Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский химико-технологический университет имени Д. И. Менделеева»

На правах рукописи

Чернухин Артём Валерьевич

АВТОМАТИЗИРОВАННАЯ СИСТЕМА ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ПРОМЫШЛЕННОГО ОБОРУДОВАНИЯ НА БАЗЕ МЕТОДОВ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Специальность 2.3.3 – Автоматизация и управление технологическими процессами и производствами

ДИССЕРТАЦИЯ

на соискание ученой степени кандидата технических наук

Научный руководитель:
доктор технических наук,
профессор
Савицкая Татьяна Вадимовна

Москва – 2025

ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ 2

ВВЕДЕНИЕ 4

ГЛАВА 1. АНАЛИЗ СОВРЕМЕННОГО СОСТОЯНИЯ НАУЧНЫХ ИССЛЕДОВАНИЙ В ОБЛАСТИ ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ПРОМЫШЛЕННОГО ОБОРУДОВАНИЯ 11

1.1 Общие подходы к построению автоматизированных систем мониторинга технического состояния промышленного оборудования 11

1.2 Интеллектуальное прогнозирование в условиях цифровой трансформации промышленности 13

1.3 Современные технологии разработки предиктивных моделей технического состояния промышленного оборудования 15

1.3.1 Общие положения прогнозирования 15

1.3.2 Анализ существующих подходов к прогнозированию 17

1.3.3 Изучение подходов, ориентированных на анализ данные 20

1.4 Применение алгоритмов машинного обучения для предсказательной аналитики технического состояния промышленного оборудования 21

1.4.1 Теоретические и прикладные аспекты интеллекта и машинного обучения 21

1.4.2 Обзор традиционных алгоритмов машинного обучения 24

1.4.3 Обзор алгоритмов глубокого обучения 25

1.5 Особенности анализа данных о состоянии промышленного оборудования 31

1.5.1 Общие положения анализа временных рядов 31

1.5.2 Методы и модели анализа временных рядов 32

1.6 Анализ современных подходов к построению автоматизированных систем прогнозирования 35

1.7 Выводы по первой главе 41

ГЛАВА 2. РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СТРУКТУРЫ И МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ПРОМЫШЛЕННОГО ОБОРУДОВАНИЯ 43

2.1. Подходы к построению функциональной структуры автоматизированной системы прогнозирования 43

2.2. Постановка задачи прогнозирования технического состояния промышленного оборудования 48

2.3 Метод прогнозирования неисправностей 48

2.3.1 Цель метода и постановка задачи 49

2.3.2 Описание предлагаемого метода идентификации потенциальных неисправностей оборудования 50

2.4 Метод прогнозирования отказов 62

2.4.1 Цель метода и постановка задачи 62

2.4.2 Описание предлагаемого метода прогнозирования отказов оборудования 63

2.5 Метод прогнозирования RUL ... 72
2.5.1 Цель метода и постановка задачи ... 72
2.6 Выводы по второй главе ... 74
ГЛАВА 3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФУНКЦИОНАЛЬНОЙ СТРУКТУРЫ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ЭКСГАУСТЕРА АГЛОМЕРАЦИОННОЙ МАШИНЫ ... 75
3.1 Объект технической диагностики – эксгаустер агломашины. ... 75
3.1.1 Описание технологии процесса спекания. ... 75
3.1.2 Характеристика и конструкция агломашины ... 77
3.1.3 Основные технологические параметры работы эксгаустера ... 80
3.1.4 Эксгаустер агломашины как объект управления ... 88
3.2 Реализация функциональной структуры автоматизированной системы прогнозирования технического состояния эксгаустера агломашины ... 90
3.2.1 Разработка подсистем импорта данных ... 92
3.2.2. Разработка подсистемы хранения и обработки данных ... 94
3.2.3. Разработка подсистемы управления моделями ... 98
3.3. Организация гибкой архитектуры системы ... 102
3.4 Программная реализация разработанных алгоритмов на Python ... 109
3.5. Программная реализация подсистемы отображения данных ... 114
3.6 Выводы по третьей главе ... 121
ГЛАВА 4. ТЕСТИРОВАНИЕ РАЗРАБОТАННОЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ДЛЯ ЗАДАЧИ КОНТРОЛЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ЭКСГАУСТЕРА АГЛОМАШИНЫ ... 123
4.1 Предобработка данных ... 123
4.2 Апробация метода прогнозирования неисправного состояния оборудования ... 131
4.2.1 Формирование тестовой выборки ... 132
4.2.2 Результаты тестирования ... 133
4.3 Апробация метода прогнозирования отказов оборудования ... 136
4.3.1 Формирование тестовой выборки ... 136
4.3.2 Результаты тестирования ... 138
4.4 Апробация метода прогнозирования остаточного ресурса оборудования ... 142
4.5 Тестирование автоматизированной системы ... 152
4.5 Выводы по четвертой главе ... 156
ЗАКЛЮЧЕНИЕ ... 158
СПИСОК ИСПОЛЬЗУЕМЫХ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ ... 160
СПИСОК ЛИТЕРАТУРЫ ... 162
Приложение 1. Акт о внедрении ООО «ГИСвер Интегро» ... 186

ВВЕДЕНИЕ

Актуальность темы исследования. Разработка автоматизированной системы прогнозирования технического состояния промышленного оборудования на основе методов искусственного интеллекта (ИИ) приобрела особую актуальность в последние годы в связи с рядом факторов.

Во-первых, непрерывно возрастающая сложность устройства современного промышленного оборудования, характеризующаяся растущим числом датчиков, исполнительных механизмов и систем управления, делает классический мониторинг и анализ технического состояния оборудования в реальном времени все более трудоемким, а в некоторых случаях и невозможным без использования решений на основе методов ИИ. Помимо этого, внедрение Индустрии 4.0 и глобальная цифровизация привели к значительному увеличению объема данных, генерируемых промышленным оборудованием. Следовательно, автоматизированные системы прогнозирования на основе ИИ могут обработать и проанализировать эти данные для выявления паттернов, трендов и аномалий, что позволяет заблаговременно проводить профилактическое обслуживание и уменьшать простои. Также, разработка автоматизированной системы прогнозирования позволит промышленности снизить затраты, связанные с обслуживанием, ремонтом и заменой оборудования. Компании, которые внедряют системы прогнозирования на основе ИИ, могут получить конкурентное преимущество, повышая общую эффективность и производительность, промышленную и экологическую безопасность, идентифицируя потенциальные отказы оборудования, которые могут привести к авариям или экологическим катастрофам. Наконец, немаловажным фактором актуальности данной задачи является то, что непредвиденные неисправности и отказы оборудования на опасных производственных объектах могут привести к серьезным травмам или даже человеческим жертвам. Таким образом, данная задача является крайне актуальной и массовое внедрение подобных систем на предприятиях приведет

к значительному росту промышленного потенциала государства в целом.

Степень разработанности темы. Теоретические основы математических методов, компьютерного моделирования и оптимизация режимов сложных технологических систем разработаны в трудах: академика АН СССР и РАН РФ Кафарова В. В., академика РАН РФ Мешалкина В. П., профессоров Егорова А. Ф., Савицкой Т. В.

Существенный вклад в разработку и изучение методов интеллектуальной поддержки принятия решений при управлении техническим состоянием оборудования внесли Н.И. Уткин, А.А. Жданович, А.Н. Бажинов, И.Г. Овчинникова, В.Л. Чугреев, С.М. Боровников, М.С. Мангалова, М. Липатов, Д.Луи (J. Lui), П.Баралди (P. Baraldi), С.Лиу (S. Liu), З. Ванг (Z. Wang), А. Вув (A. Wuw). Работы по надёжности химико-технологических систем Б.В. Палюха, М.Н. Краснянского, Д.Ю. Муромцева, В. С. Шубина, Ю. А. Рюмина внесли большой вклад в развитие направления. Широко известны работы в области совершенствования методов диагностики и управления техническим состоянием сложной техники на основе технологий нейронных сетей следующих авторов: С.В. Жернаков, А.М. Пашаев, А.Г. Кучер, П.И. Раков, Г.С. Бабу (G.S. Babu), К. Джанг (X. Zhang), К. Ли (Xiang Li), Ж. Ли (Zhe Li), Р. Хуанг (R. Huang). Отметим разработки коммерческих компаний General Electrics, Siemens, Pratt&Whitney, ЗАО РОТЕК, NASA, детальные результаты исследований которых не обнародованы, так как являются основанием для конкурентного преимущества. Наличие теоретических и практических работ российских и зарубежных ученых и исследователей подтверждает актуальность темы и характеризует определенную степень ее разработанности. Тем не менее, недостаточная научная проработанность эффективных методов, моделей и алгоритмов прогнозирования технического состояния и оценки остаточного ресурса оборудования на базе методов искусственного интеллекта определяет необходимость дополнительных исследований.

В качестве объекта исследования выступает эксгаустер – промышленное оборудование, работающее в непрерывном режиме, часть агломерационной

машины, представляет собой вентилятор (насос), основным назначением которого является удаление дымовых газов, паров и других вредных примесей.

Предметом исследования является автоматизированная система прогнозирования технического состояния промышленного оборудования.

Цель и задачи исследования. Цель исследования – разработка автоматизированной системы прогнозирования технического состояния промышленного оборудования на базе методов искусственного интеллекта.

Для достижения цели поставлены следующие задачи:

1. Проанализировать отечественные и зарубежные аналоги.
2. Разработать новую гибкую функциональную структуру автоматизированной системы прогнозирования технического состояния промышленного оборудования.
3. Разработать модели прогнозирования на основе методов искусственного интеллекта, которые смогут предсказывать возможные неисправности и отказы промышленного оборудования. Для этого требуется провести обучение моделей на исторических данных о неисправностях, отказах и рабочем состоянии оборудования.
4. Провести анализ объекта технической диагностики и собрать набор данных с информацией о неисправностях и отказах промышленного оборудования за большой временной интервал с низкой долей неисправностей, провести обработку данных.
5. Реализовать функциональную структуру автоматизированной системы прогнозирования, в том числе разработать веб-сервис, реализующий возможности хранения и обработки данных в режиме реального времени, имеющий удобный интерфейс для работы конечных пользователей.
6. Внедрить в систему функциональные возможности переобучения моделей на больших объемах непрерывно поступающих актуальных данных в автоматическом режиме без привлечения разработчиков.
7. Провести всестороннее тестирование работы системы в приближенных к реальности условиям.

Методы исследования. Решение поставленных в диссертации задач основано на использовании современных методов и теорий: модели разработаны на базе методов искусственного интеллекта, а также реализована гибкая функциональная структура системы, предполагающая автоматизацию процессов сбора и обработки большого массива данных о состоянии промышленного оборудования.

Научная новизна. В процессе выполнения диссертации впервые были получены новые научные результаты:

1. Проведен анализ современного состояния исследований в области прогнозирования технического состояния промышленного оборудования, показавший высокую актуальность разработки автоматизированной системы прогнозирования технического состояния промышленного оборудования на базе методов искусственного интеллекта.

2. Разработана новая функциональная структура автоматизированной системы прогнозирования технического состояния промышленного оборудования, включающая в себя подсистемы импорта данных, хранения и обработки данных, прогнозирования, отображения данных и уникальную подсистему управления моделями, позволяющую дообучать модели с учетом динамически изменяющихся условий производственного процесса.

3. Разработан инновационный подход к анализу и обработке данных о состоянии промышленного оборудования, учитывающий особенности автоматизированных систем прогнозирования, заключающийся в обработке данных в режиме реального времени с высокой скоростью и точностью с учетом их физической природы.

4. Предложен алгоритм диагностики технических систем, основанный на ансамблевом подходе, который позволяет создавать и обучать высокоточные модели прогнозирования неисправностей и отказов промышленного оборудования.

5. Предложен и реализован новый алгоритм прогнозирования остаточного срока службы оборудования, основанный на современных

архитектурах нейронных сетей, показавший высокие метрики точности для горизонта прогнозирования, равного одному месяцу.

6. Реализована функциональная структура автоматизированной системы, предоставляющей пользователю доступ к прогнозам и рекомендациям по предотвращению неисправностей и отказов, способной в автоматическом режиме без привлечения разработчиков обучаться на накапливающемся массиве данных.

7. Проведено тестирование автоматизированной системы прогнозирования технического состояния промышленного оборудования в приближенных к реальности условиях, подтвердившее высокую эффективность её использования для решения задач контроля технического состояния эксгаустера агломашины.

Теоретическая значимость заключается в развитии методов построения интеллектуальных систем автоматизированного прогнозирования неисправностей промышленного оборудования.

Практическая значимость заключается в том, что разработанная автоматизированная система прогнозирования представляет большой интерес для разработчиков комплексных автоматизированных систем управления производственным процессом, так как выполняет функции сбора, хранения и обработки большого объема информации о состоянии промышленного оборудования, с высокой точностью прогнозирует его неисправности в режиме реального времени, а также обладает гибкой функциональной структурой.

Система способна оперативно предоставлять лицам, принимающим решение доступ к прогнозам и рекомендациям по предотвращению неисправностей и отказов, основанным на большом массиве данных, который оператор не может эффективно обработать самостоятельно. Это достигается в том числе за счет возможности переобучения моделей в автоматическом режиме без привлечения разработчиков на накапливающемся массиве данных.

Положения, выносимые на защиту:

- математическая модель, основанная на «мягком голосовании» между

алгоритмом логистической регрессии, методом опорных векторов и сверточной нейронной сетью, способная с высоким качеством решать задачу прогнозирования неисправностей промышленного оборудования;

- математическая модель, основанная на алгоритме изолирующего леса, используемого для поиска аномальных режимов работы, а также нейронной сети краткосрочной долговременной памяти LSTM (Long short-term memory) в качестве классификатора, способная с высоким качеством решать задачу прогнозирования отказов промышленного оборудования;

- автоматизированная система прогнозирования неисправностей и отказов промышленного оборудования, обладающая уникальной гибкой функциональной структурой и включающая подсистемы импорта, хранения и обработки данных, прогнозирования, управления моделями и отображения;

результаты апробации автоматизированной системы на задачах, близких к реальным, подтверждающие высокую стабильность работы системы.

Соответствие паспорту научной специальности. Основная область исследования соответствует паспорту специальности 2.3.3 «Автоматизация и управление технологическими процессами и производствами», а именно пунктам: 6 – «Научные основы и методы построения интеллектуальных систем управления технологическими процессами и производствами», 11 – «Методы создания, эффективной организации и ведения специализированного информационного и программного обеспечения АСУТП, АСУП, АСТПП и др., включая базы данных и методы их оптимизации, промышленный интернет вещей, облачные сервисы, удаленную диагностику и мониторинг технологического оборудования, информационное сопровождение жизненного цикла изделия», 15 – «Теоретические основы, методы и алгоритмы диагностирования (определения работоспособности, поиск неисправностей и прогнозирования) АСУТП, АСУП, АСТПП и др.»

Обоснованность и достоверность. Степень достоверности полученных результатов обеспечивается применением апробированного математического аппарата, корректным использованием исходных данных, согласованностью

полученных результатов с результатами работ других исследователей.

Апробация результатов работы. Основные результаты диссертационной работы доложены на III Международной научно-практической конференции «Компьютерные приложения для управления и устойчивого развития производства и промышленности» (Душанбе, Таджикистан, 2023); Международной конференции «Scientific research of the SCO countries: synergy and integration: Proceedings of the International Conference» (Пекин, Китай, 2023 г.); Международном университетском научном Форуме «Practice Oriented Science: UAE - Russia - India» (Дубай, ОАЭ, 2024); Межвузовском международном конгрессе «Высшая школа: научные исследования» (Москва, 2024), Международном научном форуме «Научный диалог: теория и практика» (Москва, 2025).

Публикации. По теме диссертационной работы опубликовано 8 работ, из них 3 статьи – в изданиях, рекомендованных ВАК России для публикации научных результатов на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук по специальности 2.3.3. Автоматизация и управление технологическими процессами и производствами и 2 работы в зарубежных изданиях, индексируемых в международной базе научного цитирования Scopus.

Личный вклад автора. В совместных публикациях результаты, связанные с разработкой и обучением моделей прогнозирования неисправностей и отказов, проектированием и программной реализацией автоматизированной системы прогнозирования технического состоянии промышленного оборудования и проведением комплексного тестирования работы системы получены автором лично.

Структура и объем диссертации. Диссертационная работа состоит из введения, четырех глав, заключения, списка литературы и приложений. Общий объем работы составляет 186 страницу. Работа иллюстрирована 65 рисунками, содержит 7 таблиц и 1 приложение. Список литературы включает в себя 177 наименования.

ГЛАВА 1. АНАЛИЗ СОВРЕМЕННОГО СОСТОЯНИЯ НАУЧНЫХ ИССЛЕДОВАНИЙ В ОБЛАСТИ ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ПРОМЫШЛЕННОГО ОБОРУДОВАНИЯ

1.1 Общие подходы к построению автоматизированных систем мониторинга технического состояния промышленного оборудования

Ключевым аспектом, определяющим конкурентоспособность предприятий в сфере машиностроения, является качество производимой продукции. Данный параметр формируется под воздействием множества факторов на всех стадиях производственного цикла. К числу этих факторов относятся: качество проектной документации, характеристики сырья и материалов, технологии и организация производственного процесса, техническое состояние оборудования, уровень квалификации как производственного, так и управленческого персонала, а также системы контроля качества как технологических процессов, так и готовой продукции. Реализация сквозного мониторинга производства в режиме реального времени формирует замкнутый управленческий контур: данные о работе оборудования, ходе обработки деталей и качестве продукции автоматически анализируются, что исключает субъективные ошибки и позволяет мгновенно корректировать процессы [1].

В современных промышленных предприятиях наблюдается значительный переход от традиционных концепций автоматизации отдельных машин и процессов к более комплексной модели «Индустрия 4.0». Эта модель представляет собой многоуровневую и сложную технологическую и организационную систему, которая предполагает интеграцию различных операций и сопутствующих процессов в единую информационную среду [2-3]. Иерархическая структура цифрового машиностроительного производства включает несколько уровней, каждый из которых оснащен автоматизированными информационно-управляющими системами, направленными на решение специфических задач. Данные системы

охватывают управление производственными процессами, операционный мониторинг, подготовку производства, а также финансовые и кадровые модули, интегрированные для слаженной работы предприятия [4].

Однако на российских производствах сохраняется ключевая проблема — сложность объединения разрозненного оборудования и систем управления в единый цифровой контур. Это связано с тем, что разнообразие технологического оборудования, производимого различными компаниями и использующего разные протоколы взаимодействия, затрудняет контроль за его работой и агрегацию информации для передачи на более высокие уровни управления [5]. Разнородность технологического оборудования, обусловленная различиями в стандартах и протоколах обмена данными, существенно затрудняет его полную синхронизацию в рамках единой системы. В ряде случаев это приводит к принципиальной невозможности совместной работы, что ограничивает разработку гибких производственных систем, интегрирующих оборудование различных типов.

Существующие на рынке системы мониторинга технологических процессов, такие как InTouch (Wonderware, США), WinCC (Siemens, Германия), АИС Диспетчер (Станкосервис, Россия) и Citect (CI Technology, Австралия), не способны в полной мере решить данную проблему. Большинство из них являются закрытыми решениями, работающими с ограниченным набором промышленных протоколов и предназначенными преимущественно для контроля оборудования одного производителя, что ограничивает их применение в комплексных системах [6-7].

Повышение качества выпускаемой продукции и оптимизация производственных процессов требуют внедрения современных автоматизированных систем контроля. Критически важным элементом таких систем становятся решения для мониторинга оборудования и технологических параметров, обеспечивающие оперативную обратную связь [8]. Ключевыми задачами при этом выступают: унификация данных от разрозненных систем управления, использующих различные промышленные протоколы, а также

разработка промежуточного программно-аппаратного комплекса для агрегации производственных данных и их интеграции в корпоративную систему управления [9].

В настоящее время задача построения комплексной системы мониторинга на многих крупных отечественных предприятиях решена, но возникла совершенно новая – на базе этих систем возможно построение прогнозирования технического состояния оборудования.

1.2 Интеллектуальное прогнозирование в условиях цифровой трансформации промышленности

Проведённый обзор исследований [10–14] демонстрирует доминирующую роль прогнозного технического обслуживания среди современных стратегий эксплуатации промышленных активов. Динамика публикационной активности указывает на устойчивый тренд роста интереса к данной методологии.

Прогнозное обслуживание (Predictive Maintenance) представляет собой усовершенствованную концепцию технического обслуживания по состоянию, при котором работы выполняются только при наличии объективных показаний. Хотя теоретические основы этого подхода были заложены несколько десятилетий назад, его практическая реализация и доказательство эффективности стали возможны лишь в последние годы благодаря стремительному развитию цифровых технологий.

Ключевыми факторами, способствовавшими активному внедрению прогнозного обслуживания, стали распространение промышленного интернета вещей (IIoT), развитие технологий искусственного интеллекта и машинного обучения, рост вычислительных мощностей и возможностей облачных вычислений, а также расширение рынка больших данных. Эти технологические достижения создали принципиально новые возможности для точного прогнозирования состояния оборудования и анализа производственных данных, что особенно важно при управлении современными

сложными промышленными активами [15-16].

Актуальность и востребованность прогнозного обслуживания подтверждаются последними отраслевыми тенденциями. Характерным примером является решение Пентагона (США), принятое в ноябре 2018 года, о масштабном внедрении системы прогнозного технического обслуживания на основе алгоритмов машинного обучения для обслуживания вооружений армии и флота [14]. Пример показывает, как передовые методы технического обслуживания переходят из теории в практику в различных отраслях.

В четвертом квартале 2018 года национальный авиаперевозчик "Аэрофлот" инвестировал 50 миллионов рублей в разработку пилотного проекта программного обеспечения для интеллектуального мониторинга и предиктивного анализа технического состояния авиапарка с использованием технологий Big Data Analytics. Данная инициатива ознаменовала собой первый крупный коммерческий проект в сфере предиктивной аналитики в российской авиационной отрасли [17].

Прогнозное техническое обслуживание представляет собой интеллектуальный подход к повышению эксплуатационной готовности оборудования, объединяющий методы раннего выявления неисправностей, прогнозирования технического состояния и принятия обоснованных решений о необходимости вмешательства. Данная стратегия позволяет существенно сократить количество необоснованных ремонтов, оптимизировать использование ресурсов и продлить межремонтные интервалы за счет внедрения специализированных прогностических систем.

Основной принцип такого обслуживания заключается в выполнении исключительно необходимых операций, что обеспечивает значительную экономию на запасных частях, сокращает простои оборудования и уменьшает трудозатраты на техническое обслуживание. Реализация концепции строится на трех ключевых этапах: непрерывном мониторинге параметров оборудования, анализе данных для диагностики текущего состояния и прогнозирования возможных отказов, а также на принятии обоснованных

решений о сроках и объемах технического обслуживания [18].

Фундаментом для успешного внедрения данной методологии служит разработка точных прогностических моделей, позволяющих своевременно предупреждать потенциальные отказы и оптимально планировать обслуживающие мероприятия. Такой подход обеспечивает переход от планово-предупредительного ремонта к обслуживанию по фактическому состоянию оборудования, что значительно повышает экономическую эффективность эксплуатации промышленных активов.

1.3 Современные технологии разработки предиктивных моделей технического состояния промышленного оборудования

1.3.1 Общие положения прогнозирования

Работа промышленного оборудования осуществляется под влиянием множества факторов, включающих как детерминированные параметры (конструктивные особенности, заданные режимы эксплуатации), так и случайные воздействия (внешние условия). К стохастическим факторам относятся: тепловые и механические нагрузки, характеристики материалов, точность изготовления деталей, качество технического обслуживания, а также вибрационные и температурные воздействия. Совокупное влияние этих факторов вызывает постепенную деградацию технического состояния оборудования, что в пределе может привести к его отказу — потере работоспособности.

Анализ научных источников показывает, что прогнозирование технического состояния оборудования представляет собой комплексный процесс, объединяющий математические методы и алгоритмы обработки данных. В его основе лежит системный анализ исторических показателей, текущих рабочих параметров и функциональных взаимосвязей между компонентами технической системы.

Данный подход решает три взаимосвязанные задачи. Он позволяет не только выявлять ранние признаки возможных неисправностей и прогнозировать их эскалацию до критических отказов, но и точно оценивать

остаточный эксплуатационный ресурс оборудования. Кроме того, система обеспечивает постоянный мониторинг степени износа и деградации технических характеристик, анализируя динамику изменения рабочих параметров.

Ключевой особенностью таких прогностических систем является их способность учитывать сложные взаимозависимости между различными эксплуатационными параметрами и их совокупное влияние на общее состояние оборудования. Это позволяет перейти от реактивного обслуживания к превентивному управлению техническим состоянием промышленных активов.

Следовательно, прогнозирование технического состояния оборудования включает оценку его остаточного срока службы (RUL). RUL является важнейшей метрикой, которая показывает, как долго отдельные производственные единицы и сложные технические системы могут безопасно работать. Она относится к периоду с момента оценки технического состояния оборудования до его отказа, представляя собой эксплуатационный буфер, в течение которого его основные технические и эксплуатационные характеристики соответствуют нормативным и техническим стандартам. Метрики оценки остаточного ресурса оборудования варьируются в зависимости от типа техники и характера ее эксплуатации. Так, для транспортных средств ключевым показателем становится пробег в километрах, тогда как для вращающихся механизмов (подшипников, шпинделей) - количество совершённых оборотов. В авиационной технике принято учитывать наработанные лётные часы или количество взлёт-посадочных циклов. Аналогично, для промышленного оборудования могут применяться такие единицы измерения как тонны переработанного материала, кубические метры произведённой продукции или гектары обработанной площади. Выбор конкретного параметра напрямую зависит от эксплуатационных характеристик и принципа работы технического устройства. Точное определение RUL позволяет принимать оптимальные решения относительно стратегий

технического обслуживания, обеспечивая баланс между расходами, связанными с техническими операциями, и потенциальными потерями, понесенными из-за отказов оборудования.

Прогнозирование технического состояния играет ключевую роль в предотвращении аварийных ситуаций и оптимизации эксплуатационных ресурсов оборудования. Грамотно реализованная система предсказательной аналитики позволяет существенно сократить простои, продлить жизненный цикл техники и минимизировать эксплуатационные расходы благодаря переходу от реактивного к превентивному обслуживанию.

Однако, несмотря на значительный прогресс в методах диагностики и прогнозирования, достигнутый за последние десять лет, надежные системы превентивного мониторинга технического состояния пока не получили широкого распространения в промышленной практике. Современные прогностические решения остаются скорее исключением, чем общепринятым стандартом эксплуатации производственного оборудования. Методологическая сложность идентификации потенциальных отказов обусловлена мультипараметричностью контролируемых характеристик в современных технических комплексах, что показывает низкую эффективность традиционных прогностических подходов [19].

1.3.2 Анализ существующих подходов к прогнозированию

В контексте решения комплексных задач превентивной диагностики и верификации остаточного ресурсного потенциала технологического оборудования научным сообществом разработан обширный методологический спектр моделей, определяющих процессы деградации технических систем. Представленные концептуальные модели характеризуются дифференциацией по типологии исходной информативной базы, методологическим стратегиям аналитической обработки и алгоритмам репрезентации результативных кластеров [20]. Современная таксономия позволяет систематизировать широкий спектр методологических подходов, что наглядно отражено на рисунке 1.1:

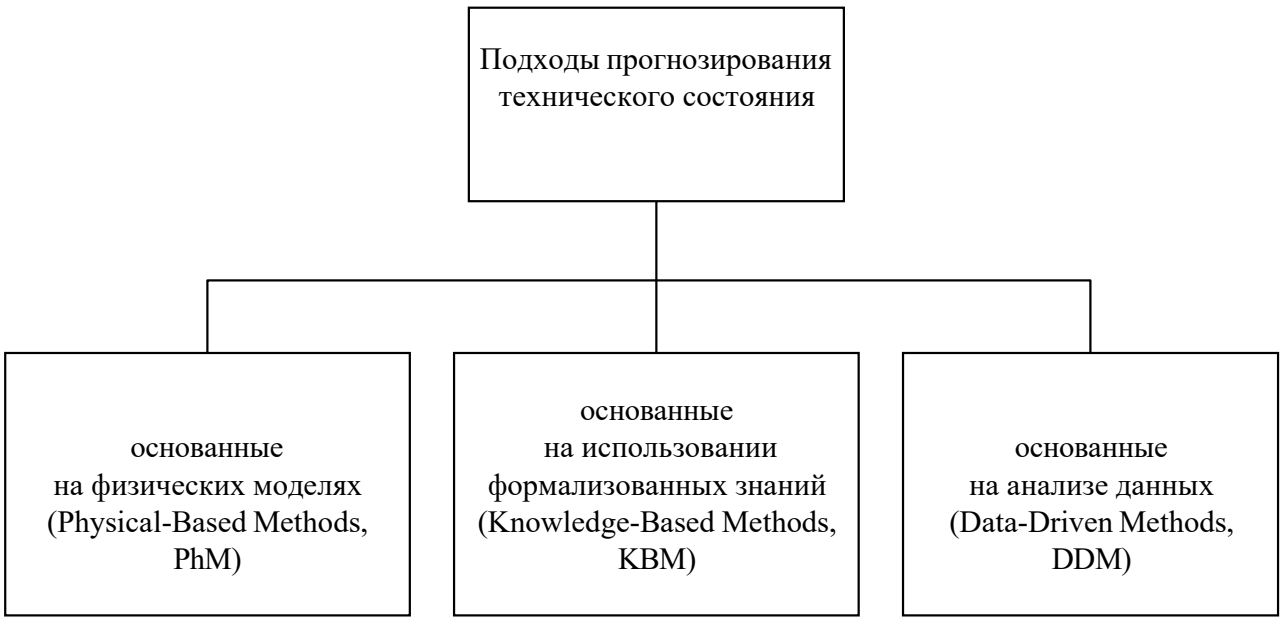


Рисунок 1.1 – Классификация методов прогнозирования отказов оборудования

Физическое моделирование (PhM) — это методологический подход, базирующийся на математическом описании физико-химических процессов износа оборудования, включая усталостные разрушения и механическую деградацию. Данная методика обеспечивает расчет показателей надежности с учетом свойств материалов, конструктивных параметров и технологических особенностей изготовления. Однако точность получаемых моделей существенно зависит от степени изученности механизмов отказов и принципов функционирования конкретной технической системы [21].

Несмотря на теоретическую обоснованность, практическая реализация физического моделирования сталкивается с существенными сложностями. Разработка адекватных физических моделей требует значительных временных и интеллектуальных затрат, что часто делает их применение затруднительным в условиях реального производства [22]. Следует отметить, что традиционные физические модели имеют существенное ограничение - их прогностическая способность снижается из-за невозможности полноценного учета всего спектра реальных эксплуатационных факторов при оценке остаточного ресурса оборудования.

Моделирование на основе знаний (KBM) опирается на экспертный опыт для поддержки принятия решений, используя такие методы как экспертные

системы и нечеткую логику [23-24]. Данный подход имитирует рассуждения специалистов через систему формализованных правил, однако его эффективность ограничена рамками заложенных знаний. Главные сложности заключаются в принципиальной трудности перевода экспертного опыта в формальные правила, что неизбежно приводит к появлению неопределенностей. При расширении базы знаний система требует значительных вычислительных ресурсов, что негативно сказывается на скорости работы [25]. Кроме того, подобные системы изначально создаются для решения узкого круга специализированных задач, что ограничивает область их применения. Нечеткая логика представляет собой перспективный подход к формализации знаний, нашедший применение в различных технических областях [26]. В отличие от классической бинарной логики, она оперирует градациями истинности, что позволяет описывать состояния системы через интуитивно понятные лингвистические переменные (например, "Низкий", "Нормальный", "Высокий"). Такой подход особенно ценен для операторов, поскольку преобразует числовые данные в качественные оценки. Однако разработка адекватных систем правил и функций принадлежности для сложных технических систем остается трудоемкой задачей.

Тем не менее, нечёткая логика эффективно сочетается с другими подходами, включая экспертные системы и нейронные сети, при моделировании процессов деградации оборудования. В отличие от data-driven моделирования (DDM), ориентированного исключительно на анализ данных мониторинга (вибрации, температура, давление и др.), применение нечёткой логики предполагает предварительную формализацию экспертных знаний.

DDM, в свою очередь, демонстрирует растущую эффективность благодаря применению статистических методов и машинного обучения. Его ключевое преимущество — универсальность, поскольку он не требует детального понимания физики системы или сложных взаимосвязей между ее компонентами, а полагается на выявление закономерностей непосредственно в данных.

1.3.3 Изучение подходов, ориентированных на анализ данные

Статистические методы оценки технического состояния оборудования основаны на вероятностных моделях, анализирующих совокупность эксплуатационных данных. Модели формируются на основе трех типов информации: лабораторных испытаний, фактических рабочих параметров и статистики обслуживания. Ведущие методики включают стохастическую фильтрацию, скрытые марковские модели и винеровские процессы, которые выявляют статистические зависимости для точной диагностики и прогнозирования состояния техники.

Критически важно отметить, что применение статистических методов может приводить к значительным ошибкам в условиях недостаточности данных, и глубокое понимание природы этих данных является решающим фактором для успешного анализа. Указанные методы подвергаются критике за их зависимость от теоретических распределений (таких как Винеровское или Гамма-распределение), которые часто не соответствуют реальным условиям эксплуатации оборудования [27]. Более того, эти подходы обладают принципиальным ограничением - они способны анализировать лишь линейные взаимосвязи, оказываясь несостоятельными при моделировании сложных нелинейных процессов в технических системах.

Эти ограничения стимулировали активное развитие методов Data-Driven Modeling (DDM), базирующихся на машинном обучении. Их растущая популярность обусловлена совокупностью факторов: появлением высокоэффективных алгоритмов (включая современные версии бустинга [28]), промышленной потребностью в повышении эффективности через цифровизацию [29], а также удешевлением технологий сбора и обработки данных [30]. DDM-подходы демонстрируют принципиально иной уровень гибкости - они поддерживают онлайн-обновление моделей, способны учитывать сложные нелинейные зависимости и эффективно дополняют традиционные статистические методы, особенно при работе с динамическими техническими системами со сложными взаимодействиями компонентов.

1.4 Применение алгоритмов машинного обучения для предсказательной аналитики технического состояния промышленного оборудования

1.4.1 Теоретические и прикладные аспекты интеллекта и машинного обучения

В эпоху Четвертой промышленной революции искусственный интеллект превратился в стратегически важную технологическую платформу, определяющую вектор развития "Индустрии 4.0". Современные ИИ-системы демонстрируют способность решать сложные когнитивные задачи, включая обработку естественного языка, логический вывод и адаптивное принятие решений, что сближает их функциональность с человеческим мышлением. Особую ценность представляет присущая этим системам способность к постоянному обучению и самооптимизации. Многообразие разработанных за последние десятилетия методов машинного обучения отражает динамичное развитие этой перспективной области знаний. Среди них выделяются машинное обучение (Machine Learning) и глубокое обучение (Deep Learning) (рисунок 1.2).

Рисунок 1.2 – Искусственный интеллект, машинное и глубокое обучение

Машинное обучение представляет собой совокупность алгоритмических подходов в области искусственного интеллекта, отличающихся тем, что они не решают задачи непосредственно, а обучаются на основе анализа множества аналогичных примеров. Глубокое обучение, являющееся одной из подкатегорий машинного обучения, опирается на архитектуры многослойных искусственных нейронных сетей. В текущий момент алгоритмы глубокого обучения занимают лидирующие позиции среди моделей машинного обучения

и функционируют по принципам, схожим с механизмами работы биологических нейронных структур.

Современные производственные системы достигают беспрецедентного уровня сложности, что делает традиционные подходы к диагностике и прогнозу отказов недостаточно эффективными [31]. Параллельно наблюдается экспоненциальный рост объемов данных от промышленных сенсоров, открывающий новые возможности для анализа. Методы машинного обучения, доказавшие свою эффективность в таких разнородных сферах как биометрическая идентификация [32], интеллектуальный веб-поиск [33], медицинская диагностика [34] и игровые системы [35], демонстрируют особую ценность для задач промышленного мониторинга.

Адаптированный для технической диагностики процесс машинного обучения (рисунок 1.3) предполагает комплексную обработку множества взаимосвязанных эксплуатационных параметров, что позволяет получить многоаспектную оценку состояния оборудования. Результаты такого анализа включают прогнозирование потенциальных отказов, выявление аномальных рабочих режимов и расчет остаточного ресурса. Теоретической основой этих оценок служат выявленные корреляционные зависимости между наблюдаемыми параметрами работы и фактическим техническим состоянием оборудования [31-35]. Задача заключается в реконструкции этой связи из исходных данных, что влечет за собой создание алгоритма, который может обеспечить надежную и точную оценку состояния оборудования на основе определенного набора эксплуатационных характеристик [36].

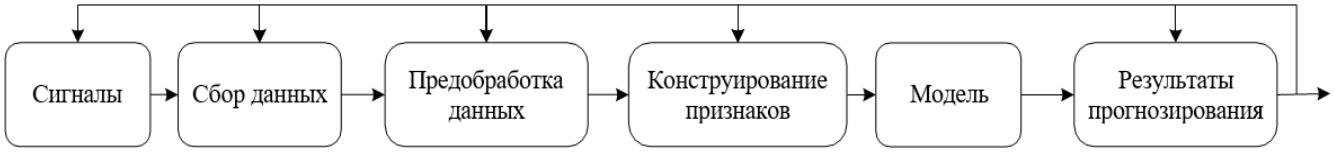


Рисунок 1.3 – Этапы машинного обучения

Суть машинного обучения как направления ИИ заключается в автоматическом анализе данных для выявления значимых закономерностей. Этот метод предполагает, что алгоритмы анализируют подготовленные наборы

данных, самостоятельно выявляют скрытые взаимосвязи и строят математические модели, которые затем используются для прогнозирования на новых данных. В отличие от традиционного программирования с жесткими правилами, машинное обучение создает адаптивные модели, способные улучшать свои прогностические способности по мере обработки новой информации. Суть подхода заключается в способности алгоритмов обучаться на примерах, выявлять сложные паттерны и применять полученные знания для решения практических задач без явного указания инструкций. Алгоритмы машинного обучения можно описать как обучение целевой функции f, которая наилучшим образом соотносит входные переменные X и выходную переменную Y: $Y = f(X)$.

Современные методы интеллектуального анализа данных для прогнозирования состояния оборудования можно классифицировать по трем ключевым направлениям. Первое направление охватывает supervised learning (обучение с учителем), где модели строятся на размеченных данных с известными целевыми значениями. Второе направление включает unsupervised learning (обучение без учителя), ориентированное на выявление скрытых закономерностей в данных без заранее известных ответов. Третье направление представляет собой semi-supervised learning (частично обучение с учителем), комбинирующее преимущества первых двух подходов за счет использования как размеченных, так и неразмеченных данных.

При обучении с учителем на вход модели подаются наборы входных сигналов X (признаковое описание системы), для которых заранее известны правильные ответы Y. Совокупность пар $(x_i, y_i)^N$ называется обучающим множеством.

В машинном обучении с учителем используется обучающая выборка, содержащая пары "входные данные - соответствующий выход". Цель обучения состоит в построении модели, восстанавливающей взаимосвязь между входными и выходными данными, что позволяет предсказывать выходные значения для новых входных данных. В зависимости от типа выходной

переменной Y различают: задачи классификации (когда Y принимает категориальные значения) и задачи регрессии (когда Y является вещественной величиной).

В отличие от этого, обучение без учителя работает с данными, для которых отсутствуют заранее известные выходные значения. Основная цель здесь - обнаружение скрытых структур и закономерностей в данных. К типичным задачам такого обучения относятся: группировка объектов (кластеризация), выявление аномальных наблюдений и установление ассоциативных правил.

Полу-контролируемое обучение представляет собой комбинированный подход, использующий как размеченные (с известными выходами), так и неразмеченные данные. Это особенно полезно, когда размеченных данных недостаточно.

Особое значение во всех типах машинного обучения имеет процесс создания и отбора признаков (Feature Engineering). Этот творческий процесс, сочетающий экспертные знания и автоматизированные методы, включает два основных этапа: генерацию новых информативных признаков и последующий отбор наиболее значимых характеристик, устраняя избыточные и слабоинформативные признаки. Успешное выполнение Feature Engineering существенно повышает качество и интерпретируемость конечных моделей.

1.4.2 Обзор традиционных алгоритмов машинного обучения

Современные алгоритмы машинного обучения демонстрируют высокую эффективность в задачах прогнозирования технического состояния оборудования, что подтверждается многочисленными исследованиями [37-54]. Особого внимания заслуживает метод k-ближайших соседей (KNN), который, основываясь на принципе пространственной близости объектов [55], показывает стабильные результаты, несмотря на существенные вычислительные затраты при обработке больших массивов данных [56].

Метод опорных векторов (SVM), впервые предложенный Кортесом и Вапником [57], зарекомендовал себя как один из наиболее точных алгоритмов

для решения задач классификации и регрессии [58]. Его эффективность значительно повышается при комбинации с методами снижения размерности признакового пространства [59].

В области кластерного анализа широкое применение находит алгоритм k-средних [60], чья популярность объясняется высокой скоростью работы, хотя он и требует априорного знания о количестве кластеров. Среди ансамблевых методов особое место занимает случайный лес (RF) [61], демонстрирующий выдающуюся устойчивость к переобучению даже в условиях высокой размерности данных.

Наиболее впечатляющие результаты в последние годы показывают методы градиентного бустинга, особенно в реализации XGBoost [62]. Эти алгоритмы, основанные на последовательном улучшении слабых классификаторов, установили новые стандарты точности в задачах промышленного прогнозирования.

1.4.3 Обзор алгоритмов глубокого обучения

Искусственные нейронные сети (ANN) представляют собой вычислительные системы, имитирующие принципы работы биологических нейронов. Их исторические корни восходят к 1940-м годам, однако настоящий прорыв произошел лишь в последнее десятилетие. Первоначально развитие технологии сдерживалось отсутствием эффективных алгоритмов обучения, пока в 1980-х не был заново открыт метод обратного распространения ошибки. Знаковым моментом стало исследование Яна Лекуна в 1989 году, где впервые были успешно объединены сверточные архитектуры с алгоритмом обратного распространения для распознавания рукописных цифр [62].

Современные нейросетевые технологии стали ключевым инструментом цифровой трансформации, проникая во все сферы - от управления ресурсами до военных применений [63]. Особый интерес вызывают глубокие нейронные сети, демонстрирующие беспрецедентные результаты в обработке естественного языка, компьютерном зрении и медицинской диагностике [64]. Их главное преимущество заключается в способности автоматически выявлять

сложные паттерны в больших данных, значительно превосходя человеческие возможности в анализе многомерных зависимостей.

Примечательно, что современные достижения стали возможны благодаря сочетанию трех факторов: появлению новых архитектурных решений, росту вычислительных мощностей и доступности больших данных. Это позволило преодолеть ограничения, сдерживавшие развитие нейросетевых технологий на протяжении десятилетий. Математическая модель искусственного нейрона была разработана Уорреном МакКаллоком и Уолтером Питтсом в 1943 году.

В данной модели нейрон обрабатывает входные сигналы x_1, x_2, ..., x_n, умножая каждый на соответствующий весовой коэффициент w_1, w_2, ..., w_n. Затем вычисляется взвешенная сумма входных сигналов $\Sigma(w_i \cdot x_i)$ с добавлением смещения b. Полученное значение передается через нелинейную функцию активации φ, которая преобразует линейную комбинацию входов в выходной сигнал нейрона $y = \varphi(\Sigma(w_i \cdot x_i) + b)$ [65] (рисунок 1.4).

Эта фундаментальная архитектура, несмотря на свою простоту, заложила основу для всех последующих разработок в области нейронных сетей. Важно отметить, что функция активации вводит в модель необходимую нелинейность, позволяя сети обучаться сложным закономерностям.

Первоначальная модель МакКаллока-Питтса использовала пороговую функцию активации, однако современные нейронные сети применяют более сложные функции, такие как сигмоида, гиперболический тангенс или ReLU.

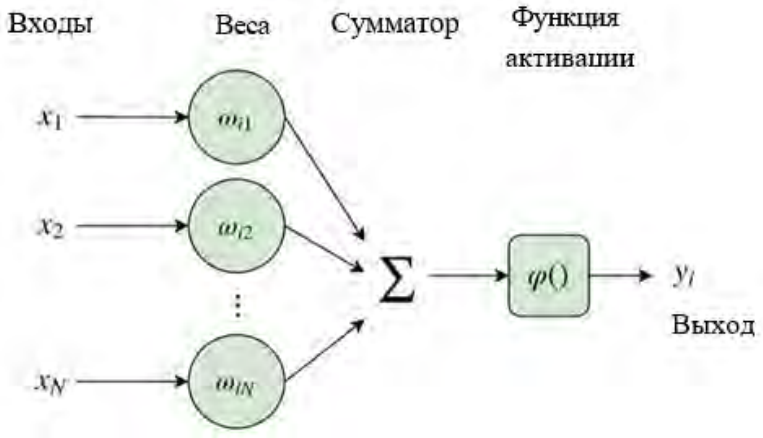


Рисунок 1.4 – Стандартная модель искусственного нейрона с нескольким входами (x_k), весовыми значениями (w_{ik}), где взвешенная сумма передается через функцию активации

φ

По сути, каждый нейрон выполняет взвешенную обработку входных сигналов, где весовые коэффициенты определяют степень влияния каждого входа на итоговый результат. Наиболее элементарная архитектура нейросети представляет собой прямое соединение входного слоя с выходным, однако подобная структура обладает ограниченными возможностями и не способна решать сложные задачи.

Для преодоления этого ограничения применяются многослойные структуры со скрытыми слоями обработки (рисунок 1.5). Ключевая особенность классических искусственных нейронных сетей (ANN) заключается в полносвязной организации - выходные сигналы каждого нейрона становятся входными данными для всех нейронов последующего слоя, что обеспечивает комплексную обработку информации.

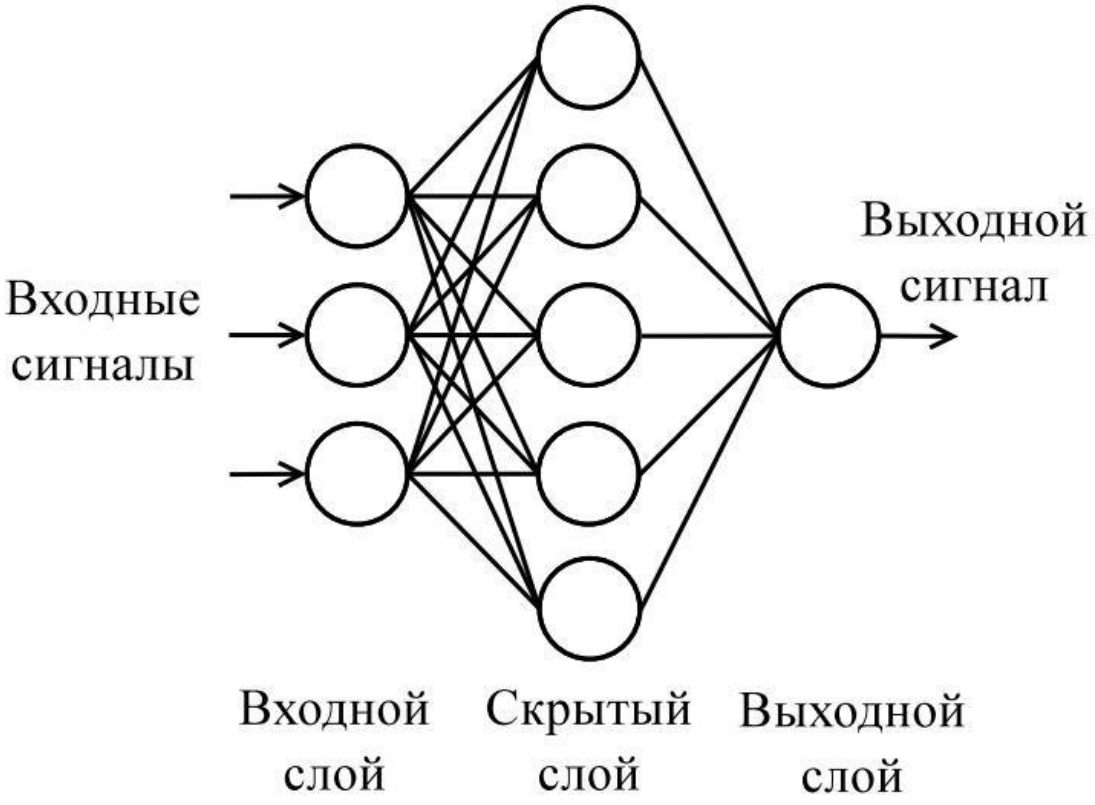
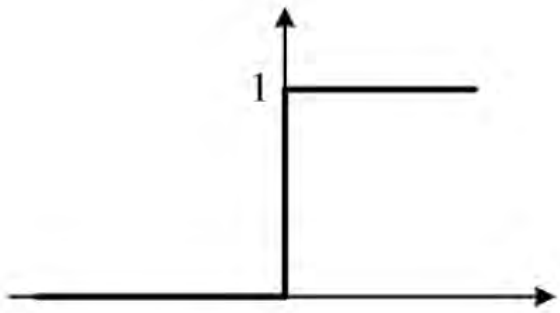


Рисунок 1.5 – Многослойная нейронная сеть с одним скрытым слоем из пяти нейронов

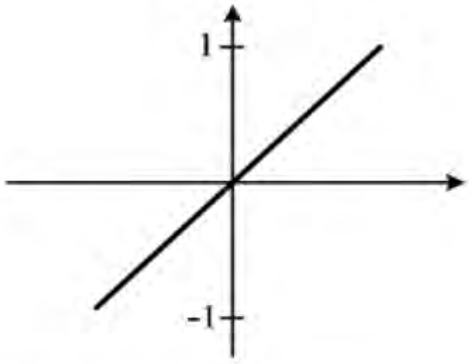
Выход y определяется функцией $f(x)$, где $f(x) = f^1(f^2(\dots f^n(x) \dots))$ и f^i – это функция в i-том слое, состоящем из весов w_i. В ходе обучения нейронной сети осуществляется циклическая подстройка весовых параметров w в каждом слое с целью постепенного уменьшения значения функции потерь. Этот

процесс можно рассматривать как оптимизационную задачу, где на каждом шаге вычисляется градиент функции ошибки относительно весов и выполняется их корректировка для постепенного уменьшения расхождения между предсказанными и истинными значениями. Функция потерь выступает в качестве критерия качества работы сети, количественно оценивающего степень отклонения текущих предсказаний от желаемых результатов. $L(x, w)$.

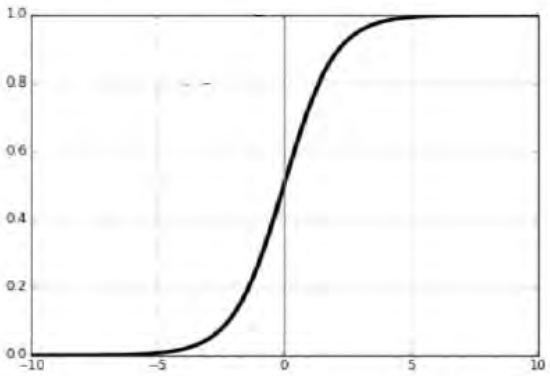
Функция активации играет ключевую роль в работе нейрона, преобразуя простую взвешенную сумму входных сигналов в осмысленный выходной сигнал. Без этого преобразования нейрон не смог бы выполнять сложные вычисления и формировать нелинейные зависимости. Как показано на рисунке 1.6, существует несколько распространенных типов активационных функций, каждая из которых по-своему обрабатывает входные данные и определяет характер выходного сигнала нейрона.



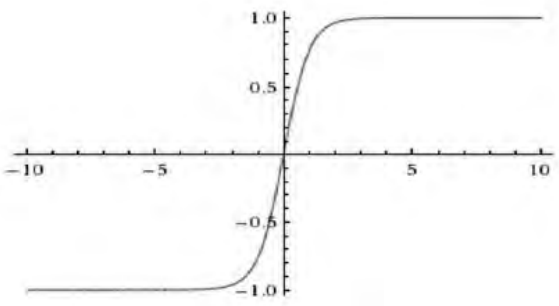
Ступенчатая функция $\varphi(x) = \begin{cases} 0, \text{если } x < 0 \\ 1, \text{если } x \geq 0 \end{cases}$



Линейная функция активации: $\varphi(x) = x$

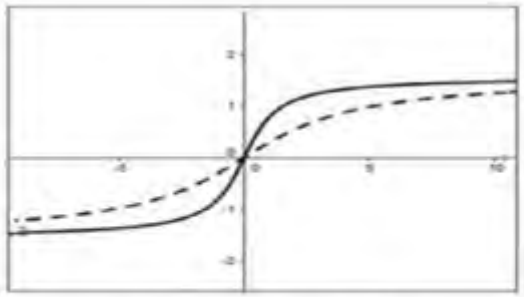


Сигмода: $\varphi(x) = \frac{1}{1+e^{-x}}$

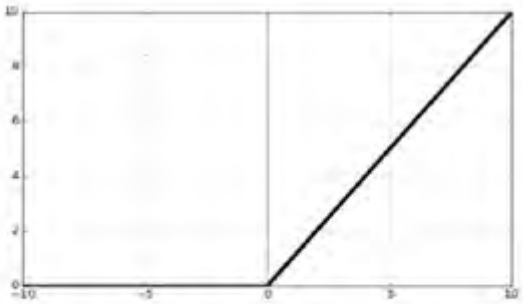


Гиперболический тангенс:

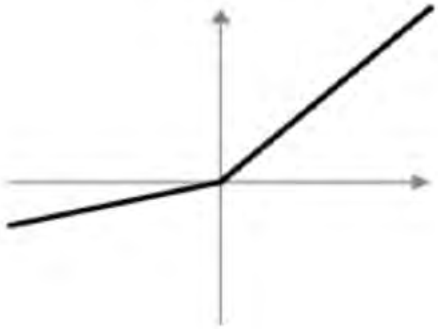
$$\varphi(x) = \tanh(x) = \frac{1}{1 + e^{-2x}} - 1$$



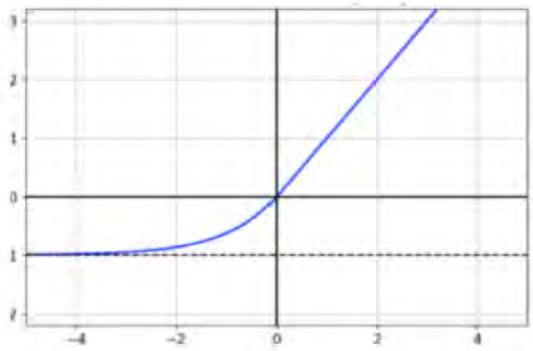
Arctan: $\varphi(x) = tan^{-1}(x)$



Функция ReLu: $\varphi(x) = \begin{cases} 0, \text{если } x < 0 \\ x, \text{если } x \geq 0 \end{cases}$



PReLu: $\varphi(x) = \begin{cases} \alpha x, \text{если } x < 0 \\ x, \text{если } x \geq 0 \end{cases}$



ELU: $\varphi(x) = \begin{cases} \alpha(e^x - 1), \text{если } x < 0 \\ x, \text{если } x \geq 0 \end{cases}$

Рисунок 1.6 – Основные активационные функции для нейронных сетей

Рост популярности глубоких нейронных сетей обусловлен несколькими ключевыми факторами. Во-первых, революция в вычислительных технологиях, особенно появление мощных графических процессоров (GPU), кардинально ускорила процесс обучения сложных нейросетевых архитектур и повысила их точность [66]. Эти достижения устранили ранее существовавшие ограничения, связанные с недостаточной вычислительной мощностью для тренировки эффективных моделей.

Во-вторых, накопление огромных массивов данных создало необходимую основу для успешного обучения глубоких сетей. В-третьих, разработка новых методов обучения позволила эффективно тренировать сети со сотнями слоев, решая такие проблемы как затухание градиентов и переобучение [67]. Синергия этих факторов обеспечила прорыв в практическом применении глубоких нейронных сетей, утвердив их лидирующие позиции среди методов машинного обучения.

Несмотря на многообразие современных нейросетевых архитектур, они

принципиально делятся на два типа: feedforward-сети (с прямым прохождением сигнала) и рекуррентные сети (с обратными связями), как показано на рисунке 1.7. Эта классификация отражает принципиальные различия в способах обработки информации и архитектурной организации нейросетевых моделей.

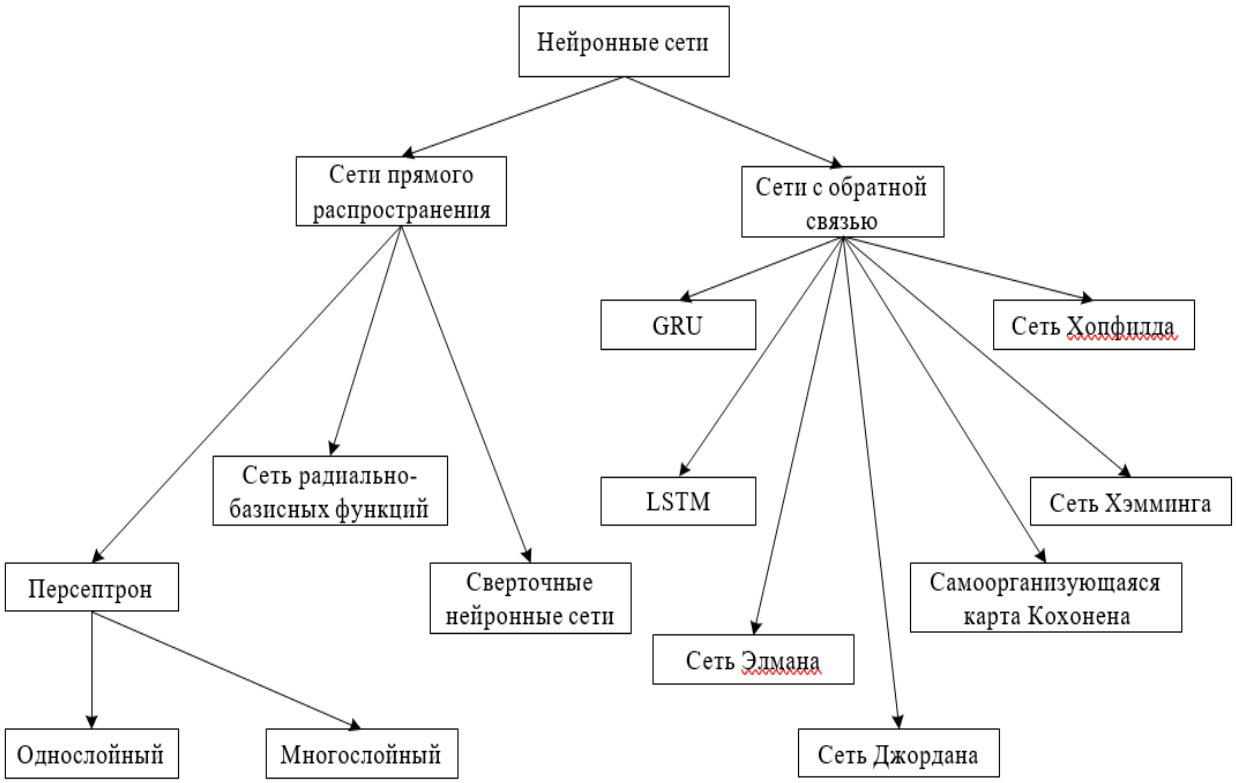


Рисунок 1.7 – Классификация нейронных сетей

Термин "глубокая нейронная сеть" в литературе интерпретируется по-разному. В данном исследовании, следуя подходу [68], мы относим к глубоким сетям архитектуры с числом скрытых слоев, превышающим один.

Среди наиболее известных и широко применяемых архитектур глубокого обучения следует отметить многослойные персептроны (MLP), сверточные нейронные сети (CNN), автоэнкодеры, рекуррентные нейронные сети (RNN) и их многочисленные модификации. Эти архитектуры демонстрируют высокую эффективность при решении различных задач машинного обучения благодаря своей способности выявлять сложные иерархические зависимости в данных. Особенностью глубоких нейронных сетей является их многоуровневая организация, позволяющая поэтапно извлекать и преобразовывать признаки на разных уровнях абстракции.

1.5 Особенности анализа данных о состоянии промышленного оборудования

1.5.1 Общие положения анализа временных рядов

Набор значений контролируемых параметров, получаемых посредством сенсоров, обычно формирует многомерный временной ряд. При адекватном анализе такого ряда возможно извлечение значимой информации о динамике состояния системы. Одной из характерных черт сложного поведения систем является наличие качественно различных закономерностей или режимов функционирования, которые могут последовательно изменяться во времени. Выявление этих изменений осуществляется с применением различных методов анализа временных рядов, что позволяет извлекать полезные знания из собранных данных [69].

Прогнозирование технического состояния промышленных систем и машин рассматривается как проблема в прогнозировании временных рядов, которые указывают на изменения технических параметров. Исследование [70] представляет методику прогнозирования неисправностей оборудования во время редких событий, используя анализ данных временных рядов, связанных с крутящим моментом приводных электродвигателей в рольганге лесозаготовительной группы, реализованный с помощью алгоритма случайного леса. Современные исследования демонстрируют значительный прогресс в области прогностической аналитики промышленного оборудования. В частности, в публикации [71] детально описана адаптивная модель прогнозирования временных рядов, обеспечивающая комплексную оценку технического состояния систем. Последующие научные работы [72-79] существенно обогатили методологическую базу, разработав различные подходы к анализу временных данных для диагностики и предупреждения отказов. Полученные результаты убедительно свидетельствуют о фундаментальной роли анализа временных рядов технологических параметров для точного мониторинга и прогнозирования работоспособности промышленных систем.

1.5.2 Методы и модели анализа временных рядов

В современном научном дискурсе представлено множество подходов к анализу и прогнозированию временных рядов. На основе общепризнанных классификаций эти методы можно в целом разделить на две основные группы: статистические методы и алгоритмы машинного обучения. Статистические методы обычно используют линейные модели для выражения взаимосвязи между прошлыми и будущими значениями с помощью определенных математических формул. Основными преимуществами этой категории методов являются их простота, адаптивность и простота интерпретации результатов. Некоторые примеры статистических методов:

– регрессионные модели;

– авторегрессионные модели (AR, ARMA, ARIMA, ARIMAX, GARCH, SARIMA);

– модели экспоненциального сглаживания;

– и другие.

Ко второй категории относятся нелинейные подходы, устанавливающие взаимосвязь между предыдущими и будущими значениями через специфические структурные преобразования и правила эволюции. В данной группе особое распространение получили архитектуры искусственных нейронных сетей и алгоритмы древовидной регрессии. Ключевыми достоинствами этих методов выступают их адаптивность, масштабируемость и принципиальная способность к выявлению сложных нелинейных зависимостей в динамических данных.

Современный этап развития характеризуется активным внедрением нейросетевых архитектур, среди которых рекуррентные нейронные сети (RNN) демонстрируют особую эффективность при решении разнообразных прикладных задач. Их успешное применение охватывает широкий спектр предметных областей, что обусловлено способностью обрабатывать последовательности данных с учетом временных зависимостей. Одним из значительных достижений является использование моделей с длинной

краткосрочной памятью (LSTM), которые значительно расширяют возможности базовой RNN.

С целью повышения точности прогнозирования в определенных задачах целесообразно использовать комбинированные подходы, что позволяет эффективно задействовать преимущества различных моделей. Данная стратегия способствует улучшению точности прогнозов по сравнению с применением отдельных методов в изоляции.

В этом контексте рассмотрим два подхода к прогнозированию временных рядов: одношаговое и многошаговое прогнозирование (рисунок 1.8):

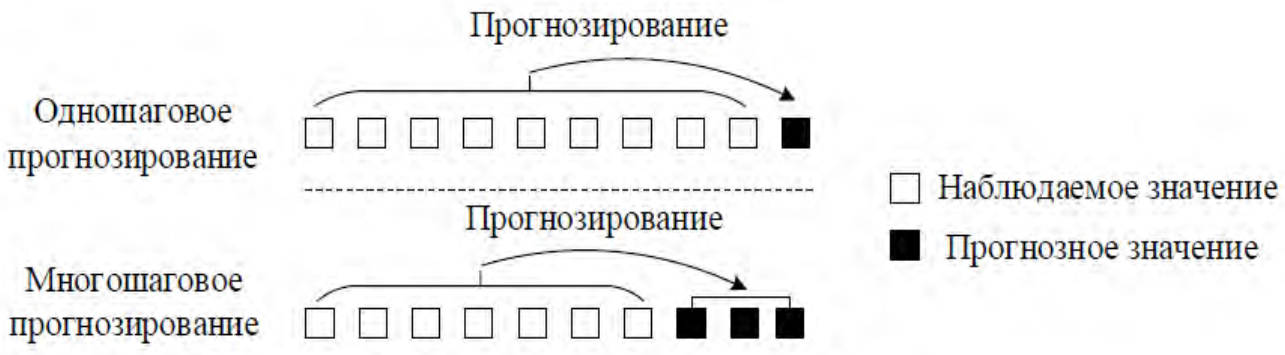


Рисунок 1.8 – Сравнение одношагового и многошагового прогнозирования

Одношаговое прогнозирование фокусируется на предсказании конкретного значения временного ряда для следующего временного интервала, обеспечивая точные краткосрочные оценки. В отличие от этого, многошаговое прогнозирование анализирует временные ряды на протяженном периоде, выявляя общие тенденции и критические точки изменения динамики процессов. Такой подход позволяет не только прогнозировать конкретные значения, но и определять характер развития процессов в перспективе, что особенно важно для стратегического планирования и принятия долгосрочных решений. Оба метода дополняют друг друга, обеспечивая комплексный анализ временных рядов как в краткосрочной, так и в долгосрочной перспективе. На рисунке 1.9 представлены основные методологические подходы, применяемые в многошаговом прогнозировании.

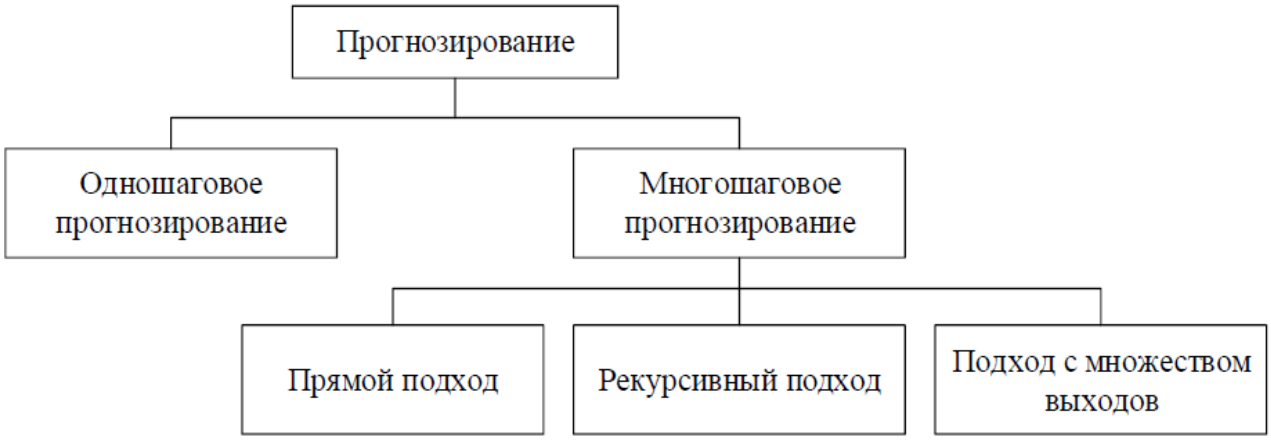


Рисунок 1.9 – Подходы к прогнозированию.

Прямой метод характеризуется необходимостью разработки отдельных прогностических моделей для каждого временного шага. Такая архитектура приводит к значительному росту вычислительной нагрузки, особенно при увеличении горизонта прогнозирования. Фундаментальным ограничением данного подхода является его неспособность учитывать временные взаимосвязи между последовательными прогнозами. Это существенно снижает точность прогнозирования, так как противоречит базовому свойству временных рядов, где текущие значения обычно коррелируют с предыдущими наблюдениями.

Рекурсивная методика [80] предлагает альтернативное решение, основанное на итеративном применении одношаговой модели, где каждый последующий прогноз строится с использованием предыдущего предсказанного значения. Однако принципиальным ограничением такого подхода становится кумулятивный эффект ошибок прогнозирования, который прогрессивно снижает точность модели по мере удаления от начальной точки.

Более совершенный метод с множественными выходами предполагает разработку единой модели, одновременно генерирующей всю последовательность прогнозов с учетом внутренних взаимосвязей в данных. Несмотря на преимущества в учете временных зависимостей, этот подход требует значительных вычислительных ресурсов для обучения, большого объема тренировочных данных и тщательного контроля за процессом обучения

для предотвращения переобучения. Сложность архитектуры таких моделей существенно возрастает по сравнению с традиционными подходами.

1.6 Анализ современных подходов к построению автоматизированных систем прогнозирования

На примере системы компании SINAMICS рассмотрим современный подход к построению автоматизированных систем прогнозирования.

Функционирование технических систем существенно зависит от изменчивых параметров окружающей среды, таких как температурные колебания, показатели влажности, вибрационные воздействия, качество энергоснабжения и химический состав окружающей среды. Эти факторы обладают выраженной стохастической природой, что значительно осложняет прогнозирование их влияния на технологическое оборудование [81]. Для обеспечения стабильной работы в таких условиях требуются многоуровневые системы мониторинга, адаптивные алгоритмы управления, предиктивные модели технического состояния и автоматизированные системы защиты. Подобные решения позволяют не только поддерживать работоспособность оборудования, но и предотвращать аварийные ситуации благодаря раннему обнаружению аномалий, прогнозированию остаточного ресурса и автоматической корректировке рабочих параметров [81].

Проектирование и внедрение современных систем управления осуществляется совместными усилиями специалистов различного профиля, включая технологов, инженеров-электриков, механиков, программистов и экономистов [82]. Такой междисциплинарный подход позволяет комплексно решать ключевые производственные задачи, такие как снижение эксплуатационных расходов на 15-30%, уменьшение энергопотребления, оптимизация затрат на обслуживание, повышение точности управления процессами, увеличение межремонтных интервалов, снижение количества бракованной продукции, улучшение условий труда персонала, повышение промышленной безопасности и упрощение документооборота [82].

В промышленности применяется широкий спектр систем автоматического управления, различающихся по степени автоматизации (полуавтоматические, автоматизированные и полностью автономные системы), архитектуре управления (централизованные, распределенные и гибридные конфигурации) и принципу работы (разомкнутые, замкнутые с обратной связью и комбинированные системы) [83]. Особый интерес представляют системы с обратной связью, работа которых основана на непрерывном мониторинге выходных параметров, их сравнении с эталонными значениями и автоматической корректировке режимов работы. Математически этот процесс описывается уравнением $y(t) = K \cdot [x(t) - y(t)]$, где $x(t)$ представляет задающее воздействие, $y(t)$ - выходной параметр, а K - коэффициент усиления [84].

Ярким примером современных систем с обратной связью являются сервоприводные контроллеры, использующие высокоточные энкодеры для измерения углового положения, определения скорости вращения и контроля ускорения/замедления.

Эти системы обеспечивают точность позиционирования до 0,01°, плавность хода при любых скоростях, быстрый отклик на управляющие сигналы, а также позволяют реализовать синхронное движение нескольких осей, компенсацию механических люфтов и адаптацию к изменяющейся нагрузке [84]. Современные тенденции развития систем управления включают внедрение методов искусственного интеллекта, использование промышленного интернета вещей (IIoT), развитие предиктивной аналитики и создание цифровых двойников оборудования, что позволяет существенно повысить точность управления, надежность систем и экономическую эффективность производства [81-84].

Блок управления SINAMICS от компании SIEMENS, способный управлять положением нескольких сервоприводов, является наглядным примером практического применения системы с обратной связью (рисунок 1.10).

Устройство управления приводами состоит из нескольких модулей. Блок

управления SINAMICS подключается к модулю LM (модуль питания), который преобразует переменное напряжение питания 380 В в постоянное напряжение 500 В. Преобразованное напряжение затем дополнительно регулируется в модулях двигателей и подается непосредственно на конкретные приводы.

Рисунок 1.10 – Программируемый логический контроллер Siemens ET 200SP.

Рассматриваемый приводной контроллер обеспечивает интеграцию с программируемым логическим контроллером ET200-SP посредством промышленного сетевого протокола PROFINET. В рамках данного взаимодействия реализован обмен данными, который может осуществляться как в виде структурированных программных блоков (Telegram), так и в форме специализированных технологических объектов (Technology Objects).

Это устройство превосходит блок управления SINAMICS, передавая необходимые целевые позиции, которые должны занимать все оси.

На представленной иллюстрации показан ПЛК, который отправляет требуемые позиции в блок управления SINAMICS и выполняет функции ведущего устройства.

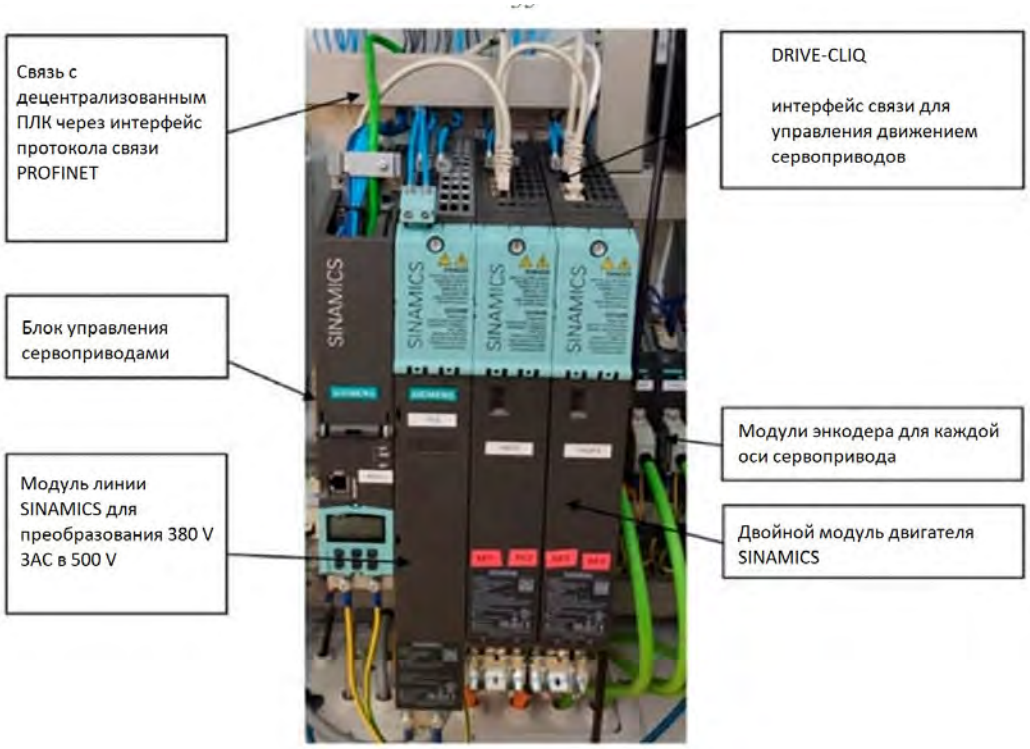


Рисунок 1.11 – Система управления с обратной связью.

Рисунок 1.11 иллюстрирует пример системы управления с обратной связью, в частности, блока управления SINAMICS, предназначенного для управления несколькими осями сервоприводов.

В отличие от систем управления с обратной связью, системы с разомкнутым контуром функционируют без механизма обратной связи. В таких системах элементы управления действуют без измерения выходной переменной, что исключает возможность сравнения фактического значения выходной переменной с заданным входным параметром. Это создает риск того, что исполнительный механизм может не достичь желаемого воздействия на процесс, что является значительным недостатком разомкнутых систем. Тем не менее, их преимущество заключается в том, что они, как правило, более просты и экономичны по сравнению с системами, использующими обратную связь.

В рамках данного исследования особое внимание уделяется системам управления с обратной связью и их ключевым компонентам.

На рисунке 1.12 представлен предложенный метод интеграции модуля прогнозной аналитики в существующую систему управления. «Данный модуль

был разработан в консольном приложении на платформе .NET и имеет доступ к локальной или удаленной базе данных, содержащей исторические данные, полученные от ПЛК. Кроме того, модуль обеспечивает доступ к текущим данным ПЛК, которые сканируются в режиме реального времени, с использованием функций из библиотеки S7.Net Plus» [85, с. 5]. Информационный обмен осуществляется через блоки данных, к которым имеет доступ модуль прогнозной аналитики.

В рамках данной архитектуры требуется модификация программного обеспечения программируемого логического контроллера путем интеграции специализированных структур данных. Эти структурированные объекты, объединяющие разнотипные переменные в единые логические блоки, обеспечивают взаимодействие между ПЛК и прогностическим модулем, реализованным на платформе .NET.

Модуль имеет возможность записи и чтения необходимых переменных из определенных блоков данных, а также доступ к локальной базе данных, которая регулярно обновляется текущими и прогнозируемыми данными.

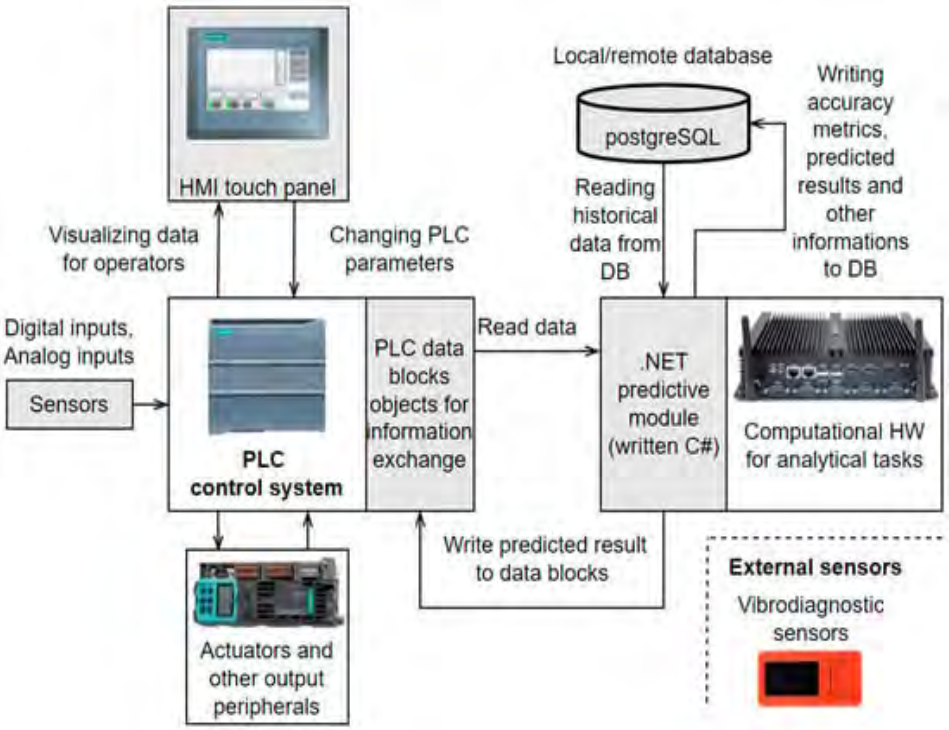


Рисунок 1.12 – Архитектура промышленной машины, оснащенной программируемым логическим контроллером (ПЛК) и реализованными функциями прогнозирования на основе искусственного интеллекта (ИИ).

В качестве базы данных выбрана объектно-реляционную систему PostgreSQL с открытым исходным кодом [86], которую можно установить локально на устройстве для аналитических вычислений или удаленно на сервере.

Предложенное решение позволяет реализовать мониторинг, обнаружение аномалий на отдельных устройствах [87], а также прогнозирование изменений определенных параметров, которые могут быть связаны с частыми ошибками или сбоями в работе машины или её отдельных компонентов. Прежде чем внедрять конкретные аналитические модели, необходимо провести анализ сбоев [88] и подготовить оборудование для сбора данных. Это может включать подключение устройства к базе данных для хранения информации или оснащение его различными датчиками для измерения других значимых величин. Предварительная обработка и начальный анализ данных могут быть выполнены с использованием доступного программного обеспечения, такого как MATLAB, или с помощью скриптов, написанных на языках программирования, таких как R, Python и других.

Если собранные данные содержат информацию о бесперебойной работе машины на протяжении длительного времени, это позволяет создать детектор аномалий [89]. В результате формируется модель, которая обучается на заданных данных, а её точность постоянно проверяется на основе предполагаемых данных, полученных от моделей ИИ, и фактических поступающих данных. При снижении значений статистических показателей, характеризующих точность модели, можно сделать вывод о том, что устройство находится в состоянии, не наблюдаемом во время сбора данных, что указывает на наличие аномального явления для данного устройства или машины. Эта информация может быть представлена оператору, что позволяет ему своевременно проверить проблемные компоненты устройства и предотвратить возможные отказы.

Если в процессе первоначального сбора данных также учитывается поведение устройства в неисправном или дефектном состоянии, обученная

модель может быть использована в качестве инструмента для моделирования процессов, происходящих на устройстве. С помощью дополнительных инструментов можно определить рекомендуемые настройки параметров, при которых предполагается отсутствие ошибок. Такие параметры должны быть утверждены специалистами с высоким уровнем компетенции и знаниями в данной области, такими как технологи, аналитики данных или другие эксперты [90].

1.7 Выводы по первой главе

1. В главе рассмотрены современные подходы к построению автоматизированных систем мониторинга и прогнозирования технического состояния промышленного оборудования в эпоху Индустрии 4.0. Основным фактором, определяющим конкурентоспособность машиностроительных предприятий, является качество продукции, которое зависит от множества факторов, включая техническое состояние оборудования. Непрерывный мониторинг технологических процессов на всех этапах производства позволяет организовать обратную связь и принимать обоснованные решения о необходимых корректировках.

2. Современные предприятия переходят к концепции «Индустрия 4.0», что подразумевает интеграцию различных операций в единую информационную систему. Однако на практике часто возникают проблемы с объединением разнородного оборудования и систем управления, что затрудняет контроль и агрегацию информации. Существующие системы мониторинга не всегда способны эффективно решать эти задачи, поскольку они часто являются закрытыми решениями, ориентированными на оборудование одного производителя. Это подчеркивает необходимость разработки новых автоматизированных систем контроля, которые обеспечат эффективный мониторинг технологических процессов и оборудования.

3. Существуют различные подходы к прогнозированию, включая физические модели, формализованные знания и методы анализа данных. Каждый из этих подходов имеет свои преимущества и недостатки. Например,

физические модели требуют глубокого понимания механизмов отказов, тогда как методы на основе данных обеспечивают гибкость и возможность учета нелинейных зависимостей.

4. Заблаговременное техническое обслуживание, осуществляющееся на основе моделей прогнозирования неисправностей и отказов, позволяет выполнять работы только по мере необходимости, что снижает затраты и увеличивает срок службы оборудования.

5. Прогнозирование технического состояния оборудования, включая оценку остаточного ресурса (RUL), становится необходимым шагом для предупреждения отказов и оптимизации затрат на техническое обслуживание. Однако современные системы прогнозирования все еще недостаточно распространены, и их реализация требует учета множества факторов, влияющих на состояние оборудования.

6. Методы машинного обучения, включая алгоритмы глубокого обучения, показывают высокую эффективность в задачах прогнозирования технического состояния. Они позволяют анализировать большие объемы данных и выявлять сложные зависимости, что делает их предпочтительными для применения в автоматизации мониторинга. Важно отметить, что успешное применение машинного обучения требует качественной подготовки данных, включая конструирование признаков и выбор подходящих алгоритмов.

7. Внедрение автоматизированной системы прогнозирования технического состояния промышленного оборудования, функционирование которой основано на применении методов искусственного интеллекта и машинного обучения, в рамках Индустрии 4.0 позволит предотвращать неисправности и отказы, прогнозировать остаточный ресурс, повысить качество продукции и оптимизировать затраты на техническое обслуживание.

ГЛАВА 2. РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СТРУКТУРЫ И МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ПРОМЫШЛЕННОГО ОБОРУДОВАНИЯ

2.1. Подходы к построению функциональной структуры автоматизированной системы прогнозирования

Как было описано ранее, задача разработки автоматизированной системы прогнозирования является очень актуальной и по данной теме выходит большое число научных публикаций. В них рассматриваются разные подходы к построению функциональной структуры автоматизированной системы прогнозирования, но основными являются два – модульный и алгоритмический [104-108].

Модульный подход к разработке функциональной структуры [109] автоматизированной системы прогнозирования представляет собой методологию, основанную на декомпозиции системы на отдельные, независимые, но взаимосвязанные модули. Каждый модуль выполняет специфические функции, связанные с анализом данных и прогнозированием будущих событий. Такой подход обеспечивает создание гибких, масштабируемых и легко поддерживаемых решений, что особенно важно в контексте прогнозирования, где требования могут быстро изменяться в зависимости от новых данных и бизнес-условий.

В рамках модульной архитектуры каждый элемент системы прогнозирования может отвечать за определенный этап аналитического процесса. Например, один модуль может быть предназначен для сбора и очистки данных, другой — для их анализа и обработки, третий — для построения моделей прогнозирования, а четвертый — для визуализации результатов и их представления конечным пользователям. Такое разделение задач позволяет командам разработки сосредоточиться на отдельных аспектах системы, что способствует повышению качества и эффективности работы.

Одним из основных преимуществ модульного подхода является

возможность параллельной разработки. Разные команды могут одновременно работать над различными модулями, что значительно ускоряет процесс создания системы. Например, одна команда может разрабатывать алгоритмы машинного обучения, в то время как другая занимается интеграцией данных из различных источников. Это позволяет быстрее реагировать на изменения требований и внедрять новые функции, что критически важно в условиях динамичного рынка.

Модульный подход также способствует более эффективному управлению сложностью системы. В прогнозировании часто применяются сложные алгоритмы и модели, и модульная архитектура позволяет разбивать их на более управляемые части. Каждый модуль может быть протестирован и оптимизирован отдельно, что упрощает процесс отладки и снижает вероятность возникновения ошибок в системе. Кроме того, это позволяет вносить изменения в отдельные модули без необходимости переработки всей системы, что особенно актуально в условиях постоянных изменений данных и бизнес-требований.

Не менее важным аспектом модульного подхода является возможность повторного использования модулей. Разработанные модули могут быть применены в других проектах или системах, что экономит время и ресурсы. Например, алгоритм предсказания, созданный для одной области, может быть адаптирован и использован в другой, что повышает общую эффективность разработки и способствует стандартизации процессов.

Кроме того, модульная структура позволяет легко интегрировать новые технологии и инструменты в систему. В условиях быстрого развития технологий в области прогнозирования возможность добавления новых модулей или замены устаревших компонентов без значительных затрат времени и ресурсов становится важным фактором для поддержания конкурентоспособности. Например, если появляется новый алгоритм машинного обучения, его можно легко интегрировать в существующую систему как новый модуль, не затрагивая остальные компоненты.

Однако модульный подход требует тщательного проектирования интерфейсов между модулями. Каждый модуль должен иметь четко определенные интерфейсы для взаимодействия с другими элементами системы, что позволяет избежать проблем совместимости и обеспечивает надежность всей системы. Это также требует от разработчиков ясного понимания процессов передачи данных между модулями и используемых форматов, что может добавить дополнительную сложность в процесс разработки.

Алгоритмический подход к созданию функциональной структуры [110-115] автоматизированной системы прогнозирования ориентирован на использование алгоритмов для решения задач, связанных с анализом данных и прогнозированием будущих событий. Этот подход основывается на формализации процессов, обеспечивающих обработку и анализ больших объемов данных, что имеет критическое значение для компаний, стремящихся извлечь ценную информацию из своих данных для обоснования бизнес-решений.

В рамках алгоритмического подхода основное внимание уделяется разработке четких и эффективных алгоритмов, способных обрабатывать, анализировать и интерпретировать данные. Эти алгоритмы могут включать как традиционные статистические методы, так и современные техники машинного обучения и искусственного интеллекта. Прогнозирование требует применения различных методов, таких как регрессионный анализ, деревья решений, нейронные сети и алгоритмы кластеризации, которые позволяют выявлять закономерности и строить модели, способные предсказывать будущие события на основе исторических данных.Ключевым элементом алгоритмического подхода является последовательность шагов, необходимых для достижения постав ленных целей. Эти шаги могут включать сбор и подготовку данных, выбор и обучение модели, оценку ее эффективности и внедрение в производственную среду. Каждый из этих этапов требует применения специфических алгоритмов, что позволяет четко структурировать процесс

разработки системы. На этапе сбора данных применяются алгоритмы, направленные на очистку и обработку информации, что обеспечивает её высокое качество и пригодность для последующего анализа. В процессе построения модели используются алгоритмы машинного обучения, которые обучаются на подготовленных данных для выявления закономерностей и зависимостей. Алгоритмический подход также включает методы анализа и оптимизации, способствующие повышению производительности алгоритмов и улучшению качества предсказаний. Это предполагает оценку временной и пространственной сложности алгоритмов, что позволяет разработчикам выбирать наиболее эффективные решения для конкретных задач. В контексте прогнозирования может возникнуть необходимость оптимизации алгоритмов для работы с большими объемами данных, что требует внедрения таких методов, как отбор признаков, регуляризация и кросс-валидация. Кроме того, алгоритмический подход способствует модульности и повторному использованию кода. Каждый алгоритм или модель может быть реализован как отдельный компонент, который можно интегрировать в более крупную систему [116]. Это позволяет разработчикам сосредоточиться на специфических аспектах системы, что упрощает процессы тестирования и отладки. Например, новый алгоритм предсказания может быть легко протестирован и интегрирован в существующую систему без необходимости полной переработки архитектуры. Ключевым аспектом алгоритмического подхода является способность адаптироваться к изменяющимся условиям. В области прогнозирования данные и бизнес-требования могут быстро меняться, и алгоритмический подход обеспечивает гибкость в реагировании на эти изменения. Например, при появлении новых источников данных или изменении бизнес-правил алгоритмы могут быть адаптированы или заменены без значительных затрат времени и ресурсов. Это особенно важно для организаций, стремящихся сохранить конкурентоспособность на быстро меняющемся рынке. Алгоритмический подход также охватывает аспекты мониторинга и управления производительностью системы. После внедрения

алгоритмов в производственную среду необходимо отслеживать их эффективность и точность предсказаний. Это может включать использование метрик, таких как точность, полнота и F-мера, для оценки качества работы алгоритмов. Если наблюдается снижение производительности алгоритмов, они могут быть оптимизированы или заменены новыми моделями, что позволяет поддерживать высокий уровень качества предсказаний.

Успешно комбинируя данные подходы, была разработана новая функциональная структура автоматизированной системы прогнозирования технического состояния промышленного оборудования, представленная на рисунке 2.1.

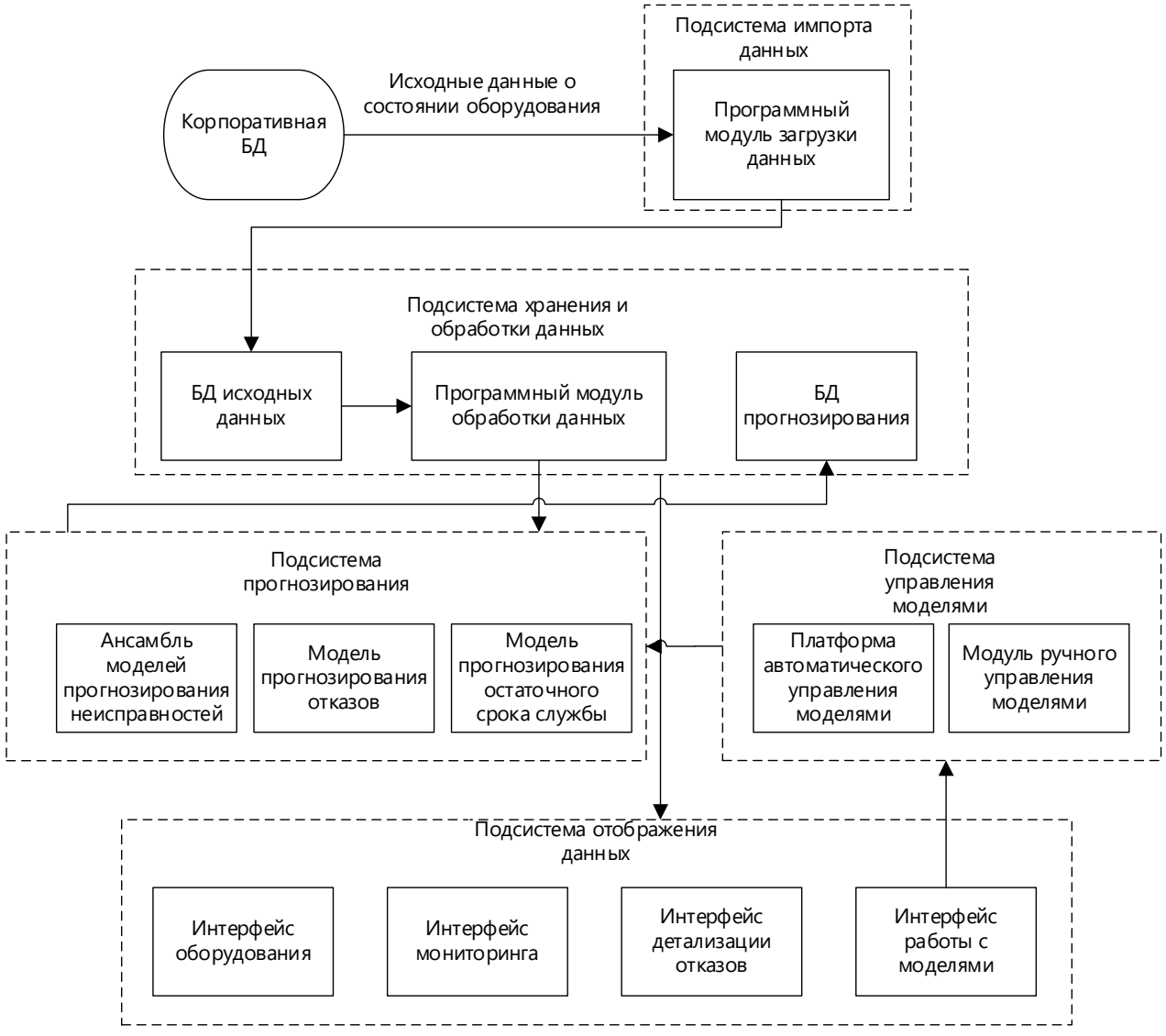


Рисунок 2.1 – Функциональная структура автоматизированной системы прогнозирования технического состояния промышленного оборудования

Подробно каждая из подсистем будет рассмотрена в третьей главе, а далее описаны принципы построения одной из ключевых подсистем – «Подсистемы прогнозирования».

2.2. Постановка задачи прогнозирования технического состояния промышленного оборудования

В условиях современного промышленного производства, характеризующегося высокой степенью автоматизации и конкуренции, обеспечение надежности и эффективности работы оборудования становится критически важным. Традиционные методы технического обслуживания, основанные на периодическом осмотре и ремонте, не всегда способны предотвратить внезапные отказы, что приводит к значительным экономическим потерям. В этой связи прогнозирование состояния оборудования в реальном времени представляет собой актуальное направление исследований и практического применения. Проблема, рассматриваемая в данной научной работе, заключается в необходимости разработки эффективной автоматизированной системы прогнозирования, способной на основе анализа больших объемов данных, получаемых от датчиков и систем мониторинга, предсказывать техническое состояние промышленного оборудования. Основные вызовы в рамках данной работы:

- Сложность и многомерность данных. Данные, получаемые с различных датчиков, могут быть зашумленными, неполными и иметь различные временные интервалы, что затрудняет их обработку и анализ.
- Неопределенность и изменчивость. Оборудование может функционировать в различных режимах, и его состояние может изменяться в зависимости от внешних факторов, таких как температура, влажность и нагрузка.
- Выбор методов анализа: существуют различные методы и алгоритмы машинного обучения, и необходимо определить наиболее подходящие для конкретной задачи предсказания состояния оборудования.

2.3 Метод прогнозирования неисправностей

2.3.1 Цель метода и постановка задачи

В период эксплуатации оборудование может находиться в различных состояниях:

1) Исправное (“Normal”) – оборудование соответствуем всем требованиям нормативно-технической документации;

2) Работоспособное (“Operational”) – оборудование соответствует требованиям нормативно-технической документации по параметрам, напрямую связанным с выполнением заданных задач;

3) Неисправное (“Faulty”) – состояние, при котором заданные задачи выполняются в частичном или полном объеме, но оборудование не соответствует хотя бы одному требованию нормативно-технической документации. Необходимо планирование ремонтных работ;

4) Критическое (“Critical”) – состояние, при котором дальнейшая эксплуатация неизбежно ведет к отказу, требуется срочное воздействие на оборудование;

5) Состояние отказа (“Breakdown”) – состояние, при котором оборудование не выполняет заданные задачи.

Целью метода является совместное распознавание неисправного и критического состояния для технических мест промышленного оборудования по показаниям датчиков. Формализуем задачу: В наборе данных D имеется n единиц однотипного оборудования, каждое из которых анализируется одинаковым количеством датчиков ($s(D_1) = s(D_n)$) и содержит фиксированный набор совпадающих технических мест ($p(D_1) = p(D_n)$). При этом продолжительность эксплуатации, причины и динамика развития деградаций для каждой единицы оборудования и для каждого технического места могут различаться. Требуется по информации, заданной множеством D построить алгоритм F:

$$\forall\, p \in Di, S_t(D_i) \xrightarrow{F} C_p = \{C_{Norm}, C_{Faul}\} \tag{2.1}$$

Который позволяет по входящим значениям датчиков за определенный

временной интервал (S_t) единицы оборудования (D_i) отнести состояние каждого технического места (p) к одному из возможных классов работоспособности (C_p): нормальное состояние (объединяет исправное и работоспособное состояния) (C_{Norm}) и неисправное состояние (объединяет неисправное и критическое) (C_{Faul}) (рисунок 2.2).

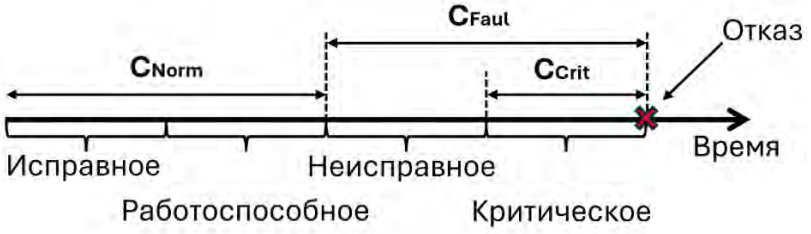


Рисунок 2.2 – Иллюстрация процесса формирования меток в зависимости от состояния оборудования

В решении рассматриваемой задачи применяется упрощенный подход бинарной классификации, т.е. по набору измеряемых физических параметров происходит установление одного из состояний оборудования: нормального или неисправного. Данный метод позволяет определять необходимость ремонта и профилактических работ как для единиц оборудования, задействованных в обучении (на последующих временных интервалах, не рассматриваемых ранее) так и для новых единиц оборудования того же типа, анализируемого тем же количеством и типом датчиков.

2.3.2 Описание предлагаемого метода идентификации потенциальных неисправностей оборудования[1]

В связи со сложной взаимосвязью между показаниями датчиков и состоянием промышленного оборудования было принято решение о создании модели прогнозирования, основанной на «мягком голосовании» между тремя алгоритмами, имеющими различные подходы к классификации (рисунок 2.3).

Суть данного ансамблевого метода заключается в подборе к каждому из предсказательных алгоритмов весов, отражающих значимость прогноза.

[1] В данном разделе использованы материалы, ранее опубликованные в работе:
Программные продукты Чернухин А. В. Построение модели предиктивной аналитики неисправностей промышленного оборудования / А. В. Чернухин, Е.А. Богданова, Т. В. Савицкая // Программные продукты и системы. – 2024. – № 2. – С. 254-261.

Подбор весов осуществлялся с использованием метода градиентного бустинга. Так, для каждого объекта рассчитывается вероятность принадлежности к классу по следующей формуле:

$$P_1 = w_a * p_{a1} + w_b * p_{b1} + w_c * p_{c1}, \quad (2.2)$$

где P_1 – итоговая вероятность принадлежности объекта к классу 1 (неисправность), p_{a1}, p_{b1}, p_{c1} – вероятности принадлежности объекта к классу 1, предсказанные тремя отдельными алгоритмами, w_a, w_b, w_c – веса, присваиваемые прогнозам отдельных алгоритмов.

Голосование осуществляется среди следующих подходов: сверточная нейронная сеть, логистическая регрессия и метод опорных векторов. Выбор этих методов обусловлен низкой корреляцией их предсказаний друг с другом (менее 0.6) с относительно высоким качеством предсказания для большей части анализируемых технических мест (значение индекса Жаккара выше 0.5).

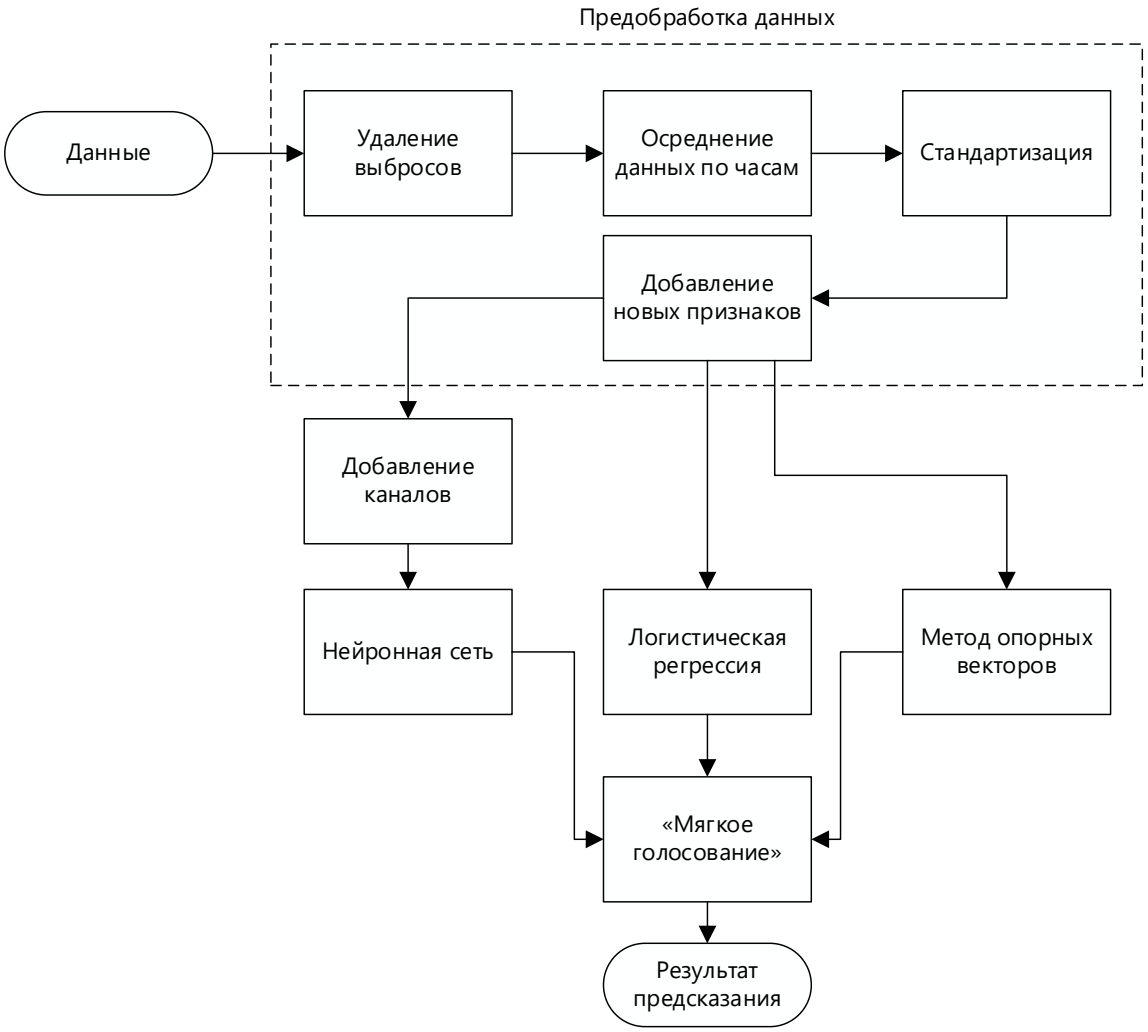


Рисунок 2.3 – Блок-схема работы модели прогнозирования.

Для логистической регрессии (Алгоритм 1) выбран базовый алгоритм оптимизации (LBFGS), а максимальное количество итераций равно 2000 (что обусловлено большим количеством признаков).

Алгоритм 1: Логистическая регрессия

1: **Вход:** Обучающие данные
2: **Начало**
3: От i = 1 до k
4: Для каждого экземпляра обучающих данных d_i
5: Установить целевое значение для регрессии $z_i = \frac{y_i - P(1|d_j)}{[P(1|d_j)(1-P(1|d_j))]}$
6: Инициализировать вес экземпляра d_j значением $[P(1|d_j)(1 - P(1|d_j))]$
7: Определить функцию f(j) для данных со значением класса Z_j и весом w_j
8: **Решение для классификации**
9: Присвоить метку класса 1 если $P_{id} > 0{,}5$, в противном случае присвоить метку класса 2
10: **Конец**

Метод опорных векторов (Алгоритм 2) обучался с использованием ядра RBF (Radial Basis Function) в связи с нелинейной разделимостью объектов в анализируемом наборе данных.

Алгоритм 2: Метод опорных векторов с RBF ядром

1: **Вход:** Обучающие данные
2: **Начало**
3: Определить параметры: C – значение регуляризации, γ – параметр ядра
4: Определить RBF ядро: $K(x_i, x_j) = \exp(-\gamma * \|x_i - x_j\|^2)$
5: Обучить модель: $\min(\frac{1}{2}\|w\|^2 + C * \sum(\max(0, 1 - y_i * (w \cdot \varphi(x_i) + b))))$, где $\varphi(x)$ – функция преобразования
6: **Решение для классификации**
7: Присвоить метку класса 1 если $P_i > 0{,}5$, в противном случае присвоить метку класса 2
8: **Конец**

Данные на вход моделям подавались в виде векторов размера 96 (по 6 признаков для каждого из 16 датчиков), ответом являлась метка 0 или 1 в случае неисправности в течение часа, описываемого вектором признаков.

Во время проделанной работы также было опробовано несколько архитектур нейронных сетей с разным количеством сверточных и полносвязных слоев. По итогу подбора была сконструирована сверточная нейронная сеть.

Свёрточная нейронная сеть (англ. Convolutional Neural Network, CNN) [123-128] – это тип многослойного перцептрона, который отличается наличием операции, называемой «свёртка». Для понимания этого процесса необходимо ввести некоторые термины, используемые в рамках CNN. Ядро свёрточного слоя представляет собой фильтр, содержащий систему разделяемых весов нейронов. Размер ядра обычно небольшой, в нашем случае это вектор размером 3. Фильтр перемещается по так называемым картам, которые представляют собой области массива, принимаемые на вход ядром. Затем все значения карты поэлементно умножаются на ядро и суммируются, а полученное значение записывается в матрицу признаков следующего слоя. Процесс свёртки показан на рисунок 2.4.

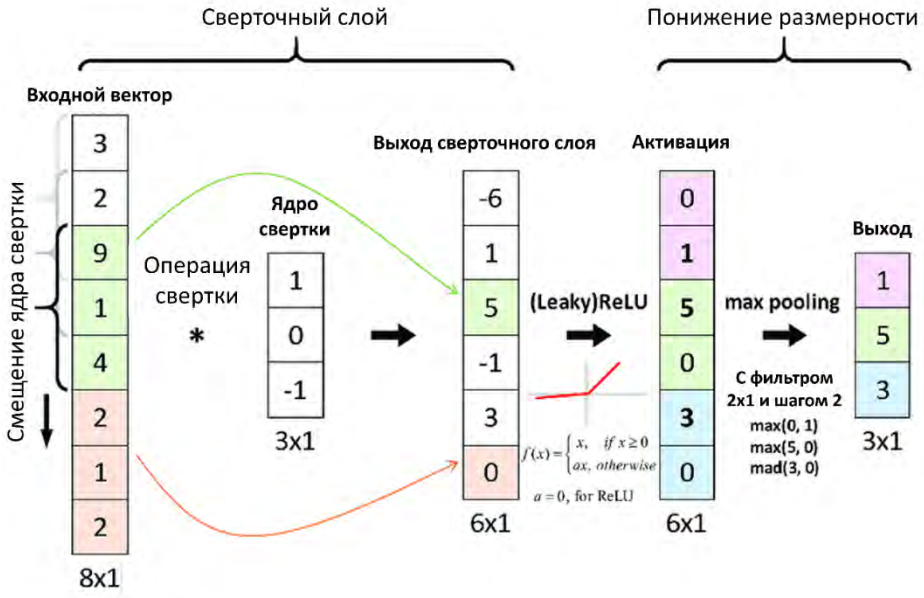


Рисунок 2.4 – Алгоритм работы операции одномерной свертки (Conv1D)

Помимо свёрточных слоев в CNN также присутствует подвыборочный слой, задача которого заключается в уменьшении размерности карты признаков путем выбора максимальных значений в каждой ячейке, соответствующей размерности ядра. Эта операция называется MaxPooling.

Одним из самых важных этапов в разработке алгоритма на основе нейронных сетей является процесс обучения, который включает в себя переопределение весов нейронов с целью минимизации функции потерь и максимизации точности. За это отвечает большое число гиперпараметров:

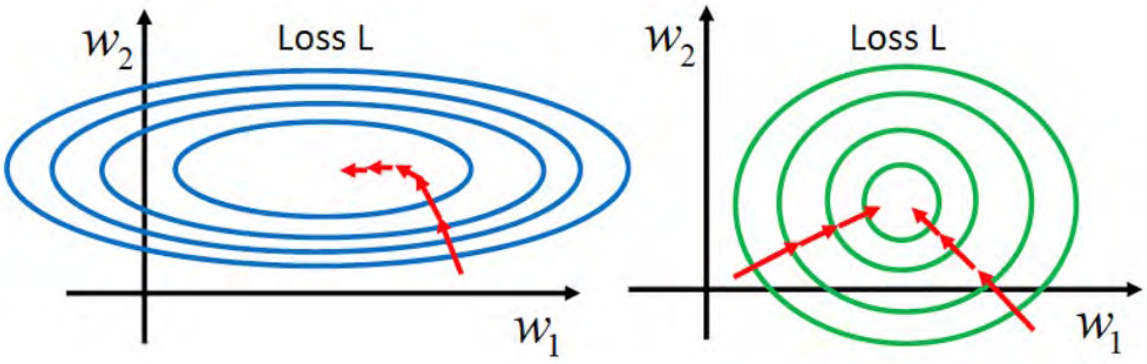
- ❖ Функция активации.

Функция активации – это функция, применяемая в нейронных сетях для вычисления взвешенной суммы входных данных и смещений, позволяющая определить, будет ли активирован нейрон. Она обрабатывает представленные данные с помощью определенной техники градиентного спуска и генерирует выходные данные для нейронной сети, содержащие параметры в этих данных. Функция активации может быть как линейной, так и нелинейной, в зависимости от представляемой ею функции, и используется для управления выходами нейронных сетей в различных областях. К одним из наиболее используемых функций активации для решения задачи классификации относятся Sigmoid, Tanh и ReLU.

- ❖ Нормализация

Нормализация входных данных нейронных сетей к нулевому среднему и фиксированному стандартному отклонению уже несколько десятилетий (LeCun, 2018) считается эффективным методом для обучения нейронных сетей. В глубоких сетях аналогичный процесс применяется и к промежуточным слоям, но с акцентом на повышение производительности: на каждой эпохе обучения нормализация осуществляется не для всех нейронов слоя, а лишь для небольшой их части, известной как мини-пакет (англ. mini-batch). Этот подход получил название пакетной нормализации (Batch Normalization, BN) [129]. В своей первоначальной работе [130] Иоффе и Сегеди предположили, что BN может уменьшать «внутренний ковариантный сдвиг»

— явление, при котором распределения значений признаков в скрытых узлах слоёв имеют различные параметры (математическое ожидание, дисперсия и т.д.) в процессе обучения (рисунок 2.5).



$$x \rightarrow \hat{x} = \frac{x - \mu}{\sigma} \rightarrow y = \gamma\hat{x} + \beta \tag{2.3}$$

Рисунок 2.5 – Минимизация функции потерь: слева представлена ситуация без пакетной нормализации, а справа — с её применением. На осях отображены веса нейронов. Внизу показан алгоритм пакетной нормализации, где x обозначает входные данные, y — выходные значения слоя; μ — среднее значение x в мини-пакете; σ — стандартное отклонение x в мини-пакете; γ и β — параметры, используемые для масштабирования и смещения

❖ Функция потерь.

Для оптимизации значений весов связей необходимо видеть их влияние на точность предсказаний. Для этого вводится функция потерь, которая принимает большие значения при плохой классификации, а при улучшении качества классификации, её значение падает. Таким образом, при оптимизации параметров стоит задача минимизации функции потерь.

❖ Алгоритм оптимизации

Таким образом, процесс обучения нейронной сети заключается в настройке весов нейронов и смещений так, чтобы предсказанные метки максимально соответствовали истинным значениям. Для нахождения оптимальных параметров необходимо минимизировать функцию потерь, для чего разработано множество алгоритмов.

Одним из наиболее популярных методов оптимизации в области глубокого обучения является стохастический алгоритм Adam [131], который

был предложен как адаптивный подход к регулированию скорости обучения в глубоких нейронных сетях. Основная идея метода заключается в индивидуальной настройке скоростей обучения для каждого параметра, при этом обновления параметров не зависят от масштабирования градиента.

❖ Регуляризация

Одним из основных недостатков глубоких нейронных сетей является переобучение, которое возникает из-за большого количества параметров. В результате алгоритм не находит закономерности, а просто запоминает классы для конкретных комбинаций признаков. Это приводит к тому, что точность предсказаний на обучающей выборке возрастает, в то время как на тестовой выборке наблюдается плато или даже снижение точности с увеличением числа эпох.

Dropout [132] — это метод регуляризации, предназначенный для борьбы с переобучением. Его суть заключается в случайном обнулении некоторых нейронов и их связей (рисунок 2.6), что помогает предотвратить избыточное запоминание признаков.

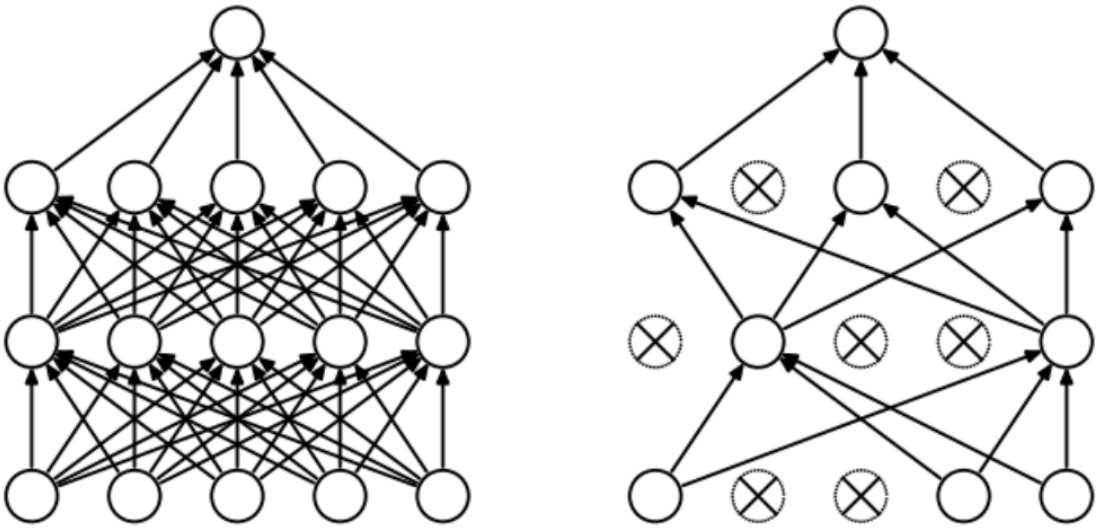


Рисунок 2.6 – Схема работы Dropout. Слева – архитектура нейронной сети без прореживания, справа – та же нейронная сеть с добавлением Dropout [133]

Еще одним популярным способом борьбы с переобучением является регуляризация, которая направлена на снижение значений весов связей, что уменьшает риск запоминания классификаций для определенных наборов признаков. Регуляризация добавляется в качестве дополнительного

компонента в уравнение функции оптимизации:

$$J\left(w^{[1]}, b^{[1]}, \dots, w^{[L]}, b^{[L]}\right) = \frac{1}{m}\sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m}\sum_{l=1}^{L}\left\|w^{[L]}\right\|_F^2$$

(2.4)

где L – функция потерь, w – вес нейрона, λ – параметр регуляризации.

С учетом подбора и оптимизации всех вышеописанных гиперпараметров итоговая нейронная сеть (рисунок 2.7) содержит базовые слои, в которых происходит подбор весов:

сверточный слой (Conv1D) - выполняет функцию обобщения признаков в каналах, а также выявления значимых паттернов с увеличением числа каналов с 6 до 32. В качестве каналов используются статистические характеристики, рассчитанные для показателей каждого из 16 датчиков, в частности: среднее, медиана, максимальное и минимальное значения за час, а также разность между средним и медианой за час и за месяц. В связи с размером входного тензора, выбран размер ядра свертки (kernel size), равный 3. После сверточного слоя располагается слой подвыборки по максимальному значению (MaxPooling), который уменьшает в два раза размер тензора. В качестве функции активации использовалась выпрямленная линейная функция активации (ReLu);

Два полносвязных слоя (Linear), с уменьшением количества каналов (100, 1). Для первого слоя использовалась функция активации ReLu. В последнем слое непосредственно и происходит предсказание наличия неисправности, и используется сигмоидная функция активации (Sigmoid), которая выводит вероятность принадлежности объекта к классу 1 (наличие неисправности).

Таким образом на вход алгоритм принимает 4D тензор размера 6x16, на выходе получаем одно число, отражающее вероятность наличия неисправности за час, который описывается входным тензором.

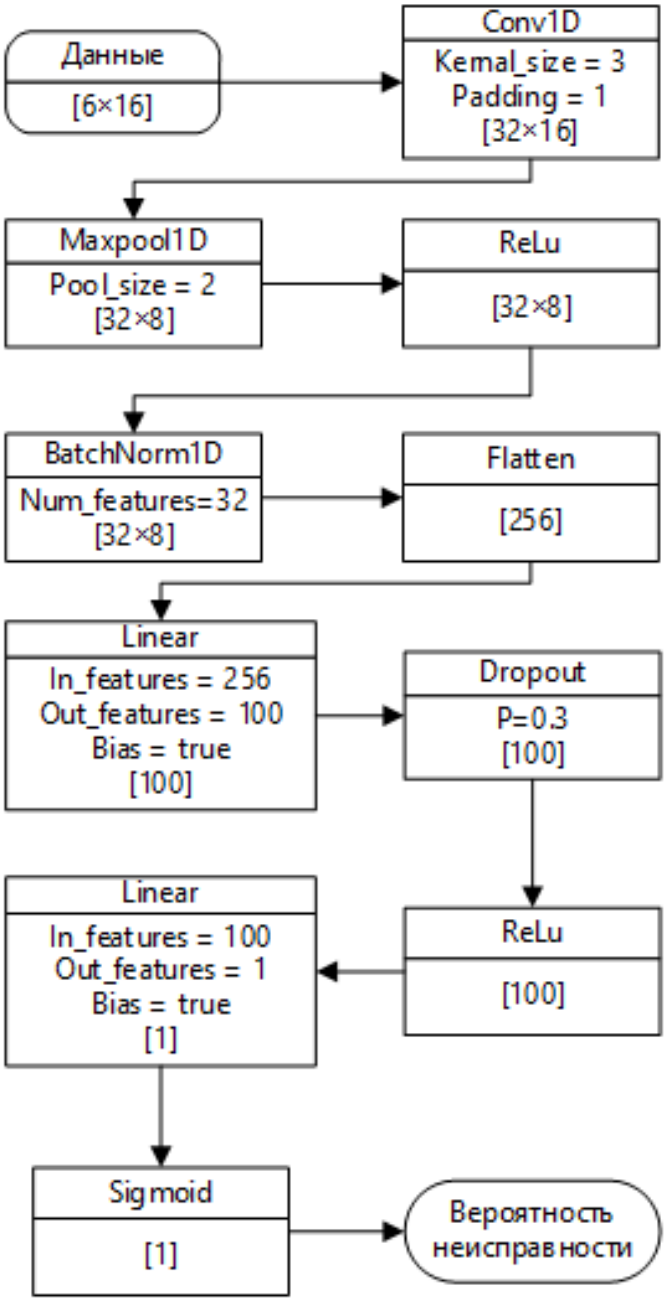


Рисунок 2.7 – Архитектура используемой сверточной нейронной сети

В целях нормализации и регуляризации после сверточных слоев использовались пакетная нормализация (Batch Normalization), а после первых двух полносвязных слоев - метод прореживания (Dropout) со значением отсева 0,3 (из-за наличия пакетной нормализации значение выбрано меньше стандартного 0,5).

В качестве оптимизатора был выбран оптимизатор Адама (AdamW). Выбор оптимизатора обусловлен его эффективностью и возможностью добавления L2 регуляризации (параметр weight_decay), которая необходима

для ограничения значения весов для признаков, что препятствует переобучению, и способствует поиску общих закономерностей (Алгоритм 3). В результате подбора гиперпараметров (настраиваемых вручную параметров обучения) была использована скорость обучения, равная 0,0001 и weight_decay=0,001.

Алгоритм 3: Оптимизатор Адам с ограничением весов (AdamW)

1: **Учитывая:** $\alpha = 0{,}001$, $\beta_1 = 0{,}9$, $\beta_2 = 0{,}999$, $\varepsilon = 10^{-8,}$ $\lambda \in R$
2: **Инициализировать** временной шаг $t \leftarrow 0$, вектор параметров $\Theta_{t=0} \in R^n$, вектор первого момента градиентов $m_{t=0} \leftarrow 0$, вектор второго момента градиентов $v_{t=0} \leftarrow 0$, график оптимизации
3: **Повторить**
4: $t \leftarrow t + 1$
5: $\nabla f_t(\Theta_{t-1}) \leftarrow$ SelectBatch(Θ_{t-1}) Выбрать пакет и вернуть соответствующий градиент
6: $g_t \leftarrow \nabla f_t(\Theta_{t-1})$
7: $m_t \leftarrow \beta_1 m_{t-1} + (1- \beta_1) g_t$ Здесь и далее операции выполняются поэлементно
8: $v_t \leftarrow \beta_2 v_{t-1} + (1- \beta_2) g_t$
9: $\widehat{m} \leftarrow m_t / (1- \beta_1^t)$ β_1 возводится в степень t
10: $\hat{v} \leftarrow v_t / (1- \beta_2^t)$ β_2 возводится в степень t
11: $\eta_t \leftarrow$ SetScheduleMultiplier(t) Можно исправить затухание или сделать перезапуск
12: $\Theta_t \leftarrow \Theta_{t-1} - \eta_t(\alpha \widehat{m}_t/(\sqrt{\hat{v}_t} + \varepsilon) + \lambda\theta_{t-1})$
13: **пока** не будет выполнен критерий остановки
14: **вернуть** оптимизированные параметры Θ_t

Для расчета ошибки использовалась функция потерь BCE, которая оптимальна для решения задачи бинарной классификации.

$$BCE = -\frac{1}{N}\sum_{i=0}^{N} y_i \cdot log(\hat{y}_i) + (1 - y_i) \cdot log(1 - \hat{y}_i)$$

(2.5)

При оптимизации так же подбирался batch size, оказывающий влияние на эффективность Batch Normalization (Алгоритм 4) и на продуктивность обучения, по итогу, исходя из размера обучающей выборки выбран batch size=16.

Алгоритм 4: Пакетная нормализация (BN)

1: **Вход:** Значения x в мини-пакетах: $B = \{x_{1\ldots m}\}$; обучаемые параметры: γ, β
2: $\mu_B \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i$ Поиск среднего мини-пакета

3: $\sigma_B^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2$	Поиск отклонения мини-пакета
4: $\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}}$	Нормализация
5: $y_i \leftarrow \gamma\hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i)$	Масштабировать и сдвинуть
6: **Выход:** $\{y_i = BN_{\gamma,\beta}(x_i)\}$	

Обучение проводилось в течение 30 эпох, сохранялась модель с наименьшим значением функции потерь на валидационном наборе данных.

В задаче прогнозирования неисправностей решается задача классификации. Так, для каждого из технических мест промышленного оборудования осуществляется бинарная классификация на два варианта: класс 1 – работоспособное состояние оборудования, класс 2 – наличие какой-либо неисправности.

Для формализованного описания метрик качества, применяемых для оценки решения задачи классификации, необходимо ввести важную концепцию для описания в терминах ошибок классификации — confusion matrix (матрица ошибок). Допустим, что у нас есть два класса и алгоритм, предсказывающий принадлежность каждого объекта одному из классов, тогда матрица ошибок классификации будет выглядеть следующим образом:

Таблица 2.1. Матрица ошибок.

	Фактическое положительное (1)	Фактическое отрицательное (0)
Предсказано положительное (1)	TruePositive (TP)	FalsePositive (FP)
Предсказано отрицательное (0)	FalseNegative (FN)	TrueNegative (TN)

На пересечении столбцов и строк записаны основные параметры, характеризующие истинность или ложность предсказания:

- **True Positive (TP)**: количество правильно классифицированных положительных примеров (модель предсказала 1, и на самом деле это 1).

- **True Negative (TN)**: количество правильно классифицированных отрицательных примеров (модель предсказала 0, и на самом деле это 0).
- **False Positive (FP)**: количество неправильно классифицированных отрицательных примеров (модель предсказала 1, а на самом деле это 0). Также называется ошибкой первого рода.
- **False Negative (FN)**: количество неправильно классифицированных положительных примеров (модель предсказала 0, а на самом деле это 1). Также называется ошибкой второго рода.

Таким образом, любую метрику качества, используемую в задачах классификации можно выразить через параметры матрицы ошибок. Так, одна из самых распространенных метрик качества – точность предсказания (англ. Accuracy) будет определяться следующим образом:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.6)$$

Выбор метрики зависит от сбалансированности набора данных, а также от требуемой чувствительности модели к ошибкам первого и второго рода.

В решаемой задаче набор данных как в обучающей, так и в тестовой выборке является несбалансированным, так как неисправность для многих технических мест является редким событием. Данный факт осложняет оценку точности модели и препятствует использованию ряда метрик качества, таких как Accuracy. Также важным требованием к модели является минимизация ложноположительных предсказаний, так как в реальных условиях ошибки подобного рода будут тратить временные ресурсы рабочей бригады, отвечающей за ремонт и обслуживание оборудования. В связи с этими особенностями решаемой задачи метрикой качества выступает индекс Жаккара, рассчитанный по формуле:

$$J = \frac{TP}{TP+FP+FN} \quad (2.7)$$

где J - индекс Жаккара, TP - истинно положительные значения из матрицы путаницы, FP - ложно положительные, FN - ложно отрицательные.

2.4 Метод прогнозирования отказов

2.4.1 Цель метода и постановка задачи

Целью метода является прогнозирование отказа на основе распознавания критического состояния для технических мест промышленного оборудования по показаниям датчиков. Множество единиц оборудования и пространство признаков совпадают с данными, используемыми в предыдущем методе. Однако изменения претерпели целевые метки состояния оборудования. Так, стоит задача: по информации, заданной множеством D построить алгоритм F:

$$\forall\, p \in Di, S_t(D_i) \xrightarrow{F} C_p = \{C_{NotCr}, C_{Cr}\} \quad (2.8)$$

Который позволяет по входящим значениям датчиков за определенный временной интервал (S_t) единицы оборудования (D_i) отнести состояние каждого технического места (p) к одному из возможных классов работоспособности (C_p): некритическое (объединяет исправное, работоспособное и неисправное состояния) ($C_{NotCrit}$) и критическое состояние (C_{Crit}) (рисунок

2.8).

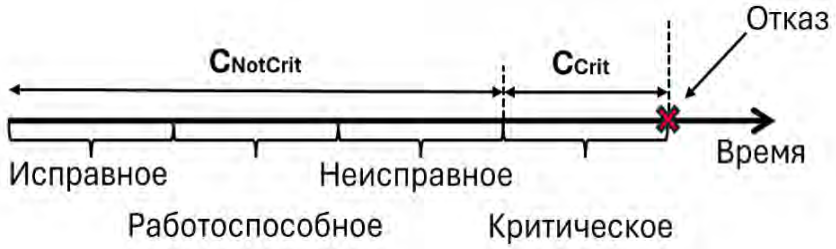


Рисунок 2.8 – Иллюстрация процесса формирования меток в зависимости от возможности возникновения отказа оборудования

При таком разделении состояний на классы происходит распознавание только тех, которые неизбежно приведут к отказу оборудования при дальнейшей эксплуатации.

Значительную сложность в решении этой задачи представляет наличие малого количества данных об отказах оборудования. В связи с этим, а также с

тем, что нет четких критериев отделения критичного состояния от неисправного, необходимо для каждого технического места, в соответствии с природой аномалий, определить горизонт прогнозирования отказов, для которого соблюдается и высокая точность предсказания, и наличие достаточного временного интервала для проведения ремонта.

Оптимальным подходом в данном случае будет являться предварительное определение аномальных режимов работы оборудования с их последующей классификацией по состоянию и техническим местам.

При успешном решении данной задачи возможно заблаговременно получить информацию о развитии критичной аномалии и провести срочные ремонтные работы.

2.4.2 Описание предлагаемого метода прогнозирования отказов оборудования

Перед подачей данных на вход модели необходимо провести предобработку показаний датчиков.

Отказ важного промышленного оборудования является редким и краткосрочным явлением (происходит лишь несколько раз в год). В связи с этим для тестового датасета выбирались из исходного набора интервалы с наиболее поздними случаями отказов (по 3 для каждого технического места): за три месяца до отказа и оставшийся интервал после. Все предшествующие данные добавлялись в обучающий датасет.

Для последующей классификации выявленных аномалий создавался обучающий набор данных, содержащий данные за месяц до и после каждого отказа с меткой технического места, ставшего причиной отказа.

Процесс ранней диагностики отказов промышленного оборудования состоит из двух этапов (рисунок 2.9):

1. Идентификация аномального режима работы оборудования.
2. Классификация интервалов, определяемых как аномальные, по техническим местам, ставшими причиной отказа оборудования.

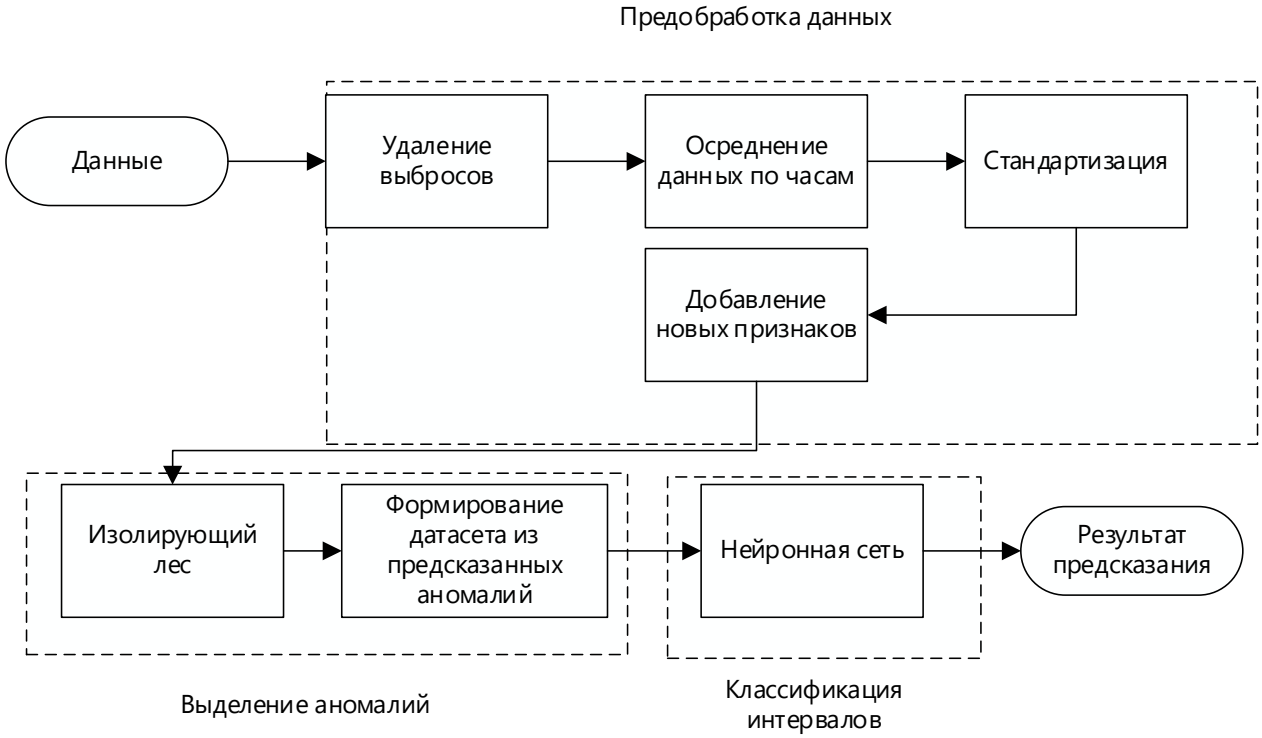


Рисунок 2.9 – Блок-схема работы модели прогнозирования

Изолирующий лес является одним из наиболее производительных методов поиска аномалий (рисунок 2.10) (Алгоритм 5). Суть его работы заключается в изоляции наиболее отличающихся точек от нормальных с помощью ансамбля бинарных деревьев решений (Алгоритм 6). Так как аномальные точки достаточно сильно будут отличаться от остальных, они достаточно быстро будут отнесены деревом решений в отдельную ветвь.

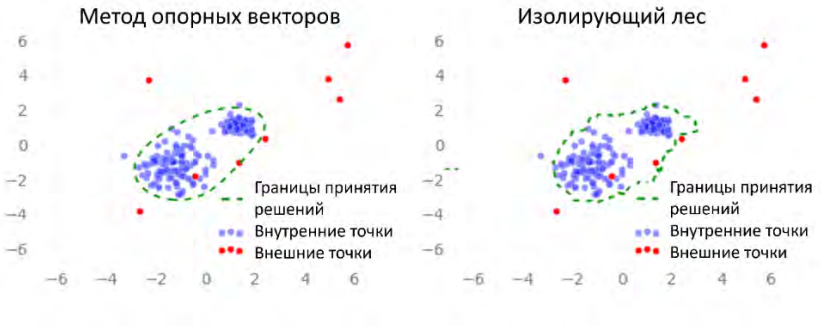


Рисунок 2.10 – Алгоритм выделения аномальных значений методом изолирующего леса

Алгоритм 5: Изолирующий лес

1: **Вход:** X – входящие данные, n – количество деревьев, ψ – размер выборки
2: лес ← пустой список размера n
3: $h_{max} \leftarrow \lceil log_2 |X| \rceil$
4: **для** i = 1 до n выполнить

5: $\hat{X} \leftarrow$ выборка (X, ψ)
6: лес[i] ← Изолирующее дерево $(\hat{X}, 0, h_{max})$
7: **Конец**
6: **Выход:** лес

Алгоритм 6: Изолирующее дерево

1: **Требуется:** Матрица X ϵ R^{nxp}
2: t – пустое дерево
3: **если** nrow(X) == 1 **тогда вернуть** t
4: **конец условия**
5: Случайно выбрать x_i признак X
6: Случайно выбрать разделяющие точки $p \in \{\min(x_i), \max(x_i)\}$
7: Добавить t к узлу N
6: Определить X_l и X_r как матрицы с примерами из X, где значения x_i больше и меньше p
7: Повторить алгоритм с X = X_l. Связать полученное дерево как левое дочернее ответвление t
8: Повторить алгоритм с X = X_r. Связать полученное дерево как правое дочернее ответвление t

Значение параметра contamination (задает долю аномальных значений в данных) было выбрано равным 0,03, что является достаточным для поиска режимов отказа оборудования. Изолирующий лес относится к методам машинного обучения «без учителя», потому, после обучения необходимо проверить его работу на конкретных примерах. На рисунке 2.11 показано, что периоды перед отказом оборудования изолирующий лес выделил как продолжительный режим аномальной работы, что позволяет использовать данный подход в решении подобной задачи.

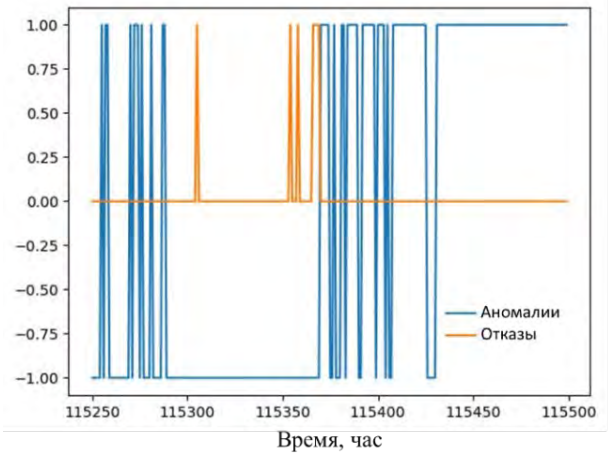


Рисунок 2.11 – Предсказание аномалий методом изолирующего леса (синий): 1 – нормальные точки, -1 – аномальные. Моменты отказа промышленного оборудования обозначены оранжевым (0 – нормальная работа, 1 – отказ оборудования)

Часть интервалов, размеченных моделью как аномальные, отказами не являются и к ним не ведут. Также данный метод не позволяет определить, в каком техническом месте развивается аномальный режим работы.

В связи с этим вводится второй этап предсказания, основанный на нейронной сети краткосрочной долговременной памяти LSTM (Long short-term memory) [134-140]. Данная нейронная сеть является одним из наиболее оптимальных методов работы с временными рядами, а суть метода заключается в анализе всей последовательности данных, с сохранением как особенностей отдельных интервалов, так и состояния всего периода (рисунок 2.12).

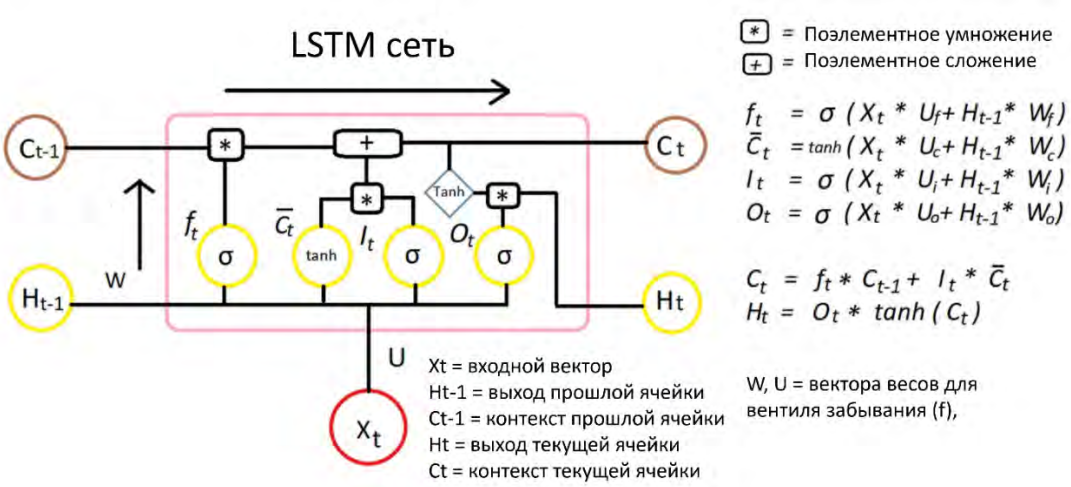


Рисунок 2.12 – Алгоритм работы LSTM слоя

Характерной особенностью архитектуры LSTM является наличие вентиля забывания (англ. forget gate), через который проходит вектор контекста (рисунок 2.13)

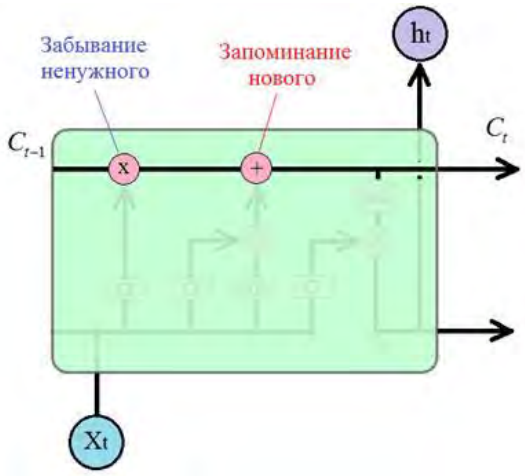


Рисунок 2.13 – Вентиль забывания RNN блока

Данная архитектура LSTM-блока обеспечивает эффективное сохранение и передачу долгосрочных зависимостей в процессе рекуррентных вычислений. Механизм поэлементного умножения выполняет селективную фильтрацию информации, отбрасывая нерелевантные данные, тогда как операция поэлементного сложения отвечает за аккумуляцию нового контекста. Входными параметрами для операции умножения служат: один (C_{t-1}) — вектор контекста, а второй (f_t)— оценочный вектор (рисунок 2.14).

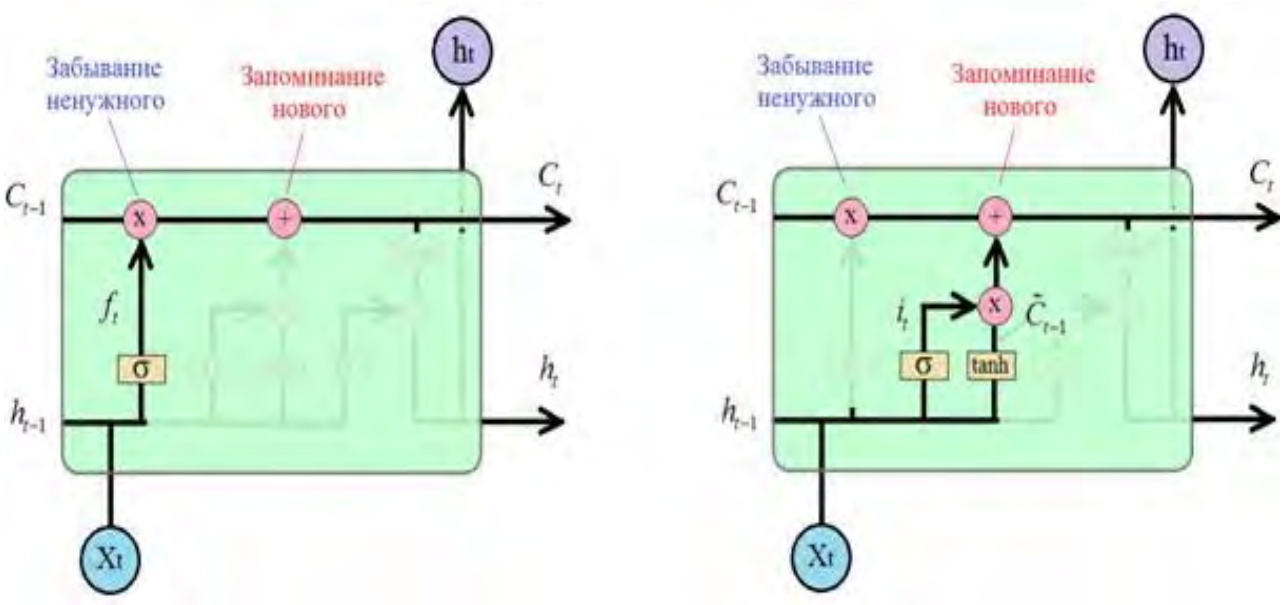


Рисунок 2.14 – Механизм выделения долгосрочной информации в RNN блоке

Из рисунка видно, что вектор f_t образуется как выход полносвязного слоя с сигмоидальной функцией активации:

$$f_t = \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right) \tag{2.9}$$

Здесь W_f - весовые коэффициенты этого слоя; $\left[h_{t-1}, x_t\right]$ - объединенный вектор из двух векторов; b_f - смещение; $\sigma(\cdot)$ - сигмоидальная функция (на выходе дает диапазон [0; 1]). В результате вектор f_t будет состоять из чисел от 0 до 1 и иметь ту же размерность, что и вектор контекста C_{t-1}:

$$f_t = \left[0,2; 0,3; 0,9; 0,1; 0,5, \ldots\right]^T \tag{2.10}$$

И при поэлементном умножении из вектора C_{t-1} будут убираться

незначимые, с точки зрения блока LSTM, величины. Что оставить, а что нет определяется в процессе ее обучения, то есть, подбором весовых коэффициентов W_f. Правильно обученная сеть будет, в среднем, корректно ослаблять ненужные величины вектора долгосрочного контекста C_{t-1}.

Давайте детально рассмотрим процесс обновления контекста через операцию поэлементного сложения. Добавляемая информация формируется следующим образом: на первом этапе система вычисляет управляющий оценочный вектор, определяющий значимость входящих данных, и временный вектор контекста, содержащий потенциально новую информацию для запоминания. Оба этих компонента рассчитываются по тому же принципу, который был подробно описан ранее для работы LSTM-сети x_t и предыдущего скрытого состояния h_{t-1}:

$$\begin{aligned} i_t &= \sigma\left(W_p \cdot \left[h_{t-1}, x_t\right] + b_i\right) \\ \tilde{C}_t &= \tanh\left(W_C \cdot \left[h_{t-1}, x_t\right] + b_C\right) \end{aligned} \tag{2.11}$$

а, затем, они поэлементно умножаются:

$$a_t = i_t \otimes \tilde{C}_t \tag{2.12}$$

Благодаря этому умножению из вектора добавочного контекста будет удаляться ненужная для долгосрочного запоминания информация. И, далее, сформированный вектор a_t поэлементно сложится с вектором C_{t-1}:

$$C_t = f_t \otimes C_{t-1} + a_t \oplus C_{t-1} \tag{2.13}$$

Так в блоке LSTM меняется долгосрочный контекст от итерации к итерации.

Наконец, последний этап работы блока – это формирование выходного скрытого состояния h_t (рисунок 2.15).

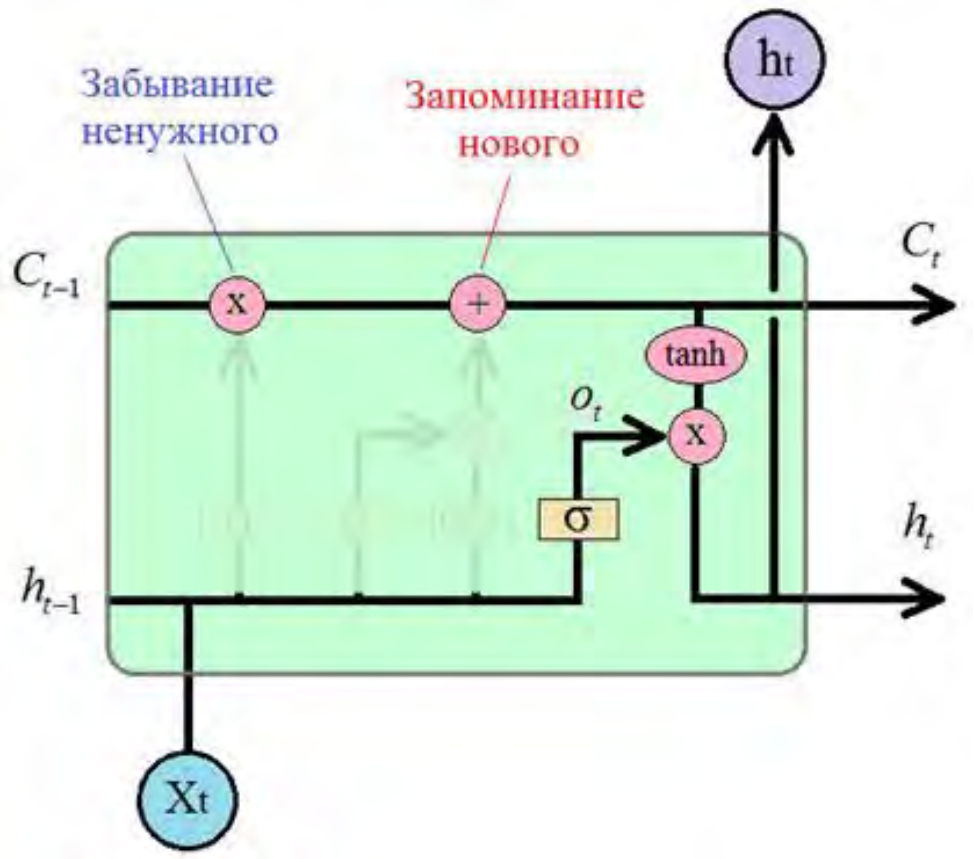


Рисунок 2.15 – Формирование скрытого состояния в блоке RNN

Принцип все тот же, сначала формируется оценочный вектор с помощью полносвязного слоя с сигмоидальной функцией активации:

$$o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \tag{2.14}$$

Далее, вектор долгосрочного контекста (памяти) C_t нормируется, проходя поэлементно через функцию tanh – гиперболический тангенс. На выходе получим значения в диапазоне [-1; 1]. Затем, эти нормированные величины поэлементно умножаются на оценочный вектор и формируется вектор скрытого состояния текущей итерации:

$$h_t = o_t \otimes \tanh(C_t) \tag{2.15}$$

Архитектура итогового классификатора (рисунок 2.16) состоит из одного LSTM-слоя (размер скрытого состояния = 16, выпрямленной линейной функции активации (ReLu)) и двух полносвязных слоёв с сигмоидной функцией активации на выходном слое.

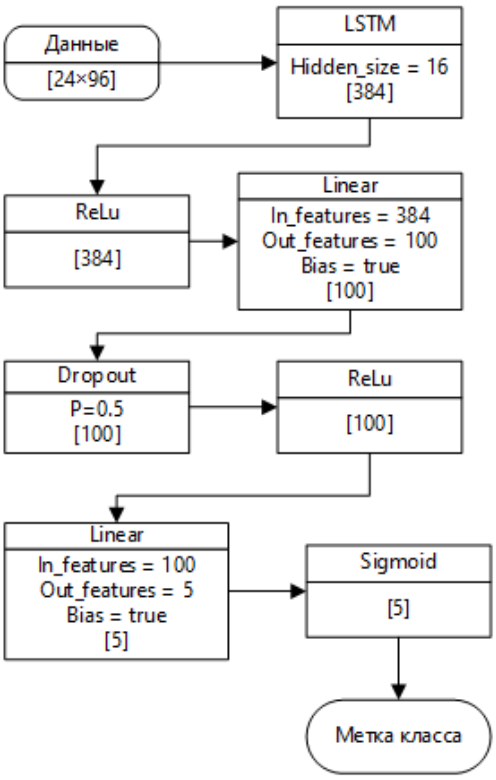


Рисунок 2.16 – Архитектура нейронной сети

Для обучения использовался оптимизатор AdamW (lr = 0.0001, weight_decay = 0.0001). В качестве функции потерь использовалась CrossEntropyLoss.

$$CrossEntropyLoss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{i,j}\cdot\log\left(\frac{e^{x_{i,j}}}{\sum_{k=1}^{C} e^{x_{i,k}}}\right)$$

(2.16)

Для борьбы с переобучением использовалась регуляризация: метод прореживания (Dropout) со значением отсева 0,6 и L2-регуляризация, задаваемая параметром weight_decay, которая штрафует модель за большие значения весов.

Для обучения классификатора в качестве отрицательного контроля использовались интервалы, размеченные изолирующим лесом как аномальные, но которые не приводят к отказу оборудования. Периодам, размеченным как аномальные (100 часов до отказа), присваивались метки технических мест,

неисправная работа в которых привела к простою. Обучение проводилось в течение 50 эпох. На вход алгоритму передаются массивы размера 24 x 96 (24 часа и признаки, полученные из показаний датчиков соответственно) и идентификатор класса (0 – нормальный режим работы, 1-4 – неисправное состояние технического места, ведущее к отказу). По итогу для каждого интервала выводится метка принадлежности к одному из классов.

Основной особенностью данных для решения задачи предсказания отказов является крайне резкая несбалансированность данных и наличие нескольких классов, что послужило основой для подбора подходящей метрики качества.

Для этого на каждом из классов по отдельности введем такие метрики как "точность (англ. Precision)" и "полнота (англ. Recall)".

$$Precision = \frac{TP}{TP+FP} \tag{2.17}$$

$$Recall = \frac{TP}{TP+FN} \tag{2.18}$$

Использование метрики precision обусловлено необходимостью избежать отнесения всех объектов к одному классу, что неизбежно увеличивает количество ложноположительных случаев (False Positive). Recall характеризует способность модели детектировать целевой класс, тогда как precision отражает её способность корректно дифференцировать данный класс от остальных (рисунок 2.17).

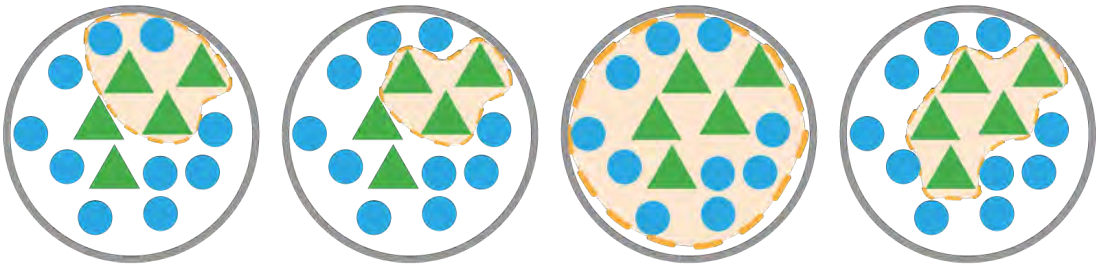


Рисунок 2.17 – Оценка разных стратегий предсказания. Зелеными треугольниками обозначены объекты класса 1, синими кружками – объекты класса 2

Часто в реальной практике стоит задача найти оптимальный баланс

между Presicion и Recall, для этого вводится такая метрика как F-мера. F-мера (в общем случае Fβ) — среднее гармоническое precision и recall:

$$F\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} \quad (2.19)$$

β в данном случае определяет вес точности в метрике, и при β=1 это среднее гармоническое (с множителем 2, чтобы в случае precision = 1 и recall = 1 иметь F1=1). F-мера достигает максимума при полноте и точности, равными единице, и близка к нулю, если один из аргументов близок к нулю. Таким образом, данная метрика качества является наиболее оптимальной при работе с несбалансированными наборами данных при решении задачи мультиклассовой классификации.

Используемая оценка F1 рассчитывается по формуле:

$$F1\text{-}score = \frac{2TP}{2TP + FP + FN}, \quad (2.20)$$

где F1-score – оценка F1, TP – истинно положительные значения из матрицы путаницы, FP – ложно положительные, FN – ложно отрицательные.

2.5 Метод прогнозирования RUL

2.5.1 Цель метода и постановка задачи

Целью данного метода является прогнозирование остаточного ресурса оборудования на основе показаний датчиков за определенный временной интервал. В данном случае решается регрессионная задача, что позволяет сделать предсказания более информативными, одновременно с эти возрастает и сложность обучения. Особенностью решения регрессионной задачи для данных, представленных временными рядами, является необходимость восстановления зависимости между показаниями датчиков и значением RUL [141], используя при этом информацию о прошлых динамических изменениях измеряемых физических характеристик. Задачу можно сформулировать следующим образом: по информации, заданной множеством D построить алгоритм F:

$$\forall\, p \in Di, S_t(D_i) \xrightarrow{F} RUL_t(p) \quad (2.21)$$

который позволяет по входящим значениям датчиков за определенный

временной интервал (S_t) единицы оборудования (D_i) оценить для каждого технического места (p) остаточный временной ресурс (RUL_t).

Важно отметить, что зависимость между временными изменениями показаний датчиков и оценками RUL в общем случае является нелинейной. При этом одной из ключевых проблем для решения данной задачи является подбор горизонта прогнозирования для каждого типа отказа в зависимости от природы развития дефекта. При этом для восстановления зависимости между показаниями датчиков и износа оборудования обучающая выборка должны содержать не только вектор признаков для момента времени, для которого происходит предсказание, но и более крупного временного интервала, предшествующего анализируемому. В связи с данной особенностью наиболее оптимальными являются нейросетевые рекуррентные алгоритмы.

Предсказание RUL является наиболее предпочтительным форматом оценки состояния оборудования из всех ранее описанных, так как позволяет оценить не только состояние оборудования в конкретный момент времени, но и динамику развития его износа (рисунок 2.18). Однако реализация такого подхода, как правило, выводит предсказания с достаточно большой ошибкой, которая может сглаживаться при комплексном применении всех вышеописанных методов.

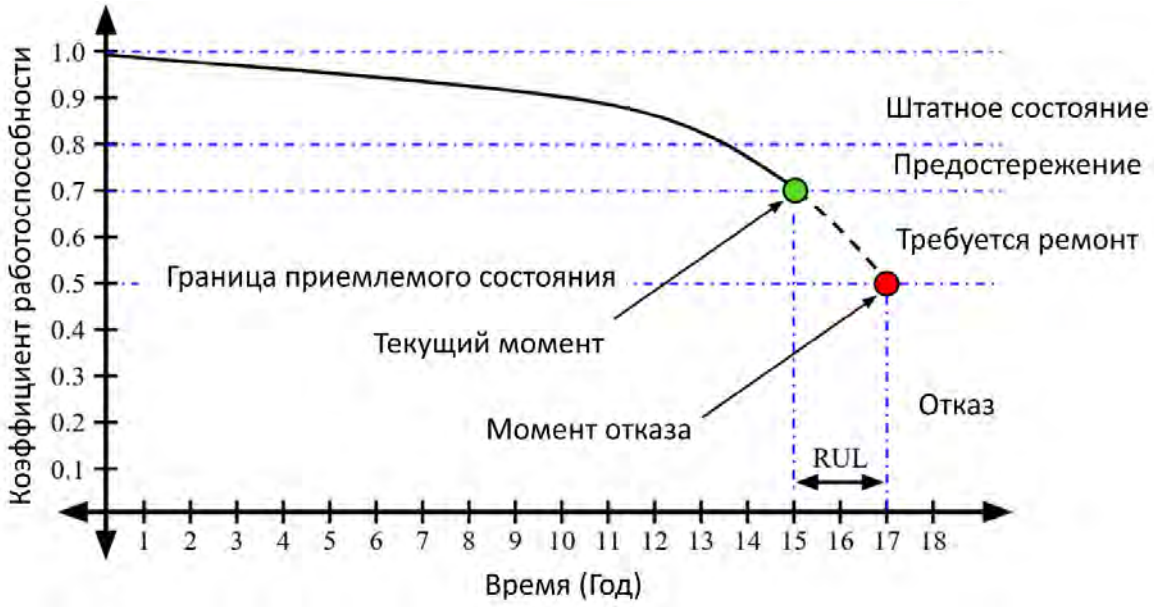


Рисунок 2.18 – Этапы функционирования оборудования

2.6 Выводы по второй главе

1. В данной главе были рассмотрены подходы к построению функциональной структуры автоматизированной системы прогнозирования. В процессе анализа выявлено, что наиболее распространёнными подходами являются модульный и алгоритмический, каждый из которых имеет свои особенности и преимущества. Была разработана новая функциональная структура автоматизированной системы, опирающаяся на оба подхода.

2. Для прогнозирования неисправностей применяется ансамблевый подход - "мягкое голосование" между алгоритмами классического машинного обучения (логистическая регрессия и метод опорных векторов) и сверточной нейронной сетью, позволяющий добиться высокой точности предсказаний.

3. Прогнозирование отказов осуществляется в два этапа. На первом этапе обучается модель на нормальном состоянии оборудования для определения аномального режима работы с использованием метода «изолирующего леса». Далее производится классификация аномалий по техническому месту, ставшему причиной внепланового простоя оборудования, с помощью нейронной сети на основе архитектуры LSTM.

4. Для прогнозирования RUL разработана рекуррентная нейронная сеть на базе архитектуры LSTM. Она является оптимальной для работы с временными рядами, так как способна учитывать информацию о состоянии оборудования за длительные периоды.

5. Для борьбы с переобучением использовались метод регуляризации Dropout и оптимизация с помощью алгоритма Adam, что позволяет улучшить обобщающую способность модели и повысить её устойчивость к шумам в данных.

ГЛАВА 3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ФУНКЦИОНАЛЬНОЙ СТРУКТУРЫ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ЭКСГАУСТЕРА АГЛОМЕРАЦИОННОЙ МАШИНЫ

3.1 Объект технической диагностики – эксгаустер агломашины.

Для обучения моделей использовались данные, полученные от ПАО «Северсталь», а в качестве оборудования выступают эксгаустеры агломерационной машины – вентилятор, осуществляющий просос воздуха через слой шихты.

3.1.1 Описание технологии процесса спекания.

Процесс спекания агломерата начинается с воспламенения верхнего слоя шихты в камерном горне, оснащенном четырнадцатью горелками, распределенными по трем секциям: начальная секция содержит 4 горелки, центральная - 6, и завершающая - 4 горелки [91-93]. Подача шихты осуществляется барабанным питателем из бункер-воронки на наклонный лоток с последующей укладкой на спекательные тележки. Критически важным параметром является регулировка высоты слоя (330-400 мм) посредством шибера, учитывающая газопроницаемость материала - отклонения от оптимального значения могут вызвать ухудшение качества агломерата, рост энергозатрат и увеличение доли возвратного продукта.

Начальная секция горна отвечает за розжиг шихты, тогда как последующие обеспечивают равномерное прогревание и стабилизацию процесса спекания. В системе используются турбулентные газовые горелки, работающие на природном газе с регулируемым давлением через газорегуляторные узлы, при этом воздух подается вентилятором "ВМ 18А". Для защиты оборудования от термических повреждений предусмотрена система водяного охлаждения металлоконструкций. Ключевым фактором эффективности процесса выступает равномерность распределения шихты по всей площади спекания. Включение газовой системы допускается исключительно при подтвержденной возможности воспламенения от

открытого пламени или раскаленного слоя шихты. Давление в газовой магистрали должно поддерживаться на уровне не менее 300 мм вод. ст., при падении ниже этого показателя происходит автоматическое отключение подачи газа с последующей остановкой агломерационной машины.

Критически важным параметром является поддержание температурного режима в зоне зажигания в пределах 1300-1450°С, что обеспечивается регулированием расходных характеристик газовоздушной смеси: оптимальные значения составляют 550-600 м³/ч для природного газа и 6500-7000 м³/ч для воздуха. Мониторинг параметров горения осуществляется как с помощью контрольно-измерительных приборов, так и визуально по цветовой характеристике пламени - синеватый оттенок свидетельствует о избыточной подаче воздуха, тогда как светло-белое пламя указывает на его недостаток.

Эффективность процесса зажигания шихты определяется комплексом факторов: оптимальным соотношением компонентов газовоздушной смеси, стабильными показателями влажности и содержания углеродистых материалов в шихте, а также равномерностью распределения шихтового материала по поверхности паллет. Технология спекания реализуется в строгом соответствии с утвержденной технологической картой, учитывающей текущее состояние оборудования и нормативные показатели расхода шихтовых материалов. Скоростной режим работы агломерационных машин преимущественно корректируется по показателям скорости фильтрации дымовых газов.

Наиболее значимые физико-химические преобразования происходят в зоне высокотемпературного плавления, где формирующиеся химические соединения определяют конечные характеристики агломерата. Особое значение имеют процессы газообразования: двуокись углерода образуется как продукт сгорания углеродистых компонентов (коксовой мелочи или антрацита), так и в результате термического разложения карбонатных соединений фосфорита, степень декарбонизации которого достигает 95% [91-93].

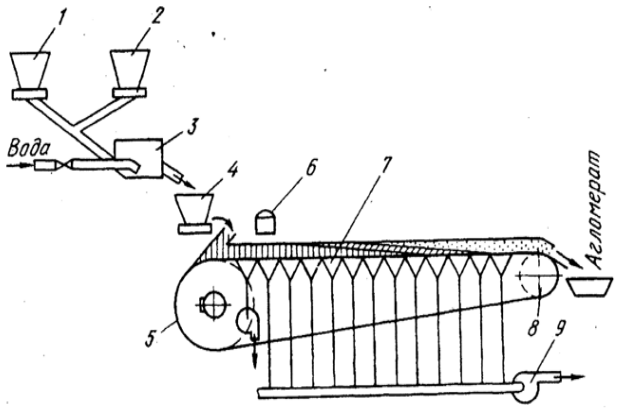
Степень декарбонизации представляет собой ключевой параметр

качества агломерата, оказывающий непосредственное влияние на эффективность последующих технологических процессов в фосфорных печах. Полноценная декарбонизация способствует существенному снижению энергопотребления, уменьшению объема образующихся печных газов и повышению чистоты конечного фосфора. Технологический процесс требует строгого соблюдения режимов спекания - работа агломерационных машин с недостаточно прогретой шихтой категорически недопустима.

При обнаружении повышенного содержания углеродистых компонентов в шихтовой смеси необходимо оперативно корректировать скорость движения агломашины для обеспечения необходимой глубины прогрева материала. Контроль содержания углерода осуществляется путем визуального анализа состояния раскаленной поверхности спека после прохождения зоны горения. Оптимальные условия процесса характеризуются равномерным пропеканием по всему объему пирога, отсутствием недопеченных участков и стабильной структурой материала. Избыточное количество топлива приводит к нежелательным изменениям морфологии агломерата - чрезмерному оплавлению поверхности, образованию крупных пор и частичному прилипанию к элементам колосниковой решетки, что негативно сказывается на качестве продукции и эксплуатационных характеристиках оборудования. [94].

3.1.2 Характеристика и конструкция агломашины

В современной металлургии доминирующее положение занимает технология ленточной агломерации, реализуемая на машинах непрерывного действия [95-97]. Принцип работы основан на создании воздушного потока, проходящего через слой шихты. Конструктивно такие установки состоят из следующих основных компонентов: несущего каркаса, замкнутой цепи спекательных тележек, приводной станции, системы загрузки шихтовых материалов с питателями, подстилающего слоя ("постели"), зажигательного устройства, вакуумных камер и автоматизированной системы смазки узлов трения (рисунок 3.1).



1, 2 – бункеры, 3 – барабанный смеситель, 4 – промежуточный бункер,
5 – ведущий барабан, приводится в движение двигателем постоянного тока, 6 – зажигательный горн,
7 – вакуум-камеры, 8 – ведомый барабан машины, 9 – эксгаустер

Рисунок 3.1 – Ленточная агломерационная машина непрерывного действия

Спекательная тележка представляет собой стальную конструкцию на четырех роликоопорах с боковыми бортами, внутри которой расположена колосниковая решетка [98]. Процесс спекания включает комплекс физико-химических превращений, при которых сжигание топлива в потоке просасываемого воздуха создает высокотемпературные условия (1400-1600°C), необходимые для плавления материалов и формирования агломерата с заданными свойствами [99]. Ключевыми технологическими параметрами выступают: температура поверхности шихты, высота слоя, скорость спекания (0,15-0,7 мм/с), скорость движения ленты (60-120 мм/с) и степень завершенности процесса.

На начальной стадии зажигания критическое значение имеет поддержание в горне продуктов сгорания с содержанием кислорода 8-10% и оптимального давления для предотвращения теплопотерь [98]. Создаваемое в вакуум-камерах разрежение регулирует скорость теплопередачи - при недостаточном значении замедляется прогрев нижних слоев, а чрезмерное разрежение приводит к перерассеиванию тепла и ухудшению качества спека.

Технологический процесс на агломашине включает три последовательные зоны [99]:

1) зону зажигания (lз) с воспламенением топливных частиц

2) зону спекания (lс) с формированием агломерационной структуры

3) зону охлаждения (lo) готового продукта

При этом зона горения толщиной 15-35 мм перемещается вниз, образуя характерный наклонный фронт (рисунок 3.2), чья форма и динамика определяются соотношением вертикальной скорости спекания и горизонтального движения ленты.

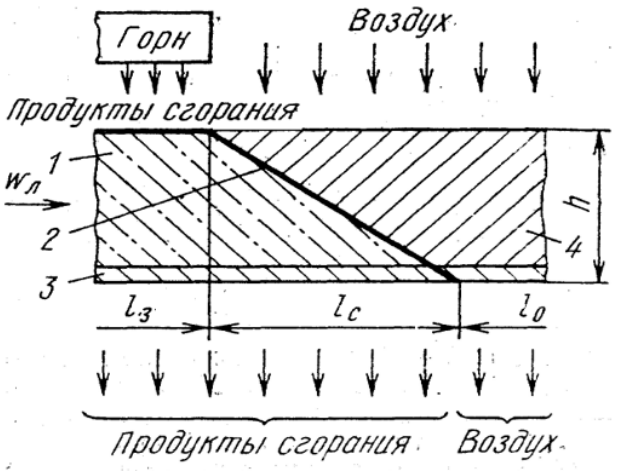


Рисунок 3.2 – Схема спекания шихты на агломашине

В установившемся режиме работы агломерационной машины основные технологические параметры - высота слоя шихты (h), время спекания (τ_c) и скорость движения ленты ($w_л$) - находятся в строгой взаимозависимости. Скорость движения ленты должна быть оптимизирована для обеспечения полного завершения процесса спекания на заданном участке, что является критически важным условием формирования качественного пористого агломерата. Температурный режим требует особого контроля, так как непосредственно определяет свойства конечного продукта. При нормальном ходе процесса агломерат должен характеризоваться равномерной степенью спекания по всему объему, при этом при выгрузке раскаленная часть не должна превышать одной трети высоты "пирога". Диагностика качества процесса осуществляется по нескольким ключевым показателям. О незавершенности спекания свидетельствуют пониженная температура отходящих газов в конечных вакуум-камерах и наличие недопеченной шихты в изломе агломерационного "пирога". Повышение температуры отходящих газов может быть следствием как технологических нарушений (замедления скорости

паллет или временной остановки машины), так и улучшения газопроницаемости шихты. Снижение температурных показателей, в свою очередь, может объясняться уменьшением топливной составляющей в шихте, переплавлением поверхностного слоя из-за чрезмерно высокой температуры зажигания, наличием паразитных подсосов воздуха либо превышением оптимальной скорости движения паллет, приводящим к сокращению времени термической обработки. Таким образом, стабильность технологического процесса и качество получаемого агломерата напрямую зависят от точного соблюдения и постоянного контроля трех основных параметров: высоты шихтового слоя, скорости движения агломерационной ленты и температурного режима спекания. Любые отклонения от оптимальных значений этих параметров неизбежно приводят к ухудшению характеристик готового продукта, что находит отражение как в изменении температуры отходящих газов, так и в структурных особенностях формируемого агломерата.

3.1.3 Основные технологические параметры работы эксгаустера

Агломерационные установки включают два ключевых технологических компонента: собственно агломерационные машины и эксгаустерные системы. Эти элементы работают в тесной взаимосвязи, создавая необходимые условия для эффективного протекания процесса спекания. Эксгаустеры выполняют двойную функцию: во-первых, они обеспечивают подачу требуемого объема воздуха через спекаемый слой, а во-вторых, отводят образующиеся газообразные продукты химических реакций. Такая организация воздушных потоков позволяет поддерживать оптимальные условия для протекания всех стадий агломерационного процесса - от зажигания шихты до формирования конечного продукта с заданными физико-химическими свойствами. [100-102].

Основные химические реакции, протекающие при спекании шихты:

$$Al_2O_3 + Na_2CO_3 \xrightarrow{700–1150^\circ C} Na_2O \cdot Al_2O_3 + CO_2\uparrow,$$

алюминат натрия

$$Fe_2O_3 + Na_2CO_3 \xrightarrow{700–1000^\circ C} Na_2O \cdot Fe_2O_3 + CO_2\uparrow,$$

феррит натрия

$$SiO_2 + Na_2CO_3 \xrightarrow{550\text{–}850^\circ C} Na_2O \cdot SiO_2 + CO_2\uparrow,$$

силикат натрия

$$SiO_2 + 2CaCO_3 \xrightarrow{\sim 900\text{–}1150^\circ C} 2CaO \cdot SiO_2 + 2CO_2\uparrow,$$

двухкальциевый силикат

$$2[Na_2O \cdot SiO_2] + Na_2O \cdot Al_2O_3 \xrightarrow{700\text{–}1000^\circ C} Na_2O \cdot Al_2O_3 \cdot 2SiO_2 + 2Na_2O,$$

$$Na_2O \cdot Al_2O_3 \cdot 2SiO_2 + 4CaO \xrightarrow{\sim 900\text{–}1150^\circ C} 2(CaO \cdot SiO_2) + Na_2O \cdot Al_2O_3,$$

$$Na_2O + Al_2O_3 \xrightarrow{700\text{–}1000^\circ C} Na_2O \cdot Al_2O_3,$$

$$CaCO_3 + C \xrightarrow{800\text{–}900^\circ C} CaO + CO_2$$

Эксгаустер представляет собой центробежный вентилятор, предназначенный для работы в условиях значительного гидравлического сопротивления на всасывающей линии (рисунок 3.3). Агрегат конструктивно включает два основных компонента: неподвижный корпус и вращающийся ротор. Корпус оборудован тремя патрубками - двумя входными и одним выходным, расположенными по касательной к траектории вращения рабочего колеса. Роторная часть состоит из центрального диска, двух боковых крышек и закрепленных между ними лопаток, образующих основную проточную полость устройства.

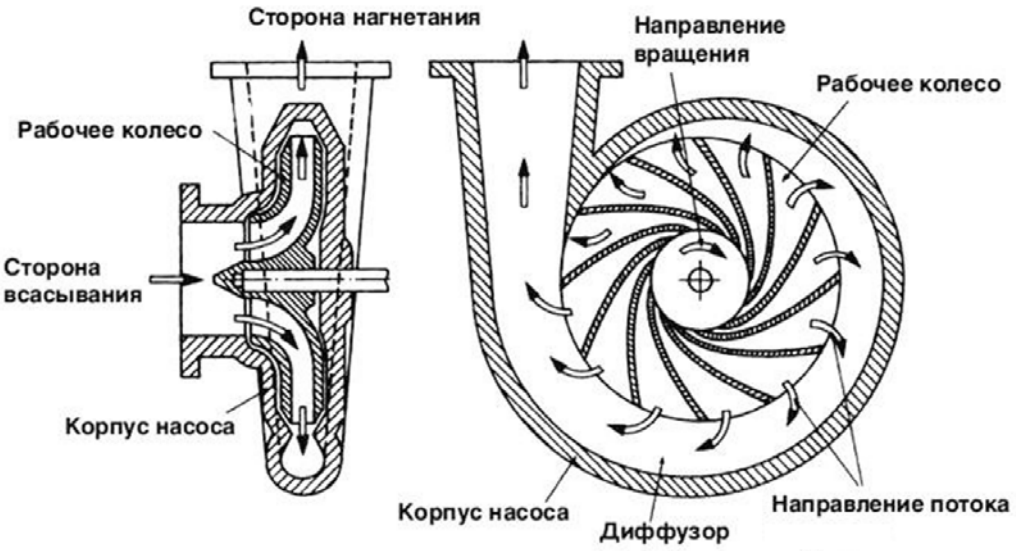


Рисунок 3.3 – Схема эксгаустера

При прохождении через межлопаточные каналы ротора осевой газовый поток преобразуется в спиралевидный. Центробежный эффект вызывает резкое нарастание скорости и кинетической энергии потока по мере его перемещения к периферии. Генерируемый напор прямо пропорционален разнице энергетических уровней газа на входном и выходном участках. Математическое описание этого процесса дается основным уравнением центробежного вентилятора, исторически выведенным выдающимся русским математиком Леонардом Эйлером. При теоретическом выводе данного уравнения были приняты следующие идеализированные допущения: полное отсутствие потерь энергии как вследствие внутреннего трения в газовом потоке, так и при его взаимодействии с поверхностями рабочего колеса; а также предположение о бесконечном количестве лопаток с ничтожно малым шагом их расположения. Эти допущения позволяют существенно упростить математическую модель при сохранении ее принципиальной физической достоверности:

$$H_T = \psi\rho\left(\frac{\pi D n}{60}\right)^2 \quad (3.1)$$

где ψ - коэффициент формы лопаток ротора.

Анализ уравнения (3.1) однозначно подтверждает, что теоретический напор центробежного вентилятора (эксгаустера) прямо пропорционален плотности газа ρ и квадратично зависит от диаметра ротора D и скорости его вращения. В агломерационных установках принципиально важна конструкция эксгаустеров с обратнозагнутыми лопатками – такое исполнение ротора не просто повышает, а максимизирует КПД, что критически важно для энергоэффективности всего оборудования:

$$H = k \cdot H_T, \quad (3.2)$$

где $k = 0{,}5 \div 0{,}85$ - поправочный коэффициент на конечное число лопаток рабочего колеса.

Основные технические параметры работы эксгаустера: производительность или подача *V* (*м³/час)*, напор *H (м)*, мощность *N (Вт)*, и коэффициент полезного действия η.

Полезной мощностью является полное приращение энергии, получаемое всем потоком газа в эксгаустере в единицу времени:

$$N_{пол} = V \cdot H \tag{3.3}$$

Отношение полезной мощности $N_{пол}$ к потребляемой *N* представляет собой полный к.п.д. эксгаустера в единицу времени:

$$\eta = \frac{N_{пол}}{N} = \frac{VH}{N} \tag{3.4}$$

Все виды потерь в эксгаустере делятся на три категории:

- гидравлические;
- объемные;
- механические.

Гидравлические потери обусловлены затратами на преодоление гидравлических сопротивлений $H_{П}$ при движении потока внутри эксгаустера. Гидравлический к.п.д. центробежного вентилятора определяется из соотношения:

$$\eta_{Г} = \frac{H_{Т} - H_{П}}{H_{Т}} = \frac{H}{H_{Т}} \tag{3.5}$$

Объемные потери обусловлены подсосами воздуха в самом эксгаустере из-за не плотностей корпуса. Объемный к.п.д. эксгаустера представляет отношение количества газа, выходящего из эксгаустера, к количеству засасываемого газа:

$$\eta_{об} = \frac{V_1 - V_{П}}{V_1} = \frac{V}{V_1} \tag{3.6}$$

Объемный к.п.д. обычно равен (для новых вентиляторов) $0{,}95 \div 0{,}96$.

Механические потери в эксгаустере обусловлены главным образом затратами энергии на преодоление сил трения в его подвижных узлах, прежде всего в подшипниковых опорах и сальниковых уплотнениях вала ротора. Эти потери количественно оцениваются механическим коэффициентом полезного

действия, который характеризует долю полезной мощности, не расходуемой на компенсацию механического сопротивления. Механический КПД является важным показателем, отражающим эффективность преобразования подводимой энергии в полезную работу без учета гидравлических и объемных потерь системы:

$$\eta_{мех} = \frac{N - N_{тр}}{N}. \tag{3.7}$$

где $N_{тр}$ - общая мощность трения внутри эксгаустера.

Для новых вентиляторов $\eta_{мех} = 0{,}97 \div 0{,}98$.

Полный к.п.д. равен произведению составляющих:

$$\eta_O = \eta_Г \cdot \eta_{об} \cdot \eta_{мех} = \frac{H}{H_T} \cdot \frac{V}{V_1} \cdot \frac{V_1 \cdot H_T}{N} = \frac{V \cdot H}{N} \tag{3.8}$$

Полный к.п.д. вентилятора определяется опытным путем ($\eta_O = 75 \div 92\%$);

Объемный и механический к.п.д. рассчитывается или определяется опытом.

Мощность двигателя, приводящего эксгаустер в действие, определяется из соотношения:

$$N_{ДВ} = a \frac{N}{\eta_П}, \tag{3.9}$$

где $\eta_П$ - к.п.д. передачи от двигателя к эксгаустеру, обычно $\eta_П = 1$, т.к. применяется прямая передача;

$a = 1{,}1 \div 1{,}2$ - коэффициент запаса мощности двигателя.

Для одного и того же эксгаустера с изменением скорости вращения ротора основные показатели работы (производительность *V*, напор *H* и потребляемая мощность *N*) меняются следующим образом:

$$V_2 = V_1 \frac{n_2}{n_1}, \tag{3.10}$$

$$H_2 = H_1 \left(\frac{n_2}{n_1}\right)^2 \tag{3.11}$$

$$N_2 = N_1 \left(\frac{n_2}{n_1}\right)^3 \tag{3.12}$$

где n_1 и n_2 – исходная и измененная скорость вращения ротора.

Рабочие характеристики эксгаустера представляют собой графические зависимости напора, потребляемой мощности и к.п.д. эксгаустера от его производительности при постоянной скорости вращения и плотности газа [103]:

$$H = \phi_1(V),\ N = \phi_2(V),\ \eta = \phi_3(V).$$

Обычный вид напорной характеристики эксгаустера приведен на рисунке 3.4.

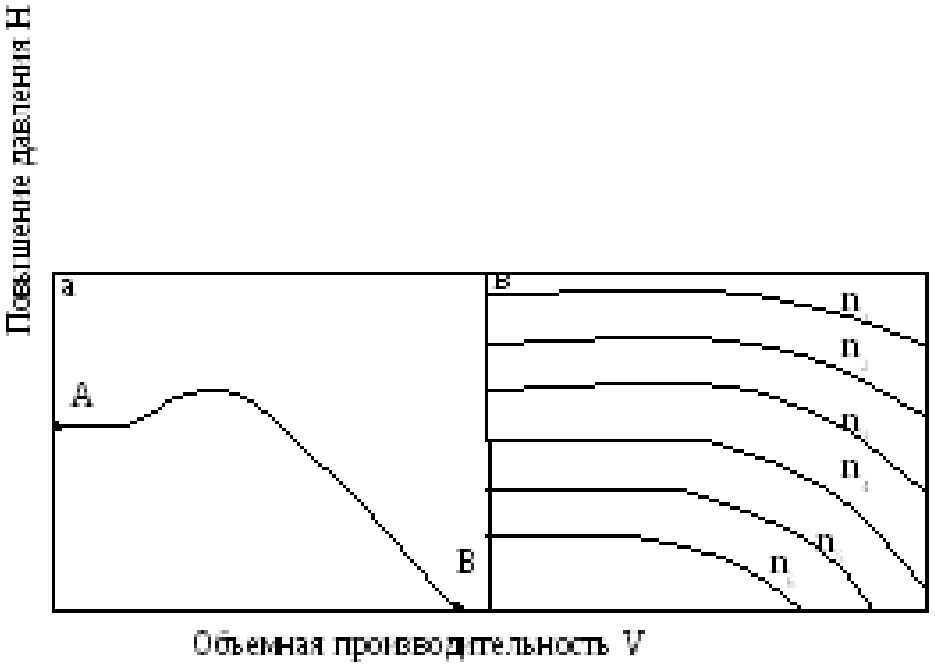


Рисунок 3.4 – Напорная характеристика центробежных вентиляторов: а – общий вид; в – семейство характеристик одного вентилятора при различной скорости вращения ротора: $n_1>n_2>n_3>n_4>n_5>n_6$.

Напорная характеристика центробежного эксгаустера формируется между двумя предельными режимами эксплуатации. Первый предельный режим (точка А) соответствует условиям холостого хода, когда производительность системы стремится к нулю вследствие полного перекрытия всасывающего или нагнетательного патрубка. В этом случае создается максимально возможное статическое давление при отсутствии газового потока. Второй предельный режим (точка В) характеризуется работой агрегата на свободный выброс, когда статическое давление на выходе падает

до нуля при максимально достижимой производительности системы. Нормальный эксплуатационный режим работы эксгаустера определяется точкой пересечения его напорной характеристики с гидравлической характеристикой подключенной сети. Эта рабочая точка всегда располагается на нисходящей ветви напорной кривой, в области, соответствующей максимальным значениям коэффициента полезного действия. Такое положение обеспечивает оптимальное соотношение между развиваемым давлением и производительностью, минимизируя энергетические потери при транспортировке газового потока через технологическую систему. Экспериментальное определение рабочей характеристики эксгаустера (рисунок 3.5) позволяет установить, что напорная характеристика может быть математически описана степенной функцией, отражающей зависимость между ключевыми параметрами работы оборудования:

$$H = \rho_T \left(a + bV_T - cV_T^2\right) = A + BV_T - CV_T^2 \tag{3.13}$$

где a, b и c – эмпирические коэффициенты, V_T – объемная производительность.

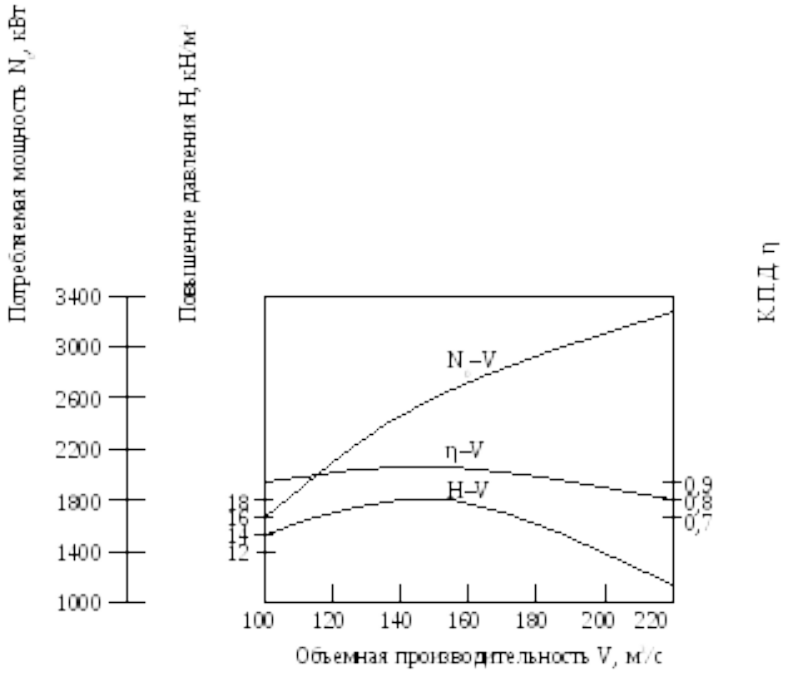


Рисунок 3.5 – Рабочие характеристики эксгаустера Д–9000–11–3

Оптимизация работы агломерационных машин может достигаться за счет применения нескольких эксгаустеров, подключенных к единой сети. Последовательное соединение двух агрегатов позволяет создать повышенное разрежение и увеличить объем прокачиваемого воздуха через спекаемый слой, однако экономическая эффективность такого решения существенно зависит от сопротивления сети - при низком сопротивлении дополнительные энергозатраты могут нивелировать выигрыш в производительности. В практике железорудной агломерации последовательная схема подключения практически не применяется. Более эффективным решением при недостаточной производительности является параллельное подключение эксгаустеров, при котором общая характеристика системы определяется суммой подач отдельных агрегатов при одинаковом напоре. Для построения суммарной характеристики производительности необходимо просуммировать значения подачи каждого эксгаустера при равных значениях напора и по полученным точкам построить общую кривую. Такая схема позволяет существенно увеличить объем прокачиваемого воздуха без пропорционального роста энергопотребления, что особенно важно при работе с высокопроницаемыми шихтами или при необходимости интенсификации процесса спекания.

Технические характеристики нагнетателя:

Объемная производительность, $м^3$/мин	12000
Повышение давления (разность между абсолютными статическими величинами конечным и начальным давлением) , мм. вод. ст.	1350
Потребляемая мощность, кВт	3100
Начальная температура газа, °С	150
Число оборотов ротора нагнетателя, об/мин	1260
Критическое число оборотов ротора, об/мин	1876
Мощность электродвигателя, кВт	4000
Номинальная нагрузка электродвигателя, А	270
Вес электродвигателя ,т	30
Вес редуктора , т	5,3
Вес нагнетателя в сборе ,т	75

3.1.4 Эксгаустер агломашины как объект управления

Эксгаустер оснащён системой управления и контроля основных параметров, а также системой автоматической защиты. Для осуществления управления контроля и автоматической защиты агрегат оснащен щитом, который установлен в помещении машинного зала. Система управления предусматривает пуск нагнетателя с соблюдением заданной последовательности операций и предотвращения неправильного действия обслуживающего персонала путем введения в систему необходимых блокировочных связей.

Система контроля предусматривает измерение следующих параметров, характеризующих работу агрегата:

- Контроль температур:
  − подшипников нагнетателя и электродвигателя;
  − обмотки статора и активного железа главного электродвигателя.
- Контроль давлений:
  − масла в системе смазки.
- Контроль вибрации ротора и электродвигателя:
  − осевой
  − радиальный

Шкаф управления установлен в помещении машинного зала. На шкафу управления установлены следующие ключи, кнопки, сигнальные лампы и приборы:

− ключ управления главным двигателем;

− избиратель управления (местное, дистанционное);

− ключ выбора режима вибродатчика;

− кнопки пуска и остановки маслонасоса;

− кнопки съема сигнала вибродатчика, командной сигнализации задвижки, сигнала аварийного отключения;

− кнопки открытия, закрытия задвижки и остановки задвижки;

− лампы «задвижка открыта» и «задвижка закрыта»;

− лампы «перегрев подшипников», «отсутствие напряжения на маслонасосе»;

− лампы повышенной вибрации, падения возбуждения главного электродвигателя, аварийного отключения эксгаустера, нарушения нормальной работы эксгаустера;

− лампы «не включен маслонасос», «масляный выключатель включен», «масляный выключатель отключен», сигнала на открытие задвижки, сигнала на закрытие задвижки;

− мегаваттметр;

− амперметр.

Эксгаустер имеет следующие блокировки:

Блокировки, позволяющие осуществить пуск главного электродвигателя только при давлении масла в системе смазки более 0,25 МПа. Импульс на разрешение пуска подается от электроконтактного манометра;

Блокировка маслонасосов предусматривает:

− автоматическое включение пускового насоса при падении давления масла до 0,25 кгс/см^2 (0,025 МПа);

− автоматическое включение резервного маслонасоса при падении давления масла до 0,2 кгс/см^2 (0,02 МПа).

Сигнализация - при достижении предельных значений измеряемых величин предусмотрена подача предупредительных световых и звуковых сигналов и аварийных светового и звукового сигнала.

В схеме аварийной остановки главного электродвигателя и подачи аварийного сигнала использованы электрические импульсы от электроконтактного манометра при снижении давления масла в системе смазки до 0,2 кгс/см^2 (0,02 МПа).

В схеме подачи предупредительного светового и звукового сигналов использованы следующие импульсы:

- от электроконтактного манометра при падении давления масла в системе смазки до 0,25 кгс/см^2 (0,025 МПа);

- от термометров, вибродатчиков сигнализирующих ТС-100 при повышении температуры подшипников нагнетателя и главного электродвигателя.

Эксгаустер будет немедленно автоматически остановлен в следующих случаях:

− при повышении вибрации;

− при снижении давления масла в напорной линии маслосистемы до 0,2 кгс/см^2 (0,02 МПа).

Эксгаустер должен быть немедленно остановлен нажатием кнопки аварийной остановки в следующих случаях:

− при появлении дыма из подшипников;

− при быстром, внезапном возрастании температуры подшипника;

− при появлении металлического шума в агрегате;

− при появлении условий, создающих угрозу безопасности обслуживающего персонала или поломки оборудования.

Следовательно, автоматизация эксплуатации эксгаустера имеет множество достоинств, включая повышение безопасности и надежности работы. Однако текущее устройство системы контроля не предполагает контроля ложных сигналов (например, из-за неисправности датчиков), что может привести к необоснованным остановкам оборудования. Также система не предполагает прогнозирования технического состояния оборудования, следствием чего является необходимость частого технического обслуживания как датчиков, так и самого эксгаустера.

3.2 Реализация функциональной структуры автоматизированной системы прогнозирования технического состояния эксгаустера агломашины

Внедрение системы удаленного мониторинга и прогнозирования позволяет эффективно управлять технологическими процессами, минимизировать риски возникновения неполадок и сокращать незапланированные простои. Это решение повышает надежность работы

оборудования и обеспечивает существенную экономию средств. На рисунке 3.6 отображена структура предложенной автоматизированной системы прогнозирования технического состояния промышленного оборудования.

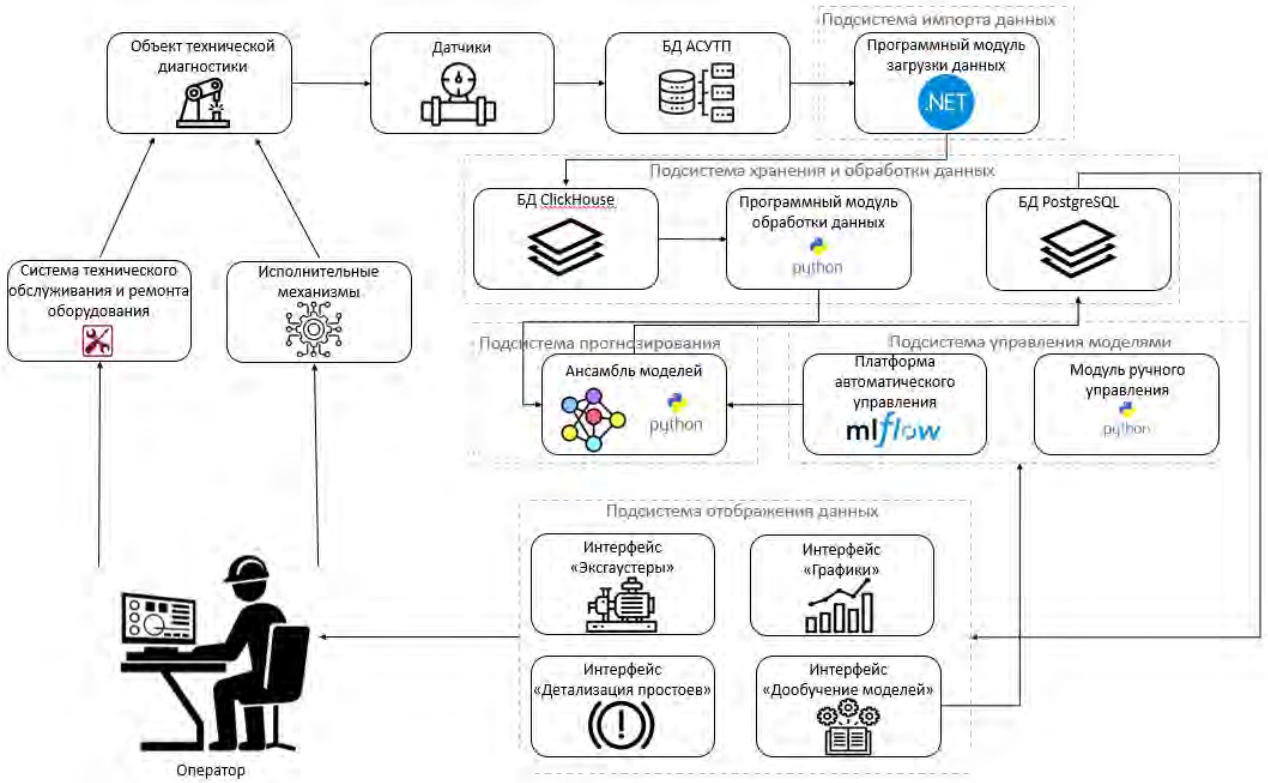


Рисунок 3.6 – Архитектура разработанной автоматизированной системы прогнозирования технического состояния промышленного оборудования

На первом этапе исходные массивы данных подсистемой импорта загружаются в подсистему хранения (отвечающую за автоматическую предобработку и длительное хранения как исходных данных, так и предобработанных), где происходит их предобработка. Далее предобработанные данные отправляются в подсистему прогнозирования, в которой реализованы модели, предсказывающие неисправности и отказы промышленного оборудования. Запуски процессов в этой подсистеме осуществляются автоматически через подсистему управления моделями, в том числе и дообучение моделей на непрерывно поступающих актуальных данных. Результаты работы моделей также записываются в подсистему хранения. Взаимодействие с моделями может осуществляться в ручном режиме через подсистему отображения данных, в которой отображаются как результаты работы моделей, так и исходные данные, которые могут использовать

конечным пользователем для мониторинга в режиме реального времени. Далее каждая из подсистем будет рассмотрена более подробно.

3.2.1 Разработка подсистем импорта данных

Разработка подсистемы импорта данных в автоматизированной системе прогнозирования технического состояния промышленного оборудования представляет собой важный этап, который обеспечивает интеграцию и обработку информации, поступающей из различных источников. Данная подсистема служит связующим звеном между физическими устройствами, такими как датчики и системы мониторинга, и аналитическими инструментами, которые используются для предсказания состояния оборудования и его возможных отказов.

Первым шагом в разработке подсистемы импорта данных являлось определение источников информации, которые будут использоваться для анализа. В контексте промышленного оборудования это были как внутренние источники, такие как датчики, установленные на самом оборудовании, так и внешние источники, например, данные из систем управления производственными процессами (SCADA) [142-144], системы управления предприятием (ERP) [145-148]. Было важно учесть, что данные могут поступать в различных форматах и протоколах, включая структурированные данные (в формате CSV или JSON) и неструктурированные данные (текстовые лог-файлы).

В качестве основного языка программирования для реализации данной подсистемы была выбрана ASP.Net (платформа разработки программного обеспечения, созданная компанией Microsoft, которая предоставляет набор инструментов и библиотек для создания различных типов приложений, включая веб-приложения, настольные программы и мобильные приложения, поддерживает несколько языков программирования, таких как C#, F# и Visual Basic, и обеспечивает кроссплатформенность), т.к. с ее помощью достаточно просто реализуются [149-150]:

1) **Мониторинг и логирование.**

Мониторинг работы подсистемы и логирование событий являются важными аспектами, позволяющими отслеживать состояние системы и выявлять возможные проблемы. Для этого можно использовать встроенные средства .NET, такие как ILogger, или интегрировать сторонние решения, такие как Serilog или NLog, для более гибкого управления логами.

2) **Обеспечение безопасности.**

Безопасность данных также является критически важным аспектом. Реализация аутентификации и авторизации пользователей может быть выполнена с использованием ASP.NET Core Identity (система управления пользователями, встроенная в ASP.NET, которая обеспечивает функциональность аутентификации и авторизации для веб-приложений). Кроме того, следует обеспечить защиту данных на этапе передачи, используя протоколы HTTPS и шифрование.

Соответственно, данные из подсистемы импорта напрямую поступают в БД подсистемы хранения и обработки данных, в которой хранятся как исходные данные, так и предобработанные.

Далее представлена часть кода алгоритма импорта с подробными комментариями.

```
using System;
using System.Threading;
using Confluent.Kafka;
using ClickHouse.Net;
using ClickHouse.Net.ADO;

class Program
{
    public static void Main(string[] args)
    {
        var kafkaConfig = new ConsumerConfig
        {
            BootstrapServers = "localhost:9092", // Адрес вашего Kafka сервера
            GroupId = "test-consumer-group", // Идентификатор группы потребителей
            AutoOffsetReset = AutoOffsetReset.Earliest // Начать с самого раннего сообщения
        };

        // Настройки подключения к ClickHouse
        var  clickHouseConnectionString  =  "Host=localhost;Port=8123;Database=your_database;User
=default;Password=your_password;";
        using (var consumer = new ConsumerBuilder<Ignore, string>(kafkaConfig).Build())
        {
            consumer.Subscribe("your-topic-name"); // Укажите имя вашей темы
            using             (var             clickHouseConnection             =             new
ClickHouseConnection(clickHouseConnectionString))
```

```
            {
                clickHouseConnection.Open();
                CancellationTokenSource cts = new CancellationTokenSource();
                Console.CancelKeyPress += (_, e) => {
                    e.Cancel = true; // Prevent the process from terminating.
                    cts.Cancel();
                };
                try
                {
                    while (true)
                    {
                        try
                        {
                            var cr = consumer.Consume(cts.Token);
                            var message = cr.Value;
                            // Здесь вы можете преобразовать сообщение в нужный формат

                            // Пример вставки данных в ClickHouse
 var query = "INSERT INTO your_table (column1, column2) VALUES (@value1, @value2)"; // Укажите
свою таблицу и колонки
                            using     (var     command     =     new     ClickHouseCommand(query,
clickHouseConnection))
                            {
                                // Добавьте параметры в команду
              command.Parameters.AddWithValue("value1", message); // Пример параметра
              command.Parameters.AddWithValue("value2",  DateTime.UtcNow);  //  Пример  параметра
времени

                                command.ExecuteNonQuery();
                            }

              Console.WriteLine($"Consumed           message           '{message}'           at:
'{cr.TopicPartitionOffset}'.");
                        }
                        catch (ConsumeException e)
                        {
                            Console.WriteLine($"Error occurred: {e.Error.Reason}");
                        }
                    }
                }
                catch (OperationCanceledException)
                {
                    //  Ensure  the  consumer  leaves  the  group  cleanly  and  final  offsets  are
committed.
                    consumer.Close();
                }
            }
        }
    }
}
```

3.2.2. Разработка подсистемы хранения и обработки данных

Ключевым аспектом работы подсистемы хранения данных является интеграция с другими системами. Это достигается с помощью различных интерфейсов и API (Application programming interface - «интерфейс программирования приложения»), которые обеспечивают обмен данными между системами мониторинга, управления и аналитики. Такой подход позволяет в реальном времени получать актуальную информацию о состоянии оборудования и оперативно реагировать на возможные отклонения от нормы.

При разработке было принято решение, что в рамках подсистемы хранения и обработки данных будут использоваться две независимые базы данных, что обусловлено спецификой хранящейся в них информации.

В подсистеме хранения данных могут использоваться реляционные и нереляционные базы данных, которые позволяют эффективно организовать и структурировать информацию. Реляционные базы данных, такие как PostgreSQL [151] или MySQL [152], позволяют хранить данные в таблицах и обеспечивают возможность выполнения сложных запросов для извлечения необходимой информации. В то же время, нереляционные базы данных, такие как MongoDB [153] или Apache Cassandra [154], могут быть использованы для хранения неструктурированных данных, что особенно важно в условиях, когда объем информации велик и разнообразен.

В качестве базы данных для хранения исходных и предобработанных данных (подсистема сбора и хранения данных) была выбрана ClickHouse [155], интерфейс взаимодействия с которой представлен на рисунке 3.7.

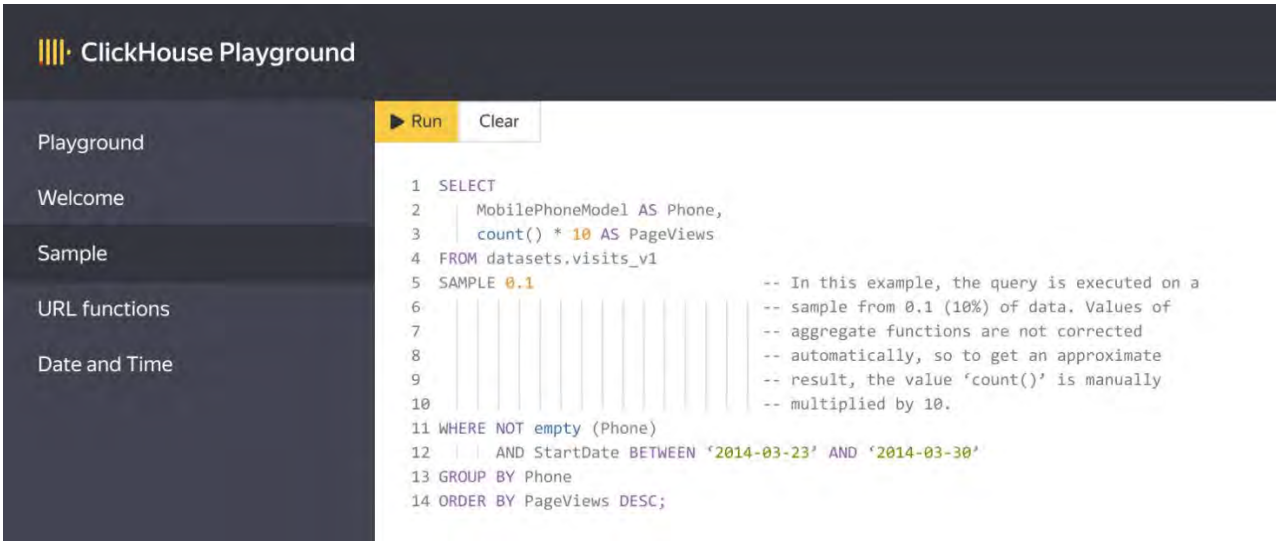


Рисунок 3.7 – Интерфейс работы с ClickHouse

ClickHouse является колоночной системой управления базами данных (СУБД), что обеспечивает её высокую производительность при выполнении запросов, особенно в контексте аналитических операций. В отличие от традиционных строковых СУБД, где данные хранятся построчно, что приводит к неэффективному использованию ресурсов при обработке больших объемов

информации, колоночная архитектура ClickHouse оптимизирует процесс анализа данных, поступающих с различных сенсоров и систем мониторинга. Это достигается благодаря тому, что при выполнении запросов обрабатываются только необходимые столбцы, что снижает объем данных и ускоряет доступ к ним.

ClickHouse оптимизирован для работы с массивами данных, что особенно актуально для систем, собирающих информацию с множества устройств и сенсоров. Система способна обрабатывать терабайты и даже петабайты информации, что делает её идеальным выбором для предприятий с высокой степенью автоматизации и мониторинга. Эффективное использование параллельной обработки данных, при которой нагрузка распределяется между несколькими узлами, значительно увеличивает скорость выполнения запросов.

Еще одной ключевой особенностью ClickHouse является его способность к быстрой агрегации и анализу данных в реальном времени, что критически важно для прогнозирования. В этой области необходимо не только хранить данные, но и оперативно их обрабатывать для получения актуальных предсказаний. ClickHouse поддерживает различные функции агрегации и оконные функции, что позволяет аналитикам и разработчикам легко извлекать необходимую информацию и формировать сложные аналитические запросы.

Помимо высокой производительности, ClickHouse предоставляет мощные возможности для масштабирования. Система может быть легко расширена за счет добавления новых узлов, что позволяет адаптироваться к растущим объемам данных без значительных затрат на переоснащение. Это особенно важно для компаний, которые планируют расширение своих операций и увеличение объема собираемых данных. ClickHouse также поддерживает SQL-подобный язык запросов, что делает его доступным для широкого круга специалистов, знакомых с традиционными реляционными СУБД. Это упрощает интеграцию ClickHouse в существующие системы и позволяет аналитикам быстро начать работу с данными, минимизируя время, необходимое для освоения новых инструментов.

Кроме того, ClickHouse предлагает возможности интеграции с другими инструментами и системами, такими как Apache Kafka для потоковой обработки данных и различные BI-инструменты для визуализации и анализа. Это позволяет создавать комплексные системы, способные автоматически собирать, обрабатывать и анализировать данные, что значительно ускоряет процесс принятия решений на основе предсказаний.

Также ClickHouse постоянно обновляется, что обеспечивает доступ к новым функциям и улучшениям. Это важно для поддержания системы в актуальном состоянии и использования последних достижений в области обработки данных и аналитики.

Соответственно, все данные из подсистемы импорта данных поступают именно в ClickHouse.

В качестве же БД для хранения данных о работе моделей было принято решение использовать PostgreSQL [156]. PostgreSQL — это объектно-реляционная система управления базами данных (СУБД), которая известна своей надежностью, стабильностью и богатым набором функций. Она была разработана в начале 1980-х годов в Университете Калифорнии в Беркли и с тех пор стала одной из самых популярных СУБД в мире. Интерфейс взаимодействия с данной СУБД представлен на рисунке 3.8.

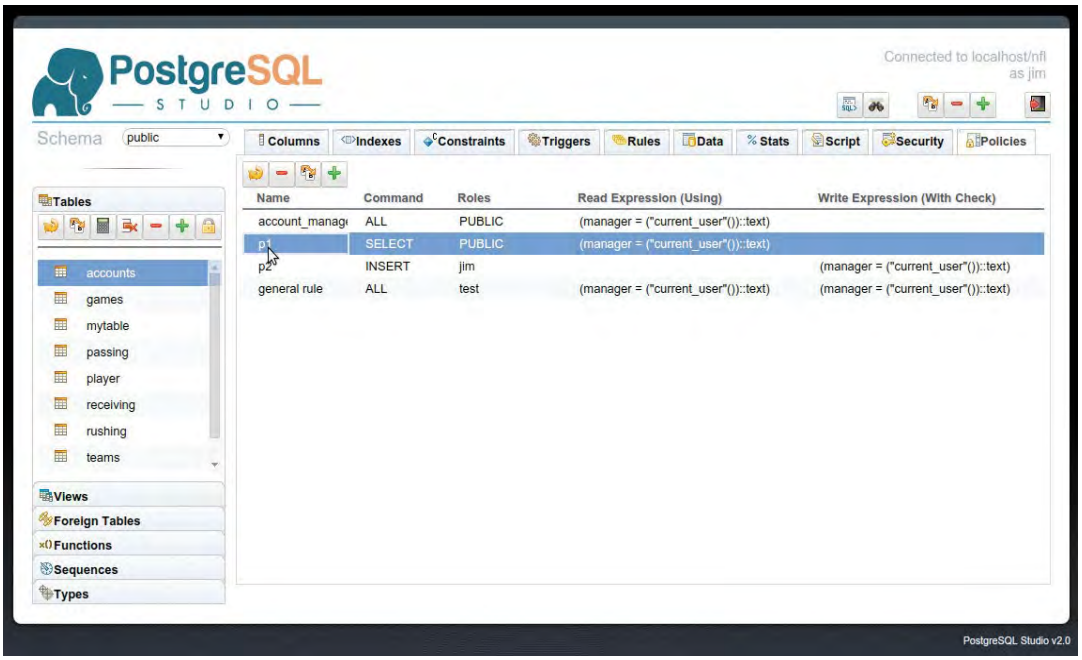


Рисунок 3.8 – Интерфейс работы с PostgreSQL

Одной из ключевых особенностей PostgreSQL является её расширяемость. Пользователи могут добавлять собственные функции, типы данных и операторы. PostgreSQL поддерживает множество типов данных, включая стандартные числовые и строковые типы, а также более сложные структуры, такие как JSON и XML, что позволяет эффективно работать с неструктурированными данными. Это особенно важно в современном мире, где объемы и разнообразие данных постоянно растут. Система управления базами данных также предлагает мощные инструменты для работы с транзакциями, включая поддержку ACID (атомарность, согласованность, изолированность, долговечность), что гарантирует надежность и целостность данных. PostgreSQL поддерживает многоверсионность (MVCC), что позволяет одновременно обрабатывать множество запросов без блокировок, улучшая производительность в многопользовательских средах. PostgreSQL обладает мощными возможностями для работы с запросами. Она поддерживает сложные SQL-запросы, включая подзапросы, объединения, оконные функции и агрегацию.

Следовательно, PostgreSQL значительно больше подходит для реализации БД, в которой будут храниться результаты работы моделей, с которой разработчик и пользователь будет активно взаимодействовать, в отличии от ClickHouse, которая значительно лучше подходит для хранения и обработки терабайтов данных в режиме реального времени, но не имеет столь широкого пользовательского функционала.

Также в рамках подсистемы хранения и обработки данных был реализован программный модель предобработки данных. Модуль реализован на Python.

3.2.3. Разработка подсистемы управления моделями

Одной из причин реализации данной подсистемы была необходимость разработать механизмы мониторинга работы моделей в реальном времени. Это позволит отслеживать длительность обучения и адаптировать их к изменениям в данных.

Также важно было внедрить систему оповещения, которая будет информировать пользователей о снижении качества предсказаний, что позволит оперативно реагировать на возможные проблемы. Для этого можно использовать специализированные решения для мониторинга моделей, такие как MLflow [157] (рисунок 3.9) или Prometheus [158].

MLflow - платформа для управления жизненным циклом моделей машинного обучения. Одной из ключевых причин, по которой MLflow наиболее подходящее решение для реализации подсистемы управления моделями, является ее способность упрощать и автоматизировать процессы, связанные с разработкой, развертыванием и отслеживанием моделей машинного обучения.

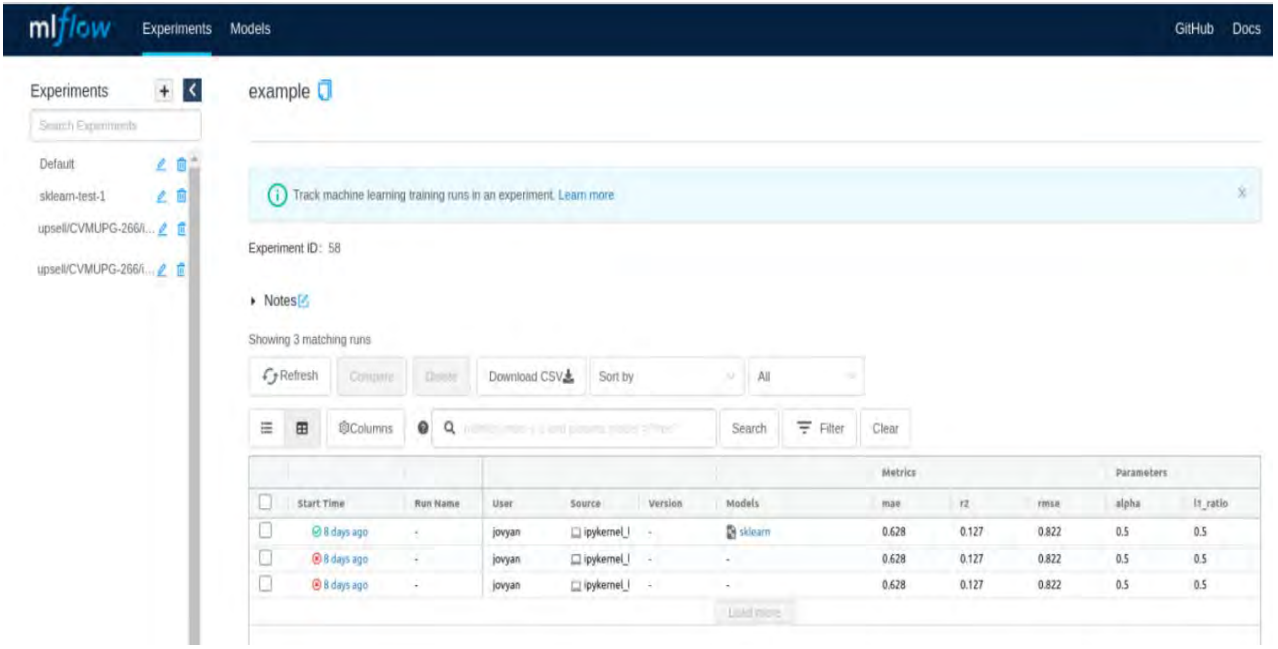


Рисунок 3.9 – Интерфейс работы с MLflow

MLflow представляет собой модульную архитектуру, состоящую из четырех ключевых компонентов: Tracking, Projects, Models и Registry. Данная структура предоставляет пользователям гибкость в выборе и комбинировании функциональных возможностей в зависимости от специфических требований проекта. Компонент Tracking позволяет осуществлять мониторинг экспериментов, фиксируя параметры, метрики и артефакты, что является критически важным для прогнозирования. В контексте мониторинга технического состояния оборудования, где данные поступают из различных

источников и обладают разными характеристиками, возможность детального отслеживания изменений в моделях и их производительности позволяет аналитикам и инженерам быстро определять, какие параметры оказывают влияние на точность предсказаний. Кроме того, MLflow обеспечивает легкость интеграции с различными библиотеками и инструментами машинного обучения, такими как TensorFlow, PyTorch и Scikit-learn. Это позволяет командам, занимающимся прогнозированием, использовать уже знакомые инструменты и библиотеки, что минимизирует время, необходимое для обучения и интеграции. Например, если уже применяется Scikit-learn для разработки моделей, она может без труда интегрировать MLflow для отслеживания экспериментов и управления версиями моделей.

Еще одним важным аспектом является управление версиями моделей. В системах прогнозирования, где модели могут часто обновляться в зависимости от новых данных и требований, управление версиями становится критически важным. MLflow Models предоставляет возможность сохранять, загружать и развертывать модели, а также контролировать их версии.

Это обеспечивает уверенность в использовании актуальной и наиболее эффективной версии модели, что особенно важно для надежности предсказаний состояния оборудования. MLflow также предлагает возможности автоматизации процессов через компонент MLflow Projects. Этот элемент позволяет упаковывать код и зависимости в стандартизированные форматы, что упрощает воспроизводимость экспериментов и развертывание моделей. В контексте прогнозирования, где требуется быстрая реакция на изменения в данных и условиях эксплуатации оборудования, возможность оперативного развертывания и тестирования новых моделей значительно повышает эффективность аналитических процессов (рисунок 3.10). Не менее важным является компонент MLflow Registry, который предоставляет централизованное хранилище для моделей. Это позволяет разработчикам эффективно управлять жизненным циклом моделей, отслеживать их состояние и проводить анализ. В системах прогнозирования, где необходим высокий уровень контроля и

надежности, наличие единого реестра моделей помогает избежать путаницы и ошибок, связанных с использованием устаревших или некорректных версий.

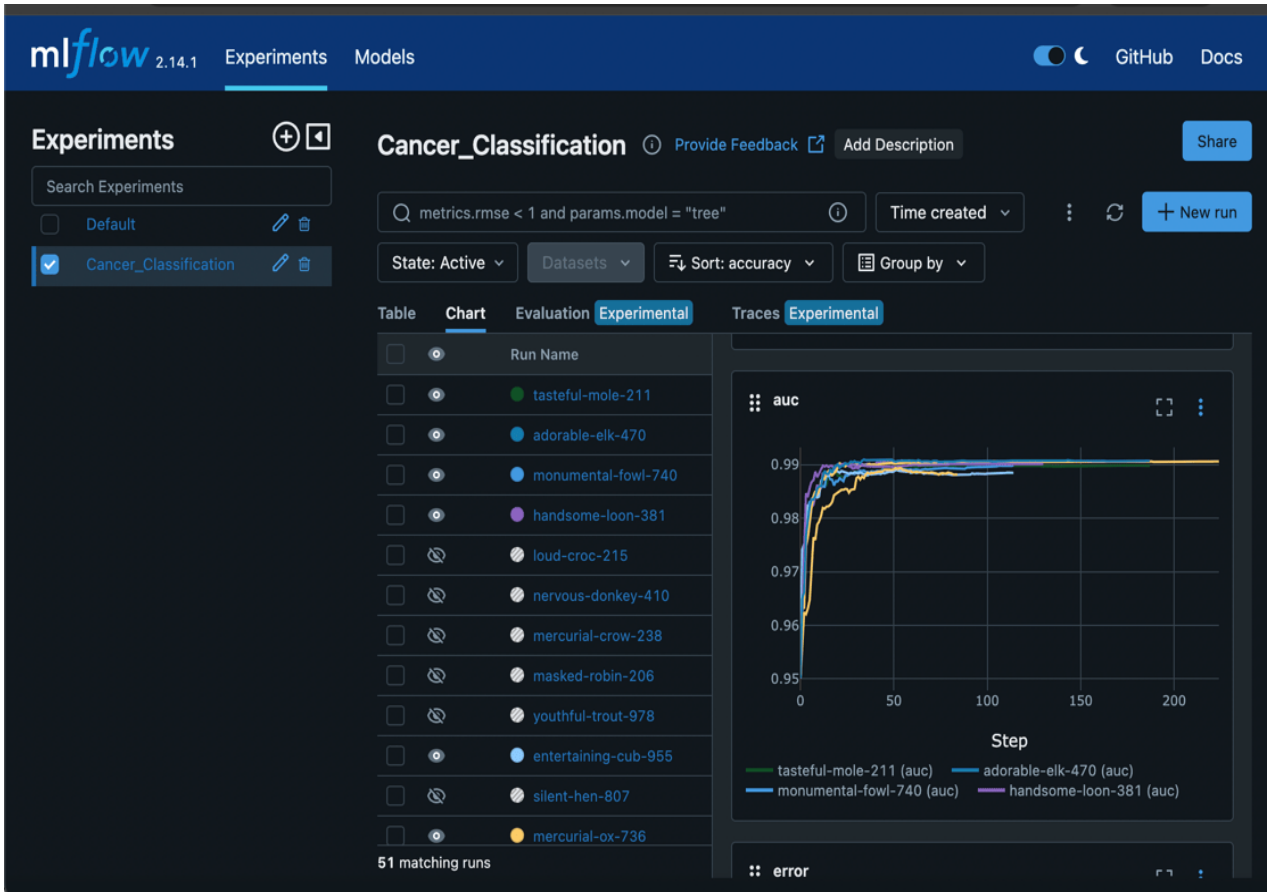


Рисунок 3.10 – Мониторинг процесса обучения в MLflow

MLflow также поддерживает интеграцию с различными платформами для развертывания моделей, такими как Docker и Kubernetes, что облегчает масштабирование решений и их внедрение в производственные среды. Это особенно актуально для прогнозирования, где модели могут требовать значительных вычислительных ресурсов для обработки больших объемов данных в реальном времени.

Безопасность и управление доступом также являются важными аспектами разработки подсистемы управления моделями. Было необходимо реализовать механизмы аутентификации и авторизации пользователей, чтобы ограничить доступ к моделям и данным. Это было реализовано с использованием JWT (JSON Web Tokens) — это открытый стандарт, который определяет компактный и автономный способ передачи информации между сторонами в виде JSON-объекта.

MLflow напрямую взаимодействует с подсистемой хранения данных, записывая все результаты вывода моделей в БД PostgreSQL.

В рамках разработки реализована возможность взаимодействия с MLflow как через отдельный интерфейс (актуально для разработчиков), так и через интерфейс автоматизированной системы (актуально для пользователей), что будет показано далее.

3.3. Организация гибкой архитектуры системы

Для организации гибкой архитектуры системы было принято решение использовать Docker [159] и Kubernetes [160]. Они представляют собой высокоэффективные инструменты для контейнеризации (это технология, которая помогает запускать приложения изолированно от основной операционной системы) и оркестрации (автоматическое размещение, координация и управление сложными компьютерными системами и службами), которые значительно упрощают процессы развертывания, масштабирования и управления приложениями, включая системы прогнозирования технического состояния оборудования. Их применение в данной области обусловлено множеством факторов, делающих их оптимальными для обработки больших объемов данных, а также для обеспечения высокой доступности и надежности аналитических решений.

Docker предоставляет возможность упаковывать приложения вместе со всеми их зависимостями в контейнеры, что обеспечивает целостность среды выполнения. Контейнеры функционируют одинаково на любом сервере, независимо от его конфигурации. В контексте прогнозирования, где используются различные библиотеки и инструменты для обработки данных и создания моделей машинного обучения, создание стандартизированных контейнеров позволяет избежать проблем с несовместимостью окружений. Это особенно актуально, когда команды работают с различными версиями библиотек или при переносе приложений между тестовыми и производственными средами.

Контейнеризация с помощью Docker также упрощает процесс

развертывания приложений. Вместо необходимости вручную настраивать серверы и устанавливать все необходимые зависимости, запускается контейнер, который уже содержит все необходимое для работы приложения. Это существенно сокращает время, необходимое для внедрения новых моделей или обновлений в системе прогнозирования, что позволяет быстрее реагировать на изменения в данных и требованиях бизнеса.

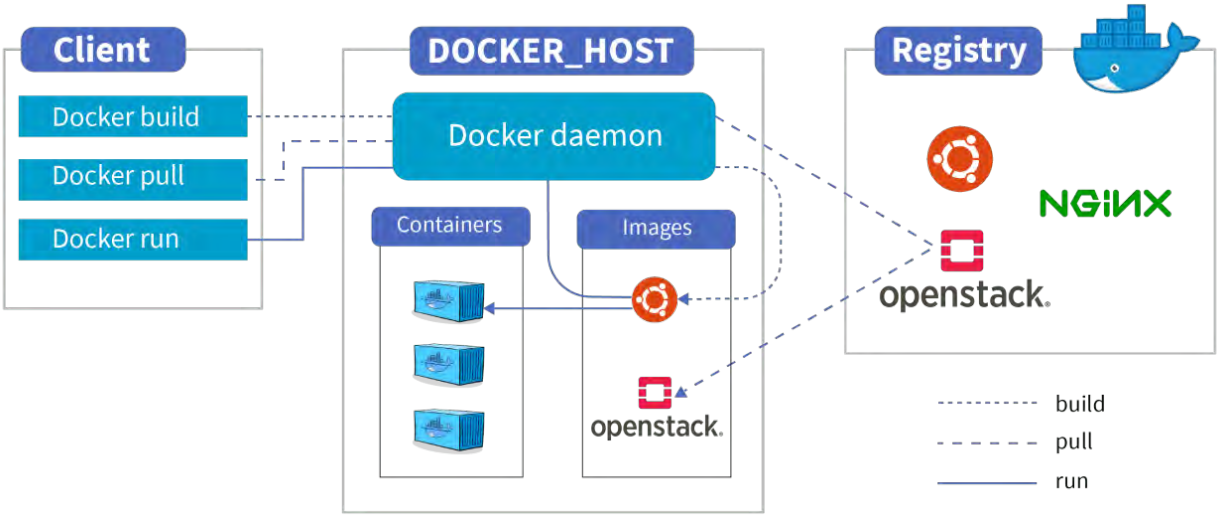


Рисунок 3.11 – Архитектура Docker

Архитектура Docker (рисунок 3.11) основана на концепции контейнеризации, которая позволяет изолировать приложения и их зависимости в переносимые контейнеры, упрощая развертывание, масштабирование и управление приложениями в различных средах. В центре этой архитектуры находится Docker Engine, состоящий из нескольких ключевых компонентов. Во-первых, это сервер (Daemon), который работает в фоновом режиме и управляет контейнерами и образами, обрабатывая команды от клиента Docker и взаимодействуя с операционной системой для управления ресурсами. Во-вторых, Docker предоставляет REST API (REpresentational State Transfer — «передача репрезентативного состояния»), позволяющий взаимодействовать с Docker Daemon, что упрощает автоматизацию операций с контейнерами и образами. Наконец, клиент (CLI) представляет собой командную строку, через которую пользователи могут вводить команды для управления Docker.

Образы Docker служат шаблонами для создания контейнеров и содержат

все необходимые файлы, библиотеки и зависимости для запуска приложений. Эти образы являются неизменяемыми, и при изменении приложения создается новый образ. Docker использует систему слоев, что позволяет эффективно хранить и передавать образы, так как общие слои могут быть использованы неоднократно.

Контейнеры, в свою очередь, представляют собой запущенные экземпляры образов. Они изолированы друг от друга и от хост-системы, что обеспечивает безопасность и стабильность. Контейнеры можно запускать, останавливать, удалять и перемещать между различными средами без изменения кода приложения. Для хранения и обмена образами Docker предоставляет облачный сервис Docker Hub, где пользователи могут загружать свои образы или загружать образы, созданные другими. Также возможно создание частных реестров для хранения образов, что полезно для корпоративных приложений.

Docker включает в себя систему управления сетями, позволяющую контейнерам взаимодействовать друг с другом и с внешними ресурсами. Поддерживаются различные типы сетей, такие как мостовые сети (bridge), сети хоста (host) и пользовательские сети, что дает возможность гибко настраивать сетевую архитектуру приложения. Для управления многоконтейнерными приложениями существует инструмент Docker Compose, который использует файл конфигурации (обычно в формате YAML) для описания всех необходимых контейнеров, их зависимостей и параметров, упрощая развертывание сложных приложений.

Для управления кластером контейнеров Docker предоставляет возможность использовать Docker Swarm, который обеспечивает оркестрацию и управление несколькими экземплярами контейнеров. Хотя Kubernetes не является частью Docker, он также широко используется для оркестрации контейнеров и может работать с образами Docker.

Далее приводится код работы с уже установленным Docker в рамках операционной системы CentOS (дистрибутив операционной системы Linux,

основанный на исходном коде Red Hat Enterprise Linux):

```
#!/bin/bash
# Имя образа и контейнера
IMAGE_NAME="my-nginx-image"
CONTAINER_NAME="my-nginx-container"
# Установка Docker, если он еще не установлен
if ! command -v docker &> /dev/null; then
    echo "Docker не установлен. Устанавливаем Docker..."
    # Устанавливаем необходимые пакеты
    sudo yum install -y yum-utils device-mapper-persistent-data lvm2
    # Добавляем репозиторий Docker
    sudo  yum-config-manager  --add-repo  https://download.docker.com/linux/centos/docker-
ce.repo
    # Устанавливаем Docker
    sudo yum install -y docker-ce
    # Запускаем и активируем Docker
    sudo systemctl start docker
    sudo systemctl enable docker
    echo "Docker установлен и запущен."
else
    echo "Docker уже установлен."
fi
# Создаем Dockerfile
echo "Создаем Dockerfile..."
cat <<EOF > Dockerfile
# Используем базовый образ
FROM centos:7
# Устанавливаем Nginx
RUN yum install -y epel-release && \
    yum install -y nginx && \
    yum clean all
# Открываем порт 80
EXPOSE 80
# Запускаем Nginx
CMD ["nginx", "-g", "daemon off;"]
EOF
# Построение Docker-образа
echo "Строим Docker-образ..."
docker build -t $IMAGE_NAME .
# Запуск контейнера
echo "Запускаем контейнер..."
docker run -d --name $CONTAINER_NAME -p 80:80 $IMAGE_NAME
# Ожидание 5 секунд, чтобы контейнер успел запуститься
sleep 5
# Проверка состояния контейнера
```

```
echo "Проверяем состояние контейнера..."
docker ps
# Остановка контейнера
echo "Останавливаем контейнер..."
docker stop $CONTAINER_NAME
# Удаление контейнера
echo "Удаляем контейнер..."
docker rm $CONTAINER_NAME
# Удаление образа
echo "Удаляем образ..."
docker rmi $IMAGE_NAME
echo "Операции с Docker завершены."
```

Kubernetes, в свою очередь, предоставляет мощные инструменты для оркестрации контейнеров, позволяя управлять развертыванием и масштабированием приложений.

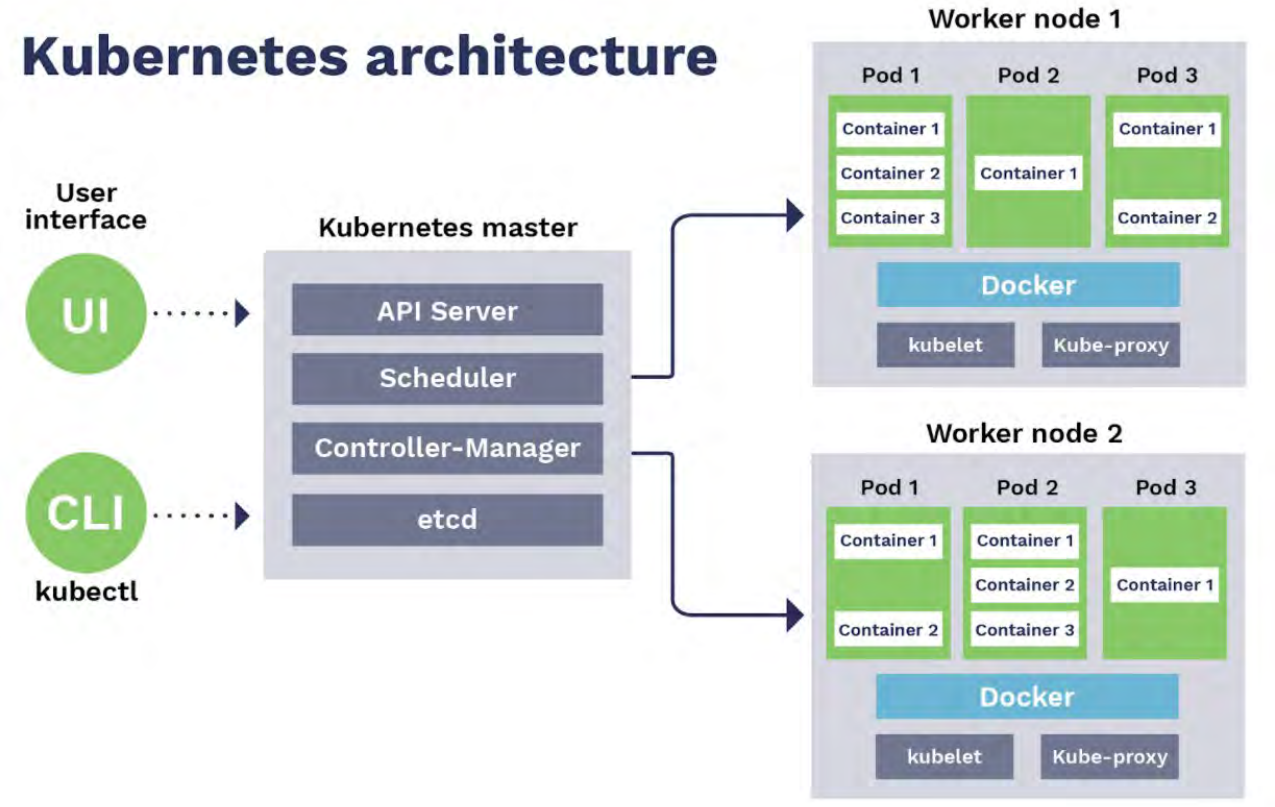


Рисунок 3.12 – Архитектура Kubernetes

В системах прогнозирования, где объем данных может варьироваться и требовать различных ресурсов в зависимости от нагрузки, Kubernetes позволяет динамически масштабировать количество экземпляров приложения в ответ на изменения в трафике или загрузке. Это означает, что в периоды высокой нагрузки система может автоматически добавлять новые контейнеры,

а в периоды низкой нагрузки — уменьшать их количество, что оптимизирует использование ресурсов и снижает затраты.

В центре архитектуры (рисунок 3.12) находится Kubernetes Master, который отвечает за управление кластером. Он включает в себя несколько ключевых компонентов. API Server является основным интерфейсом для взаимодействия с кластером, принимая команды и запросы от пользователей и других компонентов. Scheduler отвечает за распределение контейнеров по рабочим узлам на основе доступных ресурсов и требований приложений. Controller Manager управляет состоянием кластера, следя за тем, чтобы количество запущенных экземпляров приложений соответствовало заданным параметрам. etcd — это распределенное хранилище данных, где Kubernetes сохраняет всю конфигурационную информацию о кластере, обеспечивая высокую доступность и надежность.

Рабочие узлы, или Worker Nodes, представляют собой серверы, на которых запускаются контейнеризованные приложения. Каждый рабочий узел включает в себя несколько компонентов. Kubelet — это агент, который управляет состоянием контейнеров на узле, обеспечивая их запуск и мониторинг. Kube Proxy отвечает за сетевую маршрутизацию, обеспечивая доступ к сервисам в кластере и управление сетевыми правилами. Container Runtime — это программное обеспечение, которое отвечает за выполнение контейнеров, поддерживая такие технологии, как Docker, containerd и другие.

Kubernetes использует концепцию Pods, которые представляют собой минимальные развертываемые единицы, содержащие один или несколько контейнеров, которые работают вместе и разделяют ресурсы. Pods могут быть масштабированы и управляемы с помощью ReplicaSets, которые обеспечивают нужное количество экземпляров приложения. Для управления доступом к приложениям внутри кластера используются Services, которые обеспечивают стабильные сетевые адреса и балансировку нагрузки.

Kubernetes также поддерживает Namespaces, позволяющие разделять ресурсы и управление в пределах одного кластера, что особенно полезно для

многопользовательских сред. Для автоматизации развертывания и обновления приложений используются Deployments, которые управляют состоянием приложения и обеспечивают плавные обновления без простоя.

Установка и настройка Kubernetes осуществляется только после установки Docker. Код настройки приведен далее:

```
#!/bin/bash
# Устанавливаем Docker
sudo yum install -y yum-utils
sudo   yum-config-manager   --add-repo   https://download.docker.com/linux/centos/docker-ce.repo
sudo yum install -y docker-ce
sudo systemctl start docker
sudo systemctl enable docker
# Устанавливаем Kubernetes
cat <<EOF | sudo tee /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86_64
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
       https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
EOF
# Устанавливаем kubelet, kubeadm и kubectl
sudo yum install -y kubelet kubeadm kubectl
sudo systemctl enable --now kubelet
# Инициализация кластера Kubernetes
echo "Инициализируем кластер Kubernetes..."
sudo kubeadm init --pod-network-cidr=192.168.0.0/16
# Настройка kubectl для обычного пользователя
echo "Настраиваем kubectl для обычного пользователя..."
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
# Установка сетевого плагина (например, Calico)
echo "Устанавливаем сетевой плагин Calico..."
kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml
# Ожидание, пока все узлы не станут Ready
echo "Ожидание, пока все узлы не станут Ready..."
kubectl wait --for=condition=Ready nodes --all --timeout=300s
# Развертывание простого приложения (Nginx)
echo "Разворачиваем простое приложение (Nginx)..."
```

```
kubectl create deployment nginx --image=nginx
kubectl expose deployment nginx --port=80 --type=NodePort
# Проверка состояния развернутого приложения
echo "Проверка состояния развернутого приложения..."
kubectl get pods
kubectl get svc

# Очистка кластера
echo "Очистка кластера..."
kubectl delete deployment nginx
kubectl delete svc nginx
echo "Операции с Kubernetes завершены."
```

3.4 Программная реализация разработанных алгоритмов на Python

Программная реализация алгоритмов для автоматизированной системы прогнозирования технического состояния промышленного оборудования на Python включает в себя несколько ключевых этапов. Рассмотрим их более подробно на примере реализации модели прогнозирования отказов M1, включая примеры кода.

Первый шаг — загрузка данных (рассмотрена на примере CSV (Comma-Separated Values — «значения, разделённые запятыми») файла).

```
import argparse
DEFAULT_PATH = 'Data_ex_4.csv'
arg_parser = argparse.ArgumentParser()
arg_parser.add_argument('--csv_path', type=str)
args = arg_parser.parse_args()
csv_path = args.csv_path if args.csv_path else DEFAULT_PATH
```

На втором этапе настраивается окружение для работы с библиотеками, необходимыми для машинного обучения, устанавливает генератор случайных чисел для воспроизводимости и определяет, будет ли использоваться GPU (graphics processing unit – «графический процессор») или CPU (central processing unit – «центральный процессор») для вычислений.

```
import math
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import torch.nn as nn
from torch.autograd import Variable
import random
```

```
import torch
from tqdm import tqdm
def set_random_seed(seed):
    torch.manual_seed(seed)
    torch.cuda.manual_seed(seed)
    np.random.seed(seed)
    random.seed(seed)
    torch.backends.cudnn.deterministic = True
set_random_seed(42)
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
print("device is:",device)
```

Далее происходит преобразование временных данных из более частых интервалов в более редкие (средние значения за 5 минут) и производится предсказание классов на основе входных данных.

```
# converting 10-second intervals to 5-minute
# data - csv file with sensor readings of one exhauster
# step=30 - 5-minute interval contains 30 10-second intervals
def min_x_slicer(data, step):
    df_5= pd.DataFrame(columns = data.columns)
    size = data.shape[0]
    for i in tqdm(range(0, size, step)):
        if size-i > step:
            df_5.loc[len(df_5.index)] = data.iloc[i:i+step].mean().values
        else:
            df_5.loc[len(df_5.index)] = data.iloc[i:].mean().values
    return df_5
# function for predict class label for multiclass classification
def get_predictions(model, data):
    y_pred =[]

    with torch.no_grad():
        for i in tqdm(range(data.shape[0])):
            sample = torch.from_numpy(data)
            outputs = model(sample)
            predicted = torch.mean(outputs.data, axis=1)
            y = predicted[np.argmax(predicted)]
            y_pred.append(y)
    return y_pred
```

Далее представлена архитектура LSTM модели для решения задачи классификации с использованием последовательных данных. Модель включает в себя LSTM-слой для обработки временных зависимостей, полносвязные слои для преобразования выходов и функции активации для нелинейности.

```
# basic architecture of predictive model
class LSTMNet(nn.Module):
    def __init__(self, input_size, hidden_size, output_size, batch_sizes):
        super(LSTMNet, self).__init__()
        self.input_size = input_size
        self.hidden_size = hidden_size
        self.output_size = output_size
        self.batch_size = batch_sizes
        self.lstm  = nn.LSTM(self.input_size, self.hidden_size, batch_first=True)
        self.act1 = nn.ReLU()
        self.fc1 = nn.Linear(self.hidden_size, out_features=10)
        self.drop = nn.Dropout(0.6)
        self.fc2 = nn.Linear(in_features=10, out_features=3)
        self.act2 = nn.Sigmoid()



    def forward(self, x):
        x = x.float()
        x =torch.reshape(x, (x.shape[0], x.shape[1],  x.shape[2]))
        out, ht1 = self.lstm(x)
        out = self.act1(out)
        x = out[:, -1]
        x = self.act1(self.drop(self.fc1(x)))
        x = self.act2(self.fc2(x))
        return x
# input_size - number of features, output_size - shape of target labels
model = LSTMNet(input_size = 16, output_size = 3,  hidden_size=8, batch_sizes=1)
```

На последнем шаге модель выводит результаты предсказания:

```
columns = ['Тех. место', 'Вероятность отказа через месяц']
df_m1 = pd.DataFrame(columns = columns)
df_m1['Тех. место'] = pred_df.columns
df_m1['Вероятность отказа через месяц'] = pred_df_stat.values
df_m1.to_csv(f'M1_predict_ex{id_ex}.csv')
```

Пример вывода модели представлен на рисунке 3.13.

Таким образом, программная реализация моделей системы прогнозирования на Python включает в себя загрузку данных и обучение моделей. Каждый из этапов требует тщательной проработки и тестирования, чтобы обеспечить высокую точность и надежность предсказаний.

```
M1_predict_ex4.csv

df_m1
```

	Тех. место	Вероятность поломки через месяц
0	Y_ЭКСГАУСТЕР А/М №4_ЭЛЕКТРОДВИГАТЕЛЬ ДСПУ-140-...	0.426172
1	Y_ЭКСГАУСТЕР А/М №4_ПОДШИПНИК ОПОРНЫЙ ЭКСГ. №4	0.000000
2	Y_ЭКСГАУСТЕР А/М №4_УЛИТА ЭКСГ. №4	0.000000
3	Y_ЭКСГАУСТЕР А/М №4_РЕДУКТОР ГАЗ. ЗАДВИЖКИ ЭКС...	0.000000
4	Y_ЭКСГАУСТЕР А/М №4_МАСЛОНАСОС РАБОЧИЙ ЭКСГ. №4	0.000000
5	Y_ЭКСГАУСТЕР А/М №4_ЭЛЕКТРОАППАРАТУРА ЭКСГ. №4	0.405713
6	Y_ЭКСГАУСТЕР А/М №4_ПОДШИПНИК ОПОРНЫЙ №2 ЭКСГ. №4	0.000000
7	Y_ЭКСГАУСТЕР А/М №4_ЗАДВИЖКА ЭКСГ. №4	0.391424
8	Y_ЭКСГАУСТЕР А/М №4_ГСМ ЭКСГ. №4	0.000000
9	Y_ЭКСГАУСТЕР А/М №4_ВК 310С ВИБРОПРЕОБРАЗОВАТЕ...	0.000000
10	Y_ЭКСГАУСТЕР А/М №4_ТСМТ-101-010-50М-400 ТЕРМО...	0.000000
11	Y_ЭКСГАУСТЕР А/М №4_РОТОР ЭКСГ. №4	0.345080

Рисунок 3.13 – Пример вывода модели прогнозирования

Все библиотеки Python, использованные при разработке моделей, описаны в таблице 3.1.

Таблица 3.1. Библиотеки Python, использованные при разработке.

Название и версия библиотеки	**Функции**	**Назначение**
scikit-learn 1.2.2	sklearn.metrics.f1_score	Функция для расчета F1-метрики
	sklearn.linear_model.LogisticRegression()	Класс для модели линейной регрессии
	sklearn.svm .SVC()	Класс для модели метода опорных векторов
	sklearn.ensemble.IsolationForest()	Класс для модели изолирующего леса
Tensorflow 2.10.1	tf.data.Dataset()	Класс для формирования набора данных
	tf.keras.models.Sequential()	Класс для создания последовательности слоев нейронной сети
	tf.keras.layers.LSTM()	Класс для LSTM слоя

	tf.keras.layers.Dense()	Класс для полносвязного линейного слоя
	tf.keras.optimizers.RMSprop()	Класс для оптимизатора RMSprop
	tf.keras.metrics.RootMeanSquaredError	Функция для расчета RMSE
	tf.keras.callbacks.LearningRateScheduler	Функция для определения условий ранней остановки обучения нейронной сети
Torch 2.0.1 +cu118	torch.utils.data.Dataset()	Класс для формирования набора данных
	torch.utils.data.DataLoader()	Класс для создания правил итерации по данным
	torch.nn.Conv1d()	Класс для одномерного сверточного слоя
	torch.nn.MaxPool1d()	Класс для операции уменьшения размерности слоя
	torch.nn.BatchNorm1d()	Класс для одномерной пакетной нормализации
	torch.nn.Flatten()	Класс для конкатенации слоя в вектор
	torch.nn.Linear()	Класс для полносвязного линейного слоя
	torch.nn.Dropout()	Класс для операции прореживания
	torchmetrics.classification.BinaryJaccardIndex	Функция для расчета индекса Жаккарда
	torch.optim.AdamW()	Класс для оптимизатора Adam с L2-регуляризацией

Благодаря функционалу MLflow в стандартном режиме модели автоматически дообучаются раз в 4 месяца, т.к. за это время накапливается необходимый объем данных. Но в случае, если ложноположительных и ложноотрицательных предсказаний модели становится больше 5% от общего

числа, накопленный с последнего дообучения массив данных добавляется к уже имеющимся датасетам и на этой выборке производится внештатное дообучение. Также эти операции можно запустить вручную через интерфейс.

3.5. Программная реализация подсистемы отображения данных

Пользовательские функции и интерфейс, представляющие собой подсистему отображения данных, были разработаны с использованием «React - библиотеки JavaScript с открытым исходным кодом, предназначенной для создания пользовательских интерфейсов. Эта библиотека отличается компонентной архитектурой и высокой гибкостью, что делает её одним из наиболее востребованных инструментов для разработки интерфейсов» [161]. Применение React в системах прогнозирования технического состояния оборудования может значительно улучшить качество взаимодействия пользователей с данными и аналитическими инструментами. Одной из основных причин, по которой React является предпочтительным выбором, является его способность создавать динамичные интерфейсы, что особенно важно для аналитических приложений, где пользователи нуждаются в быстром доступе к информации и возможности визуализировать данные в реальном времени.

React использует компонентный подход, что позволяет разбивать интерфейс на независимые и переиспользуемые компоненты. Это не только упрощает процесс разработки и тестирования, но и позволяет командам создавать сложные интерфейсы, состоящие из множества элементов, которые можно легко изменять и обновлять. В области прогнозирования, где интерфейсы часто требуют отображения различных типов данных, таких как графики, таблицы и диаграммы, возможность создания модульных компонентов способствует эффективному управлению изменениями и добавлению новых функций без необходимости переработки всего кода. Также, высокая производительность React обеспечивается за счет использования виртуального DOM (Document Object Model – «объектная модель документа»), который минимизирует количество операций, необходимых для обновления

пользовательского интерфейса. Это особенно актуально для систем прогнозирования, где данные могут поступать в реальном времени и требуют мгновенного обновления визуализаций. Виртуальный DOM позволяет React эффективно отслеживать изменения в состоянии приложения и обновлять только те компоненты, которые действительно изменились, что обеспечивает быструю реакцию интерфейса на действия пользователя и изменения данных.

React легко интегрируется с различными библиотеками для визуализации данных, такими как D3.js, Chart.js и Recharts [162-165]. Это позволяет разработчикам создавать интерактивные и информативные графики и диаграммы, что помогает пользователям лучше интерпретировать данные и принимать обоснованные решения. В системах прогнозирования, где визуализация данных играет ключевую роль в интерпретации результатов моделей и принятии решений, возможность интеграции различных инструментов для визуализации становится особенно важной.

Еще одним значительным преимуществом React является его поддержка однонаправленного потока данных, что упрощает управление состоянием приложения. Это позволяет разработчикам четко отслеживать, как данные передаются между компонентами, и минимизировать вероятность ошибок, связанных с изменением состояния. В контексте прогнозирования, где данные могут изменяться в зависимости от различных факторов и условий, четкое управление состоянием помогает избежать путаницы и обеспечивает более предсказуемое поведение интерфейса. Помимо этого, React поддерживает различные подходы к управлению состоянием, такие как Redux (библиотека для JavaScript с открытым исходным кодом, предназначенная для управления состоянием приложения) [166] и MobX (библиотека, позволяющая разработчикам использовать инструмент для глобального использования переменных и методов между разными компонентами) [167-168], что дает разработчикам возможность выбирать наиболее подходящий инструмент в зависимости от сложности приложения и требований к производительности. Эти библиотеки помогают централизовать управление состоянием, что

особенно полезно в сложных системах прогнозирования, где взаимодействие между компонентами может быть достаточно запутанным.

Далее будут подробно рассмотрены разработанные на React интерфейсы системы (рисунок 3.14 - 3.20).

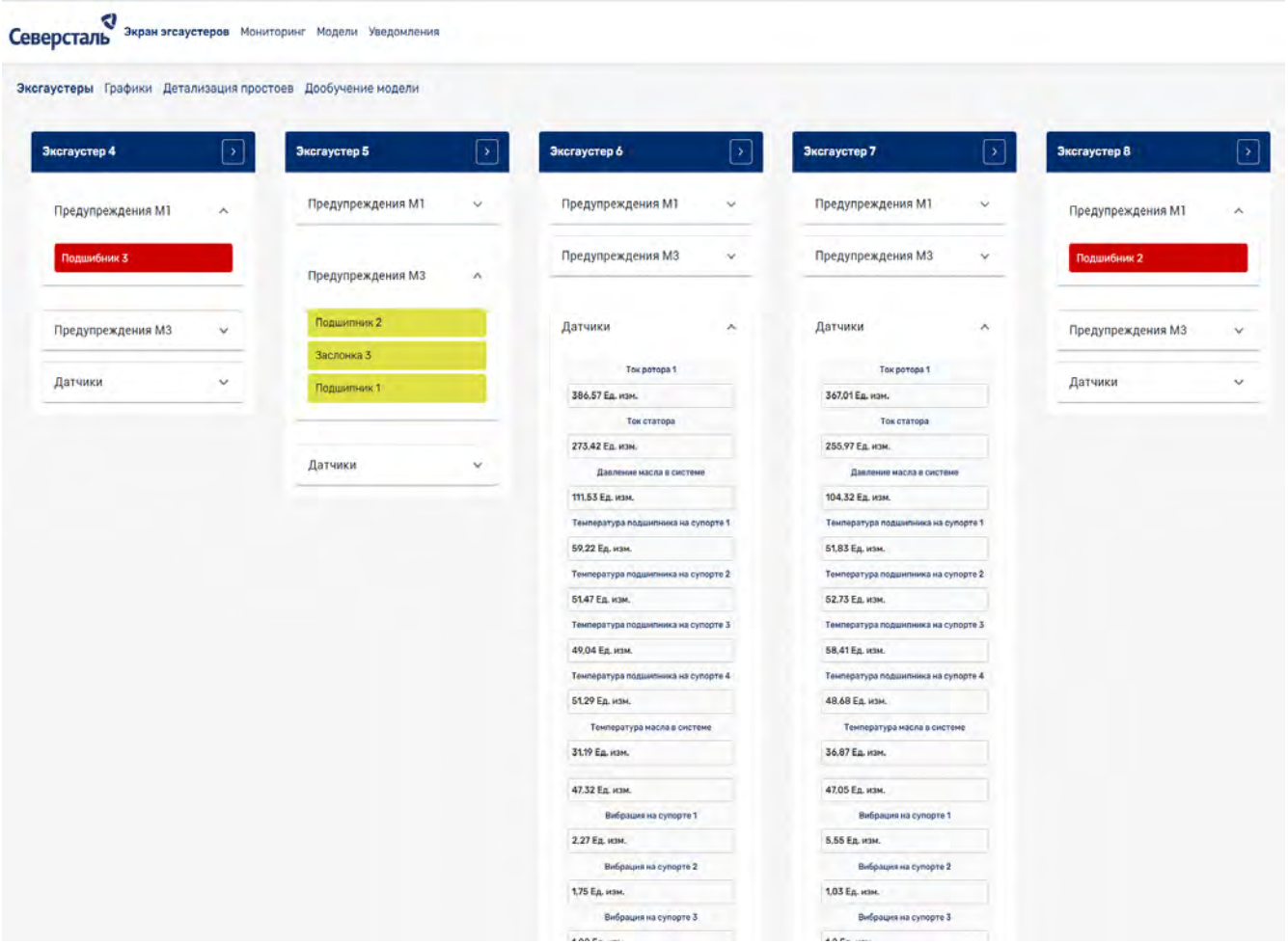


Рисунок 3.14 – Общий вид интерфейса

Интерфейс системы, реализованный в соответствии с требованиями экспертов Северстали, состоит из четырех основных элементов (рисунок 3.14):

«Эксгаустеры» – здесь в режиме реального времени можно отслеживать предупреждения о спрогнозированных отказах типа М1 (внутреннее обозначение Северстали) и неисправностях типа М3, а также показания датчиков. Красным отмечаются технические места, для которых предсказаны отказы, а желтым – неисправности.

«Графики» – для более подробного анализа можно вывести исторические данные по выбранным датчикам на гибко настраиваемый график.

«Детализация простоев» – вывод общей аналитики по простоям эксгаустеров и полной истории уведомлений о спрогнозированных неисправностях и отказах.

«Дообучение модели» – выбор файлов и кнопка для запуска дообучения модели, а также вывод таблицы с метриками качества текущих моделей.

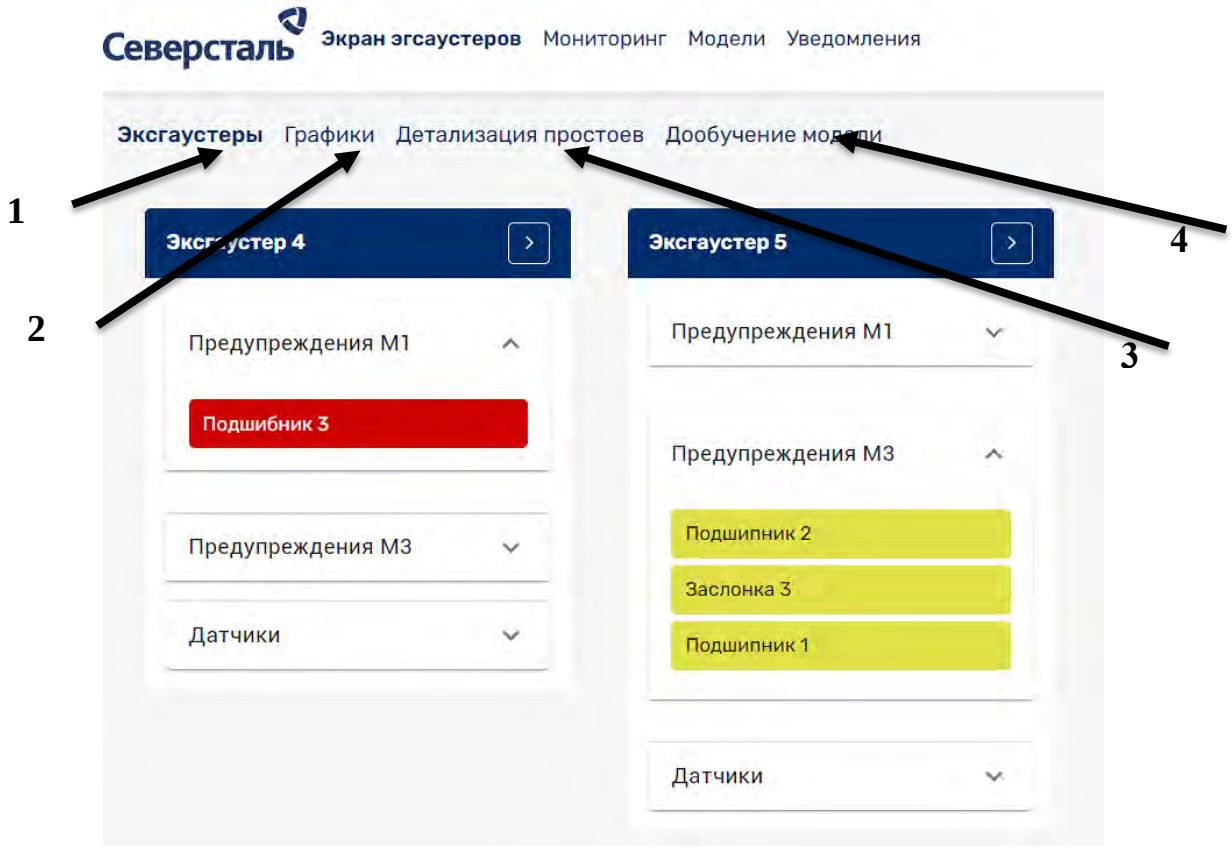


Рисунок 3.15 – Укрупненный вид интерфейса. 1 – интерфейс мониторинга спрогнозированных неисправностей и простоев, 2 – интерфейс отображения исторических данных на графиках, 3 – интерфейс вывода статистической информации о неисправностях и простоях, 4 – интерфейс дообучения модели.

Данные в систему поступают с шести эксгаустеров, в информационных ресурсах Северстали пронумерованных от 4 до 9.

На рисунке 3.16. пример вывода графика с показаниями датчика за выбранный период времени. На одном графике можно отобразить данные с неограниченного числа датчиков, относящихся к различным технические местам и эксгаустерам.

Рисунок 3.16 – Отображение графиков

На рисунке 3.17 отображается детальная информация по простоям каждого эксгаустера за выбранный промежуток времени (рассчитанная как сумма простоев на каждом техническом месте конкретного эксгаустера), суммарная продолжительность отказов (M1) по техническим местам, время простоя каждого из типов технических мест по всем эксгаустерам, а также история уведомлений о предполагаемых отказах и простоях.

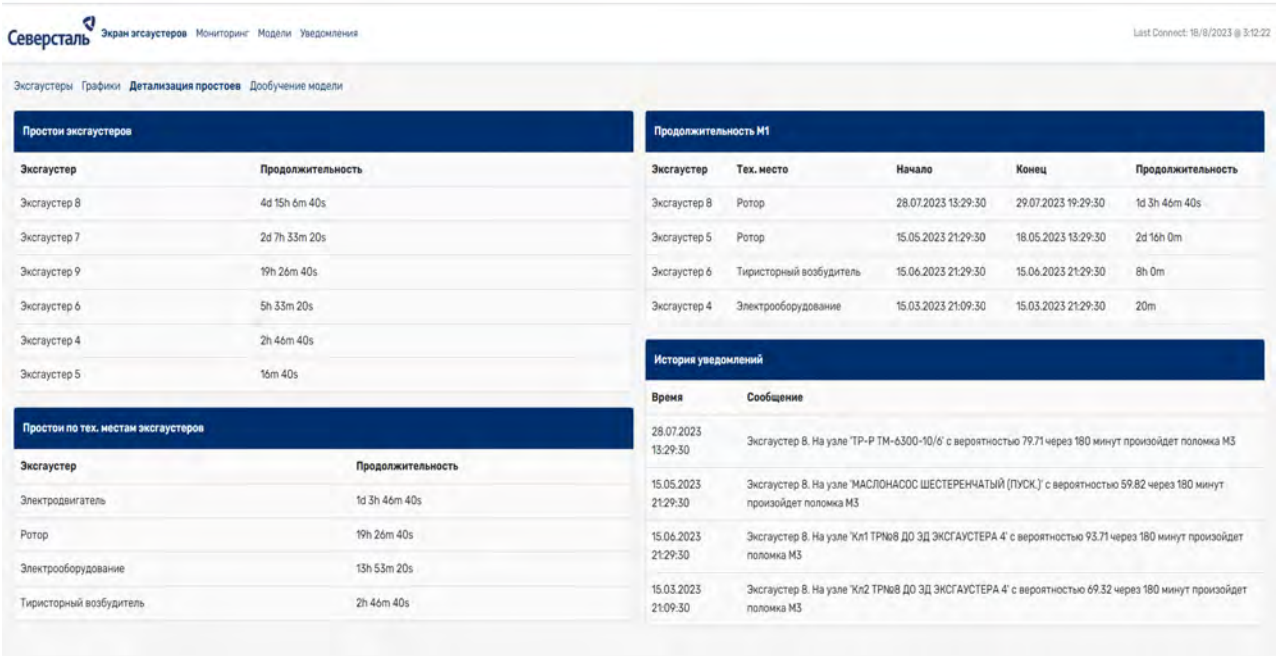


Рисунок 3.17 – Интерфейс, отображающий статистическую информацию о простоях

В данном случае прогнозирование по неисправностями типа М3 производилось на час вперед, поэтому сообщение содержит информацию о неисправностях, которые произойдут через 60 мин.

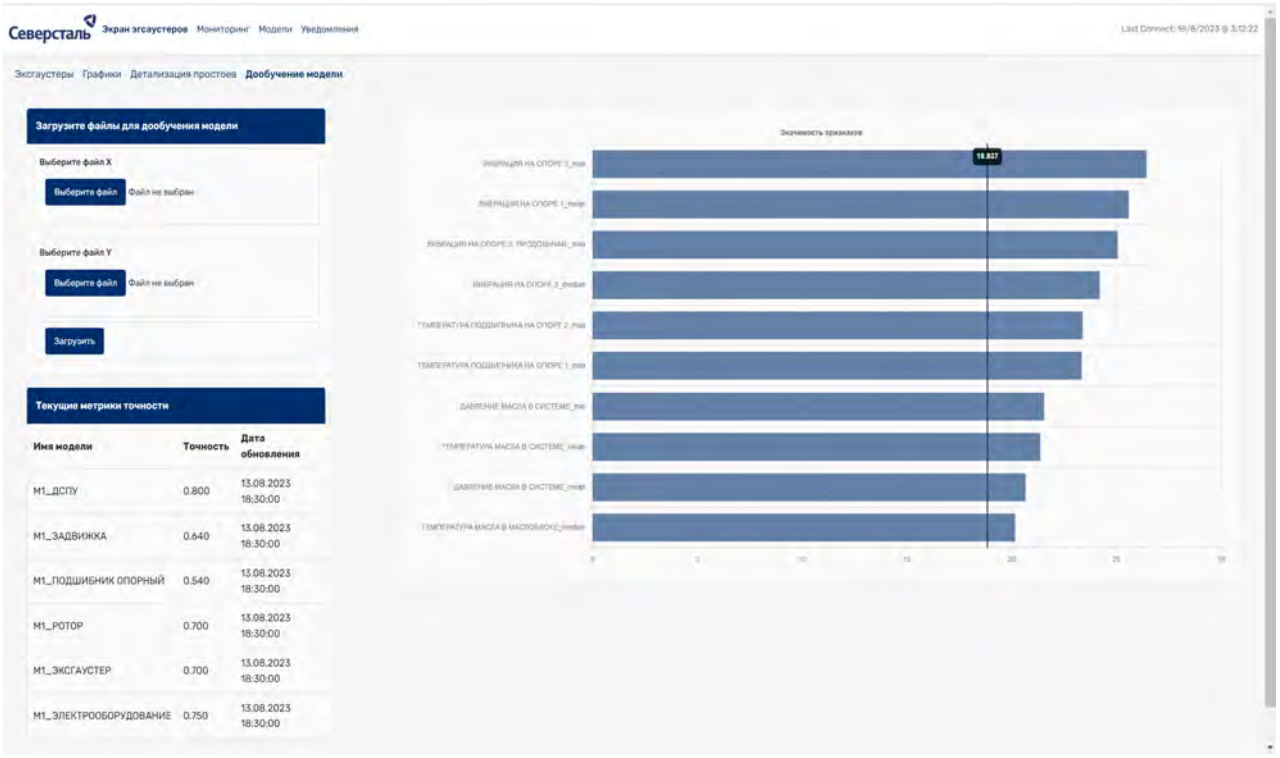


Рисунок 3.18 – Интерфейс, используемый для дообучения моделей

В системе предусмотрена возможность дообучения модели (рисунок 3.18). Для этого необходимо выбрать файлы X_train (показания датчик за определённый промежуток времени) и Y_train (разметка данных) в формате parquet или csv. Процесс обучения никак не влияет на текущую работоспособность системы. В случае необходимости, процесс можно остановить также через интерфейс.

В таблице «Текущие метрики» можно посмотреть актуальные значения метрик качества (индекса Жаккара и F1-score) с указанием времени расчета. По итогам дообучения (если оно завершилось успешно и значения точности повысились) в таблице отобразятся новые значения метрик качества.

Можно сохранить отдельно отчет (рисунок 3.19) за указанный период времени, содержащий информацию обо всех потенциальных неисправностях и отказах.

Отчет по текущим предсказаниям поломок

Таблица актуальной информации по возможным поломкам

№	Узел	Вероятность	Тип поломки	Время предполагаемой поломки
1	Э7: ТР-Р ТМ-4000-10/6 ЭКСГ. №7	99.06	M1	2023-06-27 15:14:26
2	Э9: МЕТРАН-100 ДАТЧИКИ ДАВЛЕНИЯ ЭКСГ.№9	98.86	M1	2023-06-27 15:14:28
3	Э7: МАСЛОНАСОС РАБОЧИЙ ЭКСГ. №7	97.28	M1	2023-06-27 15:14:26

Рисунок 3.19 – Пример вывода отчета

На графиках во вкладке «Мониторинг» (рисунок 3.20) отображаются исторические данные по включенным технологическим узлам со всех датчиков для всех машин за выбранный промежуток времени (в данном случае за 4 часа).

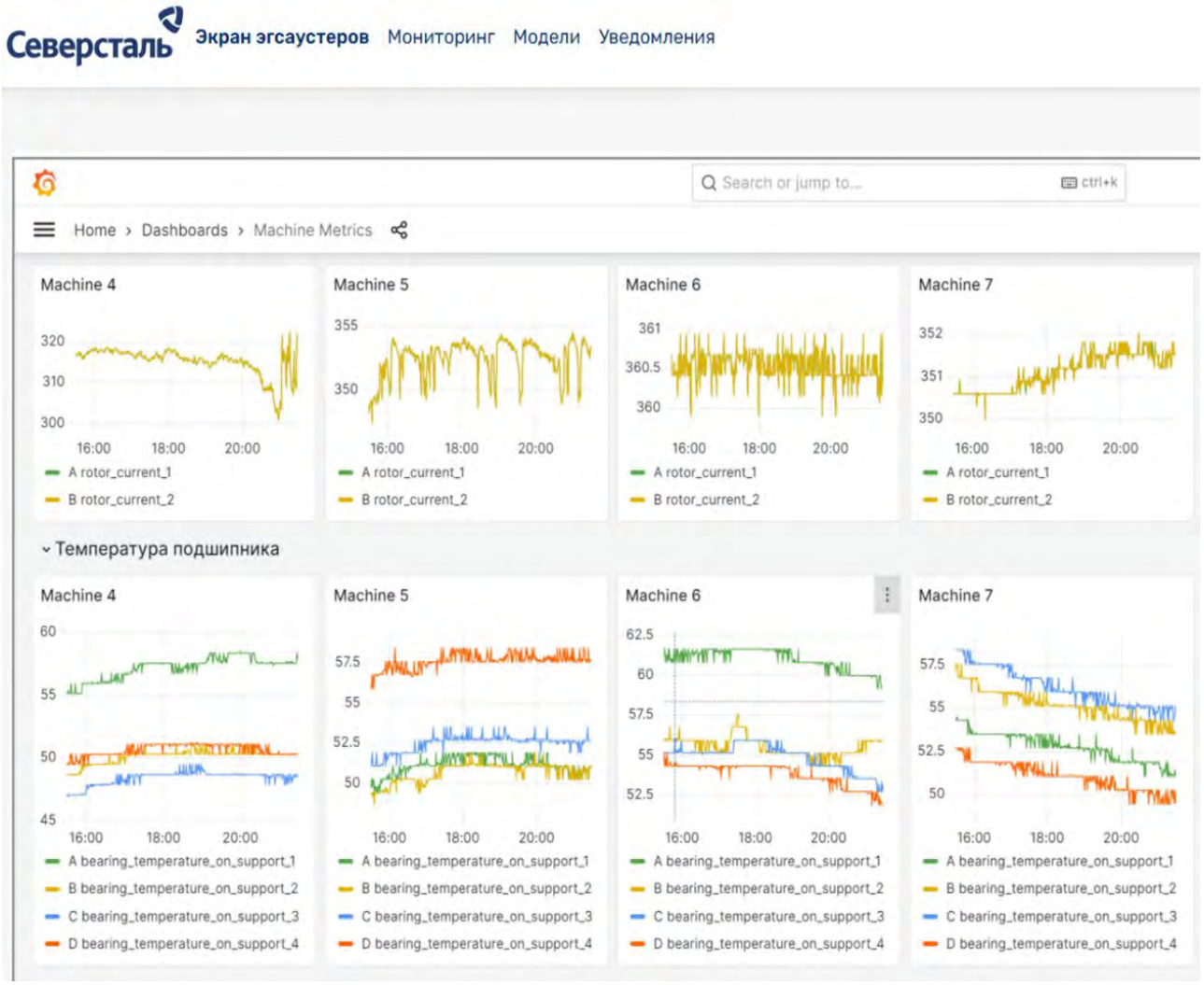


Рисунок 3.20 – Интерфейс мониторинга технического состояния оборудования

Также была настроена рассылка уведомлений на почту или в Телеграмм всем указанным заинтересованным пользователям.

За счет гибкой архитектуры разработанной автоматизированной системы прогнозирования технического состояния промышленного оборудования (рисунок 3.6), в АСУТП предприятия может быть интегрирована как каждая из ее подсистем, так и система в целом. Также предусмотрена интеграция с внешними системами, такими как система технического обслуживания и ремонта, в которой можно будет заводить наряды заблаговременно, или же оператор сможет изменять режимы работы оборудования благодаря прямому взаимодействию с исполнительными механизмами АСУТП.

3.6 Выводы по третьей главе

1. Эксгаустер является важным объектом технической диагностики и отсутствие прогнозирования технического состояния подобного оборудования требует частого технического обслуживания, что увеличивает затраты и время простоя.

2. Рассмотрена функциональная структура разработанной автоматизированной системы прогнозирования технического состояния эксгаустера агломашины.

3. Описана реализация подсистемы импорта данных с использованием платформы .NET.

4. Подсистема хранения и обработки данных реализована через использование двух СУБД. ClickHouse используется для хранения исходных и обработанных данных, получаемых из систем предприятия, а PostgreSQL - для хранения результатов работы моделей.

5. Для реализации подсистемы управления моделями применяется MLflow, т.к. данная платформа обладает гибким функционалом управления жизненным циклом моделей машинного обучения и предоставляет инструменты отслеживания, автоматизации и управления версиями моделей.

6. Гибкая архитектура системы реализована через Docker и Kubernetes, которые обеспечивают контейнеризацию и оркестрацию, что упрощает развертывание и масштабирование приложений, а также гарантирует целостность среды выполнения.

7. Подробно описана программная реализация алгоритмов для автоматизированной системы прогнозирования технического состояния промышленного оборудования на Python.

8. Разработаны пользовательские интерфейсы системы с использованием React, обеспечивающие динамичное взаимодействие с данными.

ГЛАВА 4. ТЕСТИРОВАНИЕ РАЗРАБОТАННОЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ДЛЯ ЗАДАЧИ КОНТРОЛЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ЭКСГАУСТЕРА АГЛОМАШИНЫ [2]

4.1 Предобработка данных

Особенностью реальных данных о состоянии оборудовании, собранных на основе показаний датчиков является достаточно высокая зашумленность, избыточность, наличие большого количества пропущенных значений и, как правило, сложная структура. Эффективность и производительность прогностических моделей, основанных на машинном обучении, напрямую зависит от качества и информативности входных данных. В связи с этим возникает необходимость в тщательной очистке и предобработке обучающего набора данных.

В процессе подготовки исходных данных к дальнейшему использованию моделями прогнозирования, как правило, решается несколько проблем.

Во-первых, необходимо произвести общую аналитику распределения показаний датчиков и выявить выбросы, которые негативно сказываются как на дальнейшие преобразования, так и на обучение моделей. Однако, не всегда есть возможность отличить случайные шумы и скачки, ассоциированные с изменением состояния оборудования. В связи с этим на первоначальном этапе вручную происходит удаление только тех значений датчиков, которые явно не соотносятся с реальными границами разброса измеряемого параметра. В частности, к подобным выбросам относятся отрицательные значения температуры, силы тока, колебаний и давления, а также значения,

[2] В данной главе использованы материалы, ранее опубликованные в работах:
Chernukhin A. V. Development of a model for predicting failures of high-tech production equipment based on machine learning algorithms / A. V. Chernukhin, E.A. Bogdanova, T.V. Savitskaya, A.M. Sverchkov, A.V. Dementienko // Third International Conference on Optics, Computer Applications, and Materials Science. – 2023. – № 11. – С. 1306515.
Чернухин А. В. Система предиктивной аналитики технического состояния эксгаустера агломашины с помощью методов искусственного интеллекта / А. В. Чернухин, Е. А. Богданова, Т. В. Савицкая, Д. Г. Кулаков, И. Р. Павлов // Искусственный интеллект и принятие решений. – 2024. – № 3. – С. 87-103.
Чернухин А. В. Построение модели предиктивной аналитики неисправностей промышленного оборудования / А. В. Чернухин, Е.А. Богданова, Т. В. Савицкая // Программные продукты и системы. – 2024. – № 2. – С. 254-261.
Чернухин А. В. Разработка алгоритма прогнозирования остаточного ресурса эксгаустера агломашины, основанного на машинном обучении / А. В. Чернухин, Т. В. Савицкая // Научно-технический вестник Поволжья. – 2024. – № 4. – С. 185-190

превышающие среднее значение более чем в 100 раз. Как правило, подобные скачки единичные и не несут физического смысла и информации, важной для понимания состояния оборудования. При этом идентификация подобных значений может свидетельствовать о сбоях в работе самих датчиков и возможной необходимости их настройки или замены.

Во-вторых, практически всегда в собранных данных можно обнаружить пропущенные значения, появление которых может быть ассоциировано с различными факторами, такими как сбои в работе датчиков, ремонтные и профилактические работы и др. Использование данных с пропусками для обучения и тестирования моделей прогнозирования невозможно, и, следовательно, возникает необходимость заполнения пропусков или их удаление. В связи с тем, что используемые данные относятся к временным рядам, удаление моментов времени с пропущенными значениями может привести к искажению информации о динамических характеристиках системы. Таким образом, единственным вариантом является заполнение пропущенных значений на основе известных значений во временном ряду. Существует несколько подходов обработки пропусков:

- Заполнение средними или медианными значениями ряда. Данный метод достаточно часто используется для непрерывных колеблющихся переменных. Однако такой подход имеет ряд серьезных недостатков, в частности, снижается вариативность данных, а также теряется информация о динамических изменениях таких характеристик как температура, давление и др.
- Заполнение константными значениями. Чаще всего в данном методе пропущенные значения заменяются на 0. Для временных рядов и в целом непрерывных переменных данный способ не подходит, так как вносит шум в значимую информацию о состоянии системы.
- Заполнение методом интерполяции. В данном подходе осуществляется замена пропуска путем расчета среднего значение между предыдущей известной точкой и точкой, следующей за пропуском. Подобный

метод является оптимальным для заполнения временных рядов из рассмотренных, так как сохраняет информацию о динамических изменениях характеристик и не снижает вариативность данных.

В-третьих, данные о состоянии оборудования являются достаточно избыточными, так как интервал измерения физических параметров составляет 10 секунд. Учитывая, что общая информация об оборудовании содержит измерения за несколько лет, использование изначальных интервалов не представляется возможным в связи с наличием большого количества случайных отклонений. Для сглаживания значений и оптимизации их использования в машинном обучении используется переход к минутным и даже часовым интервалам. Таким образом уменьшается вклад зашумления и улучшается обобщаемость признаков. Для перехода от 10-секундных интервалов к более крупным, происходит расчет набора статистических параметров за весь временной отрезок (например, за час). Используемые в данной работе преобразования указаны в Таблице 4.1 [117-120].

Таблица 4.1. Набор используемых функций преобразования исходных данных.

Функция преобразования	Формула
Среднее значение	$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
Стандартное отклонение	$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})}$
Медиана	$Me = x_{Me} + i_{Me} \cdot \frac{\frac{\sum f}{2} - S_{Me-1}}{f_{Me}}$
Максимум	$Max = max(x_1, \cdots, x_n)$
Минимум	$Min = min(x_1, \cdots, x_n)$

В последствии рассчитанные статистические характеристики для часовых интервалов используются в качестве признаков, описывающих состояние оборудования, при обучении моделей прогнозирования.

В-четвертых, в связи с разной природой измеряемых датчиками параметров возможно возникновение ситуации, при которой предсказательный алгоритм будет рассчитывать важность признаков не на основе его влияния на общее состояние оборудования, а на базе его увеличенного разброса относительно других признаков. Таким образом, для улучшения обобщающей способности модели необходимо производить нормализацию признаков [121]. Наиболее часто используемыми методами нормализации являются Мин-Макс нормализация и стандартизация. В первом случае каждый параметр преобразуется по следующей формуле:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}, \quad (4.1)$$

где x – текущее значение параметра, x_{min} – минимальное значение ряда, x_{max} – максимальное значение ряда. Так, данные преобразуются в значения от 0 до 1. Минусом данного вида нормализации является большой вклад шумов и выбросов в итоговое распределение. В связи с этим в данной работе для показаний датчиков используется стандартизация, так как она менее подвержена влиянию выбросов и лучше сохраняет связь между наблюдениями. Для каждого параметра при стандартизации рассчитывается новое значение по формуле:

$$x_{stand} = \frac{x - \bar{x}}{\sigma}, \quad (4.2)$$

где, x – текущее значение параметра, $\bar{x}$ – среднее значение ряда, σ – стандартное отклонение. Таким образом данные преобразуются в ряд, в котором среднее значение будет равно 0, а стандартное отклонение – 1.

В-пятых, одной из ключевых проблем использования временных рядов в качестве признаков для обучения методов классического машинного обучения является потеря информации о динамическом изменении измеряемых

параметров во времени. Это связано с тем, что каждый отдельный объект, для которого происходит предсказание, должен быть представлен вектором.

Отбор среди существующих признаков и конструирование новых является одной из ключевых точек в предобработке данных, необходимой для работы с алгоритмами машинного обучения. Существует несколько подходов решения данной задачи. Для отбора среди изначальных признаков используется корреляционный анализ. Для всех признаков попарно рассчитывается коэффициент корреляции Пирсона по формуле:

$$r_{xy} = \frac{\sum(x_i - M_x)(y_i - M_y)}{(n-1)\sigma_x\sigma_y}, \quad (4.3)$$

где x, y – значения исследуемых признаков, M_x, M_y – средние значения признаков, σ_x, σ_y – стандартные отклонения, n – количество наблюдений.

Далее, среди пары признаков с высокой корреляцией (более 0,95) отбирается один, наименее коррелирующий с остальными. Помимо этого, часто отсеиваются признаки, значения которых во времени остаются близкими к константным, так как это свидетельствует об отсутствии их связи с предсказываемой характеристикой.

Конструирование новых признаков является более сложной и неоднозначной задачей. По возможности для ее решения используются знания экспертов в предметной области [122]. Однако, в большинстве случаев применяются различные алгоритмические подходы к поиску оптимального признакового описания. В частности, для внесения в описательный вектор информации о динамических характеристиках используются дополнительные расчеты статистических характеристик с введением окна скольжения.

Так, скользящее среднее рассчитывается по формуле:

$$\hat{x}_n = \frac{1}{k}\sum_{i=k-n+1}^{n} x_i, \quad (4.4)$$

где k – ширина окна сдвига, номер члена ряда, значение которого заменяется средним.

Таким образом, в вектор, содержащий информацию о показаниях датчиков за один час, можно внести информацию о средних или медианных

значениях за последние сутки, несколько суток или неделю (в зависимости от динамических характеристик конкретного физического параметра). Добавление данных признаков позволяет внести в обучающие данные значимую информацию о более долговременных состояниях системы.

Описанные выше этапы предобработки данных являются базовыми и используются для подготовки данных к дальнейшему их использованию в качестве признаков для обучения моделей как классического машинного обучения, так и нейронных сетей.

Исследуемый в диссертационной работе набор данных включает в себя показания датчиков для шести эксгаустеров, собранный за период с 01.01.2019 по 31.12.2021. Шаг фиксации измерений равен 10 секундам. Формат данных представлен в Таблице 4.2.

Датчики производят измерения тока ротора и статора (А), температур технических мест (°С) и масла в системе, давление масла (КПа), вибраций на опорах (мм/с). Всего проводился анализ показаний 16 датчиков.

Таблица 4.2. Формат данных о показаниях датчиков за анализируемый период для одного эксгаустера

Дата	Датчик 1	Датчик 2		Датчик 15	Датчик 16
01.01.2019 00:00:00	123	0		345	17
01.01.2019 00:00:10	135	0.57		320	17
01.01.2019 00:00:20	127	0.93		387	19
…	…	…		…	…
31.12.2021 23:59:40	69	2.4		341	0
31.12.2021 23:59:50	63	1.2		390	3

Предобработка показаний датчиков включала в себя следующие этапы:

Шаг 1. Удаление выбросов – производилось обнуление отрицательных показаний датчиков, а также уменьшение амплитуды значений, превышающих

среднее более, чем в сто раз (т.к. подобные значения не являются нормальными показаниями датчиков и негативно сказываются на обучении) (рисунок 4.1).

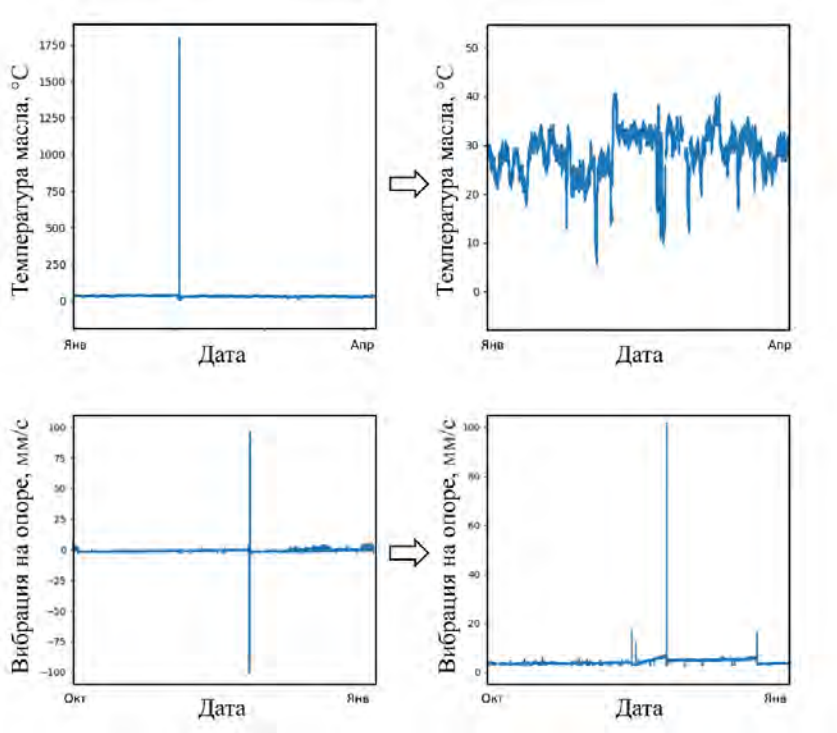


Рисунок 4.1 – Температура масла в системе и вибрация на опоре во фрагментах обучающей выборки до и после удаления выбросов

Шаг 2. Приведение данных к часовым интервалам из десятисекундных – показания с шагом в 10 секунд являются избыточными и, как следствие, неоптимальны для обучения предсказательного алгоритма в связи с зашумленностью данных и большим объёмом занимаемой памяти (необходимо хранение и работа с таблицей, содержащей 9460800 строк для каждого эксгаустера). Для дальнейшей работы для каждого из часовых интервалов (360 строк) рассчитывались статистические характеристики, такие как среднее, медиана, максимальные и минимальные значения. В дальнейшем именно эти значения использовались в качестве признаков при обучении модели.

Шаг 3. Стандартизация и замена пропущенных значений - в связи с различной природой показаний датчиков, производилась их стандартизация. Для этого рассчитывалось обобщенное для всех эксгаустеров среднее значение и стандартное отклонение для каждого датчика, которые и использовались для

стандартизации (рисунок 4.2). Оставшиеся пропущенные значения заполнялись путем интерполяции.

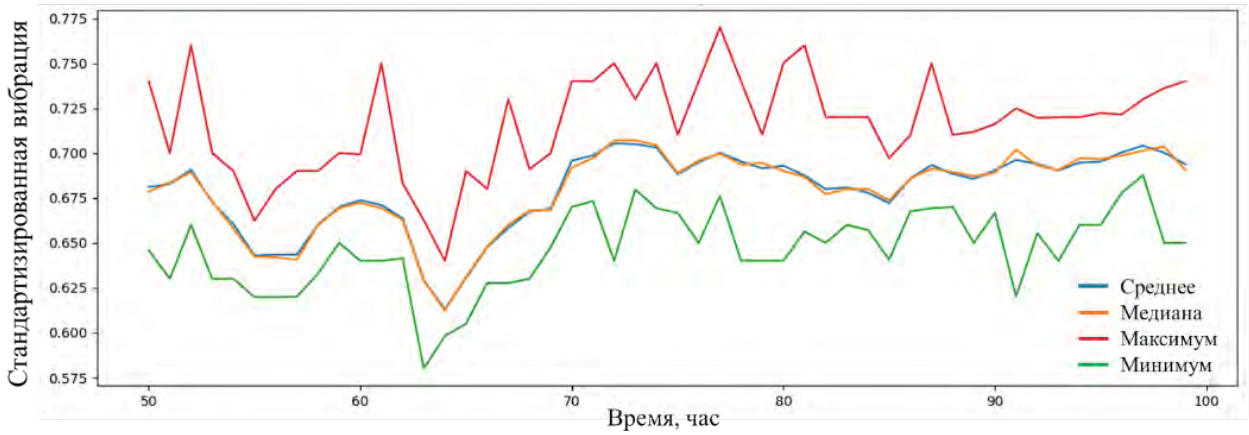


Рисунок 4.2 – Статистические характеристики, рассчитанные за часовые интервалы для вибрации

Шаг 4. Добавление новых признаков - для получения информации об отклонении показаний датчиков за текущий час от показаний за предыдущие сутки и неделю, рассчитывалась разность между средними значениями за эти периоды (рисунок 4.3). Полученные изменения в дальнейшем также использовались в качестве признаков, так как позволяют учитывать информацию не только о локальных изменениях состояния системы, но и выявлять отклонения от состояния за более долгий промежуток времени.

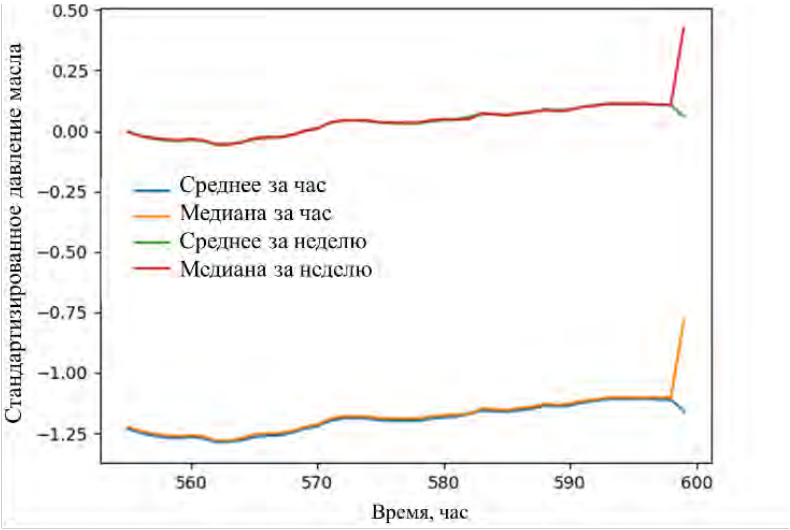


Рисунок 4.3 – Медианные и средние значения давления масла в системе, рассчитанные для часовых (синий, оранжевый) и недельных (красный, зеленый) интервалов

Информация о неисправностях технических мест эксгаустеров (Т.М. i) была предоставлена в виде таблицы с бинарным кодированием (0 - штатное

состояние технического места, 1 - неисправная работа) (табл. 4.3). Количество и набор технических мест отличался для каждого из эксгаустеров (от 20 до 35). Информация об отказах технических мест эксгаустеров предоставлялась в виде таблицы такого же формата.

Таблица 4.3. Формат данных, содержащих метки неисправностей за анализируемый период

Дата	Т.М. 1	Т.М. 2		Т.М. 20	Т.М. 21
01.01.2019 00:00:00	0	0		1	0
01.01.2019 00:00:10	0	0		1	0
01.01.2019 00:00:20	0	0		1	0
…	…	…		…	…
31.12.2021 23:59:40	1	0		0	0
31.12.2021 23:59:50	1	0		0	0

Помимо основной информации о неисправностях и отказах также предоставлялись отчеты о сообщениях при возникновении данных нештатных ситуаций с описанием причин.

На первом этапе производился отбор неисправностей, подходящий для дальнейшего анализа, в частности из обучающей выборки удалялись данные об отказах и неисправностях, связанных с человеческим фактором в связи с невозможностью их прогнозирования. Также удалялись неисправности, продолжительность которых была менее 1 минуты. Дальнейшая обработка информации о неисправностях и отказах различалась в зависимости от решаемой задачи и архитектуры предсказательного алгоритма.

4.2 Апробация метода прогнозирования неисправного состояния оборудования

Из-за сложности накопления и получения дополнительных реальных данных для тестирования, для апробации предлагаемых подходов в каждой

задаче независимо собиралась тестовая выборка из полученного изначально исторического периода. Обучающая и тестовая выборки представляют собой многомерные временные ряды и имеют сходную структуру.

Выделение из временных рядов данных для тестирования модели прогнозирования требует особого подхода, поскольку данные такого формата имеют временную зависимость. Это означает, что порядок данных важен, и случайное разбиение может привести к утечке информации из обучающей выборки в тестовую. В связи с этим выделение данных для тестирования происходило на основе следующих важных принципов:

- Сохранение хронологического порядка. Временные ряды имеют временную структуру, поэтому данные должны быть разделены в хронологическом порядке. Это означает, что тестовая выборка должна содержать данные, которые идут после данных обучающей выборки.
- Репрезентативность выборки. В наборе данных для тестирования должны присутствовать все типы неисправностей и отказов, прогнозированию которых обучается модель.
- Обобщаемость для разных единиц оборудования. В тестовом наборе данных должны присутствовать случаи неисправностей и отказов разных эксгаустеров, чтобы предотвратить переобучение модели на одном типе оборудования.

Таким образом, апробация позволяет оценить, насколько хорошо модель может обобщать знания, полученные на обучающей выборке, на новые, невидимые данные. Это критически важно для оценки того, как модель будет работать в реальных условиях. Также тестирование предсказательного алгоритма на независимом наборе данных позволяет подготовить модель к реальному использованию, обеспечивая уверенность в ее способности делать предсказания в условиях, близких к реальным.

4.2.1 Формирование тестовой выборки

В качестве тестового набора отбирались самые поздние интервалы, следующие после обучающих и валидационных фрагментов и содержащие

неисправности (рисунок 4.4). В связи с тем, что периоды неисправного состояния оборудования для разных технических мест не совпадают, для каждого из них разделение происходило индивидуально, с учетом причин, повлекших за собой неисправность.

Далее происходила предобработка показаний датчиков, включающая в себя удаление выбросов, стандартизацию (среднее и стандартное отклонения рассчитывались только по обучающей выборке) и выделение новых признаков.

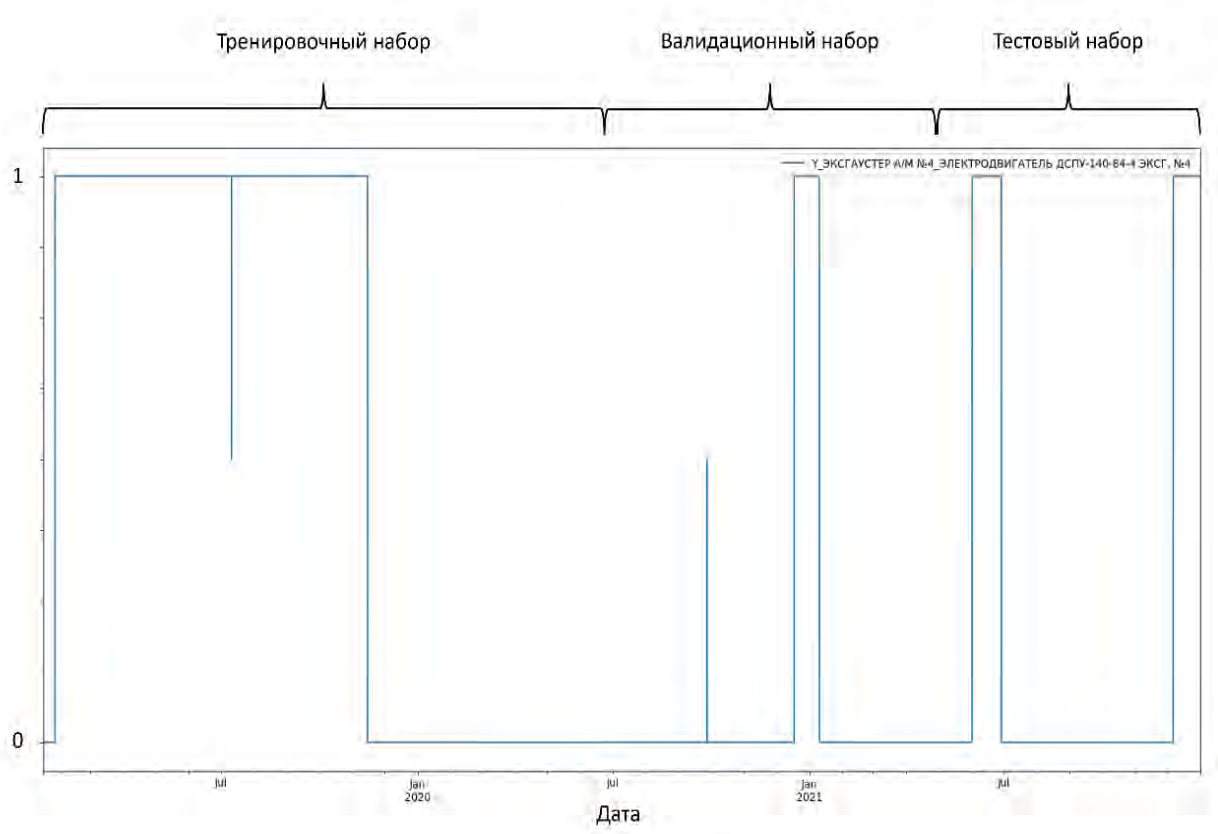


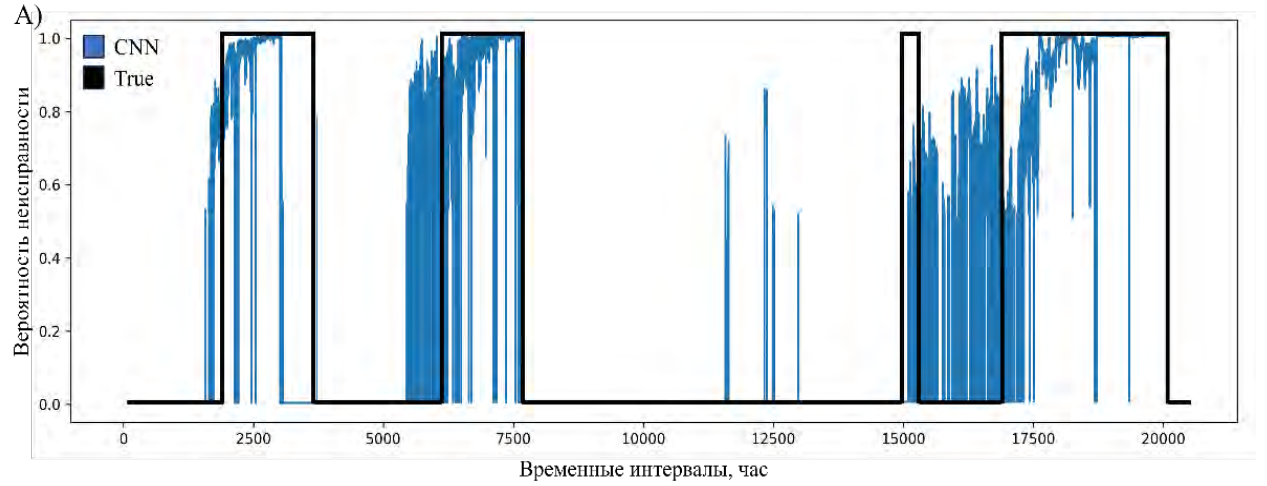
Рисунок 4.4 – Принцип выделения тестовой выборки для электродвигателя эксгаустера

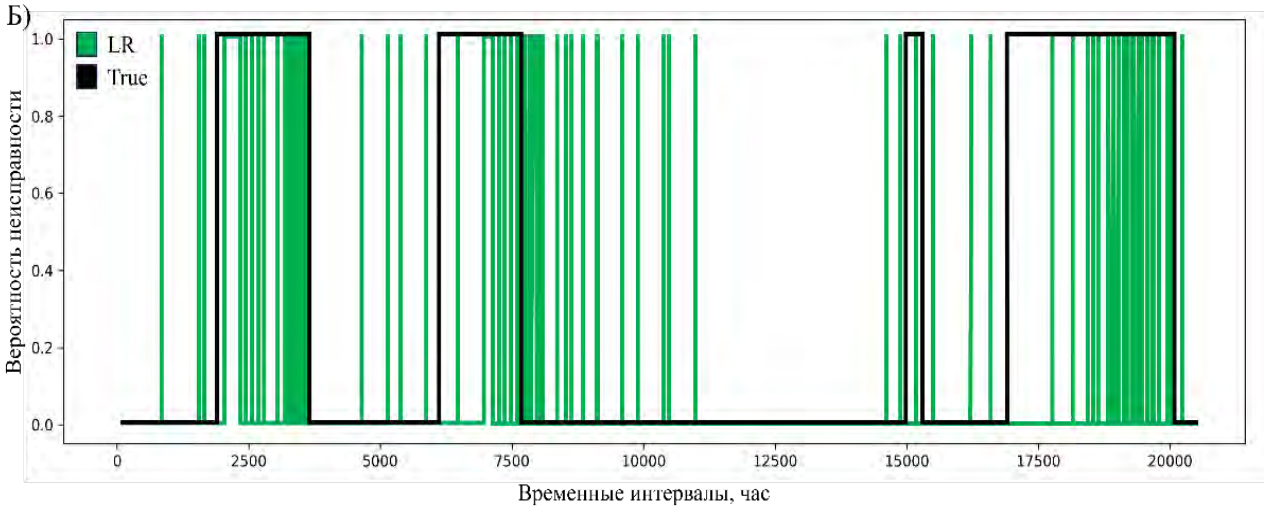
4.2.2 Результаты тестирования

На первом этапе работы были обучены модели прогнозирования по отдельности для каждого технического места. Для идентификации неисправностей в разных техместах индивидуально подбирались пороги принятия решения. На рисунке 4.5А представлено распределение вероятностей наличия неисправности на тестовой выборке для ротора (в качестве порога

бралось значение 0,5), при этом индекс Жаккара равен 0,6.

В тестовых временных интервалах для трех эксгаустеров, объединенных в общий тестовый набор, наблюдаются три продолжительные неисправности и одна более короткая (рисунок 4.5А). Можно заметить, что в трех случаях модель начинала присваивать высокую вероятность развития неисправности заранее, до официального начала, обозначенного оператором. Что, с одной стороны, отрицательно повлияло на значение метрики качества, а с другой, является положительным моментом, позволяющим диагностировать начало развития неисправности до того, как это явным образом повлияет на работу эксгаустера и будет замечено оператором. Однако, одна из неисправностей не была диагностирована, что может быть связано иной с причиной ее возникновения, которая слабо представлена в обучающем наборе данных.





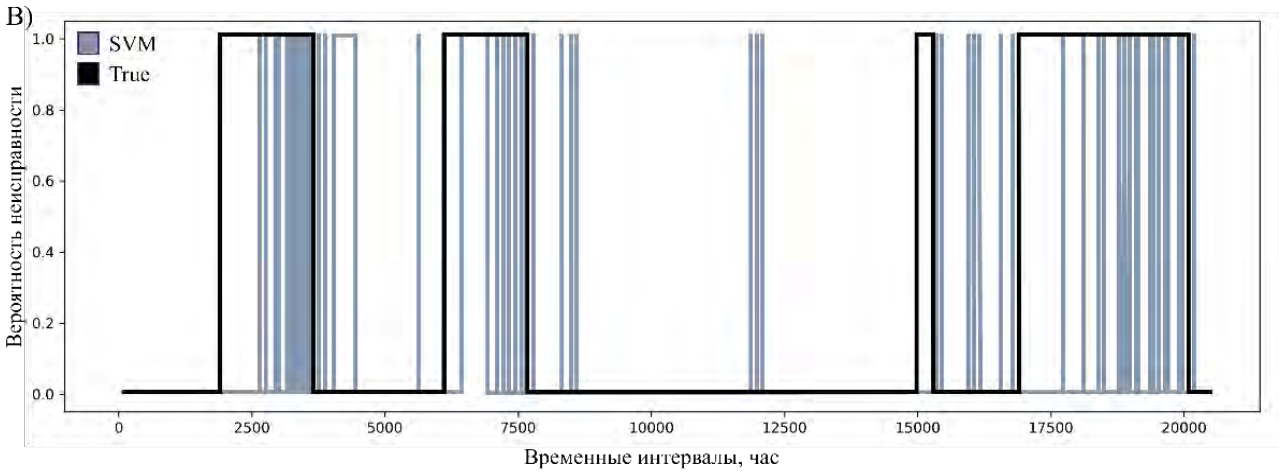


Рисунок 4.5 – Предсказания для часовых интервалов на тестовой выборке для ротора, полученные: а) нейронной сетью (CNN); б) методом логистической регрессии (LR); в) методом опорных векторов (SVM). True - истинные метки неисправностей

Также можно заметить, что результаты, полученные другими моделями, достаточно сильно отличаются и относят к неисправностям периоды их активного развития (рисунок 4.5Б, 4.5В), и также не идентифицируют непродолжительную неисправность, что подтверждает вышеупомянутое предположение. При этом значение метрики качества для логистической регрессии равно 0,51, для метода опорных векторов - 0,4 (рисунок 4.6).

Полученные данные об особенностях прогнозирования разных этапов развития неисправности ротора свидетельствуют о возможности улучшения качества предсказания при использовании ансамблевого подхода, объединяющего ранее названные модели. Метрики качества для используемых подходов, включая ансамблевые показаны на рисунке 4.6.

В результате, благодаря использованию ансамблевого подхода, включающего операцию «мягкого голосования» между моделями, для ряда технических мест удалось значительно улучшить качество предсказания на тестовой выборке (рисунок 4.6).

Наилучшие показатели наблюдаются для корпуса, улиты, электроаппаратуры и вибропреобразователя. Наиболее высокое качество предсказания получили те технические места, неисправная работа которых имеет быстрое развитие (электроаппаратура, вибропреобразователь,

электродвигатель), а также места с наиболее заметными и легко идентифицируемыми проявлениями неисправной работы (улита, корпус, вибропреобразователь). Следовательно, разметка данных для этих точек может производиться в более оперативном режиме, что важно для обучения моделей «с учителем».

Однако, для остальных техузлов, несмотря на более низкие значения метрик, обученные модели предсказывают зарождение неисправности, до момента идентификации её сотрудниками, что позволит заранее подготовиться к развитию аномалии в работе оборудования и принять необходимые превентивные меры.

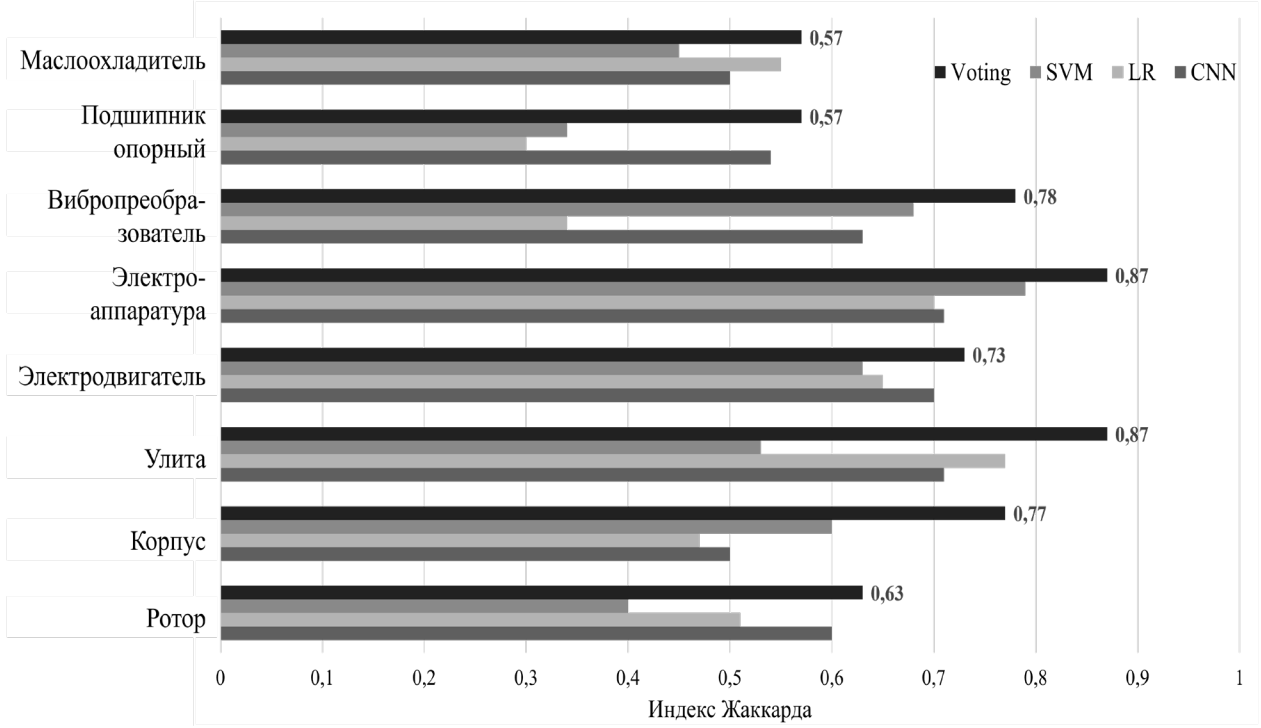


Рисунок 4.6 – Результаты предсказания на тестовой выборке. По вертикали указаны технические места, для которых осуществлялось предсказание. По горизонтали – значение метрики качества для каждого из методов прогнозирования

4.3 Апробация метода прогнозирования отказов оборудования

4.3.1 Формирование тестовой выборки

Отказ эксгаустера является редким и краткосрочным явлением (происходит лишь несколько раз в год). В связи с этим для тестового датасета выбирались из исходного набора интервалы с наиболее поздними случаями отказов (по 3 для каждого технического места): за три месяца до отказа и

оставшийся интервал после (рисунок 4.7).

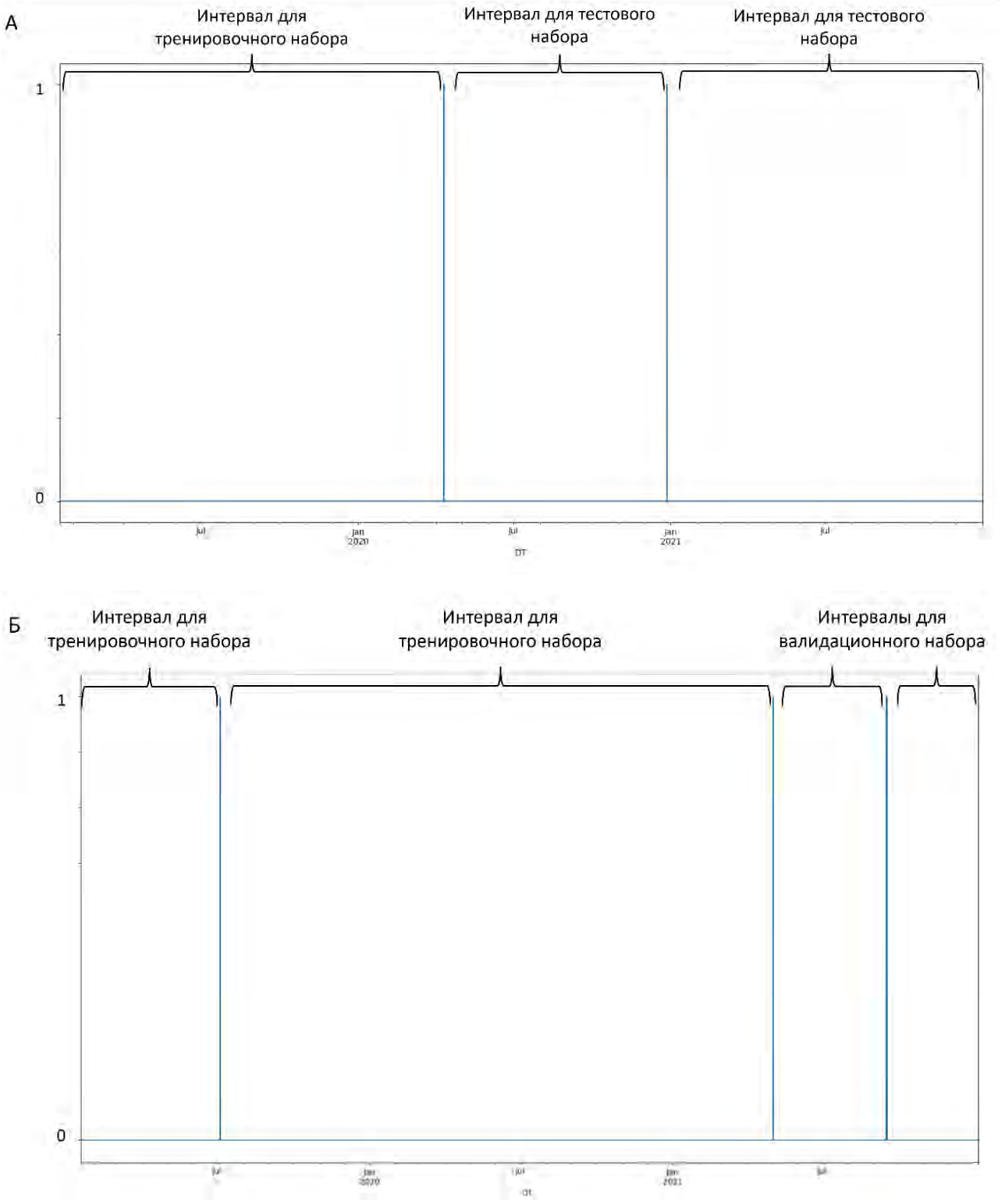


Рисунок 4.7 – Варианты разбиения данных на тренировочный, валидационный и тестовый интервалы для электродвигателя. По вертикали 1 отмечены моменты отказа оборудования, 0 – штатная работа; по горизонтали – дата; А) выделение тренировочной и тестовой выборок для эксгаустера 1; Б) выделение тренировочной и валидационной выборок для эксгаустера 2

В связи с ограниченным количеством данных об отказах за исторический период для каждого из эксгаустеров разбиение происходило индивидуально: тренировочная выборка выделялась во всех эксгаустерах, а валидационная (рисунок 4.7Б) и тестовая (рисунок 4.7А) в разных.

Таким образом было обеспечено наличие полноценной информации обо всех эксгаустерах для обучения, а также наличие валидационной и тестовой выборок из отказов в разных эксгаустерах, что обеспечивает более объективное оценивание предсказательной и обобщающей способности модели при тестировании.

4.3.2 Результаты тестирования

В ходе изучения данных было выявлено, показания каких датчиков наиболее сильно отличаются от нормального состояния при развитии неисправности, ведущей к отказу в различных технических местах. По итогу произведенного анализа случаев отказов эксгаустеров было выявлено, что развитие отказов в разных технических местах наиболее значимо отражается в показаниях определенного набора датчиков (табл. 4.4).

Таблица 4.4. Показатели, измеряемые датчиками, с учётом релевантности отображения развития критичных аномалий в работе технических мест.

Показатели, измеряемые датчиками	**Техническое место**			
	Ротор	**Электродвигатель**	**Подшипники**	**Электро-оборудование**
Температура на опорах	+	+	+	-
Вибрации на опорах	+	-	+	-
Давление масла	+	-	-	-
Температура масла	+	-	-	-
Ток ротора	-	+	-	+
Ток статора	-	+	-	+

В технической документации по работе эксгаустеров агломашины описаны критичные показания датчиков (температура, давление и т.д.), при которых происходит непосредственно отключение оборудования. При этом в задаче ранней диагностики данные показатели таких значений не достигают, следовательно, анализ осуществляется в допустимых границах всех физических параметров, что значительно осложняет поиск наиболее значимых характеристик непосредственно для раннего прогнозирования.

Одними из наиболее нагруженных, и, в связи с этим, наиболее изученных компонентов эксгаустера в плане развития неисправностей являются подшипники опор ротора. Так как работа этих технических мест связана с постоянной ударной нагрузкой, их изнашивание происходит постепенно, и развитие неисправностей связано с плавным или скачкообразным увеличением вибрации на опорах [169].

Также установлено, что при изнашивании подшипников происходит нелинейное изменение температуры в большую сторону, с возможным возникновением скачкообразного роста [170], что связано с увеличением теплоотдачи самих подшипников и увеличением температуры масла [171].

Следовательно, данные физические показатели являются одними из ключевых для определения состояния подшипников. Учитывая, что из работы предложенного алгоритма также получилось установить наибольшую роль данных показателей, можно предположить, что использован верный подход к анализу и извлечению признаков. Часть из этих отклонений можно наблюдать визуально при анализе изменений средних значений показаний датчиков за неделю. Это относится к температурам и вибрациям подшипников.

На рисунке 4.8А можно наблюдать отличие изменения температуры подшипника на опоре 2 относительно остальных, причем происходило это неоднократно с увеличением амплитуды, что по итогу привело к отказу эксгаустера по причине неисправности в электродвигателе.

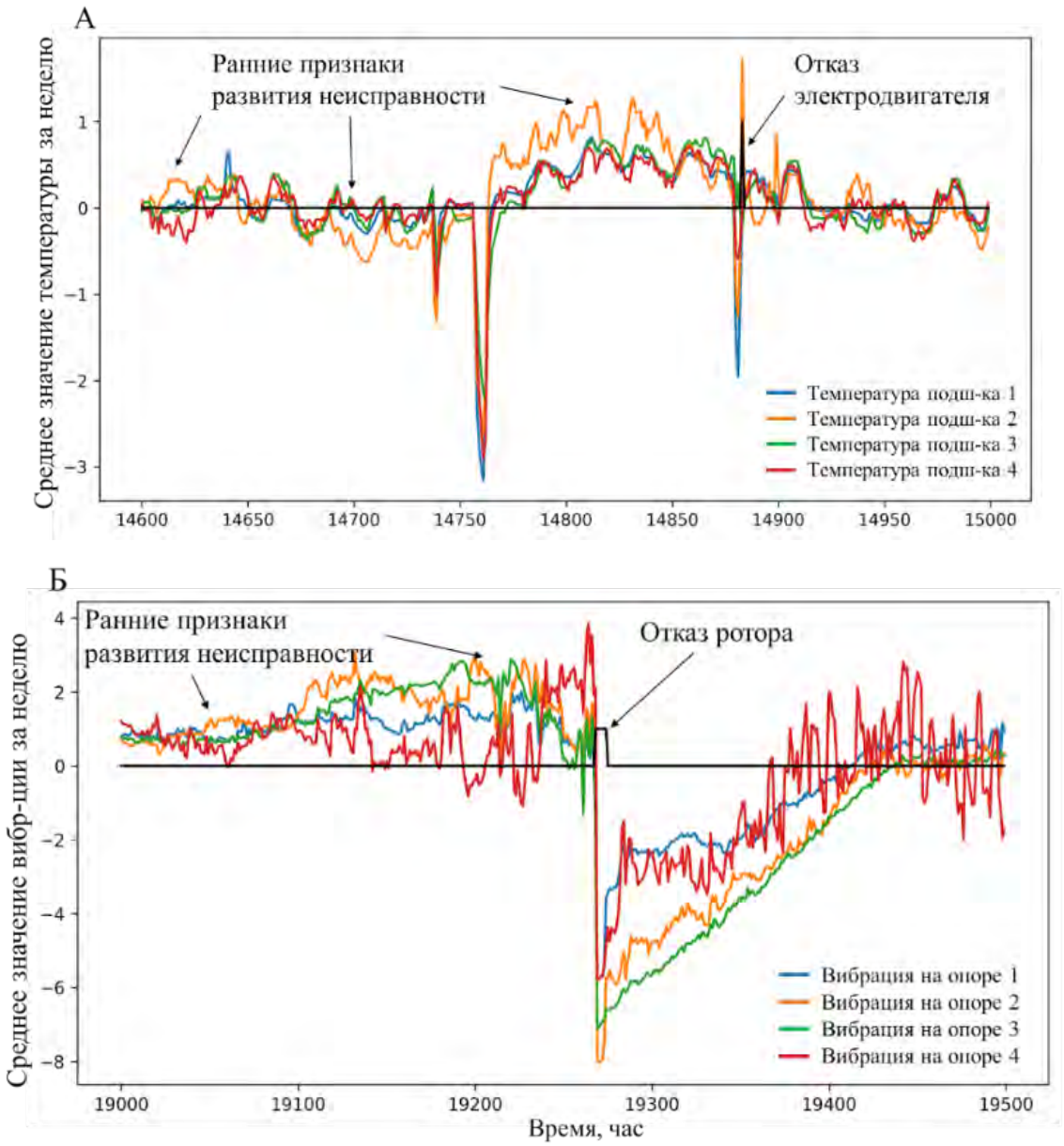
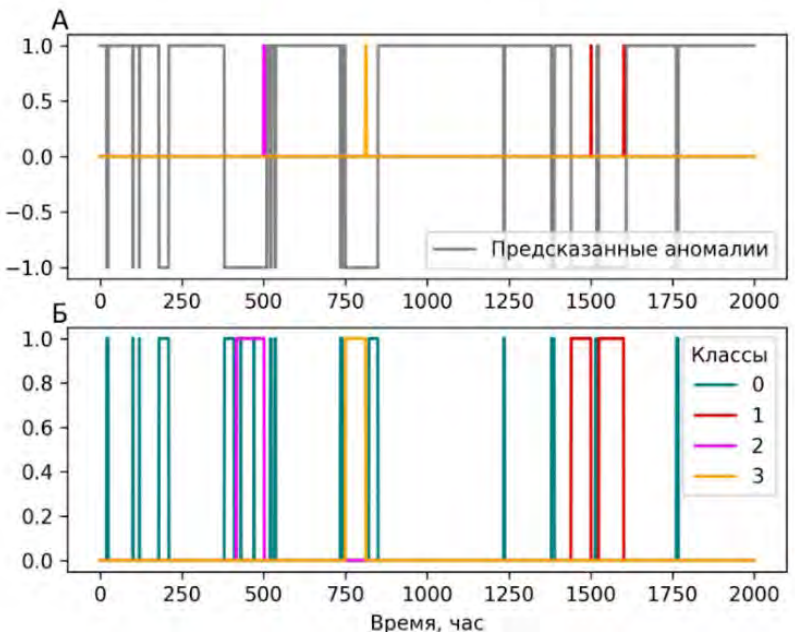


Рисунок 4.8 – Рассчитанные характеристики показаний датчиков перед отказами эксгаустера

Похожую ситуацию можно увидеть на рисунке 4.8Б, однако в данном случае происходит рассинхронизация между вибрациями на опорах 2, 3 и вибрациями на опорах 1, 4, так как их изменения за 100 часов до неисправности начинают принимать противоположные градиенты.

Для тестового набора данных предсказание осуществлялось в два этапа: поиск аномальных режимов работы с помощью обученного изолирующего леса (рисунок 4.9А, В) и классификация выделенных аномалий по техническим местам, в случае возникновения отказа (1-4 классы), в противном случае отнесение к нормальному режиму работы (класс 0) (рисунок 4.9Б, Г). Как

видно на рисунках 4.9А, В, все периоды развития неисправностей, повлекших за собой простой в работе двух эксгаустеров, относятся изолирующим лесом к аномальным интервалам, при этом для части таких периодов наличие отказа оборудования не наблюдается. Вероятно, это связано с тем, что многие зарождающиеся дефекты обнаруживаются сотрудниками при профилактических работах и исправляются до того, как их развитие приведет к критическим сбоям. По результатам классификации выделенных аномальных режимов (рисунок 4.9Б, Г) можно увидеть, что для ротора и подшипников периоды развития дефектов выделяются согласно экспериментальным данным, период перед отказом электродвигателя также верно классифицируется, однако есть локальные интервалы, определяемые как нормальная работа, что может быть связано с более динамичным развитием неисправностей в данном техническом месте. При этом, можно заметить, что предсказания для разных единиц оборудования (двух эксгаустеров) имеют стабильно высокое качество, что свидетельствует о хорошей обобщающей способности модели [172]. В результате обучения классификатора на тестовом наборе данных удалось достичь среднего значения оценки F1, равной 0,74. Также для каждого технического места было отдельно рассчитано значение метрики качества: ротор – 0,92; подшипники – 0,9; электродвигатель – 0,76; электроаппаратура – 0,38.



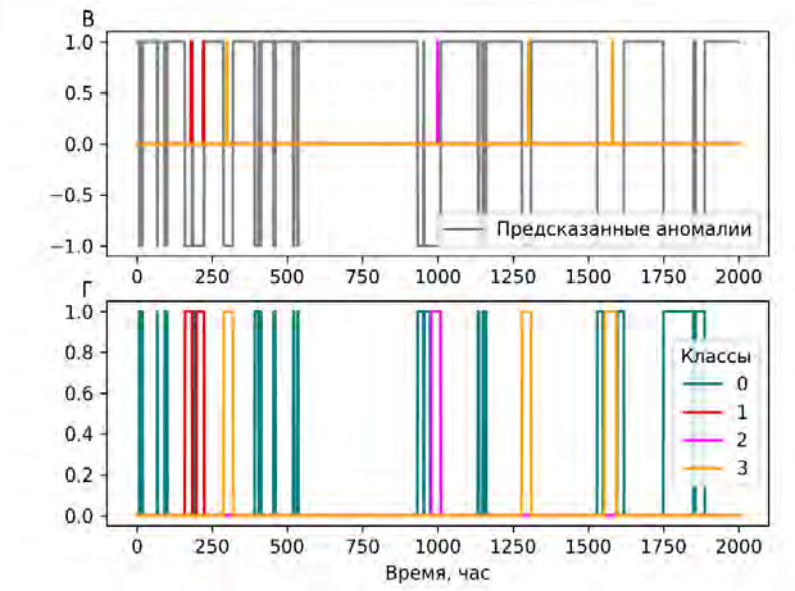


Рисунок 4.9 – Результат прогнозирования отказов на фрагментах тестового датасета для двух эксгаустеров (А, Б – эксгаустер 1, В, Г – эксгаустер 2). Нормальный режим работы - 0, ротор - 1, электродвигатель - 2, подшипники – 3

Подобное распределение значений обусловлено факторами, повлекшими неисправности. Например, все отказы для ротора и подшипников были связаны с изменением в физических характеристиках, отклонения в которых развиваются продолжительное время (температуры, вибрации, давление). Для электродвигателя добавляются причины отказа, связанные с токами, аномальные режимы в которых развиваются крайне стремительно (от нескольких минут до нескольких часов), что не позволяет заранее диагностировать развитие серьезного дефекта с возможностью идентифицировать само состояние отказа оборудования с поправкой на несколько часов до него. Следовательно, для электроаппаратуры, качество предсказания оказалось самым низким из анализируемых технических мест [173-174].

4.4 Апробация метода прогнозирования остаточного ресурса оборудования

Для дальнейшего использования полученных данных производилась их обработка. Для каждого из 16 датчиков использовалось по 4 характеристики. Дополнительно были рассчитаны статистические показатели со скользящим окном, равным 24 часам (рисунок 4.10). В последствии в обучающие данные

были добавлены разности между медианой за час и сутки, и такие же параметры для среднего.

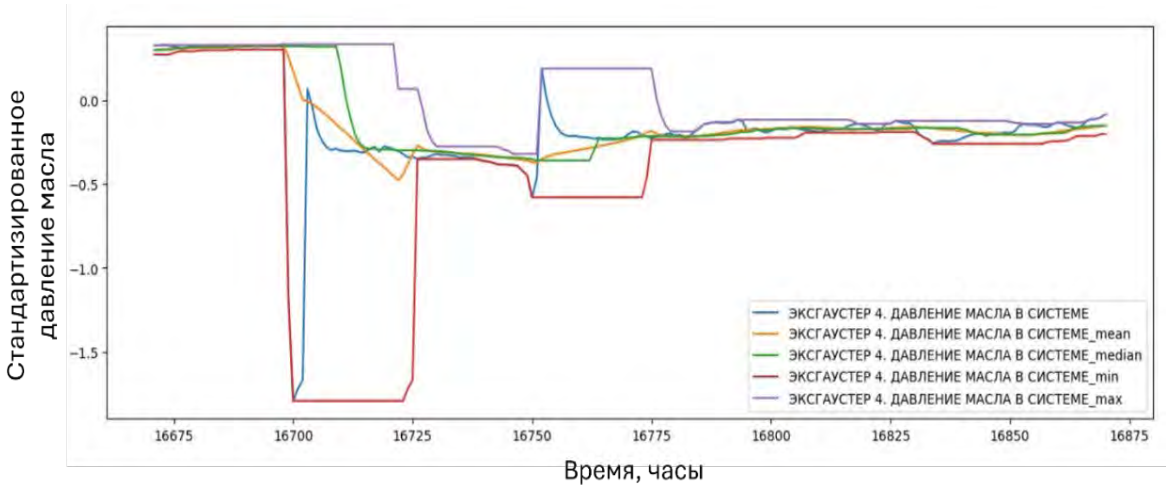


Рисунок 4.10 – Статистические характеристики, рассчитанные для давления масла в системе. Окно сдвига – 24 часа, шаг – 1 час. Показания датчика стандартизированы

Показания датчиков и данные об отказах также были преобразованы в формат, удобный для решения регрессионной задачи (предсказание RUL). Изначально из исходных данных были вырезаны фрагменты, включающие в себя интервалы от предыдущего отказа (не включительно) с отступом в 24 часа до наступления нового отказа (исключая сам момент отказа) (рисунок 4.11). Далее для каждого технического места подбирался горизонт прогнозирования: сутки, неделя, две недели, месяц (24, 168, 336, 672 часа соответственно).

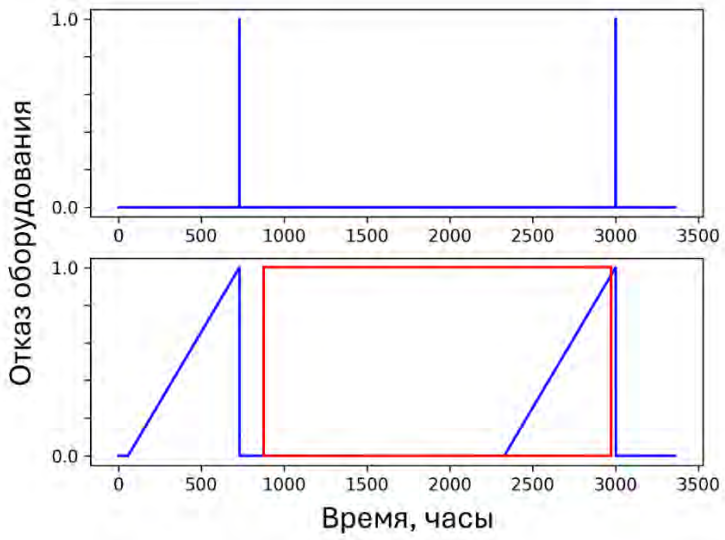


Рисунок 4.11 – Преобразование данных об отказах в новый формат. 0 – нормальная работа, 1 – отказ оборудования. Красным прямоугольником выделена область, которая в дальнейшем используется в обучении для одного из случаев отказа эксгаустера

Для этого период до отказа, равный горизонту прогнозирования размечался дробными числами от 0 (до отказа больше суток, недели и т.д.) до 1 (час наступления отказа) (рисунок 4.11). Данные значения для каждого часа рассчитывались по формуле:

$$p = t_w - t_i / t_w, \quad (4.5)$$

где t_w – размер горизонта прогнозирования в часах, t_i – время до наступления отказа в часах.

Полученные фрагменты были распределены по разным наборам в зависимости от технического места, в котором произошла критическая неисправность, повлекшая за собой простой всего эксгаустера. По итогу были собраны обучающие наборы данных для ротора, электродвигателя, подшипников и электроаппаратуры. Из каждого исходного набора был отобран валидационный (0,15 от общего набора данных) и тестовый набор (0,15) из наиболее поздних фрагментов с отказами, во избежание утечки данных из тестового набора.

В качестве алгоритма прогнозирования была выбрана нейронная сеть LSTM. Этот выбор обусловлен рядом преимуществ данной архитектуры при работе с временными рядами и возможностью анализа не одной временной точки, а целого интервала с сохранением информации о его состоянии. Так, на вход модели подавали данные показателей датчиков за 24 часа (размер массива 24x96), по итогу на выходе модель выводит показатель времени до отказа для последнего часа из последовательности. Архитектура предиктора включает в себя два LSTM-слоя, размер скрытого состояния которых равен 16 и 8 соответственно, и последующие два полносвязных слоя с размером 100 каналов и 1 соответственно. После рекуррентных слоев используется функция активации гиперболического тангенса (Tanh), после первого полносвязного слоя используется ectified linear activation unit (ReLU). После последнего полносвязного слоя функция активации не используется, в связи с тем, что там предсказывается само значение p (показатель времени до отказа). В качестве оптимизатора использовался Adam (lr = 0.0001). В качестве функции потерь

использовалась MSE (Mean Squared Error).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (4.6)$$

Для борьбы с переобучением добавили слой Dropout со значением отсева 0.5. Обучение производилось в течение 100 эпох, по результатам которого отбиралась модель с наименьшей функцией потерь на валидационной выборке.

для тестирования были собраны для ротора, электродвигателя, подшипников и электроаппаратуры. Для каждого из отобранных технических мест осуществлялось выделение нескольких вариантов горизонта прогнозирования (сутки, неделя, две недели и месяц), при этом сами показания датчиков группировались по 24-часовым интервалам. Предсказание осуществлялось для последнего часа по анализу всего фрагмента. Из исходного набора для тестирования было отобрано 15% данных, состоящих из наиболее поздних фрагментов с отказами, во избежание утечки данных из тестового набора.

Предсказание остаточного ресурса оборудования относится к задачам регрессии, т. е. необходимо предсказать не класс, а точное значение целевой метки, в данном случае долю остаточного временного ресурса от всего окна прогнозирования. Для решения подобных задач в качестве метрик качества выступают различные подходы расчета ошибки предсказания, которые также используются в качестве функций потерь для нейронных сетей (подробнее в Главе 2). Для выбора оптимальной метрики необходимо проанализировать данные на предмет выбросов и сбалансированность данных. В случае, если данные имеют большое количество выбросов и распределение целевых значений близко к нормальному или равномерному, для оценки качества предсказания модели используется метрика MAE (Средняя Абсолютная Ошибка, англ. Mean Average Error), которая рассчитывается по формуле:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (4.7)$$

Однако, если при предсказании важно минимизировать большие ошибки,

возникающие из-за узкой представленности данных определенной разновидности (такие как интервалы работы оборудования непосредственно перед отказом), оптимальнее использовать MSE (Средняя Квадратичная Ошибка, англ. Mean Squared Error). Особенностью расчета MSE является возведение ошибки в квадрат, следовательно более большие в абсолютном значении ошибки будут вносить значительный вклад в итоговое значение. Однако, в данной работе используется не сама средняя квадратичная ошибка, а ее производное, так как для последующего анализа необходимо, чтобы значение метрики было в той же размерности что и целевая переменная.

Так, для проверки обученной модели на тестовой выборке были выбраны следующие метрики качества: RMSE и WRMSE, которые рассчитываются по следующим формулам:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \tag{4.8}$$

$$WRMSE = \frac{1}{W}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}, \tag{4.9}$$

где n – количество точек, y_i – истинное значение, $\hat{y}_i$ – предсказанное значение, W – размер окна прогнозирования. Таким образом, RMSE отражает ошибку в абсолютных значениях (в нашем случае в часах), а WRMSE отражает отношение величины ошибки к размеру горизонта прогнозирования.

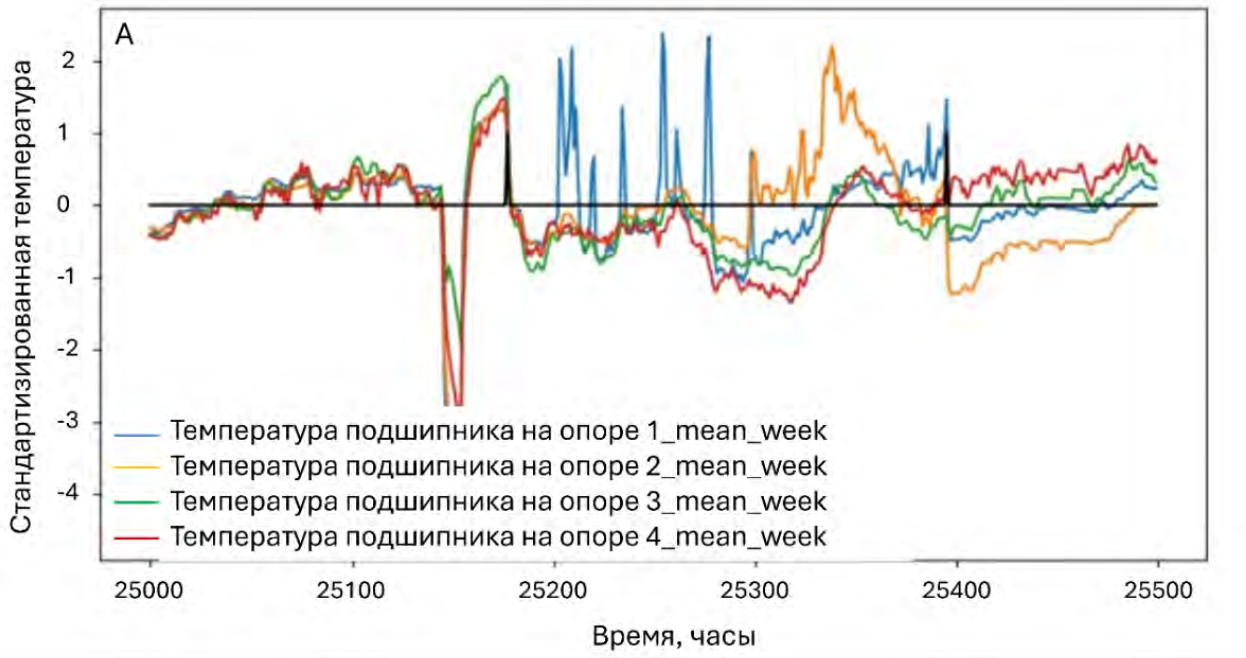
На первом этапе работы с данными осуществлялся поиск закономерностей при развитиях аномальной работы. Наиболее информативным оказался показатель разности между средним значением за час и за сутки для температуры подшипников и вибрации на опорах (рисунок 4.12). Для отказов связанных с возникновением проблем работы ротора во многих случаях в описании проблемы указываются критические изменения в температуре подшипников, вибрации на опорах и давлении масла в системе. Исходя из проведенного анализа и полученных изображений можно выявить несколько закономерностей.

Во-первых, во многих случаях отказа ротора можно увидеть

рассинхронизацию и достаточно большие скачки в изменении температуры на одной из опор (рисунок 4.12А, Б). Похожие аномалии наблюдаются и для измеренных значений вибрации на опорах (рис 4.12В, Г). В частности, наибольшие отклонения характерны для вибрации на опоре 1, которая достаточно часто находится в противофазе с изменениями вибраций на других опорах и может превышать их более чем в 5 раз.

При этом период развития данных аномалий обладает протяженностью до 300 часов и более, что позволяет более чем за неделю до отказа идентифицировать прогрессирующую неисправность в оборудовании. Однако, происходит это не во всех случаях, о чем свидетельствует первая неисправность на рисунке 4.12А. Данное явление связано с различной природой неисправностей, приводящих к отказам ротора, которые могут быть связаны как с температурой, так и с вибрацией, давлением масла, током и другими.

Учитывая такое разнообразие важных характеристик, статистических и аналитических методов недостаточно для качественного раннего прогнозирования отказов оборудования. В связи с этим было принято решение об использовании рекуррентных нейронных сетей для предсказания RUL.



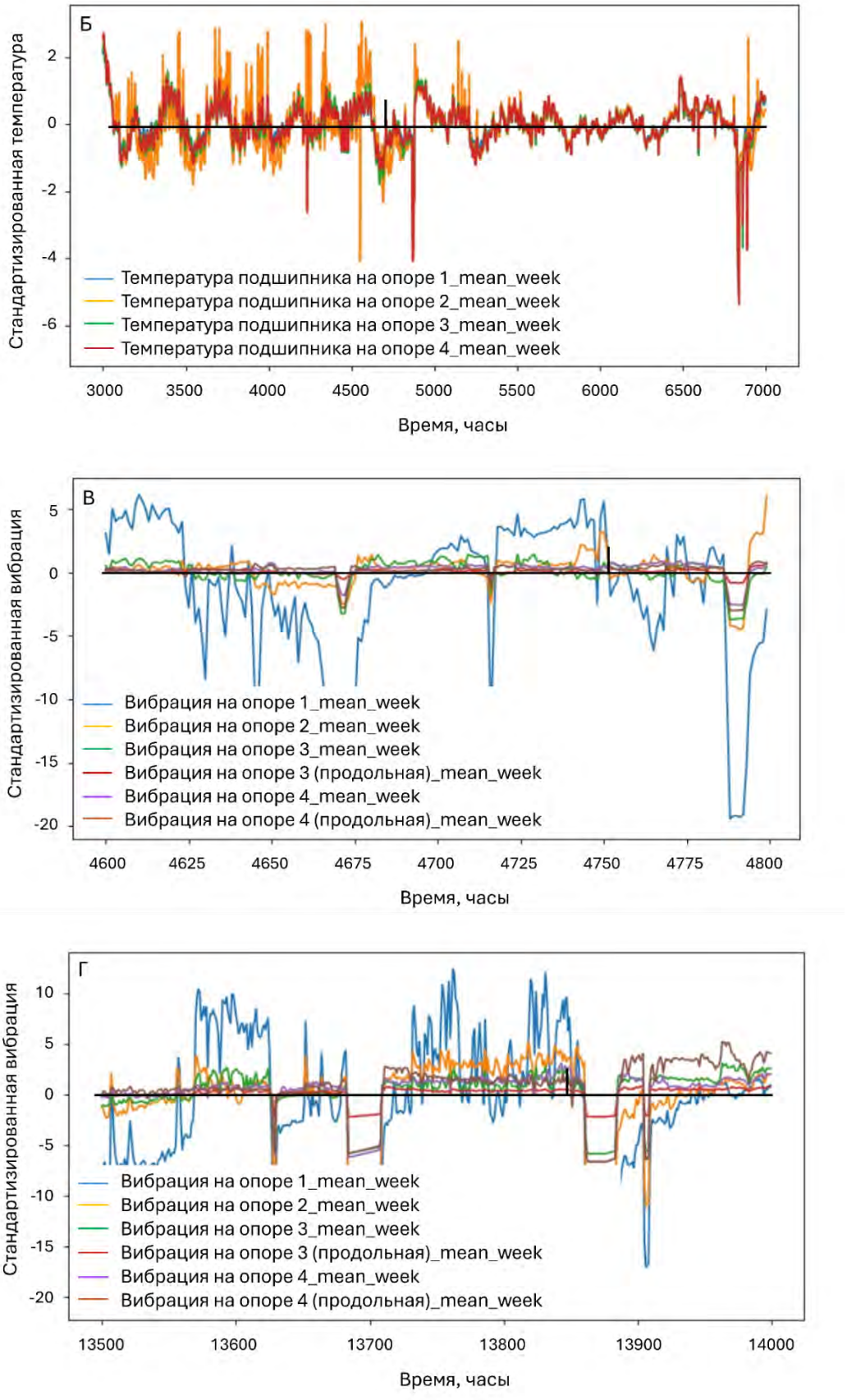


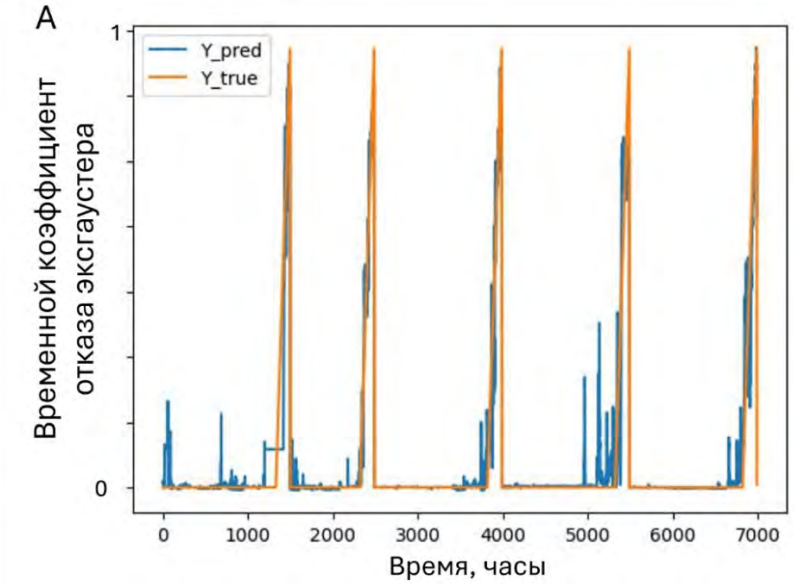
Рисунок 4.12 – Рассчитанные характеристики стандартизированных показаний датчиков температуры подшипников (А, Б) и вибрации на опорах (В, Г) перед отказами эксгаустера. Черным отмечены моменты отказа эксгаустера

Учитывая проведенный анализ изменения различных физических показателей при нормальной и аномальной работе эксгаустера, для предсказания выбрали несколько вариантов горизонта прогнозирования: сутки, неделя, две недели, месяц.

После обучения предиктора в течение 100 эпох для каждой выборки лучшая модель была проверена на тестовой выборке. Результат предсказания для ротора и подшипников с горизонтом прогнозирования, равным одной неделе показан на рисунках 4.13А и 4.13Б соответственно.

Можно заметить, что модель верно выявила все интервалы развития аномалий, повлекших за собой отказ эксгаустера. Можно заметить, что в большинстве случаев, модель идентифицировала начало появления неисправной работы более чем за 1.5–2 недели. Данное наблюдение позволяет сделать вывод о том, что в ряде определение времени до отказа ротора и подшипников возможно более чем за неделю.

Скорее всего это связано с более ранним влиянием неисправности на изменение вибраций на опорах и температуры, которые часто можно обнаружить задолго до наступления отказа. Также в ряде случаев предиктор находил неисправности, хотя они не входили в установленный горизонт прогнозирования. Это может обуславливаться обнаружением сотрудниками данных неисправностей и своевременным их устранением в процессе технических и ремонтных работ.



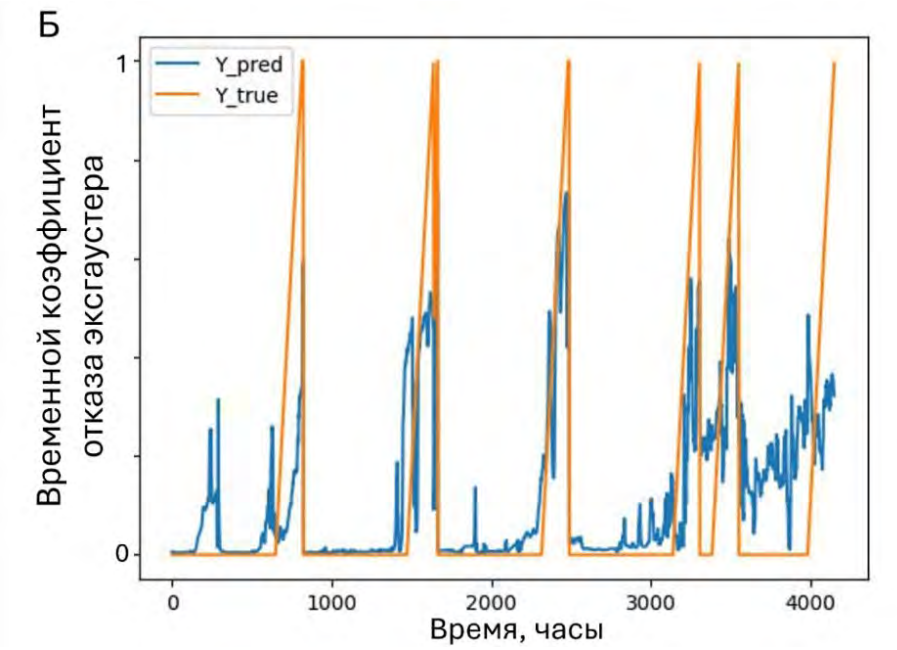


Рисунок 4.13 – Результат предсказания отказов эксгаустера, связанных с развитием неисправностей ротора (А) и подшипников (Б) на тестовой выборке с горизонтом предсказания, равным одной неделе. Y_true – истинный временной коэффициент отражающий RUL, Y_pred – коэффициент, предсказанный моделью; по вертикали 0 – до момента отказа больше 1 недели, 1 – момент отказа

Метрики качества предсказания RUL обученных моделей на тестовых выборках представлены в Таблице 4.5. Можно заметить, что наилучшее качество предсказания относительно горизонта прогнозирования характерно для ротора: WRMSE = 2.3% для прогнозирования на неделю вперед. При этом для данного технического места характерна и самая маленькая абсолютная ошибка: RMSE = 0.6 часа для горизонта равного суткам. Также достаточно высокое качество предсказания наблюдается для подшипников, при этом наилучшее соотношение соответствует предсказанию на месяц. Таким образом, RUL для подшипников при горизонте прогнозирования в месяц можно предсказать с ошибкой, равной примерно суткам, данное качество можно считать практически значимым даже при таком большом интервале предсказания. Получившиеся высокие значения метрик для этих технических мест, вероятно, обусловлены возможностью отображения неисправностей в них при анализе температур подшипников и вибраций, аномалии в которых начинают развиваться гораздо раньше возникновения отказа.

Таблица 4.5. Результаты тестирования обученной модели при разных горизонтах прогнозирования.

Горизонт прогнозирования	Техническое место	RMSE, часы	WRMSE, %
Сутки (24 часа)	Ротор	**0.6**	2.5
	Электродвигатель	7.3	30.4
	Подшипники	1.0	4.2
	Электроаппаратура	12.4	51.6
Неделя (168 часов)	Ротор	3.8	**2.3**
	Электродвигатель	19.2	**11.4**
	Подшипники	6.1	3.6
	Электроаппаратура	77.5	46.1
Две недели (366 часов)	Ротор	12.7	3.5
	Электродвигатель	44.0	12
	Подшипники	13.5	3.7
	Электроаппаратура	99.3	27.1
Месяц (672 часа)	Ротор	36.1	5.4
	Электродвигатель	98.6	14.7
	Подшипники	23.4	**3.5**
	Электроаппаратура	111.7	**16.7**

По схожим причинам не удалось получить высокого качества прогноза для электродвигателя и электроаппаратуры. Во многих случаях, отказы, связанные с данными техническими местами, можно наблюдать только при анализе токов. Неисправности, связанные с электричеством, могут развиваться крайне стремительно, что делает невозможной раннее прогнозирование возникновения отказа. В связи с этим для данных технических мест обучили модели с горизонтом прогнозирования, равным 3 часам и уменьшили временное шкалирование с часа до 5 минут. По итогу обучения удалось получить следующие метрики: для электродвигателя RMSE = 20.3 минутам,

WRMSE = 11.3, для электроаппаратуры RMSE = 43 минутам, WRMSE = 23.8. В данных условиях модель наиболее точно прогнозирует развитие неисправностей в данных технических местах, однако для практического применения ошибка остается достаточно большой.

Таким образом, с использованием разработанных моделей можно проводить исследование работы эксгаустеров в режиме реального времени, а результаты этих исследований применять для планирования работ по оптимальному техническому обслуживанию.

4.5 Тестирование автоматизированной системы

В рамках тестовой среды было проведено всестороннее тестирование работы автоматизированной системы [175].

На первом этапе тестирования была симулирована интеграция системы с различными источниками данных, включая датчики температуры, давления и вибрации, установленные на эксгаустерах. Проведена проверка, что данные корректно поступают из гипотетических систем SCADA и ERP, а также с локальных датчиков: был запущен тестовый сценарий, в котором данные с датчиков генерировались в реальном времени. Проверено, что система корректно обрабатывает данные в формате CSV и JSON. Логирование событий позволило выявить, что некоторые данные не поступали из-за неправильной конфигурации подключения к SCADA. После исправления конфигурации все данные начали поступать корректно.

```
Пример лога событий для сотрудника поддержки системы
Дата и время: 2023-10-15 14:32:10
Уровень: Ошибка
Компонент: Подсистема импорта данных
Сообщение: Не удалось получить данные от SCADA.
Описание: При попытке импорта данных из системы SCADA возникла ошибка подключения. Данные о состоянии эксгаустеров не были получены.
Причина: Неправильная конфигурация подключения к SCADA.
Решение: Провести проверку конфигурации подключения, убедиться в правильности указанных параметров (IP-адрес, порт, учетные данные).
Статус: В процессе исправления.
Ответственный: Инженер по интеграции данных (Иванов И.И.)
Дополнительная информация: Логи подключения к SCADA за последние 30 минут:
```

```
        14:30:00 - Попытка подключения к SCADA (успешно)
        14:31:05 - Запрос данных о состоянии эксгаустеров (неуспешно)
        14:31:10 - Ошибка: "Неверные учетные данные"
        14:32:00 - Повторная попытка подключения (неуспешно)
        Дата и время: 2023-10-15 14:35:45
        Уровень: Информация
        Компонент: Подсистема импорта данных
        Сообщение: Конфигурация подключения к SCADA исправлена.
        Описание: Конфигурация подключения к SCADA была проверена и исправлена. Учетные данные и
параметры подключения обновлены.
        Статус: Успешно.
        Ответственный: Инженер по интеграции данных (Иванов И.И.)
        Дополнительная информация: Повторная попытка подключения к SCADA запланирована на
14:36:00.

        Дата и время: 2023-10-15 14:36:00
        Уровень: Успех
        Компонент: Подсистема импорта данных
        Сообщение: Данные успешно получены из SCADA.
        Описание: После исправления конфигурации подключения данные о состоянии эксгаустеров были
успешно получены и импортированы в систему.
        Статус: Успешно.
        Ответственный: Инженер по интеграции данных (Иванов И.И.)
        Дополнительная информация: Общее количество полученных записей: 1500.
```

В рамках тестирования системы была проведена проверка безопасности данных. Аутентификация пользователей была протестирована с использованием тестовых учетных записей, которые имели различные уровни доступа. Это позволило убедиться, что только авторизованные сотрудники могут получать доступ к данным. Следующим этапом стало тестирование подсистемы хранения данных, использующей ClickHouse для обработки больших объемов информации. В процессе тестирования была проведена нагрузка на систему, при которой в ClickHouse было загружено несколько терабайт данных за короткий промежуток времени. Оценка скорости выполнения запросов подтвердила, что система справляется с нагрузкой, а время ответа на запросы не превышает 2 секунд. Также была протестирована функция автоматического дообучения модели. В рамках тестирования был инициирован сценарий, при котором модель должна была дообучиться на новых данных, накопленных за последние 3 месяца [176-177]. После

дообучения было проверено, что качество предсказаний улучшилось, и метрики стали выше. Тестирование подсистемы управления моделями включало проверку механизмов мониторинга работы моделей в реальном времени. Система была протестирована на предмет отслеживания производительности моделей и уведомления пользователей о снижении качества предсказаний. Для этого был создан сценарий, имитирующий ухудшение качества предсказаний, и система успешно отправила уведомления, что подтвердило корректность работы подсистемы.

На этапе тестирования подсистемы отображения данных был проверен пользовательский интерфейс, разработанный на React. Важно было убедиться, что интерфейс предоставляет доступ к необходимой информации в реальном времени. Интерфейс, отображающий данные с эксгаустеров, был протестирован на предмет мгновенного обновления графиков и диаграмм при поступлении новых данных. Также была протестирована функция фильтрации данных по времени и типу эксгаустера, что позволило пользователям легко находить нужную информацию.

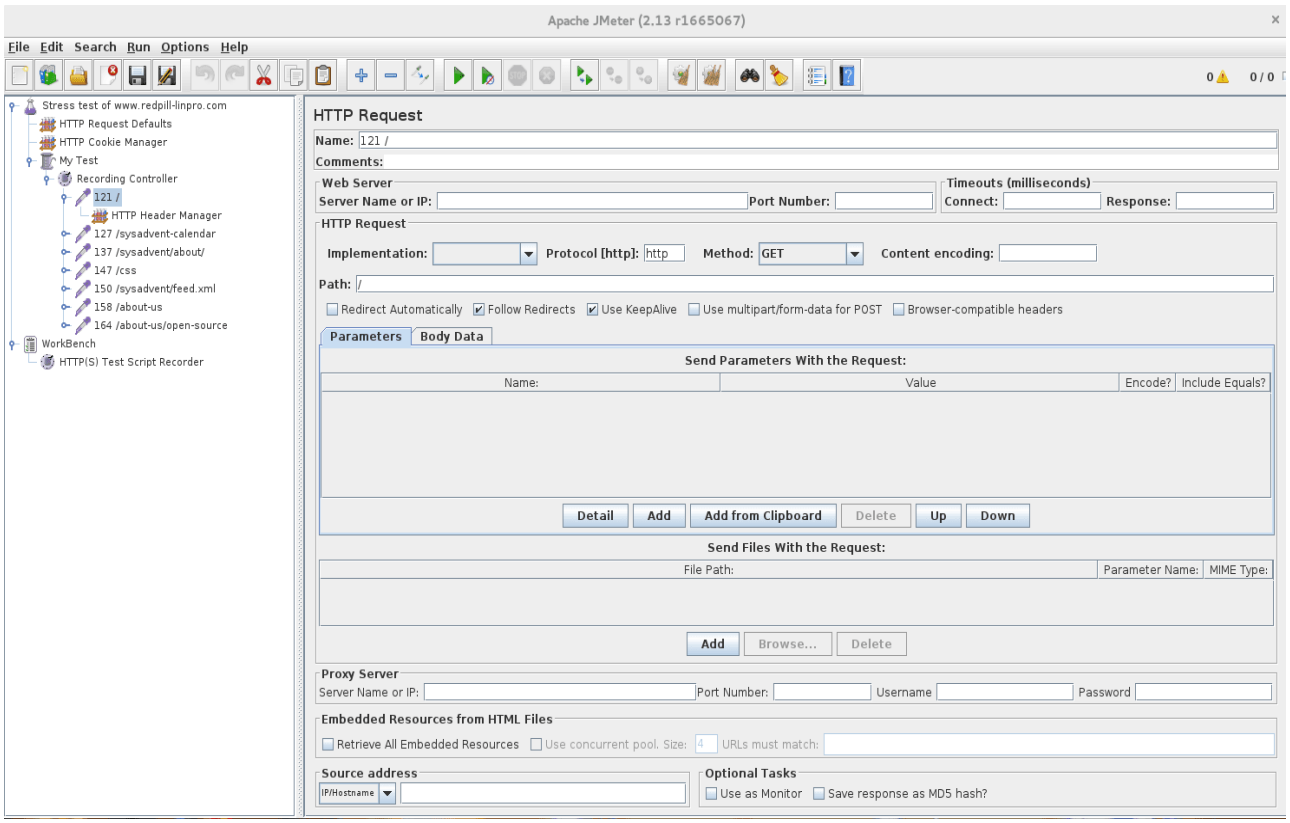


Рисунок 4.14 – Интерфейс Apache JMeter

В рамках нагрузочного тестирования была проведена серия тестов для оценки того, как система справляется с высокими нагрузками и большими объемами данных. Для этого был создан сценарий, имитирующий поступление данных от 100 датчиков одновременно. Использовались инструменты для нагрузочного тестирования, такие как JMeter - инструмент с открытым исходным кодом, предназначенный для тестирования производительности и функциональности различных приложений, включая веб-приложения, базы данных и другие, для симуляции потока данных. Результаты тестирования показали, что время отклика на запросы не превышает 3 секунд, даже при максимальной нагрузке, что подтвердило эффективность обработки данных в условиях реального времени.

Кроме того, была протестирована возможность масштабирования системы. Были добавлены дополнительные узлы в кластер, на котором работала система, и проверено, что производительность не ухудшается при увеличении числа обрабатываемых данных. Это подтвердило, что архитектура системы позволяет легко масштабироваться в ответ на растущие требования.

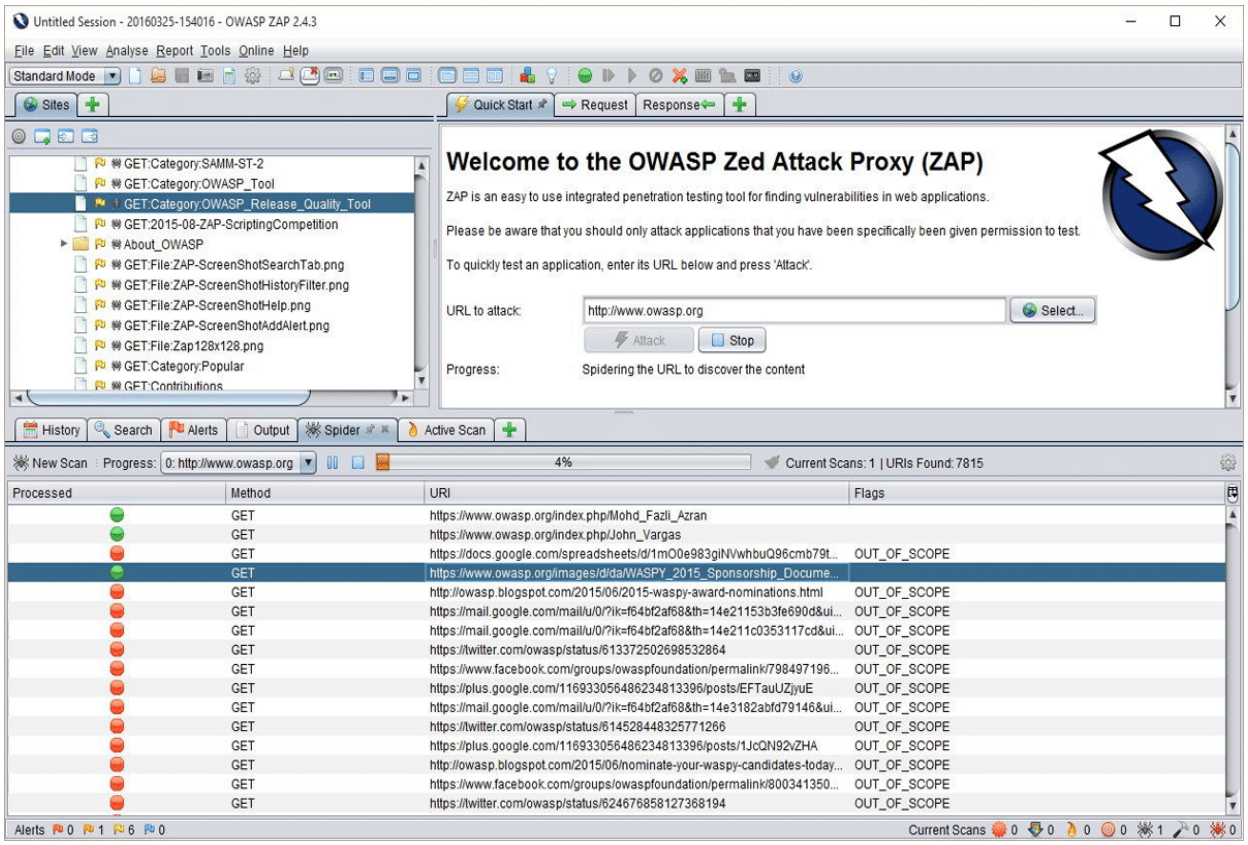


Рисунок 4.15 – Интерфейс OWASP ZAP

Тестирование безопасности системы было критически важным этапом, так как система обрабатывает чувствительные данные о состоянии оборудования и производственных процессов. Проведенные тесты на наличие уязвимостей подтвердили, что система защищена от потенциальных атак. Использовались инструменты для тестирования на проникновение, такие как OWASP ZAP - бесплатный инструмент с открытым исходным кодом, предназначенный для тестирования безопасности веб-приложений, для выявления возможных уязвимостей в веб-интерфейсе и API системы. Результаты тестирования показали, что система успешно прошла все проверки безопасности, и не было выявлено серьезных уязвимостей.

Также была проверена защита данных при передаче. Убедились, что все данные, передаваемые между клиентом и сервером, шифруются с использованием протоколов HTTPS, что предотвращает возможность перехвата данных злоумышленниками.

4.5 Выводы по четвертой главе

В результате апробации предложенных подходов к оценке и прогнозированию технического состояния эксгаустеров агломашины можно сделать следующие выводы:

1. Предобработка данных является критически важным этапом в разработке автоматизированной системы прогнозирования, так как от качества и информативности входных данных зависит успешность прогнозирования состояния оборудования и, как следствие, эффективность его эксплуатации. Предложен и реализован подход к предобработке, состоящий из 4 этапов: удаления выбросов, приведения данных к часовым интервалам, стандартизации и замены пропущенных значений и добавления новых признаков.

2. Для апробации предложенных моделей и методов были проанализированы и выбраны тестовые наборы данных индивидуально для каждой из задач, при этом соблюдалось сохранение иерархической структуры данных и репрезентативность подобранных выборок.

3. Проведена апробация метода прогнозирования неисправного состояния оборудования. Использование ансамблевого подхода с операцией «мягкого голосования» между моделями позволило значительно улучшить качество предсказания для технических мест, неисправная работа которых имеет быстрое развитие.

4. Проведена апробация метода прогнозирования отказов оборудования, показавшая высокие метрики качества, а также были выявлены закономерности, позволяющие определять развитие отказов заблаговременно.

5. Проведена апробация метода прогнозирования остаточного ресурса оборудования. Модель показала высокую точность предсказания для ротора и подшипников, с минимальными ошибками что подтверждает возможность раннего выявления критичных неисправностей в этих технических местах. Прогнозирование для электродвигателя и электроаппаратуры оказалось наиболее сложным из-за стремительного развития неисправностей, связанных с электрическими параметрами, что требует более частого мониторинга.

6. Успешно проведено тестирование работы системы в условиях, приближенных к эксплуатационным, доказавшее высокую скорость работы и отказоустойчивость системы.

ЗАКЛЮЧЕНИЕ

В результате выполненных исследований получены следующие основные научные и практические результаты:

1. Проведен анализ современного состояния исследований в области прогнозирования технического состояния промышленного оборудования, показавший высокую актуальность разработки автоматизированной системы прогнозирования технического состояния промышленного оборудования на базе методов искусственного интеллекта.

2. Разработана новая функциональная структура автоматизированной системы прогнозирования технического состояния промышленного оборудования, включающая в себя подсистемы импорта данных, хранения и обработки данных, прогнозирования, отображения данных и уникальную подсистему управления моделями, позволяющую дообучать модели с учетом динамически изменяющихся условий производственного процесса.

3. Разработан инновационный подход к анализу и обработке данных о состоянии промышленного оборудования, учитывающий особенности автоматизированных систем прогнозирования, заключающийся в обработке данных в режиме реального времени с высокой скоростью и точностью с учетом их физической природы.

4. Предложен алгоритм диагностики технических систем, основанный на ансамблевом подходе, который позволяет создавать и обучать высокоточные модели прогнозирования неисправностей и отказов промышленного оборудования.

5. Предложен и реализован новый алгоритм прогнозирования остаточного срока службы оборудования, основанный на современных архитектурах нейронных сетей, показавший высокие метрики точности для горизонта прогнозирования, равного одному месяцу.

6. Реализована функциональная структура автоматизированной системы, предоставляющей пользователю доступ к прогнозам и рекомендациям по предотвращению неисправностей и отказов, способной в

автоматическом режиме без привлечения разработчиков обучаться на накапливающемся массиве данных.

7. Проведено тестирование автоматизированной системы прогнозирования технического состояния промышленного оборудования в приближенных к реальности условиях, подтвердившее высокую эффективность её использования для решения задач контроля технического состояния эксгаустера агломашины.

Таким образом, с использованием разработанной системы прогнозирования можно проводить исследование работы эксгаустеров в режиме реального времени, а результаты этих исследований применять для планирования работ по оптимальному техническому обслуживанию и ремонту.

В текущем виде разработанное программное обеспечение может использоваться на одном из промышленных предприятий и будет показывать свою эффективность при прогнозировании неисправностей технологических узлов эксгаустера. Так как архитектура системы и модели прогнозирования разработаны с учетом специфики работы с реальными промышленными данными и не привязаны к конкретному типу оборудования, решение может эффективно масштабироваться в комплексную систему прогнозирования предприятия. Для этого достаточно провести дообучение на новых данных с сохранением базовой архитектуры алгоритма и изменением только входного слоя нейронных сетей.

СПИСОК ИСПОЛЬЗУЕМЫХ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

БД	База данных
ИИ	Искусственный интеллект
СУБД	Система управления базой данных
ТСО	Техническое состояние оборудования
ACID	Atomicity, consistency, isolation, durability (атомарность, согласованность, изолированность, долговечность)
AI	Artificial Intelligence (искусственный интеллект)
ANN	Artificial Neural Network (искусстенная нейронная сеть)
API	Application programming interface (интерфейс программирования приложения)
CNN	Convolutional neural network (свёрточная нейронная сеть)
DDM	Data-Driven Methods (методы, основанные на анализе данных)
DL	Deep Learning (глубокое обучение)
DT	Decision Trees (метод деревьев решений)
ERP	Enterprise resource planning, (планирование ресурсов предприятия)
FN	False Negative (ложно-отрицательный)
FP	False Positive (ложноположительный)
GB	Gradient Boosting (метод градиентного бустинга)
IIoT	Industrial Internet of Things (промышленный интернет вещей)
IoT	Internet of Things (интернет вещей)
JSON	JavaScript Object Notation (обозначение объектов JavaScript)
JWT	JSON Web Token (веб-токен JSON)
KBM	Knowledge-Based Methods (методы, основанные на формализованных знаниях)
KNN	k-nearest neighbor algorithm (алгоритм k-ближайших соседей)
LBFGS	Limited-memory BFGS (BFGS с ограниченной памятью)
LSTM	Long short-term memory (сеть с долгой краткосрочной памятью)
MAE	Mean absolute error (средняя абсолютная ошибка)
ML	Machine Leaning (машинное обучение)
MLP	Multilayer perceptron (многослойный персептрон)

MVCC	Multiversion concurrency control (управление параллельным доступом посредством многоверсионности)
PhM	Physic-Based methods (методы, основанные на физических моделях)
RBF	Radial Basis Function (радиальная базисная функция)
RF	Random forest (алгоритм случайного леса)
RMSE	Root mean square error (среднеквадратическая ошибка)
RNN	Recurrent neural network (рекуррентная нейронная сеть)
RUL	Remaining Useful Life (остаточный срок службы)
SCADA	Supervisory Control and Data Acquisition (диспетчерское управление и сбор данных)
SQL	Structured Query Language (язык структурированных запросов)
SVM	Support vector machine (метод опорных векторов)
TN	True Negative (истинно-отрицательный)
TP	True Positive (истинно-положительный)
WRMSE	Weighted Root Mean Squared Error (взвешенная среднеквадратическая ошибка)
XGBoost	Extreme Gradient Boosting (метод экстремального градиентного бустинга)
YAML	Yet Another Markup (язык разметки)

СПИСОК ЛИТЕРАТУРЫ

1. Сарлыбаев А. А. Разработка системы непрерывного контроля технического состояния электропечных трансформаторов: специальность 05.09.03 "Электротехнические комплексы и системы»: диссертация на соискание ученой степени кандидата технических наук. – 2016. – 189 с.

2. Шеве Г., Хюзиг С., Гумерова Г. И., Шаймиева Э. Ш. От Индустрии 3.0 к Индустрии 4.0: основные понятия, измерения и компоненты Индустрии 4.0 // Инвестиции в России. – 2019. – № 9(296). – С. 32-40.

3. Серебренников С. С. Технологический переход от Индустрии 2.0 и 3.0 к Индустрии 4.0 в промышленной отрасли / С. С. Серебренников, С. С. Харитонов // Вестник МИРБИС. – 2020. – № 4(24). – С. 67-79.

4. Управление экономическими системами различного уровня: теория и практика: Коллективная монография / И. Л. Авдеева, А. П. Андруник, А. С. Астахин [и др.]. – Орел: Среднерусский институт управления - филиал РАНХиГС, 2019. – 272 с.

5. Патент № 2649321 C1 Российская Федерация, МПК H04L 27/34. Системы и способы выбора и конфигурирования схемы модуляции и кодирования: № 2017100324: заявл. 06.01.2015: опубл. 02.04.2018 / Х. Х. Квон, А. Давыдов, С. Хан [и др.]; заявитель ИНТЕЛ АйПи КОРПОРЕЙШН.

6. Brailko A. A. Adaptive information management system of dynamic monitoring of actual water content in jet fuel in technological processes of aviation fuel supply / A. A. Brailko, V. M. Samoylenko, N. A. Druzhinin, L. A. Druzhinin // Civil Aviation High Technologies. – 2022. – Vol. 25, No. 2. – P. 20-29.

7. Лучкин Н. А. Проектирование системы автоматизированного мониторинга технологических процессов и диспетчерского управления объектами газотранспортной системы / Н. А. Лучкин, А. Г. Янишевская, Ж. В. Голикова // Динамика систем, механизмов и машин. – 2018. – Т. 6, № 2. – С. 37-43.

8. Игнатьев С. А. Мониторинг технологического процесса как элемент системы управления качеством продукции / С. А. Игнатьев, В. В. Горбунов, А. А. Игнатьев. – Саратов: Саратовский гос. технический ун-т, 2009. – 160 с.

9. Arzumanov R. M. Algorithm for managing the level of economic security of an enterprise / R. M. Arzumanov, O. M. Klimakhina, G. D. Bezkrovnaya [et al.] // Bulletin of the Kerch State Marine Technological University. – 2021. – No. 3. – P. 165-179.

10. Юмагузин У. Ф. Прогнозирование остаточного ресурса оборудования предприятий нефтегазовой отрасли // Фундаментальные исследования. – 2014. – № 2(3). – С. 277–280.

11. Fu C. Predictive maintenance in Intelligent-Control-Maintenance-Management system for hydroelectric generating unit / C. Fu, L. Ye, Y. Liu, R. Yu, B. Iung, Y. Cheng, Y. Zeng // IEEE transactions on energy conversion. – 2004. – Vol. 19, № 1. – P. 179-186.

12. Гунина И.А. Прорывное технологическое развитие промышленных комплексов в условиях цифровой трансформации / И. А. Гунина, Е. В. Шкарупета, В. В. Решетов // Инновационные кластеры цифровой экономики: теория и практика. – 2018. – С. 535–554.

13. Мезина Т. В. Классирование Индустрии 4.0 в технологическом процессе / Т. В. Мезина // Вектор экономики. – 2018. – № 6. – С. 1–31.

14. Summary of the 2018 Department of defense artificial intelligence strategy. Harnessing AI to advance our security and prosperity [Электронный ресурс]. Режим доступа: http://www.defense-aerospace.com/articles-view/reports/2/199929/pentagon-releases-artificial- intelligence-strategy.html (дата обращения: 13.11.2024).

15. Nguyen K. A. Multi-level predictive maintenance for multi-component systems / K. A. Nguyen, A. Grall // Reliability engineering & system safety. – 2015. – Vol. 144. – P. 83-94.

16. Petrov S. A. Problem-dialogue technology of forming

environmental competence among specialists for work in the Arctic and subarctic conditions of the Russian Federation / S. A. Petrov, N. L. Mamaeva, M. V. Narushko // Periodico Tche Quimica. – 2019. – Vol. 16, No. 32. – P. 362-374.

17. Первый в России серьезный тендер: «Аэрофлот» будет чинить самолеты с помощью больших данных [Электронный ресурс]. – Режим доступа: https://www.cnews.ru/news/top/2017-0529_aeroflot_zajmetsya_diagnostikoj_samolet (Дата обращения: 13.11.2024).

18. Ran Y. A Survey of Predictive Maintenance: Systems, Purposes and Approaches / Y. Ran, X. Zhou, P. Lin, Y. Wen, R. Deng // IEEE communications surveys & tutorials. – 2019. – P. 1-36.

19. Равин А. А. Инженерные методы прогнозирования остаточного ресурса оборудования / А. А. Равин, О. В. Хруцкий // Вестник АГТУ. Серия: Морская техника и технология. – 2018. – № 1. – С. 33-47.

20. Шаханов Н.И. Прогнозирование отказов оборудования в условиях малого количества поломок / Н. И. Шаханов, И. А. Варфоломеев, Е. В. Ершов, О. В. Юдина // Вестник Череповецкого государственного университета. – 2016. – № 6 (75). – С. 36-40.

21. Zhang L. A review of fault prognostics in condition-based maintenance / L. Zhang, X. Li, J. Yu. // Proc. of SPIE. – 2006. – P. 635-752.

22. Ткаченко М. Г. Прогнозирование оставшегося времени безаварийной работы нефтегазодобывающего оборудования с применением технологии искусственных нейронных сетей // М. Г. Ткаченко // Электронный научный журнал «Инженерный вестник Дона». – 2014. – № 4(1). – С. 1-11.

23. Elattar H. M. Prognostics: a literature review / H. M. Elattar, H. K. Elminir, A. M. Riad // Complex & Intelleligent Systems. – 2016. – № 2(2). – P. 125-154.

24. Estrada R. F., Starr E. A. 50 Years of Acoustical Signal Processing for Detection: Coping with the Digital Revolution / R. F. Estrada, E. A. Starr // IEEE Annals of the History of Computing. – 2005. – Vol. 27. – P. 65-

78.

25. Шайхутдинов Д. В. Методы мониторинга и диагностики динамических сложных технических систем на базе средств имитационного моделирования / Д. В. Шайхутдинов // Современные наукоемкие технологии. – 2018. – № 11(1). – С. 146-153.

26. Bashirova E. M. Application of Fuzzy Logic in Hydrogen-Containing Gas Purification Plant for Reactive Power Compensation / E. M. Bashirova, D. E. Matveev, A. S. Khismatullin, I. V. Prahov // Components of Scientific and Technological Progress. – 2022. – No. 3(69). – P. 5-8.

27. Рябов А. А. Обзор существующих методов оценки остаточного ресурса оборудования нефтегазопереработки/ А.А. Рябов //Электронный научный журнал «Нефтегазовое дело». – 2016. – № 1. – С. 198-220.

28. Chen T. XGBoost: A Scalable Tree Boosting System / T. Chen, C. Guestrin // Proc.of the Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. – 2016. – P. 785-794.

29. Дьяконов А. Г. Выявление аномалий в работе механизмов методами машинного обучения / А. Г. Дьяконов, А. М. Головина // Аналитика и управление данными в областях с интенсивным использованием данных. – 2017. – С. 389–396.

30. Mayer-Schönberger V. A Revolution That Will Transform How We Live, Work, and Think / V. Mayer-Schönberger, K. Cukier. – London, 2013. – 242 p.

31. Ying Peng. Current status of machine prognostics in condition-based maintenance: a review / Peng Ying, Dong Ming, Jian Zuo Ming // The International Journal of Advanced Manufacturing Technology. – 2010. – Vol 50, № 1. – P. 297-313.

32. Lu C. Y. Face recognition via weighted sparse representation / C. Y. Lu, H. Min, J. Gui, L. Zhu, Y. K. Lei // Journal of Visual Communication and Image Representation. – 2013. – Vol 24(2). – P. 111-116.

33. Huang P. S. Learning deep structured semantic models for web search using clickthrough data / P. S. Huang, X. He, J. Gao, L. Deng, A. Acero, L. Heck // In Proceedings of the 22nd ACM international conference on Information & Knowledge Management. – 2013. – P. 2333-2338.

34. Kourou K. Machine learning applications in cancer prognosis and prediction / K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis, D. I. Fotiadis // Computational and structural biotechnology journal. – 2015. – Vol 13. – P. 8-17.

35. Alphago: Mastering the ancient game of go with machine [Электронный ресурс]. – Режим доступа:https://ai.googleblog.com/2016/01/alphago-mastering-ancient-game-of-go.html (Дата обращения: 13.11.2024).

36. Клячкин В. Н. Прогнозирование состояния технического объекта с применением меттодов машинного обучения / В. Н. Клячкин, Д. А. Жуков // Программные продукты и системы. – 2019. –Т. 32, № 2. – С. 244-250.

37. Racha K. Direct Remaining Useful Life Estimation Based on Support Vector Regression / K. Racha, C. Brigitte, M. Simon, L. Emna, F. Farhat, Z. Noureddine // IEEE Transactions on Industrial Electronics. – 2017. – Vol 64. – P. 2276-2285.

38. Calabrese M. SOPHIA: An Event-Based IoT and Machine Learning Architecture for Predictive Maintenance in Industry 4.0 / M. Calabrese, M. Cimmino, F. Fiume, M. Manfrin, L. Romeo, et al // Information. – 2020. – Vol 11(4). – P. 1-17.

39. Kundu P. An ensemble decision tree methodology for remaining useful life prediction of spur gears under natural pitting progression / P. Kundu, A. K. Darpe, M. S. Kulkarni // Structural Health Monitoring. – 2020. – Vol. 19(3). – P. 854-872.

40. Ying Peng. Current status of machine prognostics in condition-based maintenance: a review / Peng Ying, Dong Ming, Jian Zuo Ming // The

International Journal of Advanced Manufacturing Technology. – 2010. – Vol 50, № 1. – P. 297-313.

41. Bengio Y. Learning deep architectures for AI / Y. Bengio // Foundumental Trends Machine Learning. – 2009. Vol. 2. – P. 1-127.

42. Blei D. M., Ng A. Y., Jordan M. J. Latent Dirichlet allocation / D. M. Blei, A. Y. Ng, M. J. Jordan // In Journal of Machine Learning Research. – 2003. – Vol. 3. – P. 993-1022.

43. Hutchinson B., Deng L., Yu D. Tensor deep stacking networks / B. Hutchinson, L. Deng, D. Yu // In IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2013. – Vol. 35. – P. 1944-1957.

44. Mesnil G., He X., Deng L., Bengio Y. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding / G. Mesnil , X. He, L. Deng, Y. Bengio // Interspeech. – 2013.

45. Huang H. Z. Support vector machine-based estimation of remaining useful life: current research status and future trends / H. Z. Huang, H. K. Wang, Y. F. Li, L. Zhang, Z. Liu // Journal of Mechanical Science and Technology. – 2015. – Vol 29 (1). – P. 151-163.

46. Susto G. A. Machine learning for predictive maintenance: A multiple classifier approach / G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, A. Beghi // IEEE Transactions on Industrial Informatics. – 2014. – Vol. 11(3). – P. 812-820.

47. Nieto P. G. Hybrid PSO–SVM-based method for forecasting of the remaining useful life for aircraft engines and evaluation of its reliability / P. G. Nieto, E. Garcia-Gonzalo, F. S. Lasheras, F.J. De Cos Juez // Reliability Engineering & System Safety. – 2015. – Vol. 138. – P. 219-231.

48. Yan M. Bearing remaining useful life prediction using support vector machine and hybrid degradation tracking model / M. Yan, X. Wang, B. Wang, M. Chang, I. Muhammad // ISA transactions. – 2020. – Vol. 98. – P. 471-482.

49. Zhang L. Review of remaining useful life prediction using support vector machine for engineering assets / L. Zhang, Z. Liu, D. Luo // 2013 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE). IEEE. – 2013. – P.1793-1799.

50. Ng S. S. A naive Bayes model for robust remaining useful life prediction of lithium-ion battery / S. S. Ng, Y. Xing, K. L. Tsui // Applied Energy. – 2014. Vol 118. – P.114-123.

51. Patil S. Remaining Useful Life (RUL) Prediction of Rolling Element Bearing Using Random Forest and Gradient Boosting Technique / S. Patil, A. Patil, V. Handikherkar, S. Desai, V. M. Phalle, F. S. Kazi // In ASME 2018 International Mechanical Engineering Congress and Exposition. – 2018. – P. 1-7.

52. Шаханов Н. И. Прогнозирование отказов оборудования на основе алгоритмов машинного обучения / Н. И. Шаханов, В. М. Осколков, И. А. Варфоломеев, О. В. Юдина // Вестник научных конференций. ООО Консалтинговая компания Юком. – 2016. – С. 315-317.

53. Praveenkumar T. Fault diagnosis of automobile gearbox based on machine learning techniques / T. Praveenkumar, M. Saimurugan, P. Krishnakumar, K. I. Ramachandran // Procedia Engineering. – 2014. – Vol 97. – P. 2092-2098.

54. Shao Z. A new electricity price prediction strategy using mutual informationbased SVM- RFE classification / Z. Shao, S. Yang, F. Gao, K. Zhou, P. Lin // Renewable and Sustainable Energy Reviews. – 2017. – Vol. 70. – P. 330-341.

55. Manning C. D. An Introduction to Information Retrieval / C. D. Manning, P. Raghavan. – Cambridge University Press, 2009. – 581 p.

56. Rahman S. Exploring collaborative filtering through K-Nearest Neighbors and Non-Negative Matrix Factorization / S. Rahman // Modern Innovations, Systems and Technologies. – 2024. – Vol. 4. – No. 2. – P. 201-211.

57. Cortes C., Vapnik V. "Support Vector Networks". Machine

learning.20(3): 273-297.

58. Sexton T. Hybrid datafication of maintenance logs from AI-assisted human tags / T. Sexton, M. P. Brundage, M. Hoffman, K. C. Morris // 2017 IEEE International Conference on Big Data (Big Data), Boston, MA. – 2017. – P. 1769-1777.

59. Китова О. В. Метод машин опорных векторов для прогнозирования показателей инвестиций / О. В. Китова, И. Б. Колмаков, И. А. Пеньков // Экономика, Статистика и Информатика. – 2016. – № 4. – С. 1-4.

60. Dhalmahapatra K. Decision support system for safety improvement: an approach using multiple correspondence analysis, t-SNE algorithm and K-means clustering / K. Dhalmahapatra, R. Shingade, H. Mahajan, A. Verma, J. Maiti // Computers & Industrial Engineering. – 2019. – Vol. 128. – P. 277-289.

61. Biau G. A random forest guided tour / G. Biau, E. Scornet // TEST 25. – 2016. – P. 197-227.

62. LeCun Y., Boser B., Denker J. S., Henderson D., Howard R. E., Hubbard W., Jackel L. D. Backpropagation Applied to Handwritten Zip Code Recognition / Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel // Neural Computation. – 1989. – Vol. 1. – No. 4. – P. 541-551.

63. Морозов А. А. Состояние и перспективы нейросетевого моделирования СППР в сложных социотехнических системах / А. А. Морозов, В. П. Клименко, А. Л. Ляхов, С. П. Алешин // Математические машины и системы. – 2010. – № 1. – С. 127-149.

64. Сай В. К. Модели и методы проактивной поддержки принятия решений при управлении техническим состоянием оборудования: специальность 05.13.01 "Системный анализ, управление и обработка информации (по отраслям)»: диссертация на соискание ученой степени кандидата технических наук. – 2020. – 152.

65. Goodfellow I. Deep Learning / I. Goodfellow, Y. Bengio, A. Courville. – Massachusetts, USA: MIT press, 2016. – 433 p.

66. Ciresan D. C. Simple Neural Nets for Handwritten Digit Recognition / D.C. Ciresan, U. Meier, L. M. Gambardella, J. Deep. Schmidhuber // Neural Computation. – 2010. – Vol 22, № 12. – P. 3207-3220.

67. He K. Deep Residual Learning for Image Recognition / K. He, X. Zhang, S. Ren // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). – 2016. – P. 770-778

68. Lv Y. Traffic Flow Prediction with Big Data: A Deep Learning Approach / Y. Lv, Y. Duan, W. Kang, Z. Li, F. Wang // IEEE Transactions on Intelligent Transportation Systems. – 2015. – Vol. 16. – No. 2. – P. 865-873.

69. Chen Q. Learning Deep Representation from Big and Heterogeneous Data for Traffic Accident Inference / Q. Chen, X. Song, H. Yamada // Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI-16). — 2016. — P. 338-344.

70. Антипов С. Г. Методы диагностики динамических объектов на основе анализа временных рядов / С. Г. Антипов, В. Н. Вагин, М. В. Фомина // ИТНОУ: информационные технологии в науке, образовании и управлении. – 2017. – № 2 (2). – С. 3-12.

71. Мыльников Л. А. Использование динамических предиктивных моделей для управления техническими системами с инертностью / Л. А. Мыльников, Н. А. Гергель, А. В. Кычкин, Б. Краузе // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2018. – № 26. – С. 77-91.

72. Тоноян С. А. Прогнозирование технического состояния электронных систем с адаптивными параметрическими моделями / С. А. Тоноян, А. В. Балдин, Д. В. Елисеев // Вестник Московского государственного технического университета им. НЭ Баумана. Серия «Приборостроение». – 2016. – № 6. – С. 115-125.

73. Тимофеев А.В. Идентификация стадии деградации оборудования в системах сервисного обслуживания превентивного типа / А. В. Тимофеев, В.М. Денисов // Научно- технический вестник информационных технологий, механики и оптики. – 2019. – № (6). – С. 1095-1104.

74. Kruppa J., Holste T., König I. R. Probability estimation with machine learning methods for dichotomous and multicategory outcome: Applications / J. Kruppa, T. Holste, I. R. König [et al.] // Biometrical Journal. – 2014. – Vol. 56, No. 4. – P. 564-583.

75. Kruppa J., Schwarz A., Arminger G., Ziegler A. Consumer credit risk:Individual probability estimates using machine learning / J. Kruppa, A. Schwarz, G. Arminger, A. Ziegler // Expert Systems with Applications. – 2013. – Vol. 40. – P. 5125-5131.

76. Denil M., Matheson D., de Freitas N. Narrowing the gap: Randomforests in theory and in practice / M. Denil, D. Matheson, N. de Freitas // Thirty-first International Conference on Machine Learning. – 2014. – Vol. 32. – P. 1-9.

77. Бабокин Г. И. Применение нейронных сетей для диагностики электромеханических систем / Г. И. Бабокин, Д. М. Шпрехер // Горный информационно-аналитический бюллетень (научно-технический журнал). – 2011. – № 4. – С. 132-139.

78. Malhotra P. Multi-Sensor Prognostics using an Unsupervised Health Index based on LSTM Encoder-Decoder / P. Malhotra, T. V. Vishnu // Anomaly Detection Workshop at 33rd International Conference on Machine Learning. – 2016. – P. 1-10.

79. Демидова Л. А. Применение рекуррентных нейронных сетей в задаче классификации отказов работы сложных технических систем в рамках проактивного технического обслуживания / Л. А. Демидова, Д. В. Марчев // Вестник рязанского государственного радиотехнического университета. – 2019. – № 69. – С.135-148.

80. Горбунов С. В. Система поддержки принятия управленческих решений в чрезвычайных ситуациях на основе прецедентного подхода / С. В. Горбунов, С. А. Петелин // Стратегия гражданской защиты: проблемы и исследования. – 2018. – Т. 8, № 1(14). – С. 74-87.

81. Bulej V., Stoianovici G.V., Poppeová V. Material Flow Improvement Automated Assembly Lines Using Lean logistics / V. Bulej, G.V. Stoianovici, V. Poppeová // 22nd International DAAAM Symposium Intelligent Manufacturing and Automation. – 2011. – P. 0253-0254.

82. Fedoseev S.A., Stolbov V.Y., Gitman M.B., Pustovoyt K.S. Improving the quality of the industrial enterprise management based on the network-centric approach / S.A. Fedoseev, V.Y. Stolbov, M.B. Gitman, K.S. Pustovoyt // R-Economy. – 2015. – Vol. 1. – P. 608-617.

83. Konovalov P., Bulavin Y., Volkov I. Control of the Sand Feeding System Operation as a Reserve for Improving the Adhesion Properties of the Traction Rolling Stock / P. Konovalov, Y. Bulavin, I. Volkov // Networked Control Systems for Connected and Automated Vehicles. – 2023. – Vol. 509. – P. 95-104.

84. Kronis K., Uhanova M. Performance Comparison of Java EE and ASP.NET Core Technologies for Web API Development / K. Kronis, M. Uhanova // Applied Computer Systems. – 2018. – Vol. 23. – P. 37-44.

85. Kuric I., Klačková I., Domnina K., Stenchlák V., Ságа M. Implementation of Predictive Models in Industrial Machines with Proposed Automatic Adaptation Algorithm / I. Kuric, I. Klačková, K. Domnina, V. Stenchlák, M. Ságа //Applied Sciences. – 2022. – Vol. 12. – P. 1853.

86. Tziolas T., Papageorgiou K., Theodosiou T., Papageorgiou E., Mastos T., Papadopoulos A. Autoencoders for Anomaly Detection in an Industrial Multivariate Time Series Dataset / T. Tziolas, K. Papageorgiou, T. Theodosiou, E. Papageorgiou, T. Mastos, A. Papadopoulos // Engineering Proceedings. – 2022. – Vol. 18. – P. 23.

87. Ságа M., Blatnická M., Blatnický M., Dižo J., Gerlici J. Research

of the Fatigue Life of Welded Joints of High Strength Steel S960 QL Created Using Laser and Electron Beams / M. Sága, M. Blatnická, M. Blatnický, J. Dižo, J. Gerlici //Materials. – 2020. – Vol. 13. – P. 2539.

88. Burduk A., Kochanska J. Wiecek D. Risk assessment of horizontal transport system in a copper mine / A. Burduk, J. Kochańska, D. Więcek // Acta Montanistica Slovaca. – 2021. – Vol. 26. – No. 2. – P. 303-314.

89. Kelemen M., Virgala I., Lipták T., Miková L., Filakovský F., Bulej V. A Novel Approach for an Inverse Kinematics Solution of a Redundant Manipulator / M. Kelemen, I. Virgala, T. Lipták, L. Miková, F. Filakovský, V. Bulej // Applied Sciences. – 2018. – Vol. 8. – P. 2229.

90. Экспертная система управления процессом спекания агломерата / В. И. Соловьев, Е. А. Павлова, В. М. Сухинин, А. П. Скороходов // Черные металлы. – 2006. – № 3. – С. 11-16.

91. Влияние гибридного совмещения методов интенсификации агломерационного процесса на показатели спекания и качество агломерата / С. Е. Сулименко, Н. В. Игнатов, В. В. Бочка [и др.] // Известия высших учебных заведений. Черная металлургия. – 2011. – № 12. – С. 3-6.

92. Незамаев С. В. Анализ технологических параметров обезвреживания газовых выбросов химико-энерготехнологического процесса спекания фосфоритового агломерата / С. В. Незамаев, А. А. Быков, В. И. Бобков // Вода: химия и экология. – 2024. – № 4. – С. 64-69.

93. Оптимизация технологических параметров процесса спекания на основе экспериментально-статистических моделей горячей прочности и восстановимости агломерата / С. Н. Хреева, В. Ю. Кучин, Т. В. Деткова, А. А. Елисеев // Черная металлургия. Бюллетень научно-технической и экономической информации. – 2022. – Т. 78, № 5. – С. 389-394.

94. Овчинникова Е. В. Особенности поведения MgO в процессе спекания агломератов с использованием бакальских сидеритов / Е. В. Овчинникова, А. Н. Шаповалов, В. Б. Горбунов // Черная металлургия. Бюллетень научно-технической и экономической информации. – 2016. – №

11(1403). – С. 30-34.

95. Ширяева Е. В. Влияние добавок низкощелочного красного шлама на технологические параметры процесса спекания агломерационной шихты и микроструктуру спеченного агломерата / Е. В. Ширяева // Актуальные проблемы науки на современном этапе развития: Сборник статей Международной научно-практической конференции, Екатеринбург, 18 ноября 2015 года / Ответственный редактор: Сукиасян Асатур Альбертович. Том 2. – Екатеринбург: Общество с ограниченной ответственностью "ОМЕГА САЙНС", 2015. – С. 139-141.

96. Юрьев Б. П. Влияние физико-химических процессов спекания на качество агломерата / Б. П. Юрьев, В. А. Дудко // Сталь. – 2024. – № 2. – С. 2-8.

97. Одинцов А. А. Влияние гранулометрического состава твердого топлива при двухслойной загрузке шихты на показатели процесса спекания и качество агломерата / А. А. Одинцов, В. А. Долинский // Известия высших учебных заведений. Черная металлургия. – 2011. – № 5. – С. 7-12.

98. Крахт Л. Н. Повышение эффективности производства железорудного агломерата на основе моделирования процессов спекания шихты / Л. Н. Крахт, В. О. Казарцев // Концепции современного образования: новации в системе научного знания: Сборник научных трудов. – Казань: ООО "СитИвент", 2020. – С. 249-257.

99. Юсфин Ю. С. Влияние структурных особенностей кристаллов магнетита на процесс спекания агломератов / Ю. С. Юсфин, Т. Я. Малышева // Известия высших учебных заведений. Черная металлургия. – 2015. – Т. 58, № 7. – С. 473-478.

100. Мных А. С. Исследование влияния теплового режима процесса спекания на качественные показатели агломерата / А. С. Мных // Энергосбережение. Энергетика. Энергоаудит. – 2015. – № 4(135). – С. 46-51.

101. Семакова В. Б. Исследование процесса спекания агломерата с

добавкой в шихту калиброванного возврата / В. Б. Семакова, Е. И. Пилюгин, В. В. Безруков // Вестник Приазовского государственного технического университета. Серия: Технические науки. – 2011. – № 23. – С. 40-45.

102. Багаутдинов И. З. Процесс спекания агломерата на агломашине / И. З. Багаутдинов, А. А. Галяутдинов // Форум молодых ученых. – 2017. – № 9(13). – С. 64-67.

103. Кафаров В. В. Методы кибернетики в химии и химической технологии: учебник для вузов по специальности "Основные процессы химических производств и химическая кибернетика" / В. В. Кафаров. - 4-е изд., перераб. и доп. - Москва: Химия, 1985. - 448 с.

104. Мешалкин В.П. Основы интенсификации и ресурсоэнерго эффективности химико-технологических систем. - Смоленск: Универсум, 2021. — 999 с.: ил.

105. Егоров А.Ф. Интегрированные системы управления химическими производствами: учеб. Пособие / А. Ф. Егоров. – М.: РХТУ им. Д. И. Менделеева, 2020. – 200 с.

106. Егоров А. Ф. Интегрированные автоматизированные системы управления химическими производствами и предприятиями: учебное пособие для вузов / А. Ф. Егоров. Москва: Издательство Юрайт, 2021. – 248 с.

107. Галушкин А. И., Глебов М. Б. Применение нейронных сетей в химии и химической технологии / А. И. Галушкин, М. Б. Глебов // Нейрокомпьютеры. – 2003. – № 2. – С. 1-50.

108. Волошин А. А., Лебедев А. А., Благоразумов Д. О., Вальгер В. А. Формализация функциональных требований к релейной защите и автоматике цифровой подстанции / А. А. Волошин, А. А. Лебедев, Д. О. Благоразумов, В. А. Вальгер // Вестник Московского энергетического института. Вестник МЭИ. – 2022. – № 5. – С. 30-38.

109. Ганджа Т. В. Развитие метода компонентных цепей для реализации комплекса программ моделирования химико-технологических

систем: специальность 05.13.18 "Математическое моделирование, численные методы и комплексы программ»: диссертация на соискание ученой степени доктора технических наук / Ганджа Тарас Викторович. – Томск, 2017. – 457 с.

110. Лавров В. В. Автоматизированная система анализа и прогнозирования производственных ситуаций доменного цеха : специальность 05.13.06 "Автоматизация и управление технологическими процессами и производствами (по отраслям)" : диссертация на соискание ученой степени доктора технических наук / Лавров Владислав Васильевич. – Екатеринбург, 2013. – 480 с.

111. Дьяков А.Ф., Волошин А.А., Жуков А.В., Нудельман Г.С. Применение оптимизационных методов при создании функционально интегрированных систем релейной защиты и автоматики // Релейщик. 2016. № 1. С. 24-31.

112. Борисов В. Г. О построении интегрированных систем навигации, управления и обучения операторов на базе методов интеллектуального управления / В. Г. Борисов, С. К. Данилова, В. О. Чинакал // XII Санкт-Петербургская международная конференция по интегрированным навигационным системам. – 2005. – С. 48-51.

113. Малыгин И. Г. Метод модификационного управления стоимостью жизненного цикла противопожарных технических систем / И. Г. Малыгин, О. Г. Трудов // Пожаровзрывобезопасность. – 2016. – Т. 25, № 5. – С. 5-17.

114. Голубов В. В. Автоматизация стыковки автономных мобильных роботов на основе развития метода поисковых случайных деревьев со встречным ростом / В. В. Голубов, С. В. Манько // Russian Technological Journal. – 2024. – Т. 12, № 1. – С. 7-14.

115. Конов К. И. Алгоритм интегрирования с выбором переменной интегрирования на каждом шаге / К. И. Конов // Инновационные, информационные и коммуникационные технологии. –

2017. – № 1. – С. 621-626.

116. Шуленин В. П. Робастные альтернативы стандартному отклонению при обработке данных физических экспериментов / В. П. Шуленин // Известия вузов. Физика. – 2016. – Т. 59, № 6. – С. 62-69.

117. Борбаць Н. М. Точность оценок индексов воспроизводимости процесса с учетом числа эффективных степеней свободы оценки стандартного отклонения / Н. М. Борбаць, Т. В. Школина // Справочник. Инженерный журнал. – 2023. – № 2(311). – С. 33-42.

118. Золотарева Т. А. Линейная связь стандартного отклонения ошибки вычисления коэффициентов корреляции с объемом выборки в логарифмических координатах / Т. А. Золотарева, С. В. Качалин, А. С. Боровский // Безопасность информационных технологий : Сборник научных статей по материалам IV Всероссийской научно-технической конференции: в 2 т., Пенза, 03 июля 2022 года. Том 1. – Пенза: Пензенский государственный университет, 2022. – С. 63-66.

119. Введенский В. Э. Измерение транспортабельности и стандартного геометрического отклонения транспортабельности радиоактивных аэрозолей с помощью каскадного импактора / В. Э. Введенский, С. А. Сыпко // АНРИ. – 2024. – № 1(116). – С. 62-73.

120. Широкий В. Р. Сравнение методов нормализации входных признаков в задаче нейросетевого прогнозирования уровня геомагнитных возмущений / В. Р. Широкий, И. Н. Мягкова, С. А. Доленко // Нейрокомпьютеры и их применение: Тезисы докладов, Москва, 14 марта 2017 года. – Москва: Московский государственный психолого-педагогический университет, 2017. – С. 223-224.

121. Баева О. С. Конструирование признаков данных для обучения нейросети: анализ современных подходов / О. С. Баева, С. М. Рощин // Вызовы цифровой экономики: технологический суверенитет и экономическая безопасность: Сборник статей VI Всероссийской научно-практической конференции с международным участием, Брянск, 19 мая 2023

года. – Брянск: Федеральное государственное бюджетное образовательное учреждение высшего образования "Брянский государственный инженерно-технологический университет", 2023. – С. 49-54.

122. Kozik V. I. Selection of features system and network parameters for hyperspectral images classification using convolutional neural networks / V. I. Kozik, E. S. Nezhevenko // CEUR Workshop Proceedings, Novosibirsk, 24–27 августа 2021 года. – Novosibirsk, 2021. – P. 152-160.

123. Cheng Yu. Thoughts on convolution kernel of convolutional neural network / Yu. Cheng, E. A. Anikeev // New aspects of modeling systems and processes: Proceedings of the International scientific and practical conference. – 2023. – P. 483-491.

124. Hybrid Convolutional Neural Network with Residual Neural Network for Breast Cancer Prediction Using Mammography Images // International Journal of Intelligent Engineering and Systems. – 2023. – Vol. 16, No. 1. – P. 375-387.

125. Matatula R. Covid-19 Sentiment Analysis Using Convolutional Neural Network / Reccurent Neural Network Method / R. Matatula, D. Manongga, H. Hendry // Eduvest. – 2022. – Vol. 2, No. 8.

126. Bakumenko A. N., Bakhchevnikov V. V., Derkachev V. A. Crop seed classification based on a real-time convolutional neural network / A. N. Bakumenko, V. V. Bakhchevnikov, V. A. Derkachev // Proceedings of SPIE: SPIE Future Sensing Technologies. – 2020. – P. 11525.

127. Detecting Cracks in Aerated Concrete Samples Using a Convolutional Neural Network / A. N. Beskopylnyi, E. M. Shcherban, S. A. Stelmakh [et al.] // Applied Sciences (Switzerland). – 2023. – Vol. 13, No. 3. – P. 1904.

128. Muhammad A. Adversarial Attacks and Batch Normalization: A Batch Statistics Perspective / A. Muhammad, F. Shamshad, S. Ho. Bae // IEEE Access. – 2023. – Vol. 11. – P. 96449-96459.

129. Ioffe S. Batch Normalization: Accelerating Deep Network

Training by Reducing Internal Covariate Shift [Электронный ресурс] / S. Ioffe, C. Szegedy // arXiv.org. 2015.

130. Tarasov D. A. Fixing 1-bit Adam and 1-bit LAMB algorithms / D. A. Tarasov, V. A. Ershov // Computing, Telecommunications and Control. – 2022. – Vol. 15, No. 4. – P. 86-97.

131. Hanevik E. Client predictors of therapy dropout in a primary care setting: a prospective cohort study / E. Hanevik // BMC Psychiatry. – 2023. – P. 358.

132. De Carlo V., Vismara M. Effectiveness, tolerability, and dropout rates of vortioxetine in comorbid depression: A naturalistic study / V. De Carlo, M. Vismara // Human Psychopharmacology. – 2020. – P. 2750.

133. Murugesan R. Forecasting agricultural commodities prices using deep learning-based models: basic LSTM, bi-LSTM, stacked LSTM, CNN LSTM, and convolutional LSTM / R. Murugesan, E. Mishra, A. H. Krishnan // International Journal of Sustainable Agricultural Management and Informatics. – 2022. – Vol. 8, No. 3. – P. 242.

134. Krishnan A. H. Forecasting agricultural commodities prices using deep learning-based models: basic LSTM, bi-LSTM, stacked LSTM, CNN LSTM, and convolutional LSTM / A. H. Krishnan, R. Murugesan, E. Mishra // International Journal of Sustainable Agricultural Management and Informatics. – 2022. – Vol. 8. – No. 3. – P. 242.

135. Бесерра Муриель К. К. Прогнозирование будущей стоимости колумбийского инвестиционного фонда с помощью рекуррентных нейронных сетей LSTM (LSTM) / К. К. Бесерра Муриель // System Analysis and Mathematical Modeling. – 2024. – Т. 6, № 1. – С. 78-88.

136. Phukan R. Assamese Fake News Detection: A Comprehensive Exploration of LSTM and Bi-LSTM Techniques / R. Phukan, P. J. Goutom, N. Baruah // Procedia Computer Science (см. в книгах). – 2024. – Vol. 235. – P. 2167-2177.

137. Usharani B. ILF-LSTM: enhanced loss function in LSTM to

predict the sea surface temperature / B. Usharani // Soft Computing - A Fusion of Foundations, Methodologies and Applications. – 2023. – Vol. 27, No. 18. – P. 13129-13141.

138. LSTM-SN: complex text classifying with LSTM fusion social network / W. Wei, X. Li, B. Zhang [et al.] // The Journal of Supercomputing. – 2023. – Vol. 79, No. 9. – P. 9558-9583.

139. Da Silva D. G. Comparing Long Short-Term Memory (LSTM) and bidirectional LSTM deep neural networks for power consumption prediction / D. G. Da Silva, A. A. De. M. Meneses // Energy Reports. – 2023. – Vol. 10. – P. 3315-3334.

140. Intelligent framework for degradation monitoring, defect identification and estimation of remaining useful life (RUL) of bearing / A. Kumar, Ch. Parkash, H. Tang, J. Xiang // Advanced Engineering Informatics. – 2023. – Vol. 58. – P. 102206.

141. Gupta R. Smart contract privacy protection using ai in cyber-physical systems: Tools techniques and challenges / R. Gupta // IEEE Access. – 2020. – Vol. 8. – P. 24746-24772.

142. Горелова Д. А. Разбор внедрения SCADA-систем в промышленности на примере внедрения SCADA Trace Mode 6 / Д. А. Горелова, М. А. Горелова // Студенческий вестник. – 2020. – № 40-5(138). – С. 77-82.

143. Исаев Э. В. Применение SCADA-системы Simple SCADA 2 для создания системы управления шахтным компрессором / Э. В. Исаев, С. П. Ярополов, А. Н. Савикин // Инициативы молодых - науке и производству: сборник статей VII Всероссийской научно-практической конференции молодых ученых и студентов. – 2024. – С. 387-390.

144. Григорьев А. А. Характеристика, структура, организация систем управления ERP, ERP II и ERP III / А. А. Григорьев, В. А. Титов // Фундаментальные исследования. – 2017. – № 2. – С. 48-51.

145. Hansen H. F. Investigating ERP System Customization: A Focus

on Cloud-ERP / H. F. Hansen, M. Haddara, M. Langseth // Procedia Computer Science. – 2023. – Vol. 219. – P. 915-923.

146. Фатьянова А. А. Сравнение 1С ERP с зарубежными ERP-системами / А. А. Фатьянова // Факторы успеха. – 2018. – № 2(11). – С. 117-120.

147. Артамонова Т. Е. ERP-системы. Эффективность и проблематика внедрения ERP-систем / Т. Е. Артамонова, А. В. Овсянникова, А. В. Воробьева, А. Э. Попович // Естественные и технические науки. – 2016. – № 4(94). – С. 173-174.

148. Волков М. Р. Идентификация и аутентификация в ASP.NET Core с использованием ASP.NET Core Identity / М. Р. Волков // Развитие науки и технологий в современной России (шифр - ВКРН): Сборник материалов II Всероссийской научно-практической конференции. – 2024. – С. 41-56.

149. Эсеналиева Г. А. Анализ применения ASP.NET при ра3работке информационной системы analysis of the ASP.NET development information system / Г. А. Эсеналиева, А. К. Супибекова // Вестник Кыргызстана. – 2017. – № 2(4). – С. 280-283.

150. Джуба С., Волков А. PostgreSQL 10 / С. Джуба., А. Волков // Системный администратор. – 2019. – № 9(202). – С. 76-86.

151. Мартишин С. А. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench. Методы и средства проектирования информационных систем и технологий. Инструментальные средства информационных систем: учебное пособие для студентов высших учебных заведений / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко – Москва: ИД "ФОРУМ": ИНФРА-М, 2023. – 160 с.

152. The MongoDB injection dataset: A comprehensive collection of MongoDB - NoSQL injection attempts and vulnerabilities / R. D L, A. H V, P. B. Honnavalli, N. S // Data in Brief. – 2024. – Vol. 54. – P. 110289.

153. MoSDeF Cassandra: A complete Python interface for the

Cassandra Monte Carlo software / R. S. Defever, A. W. Dowling, E. J. Maginn [et al.] // Journal of Computational Chemistry. – 2021. – Vol. 42, No. 18. – P. 1321-1331.

154. Филиппов О. Яндекс ClickHouse. Быстрее некуда / О. Филиппов // Системный администратор. – 2017. – № 1-2(170-171). – С. 56-58.

155. Нарушев И. Р. Использование СУБД Postgresql при создании ведомственных информационных ресурсов / И. Р. Нарушев, А. В. Мельников // Актуальные вопросы эксплуатации систем охраны и защищенных телекоммуникационных систем: Сборник материалов Всероссийской научно-практической конференции. – 2016. – С. 201-203.

156. Серебренников И. В. Настройка mlflow для работы с моделями через доменное имя / И. В. Серебренников // Студент года 2024: сборник статей Международного научно-исследовательского конкурса. – 2024. – С. 84-88.

157. Abakumov A. N. RTO "Formation of the Account in SDO "Prometheus" / A. N. Abakumov, V. N. Lukyanchikov, A. A. Tatevosyan // Навигатор в мире науки и образования. – 2016. – No. 2-2(31). – P. 274-279.

158. Zhao N., Lin M., Albahar H. An End-to-End High-Performance Deduplication Scheme for Docker Registries and Docker Container Storage Systems / N. Zhao, M. Lin, H. Albahar // ACM Transactions on Storage. – 2024. – Vol. 20. – P. 1-35.

159. Безруков В. Автоматизация управления тестовыми средами / В. Безруков, Ю. Конов, И. Стечкин // Открытые системы. СУБД. – 2015. – № 1. – С. 14-15.

160. Клочков Д. В. Общие компоненты при кроссплатформенной разработке для web-и мобильных приложений с использованием react и react-native / Д. В. Клочков // Молодой ученый. – 2018. – № 36(222). – С. 1-5.

161. Кирюхина Е. А., Коденков Р. Р., Смирнов А. А., Вихтенко Э. М. Разработка инструментария для работы с экспертами при создании нечетких управляющих систем / Е. А. Кирюхина, Р. Р. Коденков, А. А.

Смирнов, Э. М. Вихтенко // ТОГУ-Старт: фундаментальные и прикладные исследования молодых: Материалы региональной научно-практической конференции. – 2021. – С. 276-281.

162. Зотов Е. Н. Разработка масштабируемого клиент-серверного приложения на основе Webflux для анализа и прогнозирования выручки / Е. Н. Зотов // Молодой исследователь: вызовы и перспективы: Сборник статей по материалам CCCLIV международной научно-практической конференции. – 2024. – С. 219-224

163. Калганов А. С. Визуализация спутниковых данных при помощи инструментов ArcGIS API и Chart.js / А. С. Калганов // Научный аспект. – 2024. – Т. 40, № 1. – С. 5168-5175.

164. Зотов М. А. Визуализация алгебраических байесовских сетей с помощью javascript библиотеки d3.js / М. А. Зотов, А. В. Иванова, А. А. Золотин // Интеллектуальные системы и технологии: современное состояние и перспективы: Сборник научных трудов IV Международной летней школы-семинара по искусственному интеллекту для студентов, аспирантов, молодых ученых и специалистов. – 2017. – С. 86-94.

165. Лыкошин А. С. Визуализируем данные на JavaScript с помощью D3.js / А. С. Лыкошин // Хакер. – 2015. – № 195. – С. 94-101.

166. Свищев А. В. Сравнение Flux и Redux архитектуры клиентских веб-приложений / А. В. Свищев, В. А. Кнышов // Международный журнал гуманитарных и естественных наук. – 2022. – № 11-2(74). – С. 121-124.

167. Колышкина Н. С. Сравнение библиотек Redux и mobx / Н. С. Колышкина, Л. Е. Малкова // Modern Science. – 2020. – № 6. – С. 267-270.

168. Звонарев С. Л., Зубко А. И. О возможных причинах отказов подшипников // Вестник Самарского государственного аэрокосмического университета им. академика С.П. – 2012. – № 3. – С. 16-22.

169. Исмагилов Р. Н., Гареев Р. Р., Ямалиев В. У., Мацибора А. А. Прогнозирование остаточного ресурса подшипника по уровню вибрации

механизма // Экспозиция Нефть Газ. – 2015. – № 3(42). – С. 65-68.

170. Wang A., Wang J. Temperature distribution and scuffing of tapered roller bearing / A. Wang, J. Wang // Chinese Journal of Mechanical Engineering. – 2014. – Vol. 27. – P. 1272–1279.

171. Chernukhin A. V. Development of a model for predicting failures of high-tech production equipment based on machine learning algorithms / A. V. Chernukhin, E.A. Bogdanova, T.V. Savitskaya, A.M. Sverchkov, A.V. Dementienko // Third International Conference on Optics, Computer Applications, and Materials Science. – 2023. – № 11. – С. 1306515.

172. Чернухин А. В. Система предиктивной аналитики технического состояния эксгаустера агломашины с помощью методов искусственного интеллекта / А. В. Чернухин, Е. А. Богданова, Т. В. Савицкая, Д. Г. Кулаков, И. Р. Павлов // Искусственный интеллект и принятие решений. – 2024. – № 3. – С. 87-103.

173. Чернухин А. В. Построение модели предиктивной аналитики неисправностей промышленного оборудования / А. В. Чернухин, Е.А. Богданова, Т. В. Савицкая // Программные продукты и системы. – 2024. – № 2. – С. 254-261.

174. Палюх Б. В. Надежность информационных систем: Учебное пособие для студентов и магистрантов / Б. В. Палюх, В. К. Кемайкин // Тверской государственный технический университет. – 2-е издание, переработанное. – Тверь: Тверской государственный технический университет, 2022. – 156 с.

175. Кобринский Б. А., Аверкин А. Н., Грибова В. В. XXI национальная конференция по искусственному интеллекту / Б. А. Кобринский, А. Н. Аверкин, В. В. Грибова // Искусственный интеллект и принятие решений. – 2024. – № 1. – С. 129-141.

176. Обухов, А. Д. Нейросетевой метод прогнозирования состояния информационных объектов / А. Д. Обухов // Вестник компьютерных и информационных технологий. – 2021. – Т. 18, № 1(199). –

С. 37-43.

177. Чернухин А. В. Разработка алгоритма прогнозирования остаточного ресурса эксгаустера агломашины, основанного на машинном обучении / А. В. Чернухин, Т. В. Савицкая // Научно-технический вестник Поволжья. – 2024. – № 4. – С. 185-190.

Приложение 1. Акт о внедрении ООО «ГИСвер Интегро»

Общество с ограниченной ответственностью
«ГИСвер Интегро»

115280, г. Москва, ул Мастеркова, д.4, этаж 13, помещение I, комн.4
Тел. +7 (495) 725-08-78, E-Mail query@gisw.ru
ИНН/КПП 7737105897/772501001

от 17.04.2025 г. *№* 25A01 *в* ФГБОУ ВО РХТУ им. Д.И. Менделеева
на № _____ *от* _____

АКТ

о внедрении результатов диссертационной работы Чернухина Артёма Валерьевича «Автоматизированная система прогнозирования технического состояния промышленного оборудования на базе методов искусственного интеллекта»

Настоящим актом подтверждаю, что результаты диссертационной работы Чернухина Артёма Валерьевича, выполненной на кафедре Кибернетики химико-технологических процессов Российского химико-технологического университета имени Д. И. Менделеева, используются компанией ООО «ГИСвер Интегро» в рамках разрабатываемого нами продукта «НостраСУ» – комплексной автоматизированной системы управления производственным процессом.

Разработанная А. В. Чернухиным система осуществляет комплексный мониторинг промышленного оборудования, обеспечивая надежный сбор, систематизированное хранение и аналитическую обработку эксплуатационных данных. Ключевыми особенностями системы являются: реализация предиктивной аналитики для точного прогнозирования отказов в режиме реального времени, а также модульная архитектура, обеспечивающая адаптивность решения под различные производственные задачи.

Важным компонентом системы является инновационная подсистема управления моделями, которая обеспечивает непрерывное адаптивное обучение AI-моделей в реальном времени. Эта функциональность позволяет гибко подстраивать алгоритмы под изменяющиеся параметры технологического процесса без остановки производства, что значительно повышает точность и актуальность прогнозных моделей в динамической производственной среде.

С уважением,
Генеральный директор
ООО «ГИСвер Интегро»

_____ / Горанский А.В. /