DOI:10.33266/1024-6177-2025-70-4-46-54

В.Г. Барчуков, А.А. Болотов, Е.Н. Жирнов, А.С. Самойлов, С.М. Шинкарев, И.Б. Ушаков, И.К. Теснов, А.С. Галузин, Д.А. Кудинова, В.Ю. Лизунов

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ ОБЕСПЕЧЕНИЯ РАДИАЦИОННОЙ БЕЗОПАСНОСТИ ПРИ ВЫВОДЕ ИЗ ЭКСПЛУАТАЦИИ РАДИАЦИОННЫХ И ЯДЕРНЫХ ОБЪЕКТОВ

Федеральный медицинский биофизический центр им. А.И. Бурназяна ФМБА России, Москва

Контактное лицо: Александр Александрович Болотов, e-mail: abolotov@bk.ru

РЕФЕРАТ

Актуальность: Вывод из эксплуатации радиационно-опасных объектов (РОО) – сложный процесс, требующий соблюдения законодательных и нормативных требований. Для их выполнения и обеспечения безопасности необходимы эффективное управление документацией и постоянное обучение персонала. Искусственный интеллект (ИИ) может значительно упростить и автоматизировать обработку и управление документацией, снижая нагрузку на персонал и минимизируя ошибки. Он также помогает в соблюдении нормативных требований, автоматически отслеживая изменения и обеспечивая соответствие стандартам. Дополнительно ИИ способен анализировать большие объемы данных, выявлять потенциальные риски и предлагать оптимальные решения на основе предсказательной аналитики.
Цель: Создание сервиса на основе ИИ, способного поддерживать полноценный и информированный диалог по вопросам вывода из эксплуатации РОО. Для достижения этой цели была выбрана модель обработки естественного языка (NLP) Keras. Для обучения модели был создан набор данных, включающий 5 основных нормативных документов по радиационной безопасности. Документы были разделены на отдельные контексты, по которым эксперты задавали вопросы и формулировали ответы на них. Всего было введено 429 контекстов и по ним было задано 6512 вопросов и ответов.
Результат: Модель тестировалась в специально разработанном приложении, аналогичном ChatGPT, которое помогает специалистам находить ответы на вопросы, возникающие в процессе вывода из эксплуатации РОО. Кроме того, была реализована функция динамического обновления базы знаний, позволяющая оперативно учитывать изменения в нормативной документации.
Разработанная система продемонстрировала высокую точность в ответах на вопросы, связанные с нормативными аспектами вывода из эксплуатации. Алгоритмы машинного обучения, созданные на основе нашего набора данных для обработки и интерпретации текста, оказались эффективными в распознавании и обработке пользовательских запросов. Система была протестирована в различных сценариях, включая внутренние оценки модели Keras и тестовые вопросы, не вошедшие в набор данных для обучения модели. Полученные результаты подтвердили перспективность использования технологий искусственного интеллекта в управлении процессами вывода РОО из эксплуатации. Дополнительно были проведены испытания на реальных данных, что позволило выявить ключевые зоны для дальнейшего улучшения системы и расширения её функциональных возможностей.

Ключевые слова: *искусственный интеллект, вывод из эксплуатации радиационных и ядерных объектов, нормативные документы, радиационная безопасность, языковые модели, Keras*

Для цитирования: Барчуков В.Г., Болотов А.А., Жирнов Е.Н., Самойлов А.С., Шинкарев С.М., Ушаков И.Б., Теснов И.К., Галузин А.С., Кудинова Д.А., Лизунов В.Ю. Использование технологий искусственного интеллекта для обеспечения радиационной безопасности при выводе из эксплуатации радиационных и ядерных объектов // Медицинская радиология и радиационная безопасность. 2025. Т. 70. № 4. С. 46–54. DOI:10.33266/1024-6177-2025-70-4-46-54

DOI:10.33266/1024-6177-2025-70-4-46-54

V.G. Barchukov, A.A. Bolotov, Y.N. Zhirnov, A.S. Samoylov, S.M. Shinkarev, I.B. Ushakov, I.K. Tesnov, A.S. Galuzin, D.A. Kudinova, V.U. Lizunov

Using Artificial Intelligence Technologies for Radiation Protection during Decommissioning of Radiation and Nuclear Facilities

A.I. Burnazyan Federal Medical Biophysical Center, Moscow, Russia

ABSTRACT

Background: The decommissioning of radiation-hazardous facilities (RHF) is a complex process that requires compliance with legislative and regulatory requirements. Effective document management and continuous personnel training are essential for ensuring safety and regulatory adherence. Artificial Intelligence (AI) can significantly simplify and automate documentation processing and management, reducing staff workload and minimizing errors. Additionally, AI helps ensure regulatory compliance by automatically tracking changes and maintaining adherence to standards. Furthermore, AI can analyze large volumes of data, identify potential risks, and propose optimal solutions based on predictive analytics.
Purpose: To develop an AI-based service capable of supporting a comprehensive and informed dialogue on RHF decommissioning. To achieve this, we selected a natural language processing (NLP) model based on Keras. A dataset was created for training the model, consisting of five key regulatory documents on radiation safety. The documents were divided into separate contexts, with experts formulating corresponding questions and answers. In total, 429 contexts were processed, and 6,405 questions and answers were generated.
Results: The model was tested in a specially developed application similar to ChatGPT, designed to help specialists find answers to questions arising during the decommissioning process. Additionally, a dynamic knowledge base update feature was implemented, allowing for real-time adjustments to regulatory documentation changes. The developed system demonstrated high accuracy in answering questions related to regulatory aspects of decommissioning. Machine learning algorithms trained on our dataset for text processing and interpretation

proved effective in recognizing and handling user queries. The system was tested in various scenarios, including internal Keras model evaluations and test questions not included in the training dataset.
Conclusion: The obtained results confirmed the potential of AI technologies in managing RHF decommissioning processes. Furthermore, tests conducted on real-world data helped identify key areas for further system improvement and functional expansion.

Keywords: *Artificial intelligence, decommissioning of radiation and nuclear facilities, regulatory documents, radiation safety, Natural Language Processing (NLP), Keras*

For citation: Barchukov VG, Bolotov AA, Zhirnov YN, Samoylov AS, Shinkarev SM, Ushakov IB, Tesnov IK, Galuzin AS, Kudinova DA, Lizunov VU. Using Artificial Intelligence Technologies for Radiation Protection during Decommissioning of Radiation and Nuclear Facilities. Medical Radiology and Radiation Safety. 2025;70(4):46–54. (In Russian). DOI:10.33266/1024-6177-2025-70-4-46-54

Введение

Обеспечение безопасного вывода из эксплуатации объектов ядерного оружейного комплекса требует строгого контроля радиационной безопасности. Устаревание и износ таких объектов увеличивают экологические и санитарно-гигиенические риски, поэтому важно оценивать уровни и дозовые нагрузки, а также минимизировать вероятность загрязнения. Этот процесс регулируется сложной системой стандартов и нормативных документов, а точная работа с ними необходима для защиты людей, окружающей среды и успешного завершения операций по выводу объектов из эксплуатации.

Основная сложность работы с нормативными документами в сфере обеспечения радиационной безопасности заключается в их большом объеме, высокой сложности и разнообразии. Регулирующая база включает множество законов, стандартов, постановлений и методических рекомендаций, которые регулярно обновляются и дополняются.

Поиск необходимой информации в этом массиве документов требует значительных временных затрат, особенно если учитывать все изменения и дополнения. Сложный язык и специализированная терминология делают процесс интерпретации сложным и требуют высокой квалификации специалистов. Нормативные акты, разработанные различными ведомствами, часто разрознены и несогласованы, что осложняет их систематизацию и понимание. Такая фрагментированность может приводить к дублированию или противоречиям в требованиях.

Постоянные изменения нормативной базы создают дополнительные трудности с ее своевременной актуализацией: задержки в обновлении информации могут привести к несоответствию принимаемых решений текущим требованиям, что особенно критично для обеспечения радиационной безопасности. Кроме того, отсутствие единого стандарта представления данных приводит к разнообразию форматов, что усложняет их интеграцию в автоматизированные системы.

Недостаток удобных инструментов для взаимодействия между специалистами и регуляторами затрудняет обсуждение и применение нормативных требований, увеличивая риск ошибок. Эти проблемы значительно усложняют работу с нормативной документацией, повышая риски для здоровья людей и радиационную нагрузку на окружающую среду. Решение таких проблем возможно через внедрение современных технологий, таких как автоматизация поиска, анализ и актуализация нормативной базы с использованием инструментов искусственного интеллекта.

Диалоговые системы, подобные ChatGPT, могут значительно упростить работу с нормативной базой и повысить уровень радиационной безопасности при выводе из эксплуатации радиационно-опасных объектов. Чат-боты способны оперативно находить нужные нормативные документы или их конкретные разделы по запросам специалистов, экономя время и ресурсы. Например, запрос “Какие требования предъявляются к утилизации радиоактивных отходов?” будет обработан моментально, и система предоставит точный ответ. Благодаря контекстному поиску такие системы могут учитывать предыдущие вопросы и предоставлять более релевантные ответы.

Кроме того, диалоговые системы способны интерпретировать сложный юридический и технический язык нормативных документов, превращая его в более доступные для понимания формулировки. Например, система может преобразовать тяжёлые для восприятия требования в простые пошаговые рекомендации, которые помогут специалистам правильно выполнить все действия. Чат-боты могут также отслеживать изменения в нормативной базе и уведомлять специалистов о нововведениях, предоставляя краткое резюме и ссылки на обновлённые документы. Таким образом, специалисты всегда будут в курсе последних изменений и смогут своевременно актуализировать свои действия.

Диалоговые системы могут выявлять противоречия и дублирование в различных нормативных документах, что облегчит их согласование. Например, если два документа содержат противоположные требования, чат-бот может предупредить об этом и предложить пути разрешения конфликта. Также такие системы могут генерировать аналитические отчёты, которые помогут специалистам быстро разобраться в ситуации и принять необходимые меры.

Кроме того, диалоговые системы могут быть полезны в обучении специалистов. Виртуальные консультанты смогут отвечать на вопросы в реальном времени, например: “Какие меры радиационной защиты должны быть применены при демонтаже оборудования?” и обучать их через сценарии на основе нормативных требований. Также чат-боты могут поддерживать специалистов в принятии решений, анализируя предоставленные данные (например, радиационные риски) и предоставляя рекомендации на основе нормативных документов. Они могут также проверять, соответствуют ли текущие действия специалиста требованиям регуляторов, что значительно повысит уровень радиационной безопасности.

Чат-боты также могут интегрироваться с другими системами, такими как автоматизированные системы управления документооборотом или аналитическими платформами, обеспечивая быстрый доступ к актуальной информации. Они могут извлекать текст из сканов нормативных документов с помощью технологии OCR (оптическое распознавание текста) и отвечать на вопросы по их содержанию, обеспечивая ещё большую гибкость и универсальность.

Таким образом, диалоговые системы, подобные ChatGPT, являются мощным инструментом для работы с нормативной документацией, обеспечивая оперативность, точность и эффективность. С их помощью специалисты смогут снизить нагрузку на решения рутинных задач, повысить качество принятия решений и обеспечить более высокий уровень радиационной безопасности при выводе радиационно-опасного объекта из эксплуатации.

Разработка системы на основе технологий ИИ для работы с нормативными документами: особенности и ключевые шаги

Целью данной работы было создание сервиса, построенного на основе ИИ и способного поддерживать полноценный и осознанный диалог по вопросам вывода из эксплуатации РОО.

Для решения поставленной цели на первом этапе необходимо было решить следующие задачи:

1. Выбрать языковую модель (NLP) и алгоритмы машинного обучения для информационно-справочной поддержки пользователей при выводе из эксплуатации РОО.
2. Собрать и подготовить данные (Dataset) для последующего обучения модели ИИ.
3. Обучить выбранную модель и оценить ее на отдельной тестовой выборке.
4. Создать прототип приложения, подобного ChatGPT, для ответов на возникающие при ВЭ РОО вопросы.

Языковые (NLP) модели

Обработка естественного языка – это область ИИ, которая обеспечивает связь между людьми и машинами, позволяя компьютерам понимать и генерировать человеческий язык. В современном мире одним из самых актуальных направлений в области обработки естественного языка является создание систем вопрос–ответ (QA). Такие системы находят применение в чат-ботах, поисковых системах и виртуальных ассистентах. Основой их работы являются языковые модели, которые позволяют анализировать и понимать естественный язык. Для задачи вопрос–ответ нами были учтены несколько ключевых факторов:

– Предметная область – ограниченного характера, касающаяся проблем радиационно-гигиенического обеспечения безопасности при выводе из эксплуатации радиационно опасных обьектов (ВЭ РОО).
– Пользователи системы – специалисты в области радиационной безопасности и вывода из эксплуатации РОО.
– Высокая точность и полнота ответов, которые модель может генерировать на основе заданного вопроса и контекста.
– Необходимость привязки ответа к нормативным и регламентирующим документам.
– Вычислительная мощность оборудования и время, необходимое для обучения модели.
– Количество данных, доступных для обучения и тестирования модели.
– Необходимость работы с многозначными вопросами или специфическая тематика вопросов и ответов.
– Возможность использования текстовой, графической, аудио и видеоинформации в качестве ответа на текстовый запрос пользователей.
– Легкость интеграции модели в существующую инфраструктуру и ее масштабируемость.

Существует несколько известных языковых моделей: BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pre-trained Transformer), RoBERTa (A Robustly Optimized BERT Pretraining Approach), Keras.

BERT, разработанная Google, является одной из самых популярных моделей для задач вопрос–ответ [1]. Основное преимущество BERT заключается в ее двунаправленном обучении, что позволяет модели учитывать контекст слова с обеих сторон (слева и справа). Модель обладает высоким качеством предсказаний, особенно для задач понимания текста. Она требует значительных вычислительных ресурсов, особенно при использовании больших вариантов моделей, таких как BERT-Large. Хорошо работает с большими объемами данных [2]. Отлично подходит для задач, требующих глубокого понимания контекста. Широкая поддержка и наличие готовых библиотек, таких как Hugging Face Transformers, облегчают интеграцию [3].

GPT, разработанный OpenAI, является мощной генеративной моделью, которая может использоваться для создания связных и осмысленных ответов на вопросы [4]. У этой модели качество предсказаний очень высокое, особенно для генерации текста. Последние версии модели, такие как GPT-3 и GPT-4, требуют значительных вычислительных ресурсов [5]. Для обучения и предобучения этой модели требуются большие объемы данных. Модель идеальна для задач, где необходима генерация текста, но может быть адаптирована для точных задач вопрос–ответ. Для этой модели доступны API от OpenAI, что упрощает интеграцию в приложения.

RoBERTa – это улучшенная версия BERT с оптимизированными методами предобучения [6]. Модель дает отличное качество предсказаний**,** благодаря улучшенным методам предобучения и требует значительных ресурсов, аналогично BERT. Она работает лучше с большими объемами данных. Прекрасно подходит для задач понимания текста и вопрос–ответ. Хорошо поддерживается, легко интегрируется через существующие NLP библиотеки.

Keras – это высокоуровневая библиотека глубокого обучения, написанная на языке Python [7]. Она позволяет создать модель с «нуля», используя только собственный набор данных. Обучение модели достаточно быстрое с высокой точностью ответов на запросы. С момента интеграции Keras в TensorFlow (open-source machine learning framework developed by Google) использование Keras стало еще более удобным. Модели Keras могут использовать все преимущества TensorFlow, включая возможности работы на GPU (графических процессорах видеокарт) и распределенное обучение [8]. Keras имеет активное сообщество пользователей и разработчиков, что способствует быстрому решению проблем и обмену опытом. Существует множество ресурсов, таких как документация, учебники и примеры кода, которые помогают быстро освоиться с библиотекой.

Особенностями модели Keras являются удобство для пользователя благодаря согласованному и простому API, которое минимизирует количество действий, необходимых для решения задач. Модульность Keras позволяет работать с последовательностью или графом автономных, полностью сконфигурированных модулей, таких как нейронные слои, функции ошибки, оптимизаторы, функции активации и схемы регуляризации, которые можно комбинировать для создания модели. Расширяемость платформы обеспечивает простоту добавления новых классов, модулей и функций, что делает её подходящей для исследований. Все программное обеспечение модели написано на языке программирования Python, что делает код компактным и легко читаемым.

Для создания сервиса, способного поддерживать полноценный и осознанный диалог по вопросам ВЭ РОО, мы использовали языковую модель Keras. Для оценки ее пользовательских характеристик было проведено пробное тестирование модели Keras на различных наборах данных и получены следующие результаты:

– Позволяет создать модель с «нуля» – из DATASET.
– Использует пакет TENSORFLOW, Keras для обучения.
– Словарь – формируется из DATASET.

– Обучение – достаточно быстрое, можно использовать офисные компьютеры.
– Высокая точность ответов на запросы, т.к. используется QA-модель.
– Позволяет «настраивать» структуру модели и DATASET под конкретные требования.
– Допускает использование текстовой, графической, аудио и видеоинформации в качестве ответа на текстовый запрос пользователей.
– Позволяет разбивать (осуществлять декомпозицию) DATASET модели и их отлаживать отдельно.
– Имеет простую структуру модели.
– Программное обеспечение (ПО) обучения работает на нижнем уровне библиотек – позволяет настраивать параметры модели (в т.ч. параметры нейросети, критерии точности, потерь и др.);
– ПО для получения ответов на запросы работает достаточно быстро на обычных офисных компьютерах.
– Модель также обеспечивает решение задач классификации (диагностика, прогноз).
– Документация по подходам модели достаточно полна, имеется множество примеров использования такой модели на языке Python.

Модель Keras оказалась наиболее удобной и гибкой для разработки, что позволило эффективно адаптировать ее под наши специфические данные и требования.

Файловый состав модели Keras состоит из 4 файлов следующего вида:
– **model1.h5** – обученная языковая Модель;
– **classes.pkl** – классы (группы ответов) Модели (в бинарном виде);
– **words.pkl** – словарь Модели (в бинарном виде);
– **dataset.json** – DATASET Модели.

Данные (Dataset) для обучения модели ИИ

Для обучения системы ИИ требуется собрать большой объем документов, связанных с ВЭ РОО. Это могут быть нормативные акты, федеральные законы, регламенты и другие. На этом этапе также важно провести аннотацию данных, выделяя ключевые понятия и связи. Данные являются основой любой системы ИИ. Качество и количество данных сильно влияют на производительность модели ИИ.

Данные должны быть точны, полны и последовательны. Плохое качество данных может привести к значительному уменьшению точности модели и прогноза. Предварительная обработка данных включает очистку и преобразование сырых данных в пригодный для использования формат.

В соответствии с этими требованиями была проведена работа по максимальному учету всех регулирующих вывод из эксплуатации документов с учетом специфики РОО и этапов их эксплуатации. Была составлена подборка федеральных законов, санитарных и федеральных норм и правил, применимых к ВЭ РОО. Число таких документов с дополнительными материалами по радиационной безопасности – около 200. Работа специалистов по ВЭ РОО с этими документами требует больших временных затрат на поиск нужного документа, его раздела и требуемой информации для конкретного этапа ВЭ.

Основными документами, регламентирующими деятельность по ВЭ РОО в России, являются:
– Федеральные законы
– Указы Президента Российской Федерации
– Технические регламенты
– Постановления и Распоряжения Правительства Российской Федерации
– Нормативные документы Госкорпорации «Росатом» и подведомственных дивизионов
– Документы Роспотребнадзора и ФМБА России
– Методические документы ФМБА России
– Документы Ростехнадзора
– Международные документы.

Представление документа для последующего машинного обучения должно учитывать несколько ключевых аспектов, позволяющих обеспечить успешный анализ и интерпретацию данных. Документ должен быть представлен в машиночитаемом формате и структурирован таким образом, чтобы ИИ мог легко извлекать информацию. Перед представлением документа необходимо провести предварительную обработку данных: исключить ненужную информацию, привести текст к единому регистру и удалить лишние пробелы и специальные символы, устранить неоднозначности и исправить грамматические ошибки. Для успешного обучения моделей ИИ часто требуется разметка данных с присвоением тегов текстовым сегментам для классификации.

Как отмечалось выше, наиболее эффективными для обучения моделей ИИ считаются датасеты формата вопрос–ответ (QA). Во-первых, они точно имитируют реальные сценарии, где пользователи задают вопросы и ожидают конкретных ответов. Такая прямая применимость улучшает практическую полезность моделей ИИ в различных областях.

Во-вторых, они предоставляют структурированный подход к обучению путем связывания вопросов с конкретными ответами. Такие датасеты предлагают четкие, однозначные примеры для обучения. Структурированный формат помогает моделям ИИ лучше усваивать ассоциации между вопросами и их правильными ответами по сравнению с менее структурированными датасетами.

В-третьих, QA-датасеты могут включать контекстную информацию, позволяя моделям ИИ учиться извлекать релевантную информацию из больших текстов или документов. Такое понимание контекста значительно повышает производительность моделей.

В-четвертых, такие датасеты могут включать разнообразные типы вопросов и форматы ответов, что помогает моделям ИИ учиться справляться с неоднозначностью и вариабельностью в языке. Это разнообразие тренирует модели быть более гибкими и устойчивыми в понимании и генерировании точных ответов в различных контекстах и стилях вопросов.

И наконец, они способствует улучшению возможностей моделей по пониманию естественного языка. Такие датасеты учат модели интерпретировать вопросы, понимать нюансы и предоставлять контекстуально правильные ответы, что важно для более сложных приложений ИИ, таких как чат-боты и автоматизированные системы поддержки клиентов.

Датасет вопрос–ответ (QA) для ИИ обычно состоит из пар вопросов и соответствующих им ответов. Хороший датасет включает разнообразные типы вопросов для повышения устойчивости модели ИИ. Дополнительно в качестве ответов целесообразно использовать не только текстовые данные, но и другие виды информации: цифровые рисунки, сканы, изображения, а также аудио и видеоинформацию. Кроме того, важным является возможность доступа пользователей к исходным документам, причем с позиционированием к информации в документе, представленной в ответе.

На основании вышеизложенного был выбран подход по созданию датасета вопрос–ответ с исходными структурированными документами, хранящимися в реляционной базе данных (БД) PostgreSQL. Реляционные

базы данных структурируются по нескольким таблицам, которые можно объединить с помощью первичного или внешнего ключа. Эти уникальные идентификаторы создают различные отношения, существующие между таблицами.

Для создания датасета было разработано тестовое веб-приложение, позволяющее пользователям наполнять БД необходимыми документами. На начальном этапе было выбрано 5 нормативных документов:

– Федеральный закон «О радиационной безопасности населения» от 09.01.1996 № 3-ФЗ.
– Федеральный закон «Об использовании атомной энергии» от 21.11.1995 № 170-ФЗ.
– Нормы радиационной безопасности НРБ-99/2009.
– Основные санитарные правила обеспечения радиационной безопасности (ОСПОРБ-99/2010).
– Санитарные правила СП 2.6.1.23-05 «Обеспечение радиационной безопасности при выводе из эксплуатации комплектующего предприятия» (СП ВЭКП-05).

Каждый документ с помощью эксперта в области радиационной безопасности был разбит на отдельные смысловые контексты, которые были сохранены в базе данных с использованием разработанного для этих целей веб-приложения «Заря». Затем группа экспертов в области радиационной безопасности (12 человек) задавала один или несколько вопросов по каждому контексту документа и формулировала возможные ответы на эти вопросы. Общее количество выделенных контекстов составило 429. По ним было задано 6378 вопросов и ответов.

Вводимые данные сохранялись в БД, откуда их можно было сгруппировать по запросу в единый файл формата JSON для передачи в программу машинного обучения модели. Формат файла соответствовал требованиям обучающей модели Keras.

Перед обучением данные должны редактироваться с целью удаления повторяющихся вопросов и ответов, а также удаления HTML-тегов, специальных символов и т. д.

При выборе данных формируются два файла JSON. Один используется для обучения модели, другой – для ее тестирования. Файл, используемый для обучения, содержит примерно 85 % вопросов и ответов по каждому контексту из общего количества. Оставшиеся 15 % вопросов и ответов помещается в файл для тестирования.

Обучение и тестирование системы

Общая структура нейросети разрабатывается с помощью объекта модели Keras Sequential(), который создает последовательную модель с пошаговым добавлением в нее слоев. Слои Keras являются основным строительным блоком моделей Keras. Каждый слой получает входную информацию, выполняет некоторые вычисления и, наконец, выводит преобразованную информацию. Выходные данные одного слоя будут передаваться следующему слою в качестве его входных данных. В модели Keras имеется большой набор возможных слоев: Dense, Dropout, Reshape, Permute, Lambda, Embedding и другие.

Dense-слой является необходимым и базовым. Это слой нейронной сети с глубокими связями, который используется для языковых моделей. Он отвечает за соединение нейронов из предыдущего и следующего слоя. Например, если первый слой имеет 5 нейронов, а второй – 3, то общее количество соединений между слоями будет равно 15. Dense-слой отвечает за эти соединения, и у него есть настраиваемые гиперпараметры: количество нейронов, тип активации, инициализация типа ядра. Dropout-слой помогает избавиться от переобучения модели, задавая уровень отсева. Таким образом, некоторые нейроны становятся равными 0, и это сокращает вычисления в процессе обучения.

Другие слои (Reshape, Permute, Lambda …) носят служебный характер и предназначены для различных преобразований информации в слоях.

Каждый нейрон сети получает один или несколько входных сигналов от нейронов предыдущего слоя. Если речь идет про первый скрытый слой, то он получает данные из входного потока. Начиная со второго скрытого слоя, на вход каждому нейрону подается взвешенная сумма нейронов из предыдущего слоя. На втором скрытом слое происходит активация входных данных за счет функции активации. С помощью функции активации становится доступной реализация обратного распространения ошибки, которая помогает скорректировать веса нейронов. Методом обратного распространения ошибки происходит обучение модели. Существует несколько вариантов функции активации. Сеть берет одну обучающую выборку и использует ее значения в качестве входных данных слоя, далее происходит активация этих данных, и на вход следующему слою подаются уже новые взвешенные данные после активации. Аналогичный процесс происходит и на последующих слоях. Результат последнего слоя будет прогнозом для обучающей выборки. Далее применяется функция потерь, которая показывает, насколько сильно ошибается модель. Чтобы эту ошибку уменьшить, применяется еще одна функция – оптимизатор. Она использует производные для ответа на вопрос – насколько сильно изменится функция потерь при небольшом изменении весов между нейронами.

Программное обеспечение процесса обучения и создания модели реализовано на языке Python и условно состоит из трех блоков (A, B, C):

В блоке «A» считывается подготовленный Dataset в формате json. Структура такого Dataset содержит языковую (tag, patterns) часть и информационную (все оставшиеся разделы) ({tag, name_tag, patterns, responses, doc_id, image_id, context}).

Раздел context определяет текстовый фрагмент данных, к которому необходимо обеспечить доступ в системе на основе запроса на естественном языке.

По всем наборам вопросов к контексту, представленных в разделе patterns, формируются поисковые вектор-запросы и словарь модели. Список классов (или групп ответов) создается из раздела tag. Доступ к самому документу обеспечивает раздел doc_id. Аудио, видео и изображения, необходимые для этого контекста, задаются в разделе image_id. В responses задается краткий ответ на возможные вопросы к контексту.

В блоке «B» формируется массив обучающих и тестовых (для валидации) данных в виде набора векторов слов из списка вопросов и классов модели.

В блоке «C» создается модель на основе данных обучения и конфигурационных параметров модели обучения. Рассмотрим эти параметры более подробно.

Методы активации или функции активации

Методы активации или функции активации – это ключевой элемент в архитектуре нейронных сетей. Они определяют, как нейрон должен реагировать на сумму входных сигналов. Эти функции могут быть простыми, как линейная функция, или сложнее, как нелинейная сигмоидная или ReLU (Rectified Linear Unit). Основная цель функции активации – добавить нелинейность в нейронную сеть. Это позволяет модели обучаться и

выполнять более сложные задачи, имитируя сложные процессы человеческого мозга. Функции активации оказывают значительное влияние на способность нейронной сети учиться. Нелинейные функции позволяют нейронным сетям учиться глубоким представлениям данных, что особенно важно в задачах, требующих абстрактного мышления, таких как распознавание образов и естественный язык. Эффективность выбранной функции активации можно оценить на основе экспериментов и анализа производительности модели на валидационных данных. При этом следует учитывать, что нет универсальной функции активации, которая была бы лучше всех во всех сценариях. Выбор всегда зависит от конкретной задачи и данных. ReLU получила широкое распространение в современных архитектурах нейронных сетей по нескольким причинам:

– В отличие от сигмоида и Tanh, градиент ReLU не сходится к нулю при больших положительных значениях, что помогает ускорить обучение глубоких нейронных сетей.
– ReLU требует меньше вычислительных ресурсов, так как она включает в себя простые операции сравнения и присвоения, в отличие от экспоненциальных вычислений в сигмоиде и Tanh.
– В ReLU все отрицательные входы обнуляются, что приводит к разреженности активаций в нейронной сети. Это может улучшить эффективность и уменьшить переобучение.
– Во многих практических приложениях, особенно в глубоких нейронных сетях, ReLU показала отличные результаты, опережая другие функции активации.

Для входных слоев была выбрана простая, но мощная функция активации ReLU, широко используемая в нейронных сетях. Если входное значение X положительное, функция возвращает это значение, а если X отрицательное, функция возвращает 0.

Для выходного слоя выбран метод активации Softmax, поскольку данная функция активации широко используется в нейронных сетях, особенно в контексте задач классификации, т.е. когда необходимо отнести описанный объект к одному из нескольких классов. Она преобразует вектор вещественных чисел (так называемые логиты – необработанные результаты, полученные на последнем уровне нейронной сети) в вероятностное распределение. Эта функция активации принимает в себя N вещественных чисел, которые вернула ей нейронная сеть, после чего возвращает N чисел, располагающихся в промежутке (0; 1) и дающих в сумме единицу.

Таким образом, каждое значение, получаемое функцией активации Softmax, интерпретируется как вероятность принадлежности к соответствующему классу.

Оптимизатор

Оптимизатор реализует обычную стохастическую процедуру обучения методом градиентного спуска, которая поддерживает различные функции потерь и штрафы для классификации. Был выбран стохастический градиентный спуск с ускоренным градиентом Нестерова, который дает хорошие результаты для задачи множественной классификации.

Функция потерь

Функция потерь – это один из аргументов, необходимых для компиляции модели Keras. Целью функций потерь является вычисление величины, которую модель должна стремиться минимизировать во время обучения. Для этого была выбрана категориальная кросс-энтропия. Она вычисляет потерю кросс-энтропии между истинными метками и предсказанными метками. Категориальная кросс-энтропия широко используется во многих приложениях машинного обучения. Преимущества категориальной кросс-энтропии заключаются в следующем:

– Имеет вероятностную интерпретацию, позволяет модели выводить вероятности каждого класса, что может быть полезно для понимания уверенности модели в своих решениях.
– При использовании с функцией активации Softmax категориальная кросс-энтропия может быть довольно стабильной, поскольку она с наименьшей вероятностью достигает крайних значений.
– Она часто приводит к лучшей производительности в задачах классификации по сравнению с другими функциями потерь, такими как среднеквадратичная ошибка.

Метрика или показатели точности

Метрика — это функция, которая используется для оценки работы модели. Метрические функции указываются в параметре метрики при компиляции модели. Метрическая функция похожа на функцию потерь, за исключением того, что результаты оценки метрики не используются при обучении модели. В качестве метрической функции можно использовать любую из функций потерь. Выбрана метрика accuracy, она рассчитывает как часто прогнозы совпадают с обучающими данными (метками).

Количество эпох

Количество эпох показывает, сколько раз модель подвергается воздействию обучения. Эпоха – один проход вперед или назад для всех примеров обучения.

Batch size – количество обучающих примеров за одну итерацию. Чем больше batch size, тем больше места будет необходимо в памяти компьютера.

В результате выполнения программы обучения и сохранения модели формируются 3 бинарных файла: модели, словаря и классов.

Для оптимальной настройки конфигурации модели обучения был проведен ряд тестовых прогонов, в которых изменялись параметры модели. В качестве критерия качества был выбран показатель accuracy (точность). Было замечено, что наибольшее влияние на качество модели оказывают следующие параметры: количество эпох, количество слоев нейросети, параметры dropout слоев, количество входов слоя и т. д.

Процесс обучения в модели Keras оказался очень быстрым: при размере датасета в 500К время обучения составляет около 20 секунд на обычном офисном компьютере при 200 эпохах. Это позволило экспериментальным путем настроить конфигурацию обучаемой модели: вначале оптимизировалось количество эпох, затем количество слоев, параметры dropout слоев и т. д.

При обучении были исследованы нейросети, содержащие 4, 6, 8 и 10 слоев, а количество эпох варьировалось от 10 до 200. Лучшей по характеристикам (скорость обучения, точность и минимальный объем) оказалась модель на основе 6 слоев: один входной, два скрытых, два dropout-слоя и один выходной, как показано рис.1.

На рис. 2 представлены показатели потерь и точности такой языковой модели. Показатель “точность” характеризует точность обучения модели: чем он ближе к единице, тем лучше качество обучения. Показатель “потери” в глубоком обучении определяет значение, которое нейронная сеть пытается минимизировать. Чем оно ближе к нулю, тем лучше качество обучения. Нейронная

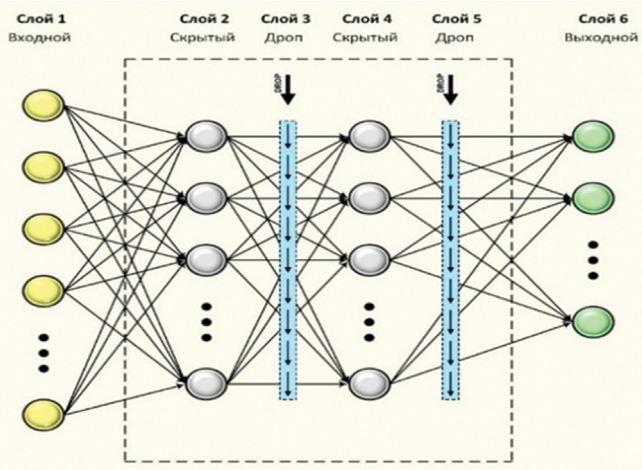

Рис. 1. Структура нейросети
Fig. 1. Neural network structure

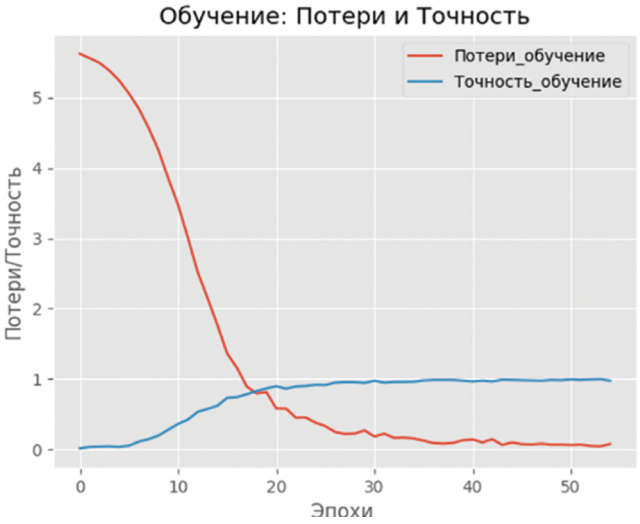

Рис. 2. Показатели потерь и точности языковой модели обучения для обучающей выборки
Fig. 2. Indicators of loss and accuracy of the language learning model for the training sample

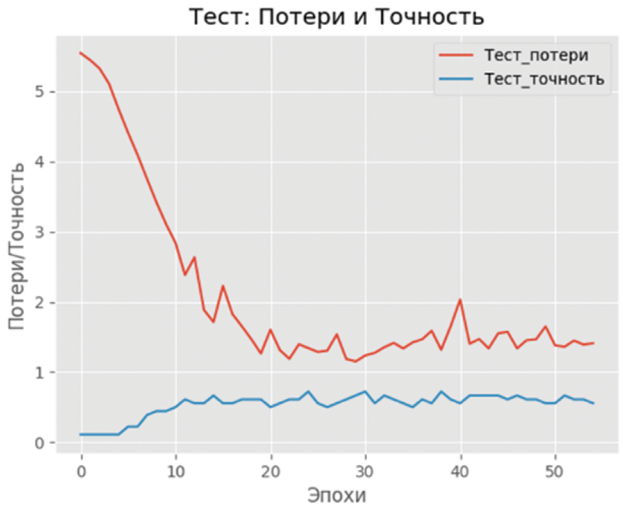

Рис. 3. Показатели потерь и точности языковой модели обучения для тестовой выборки
Fig. 3. Indicators of loss and accuracy of the language learning model for the test sample

сеть обучается, корректируя свои внутренние веса таким образом, чтобы уменьшить потери. На рис. 3 видно, что оптимальное число эпох для обучающей выборки составляет около 50, когда достигаются максимальное значение точности, а также минимальное значение потерь. На рис. 3 представлены показатели потерь и точности языковой модели для тестовых данных, которые составляли 15 % от общего количества данных. Показатель “Тест_потери” – это потери на тестовой (валидационной) выборке, который характеризует работу модели на данных, не представленных в обучающей выборке. Показатель “Тест_точность” определяет точность модели по тестовой выборке.

Из рисунка видно, что максимальная точность и минимальные потери были достигнуты при 45 эпохах. Показатель потерь равнялся 1, что отличается от показателя потерь на обучающей выборке из-за недостаточного количества и разнообразия вопросов-ответов в обучающей выборке на момент тестирования.

Система организации диалога с пользователем (chat)

Программный модуль для получения ответов на запросы пользователей (т.е. общения пользователя с системой) был реализован на языке Python.

Структура ПО содержит 5 основных блоков:

– графический интерфейс (форма) из 5 панелей с элементами управления;
– блок загрузки информации (модели, словаря, классов, датасета);
– блок формирования запроса на основе динамически подключаемого словаря;
– токенизация поискового запроса и формирование запроса в векторном виде;
– блок получения результата на вектор поискового запроса от нейронной сети и формирования ответа.

Пользовательский интерфейс представляет собой обычное окно, состоящее из 5 панелей (рис. 5).

1 панель предназначена для ввода запроса пользователя к системе в виде набора слов на естественном языке. При старте программы заносится слово «Справка», и при нажатии кнопки «Ответ» система выдает расширенную справку по системе и принципам ее работы.

2 панель содержит запросы и ответы системы. Это динамическая область текста включает в себя все запросы и ответы системы с возможностью выделения и выбора набора слов для создания нового запроса.

3 панель содержит информацию о Контексте модели и служит для получения более полного ответа на запрос.

4 панель предназначена для дополнительного выбора информации. Содержит динамически отображаемый словарь (при вводе запроса в панели 1) и обеспечивает доступ к документам, графическим материалам, аудио- и видеоинформации для конкретного запроса.

5 панель служебного характера и обеспечивает технической информацией по системе и запросу пользователя в виде наиболее подходящих классов ответа системы и их вероятностей. Может отключаться с помощью снятия флажка «Отладка» на 4 панели.

Ввод текста запроса поддерживается динамическим способом посредством словаря модели в панели 4. При вводе части слова (шаблона слова) динамически представляется набор слов из найденного в словаре по шаблону в панели 4, достаточно щелкнуть мышкой по слову в этой панели, и оно перенесется в панель 1 – панель запросов. Последовательно проводя такую процедуру ввода, можно быстрее и эффективнее сформировать необходимый запрос, т.к. он будет содержать все слова из

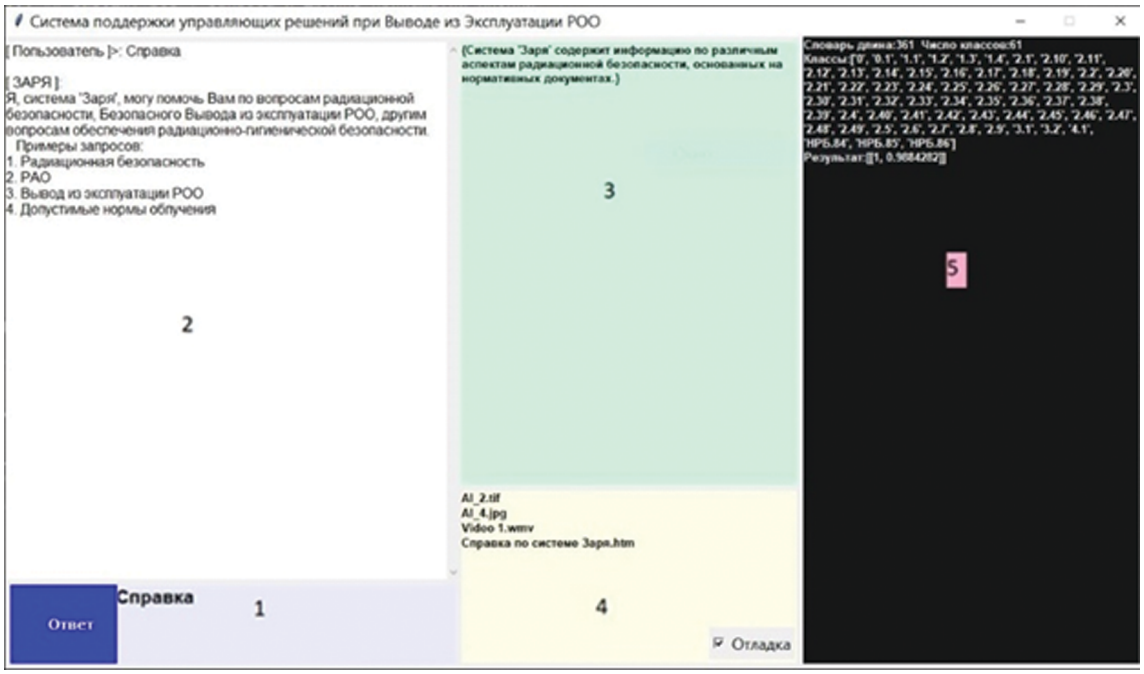

Рис. 4. Интерфейс системы организации диалога с пользователем и пример работы на запрос «Справка»

Fig. 4. The interface of the system for organizing a dialogue with the user and an example of how to respond to the «Help» request

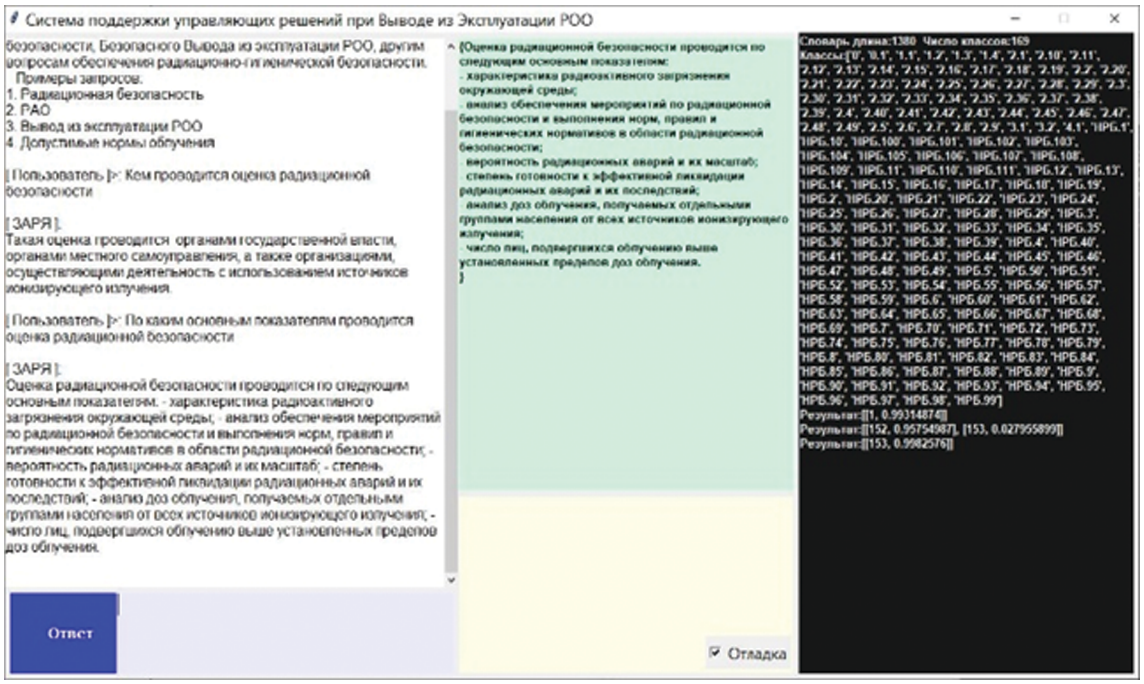

Рис. 5. Пример работы системы диалога с пользователем по запросу о Нормах радиационной безопасности

Fig. 5. An example of the operation of the system of dialogue with the user on request about radiation safety standards

словаря системы, что обеспечит наиболее релевантный поисковый запрос.

Предусмотрена возможность ввода части запроса выделением слов из панели 2 системы. После процедуры выделения слов в этой панели они автоматически переносятся в панель 1 системы.

На рис. 4 приведен пример работы на запрос «Справка». После нажатия пользователем на клавишу «Ответ» на панели 1, на панели 2 представлен ответ системы, а на панели 4 – дополнительные графические, видео и текстовые материалы, которые можно развернуть, щелкнув мышкой компьютера на соответствующий элемент. В итоге можно получить 2 графических изображения, один видео и один текстовый материал информационно-справочного характера по возможностям системы.

На рис. 5 приведен пример запроса пользователя о нормах радиационной безопасности, взятого из тестовой выборки.

Заключение

Моделирование системы вопрос–ответ, предназначенной для поддержки процессов ВЭ РОО, показало возможность создания эффективного инструмента, способного улучшить качество и скорость принятия решений в условиях сложных и часто изменяющихся требований регуляторных органов.

Система продемонстрировала высокую точность в ответах на вопросы, связанные с регуляторными аспектами вывода из эксплуатации, основанных на документах, входящих в набор данных для машинного обучения. Алгоритмы, использованные для обработки и интерпретации текстовой информации, показали высокую эффективность в распознавании и обработке запросов. Система была протестирована на различных сценариях, включая внутреннюю оценку алгоритма машинного обучения модели Keras и тестовые вопросы-ответы, не вошедшие в набор данных для машинного обучения.

Система вопрос–ответ может существенно упростить доступ к критически важной информации для специалистов, занятых в процессе вывода из эксплуатации. Это позволит сократить время на поиск данных и повысить точность выполняемых операций, что особенно важно в условиях высокой степени риска и строгих нормативных требований. Она также может способствовать снижению вероятности ошибок, которые возникают при интерпретации сложных регуляторных документов.

В дальнейшем система требует внесения всего объема нормативных документов, регулирующих ВЭ РОО, а также постоянного обновления базы знаний для поддержания актуальности и точности предоставляемых ответов. Следует сосредоточить усилия на расширении функционала системы для поддержки более широкого спектра вопросов, связанных с выводом из эксплуатации, а также на интеграции с другими инструментами и системами управления данными. Важно также продолжить исследования в области улучшения алгоритмов обработки естественного языка и их адаптации к специфике терминологии, используемой в ядерной отрасли.

Предложенный подход, структура модели и имеющиеся открытое ПО для Keras по нейросетям позволяют создать специализированный чат по объекту, выводимому из эксплуатации, где будет собрана вся имеющаяся (текстовая, графическая и др.) информация об объекте и с помощью запросов на естественном языке пользователи могут получать необходимые сведения.

Разработанная гибкая структура модели позволяет без особых проблем увеличить возможности поддержки принятия решений пользователей механизмами подключения расчетных блоков, модулей запросов к базам данных, визуализации радиационной обстановки на объекте, подлежащему выводу из эксплуатации и др.

Применение предложенной системы не только гарантирует соблюдение регламентов по радиационной безопасности, но и улучшит общую управляемость процессами на всех этапах работы. В долгосрочной перспективе это позволит снизить затраты на радиационно-гигиенические мероприятия и минимизировать потенциальные негативные последствия для окружающей среды и здоровья населения.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Devlin J., Chang M.W., Lee K., Toutanova K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805.
2. Sun C., Qiu X., Xu Y., Huang X. How to Fine-Tune BERT for Text Classification? China National Conference on Chinese Computational Linguistics. Springer, Cham. P. 194-206.
3. Wolf T., Debut L., Sanh V., Chaumond J., Delangue C., Moi A., Rush A.M. Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. P. 38-45.
4. Radford A., Narasimhan K., Salimans T., Sutskever I. Improving Language Understanding by Generative Pre-Training. OpenAI. 2018.
5. Floridi L., Chiriatti M. GPT-3: Its Nature, Scope, Limits, and Consequences. Minds and Machines. 2020;30;4:681-694.
6. Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Stoyanov V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv. 2019 preprint arXiv:1907.11692.
7. Gulli A., Sujit P. Deep Learning with Keras: Implementing Deep Learning Models and Neural Networks with the Power of Python. Birmingham, Packt, 2017. 318 p.
8. Anshik. AI for Healthcare: Keras and TensorFlow. AI and Machine Learning for Healthcare. New Delhi, 2021. 381 p.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.
Финансирование. Исследование не имело спонсорской поддержки.
Участие авторов. Статья подготовлена с равным участием авторов.
Поступила: 20.03.2025. Принята к публикации: 25.04.2025.

Conflict of interest. The authors declare no conflict of interest.
Financing. The study had no sponsorship.
Contribution. Article was prepared with equal participation of the authors.
Article received: 20.03.2025. Accepted for publication: 25.04.2025.