УДК 004.65:61 + 002
https://doi.org/10.33186/1027-3689-2026-10-142-161

О разработке модели процессов управления знаниями в системе здравоохранения

Д. И. Федорова

Научный центр экспертизы средств медицинского применения Минздрава России, Москва, Российская Федерация, 2523101@gmail.com

Аннотация. Статья посвящена разработке и обоснованию подходов к построению надежной, масштабируемой и защищенной информационной системы, ядром которой выступает современная инфраструктура управления данными. Актуальность исследования определяется цифровой трансформацией организаций, постоянным увеличением объема, разнообразия и скорости поступления данных, а также возрастанием требований к непрерывности предоставления информационных услуг, достоверности сведений, производительности прикладных сервисов и защите информации. В этих условиях база данных перестает быть исключительно техническим компонентом программного комплекса и становится критически важным элементом управления операционной деятельностью, принятия управленческих решений и обеспечения устойчивости организации.

Целью статьи является систематизация архитектурных, технологических и организационных решений, обеспечивающих эффективное проектирование, эксплуатацию и развитие информационных систем обработки данных. В работе рассмотрены основные модели организации хранилищ данных, особенности применения реляционных, нереляционных и гибридных решений, а также критерии выбора между SQL-, NoSQL- и NewSQL-подходами. Проанализированы архитектуры shared-nothing и shared-everything, механизмы вертикального и горизонтального масштабирования, шардирование, репликация, партиционирование, использование кластерных конфигураций и облачных сервисов Database-as-a-Service.

Практическая значимость исследования заключается в возможности использования представленных положений при проектировании новых, модернизации действующих и сопровождении высоконагруженных корпоративных информационных систем. Материалы статьи могут быть применены специалистами по проектированию информационных систем, администраторами баз

данных, архитекторами, DevOps- и DBRE-инженерами, а также руководителями цифровых и инфраструктурных проектов при формировании требований к системам хранения, обработки и защиты информации.

Ключевые слова: модель информационных систем, база данных, управление знаниями, здравоохранение, DBRE, нормализация, транзакции, архитектура данных, нейросети, искусственный интеллект

Для цитирования: Федорова Д. И. О разработке модели процессов управления знаниями в системе здравоохранения // Научные и технические библиотеки. 2026. № 10. С. 142–161. https://doi.org/10.33186/1027-3689-2026-10-142-161

UDC 004.65:61 + 002
https://doi.org/10.33186/1027-3689-2026-10-142-161

On developing the model of knowledge management processes in the public health system

Darya I. Fedorova

Scientific Center for Expert Evaluation of Medical Products,
Ministry of Health of the Russian Federation, Moscow, Russian Federation,
2523101@gmail.com

Abstract. The author substantiates the approaches toward building reliable, scalable and safe information system and its core of modern data management infrastructure. The study urgency is determined by the institutional digital transformation, substantially increasing data volume, diversity and arrival rate? as well as by growing requirements of information service continuity, information accuracy, applied services performance and information security. In this context, the databases cease to be a software complex component, though a significant one, and become critical element of operation management, decision making, organizational sustainability.

The purpose of the study is to classify architectural, technological and organizational solutions for effecting design, operation and development of data

processing information systems. The author examines the key models of data warehouses, application of relational, non-relational, and hybrid solutions, and criteria of choice between SQL-, NoSQL- and NewSQL- approaches. She analyzes shared-nothing and shared-everything architecture, vertical and horizontal scaling mechanisms, sharding, replication, partitioning, use of cluster configurations and Database-as-a-Service cloud services.

The practical value of the study resides in the possibility to use the above provisions in design, modernization and support of highload corporate information systems. The findings may be useful for information system designers, database administrators, architects and DevOps- and DBRE-engineers, as well as the managers of digital and infrastructural projects for formulating requirements to data storage, processing and security.

Keywords: information system model, database, knowledge management, public health, DBRE, normalization, transaction, data architecture, neural network, artificial intelligence

Cite: Fedorova D. I. On developing the model of knowledge management processes in the public health system // Scientific and technical libraries. 2026. No. 10, pp. 142–161. https://doi.org/10.33186/1027-3689-2026-10-142-161

Введение

В современном обществе развитие информационных систем и экспоненциальный рост массивов данных ставят вопрос о достоверности, полноте и актуальности информационных баз. Информационные ресурсы, аккумулирующие научные и практические знания, являются ядром развития медицинской науки и практики. Это диктует необходимость разработки методологических основ формирования единой среды общих данных – коллективного инструмента управления, хранения и обмена знаниями.

Целью данной работы является анализ архитектурных и технологических аспектов построения среды баз данных как основы для модели процессов управления знаниями в здравоохранении. В статье рассматриваются уровни абстракции данных, методы обеспечения надеж-

ности (DBRE) и стратегии масштабирования, критически важные для медицинских систем высокого уровня доступности.

Проектирование новых, модернизация действующих и сопровождение высоконагруженных корпоративных информационных систем

Российская система здравоохранения применяет указанные модели информационных систем при создании интегрированных баз данных, используемых в медицинских учреждениях. В частности, базы данных используются для сопровождения научных разработок в области здравоохранения на различных стадиях инновационного цикла и содействия поиску индустриальных партнеров.

Информационная система представляет собой интегрированную совокупность взаимосвязанных компонентов, предназначенную для сбора, хранения, обработки и передачи информации. Разнообразие практических применений обусловливает существование информационных систем, различающихся по целевому назначению. В частности, системы могут быть ориентированы на управление знаниями в здравоохранении, мониторинг патентных материалов научных и образовательных учреждений, подведомственных органам исполнительной власти.

Одной из фундаментальных задач проектирования информационной системы являются обеспечение надежного доступа к полной и актуальной информации, создание условий для эффективной работы пользователей и организации достоверного анализа данных, позволяющего принимать обоснованные управленческие и научные решения. Информация формируется в базы данных.

Базы данных наполняются, обновляются и актуализируются посредством использования формата электронных таблиц (.xls/.xlsx) с применением процедур автоматической выгрузки данных из унифицированных реестров и справочников Министерства здравоохранения России (ЕГРЮЛ, системы С1 и прочих). Информация, загружаемая в базы данных, поступает из множественных источников: официальных писем, результатов взаимодействия с заинтересованными организациями и иных каналов.

Среда базы данных представляет собой комплексную многоуровневую инфраструктуру, объединяющую технические, программные

и организационные компоненты. Эффективность функционирования медицинских информационных систем (МИС) напрямую зависит от грамотного проектирования этой среды.

В условиях экспоненциального роста объемов данных и повышения требований к надежности, доступности и производительности информационных систем проектирование и эксплуатация среды баз данных становится критическим фактором успеха организаций [**1**].

Среда базы данных как система может быть декомпозирована на несколько взаимосвязанных уровней. На техническом уровне среда включает экземпляры баз данных (database instances) – работающие копии систем управления базами данных, обрабатывающие запросы и управляющие данными; схемы (schemas) – логические структуры организации объектов базы данных, включающие таблицы, представления, хранимые процедуры и функции; узлы (nodes) – физические или виртуальные серверы, на которых развернуты компоненты БД; а также прикладные программные интерфейсы (API), обеспечивающие взаимодействие приложений с базой данных [Там же].

Функциональная типология сред базы данных предполагает разделение на продуктивную среду (production environment), среду обеспечения качества (QA environment) и среду разработки (development environment). Продуктивная среда представляет собой основную рабочую инфраструктуру, обслуживающую реальных пользователей и бизнес-процессы, что обусловливает критические требования к показателям надежности, доступности (availability) и производительности. Среда обеспечения качества предназначена для верификации функциональных и нефункциональных требований перед развертыванием в продуктивную среду, включая функциональное и нагрузочное тестирование, проверку CRUD-операций и валидацию производительности. Среда разработки обеспечивает изолированное пространство для экспериментов, создания пользовательских функций и отладки программных компонентов [Там же].

Выбор архитектурной парадигмы среды базы данных определяется требованиями к масштабируемости, доступности и согласованности данных. Архитектура shared-nothing характеризуется полной независимостью узлов, не разделяющих общих ресурсов, что обеспечивает

линейное горизонтальное масштабирование путем добавления новых узлов в кластер. Данная архитектура реализована в таких системах, как Greenplum, Vertica и платформах обработки больших данных (BigData), однако сопряжена с усложнением управления распределенными транзакциями и необходимостью реализации механизмов секционирования данных (sharding) [**3**].

Альтернативная парадигма shared-everything предполагает совместное пользование узлами общего хранилища данных, что упрощает обеспечение согласованности и поддержку распределенных транзакций. Системы Oracle и DB2 с поддержкой кластеризации демонстрируют возможности параллелизма как между запросами (inter-query parallelism), так и внутри отдельного запроса (intra-query parallelism). Однако данная архитектура характеризуется ограниченной масштабируемостью, обусловленной пропускной способностью сети хранения данных и конкуренцией за общие ресурсы [Там же].

На техническом уровне среда декомпозируется на следующие элементы:

1. Экземпляры баз данных (Instances): работающие копии СУБД, обрабатывающие запросы.

2. Схемы (Schemas): логические структуры, включающие таблицы, представления и хранимые процедуры.

3. Узлы (Nodes): физические или виртуальные вычислительные мощности.

Функциональная типология предполагает разделение на продуктивную среду (Production), среду обеспечения качества (QA) и среду разработки (Development). Для медицинских систем критически важна изоляция этих сред для предотвращения компрометации чувствительных данных пациентов при тестировании и разработке. Таким примером может служить ЕГИСЗ – Федеральная информационная система с детальной структурой данных для электронной медицинской карты, включая объекты учета (Пациент → Случай обслуживания → Посещение → Диагнозы → Назначения). Такая система содержит в себе:

– назначение и масштаб системы (5,3 млн пациентов, 42 млн документов);

– архитектуру (федеральный центр, региональные сегменты);

– 15 обязательных разделов ЭМК (таблица со всеми разделами);

– модель данных с полной SQL-схемой (3NF-нормализация);

– безопасность (DAC, MAC, Bell-LaPadula модель, шифрование AES-256);

– требования интеграции медорганизаций (приказы 530-н, 90-н, 555).

Реальный сценарий: полный путь пациента от регистрации до синхронизации. Выбор архитектурной парадигмы определяется требованиями к масштабируемости. Архитектура shared-nothing (независимые узлы) обеспечивает линейное масштабирование, характерное для аналитических систем (Greenplum, Vertica), но усложняет управление распределенными транзакциями. Архитектура shared-everything (общее хранилище), реализованная в Oracle RAC, упрощает согласованность данных, но имеет пределы масштабируемости из-за конкуренции за ресурсы.

Проектирование и нормализация данных

Проектирование структуры базы данных для моделей информационных систем в системе здравоохранения осуществляется итеративно на концептуальном, логическом и физическом уровнях.

На этапе логического проектирования применяется теория нормализации для минимизации избыточности и предотвращения аномалий обновления:

1. Первая нормальная форма (1NF): атомарность атрибутов.

2. Вторая нормальная форма (2NF): устранение частичных функциональных зависимостей.

3. Третья нормальная форма (3NF): исключение транзитивных зависимостей.

4. Нормальная форма Бойса – Кодда (BCNF) и высшие формы (4NF, 5NF): устранение многозначных зависимостей и зависимостей соединения.

Однако в аналитических подсистемах здравоохранения (хранилища данных) допустима контролируемая денормализация для оптимизации сложных аналитических запросов. Примером такой системы служит РМИС – Региональная система (программное обеспечение регионального сегмента единой государственной информационной системы в сфере здравоохранения, которое обеспечивает автоматизацию

основных процессов медицинского учреждения, а также обмен информацией между региональными и федеральными информационными ресурсами в лечебно-диагностическом процессе) с иерархической моделью, примерами SLO для мониторинга надежности (99,9% доступность) и использованием Infrastructure as Code (Terraform), содержащая в себе:

– назначение и место в архитектуре;

– основные функции (интеграция, маршрутизация, финансы, аналитика);

– архитектуру системы (слои: интеграция, хранение, приложения, безопасность);

– Service Level Objectives с конкретными числами (99,9% доступность, < 500 мс ответ);

– Infrastructure as Code (Terraform-конфигурация для AWS с RDS, Multi-AZ, KMS-шифрование);

– управление миграциями БД (Flyway с SQL-примерами версионирования).

Инжиниринг надежности (DBRE)

Внедрение методологии Database Reliability Engineering (DBRE) позволяет перейти от реактивного администрирования к проактивному обеспечению надежности.

DBRE представляет собой современную методологию управления жизненным циклом баз данных, адаптирующую принципы Site Reliability Engineering для обеспечения надежности, доступности и производительности систем баз данных. Фундаментальным понятием DBRE являются Service Level Objectives (SLO) – количественные целевые показатели качества обслуживания, такие как доступность (например, 99,9% = 43,8 минуты простоя в месяц), время отклика запросов (например, 95-й перцентиль < 100 мс) и пропускная способность [**2**].

Метрики Mean Time To Recovery (MTTR) и Mean Time Between Failures (MTBF) количественно характеризуют надежность системы. MTTR представляет среднее время восстановления после инцидента и является ключевым показателем эффективности процессов реагирования на сбои.

Ключевые метрики надежности включают:

– MTTR: среднее время восстановления после инцидента,

– MTBF: среднее время между отказами.

Для обеспечения консистентности конфигураций применяется концепция «неизменяемой инфраструктуры» (immutable infrastructure) и подход «Инфраструктура как код» (IaC) с использованием инструментов Terraform или Ansible. Версионирование схем баз данных (Liquibase, Flyway) в системах контроля версий является обязательным стандартом для сопровождения эволюции медицинских моделей данных.

Практическое применение нормализации требует баланса между устранением избыточности и обеспечением производительности запросов. Денормализация – контролируемое введение избыточности – может быть оправдана для оптимизации частых аналитических запросов, особенно в хранилищах данных (data warehouses) [**5**].

Физическое проектирование трансформирует логическую схему в физическую модель, оптимизированную для конкретной СУБД. Критическим аспектом является проектирование индексов – вспомогательных структур данных, ускоряющих поиск записей. B-tree индексы обеспечивают эффективный поиск по диапазонам значений и применяются для большинства запросов. Хеш-индексы оптимизированы для точного поиска по ключу, но не поддерживают диапазонные запросы. Bitmap-индексы эффективны для колонок с низкой кардинальностью в аналитических системах [**4, 5**].

Секционирование (partitioning) таблиц обеспечивает горизонтальное или вертикальное разделение данных для повышения производительности и упрощения администрирования. Горизонтальное секционирование распределяет строки по нескольким физическим сегментам на основе значений ключа секционирования, что позволяет выполнять параллельную обработку и изолировать исторические данные. Вертикальное секционирование разделяет таблицу по колонкам, размещая часто запрашиваемые атрибуты отдельно от редко используемых [**5**].

Концепция неизменяемой инфраструктуры (immutable infrastructure) предполагает создание полностью настроенных образов серверов, которые при необходимости изменений полностью заменяются новыми образами, а не обновляются на месте. Данный подход, реали-

зуемый с использованием инструментов Packer от Hashicorp для создания Amazon Machine Images (AMI), обеспечивает консистентность конфигурации, упрощает откат изменений и повышает предсказуемость поведения системы [**2**].

Service Discovery – механизм автоматического обнаружения сервисов в динамической инфраструктуре – критичен для управления распределенными системами. Инструменты Consul, Zookeeper и Etcd обеспечивают координацию узлов, управление конфигурацией и выполнение проверок работоспособности (health checks). При развертывании кластера MySQL, например, каждый узел регистрируется в Consul с использованием уникального идентификатора шарда (shardid), что позволяет клиентским приложениям динамически обнаруживать доступные серверы баз данных [Там же].

Версионирование схемы базы данных в системах контроля версий (VCS) является критическим аспектом DBRE-практик. Хранение DDL-скриптов (Data Definition Language) в репозиториях Git или Bitbucket обеспечивает отслеживание истории изменений структуры, проведение code review и откат проблемных изменений. Инструменты управления миграциями базы данных – Liquibase, Flyway, Alembic – автоматизируют применение изменений схемы и поддерживают как прямые (forward), так и обратные (rollback) миграции [Там же].

Проблема миграций с нулевым временем простоя (zero-downtime migrations) актуальна для крупных таблиц в высоконагруженных системах. Стратегии включают поэтапное добавление колонок с последующим заполнением данных в фоновом режиме, использование теневых (shadow) таблиц и онлайн-алгоритмов изменения структуры, таких как gh-ost от GitHub [Там же].

Транзакционная семантика и согласованность

Целостность данных обеспечивается соблюдением требований ACID (Atomicity, Consistency, Isolation, Durability).

Атомарность гарантирует выполнение транзакции целиком или ее полный откат.

Согласованность обеспечивает перевод базы данных из одного корректного состояния в другое с проверкой ограничений целостности.

Изолированность регулирует влияние параллельных транзакций друг на друга. Уровни изоляции варьируются от Read Uncommitted до Serializable.

В распределенных системах вступает в силу теорема CAP (Consistency, Availability, Partition tolerance), постулирующая невозможность одновременного достижения согласованности, доступности и устойчивости к разделению. Реляционные системы традиционно выбирают CA-сегмент, тогда как NoSQL-решения могут жертвовать немедленной согласованностью ради доступности (AP-системы), реализуя модель BASE (Basically Available, Soft state, Eventually consistent).

Масштабирование и безопасность

Для обработки растущих объемов данных применяются стратегии масштабирования:

– вертикальное (Scale-up): наращивание ресурсов одного узла,

– горизонтальное (Scale-out): секционирование (sharding) данных по множеству узлов.

Безопасность данных обеспечивается сочетанием дискреционного (DAC) и мандатного (MAC) управления доступом. В контексте обработки персональных данных и врачебной тайны критически важно применение шифрования (at rest, in transit), аудита действий и маскирования данных в непродуктивных средах.

Облачные технологии вводят модель Database-as-a-Service (DBaaS), при которой управление инфраструктурой баз данных делегируется провайдеру облачных услуг. Сервисы Amazon RDS, Aurora, Google Cloud SQL обеспечивают автоматизацию резервного копирования, обновлений и масштабирования, снижая административные издержки при сохранении высоких показателей доступности. Применение концепции Infrastructure as Code (IaC) с использованием инструментов Terraform, Ansible, Chef и Puppet позволяет версионировать конфигурацию инфраструктуры и обеспечивать повторяемость развертывания [**1**].

Подсистема хранения данных оказывает определяющее влияние на производительные характеристики среды базы данных. Базы данных в оперативной памяти (In-Memory Databases, IMDB) размещают данные в RAM, обеспечивая минимальные задержки доступа, что критично для приложений реального времени. Системы SAP HANA, Redis

и Memcached демонстрируют на порядок более высокую производительность по сравнению с дисковыми системами, однако требуют реализации механизмов персистентности данных для обеспечения долговечности (durability) [**1**, **3**].

Колоночное хранение (columnar storage) оптимизировано для аналитических запросов, характерных для OLAP-систем (On-Line Analytical Processing), обеспечивая эффективное сжатие данных и ускорение агрегационных операций. Данная технология реализована в системах Amazon Redshift, Google BigQuery и Apache Parquet [**1**].

Проектирование структуры базы данных осуществляется итеративно на трех уровнях абстракции: концептуальном, логическом и физическом. На концептуальном уровне формируется модель предметной области, независимая от конкретной системы управления базами данных. ER-диаграммы (Entity-Relationship) обеспечивают визуализацию сущностей, их атрибутов и взаимосвязей, определяя кардинальность отношений: один к одному (1:1), один ко многим (1:M) и многие ко многим (M:N). Унифицированный язык моделирования (UML) расширяет возможности концептуального моделирования за счет поддержки объектно-ориентированных конструкций, включая наследование, инкапсуляцию и полиморфизм [**4, 1**].

Логическое проектирование преобразует концептуальную модель в реляционную схему с применением теории нормализации для минимизации избыточности данных и предотвращения аномалий обновления. Первая нормальная форма (1NF) требует атомарности значений атрибутов и устранения повторяющихся групп. Вторая нормальная форма (2NF) устраняет частичные функциональные зависимости неключевых атрибутов от составных ключей. Третья нормальная форма (3NF) исключает транзитивные зависимости, при которых неключевой атрибут зависит от другого неключевого атрибута [**5**].

Нормальная форма Бойса – Кодда (BCNF) представляет собой усиленную версию 3NF, требующую, чтобы каждая функциональная зависимость имела в качестве детерминанта потенциальный ключ. Четвертая нормальная форма (4NF) устраняет многозначные зависимости (multivalued dependencies), возникающие при независимых множественных отношениях. Пятая нормальная форма (5NF, Project-Join

Normal Form) декомпозирует отношения, имеющие зависимости соединения, обеспечивая максимальную степень нормализации [**5**, **3**].

План выполнения запроса (query execution plan) представляет стратегию, выбранную оптимизатором СУБД для выполнения SQL-запроса. Различают прогнозируемый план (estimated plan), строящийся на основе статистики распределения данных без фактического выполнения, и актуальный план (actual plan), отражающий реально выполненные операции. Анализ плана выполнения идентифицирует операции сканирования таблиц (table scan), поиска по индексу (index seek), соединения таблиц (nested loop join, hash join, merge join), сортировки и агрегации [**4**].

Стоимостная модель оптимизатора оценивает каждую операцию в единицах вычислительных ресурсов. Операции с высокой стоимостью представляют узкие места, требующие оптимизации путем добавления индексов, переформулирования запроса или денормализации схемы. Hints (подсказки) позволяют явно управлять поведением оптимизатора: NOLOCK отключает блокировки на чтение (эквивалентно READ UNCOMMITTED), ROWLOCK / PAGLOCK / TABLOCK управляют уровнем гранулярности блокировок, HOLDLOCK удерживает блокировки до завершения транзакции [Там же].

Covering index – индекс, включающий все колонки, необходимые для выполнения запроса, – устраняет необходимость обращения к основной таблице, существенно ускоряя выполнение. Техника index-only scan использует только индекс для получения результатов [**5**].

Теорема CAP, сформулированная Eric Brewer, утверждает, что распределенная система может одновременно гарантировать не более двух из трех свойств: согласованность (consistency) – все узлы видят одинаковые данные, доступность (availability) – каждый запрос получает ответ, устойчивость к разделению (partition tolerance) – система функционирует при сетевых разделениях [**3**].

Традиционные реляционные системы выбирают согласованность и доступность, жертвуя устойчивостью к разделению, что приемлемо для локальных сетей с высокой надежностью. Распределенные NoSQL-системы обычно выбирают доступность и устойчивость к разделению, жертвуя немедленной согласованностью [Там же].

Модель BASE представляет альтернативу ACID для распределенных систем. Базовая доступность (basically available) означает, что система доступна большую часть времени, допуская временные сбои отдельных компонентов. Мягкое состояние (soft state) подразумевает, что состояние системы может изменяться даже без поступления запросов, в процессе достижения согласованности. Финальная согласованность (eventually consistent) гарантирует, что при отсутствии новых обновлений все реплики в конечном итоге придут к единому состоянию [**3**].

Настраиваемая согласованность (tunable consistency) в системах типа Cassandra позволяет для каждого запроса указывать требуемый уровень согласованности через параметры кворума: R – количество реплик, от которых требуется ответ при чтении, W – количество реплик, подтверждающих запись, N – общее количество реплик. Условие $R + W > N$ гарантирует строгую согласованность, так как множества реплик для чтения и записи пересекаются [Там же].

Непрерывная интеграция и доставка для баз данных (Database CI/CD) распространяет DevOps-практики на управление схемой и данными. Автоматизированное тестирование миграций в CI-пайплайне (Jenkins, GitLab CI, CircleCI) выявляет проблемы до развертывания в продуктивную среду [Там же].

Blue-green deployment поддерживает два идентичных продуктивных окружения (blue и green), одно из которых активно обслуживает пользователей, а другое бездействует. Новая версия развертывается в неактивном окружении, тщательно тестируется, затем трафик мгновенно переключается на новое окружение. При обнаружении проблем возможен мгновенный откат переключением трафика обратно [Там же].

Канареечное развертывание (canary deployment) постепенно направляет возрастающий процент пользовательского трафика на новую версию, непрерывно мониторя ключевые метрики. При отсутствии деградации метрик доля трафика увеличивается до 100%; при выявлении проблем развертывание приостанавливается и выполняется откат [Там же].

Использование нейросетей

Основная сложность анализа патентной документации в сфере здравоохранения связана с обработкой больших объемов разнородных данных, содержащих сложную специализированную информацию в различных видах: текстовые описания, формулы изобретения, структурные формулы химических соединений, перечни последовательностей аминокислот, штаммы микроорганизмов, культуры клеток растений и животных, генетические и белковые конструкции, чертежи, схемы, рисунки и другие графические иллюстрации. Для повышения эффективности анализа в работе применялись инструменты искусственного интеллекта (нейросетей), в том числе методы обработки естественного языка и векторного представления документов для автоматизированного анализа текстовых частей патентов и их тематической кластеризации.

Использование нейросетей может быть ориентировано на решение нескольких ключевых задач. Во-первых, для поддержки разработанной тематической классификации патентного массива по 14 технологическим направлениям применялись модели, автоматически присваивающие патентам тематические метки и выявляющие мультидисциплинарные разработки. Во-вторых, нейросети использовались для обнаружения аномалий и противоречий в атрибутах патентных записей, что способствовало повышению качества данных и реализации системы метрик достоверности, актуальности и полноты. В-третьих, в рамках информационно-системного подхода к выявлению «упущенных» патентов были сформированы критерии и предикторы риска потери правовой охраны, на основе которых нейросети позволяли выделять приоритетные объекты для мониторинга.

В рамках концепции среды общих данных сервисы нейросети рассматриваются как интегрированный слой интеллектуального анализа, подключенный к отраслевой патентной базе знаний. Результаты работы нейросетей (тематические метки, индикаторы риска, оценки качества данных) сохраняются в виде дополнительных атрибутов и могут использоваться в аналитических запросах, при формировании отчетности и поддержке управленческих и научно-технических решений. Это обеспечивает сокращение сроков первичной тематической

разметки и подготовки аналитических выборок, снижение трудоемкости ручной проверки данных и повышение воспроизводимости патентного анализа при работе с крупными массивами информации.

При этом применение нейросетей требует учета ряда ограничений. Результаты автоматизированного анализа подлежат экспертной валидации на входе и выходе данных, иногда зачастую необходима и промежуточная экспертная валидация, а качество моделей определяется полнотой и репрезентативностью задействованных обучающих выборок. Для использования ИИ-оценок в управленческих процессах здравоохранения важна их интерпретируемость, а также регулярное обновление и дообучение моделей по мере расширения патентного массива и изменения технологического ландшафта.

Роль промптов и требования к их формулированию

При использовании инструментов искусственного интеллекта в задачах анализа патентной и клинико-аналитической информации критическую роль играет корректная и формализованная постановка задания (промпта), выполняемая экспертами предметной области. Грамотно сформулированный промпт задает границы анализа (тип источников, временной период, предметная область), фиксирует используемую терминологию и формат результирующих данных, а также явно ограничивает допустимые выводы только теми фактами, которые присутствуют в исходных документах. Для минимизации ошибок и ложных обобщений в промпт включаются требования к консервативному поведению ИИ-системы: разделение фактических данных и интерпретаций, обязательное указание источников по ключевым выводам, обработка неопределенности через фиксацию отсутствия информации, а не ее домысливание, а также согласование результатов с существующими классификаторами и моделями данных отраслевой базы знаний. Проверка выходных данных осуществляется в научном стиле и включает логико-семантический контроль согласованности атрибутов, сопоставление с первичными документами по выборке, оценку воспроизводимости результатов при вариации формулировок промптов и использование разработанных метрик достоверности, актуальности и полноты данных, что обеспечивает интеграцию ИИ-генерируемых результатов в общую систему управления знаниями.

Выводы

Формирование, внедрение и использование среды базы данных представляет собой сложный многоуровневый процесс, требующий интеграции теоретических основ управления данными, инженерных практик надежности и организационных процессов. Проведенный анализ позволяет сформулировать следующие выводы.

Архитектурные решения среды базы данных определяют фундаментальные характеристики масштабируемости, доступности и согласованности. Выбор между архитектурами shared-nothing и shared-everything, между реляционными и NoSQL-системами, между локальным развертыванием и облачными сервисами DBaaS должен основываться на систематическом анализе функциональных и нефункциональных требований с учетом ограничений CAP-теоремы [**1** – **3**].

Перспективы развития среды баз данных связаны с дальнейшей автоматизацией операций через применение машинного обучения для оптимизации запросов и самонастройки параметров производительности, распространением NewSQL-систем, сочетающих масштабируемость NoSQL с транзакционными гарантиями реляционных СУБД, внедрением квантово-устойчивого шифрования для защиты от будущих угроз квантовых вычислений, а также развитием технологий обработки данных в памяти (in-memory) с энергонезависимой памятью (persistent memory) для устранения разрыва между производительностью RAM и долговечностью дисковых систем хранения.

Нейросети рассматриваются как интегрированный слой интеллектуального анализа, подключенный к отраслевой патентной базе знаний. Результаты автоматизированного анализа подлежат экспертной валидации на входе и выходе данных, иногда зачастую необходима и промежуточная экспертная валидация, а качество моделей определяется полнотой и репрезентативностью используемых обучающих выборок.

Таким образом, эффективное формирование, внедрение и использование среды базы данных требует системного подхода, интегрирующего архитектурное проектирование, инженерные практики надежности, управление транзакционной семантикой, стратегии масштабирования и многоуровневую безопасность в рамках единой мето-

дологии, ориентированной на достижение определенных количественных показателей качества обслуживания.

Формирование среды базы данных для управления знаниями в здравоохранении требует системной интеграции теоретических основ управления данными и инженерных практик надежности. Проведенный анализ показывает, что выбор архитектурных решений (SQL vs NoSQL, shared-nothing vs shared-disk) должен базироваться на строгом анализе функциональных требований и ограничений CAP-теоремы. Применение методологии DBRE, строгих правил нормализации и многоуровневой защиты информации создает необходимый фундамент для построения надежных и масштабируемых моделей управления знаниями в медицинской сфере.

Информационные системы являются критически важным инструментом в современном здравоохранении, позволяя организовывать, обрабатывать и передавать информацию в соответствии с установленными требованиями качества и безопасности. Понимание особенностей различных моделей информационных систем и их применения в практике здравоохранения способствует эффективному использованию информационных ресурсов и принятию обоснованных управленческих решений.

Список источников

1. **Шрайберг Я. Л.** Современные тенденции развития цифровизации общества: научно-образовательная и библиотечно-информационная среда : монография. Москва : ИНФРА-М, 2024. 663 с. С. 568–659. ISBN 978-5-16-020046-0, 978-5-16-112590-8. DOI 10.12737/2155873.

2. **Учебно-методические** материалы : сб. учеб. программ / науч. рук. Я. Л. Шрайберг; авт. программ: А. О. Адамьянц, Ф. С. Воройский, А. И. Земсков [и др.] / Моск. гос. ун-т культуры и искусств. Каф. информ. технологий и электрон. б-к. Москва, 2003. Вып. 2. 66 с.

3. **Таненбаум Э.** Компьютерные сети : учебное пособие / пер. с англ. 4-е изд. Санкт-Петербург : Питер, 2008. 991 с. С. 75–77.

4. **Когаловский М. Р.** Энциклопедия технологий баз данных. Москва : Финансы и статистика, 2002. 800 с.

5. **ГОСТ Р 59793-2021.** Национальный стандарт Российской Федерации. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания. Москва : Стандартинформ, 2021.

6. **ГОСТ Р 59792-2021.** Национальный стандарт Российской Федерации. Информационные технологии. Комплекс стандартов на автоматизированные системы. Виды испытаний автоматизированных систем. Москва : Стандартинформ, 2021.

7. **ГОСТ Р 59853-2021.** Национальный стандарт Российской Федерации. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения. Москва : Стандартинформ, 2021.

8. **ГОСТ 34.201-2020.** Межгосударственный стандарт. Информационные технологии. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем. Москва : Стандартинформ, 2020.

9. **DAMA-DMBOK:** Свод знаний по управлению данными. 2-е изд. Москва : Технологии, 2020.

10. **Кузнецов С. Д.** Базы данных. Москва, 2020.

11. **Кэмпбелл Л., Мейджорс Ч.** Базы данных. Инжиниринг надежности / пер. с англ. Москва : ДМК Пресс, 2020. 304 с. ISBN 978-5-4461-1310-1.

12. **Волк В. К.** Базы данных: проектирование, программирование, управление и администрирование. Санкт-Петербург : Лань, 2020. 244 с. ISBN 978-5-8114-4189-1.

13. **Куликов С. С.** Реляционные базы данных в примерах: практическое пособие для программистов и тестировщиков. Минск : Четыре четверти, 2020.

References

1. **Shrai`berg Ia. L.** Sovremenny`e tendentcii razvitiia tcifrovizatcii obshchestva: nauchno-obrazovatel`naia i bibliotechno-informatcionnaia sreda : monografiia. Moskva : INFRA-M, 2024. 663 s. S. 568–659. ISBN 978-5-16-020046-0, 978-5-16-112590-8. DOI 10.12737/2155873.

2. **Uchebno-metodicheskie** materialy` : sb. ucheb. programm / nauch. ruk. Ia. L. Shrai`berg; avt. programm: A. O. Adam`iantc, F. S. Voroi`skii`, A. I. Zemskov [i dr.] / Mosk. gos. un-t kul`tury` i iskusstv. Kaf. inform. tekhnologii` i e`lektron. b-k. Moskva, 2003. Vy`p. 2. 66 **s**.

3. **Tanenbaum E`.** Komp`iuterny`e seti : uchebnoe posobie / per. s angl. 4-e izd. Sankt-Peterburg : Peter, 2008. 991 s. S. 75–77.

4. **Kogalovskii` M. R. E`**ntciclopediia tekhnologii` baz danny`kh. Moskva : Finansy` i statistika, 2002. 800 s.

5. **GOST R 59793-2021.** Natcional`ny`i` standart Rossii`skoi` Federatcii. Informatcionny`e tekhnologii. Kompleks standartov na avtomatizirovanny`e sistemy`. Avtomatizirovanny`e sistemy`. Stadii sozdaniia. Moskva : Standartinform, 2021.

6. **GOST R 59792-2021**. Natcional`ny`i` standart Rossii`skoi` Federatcii. Informatcionny`e tekhnologii. Kompleks standartov na avtomatizirovanny`e sistemy`. Vidy` ispy`tanii` avtomatizirovanny`kh sistem. Moskva : Standartinform, 2021.

7. **GOST R 59853-2021.** Natcional`ny`i` standart Rossii`skoi` Federatcii. Informatcionny`e tekhnologii. Kompleks standartov na avtomatizirovanny`e sistemy`. Avtomatizirovanny`e sistemy`. Terminy` i opredeleniia. Moskva : Standartinform, 2021.

8. **GOST 34.201-2020.** Mezhgosudarstvenny`i` standart. Informatcionny`e tekhnologii. Kompleks standartov na avtomatizirovanny`e sistemy`. Vidy`, komplektnost` i oboznachenie dokumentov pri sozdanii avtomatizirovanny`kh sistem. Moskva : Standartinform, 2020.

9. **DAMA-DMBOK:** Svod znanii` po upravleniiu danny`mi. 2-e izd. Moskva : Tekhnologii, 2020.

10. **Kuznetcov S. D.** Bazy` danny`kh. Moskva, 2020.

11. **Ke`mpbell L., Mei`jours Ch.** Bazy` danny`kh. Inzhiniring nadezhnosti / per. s angl. Moskva : DMK Press, 2020. 304 s. ISBN 978-5-4461-1310-1.

12. **Volk V. K.** Bazy` danny`kh: proektirovanie, programmirovanie, upravlenie i administrirovanie. Sankt-Peterburg : Lan`, 2020. 244 s**.** ISBN 978-5-8114-4189-1.

13. **Kulikov S. S.** Reliatcionny`e bazy` danny`kh v primerakh: prakticheskoe posobie dlia programmistov i testirovshchikov. Minsk : Chety`re chetverti, 2020.

Информация об авторе / Author

Федорова Дарья Ильинична – первый заместитель директора Центра трансфера медицинских технологий Научного центра экспертизы средств медицинского применения Минздрава России, Москва, Российская Федерация

2523101@gmail.com

Darya I. Fedorova – First Deputy Director, Center of Medical Technologies Transfer, Scientific Center for Expert Evaluation of Medical Products, Ministry of Health of the Russian Federation, Moscow, Russian Federation

2523101@gmail.com