



УДК 004.891

МРНТИ 28.23.29

https://doi.org/10.53364/24138614_2026_40_1_18

А. Исмухамедова¹, С. Бельгинова^{2*}, А. Баканова³, Т. Рысбаев³, А. Халимов³

¹Восточно-Казахстанский технический университет им. Д. Серикбаева,
Усть-Каменогорск, Казахстан

² Университет "Туран", Алматы, Казахстан

³Казахстанско-Американский Свободный Университет (КАСУ), Усть-Каменогорск,
Казахстан

*E-mail: sbelginova@gmail.com

АРХИТЕКТУРА ИНТЕЛЛЕКТУАЛЬНОЙ ВЕБ-СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ КЛИНИЧЕСКИХ РЕШЕНИЙ ДЛЯ ПРОГНОЗИРОВАНИЯ СМЕРТНОСТИ В ОТДЕЛЕНИЯХ ИНТЕНСИВНОЙ ТЕРАПИИ

Аннотация. В этой статье рассматривается проект веб-ориентированной системы поддержки принятия врачебных решений (СППВР), предназначенной для прогнозирования риска смерти у пациентов отделения интенсивной терапии (ICU). Предлагаемое решение объединяет современные подходы к машинному обучению с асинхронной веб-архитектурой и интеллектуальным диалоговым интерфейсом. Система использует микросервисный подход и использует каналы Django и WebSocket. Такой подход обеспечивает отличный пользовательский интерфейс и позволяет обрабатывать большое количество параллельных соединений в режиме реального времени.

Клинические данные набора MIMIC-IV послужили основой для обучения аналитического ядра системы. Он включает в себя многоступенчатый пайплайн обработки данных с импутацией пропусков, инженерией признаков и ансамблевым моделированием на основе градиентного бустинга LightGBM. Результаты эксперимента показали, что модель имеет высокую прогностическую эффективность (AUC-ROC 0,982) при сохранении правильной калибровки вероятностных оценок. Интерпретация прогнозов методом SHAP повысила доверие со стороны клиницистов и объяснила ключевые клинические факторы. Особое внимание уделяется поддержке мультимодальных входных данных, таких как медицинские документы в форматах PDF и Excel, а также текстовые сообщения, что делает систему более пригодной для клинических операций.

Ключевые слова: MIMIC-IV, архитектура веб-приложений, каналы Django, машинное обучение, прогнозирование смертности, интерпретируемость моделей, WebSocket, медицинская информатика, система поддержки принятия клинических решений.

Введение.

Генерация огромных объемов данных характерна для современной интенсивной терапии. Непрерывный поток информации, генерируемый мониторами пациентов, лабораторными информационными системами (LIS) и электронными медицинскими картами (EHR), может быть анализирован в режиме реального времени в большей степени, чем можно было бы ожидать от человека. Алгоритмы машинного обучения (ML) имеют большой потенциал для прогнозирования клинических исходов, таких как сепсис [1],

острая почечная недостаточность [2] и летальный исход [3]. Эти возможности были продемонстрированы существующими исследованиями [4]. Несмотря на высокую теоретическую точность, существует значительный разрыв между исследовательскими моделями и их применением в реальном мире у постели больного [5].

Большинство современных прогностических моделей представляют собой статические отчеты или изолированные скрипты, такие как Jupyter Notebooks, которые не включены в клинический рабочий процесс. Прикладные задачи машинного обучения, особенно в медицине, требуют методов объяснения предсказаний сложных моделей, таких как SHAP [6]. Когда врачи вынуждены вручную переносить данные, время принятия решений увеличивается, а риск ошибок ввода увеличивается. Кроме того, многие модели «черного ящика», также известные как «черный ящик», не дают объяснений своих прогнозов, что снижает доверие пациентов к ним [7]. Использование непрозрачных моделей искусственного интеллекта в здравоохранении растет, что подчеркивает важность обеспечения безопасности и доверия при принятии клинических решений с помощью объяснимого искусственного интеллекта [8].

Целью этой работы является разработка и внедрение архитектуры веб-приложения, которое будет служить «мостиком» между сложными прогностическими моделями, обученными на данных MIMIC-IV, и конечными пользователями, также известными как клиенты. Благодаря разработанной системе можно вводить данные пациента (включая неструктурированные медицинские выписки) и мгновенно оценивать риск с интерпретацией факторов влияния.

Материалы и методы исследования.

В этом исследовании использовалась база данных MIMIC-IV (версия 3.1), содержащая деперсонализированные клинические данные пациентов, госпитализированных в отделении интенсивной терапии медицинского центра Beth Israel Deaconess Medical Center (Бостон, США) с 2008 по 2019 год [9], что послужило основой для разработки и последующей валидации прогностического ядра. База данных MIMIC-IV широко используется для исследований в области медицинской аналитики и машинного обучения, поскольку она включает демографические характеристики, результаты лабораторных исследований, витальные показатели, назначения лекарственных средств и другие клинически значимые параметры.

Для проведения анализа были определены критерии включения для исследуемой группы пациентов. В выборку были включены пациенты старше 18 лет, которые впервые были госпитализированы в отделение интенсивной терапии, что позволило исключить влияние повторных госпитализаций на результаты моделирования. Дополнительным условием было непрерывное наблюдение витальных показателей в течение не менее 24 часов, что необходимо для правильного формирования входных признаков прогностической модели.

Диаграмма потока пациентов показывает процесс формирования финальной выборки пациентов из базы данных MIMIC-IV, который включает в себя этапы применения критериев исключения и определения окончательного размера когорты (рисунок 1).

Благодаря использованию языка программирования Python [10], разработанная система основана на микросервисной архитектуре и обеспечивает модульность, расширяемость и тесную интеграцию веб-компонентов с библиотеками анализа данных и машинного обучения, такими как Pandas, Scikit-learn и LightGBM. Представленный набор технологий выбран из-за их развитой экосистемы инструментов для обработки медицинских данных и их широкой применимости в научных и прикладных исследованиях.

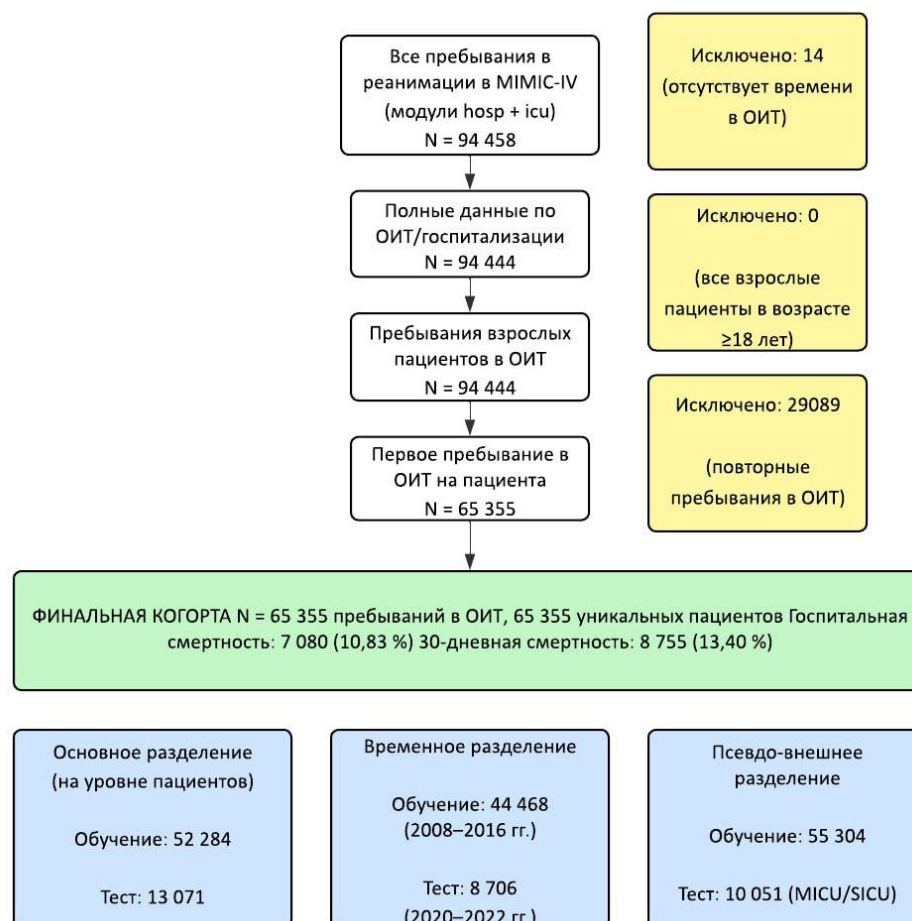


Рисунок 1 – Диаграмма потока пациентов

Архитектура системы состоит из нескольких важных частей, каждая из которых выполняет определенную функциональную функцию. Встроенный в фреймворк Django [11] (версия 4.2.7), серверная часть отвечает за управление пользователями, защиту данных и взаимодействие объектно-реляционно с базами данных PostgreSQL. Асинхронный шлюз каналов Django [12] реализует протокол WebSocket для двустороннего взаимодействия с клиентской частью в режиме реального времени. Данный метод позволяет серверу запускать передачу данных клиенту, что особенно важно для длительных вычислений, таких как анализ загружаемых файлов и уведомление о завершении обработки без необходимости перезагрузки страницы, что отличается от обычного HTTP-опроса.

Redis [13] используется в качестве брокера сообщений для маршрутизации сообщений между различными версиями приложения. Такое решение позволяет увеличивать количество одновременных пользователей и обеспечивает горизонтальную масштабируемость системы. Отдельное место в архитектуре занимает модуль искусственного интеллекта-оркестратора, который должен управлять логикой диалога, собирать пользовательский контекст и координировать вызовы моделей машинного обучения. Данный модуль обеспечивает согласованную и воспроизводимую работу интеллектуальных компонентов, связывая пользовательский интерфейс с аналитическим ядром системы.

WebSocket-потребитель, реализованный в виде класса ChatConsumer, наследуемого AsyncWebSocketConsumer, является основным компонентом асинхронного взаимодействия между клиентской и серверной частями системы. Поэтому такая часть отвечает за обработку входящих сообщений в режиме реального времени и выполняет несколько

жизненно важных функций, таких как подтверждение личности пользователя на этапе установления соединения, маршрутизация сообщений в группы с уникальным видом `user_{id}_{session_id}` и асинхронный запуск задач анализа данных без блокировки основного цикла ввода-вывода.

Асинхронная модель обработки запросов позволяет разделять вычислительно затратные этапы инференса моделей машинного обучения и операции сетевого взаимодействия. Сервер начинает выполнять аналитическую задачу в фоновом режиме после получения сообщения от клиента.

Упрощённый пример реализации метода `receive` приведён ниже:

```
async def receive (self, text_data):
    data = json.loads(text_data)
    # Асинхронная отправка подтверждения получения
    await self.send (json.dumps ({'type': 'ack'}))
    # Запуск ML-инференса в фоновом режиме
    asyncio.create_task (self.process_ai_response(data))
```

Представленный подход позволяет поддерживать высокую отзывчивость пользовательского интерфейса, обеспечивая среднюю задержку обработки сообщений менее 100 мс даже при выполнении ресурсоёмких вычислительных задач. Поэтому особенно важно в контексте интерактивных медицинских систем поддержки принятия решений, где задержки могут негативно влиять на пользовательский опыт и эффективность работы клинициста.

Для достижения высокой точности и устойчивости прогнозов был разработан многоступенчатый пайплайн обработки данных, включающий этапы предварительной очистки, нормализации, формирования признаков и инференса моделей машинного обучения [14]. Общая структура пайплайна представлена на рисунке 2. Такое модульное построение обеспечивает воспроизводимость результатов, упрощает экспериментирование с различными конфигурациями моделей и позволяет адаптировать систему к расширению набора входных данных или смене алгоритмов прогнозирования.

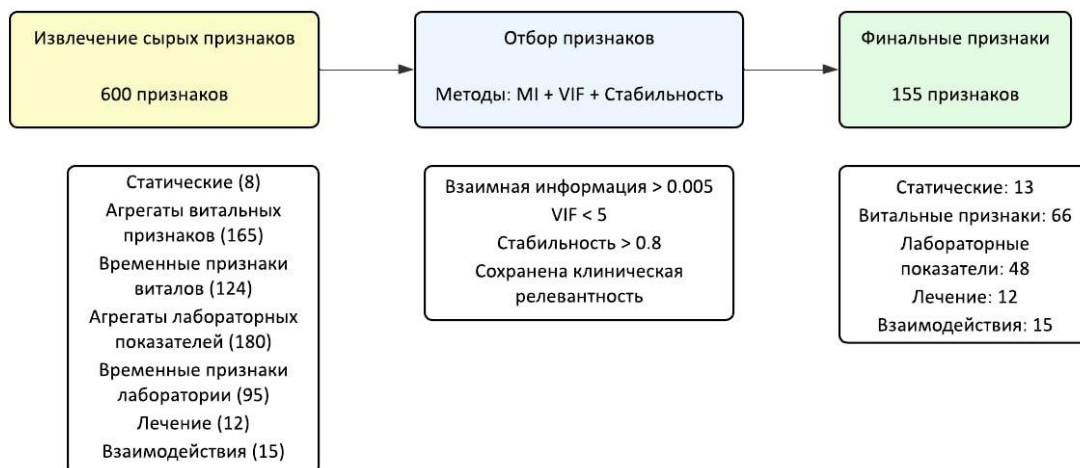


Рисунок 2 – Схема обработки данных: от сырых временных рядов до генерации признаков и обучения модели.

На этапе предварительной обработки данных использовался метод Multiple Imputation by Chained Equations (MICE). Данный метод позволяет учитывать взаимосвязи между признаками и уменьшает смещение, вызванное недостаточностью клинических наблюдений. Для данных отделений интенсивной терапии, которые характеризуются высокой долей пропусков и нерегулярностью измерений, этот метод особенно полезен.

На этапе инженерии признаков было создано расширенное количество входных переменных, содержащее более 300 признаков. Он включал статистические агрегаты временных рядов витальных показателей, таких как минимальные и максимальные значения, средние уровни и параметры динамики, включая линейные тренды, для клинически значимых показателей, таких как концентрация лактата, уровень глюкозы и уровень рН. Таким образом, можно представить как текущее состояние пациента, так и его динамику в кратчайшие сроки.

На этапе моделирования библиотека LightGBM использовалась для обучения ансамбля моделей градиентного бустинга на решающих деревьях. Фреймворк Optuna использовался для выбора наилучших гиперпараметров, что позволило автоматизировать поиск конфигураций, чтобы максимизировать качество прогнозирования при сохранении вычислительной эффективности модели.

Модуль ChatOrchestrator, который отвечает за управление потоком данных между аналитическим ядром и пользовательским интерфейсом, является ключевым компонентом разработанной системы. Такой компонент координирует обработку запросов, агрегирует контекст и маршрутизирует данные к соответствующим вычислительным модулям. Рисунок 3 показывает общую схему обработки пользовательского запроса.

Интернет-интерфейс, основанный на фреймворке Bootstrap 5, обеспечивает кроссплатформенную доступность и удобство взаимодействия с системой. Защитное WebSocket-соединение (wss://) используется для передачи файлов и текстовых сообщений, что обеспечивает безопасность и конфиденциальность передаваемых данных.

Слой маршрутизации, который использует инфраструктуру ASGI и библиотеки каналов Django, принимает входящие соединения и отправляет их соответствующему WebSocket-потребителю. Данная стратегия позволяет эффективно обрабатывать множественные параллельные соединения и поддерживать асинхронный режим взаимодействия с пользователем.

Логика обработки запроса меняется на этапе оркестрации в зависимости от типа входных данных. Возраст пациента и лабораторные значения, такие как уровень лактата, проверяются после получения текстового сообщения. Когда вы загружаете файлы в формате PDF или Excel, активируется модуль оптического распознавания текста и парсинга, известный как `file_processors.py`. Он используется для получения структурированных данных из документов и медицинских выписок, которые не структурированы.

После этого данные передаются в аналитическое ядро, которое собирает признаки в контексте ChatSession пользователя. Когда механизм анализа полноты данных (Gap Analysis) собирает достаточно данных, сформированный запрос передается в обученную модель машинного обучения для выполнения прогностического расчёта.

На последнем этапе создается ответ, который будет доступен пользователю. В интерпретации используется агент LLM, чтобы дополнить вероятностную оценку риска, представленную результатами прогнозирования, что позволяет создавать текстовый ответ на естественном языке, который понятен медицинскому специалисту и ориентирован на клиническое восприятие. Такой подход повышает практическую ценность системы в качестве помощника в принятии решений.

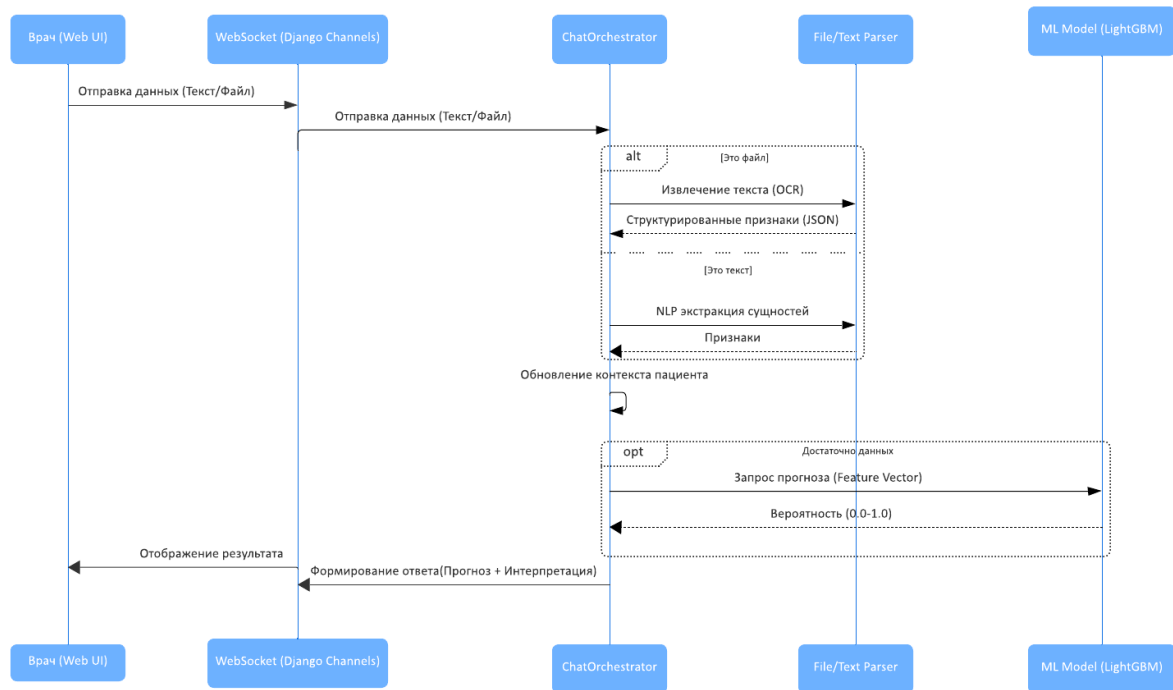


Рисунок 3 – Диаграмма последовательности обработки запроса в системе

Описание архитектуры системы

Система представляет собой веб-ориентированную платформу поддержки принятия врачебных решений (СППВР) с микросервисной архитектурой и асинхронным взаимодействием. Асинхронная микросервисная веб-архитектура в основе которой событийно-ориентированный (event-driven) паттерн через WebSocket.

Ключевые компоненты такого подхода являются Клиентская часть (Frontend), серверная часть (Backend), так же входят такие части как аналитическое ядро (ML и Data processing), генеративный и модуль оркестрации.

Технологический стек представляет из себя для Frontend веб-интерфейс на базе Bootstrap 5, общается с сервером через защищенные протоколы WebSocket которые необходимы для отправки файлов и текстовых сообщений. Backend представляется из себя фреймворк Django, сюда относится безопасность, управление пользователями и ORM. База данных PostgreSQL. Асинхронный шлюз, который обрабатывает соединения.

В свою очередь модули оркестрации и генеративный нужны для поддержки центрального узла, который управляет логикой диалога и собирает контент. Агент же на базе языковой модели для формирования текстового ответа на естественном языке, объединяет прогнозы и объяснения.

Результаты и их обсуждение.

В экспериментальном исследовании была проведена оценка производительности нескольких архитектур моделей машинного обучения, таких как логистическая регрессия, случайный лес (Random Forest) и градиентный бустинг на решающих деревьях (LightGBM). Дисбаланс классов, вызванный относительно низкой долей целевого события (летальность в отделении интенсивной терапии составляет 10–15%). Для ее нивелирования использовались различные методы. К ним относятся взвешивание классов, метод синтетического увеличения выборки SMOTE, а также гибридный подход, который объединяет несколько методов балансировки [15].

Для сравнения качества моделей использовались стандартные методы бинарной классификации для отложенной тестовой выборки. На рисунке 4 показаны результаты в

виде ROC-кривых, а также в сводной таблице метрик (таблица 1). Как видно из данных, модели на основе градиентного бустинга показали лучшие результаты по большинству критериев качества.

Рисунок 4 сравнение ROC-кривых для различных алгоритмов машинного обучения. Отличная разделяющая способность модели LightGBM демонстрируется ее наибольшей площадью под кривой (AUC-ROC).

Модель LightGBM с использованием взвешивания классов продемонстрировала наилучший баланс чувствительности и точности. При высокой точности классификации и значительной чувствительности (Recall) на уровне 80,2% эта модель особенно подходит для медицинских задач, где крайне важно уменьшить количество ложноотрицательных прогнозов. Способность модели отлично различать пациентов с высоким и низким риском неблагоприятного исхода подтверждается высоким значением AUC-ROC (0,982).

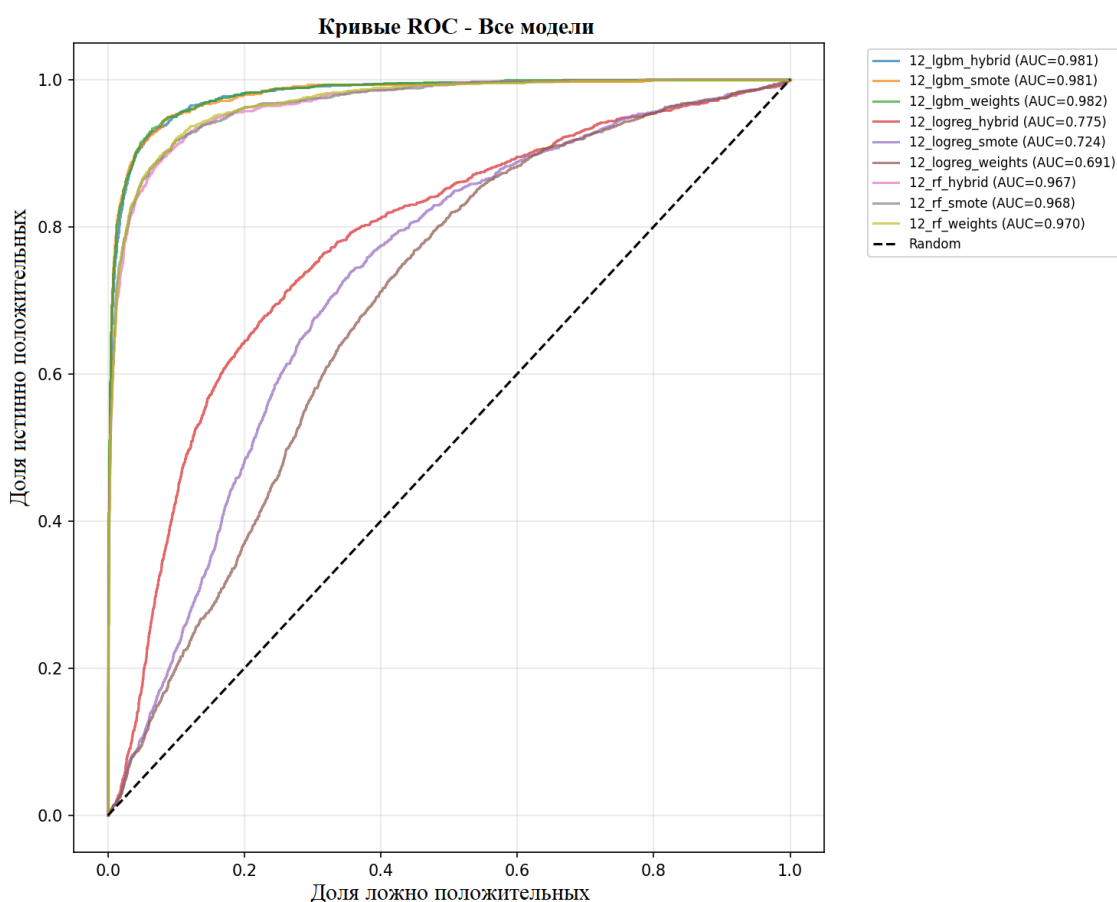


Рисунок 4 – Сравнение ROC-кривых для различных алгоритмов машинного обучения. Модель LightGBM показывает наибольшую площадь под кривой.

Таблица 1 - Метрики качества моделей на тестовой выборке

Модель	AUC-ROC	Precision	Recall	F1-Score
LightGBM (Weights)	0.982	0.883	0.802	0.840
LightGBM (Hybrid)	0.981	0.757	0.883	0.815
Random Forest (Weights)	0.970	0.911	0.610	0.731
Logistic Regression	0.775	0.258	0.098	0.142

Для дополнительной оценки надёжности вероятностных предсказаний была проведена проверка калибровки моделей с использованием калибровочных кривых. Результаты данной оценки представлены на рисунке 5. Близость к диагональной линии

свидетельствует о высокой степени соответствия между предсказанными вероятностями и наблюдаемыми частотами неблагоприятных исходов, что указывает на корректную вероятностную интерпретацию выходов модели и повышает доверие к её применению в клинической практике.

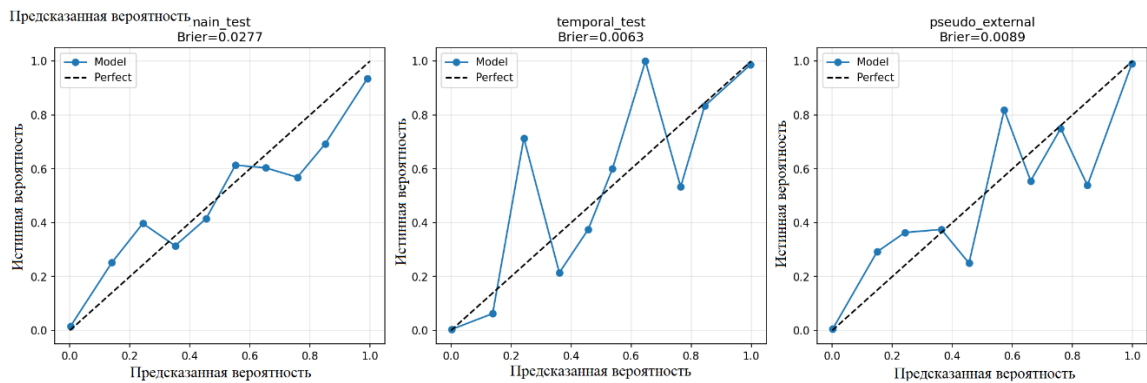


Рисунок 5 – Оценка калибровки моделей. Близость к диагонали указывает на соответствие предсказанной вероятности реальной частоте исходов.

Одним из ключевых требований к клиническим системам поддержки принятия врачебных решений является прозрачность и объяснимость выдаваемых прогнозов. В связи с этим в разработанную систему был интегрирован модуль интерпретации на основе метода SHAP (SHapley Additive exPlanations), позволяющий количественно оценивать вклад отдельных признаков в итоговое предсказание модели.

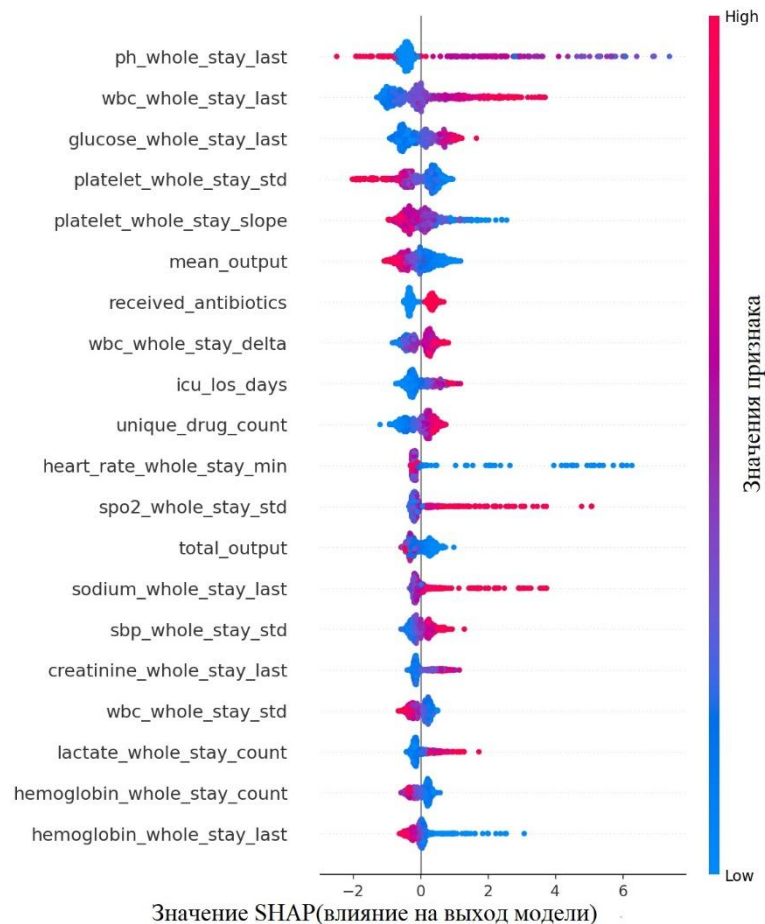


Рисунок 6 – Вклад признаков в прогноз модели. Красный цвет означает высокие значения признака. Видно, что высокие уровни лактата и низкий pH повышают риск.

Сводная диаграмма важности признаков, представленная на рисунке 6, демонстрирует влияние ключевых клинических параметров на риск неблагоприятного исхода. Анализ результатов показывает, что повышенные значения уровня лактата и сниженные показатели рН крови вносят наибольший вклад в увеличение прогнозируемого риска, что согласуется с известными клиническими представлениями о патофизиологии критических состояний.

Важной особенностью системы является то, что она не ограничивается выдачей числовой вероятности, а формирует текстовое клиническое пояснение к результату прогноза. Например, при высоком уровне риска система генерирует интерпретируемое сообщение следующего вида: «Риск высокий (85%). Основные факторы: уровень лактата 4,5 ммоль/л (повышен), систолическое артериальное давление 85 мм рт. ст. (понижено)». Такой формат представления результатов повышает их клиническую применимость и облегчает интеграцию системы в повседневную практику врача.

Для оценки эффективности разработанной архитектуры было проведено тестирование нагрузки. Результаты показали, что один экземпляр приложения (Worker) может одновременно обслуживать до 500 активных WebSocket-соединений с задержкой передачи сообщений менее 200 секунд, что соответствует требованиям интерактивного веб-интерфейса. Асинхронная обработка файлов PDF занимает в среднем 1,5–3 секунды, что считается приемлемым временем для клинической практики, поскольку он обеспечивает баланс между скоростью отклика и качеством извлечения данных.

Работа успешно решила задачу интеграции сложных прогностических моделей машинного обучения в веб-интерфейс, который был прост и понятен. Предлагаемая диалоговая система с диалоговой интерфейсной средой позволяет врачам взаимодействовать с системой более естественно, в отличие от традиционных медицинских калькуляторов, которые требуют ручного ввода данных в строго определенные поля. Можно использовать загрузку результатов анализов или вводить данные в текстовом виде, что снижает риск ошибок и повышает удобство использования.

При использовании протокола WebSocket вместо обычных HTTP REST API было несколько важных преимуществ:

1) Снижение задержек. Сервер может отправлять промежуточные сообщения пользователю (например, «Файл распознан, выполняется анализ рисков...») без необходимости повторять запросы, что снижает латентность обмена данными;

2) Двухнаправленность. Сервер может отправлять промежуточные сообщения пользователю (например, «Файл распознан, выполняется анализ рисков...») без дополнительных расходов на установку нового HTTP-соединения (handshake)

Таким образом, архитектурные решения обеспечивают высокую производительность, интерактивность и клиническую применимость системы. Эти решения создают основу для дальнейшего масштабирования и внедрения в медицинскую практику.

Достигнутые показатели эффективности модели ($AUC = 0.982$) значительно превосходят результаты традиционных шкал оценки тяжести состояния пациентов, таких как APACHE II и SOFA, которые обычно демонстрируют AUC в диапазоне 0,7–0,8 [4]. Высокая чувствительность модели ($Recall \approx 80\%$) делает систему пригодной для использования в качестве инструмента скрининга, позволяя своевременно выявлять пациентов группы риска и инициировать ранние вмешательства.

Дополнительным преимуществом является интеграция методов интерпретации, основанных на SHAP, что позволяет преодолеть ограничения «черного ящика» машинного обучения. Врач может визуально оценивать вклад ключевых признаков в прогноз, сопоставляя их с собственными клиническими знаниями, что повышает доверие к результатам и способствует безопасной интеграции системы в клиническую практику.

Ключевыми ограничениями текущей реализации системы являются:

1) зависимость от качества входных данных: точность прогнозов напрямую связана с полнотой и достоверностью лабораторных показателей и других клинических данных. Неполные или неточные данные могут снижать надежность предсказаний;

2) необходимость внешней валидации (External Validation): модель обучена исключительно на данных MIMIC-IV (Бостон, США), что ограничивает прямую переносимость результатов на другие клинические популяции. Для расширения применимости системы требуется тестирование на независимых наборах данных из различных медицинских центров с целью оценки устойчивости и корректности прогнозов.

Устранение этих ограничений является приоритетным направлением дальнейших исследований и разработки, особенно в контексте масштабируемого внедрения системы в клиническую практику

Заключение.

В этом исследовании рассматривается разработка веб-ориентированной системы поддержки принятия врачебных решений (СППВР), которая успешно объединяет передовые методы машинного обучения с простым и интерактивным клиническим интерфейсом. Благодаря использованию современных веб-технологий, таких как каналы Django и протокол WebSocket, система была обеспечена высокой скоростью реакции и была предоставлена возможность интерактивного взаимодействия с пользователем в режиме реального времени.

Аналитическое ядро, построенное на ансамбле моделей градиентного бустинга (LightGBM), показало отличные показатели качества прогнозов ($AUC > 0,98$), что подтверждает, что решения, вырабатываемые системой, надежны и имеют клиническую применимость. Интеграция методов интерпретации (SHAP) позволяет врачам оценивать вклад отдельных признаков в прогноз, что повышает доверие к результатам и гарантирует, что модель может быть безопасно использована в реальных клинических ситуациях.

Благодаря масштабируемости и гибкости концепции, основанной на микросервисах и асинхронной оркестрации, система может быть адаптирована для решения других задач медицины, таких как прогнозирование сепсиса, повторная госпитализация или осложнения критических состояний. Чтобы расширить применение СППВР в клинической практике и обеспечить прямое взаимодействие с госпитальными информационными системами, перспективным направлением дальнейшей работы является интеграция со стандартами обмена медицинскими данными, такими как HL7 FHIR.

Таким образом, представленная система сочетает в себе высокую точность прогнозирования, интерпретируемость и простоту использования, что в свою очередь создает основу для дальнейшего внедрения интеллектуальных инструментов поддержки принятия решений в реальном медицинском окружении.

Благодарность.

Данное исследование финансировалось Комитетом науки Министерства образования и науки Республики Казахстан. Проект No AP22683316 "Применение алгоритмов машинного обучения для систем поддержки принятия медицинских решений".

Список литературы

1. Islam, K. R., Prithula, J., Kumar, J., Tan, T. L., Reaz, M. B. I., Sumon, M. S. I., & Chowdhury, M. E. H. (2023). Раннее прогнозирование сепсиса на основе машинного обучения с использованием электронных медицинских записей: систематический обзор. *Journal of Clinical Medicine*, 12(17). <https://doi.org/10.3390/jcm12175658>
2. Li, X., Hu, X., Xu, H., Yu, P., & Ju, H. (2025). Модели прогнозирования риска смертности на основе машинного обучения у пациентов с острой почечной недостаточностью, связанной с сепсисом: систематический обзор. *Frontiers in Medicine*, 12. <https://doi.org/10.3389/fmed.2025.1680180>
3. Rakhmetullina, Z., Belginova, S., Karlygash, A., Ismukhamedova, A., & Tezekpaeva,

- S. (2024). Исследование и внедрение алгоритма анализа медицинских текстов для прогнозирования смертности. *Indonesian Journal of Electrical Engineering and Computer Science*, 34(3), 1965–1977. <https://doi.org/10.11591/ijeecs.v34.i3.pp1965-1977>
4. Ismukhamedova, A., Uvaliyeva, I., & Belginova, S. (2024). Интеграция машинного обучения в электронный медицинский паспорт на основе исследования ВОЗ и ресурсов здравоохранения. *Informatics in Medicine Unlocked*, 44. <https://doi.org/10.1016/j.imu.2023.101428>
5. Hill, N. R. (2025). Преодоление барьеров внедрения: сокращение разрыва между разработкой алгоритмов искусственного интеллекта и их применением в здравоохранении. *Journal of Medical Artificial Intelligence*, 8. <https://doi.org/10.21037/jmai-24-385>
6. Pierce, R. L., Van Biesen, W., Van Cauwenberge, D., Decruyenaere, J., & Sterckx, S. (2022). Объяснимость в медицине в эпоху систем поддержки клинических решений на основе ИИ. *Frontiers in Genetics*, 13. <https://doi.org/10.3389/fgene.2022.903600>
7. González-Alday, R., García-Cuesta, E., Kulikowski, C. A., & Maojo, V. (2023). Обзор прогресса, применимости и перспектив объяснимого искусственного интеллекта в медицине. *Applied Sciences (Switzerland)*, 13(19). <https://doi.org/10.3390/app131910778>
8. Sadeghi, Z., Alizadehsani, R., CIFCI, M. A., Kausar, S., Rehman, R., Mahanta, P., Bora, P. K., Almasri, A., Alkhaldeh, R. S., Hussain, S., Alatas, B., Shoeibi, A., Moosaei, H., Hladik, M., Nahavandi, S., & Pardalos, P. M. (2024). Обзор объяснимого искусственного интеллекта в здравоохранении. *Computers and Electrical Engineering*, 118. <https://doi.org/10.1016/j.compeleceng.2024.109370>
9. Johnson, A., Bulgarelli, L., Pollard, T., Gow, B., Moody, B., Horng, S., Celi, A. L., & Mark, R. (2024). MIMIC-IV (версия 3.1) [Электронный ресурс]. *PhysioNet*. <https://doi.org/10.13026/kpb9-mt58>
10. Joshi, A., & Tiwari, H. (2023). Обзор библиотек Python для Data Science. *Journal of Engineering Technology and Applied Physics*, 5(2). <https://doi.org/10.33093/jetap.2023.5.2.10>
11. Official Django REST Framework Documentation. (n.d.). Официальная документация Django REST Framework [Электронный ресурс]. <https://www.django-rest-framework.org/>
12. Django Channels Documentation. (2022). Документация Django Channels [Электронный ресурс]. <https://channels.readthedocs.io/>
13. Ltd., R. (2023). Официальная документация Redis [Электронный ресурс]. <https://redis.io/documentation>
14. Ismukhamedova, A. M., Bakanova, A. P., & Belgimova, S. A. (2025). Оптимизированная обработка и формирование признаков для комплексного анализа медицинских данных. *Research and Development – 2025: Proceedings of the III International Scientific and Practical Conference*, 187. <https://doi.org/10.46916/10102025-3-978-5-00215-887-4>
15. Ismukhamedova, A., Uvaliyeva, I., & Rakhmetullina, Z. (2024). Экспериментальное исследование модели анализа медицинских данных на основе сравнительной эффективности алгоритмов классификации. *Indonesian Journal of Electrical Engineering and Computer Science*, 36(1), 672–684. <https://doi.org/10.11591/ijeecs.v36.i1.pp672-684>

References

1. Islam, K. R., Prithula, J., Kumar, J., Tan, T. L., Reaz, M. B. I., Sumon, M. S. I., & Chowdhury, M. E. H. (2023). Machine learning-based early prediction of sepsis using electronic health records: A systematic review. *Journal of Clinical Medicine*, 12(17). <https://doi.org/10.3390/jcm12175658>
2. Li, X., Hu, X., Xu, H., Yu, P., & Ju, H. (2025). Machine learning-based mortality risk prediction models in patients with sepsis-associated acute kidney injury: A systematic review. *Frontiers in Medicine*, 12. <https://doi.org/10.3389/fmed.2025.1680180>

3. Rakhmetullina, Z., Belginova, S., Karlygash, A., Ismukhamedova, A., & Tezekpaeva, S. (2024). Research and implementation of the medical text analysis algorithm for predicting mortality. *Indonesian Journal of Electrical Engineering and Computer Science*, 34(3), 1965–1977. <https://doi.org/10.11591/ijeecs.v34.i3.pp1965-1977>
4. Ismukhamedova, A., Uvaliyeva, I., & Belginova, S. (2024). Integrating machine learning in electronic health passport based on WHO study and healthcare resources. *Informatics in Medicine Unlocked*, 44. <https://doi.org/10.1016/j.imu.2023.101428>
5. Hill, N. R. (2025). Overcoming adoption challenges: Bridging the gap between artificial intelligence algorithm development and implementation in healthcare. *Journal of Medical Artificial Intelligence*, 8. <https://doi.org/10.21037/jmai-24-385>
6. Pierce, R. L., Van Biesen, W., Van Cauwenberge, D., Decruyenaere, J., & Sterckx, S. (2022). Explainability in medicine in an era of AI-based clinical decision support systems. *Frontiers in Genetics*, 13. <https://doi.org/10.3389/fgene.2022.903600>
7. González-Alday, R., García-Cuesta, E., Kulikowski, C. A., & Maojo, V. (2023). A scoping review on the progress, applicability, and future of explainable artificial intelligence in medicine. *Applied Sciences (Switzerland)*, 13(19). <https://doi.org/10.3390/app131910778>
8. Sadeghi, Z., Alizadehsani, R., CIFCI, M. A., Kausar, S., Rehman, R., Mahanta, P., Bora, P. K., Almasri, A., Alkhaldeh, R. S., Hussain, S., Alatas, B., Shoeibi, A., Moosaei, H., Hladik, M., Nahavandi, S., & Pardalos, P. M. (2024). A review of explainable artificial intelligence in healthcare. *Computers and Electrical Engineering*, 118. <https://doi.org/10.1016/j.compeleceng.2024.109370>
9. Johnson, A., Bulgarelli, L., Pollard, T., Gow, B., Moody, B., Horng, S., Celi, A. L., & Mark, R. (2024). MIMIC-IV (Version 3.1) [Electronic resource]. *PhysioNet*. <https://doi.org/10.13026/kpb9-mt58>
10. Joshi, A., & Tiwari, H. (2023). An overview of Python libraries for data science. *Journal of Engineering Technology and Applied Physics*, 5(2). <https://doi.org/10.33093/jetap.2023.5.2.10>
11. Official Django REST Framework Documentation. (n.d.). Official Django REST Framework Documentation [Electronic resource]. <https://www.django-rest-framework.org/>
12. Django Channels Documentation. (2022). Django Channels Documentation [Electronic resource]. <https://channels.readthedocs.io/>
13. Ltd., R. (2023). Official Redis documentation [Electronic resource]. <https://redis.io/documentation>
14. Ismukhamedova, A. M., Bakanova, A. P., & Belgimova, S. A. (2025). Optimized data processing and feature engineering for comprehensive medical data analysis. In *Research and Development – 2025: Proceedings of the III International Scientific and Practical Conference*, 187. <https://doi.org/10.46916/10102025-3-978-5-00215-887-4>
15. Ismukhamedova, A., Uvaliyeva, I., & Rakhmetullina, Z. (2024). Experimental study of a medical data analysis model based on comparative performance of classification algorithms. *Indonesian Journal of Electrical Engineering and Computer Science*, 36(1), 672–684. <https://doi.org/10.11591/ijeecs.v36.i1.pp672-684>

ЖАН САҚТАУ БӨЛІМДЕРІНДЕ ӨЛІМ-ЖІТІМДІ БОЛЖАУҒА АРНАЛҒАН КЛИНИКАЛЫҚ ШЕШІМ ҚАБЫЛДАУДЫ ҚОЛДАУШЫ ИНТЕЛЛЕКТУАЛДЫ ВЕБ-ЖҮЙЕНІҢ АРХИТЕКТУРАСЫ

Аңдатпа. Бұл мақалада ICU пациенттерінің өлім қаупін болжауға арналған веб-бағытталған дәрігерлік шешімдерді қолдау жүйесінің (SPPVR) жобасы қарастырылады. Ұсынылған шешім заманауи Машиналық оқыту тәсілдерін асинхронды веб-архитектурамен және интеллектуалды диалог интерфейсімен біріктіреді. Жүйе микросервистік тәсілді қолданады және Django және WebSocket арналарын қолданады.

Бұл тәсіл тамаша пайдаланушы интерфейсін қамтамасыз етеді және нақты уақыт режимінде көптеген параллель қосылымдарды өңдеуге мүмкіндік береді.

Mimic-IV жиынтығының клиникалық деректері жүйенің аналитикалық ядросын оқытуға негіз болды. Ол lightgbm градиентті күшейткіш негізінде өткізгіштік импутациясы, ерекшелік инженериясы және ансамбльдік модельдеуі бар көп сатылы құбыр желісін қамтиды. Эксперимент нәтижелері ықтималдық бағалауларының дұрыс калибрлеуін сақтай отырып, модельдің жоғары болжамдық тиімділігі (AUC-ROC 0,982) бар екенін көрсетті. SHAP болжамдарын түсіндіру клиниктердің сенімін арттырды және негізгі клиникалық факторларды түсіндірді. PDF және Excel форматындағы медициналық құжаттар, сондай-ақ мәтіндік хабарламалар сияқты мультимодальды кірістерді қолдауға баса назар аударылады, бұл жүйені клиникалық операцияларға қолайлы етеді.

Түйін сөздер: MIMIC-IV, веб-қосымшалардың архитектурасы, Django арналары, Машиналық оқыту, өлімді болжау, модельдік интерпретация, WebSocket, Медициналық информатика, клиникалық шешімдерді қолдау жүйесі.

ARCHITECTURE OF AN INTELLIGENT WEB-BASED CLINICAL DECISION SUPPORT SYSTEM FOR MORTALITY PREDICTION IN INTENSIVE CARE UNITS

Abstract. This article discusses the project of a web-based Medical Decision Support System (CBSA) designed to predict the risk of death in ICU patients. The proposed solution combines modern approaches to machine learning with asynchronous web architecture and intelligent interactive interface. The system uses a microservice approach and uses Django and WebSocket channels. This approach provides an excellent user interface and allows you to process a large number of parallel connections in real time.

The clinical data from the MIMIC-IV kit served as the basis for training the analytical core of the system. It includes a multi-stage data processing pipeline with skip imputation, feature engineering, and ensemble modeling based on LightGBM gradient boosting. The experimental results showed that the model has a high predictive efficiency (AUC-ROC 0.982) while maintaining the correct calibration of probabilistic estimates. The interpretation of forecasts by the SHAP method increased the confidence of clinicians and explained the key clinical factors. Special attention is paid to the support of multimodal input data, such as medical documents in PDF and Excel formats, as well as text messages, which makes the system more suitable for clinical operations.

Keywords: MIMIC-IV, web application architecture, Django channels, machine learning, mortality prediction, interpretability of models, WebSocket, medical informatics, clinical decision support system.

Авторлар туралы мәлімет

Исмухамедова Айгерим Мэлсатовна	PhD, AP22683316 ғылыми жобасының жетекшісі, компьютерлік ғылымдар мектебі, Д. Серікбаев атындағы Шығыс Қазақстан техникалық университеті, Өскемен, Қазақстан, E-mail: Aigerim.ismukhamedova1@gmail.com
Бельгинова Сауле Аскербековна	PhD, қауымдастырылған профессор, Ақпараттық технологиялар жоғары мектебі, "Тұран" университеті, Алматы, Қазақстан, E-mail: sbelginova@gmail.com
Баканова Анна Павловна	т. ғ. к., оқытушы, Қазақстан-Америка Еркін Университеті (ҚАЕУ), Өскемен, Қазақстан, E-mail: bakanova55anna@gmail.com
Рысбаев Тимур	"Ақпараттық жүйелер" мамандығының магистранты, Қазақстан-Америка Еркін Университеті (ҚАЕУ), Өскемен, Қазақстан, E-mail: timur.rysbayev1@gmail.com
Халимов Айсат	"Ақпараттық жүйелер" мамандығының магистранты, Қазақстан-Америка Еркін Университеті (ҚАЕУ), Өскемен, Қазақстан, E-mail: Khalais2012@gmail.com

Сведение об авторах

Исмухамедова Айгерим Мэлсатовна	PhD, Руководитель научного проекта AP22683316, Школа компьютерных наук, Восточно-Казахстанский технический университет имени Д. Серикбаева, Усть-Каменогорск, Казахстан, E-mail: Aigerim.ismukhamedova1@gmail.com
Бельгинова Сауле Аскербековна	PhD, ассоциированный профессор, Высшая школа информационных технологий, Университет "Туран", Алматы, Казахстан, E-mail: sbelginova@gmail.com
Баканова Анна Павловна	к.т.н., преподаватель, Казахстанско-Американский Свободный Университет (КАСУ), Усть-Каменогорск, Казахстан, E-mail: bakanova55anna@gmail.com
Рысбаев Тимур	Магистрант специальности «Информационные системы», Казахстанско-Американский Свободный Университет (КАСУ), Усть-Каменогорск, Казахстан E-mail: timur.rysbayev1@gmail.com
Халимов Айсат	Магистрант специальности «Информационные системы», Казахстанско-Американский Свободный Университет (КАСУ), Усть-Каменогорск, Казахстан E-mail: Khalais2012@gmail.com

Information about the authors

Ismukhamedova Aigerim	PhD, Head of the scientific project AP22683316, School of Computer Science, D. Serikbayev East Kazakhstan Technical University, Ust-Kamenogorsk, Kazakhstan, E-mail: Aigerim.ismukhamedova1@gmail.com
Belginova Saule	PhD, Associate Professor, Graduate School of Information Technology, Turan University, Almaty, Kazakhstan, E-mail: sbelginova@gmail.com
Bakanova Anna	PhD, Lecturer, Kazakh-American Free University (KAFU), Ust-Kamenogorsk, Kazakhstan, E-mail: bakanova55anna@gmail.com
Rysbayev Timur	Master's degree in Information Systems, Kazakh-American Free University (KAFU), Ust-Kamenogorsk, Kazakhstan, E-mail: timur.rysbayev1@gmail.com
Khalimov Aisat	Master's degree in Information Systems, Kazakh-American Free University (KAFU), Ust-Kamenogorsk, Kazakhstan, E-mail: Khalais2012@gmail.com