsi A., Gonzalez J., Shenker S., Stoica I. Apache Spark: A Unified Engine for Big Data Processing. *Communications of the ACM*, 2016, vol. 59, no. 11, pp. 56-65. DOI 10.1145/2934664.

18. Tang J., Wang B., Yang Y., Hu P., Zhao Y., Yan X., Gao B., Huang M., Xu P., Li W., Usadi A. K. PatentMiner: Topic-driven Patent Analysis and Mining. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2012).* New York, ACM, 2012. Pp. 1366-1374. DOI 10.1145/2339530.2339741.

19. Johnson J., Douze M., Jégou H. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 2019, vol. 7, no. 3, pp. 535-547.

Статья поступила в редакцию 31.10.2025; одобрена после рецензирования 19.01.2026; принята к публикации 07.04.2026
The article was submitted 31.10.2025; approved after reviewing 19.01.2026; accepted for publication 07.04.2026

Информация об авторах / Information about the authors

Дмитрий Михайлович Коробкин – кандидат технических наук, доцент; доцент кафедры систем автоматизированного проектирования и поискового конструирования; Волгоградский государственный технический университет; dkorobkin80@mail.ru

Dmitriy M. Korobkin – Candidate of Technical Sciences, Assistant Professor; Assistant Professor of the Department of Computer-aided Design and Search Engineering Sistems; Volgograd State Technical University; dkorobkin80@mail.ru

Сергей Алексеевич Фоменков – доктор технических наук, профессор; профессор кафедры систем автоматизированного проектирования и поискового конструирования; Волгоградский государственный технический университет; saf@vstu.ru

Артем Владимирович Бобунов – аспирант кафедры систем автоматизированного проектирования и поискового конструирования; Волгоградский государственный технический университет, btema1999@yandex.ru

Sergey A. Fomenkov – Doctor of Technical Sciences, Professor; Professor of the Department of Computer-aided Design and Search Engineering Sistems; Volgograd State Technical University; saf@vstu.ru

Artyom V. Bobunov – Postgraduate Student of the Department of Computer-aided Design and Search Engineering Sistems; Volgograd State Technical University; btema1999@yandex.ru