

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования

**«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра системного программирования**

РАБОТА ПРОВЕРЕНА

Рецензент
Заместитель директора центра
виброиспытаний и мониторинга
состояния конструкций, доцент
кафедры летательных аппаратов
ФГАОУ ВО «ЮУрГУ (НИУ)», к.т.н.

_____ А.В. Ерпалов

«___» _____ 2024 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой, д.ф.-м.н.,
профессор

_____ Л.Б. Соколинский

«___» _____ 2024 г.

**Разработка системы диагностики состояния технологического
оборудования на основе интеллектуальной обработки
многомерных временных рядов**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
ЮУрГУ – 09.04.04.2024.308-1475.ВКР

Научный руководитель,
доцент кафедры СП, к.ф.-м.н.
_____ С.У. Турлакова

Автор работы,
студент группы КЭ-228
_____ Д.К. Лебедев

Ученый секретарь
(нормоконтролер)
_____ И.Д. Володченко
«___» _____ 2024 г.

Челябинск, 2024 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования

**«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра системного программирования**

УТВЕРЖДАЮ

Зав. кафедрой СП

_____ Л.Б. Соколинский

29.01.2024 г.

ЗАДАНИЕ

на выполнение выпускной квалификационной работы магистранта

студенту группы КЭ-228

Лебедеву Денису Константиновичу,

обучающемуся по направлению

09.04.04 «Программная инженерия»

(магистерская программа «Искусственный интеллект и инженерия данных»)

1. Тема работы (утверждена приказом ректора от 22.04.2024 г. № 764-13/12)

Разработка системы диагностики состояния технологического оборудования на основе интеллектуальной обработки многомерных временных рядов.

2. Срок сдачи студентом законченной работы: 20.05.2024 г.

3. Исходные данные к работе

3.1. Zhang S., Zhang S., Wang B., Habetler T.G. Deep Learning Algorithms for Bearing Fault Diagnostics – A Comprehensive Review. // IEEE Access, 2020. – С. 29857–29881.

3.2. Wu G., Yan T., Yang G., Chai H., Cao C. A Review on Rolling Bearing Fault Signal Detection Methods Based on Different Sensors. // Sensors, 2022. – 8330 с.

3.3. Touret T., Changenet C., Ville F., Lalmi M., Becquerelle S. On the use of temperature for online condition monitoring of geared systems – A review. // Mechanical Systems and Signal Processing, 2018. – С. 197–210.

3.4. Chen Z., Zhu H., Wu J., Fan L. Health indicator construction for degradation assessment by embedded LSTM-CNN autoencoder and growing self-organized map. // Knowledge-Based Systems, 2022. – 13 с.

3.5. Zhang A., Zhao X., Wang L. CNN and LSTM based Encoder-Decoder for Anomaly Detection in Multivariate Time Series. // 2021 IEEE 5th Information

Technology, Networking, Electronic and Automation Control Conference (ITNEC), Xi'an, China, 2021. – С. 571–575.

4. Перечень подлежащих разработке вопросов

- 4.1. Провести обзор методов диагностики состояния подшипников технологического оборудования.
- 4.2. Разработать метод интеллектуальной обработки сигналов с датчика температуры для диагностики состояния подшипников.
- 4.3. Реализовать систему диагностики состояния подшипников технологического оборудования на основе разработанного метода.

5. Дата выдачи задания: 29.01.2024 г.

Научный руководитель,
доцент кафедры СП, к.ф.-м.н.

С.У. Турлакова

Задание принял к исполнению

Д.К. Лебедев

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	7
1.1. Обзор литературы.....	7
1.2. Постановка задачи.....	11
2. МЕТОД ДИАГНОСТИКИ.....	15
2.1. Описание и предобработка данных.....	15
2.2. Ансамбль автокодировщиков.....	21
2.3. Тестирование метода.....	24
3. СИСТЕМА ДИАГНОСТИКИ.....	34
3.1. Проектирование системы.....	34
3.2. Реализация системы.....	37
3.3. Тестирование системы.....	41
ЗАКЛЮЧЕНИЕ.....	42
ЛИТЕРАТУРА.....	43

ВВЕДЕНИЕ

Актуальность

Станы горячей и холодной прокатки обеспечивают большую часть прибыли в производстве листовой стали. Незапланированные простои сталепрокатных линий из-за поломок оборудования приводят к разрушительным последствиям и колоссальным потерям прибыли для металлургического бизнеса. Исследование [1] показывает, что 82% компаний в среднем имеют 2 незапланированных простоя за последние 3 года. При этом доля затрат на техническое обслуживание может достигать 40%, а иногда и 70% от себестоимости выпускаемой продукции [2]. В условиях глобальной конкуренции нулевое время незапланированных простоев и сокращение затрат на ремонт оборудования является главным приоритетом для большинства металлургических предприятий.

Использование инновационных решений предиктивного обслуживания производственного оборудования позволяет поддерживать конкурентоспособность за счет повышения качества производства и сокращения ремонтных затрат. Предиктивное обслуживание позволяет уменьшить затраты на ремонт на 25–30%; повысить скорость диагностики и ремонта на 70–75%; уменьшить время простоя оборудования на 35–45% по сравнению с традиционными стратегиями технического обслуживания [3, 4].

Предиктивное обслуживание стало реализуемым на практике за счет подходов Индустрии 4.0, которые включают использование новых типов сенсоров в комплексе с применением современных информационных технологий: элементов искусственного интеллекта и методов машинного обучения [5–7]. Сенсорные сети собирают актуальную информацию с технологического оборудования, а методы интеллектуальной обработки данных позволяют выделить ценную диагностическую информацию о состоянии этого оборудования.

Постановка задачи

Целью выпускной квалификационной работы является разработка системы диагностики состояния технологического оборудования на основе интеллектуальной обработки многомерных временных рядов. Для достижения поставленной цели необходимо решить следующие задачи:

- 1) провести обзор методов диагностики состояния подшипников технологического оборудования;
- 2) разработать метод интеллектуальной обработки сигналов с датчика температуры для диагностики состояния подшипников;
- 3) реализовать систему диагностики состояния подшипников технологического оборудования на основе разработанного метода.

Структура и содержание работы

Работа состоит из введения, трех глав, заключения и списка литературы. Объем работы составляет 49 страниц, объем списка литературы – 66 источников.

В первой главе представлен обзор литературы по тематике исследования, а также проведена постановка задачи.

Вторая глава посвящена описанию метода диагностики. В главе приводится описание анализируемых данных, алгоритм их предобработки, а также процесса обучения используемых нейросетевых моделей. В конце главы приведено тестирование разработанного метода.

В третьей главе содержится описание реализации системы диагностики состояния подшипников технологического оборудования на основе разработанного во второй главе метода.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1. Обзор литературы

Подшипник качения является наиболее критичным элементом любого металлообрабатывающего оборудования, что объясняет необходимость разработки методов получения и обработки информации о его состоянии [8–12]. Подшипники входят в состав роликов прокатных станов, изгибо-растяжных машин, машин для сматывания стальной полосы, телег барабанных накопителей. Вибрация, загруженность и температурные режимы работы данного оборудования существенно сокращают срок службы подшипников [13–18]. Внезапный выход из строя подшипника может привести как к появлению брака в стальном подкате, так и к полной остановке производственной линии. Однако анализ рабочих параметров подшипника, таких как уровень вибрации, температуры, акустического шума, состав смазки, может предупредить появление критических разрушений. Выбор анализируемого параметра зависит от места установки подшипника, доступности для установки сенсоров и окружающих условий [19].

В условиях производственного шума и повышенной вибрации металлопрокатных станов температура обладает большим потенциалом как источник информации о состоянии подшипников [20]. Тепло, выделяемое подшипником, возникает в основном за счет теплоты трения между телом качения и кольцами подшипника, а также телом качения и сепаратором. Таким образом, тепло подшипника является фрикционным, и на его количество влияют скорость подшипника, уровень и направление нагрузки, а также параметры смазки [21]. Распределение тепла в подшипниках в основном происходит посредством теплопередачи, однако нагретый подшипник также излучает теплоту. Поэтому информацию о температуре подшипника можно получить как контактными, так и бесконтактными методами [22].

Бесконтактный способ получения информации основан на анализе тепловых изображений. В книге [23] автор установил, что использование

тепловизионной камеры позволяет получить информацию о температуре роликового подшипника, сравнимую по точности с контактным методом. В работах [24–26] показано, что применение сверточной нейронной сети (CNN) к тепловым изображениям подшипников позволяет выявить наличие различных дефектов с высокой точностью. В работе [27] авторы разработали модифицированную гауссовскую сверточную сеть глубокого доверия для классификации дефектов системы ротор-подшипник по инфракрасным тепловым изображениям. В работе [28] авторы предложили архитектуру расширенной сверточной нейронной сети (ECNN) с масштабированной экспоненциальной линейной единицей (SELU) и модифицированным стохастическим градиентным спуском (MSGD) для классификации тепловых изображений различных дефектов подшипников.

Дополнительная обработка тепловых изображений может повысить производительность классификаторов. В работе [29] авторы показали, что применение 2D-дискретного вейвлет-преобразования с последующим устранением ненужных особенностей с использованием анализа главных компонент к тепловым изображениям подшипника двигателя повышает производительность таких классификаторов как сложное дерево решений (CDT), линейный дискриминантный анализ (LDA) и метод опорных векторов (SVM).

Однако, для обеспечения точности измерения температуры чрезвычайно важно определить коэффициент черноты для каждой из контролируемых поверхностей подшипника. Если определение коэффициента черноты невозможно, то рекомендуется покрыть контролируемые поверхности слоем вещества с известным, высоким коэффициентом черноты, например, матовой черной краской. Совершенно очевидно, что в тяжелых производственных условиях определение коэффициента черноты подшипника является нетривиальной задачей, так как окалина, смазочно-охлаждающие жидкости, кислоты оказывают негативное влияние на поверхность подшипника. Кроме того, необходимо обеспечить доступ к самому подшип-

нику, что также не всегда является возможным. Например, подшипники прокатных станов находятся внутри роликов и визуальный доступ к ним невозможен [30].

Контактный способ измерения температуры заключается в размещении датчика, например, терморезистора или термопары, в контакте с поверхностью подшипника, например, на внешнем или внутреннем кольце подшипника. В работе [31] авторы показали, что измерение температуры внешнего кольца подшипника с помощью термистора значительно увеличивает точность нейросетевой диагностической модели. В работе [32] авторы описывают беспроводной LC-датчик температуры в виде серебряной пленки, которая наносится на внутреннее кольцо подшипника для обнаружения аномалий в изменении температуры.

Кроме того, существуют готовые коммерческие решения для измерения температуры в некоторой точке подшипника. Например, температурный сенсор серии ADB для подшипниковых узлов производства 4B Braime Components [33] представляет собой конструкцию из измерительного щупа с термопреобразователем, который устанавливается в корпус подшипникового узла в канал подачи смазки и контактирует с внешним кольцом подшипника. Температурные сенсоры производства Muller Beltex [34] также обеспечивают измерение температуры с внешнего кольца подшипника.

Общий недостаток контактного способа измерения температуры подшипников заключается в том, что температурный сенсор, как правило, закрепляется на внешнем или внутреннем кольце подшипника. Однако существует ряд исследований, которые доказывают, что самая высокая температура в подшипнике наблюдается у тел качения и в зоне контакта тел качения с дорожкой качения [32, 35–42]. Следовательно, сенсор, закрепленный на кольцах подшипника, может не отображать фактическую температуру.

В работе [43] авторы предложили конструкцию миниатюрной термомпары, которую можно закрепить непосредственно на теле качения роликового подшипника. Результаты испытаний показали, что измеренная температура тел качения на 6–10 °С выше, чем температура наружного кольца. Однако измерение температуры тел качения подшипника контактным методом все еще представляется проблемой из-за требований к миниатюризации сенсора.

В этом направлении куда более целесообразно измерять температуру жидкой или консистентной смазки подшипника [44]. Смазка подшипника непрерывно контактирует с телами качения и внутренними дорожками колец подшипника и отводит от них тепло [45].

В данной работе рассматривается применение беспроводного мультисенсорного температурного датчика. Конструкция такого датчика позволяет установить термопреобразователь в канал смазки роликового подшипника центрирующего блока накопительной телеги металлопрокатного стана. Такой подход позволяет проводить измерение температуры одновременно в нескольких точках по всему каналу смазки, включая точку расположения подшипника. Сенсор регистрирует не только абсолютное значение температуры, но и распределение температуры в окружающем подшипник материале.

Особенности получаемых таким образом данных позволяют разработать новый метод диагностики подшипников технологического оборудования на основе интеллектуальной обработки данных. Для этого в данной работе информация о тепловом потоке подшипникового узла анализируется с помощью нейронной сети специальной архитектуры. Получение и обработка данных, а также метод диагностики состояния подшипников описаны в следующей главе.

1.2. Постановка задачи

Современное производство холодного проката в металлургической промышленности основано на использовании непрерывных прокатных станков. Стан холоднй прокатки работает в непрерывном режиме и включает ряд вспомогательных агрегатов, обеспечивающих непрерывную подачу полосы на входе стана. Среди таких агрегатов выделяют размотчики для подачи горячекатанной полосы и травильную линию для удаления с этой полосы окалины. Согласование скоростей между станом, травильной линией и размотчиками осуществляется с использованием входного и выходного петленакопителей (рисунок 1). Стабильность и безотказность работы петленакопителей обеспечивает непрерывность работы всего стана в целом.

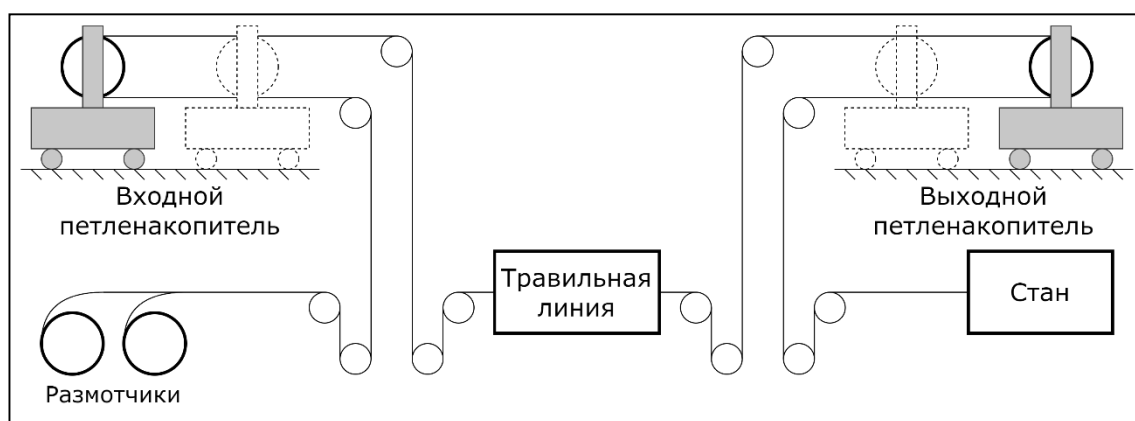


Рисунок 1 – Схема линии холодной прокатки

Петленакопитель включает несколько барабанов, закрепленных на подвижном основании – телеге. Телега перемещается по рельсам и удерживается на них с использованием четырех центрирующих блоков. Каждый центрирующий блок включает два подвижных ролика с подшипниками качения. Работа данных подшипников происходит в условиях повышенных вибраций, большого количества окалины, частых разгонов и торможений при относительно небольшой скорости передвижения телеги.

Выход из строя подшипников центрирующего блока может привести к заклиниванию ролика и сходу с рельсов телеги накопителя, что в

свою очередь приводит к дорогостоящим простоям стана. Применение традиционных методов вибродиагностики в подобных условиях не позволяет достоверно и своевременно обнаружить выход подшипника из строя. Альтернативный подход – диагностика подшипников по тепловому потоку. При возникновении неисправности или недостаточном количестве смазки повышается трение между элементами подшипника, нарушаются условия теплопередачи, что приводит к повышению его температуры.

В данной работе для определения состояния подшипников центрирующего блока петленакопителя используется метод диагностики, основанный на анализе теплового потока. Отличие данного метода от диагностики по температуре заключается в количестве анализируемой информации. При диагностике по тепловому потоку используются данные, полученные одновременно с нескольких сенсоров, расположенных в различных, относительно диагностируемого подшипника, температурных точках. Анализ таких данных позволяет определять не только перегрев, но и его характер. В контексте диагностики подшипников это имеет значение, так как дает возможность учитывать различные сторонние факторы. Например, повышение температуры окружающей среды может значительно повлиять на температуру подшипника, однако это не свидетельствует о развитии дефекта в подшипнике. Таким образом, обработка информации о тепловом потоке позволяет значительно повысить чувствительность диагностической модели, речь о которой пойдет далее.

Одним из наиболее популярных подходов к решению задач технической диагностики является подход моделирования нормального поведения системы (Normal behavior model, NBM) [46]. Существует множество алгоритмов для моделирования нормального поведения системы, однако все они имеют общий принцип. Данные АСУТП (SCADA), полученные во время нормального режима работы подвергаются предварительной обработке (фильтрация шумов, удаление пропусков, выбросов). Модель NBM обучается на очищенных нормальных данных и может использоваться для

прогнозирования ожидаемого нормального поведения в наборе новых данных, поступающих в режиме реального времени и прошедших ту же процедуру предобработки. На следующем этапе анализируется разница между наблюдаемым и предсказанным моделью значениями. Большое различие говорит об отклонении текущего поведения системы от нормального и возможной неисправности. Можно выделить три группы алгоритмов моделирования нормального поведения [47], основанные на: статистических методах, традиционных методах машинного обучения, методах глубокого обучения.

Статистические алгоритмы, например, такие как метод наименьших квадратов, ARIMA (Autoregressive Integrated Moving Average), OLS (Ordinary Least Squares), и их комбинации [48] имеют преимущество перед более сложными моделями машинного обучения в виде небольшого количества необходимых для их работы нормальных данных. Это делает их подходящими в качестве инструмента первичного анализа или в областях, где существуют ограничения на вычислительную нагрузку или количество доступных данных. Однако недостатком является то, что они относительно просты, что делает их в целом непригодными для моделирования очень сложной нелинейной динамики.

Под традиционным машинным обучением подразумеваются такие модели, как деревья решений, случайные леса, градиентный бустинг и метод опорных векторов [49–52]. Эти модели более сложны, чем традиционные статистические модели, и лучше подходят для моделирования нелинейной динамики. Однако они, как правило, требуют больше обучающих данных и времени. Примерами алгоритмов, которые используются в настоящее время, являются ансамбли деревьев решений.

В последние годы самым популярным методом построения модели нормального поведения стали нейронные сети. Модели глубокого обучения более сложны, чем традиционные модели машинного обучения, для их обучения требуется еще больше данных и вычислительных ресурсов. Они

хорошо моделируют нелинейную динамику и являются самой популярной моделью NBM в современных исследованиях, особенно нейронная сеть, имеющая архитектуру «автокодировщик» [53–56]. В качестве модели нормального поведения в данной работе будет использована нейронная сеть именно такой архитектуры.

Таким образом, результатом данной работы будет являться система диагностики подшипников роликов центрирующего блока петленакопителя, реализующая метод диагностики, описанный в следующей главе.

Метод диагностики основан на следующих двух идеях:

- анализ данных о тепловом потоке для определения неисправностей подшипников качения;
- применение модели нормального поведения для обнаружения аномалий в данных.

Система диагностики объединяет следующие компоненты:

- алгоритм обработки данных;
- модели нормального поведения (автокодировщики);
- взаимодействие с базой данных.

Система диагностики будет представлять собой скрипт, запускаемый на сервере металлопрокатного предприятия в виде демона (daemon) Linux. Также в системе будет реализовано взаимодействие с базой данных для получения сырых данных и записи результата обработки. Проектирование и реализация системы диагностики описаны в третьей главе.

2. МЕТОД ДИАГНОСТИКИ

2.1. Описание и предобработка данных

В представленной работе в качестве типичных данных о тепловом потоке рассматриваются данные с мультисенсорного датчика температуры (рисунок 2), получающего измерения одновременно в четырех точках. Три точки находятся на концах и в центре щупа, еще одна на корпусе электронного блока датчика. Такой датчик устанавливается в канал смазки подшипникового узла, что позволяет получать информацию о температурном градиенте от непосредственно самого узла до окружающей среды.

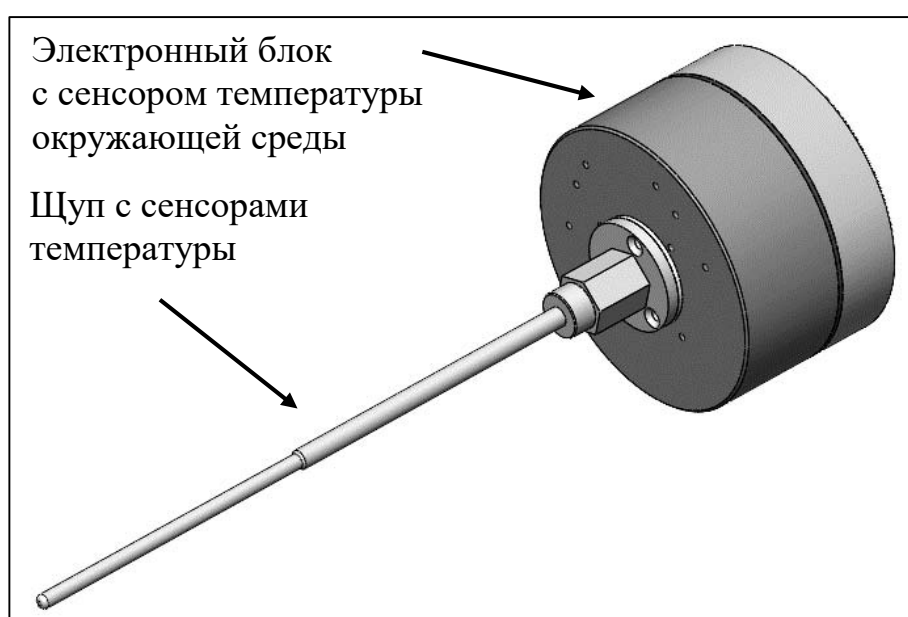


Рисунок 2 – Мультисенсорный датчик температуры

Измерения с мультисенсорного температурного датчика представляют собой многомерный временной ряд. Значения снимаются раз в минуту. Заданы следующие обозначения для данных, получаемых с сенсоров температуры: t_1 – температура на конце щупа (вблизи подшипника), t_2 – температура в центре щупа, t_3 – температура в начале щупа (вблизи электронного блока), t_4 – температура окружающей среды.

Данные о температуре подшипника, получаемые в процессе эксплуатации технологического оборудования, часто имеют выбросы. Это связано с возникновением электромагнитных помех и стабильностью передачи

сигнала. На рисунке 3 представлены исходные показания датчика, где выделяется несколько аномально больших значений температуры, которые являются выбросами. Перед построением диагностической модели необходимо отфильтровать такие выбросы.

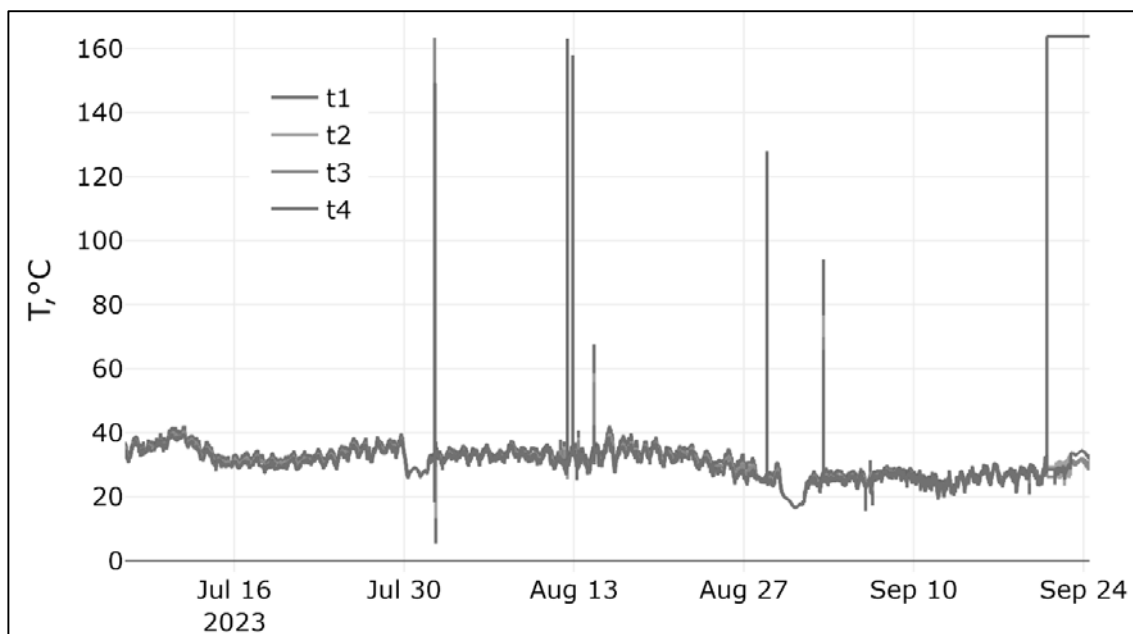


Рисунок 3 – Исходные показания датчика

Видимые выбросы легко отфильтровать, используя минимальное и максимальное пороговые значения для полученных измерений. На рисунке 4 изображен временной ряд после такой грубой фильтрации. Однако, такой подход не позволяет очистить все выбросы: несколько значений остались неотфильтрованными.

Также, при детальном рассмотрении полученного временного ряда, можно обнаружить небольшие выбросы, пропущенные грубой фильтрацией (рисунок 5). Почти все такие выбросы представляют собой резкий скачок значения температуры в одной временной точке. Такой скачок возможен как в положительную, так и в отрицательную сторону. Также стоит отметить, что выброс может наблюдаться как одновременно для всех значений температуры, так и только для некоторых. Наличие подобных выбросов серьезно осложняет построение модели.

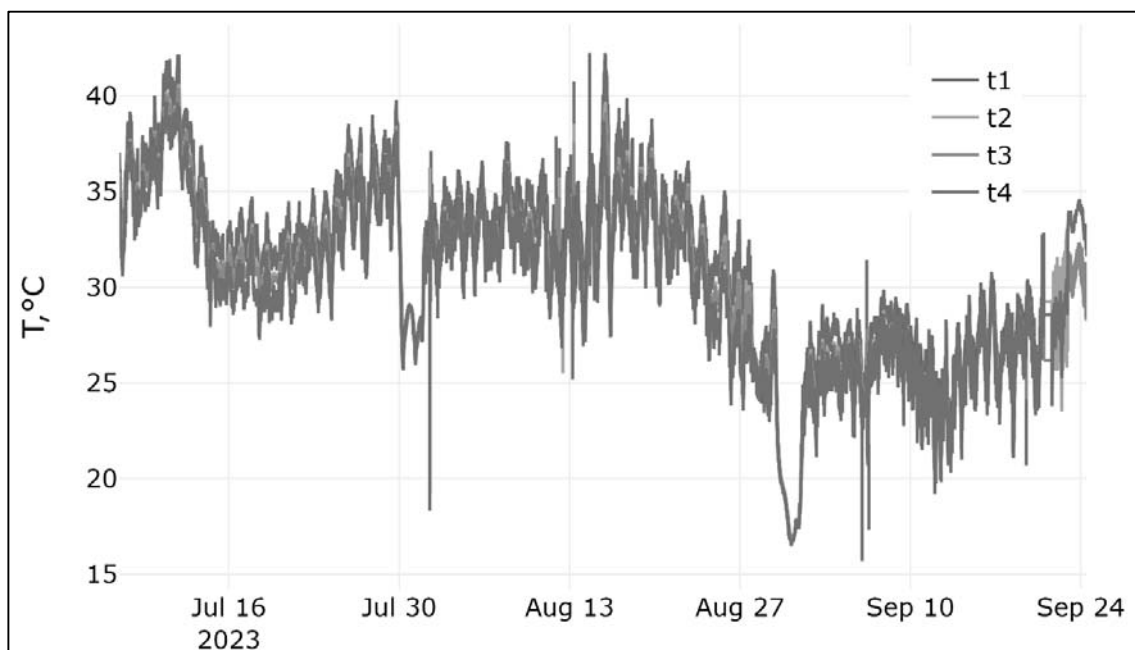


Рисунок 4 – Данные после грубой фильтрации

При разработке алгоритма фильтрации таких выбросов необходимо учесть, что графики измеряемых температур имеют колебания разной амплитуды, связанные со скоростью вращения подшипника и их важно сохранить, так как они несут информацию для диагностической модели о нормальном поведении оборудования.

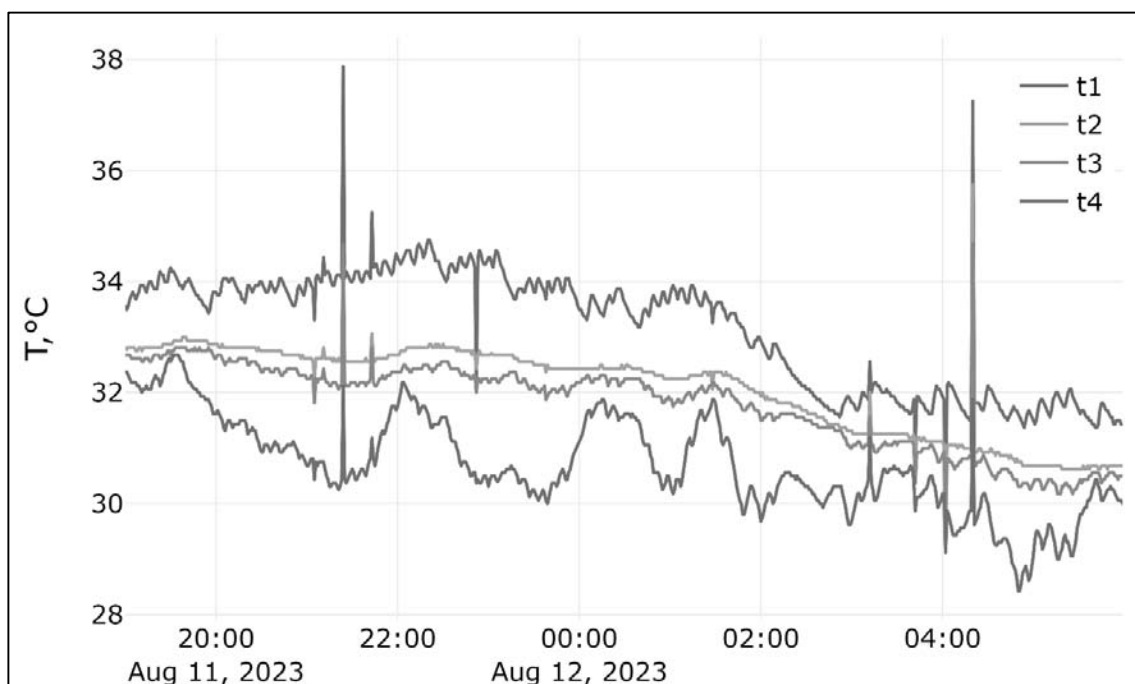


Рисунок 5 – Пример небольших выбросов

Для фильтрации рассматриваемых выбросов предлагается использование медианного фильтра с адаптивным порогом. Алгоритм работы такого фильтра следующий.

Все сигналы – одномерные временные ряды – фильтруются по отдельности с помощью скользящего окна, движущегося слева направо с шагом 1. На каждом шаге берется L подряд идущих значений временного ряда и вычисляется среднеквадратическое отклонение (СКО). Если вычисленное СКО превышает некоторое пороговое значение, это говорит о наличии выброса в окне, который будет находиться в крайнем правом положении. Найденный выброс заменяется на медиану, то есть на одно из двух нормальных значений сигнала в текущем окне. Ширина окна $L = 3$ была выбрана исходя из следующих соображений. Нельзя взять $L = 2$, так как в таком случае медиана окажется равна полусумме нормального значения и выброса, что, как правило, будет слишком большим значением. Если взять $L > 3$, то сигнал будет сильнее искажен, из-за возможной замены выброса на значение в более удаленной точке.

В качестве порогового значения разумно взять среднеквадратическое отклонение значений временного ряда. Поскольку данные с каждого сенсора имеют различный разброс значений температуры, порог должен быть адаптивным и подстраиваться под конкретный временной ряд.

Каждое следующее значение порога вычисляется с помощью скользящего окна шириной W . Важно отметить, что окно в W отсчетов для определения порога движется «с запаздыванием» относительно окна в L отсчетов для определения наличия выброса, а также учитывает значения уже отфильтрованного ряда.

Ширина окна W для вычисления порога, в отличие от ширины окна L для определения выброса, не может быть мала, так как для корректной работы фильтра необходимо учитывать достаточно долгую историю изменений значений температуры. Помимо этого, если значения температуры в течение какого-то времени, соответствующему ширине окна W , остаются

постоянными, то значение порога станет равно нулю. Соответственно, все следующие значения будут расценены как выбросы и заменены на одно и то же значение, а порог так и останется равен нулю.

Чтобы избежать указанных проблем, необходимо подобрать значение ширины окна W . Для рассматриваемых сигналов было выбрано значение $W = 60$. Таким образом, для вычисления порога используется информация об изменении температуры за последний час, поскольку исследуемые данные были получены с частотой раз в минуту. Такая ширина окна W связана, с одной стороны, с медлительностью тепловых процессов, с другой стороны, с необходимостью рассчитывать достаточно низкое значение порога для более высокой чувствительности фильтра к выбросам.

Предполагается, что в начале фильтрации в окне для вычисления первого значения порога выбросов нет. При необходимости, это проверяется вручную, путем построения графика первых W значений временного ряда.

Многомерный временной ряд после описанной фильтрации приведен на рисунке 6.

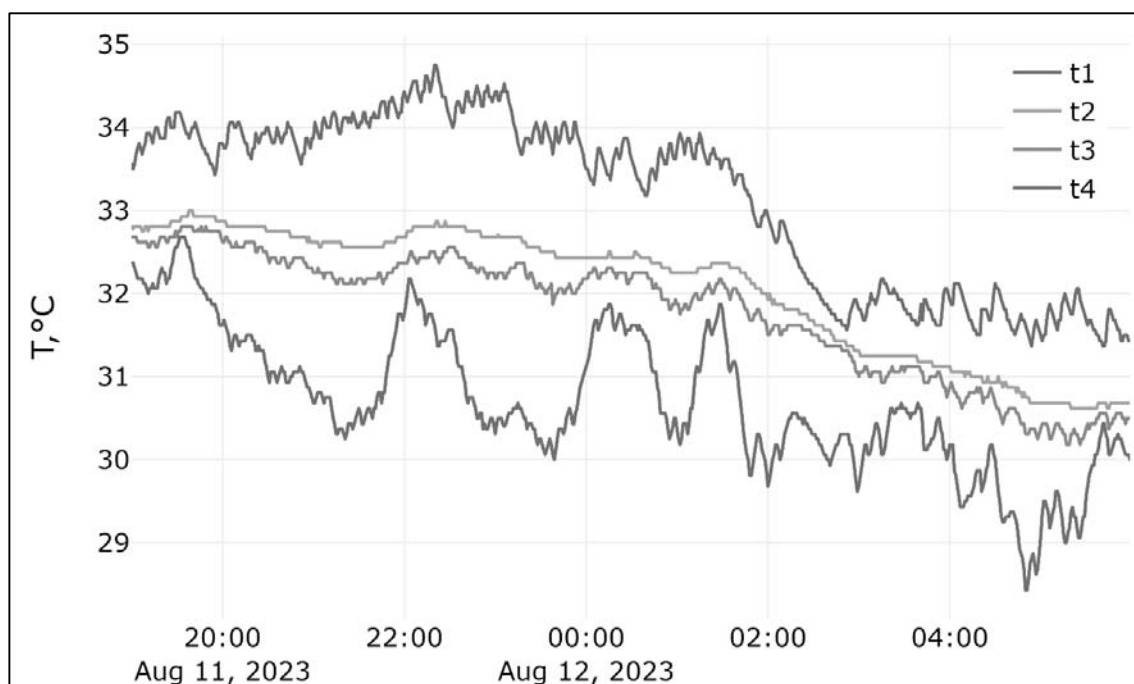


Рисунок 6 – Временной ряд после фильтрации

Предложенный алгоритм фильтрации вносит минимальные искажения во временной ряд, имеет адаптивный порог, вычисляемый на основе величины разброса значений временного ряда, а также сохраняет модуляцию температуры скоростью. Также стоит отметить, что предложенный алгоритм не требует предварительной фильтрации выбросов с аномально большими значениями, так как сам способен найти и отфильтровать их.

Диагностика по тепловому потоку подразумевает под собой анализ скорости и характера изменений разницы температур между сенсорами. Поэтому, в процессе предобработки сигналов, помимо фильтрации данных от выбросов, необходимо преобразовать вектор признаков для моделей. Новый вектор признаков состоит из разностей температур: $t_1 - t_2$, $t_1 - t_3$, $t_1 - t_4$, $t_2 - t_3$, $t_2 - t_4$, $t_3 - t_4$; всего шесть признаков. После формирования новых признаков необходимо нормализовать входные данные. В данной работе для этого используется Z-нормализация.

Помимо этого, входные данные моделей должны содержать информацию о разнице температур за некоторый временной промежуток, а не просто в определенный момент времени. Для этого формируются сэмплы – матрицы значений, где столбцы являются признаками, а строки – значениями этих признаков в определенный момент времени. Количество строк в каждом сэмпле одинаково и определяется заранее. Сигналы получены с частотой одно измерение в минуту, поэтому, учитывая особенности работы диагностируемого оборудования, количество измерений в сэмпле было определено равным 30. Сэмплы состояются с шагом в одно измерение, то есть $N - 1$ из N измерений в текущем сэмпле также присутствуют в предыдущем. Это позволяет осуществлять диагностику непрерывно и вовремя реагировать на появление аномальных данных.

После выполнения всех вышеописанных действий над исходными сигналами получают данные, подготовленные на вход для автокодировщиков, которые производят дальнейшую их обработку.

2.2. Ансамбль автокодировщиков

Для диагностики состояния подшипникового узла центрирующего блока петленакопителя по тепловому потоку в данной работе используется ансамбль из трех нейронных сетей архитектуры «автокодировщик», каждая из которых выполняет свою задачу.

Основная нейросеть представляет собой модель нормального поведения и предназначена для выявления аномальных данных. Появление аномальных данных в сигналах теплового потока с подшипника свидетельствует о нештатных условиях работы, что может быть вызвано развитием или наличием неисправности.

Две другие модели позволяют определить принадлежность аномальных данных к одному из двух выделенных типов: «прокат повышенной температуры» или «заклинивание подшипника».

Сигналы, соответствующие обоим типам данных, сильно отличаются от нормальных, при этом редко встречаются в обучающих данных. Ввиду этого основная модель реагирует на них как на аномальные данные. Если в случае «заклинивания» эта реакция оправдана, то для «проката повышенной температуры» срабатывание автокодировщика является ложным, так как не свидетельствует о неисправности. Модель, определяющая «заклинивание», дает возможность получить дополнительную информацию о характере неисправности.

Работа автокодировщика заключается в сжатии входных данных с помощью кодировщика и последующем их восстановлении декодировщиком таким образом, чтобы разница между входными и выходными данными модели была минимальной. Для того, чтобы ошибка восстановления исходного сигнала была как можно меньше, модели нужно научиться представлять данные меньшим числом признаков. Если модели, обученной на нормальных данных, на вход поступят аномальные сигналы, ошибка восстановления будет большой.

Традиционно используется нейронная сеть, симметричная относительно сжатого вектора признаков. Ряд последних публикаций [57–59] говорит о высокой эффективности комбинации кодировщика, основанного на одномерных сверточных слоях CNN с декодировщиком на основе рекуррентных слоев LSTM. Именно такая архитектура моделей (рисунок 7) используется в данной работе.

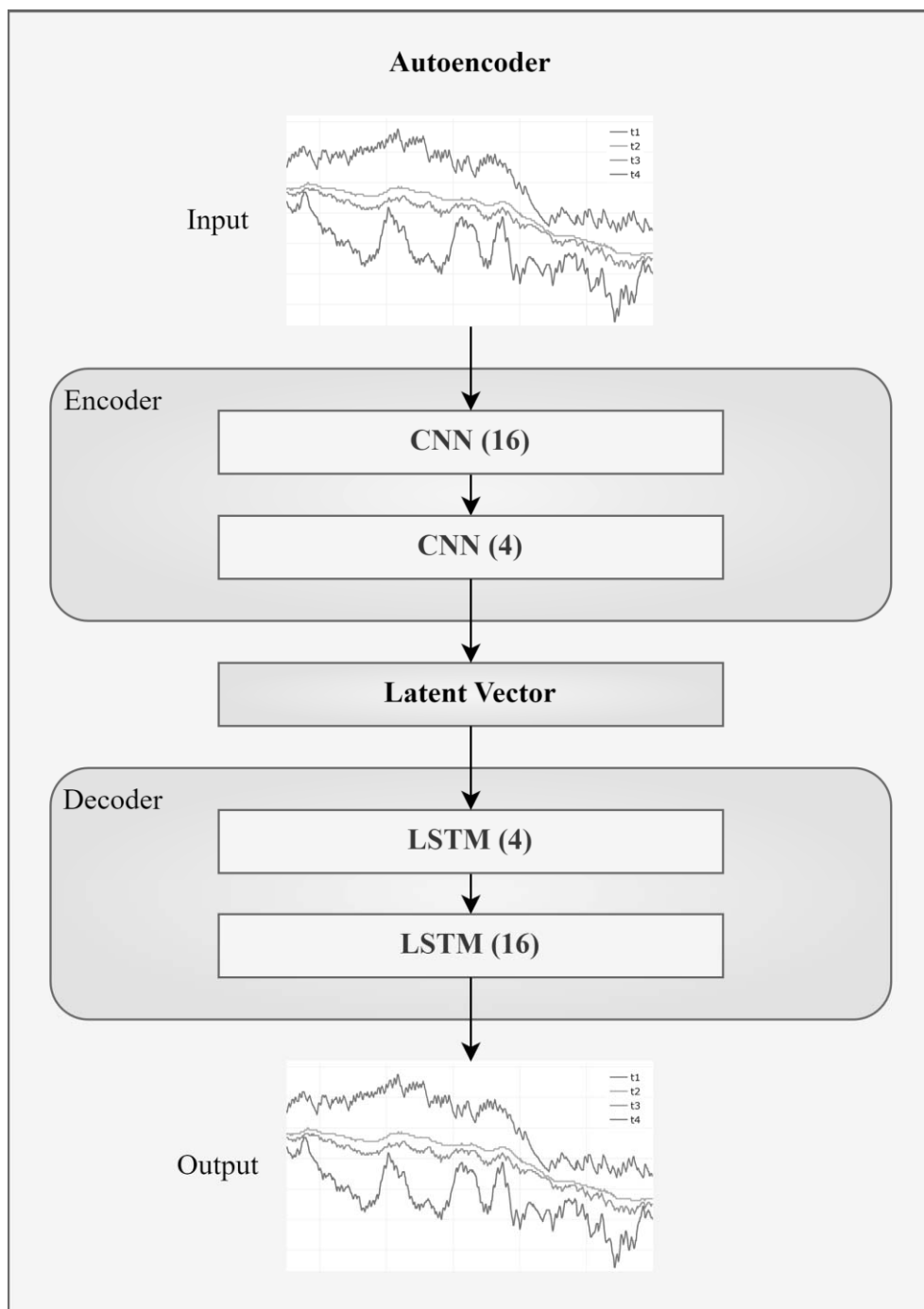


Рисунок 7 – Архитектура моделей ансамбля

Кодировщик состоит из двух одномерных сверточных CNN слоев с числом нейронов 16 в первом слое и 4 во втором. Декодировщик представляет собой два слоя LSTM по 4 и 16 нейронов соответственно. На каждом слое использовалась функция активации ReLU. В качестве оптимизатора модели использовался Adam.

В качестве функции потерь, характеризующей ошибку модели по реконструкции входного сигнала, которая минимизируется в ходе обучения автокодировщика, а также в качестве метрики для оценки результатов работы модели (ошибки реконструкции) в данной работе используется средняя абсолютная ошибка MAE , вычисляемая по формуле (1):

$$MAE = \frac{1}{IN} \sum_{i=1}^I \sum_{n=1}^N |\hat{x}_n^{(i)} - x_n^{(i)}|, \quad (1)$$

где $x_n^{(i)}$ ($n = 1, \dots, N$) – исходные значения i -го признака ($i = 1, \dots, I$);

$\hat{x}_n^{(i)}$ – восстановленные значения i -го признака.

Схема работы всего ансамбля приведена на рисунке 8. Данные обрабатываются моделями последовательно, при этом использование каждой из дополнительных моделей зависит от результатов обработки данных предыдущей моделью.

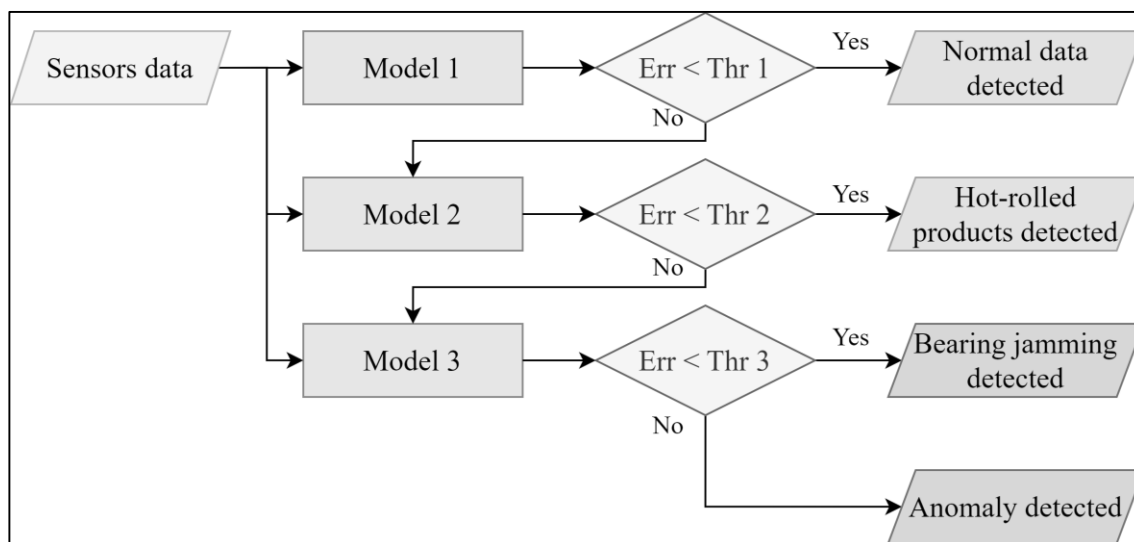


Рисунок 8 – Схема работы ансамбля

Если основная модель определила текущие данные как аномальные, то они посылаются на вход следующей модели ансамбля, предназначенной для определения «проката повышенной температуры». Если же эта модель не присвоила им соответствующий статус, то далее данные передаются в третью модель, определяющую «заклинивание». В конечном итоге, третья модель присваивает входным данным либо статус «заклинивание подшипника», либо статус «аномальные данные», что говорит о возможном развитии дефекта подшипникового узла.

2.3. Тестирование метода

Обучение всех трех моделей в ансамбле проводилось последовательно, так как входные данные второстепенных моделей зависят от результата работы основной. Модели обучались на разных данных в соответствии со своим предназначением.

Обучающая выборка для основной модели, предназначенной для определения аномалий, была составлена из данных о тепловом потоке за пять периодов, в течение которых оборудование работало штатно.

Каждый период обучающей выборки составляет двое суток:

- с 12.07.2023 г. до 14.07.2023 г.;
- с 22.07.2023 г. до 24.07.2023 г.;
- с 12.08.2023 г. до 14.08.2023 г.;
- с 22.08.2023 г. до 24.08.2023 г.;
- с 12.09.2023 г. до 14.09.2023 г.

Валидационная выборка была составлена из трех периодов штатной работы также по двое суток:

- с 10.07.2023 г. до 12.07.2023 г.;
- с 10.08.2023 г. до 12.08.2023 г.;
- с 10.09.2023 г. до 12.09.2023 г.

Тестирование обученной модели проводилось на всех имеющихся данных с 07.07.2023 г. по 21.09.2023 г.

Все модели ансамбля были построены с помощью библиотеки Keras [60] для языка программирования Python. Для обучения автокодировщиков изначально было установлено количество эпох равное 1000. Однако при обучении был использован механизм ранней остановки обучения из библиотеки Keras. Этот механизм был настроен таким образом, что если в течение 20 эпох ошибка модели на валидационной выборке не уменьшалась, то обучение прекращалось.

Обучение модели заняло 459 эпох. Ошибка реконструкции (RE) основной модели представлена на рисунке 9. Превышение ошибки реконструкции (синяя линия) порогового значения (красная линия) свидетельствует об аномальных данных. Порог определяется по формуле (2):

$$Threshold = mean(RE) + 5 \cdot std(RE), \quad (2)$$

где $mean(RE)$ – среднее значение ошибки реконструкции на обучающих данных;

$std(RE)$ – среднеквадратическое отклонение ошибки реконструкции на обучающих данных.

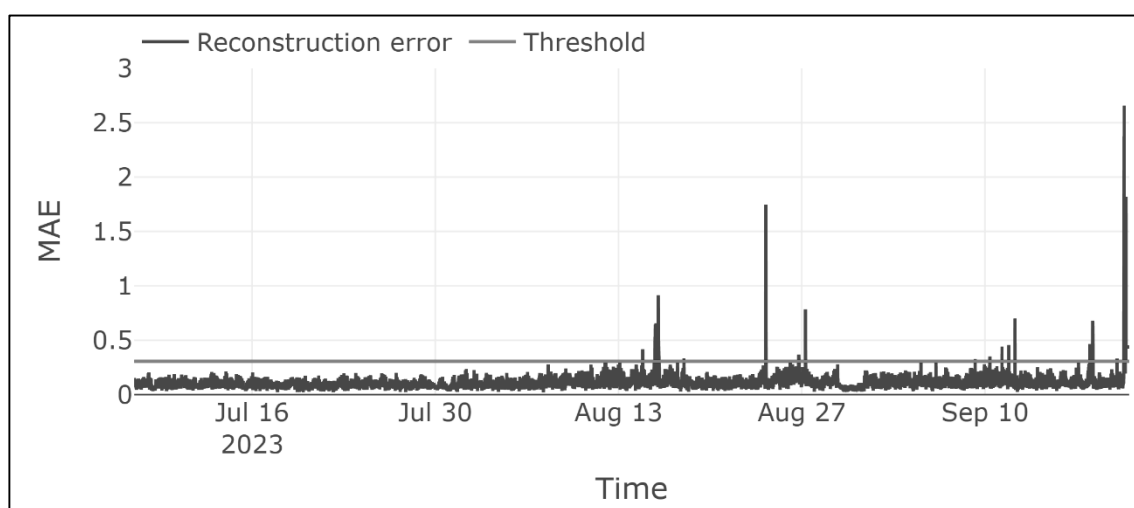


Рисунок 9 – Ошибка реконструкции основной модели

Так как MAE положительна, порог является границей диапазона, который по правилу пяти сигм [61] содержит значения ошибки реконструкции, соответствующей нормальным данным, с вероятностью 0,9999994.

На представленном графике присутствует несколько «реакций» модели – превышений ошибки реконструкции установленного порога. Среди этих реакций можно выделить два ранее указанных типа: реакции на прокат повышенной температуры и реакции на заклинивание подшипника.

На рисунке 10 представлена ошибка реконструкции основной модели 11.09.2023 г. с 05:20 до 08:20, а также соответствующие сигналы сенсоров температуры. Этот пример демонстрирует реакцию модели на прокат повышенной температуры.

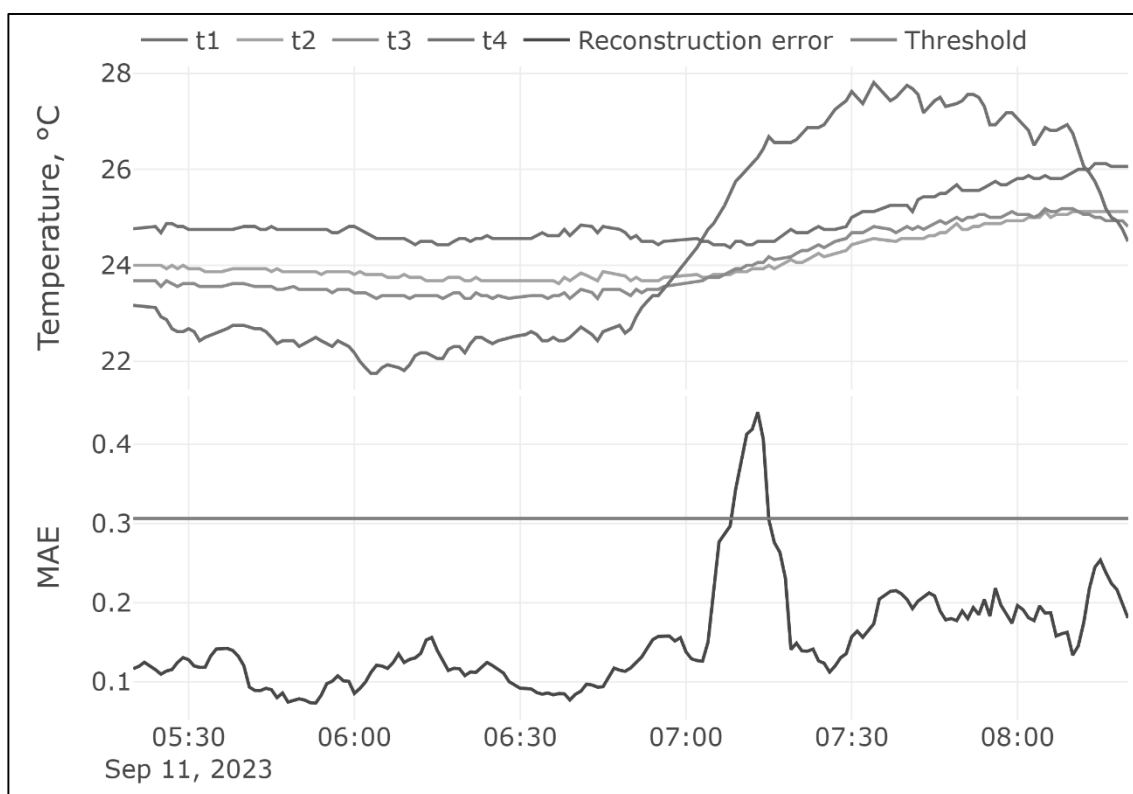


Рисунок 10 – Реакция модели на прокат повышенной температуры

На графике хорошо виден резкий рост температуры окружающей среды t_4 , вызывающий рост температуры в остальных точках, что соответствует прохождению через накопитель проката повышенной температуры.

На рисунке 11 представлена ошибка реконструкции основной модели 27.08.2023 г. с 04:00 до 09:00, а также соответствующие сигналы сенсоров. Этот пример демонстрирует реакцию модели на заклинивание подшипника.

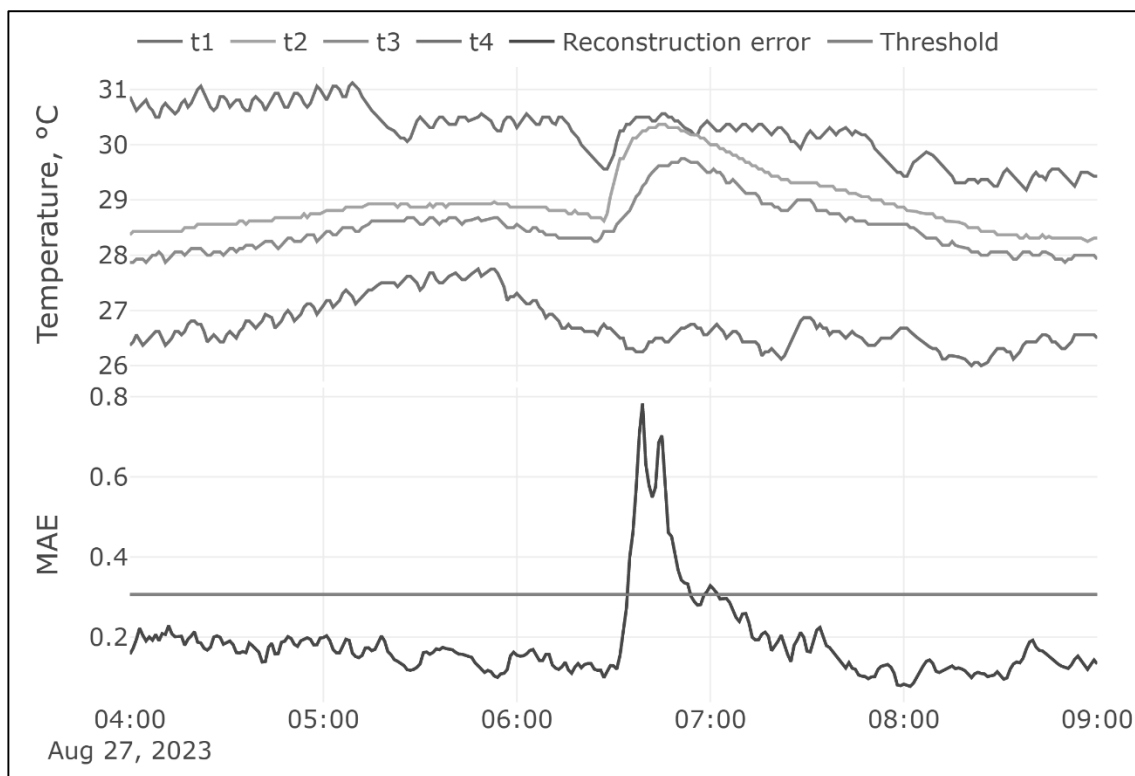


Рисунок 11 – Реакция модели на заклинивание подшипника

При нормальном режиме работы нагрев сенсора, ближайшего к подшипниковому узлу, происходит из-за работы двух подшипников качения, удерживающих ролик. Далее тепло распространяется вдоль щупа, образуя температурный градиент. При этом ролик катится по рельсу.

При заклинивании ролик не катится, а скользит по рельсу, испытывая силу трения и, соответственно, разогрев. Количество тепла от роликакратно больше, чем при нормальном режиме работы. Более того, при заклинивании подшипника тепло от разогретой шайбы переходит через устойчивое пятно контакта тел качения подшипника на неподвижное кольцо и дальше вверх. Таким образом, происходит быстрый разогрев остальной части конструкции.

При взгляде на график, складывается впечатление, будто t_2 растет сильнее, чем t_1 . Однако, суть в том, что t_1 перестает греться раньше, чем t_2 , которая продолжает греться «по инерции» в результате выравнивания температур.

Для обучения моделей, предназначенных для определения как проката повышенной температуры, так и заклинивания подшипника, были отобраны сигналы за соответствующие временные периоды. Так как в имеющихся сигналах присутствует только небольшое количество случаев, соответствующих указанным типам аномальных данных, для качественного обучения второстепенных моделей необходимо увеличить размеры обучающих выборок.

Для решения этой задачи можно использовать генеративно-состязательную нейронную сеть TimeGAN [62], специально спроектированную для генерации временных рядов. Для демонстрации работы TimeGAN использован фрагмент сигнала, соответствующий прокату повышенной температуры и представленный на рисунке 12.

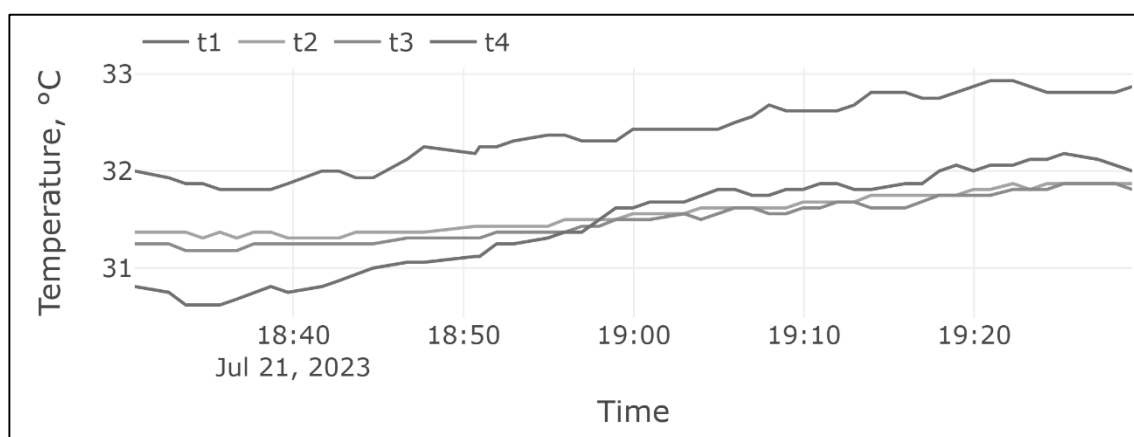


Рисунок 12 – Фрагмент сигнала для обучения TimeGAN

Из представленного фрагмента были сформированы сэмплы, которые составили обучающую выборку для TimeGAN. В результате нейросеть сгенерировала новые сэмплы в количестве, равном количеству элементов обучающей выборки, в данном случае 30 сэмплов. Таким образом были составлены обучающие выборки для второстепенных моделей ансамбля, ко-

торые предназначены для определения, соответствует ли входной сигнал прокату повышенной температуры или заклиниванию подшипника. На рисунке 13 представлен пример сгенерированного сигнала.

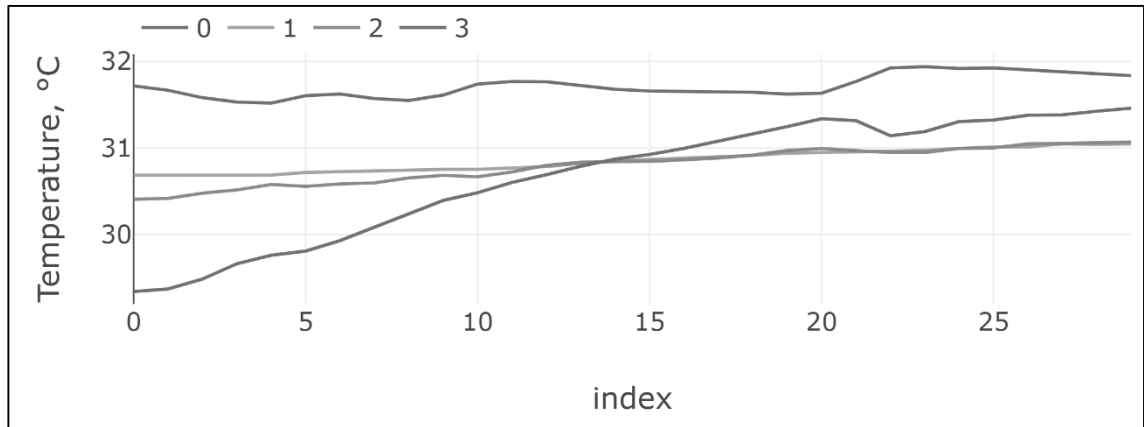


Рисунок 13 – Сгенерированный сигнал

На основе составленных выборок были обучены описанные ранее модели. Фрагмент ошибки реконструкции обученной модели для определения проката повышенной температуры (горячего проката) представлен на рисунке 14, для заклинивания подшипника – на рисунке 15.

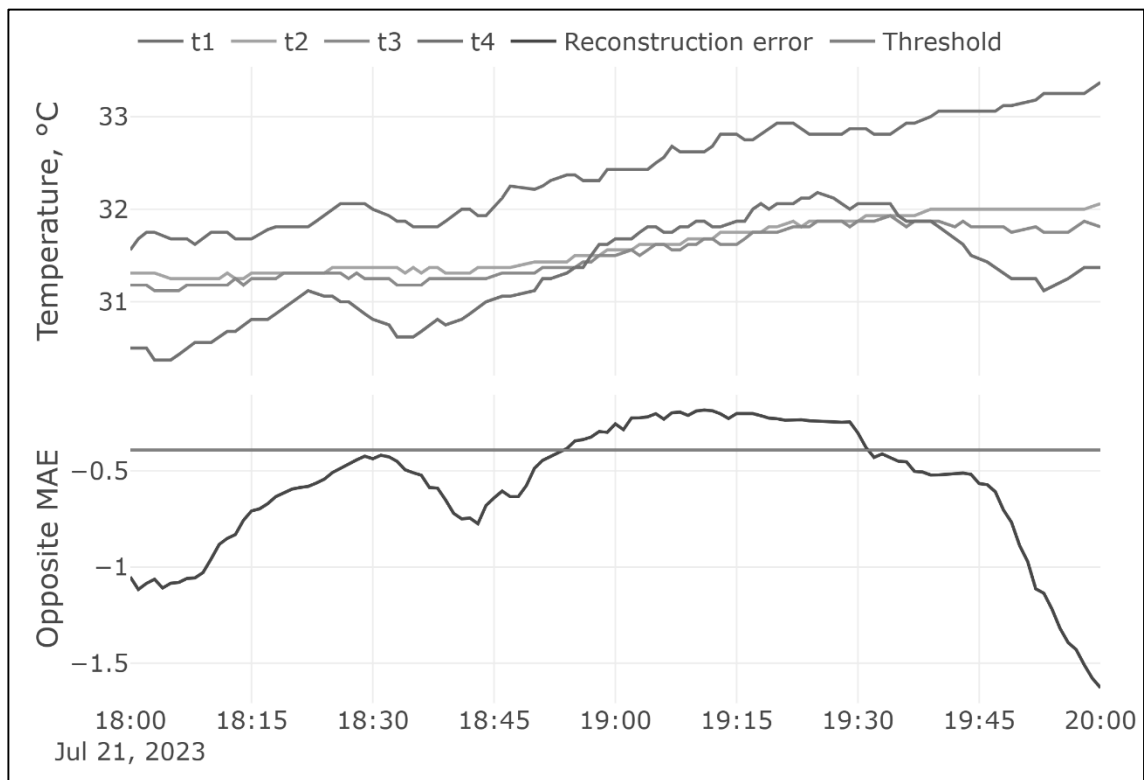


Рисунок 14 – Ошибка реконструкции модели горячего проката

Значения MAE представлены отрицательными для более понятной визуализации, чтобы поставить в соответствие периоды прохождения проката повышенной температуры и превышения ошибкой реконструкции порогового значения. В действительности график ошибки реконструкции должен быть зеркальным с положительными значениями. Вторая модель определяет, что входные данные соответствуют прокату повышенной температуры, если ошибка на этих данных меньше порогового значения. В противном случае, если ошибка на текущем сэмпле больше порогового значения, этот сэмпл подается на вход следующей – третьей – нейросети.

Как видно на графике, значения ошибки «превышают» порог в соответствии с данными, характеризующими прокат повышенной температуры. Таким образом, можно сделать вывод, что модель определения «горячего проката» успешно справляется со своей задачей.



Рисунок 15 – Ошибка реконструкции модели заклинивания подшипника

Модель, определяющая заклинивание также успешно справляется со своей задачей. Для демонстрации работы всего ансамбля рассматривается следующий случай.

В конце тестового периода 20.09.2023 г. ошибка модели начала сильно возрастать. В этот день начала развиваться трещина на центрирующем блоке накопителя, которая привела к отрыву всего блока. В процессе разрушения значение ошибки находилось выше порогового.

На рисунке 16 представлена ошибка реконструкции основной модели 20.09.2023 г. с 12:00 до 21:00, а также соответствующие сигналы сенсоров температуры. Благодаря высокой чувствительности, модель смогла определить развитие дефекта за 4 часа до аварии.

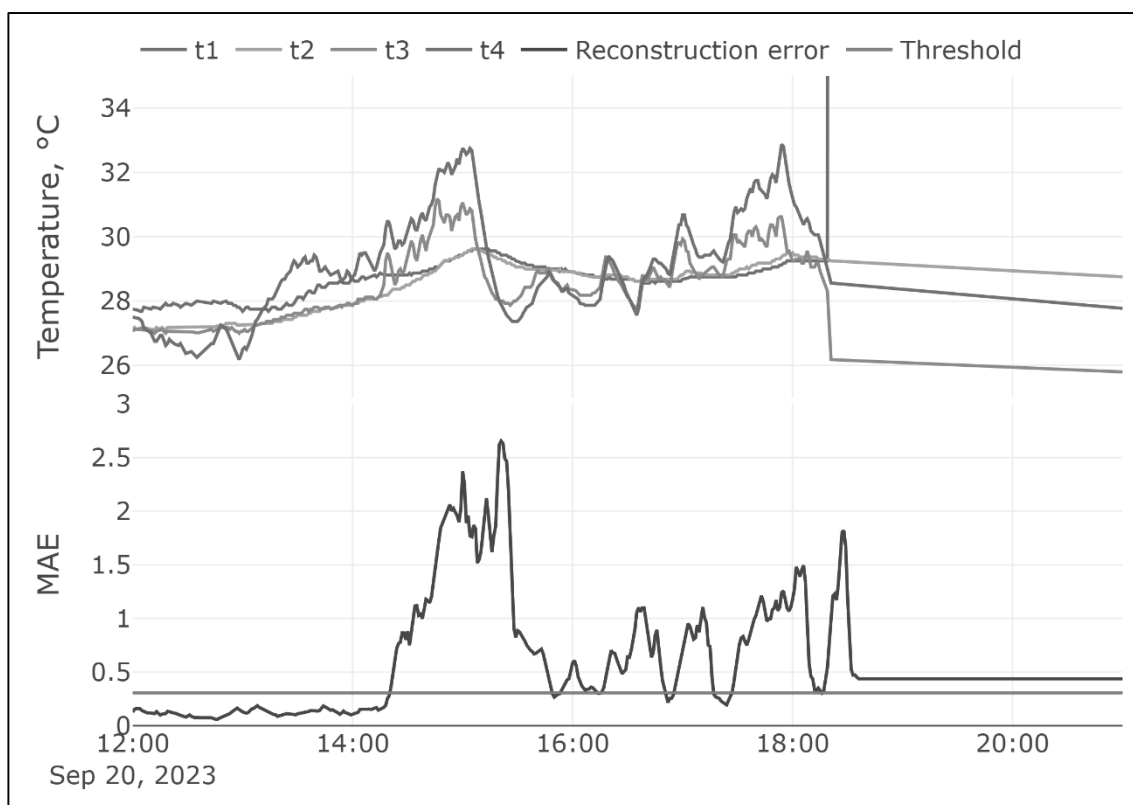


Рисунок 16 – Реакция основной модели на развитие дефекта

Так как ошибка реконструкции основной модели превышает пороговое значение, эти данные были поданы на вход второй модели, определяющей прокат повышенной температуры.

Ошибка реконструкции второй модели 20.09.2023 г. с 12:00 до 21:00, а также соответствующие сигналы сенсоров температуры представлены на рисунке 17.

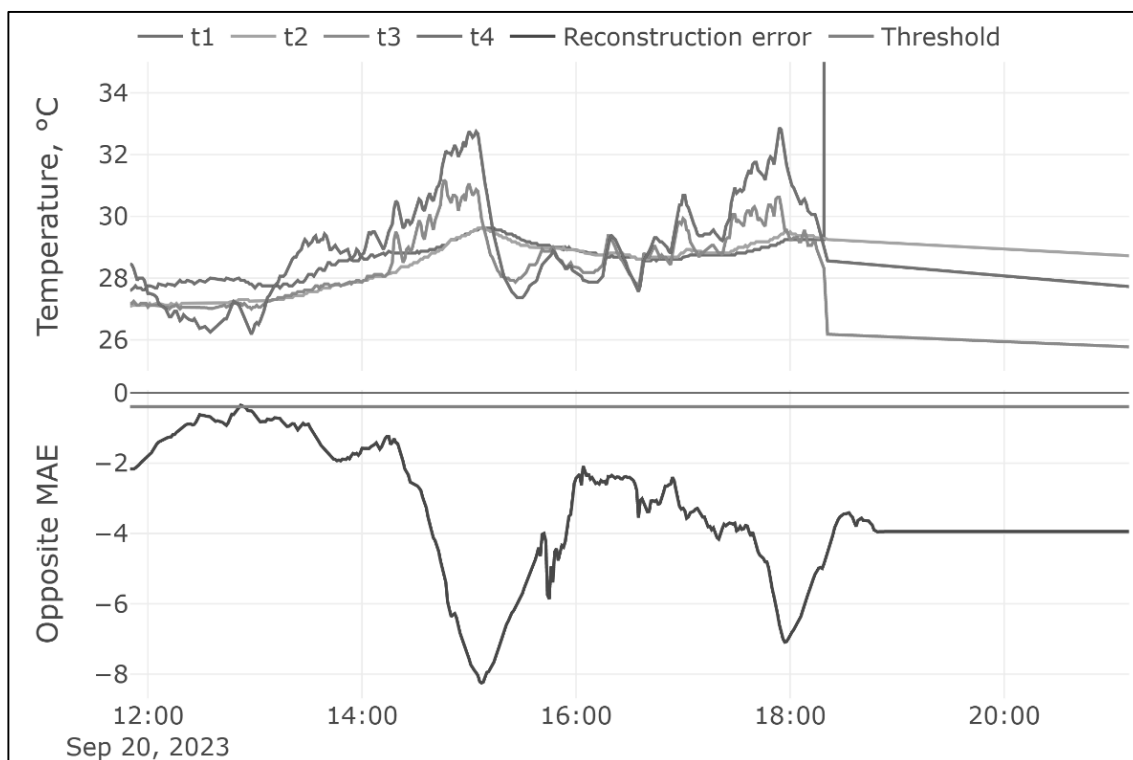


Рисунок 17 – Ошибка второй модели в период развития дефекта

Ошибка реконструкции второй модели не «превысила» пороговое значение. Это означает, что сигналы не соответствуют прокату повышенной температуры. Поэтому данные подаются на вход третьей модели, определяющей заклинивание подшипника.

Ошибка реконструкции третьей модели 20.09.2023 г. с 12:00 до 21:00, а также соответствующие сигналы сенсоров температуры представлены на рисунке 18. Ошибка реконструкции третьей модели не «превысила» пороговое значение. Это означает, что сигналы не соответствуют заклиниванию подшипника. Таким образом, временным меткам, которые соответствуют этим данным, при формировании результата обработки будет назначен статус «аномальные данные».

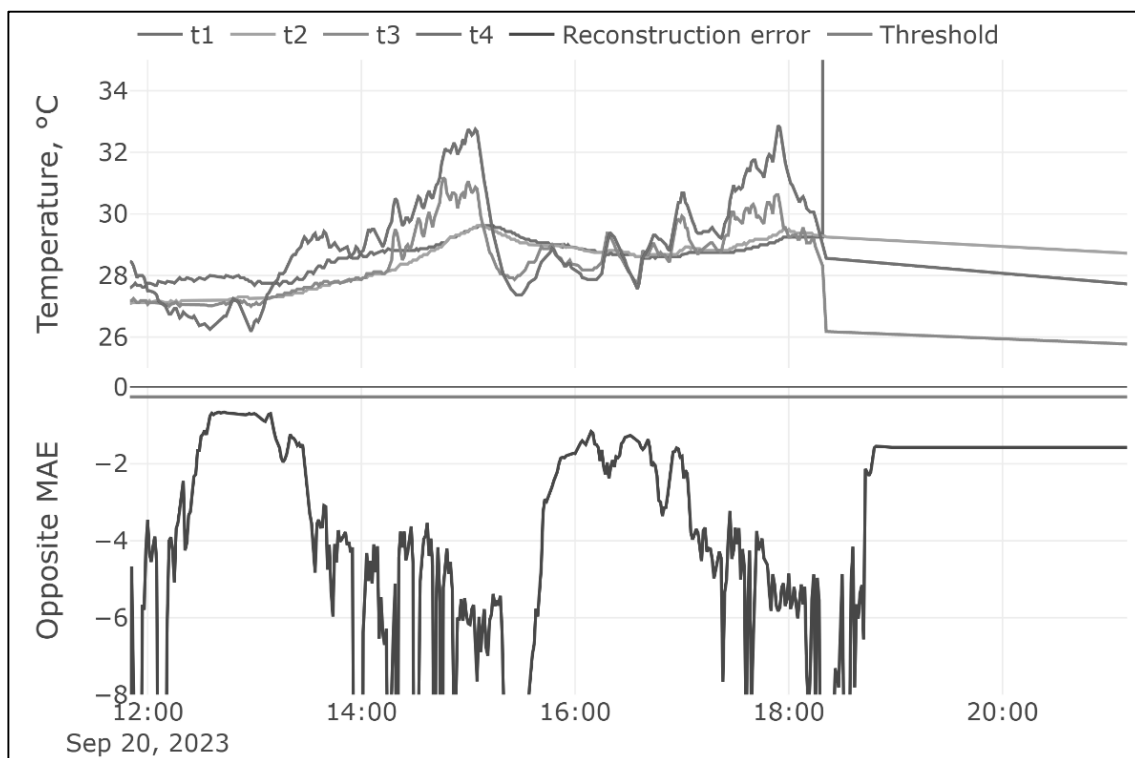


Рисунок 18 – Ошибка третьей модели в период развития дефекта

В дальнейшем, данные, помеченные как «аномальные» могут быть использованы для выделения новых видов аномалий и использованы для расширения и улучшения ансамбля автокодировщиков.

Разработанный в этой главе метод диагностики состояния подшипников по тепловому потоку, включающий в себя алгоритм фильтрации и предобработки входных данных, а также ансамбль из трех автокодировщиков является основой для реализации системы диагностики подшипников центрирующего блока петленакопителя стана холодной прокатки.

3. СИСТЕМА ДИАГНОСТИКИ

3.1. Проектирование системы

Система диагностики реализует метод диагностики, описанный в предыдущей главе, а также взаимодействие с базой данных (БД) металлопрокатного предприятия. Для частного случая реализации системы с целью внедрения на металлопрокатное предприятие были выделены перечисленные ниже требования.

Система должна соответствовать следующим функциональным требованиям.

1. Система должна взаимодействовать с БД для получения из нее сырых данных с датчиков температуры.

2. Система должна взаимодействовать с БД для записи результата диагностики в БД.

3. Система должна проводить предобработку данных.

4. Система должна определять статус входных данных, то есть формировать результат диагностики.

5. Система должна записывать время последней обработки для осуществления непрерывной диагностики в случае нештатного прекращения работы.

Система должна соответствовать следующим нефункциональным требованиям.

1. Программная реализация системы должна быть написана на языке программирования Python.

2. Система должна использовать библиотеку Keras для работы с нейросетевыми моделями.

3. Система должна использовать предобученные нейронные сети для определения статуса входных данных.

4. Система должна использовать «config» файл для определения параметров работы.

Система диагностики представляет собой скрипт, запускаемый на сервере металлопрокатного предприятия в виде фонового процесса – демона (daemon) Linux. Средствами ОС сервера обеспечивается перезапуск скрипта при нештатной остановке работы демона. Алгоритм работы скрипта описан далее.

Скрипт работает постоянно, но итерационно. Между итерациями скрипт находится в «режиме сна» одну минуту. Длительность ожидания связана с частотой получения данных с температурных сенсоров и может быть задана в конфигурационном файле. При запуске скрипт сначала считывает параметры запуска из конфигурационного файла, а затем временную метку последней обработки. Работу скрипта можно разделить на три основных этапа: получение данных, обработка данных и запись результата в базу данных.

На этапе получения данных программа обращается к БД с целью получения сырых данных с датчиков температуры. Для этого сначала определяется обрабатываемый период: с момента последней обработки до времени запуска скрипта. Затем формируется и выполняется запрос к БД. В случае возникновения каких-либо ошибок, например, при отсутствии возможности подключения к БД или недостаточном для обработки количестве данных, текущая итерация прерывается.

На этапе обработки данных о тепловом потоке при успешном их получении на предыдущем этапе, они проходят предобработку: фильтруются с помощью описанного ранее медианного фильтра, формируются признаки для моделей автокодировщиков, проводится Z-нормализация, а также прочие необходимые преобразования. Далее подготовленные данные подаются на вход ансамблю автокодировщиков для дальнейшей обработки. В зависимости от результатов работы моделей ансамбля, в конце данного этапа на выходе формируется результат обработки, в котором каждой временной метке, соответствующей обработанному сэмплу, присваивается одно из следующих возможных значений статуса:

- «нормальные данные»;
- «прокат повышенной температуры»;
- «заклинивание подшипника»;
- «аномальные данные».

На последнем этапе формируется соответствующий запрос, и полученный результат обработки записывается в БД. После этого происходит запись времени последней обработки.

Для графической демонстрации одной итерации алгоритма работы скрипта была построена диаграмма деятельности (рисунок 19).



Рисунок 19 – Диаграмма деятельности одной итерации работы скрипта

3.2. Реализация системы

Для реализации системы диагностики был использован язык программирования Python версии 3.10.0, а также среда разработки Jupyter Lab версии 4.1.4. Помимо этого, был использован ряд библиотек. Основные библиотеки перечислены далее:

- JSON 3.12.3 [63] – библиотека, реализующая взаимодействие с файлами соответствующего формата; в данной программной реализации используется для работы с файлом конфигурации и файлом, содержащем временную метку последней обработки;
- Keras 2.10.0 [60] – библиотека для работы с нейронными сетями;
- Pandas 2.2.1 [64] – библиотека для работы с данными;
- Scikit-learn 1.4.1 [65] – библиотека, реализующая многие функции и модели из области машинного обучения; в данной программной реализации используется функция Z-нормализации данных;
- SQLAlchemy 2.0.28 [66] – библиотека для взаимодействия с базами данных.

Система включает в себя следующие файлы-компоненты:

- скрипт, реализующий метод диагностики и взаимодействие с БД;
- предобученные модели ансамбля;
- файл с коэффициентами, необходимыми для Z-нормализации входных данных в процессе предобработки;
- конфигурационный файл;
- файл, хранящий временную метку последней обработки;
- прочие вспомогательные файлы.

Так как подробное описание метода диагностики, включая детали реализации, приведено в предыдущей главе, далее будут описаны детали реализации взаимодействия системы диагностики с БД, реализация логирования, а также процесс создания демона Linux.

Подключение к БД

Данные с датчика температуры хранятся в БД. Подключение и отправка запроса к базе данных для получения данных производится с помощью функции `read_sql()` библиотеки `Pandas`. Для этого необходимо передать в указанную функцию как минимум два объекта: запрос к БД и объект соединения с БД. Запрос к БД составляется в текстовом формате на языке `SQL`. Объект соединения с БД создается с помощью функции `create_engine()` из библиотеки `SQLAlchemy`.

Пример кода для получения данных с датчика температуры представлен на рисунке 20. Пример кода для записи результата обработки в БД представлен на рисунке 21.

```
##### Getting configs
db_ip    = config['db_ip']
db_port  = config['db_port']

##### Get processing period
period_start, period_end = get_processing_period(config)

##### Query to DB for getting sensor data
query = f"""
SELECT  time, t1, t2, t3, t4
FROM    sens.temp_lora
WHERE   time > '{period_start}'
AND     time < '{period_end}'
AND     id = {sensor_num}
"""

##### Creating SQLAlchemy engine
engine = create_engine(f'postgresql://iba@{db_ip}:{db_port}/iba_stan_mc')

##### Getting sensor data from database
try:
    df = pd.read_sql(sql=query, con=engine, index_col='time')
    return df
except Exception as err:
    logging(f'Unable to connect to the database\n{err}\nAnother try...')
    return pd.DataFrame()
```

Рисунок 20 – Пример подключения к БД для получения данных

```

#### Getting configs
db_ip      = config['db_ip']
db_port    = config['db_port']
table_name = config['table_name']
tag        = config['tag_' + sensor_num]

#### Creating SQLAlchemy engine
engine = create_engine(f'postgresql://iba@{db_ip}:{db_port}/iba_stan_mc')

#### Writing status in database
for idx in result_df.index:
    query = f"""INSERT INTO {table_name} VALUES
              ({tag}', '{idx}', {result_df.loc[idx, 'status']})"""
    try:
        with engine.begin() as conn:
            conn.execute(text(query))
    except Exception as err:
        logging(f'Failure when writing the result to the data-
base\n{err}\n')
        continue

```

Рисунок 21 – Пример подключения к БД для записи результата обработки

Создание демона

Для запуска демона в Linux был использован менеджер служб (сервисов) systemd. Сначала необходимо создать unit-файл (модуль) и добавить в него описание создаваемого сервиса. Модули, создаваемые администратором, содержатся в директории `/etc/systemd/system`. Был создан файл «test_script.service». Пример содержания такого файла представлен на рисунке 22. После создания модуля необходимо перезагрузить systemd с помощью команды `systemctl daemon-reload`, включить автозапуск сервиса командой `systemctl enable test_script.service`, а затем запустить сервис командой `systemctl start test_script`.

```

[Unit]
Description=Diagnostic script

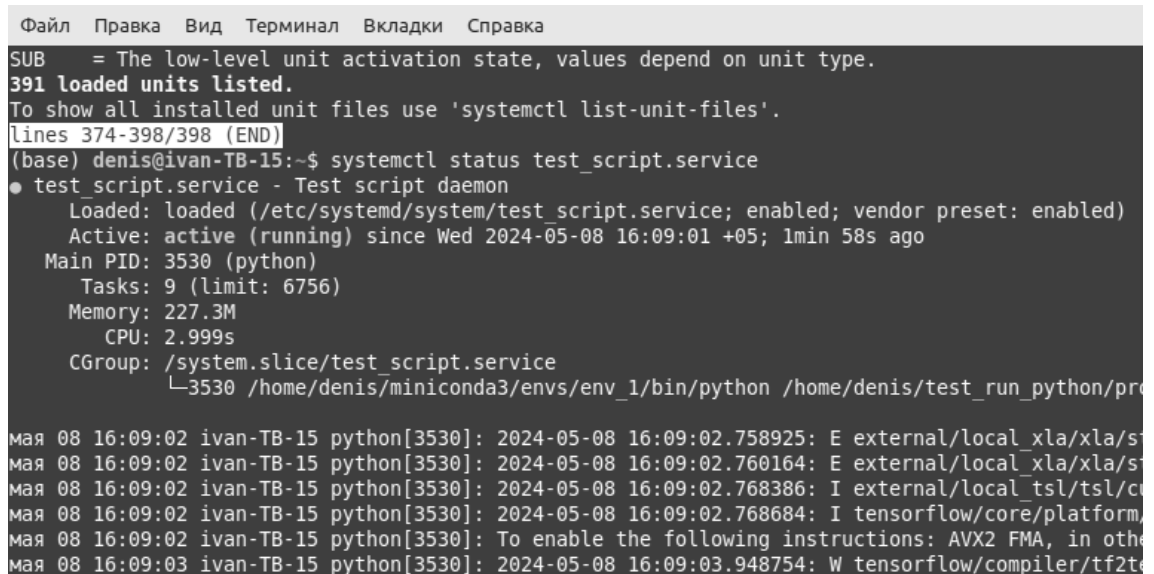
[Service]
ExecStart=python /home/diagnostic_script/script.py
WorkingDirectory=/home/diagnostic_script
Restart=always

[Install]
WantedBy=multi-user.target

```

Рисунок 22 – Содержание созданного модуля

Проверить статус созданного сервиса можно с помощью команды `systemctl status test_script.service` (рисунок 23).

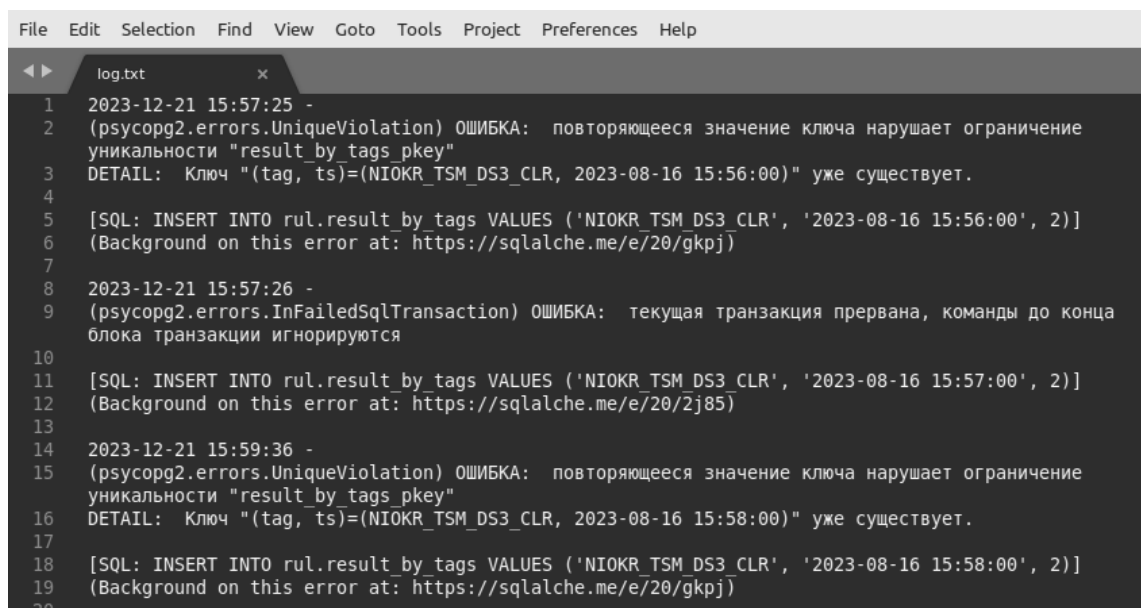


```
Файл  Правка  Вид  Терминал  Вкладки  Справка
SUB    = The low-level unit activation state, values depend on unit type.
391 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'.
lines 374-398/398 (END)
(base) denis@ivan-TB-15:~$ systemctl status test_script.service
● test_script.service - Test script daemon
   Loaded: loaded (/etc/systemd/system/test_script.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-05-08 16:09:01 +05; 1min 58s ago
     Main PID: 3530 (python)
        Tasks: 9 (limit: 6756)
       Memory: 227.3M
          CPU: 2.999s
      CGroup: /system.slice/test_script.service
             └─3530 /home/denis/miniconda3/envs/env_1/bin/python /home/denis/test_run_python/pr
мая 08 16:09:02 ivan-TB-15 python[3530]: 2024-05-08 16:09:02.758925: E external/local_xla/xla/st
мая 08 16:09:02 ivan-TB-15 python[3530]: 2024-05-08 16:09:02.760164: E external/local_xla/xla/st
мая 08 16:09:02 ivan-TB-15 python[3530]: 2024-05-08 16:09:02.768386: I external/local_tsl/tsl/c
мая 08 16:09:02 ivan-TB-15 python[3530]: 2024-05-08 16:09:02.768684: I tensorflow/core/platform
мая 08 16:09:02 ivan-TB-15 python[3530]: To enable the following instructions: AVX2 FMA, in othe
мая 08 16:09:03 ivan-TB-15 python[3530]: 2024-05-08 16:09:03.948754: W tensorflow/compiler/tf2t
```

Рисунок 23 – Проверка статуса созданного сервиса

Логирование

В системе предусмотрен механизм логирования. В коде скрипта в определенных местах вызывается функция `logging()`, которая записывает переданное ей сообщение в журнал логов (рисунок 24). Код функции представлен на рисунке 25.



```
File  Edit  Selection  Find  View  Goto  Tools  Project  Preferences  Help
log.txt
1  2023-12-21 15:57:25 -
2  (psycopg2.errors.UniqueViolation) ОШИБКА: повторяющееся значение ключа нарушает ограничение
   уникальности "result_by_tags_pkey"
3  DETAIL: Ключ "(tag, ts)=(NIOKR_TSM_DS3_CLR, 2023-08-16 15:56:00)" уже существует.
4
5  [SQL: INSERT INTO rul.result_by_tags VALUES ('NIOKR_TSM_DS3_CLR', '2023-08-16 15:56:00', 2)]
6  (Background on this error at: https://sqlalche.me/e/20/gkpkj)
7
8  2023-12-21 15:57:26 -
9  (psycopg2.errors.InFailedSqlTransaction) ОШИБКА: текущая транзакция прервана, команды до конца
   блока транзакции игнорируются
10
11 [SQL: INSERT INTO rul.result_by_tags VALUES ('NIOKR_TSM_DS3_CLR', '2023-08-16 15:57:00', 2)]
12 (Background on this error at: https://sqlalche.me/e/20/2j05)
13
14 2023-12-21 15:59:36 -
15 (psycopg2.errors.UniqueViolation) ОШИБКА: повторяющееся значение ключа нарушает ограничение
   уникальности "result_by_tags_pkey"
16 DETAIL: Ключ "(tag, ts)=(NIOKR_TSM_DS3_CLR, 2023-08-16 15:58:00)" уже существует.
17
18 [SQL: INSERT INTO rul.result_by_tags VALUES ('NIOKR_TSM_DS3_CLR', '2023-08-16 15:58:00', 2)]
19 (Background on this error at: https://sqlalche.me/e/20/gkpkj)
20
```

Рисунок 24 – Пример содержания журнала логов


```
def logging(msg):
    with open('log.txt', 'a') as file:
        file.write(str(datetime.now()).split('.')[0]+' - '+msg+'\n')
```

Рисунок 25 – Функция логирования

3.3. Тестирование системы

Результаты тестирования разработанной системы диагностики представлены в таблице 1.

Таблица 1 – Результаты тестирования системы диагностики

Номер теста	Входные данные	Ожидаемый результат	Полученный результат
1	Система отправляет запрос к БД для получения данных.	Система получает из БД сигналы датчиков температуры за указанный в запросе период.	Совпадает с ожидаемым.
2	Система отправляет запрос к БД для записи результата обработки.	Система успешно записывает в БД результат обработки и завершает текущую итерацию работы скрипта.	Совпадает с ожидаемым.
3	Системе не удается подключиться к БД.	Система завершает текущую итерацию работы скрипта.	Совпадает с ожидаемым.
4	Система получает недостаточное количество данных для обработки.	Система завершает текущую итерацию работы скрипта.	Совпадает с ожидаемым.
5	Система не обнаруживает один из файлов-компонентов.	Система завершает текущую итерацию работы скрипта.	Совпадает с ожидаемым.
6	В системе возникает обрабатываемая ошибка.	Система делает запись в журнал логов и завершает текущую итерацию работы скрипта.	Совпадает с ожидаемым.
7	В системе возникает необрабатываемая ошибка.	Система перезапускается средствами менеджера служб.	Совпадает с ожидаемым.

Система была протестирована на наборе, состоящем из штатных и нештатных ситуаций. Все тесты были успешно пройдены, система функционирует ожидаемым образом.

ЗАКЛЮЧЕНИЕ

В рамках данной работы была разработана система диагностики состояния подшипников технологического оборудования металлопрокатного производства на основе интеллектуальной обработки данных о тепловом потоке, представленных в виде многомерных временных рядов. При этом были решены следующие задачи.

1. Проведен обзор методов диагностики состояния подшипников технологического оборудования.

2. Разработан метод интеллектуальной обработки сигналов с датчика температуры для диагностики состояния подшипников.

3. Реализована система диагностики состояния подшипников технологического оборудования на основе разработанного метода.

В настоящий момент производится подготовка разработанного решения к дальнейшему внедрению на металлопрокатное предприятие. В будущем планируется продолжать разработку и улучшение системы диагностики, в частности, улучшение качества работы нейросетевых моделей, а также оптимизация алгоритмов обработки.

Результаты данной работы будут представлены на XXXIV Международном научном симпозиуме «Метрология и метрологическое обеспечение 2024», который состоится с 7-го по 11-ое сентября 2024 года в городе Созополь, Болгария.

ЛИТЕРАТУРА

1. After the Fall: The Costs, Causes & Consequences of Unplanned Downtime. [Электронный ресурс] URL: <https://www.techinnews.com/fall-costs-causes-consequences-unplanned-downtime> (дата обращения: 25.04.2024 г.).
2. Lemes L.C., Hvam L. Maintenance Costs in the Process Industry: A Literature Review. // Proceedings of the 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Macao, China, December 15–18, 2019. IEEE, 2020. – С. 1481–1485.
3. Predictive Maintenance. [Электронный ресурс] URL: <https://upkeep.com/learning/predictive-maintenance> (дата обращения: 25.04.2024 г.).
4. Cakir M., Guvenc M.A., Mistikoglu S. The experimental application of popular machine learning algorithms on predictive maintenance and the design of IIoT based condition monitoring system. // Computers & Industrial Engineering, 2021. – 14 с.
5. Sazzadur Rahman Md., Tapotosh G., Nahid Ferdous A., Kaiser M.S., Mehrin A., Sanwar Hosen A.S.M. Machine learning and internet of things in industry 4.0: A review. // Measurement: Sensors, 2023. – 19 с.
6. Tanane B., Bentaha M.L., Dafflon B., Ferreiro V., Moalla N. Toward an Adaptive Approach to Implement Predictive Maintenance in Metallurgical Industry SMEs Through IoT and AI. // Springer International Publishing, 2022. – 12 с.
7. Sun S., Shen C., Wang D. Editorial for Special Issue: Machine Health Monitoring and Fault Diagnosis Techniques. // Sensors, 2023. – 3493 с.
8. Zhang S., Zhang S., Wang B., Habetler T. G. Deep Learning Algorithms for Bearing Fault Diagnostics—A Comprehensive Review. // IEEE Access, 2020. – 24 с.
9. Duy-Tang H., Hee-Jun K. A survey on Deep Learning based bearing fault diagnosis. // Neurocomputing, 2019. – С. 327–335.

10. Chen X., Yang R., Xue Y., Huang M., Ferrero R., Wang Z. Deep Transfer Learning for Bearing Fault Diagnosis: A Systematic Review Since 2016. // *IEEE Transactions on Instrumentation and Measurement*, 2023. – 21 c.
11. Peng B., Bi Y., Xue B., Zhang M., Wan S. A Survey on Fault Diagnosis of Rolling Bearings. // *Algorithms*, 2022. – 347 c.
12. Althubaiti A., Elasha F., Teixeira J. A. Fault diagnosis and health management of bearings in rotating equipment based on vibration analysis – a review. // *Journal of Vibroengineering*, 2021. – C. 46–74.
13. Zhirkin Y.V., Puzik E.A., Filatov A.A., Sultanov N.L. Prolonging the Service Life of the Rolling Bearings of the Work Rolls of the 2000 Tandem Cold-Rolling Mill at the Magnitogorsk Metallurgical Combine. // *Metallurgist*, 2017. – C. 1180–1182.
14. Mukhopadhyay G., Bhattacharya S. Failure Analysis of a Cylindrical Roller Bearing from a Rolling Mill. // *Journal of Failure Analysis and Prevention*, 2011. – C. 337–343.
15. Xu F, Ding N., Li N., Liu L., Hou N., Xu N., Guo W., Tian L., Xu H., Wu C.-M. L., Wu X., Chen X. A review of bearing failure Modes, mechanisms and causes. // *Engineering Failure Analysis*, 2023. – 33 c.
16. Lin S., Sun J., Peng Y. Analysis of axial fretting mode and mechanical model for a four-row tapered roller bearing. // *International Journal of Mechanical Sciences*, 2022. – 15 c.
17. Hou X., Diao Q., Liu Y., Liu C., Zhang Z., Tao C. Failure Analysis of a Cylindrical Roller Bearing Caused by Excessive Tightening Axial Force. // *Machines*, 2022. – 322 c.
18. Li J, Chen W., Xie Y. Experimental study on skid damage of cylindrical roller bearing considering thermal effect. // *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 2014. – 10 c.
19. Wu G., Yan T., Yang G., Chai H., Cao C. A Review on Rolling Bearing Fault Signal Detection Methods Based on Different Sensors. // *Sensors*, 2022. – 8330 c.

20. Touret T., Changenet C., Ville F., Lalmi M., Becquerelle S. On the use of temperature for online condition monitoring of geared systems – A review. // *Mechanical Systems and Signal Processing*, 2018. – C. 197–210.
21. Wang W., Taylor J., Rees R.J. Recent Advancement of Deep Learning Applications to Machine Condition Monitoring Part 1: A Critical Review. // *Acoustics Australia*, 2021. – C. 207–219.
22. Sun Y., Zhang C., Zhao X., Liu X., Lu C., Fei J. Transient Thermal Analysis Model of Damaged Bearing Considering Thermo-Solid Coupling Effect. // *Sensors*, 2022. – 8171 c.
23. Childs P.R.N. *Practical Temperature Measurement*. // Butterworth-Heinemann, 2001. – 372 c.
24. Choudhary A., Mian T., Fatima S. Convolutional neural network based bearing fault diagnosis of rotating machine using thermal images. // *Measurement*, 2021. – 7 c.
25. Li Y., Du X., Wan F., Wang X., Yu H. Rotating machinery fault diagnosis based on convolutional neural network and infrared thermal imaging. // *Chinese Journal of Aeronautics*, 2020. – C. 427–438.
26. Shao H., Xia M., Han G., Zhang Y., Wan J. Intelligent Fault Diagnosis of Rotor-Bearing System Under Varying Working Conditions With Modified Transfer Convolutional Neural Network and Thermal Images. // *IEEE Transactions on Industrial Informatics*, 2021. – 8 c.
27. Xin L., Haidong S., Hongkai J., Jiawei X. Modified Gaussian convolutional deep belief network and infrared thermal imaging for intelligent fault diagnosis of rotor-bearing system under time-varying speeds. // *Structural Health Monitoring*, 2022. – C. 339–353.
28. Shao H., Li W., Xia M., Zhang Y., Shen C., Williams D., Kennedy A., de Silva C.W. Fault Diagnosis of a Rotor-Bearing System Under Variable Rotating Speeds Using Two-Stage Parameter Transfer and Infrared Thermal Images. // *IEEE Transactions on Instrumentation and Measurement*, 2021. – 11 c.

29. Choudhary A., Goyal D., Letha S.S. Infrared Thermography-Based Fault Diagnosis of Induction Motor Bearings Using Machine Learning. // IEEE Sensors Journal, 2021. – С. 1727–1734.
30. Jakubek B., Grochalski K., Rukat W., Sokol H. Thermovision measurements of rolling bearings. // Measurement, 2022. – 13 с.
31. Thuan N.D., Dong T.P., Nguyen H.T., Hoang H.S. Efficient bearing fault diagnosis with neural network search and parameter quantization based on vibration and temperature. // Engineering Research Express, 2023. – 11 с.
32. Kumar M.P., Rao C.J. Structural and Thermal Analysis on a Tapered Roller Bearing. // International Journal of Innovative Science Engineering and Technology, 2015. – С. 502–511.
33. ADB Series Bearing Temperature Sensors. [Электронный ресурс] URL: <https://go4b.co.uk/products/electronic-monitoring-equipment/bearing-temperature-sensors/adb-series-bearing-sensors/253> (дата обращения: 25.04.2024 г.).
34. Earing Temperature Monitoring. [Электронный ресурс] URL: <https://www.mullerbeltex.com/process-monitoring-systems/bearing-temperature-monitoring> (дата обращения: 25.04.2024 г.).
35. Xu J., Zhang J., Huang Z., Wang L. Calculation and finite element analysis of the temperature field for high-speed rail bearing based on vibrational characteristics. // Journal of Vibroengineering, 2015. – С. 720–732.
36. Wang A., Wang J. Temperature distribution and scuffing of tapered roller bearing. // Chinese Journal of Mechanical Engineering, 2014. – 7 с.
37. Ai S., Wang W., Wang Y., Zhao Z. Temperature rise of double-row tapered roller bearings analyzed with the thermal network method. // Tribology International, 2015. – С. 11–22.
38. Yan K., Wang N., Zhai Q., Zhu Y., Zhang J., Niu Q. Theoretical and experimental investigation on the thermal characteristics of double-row tapered roller bearings of high speed locomotive. // International Journal of Heat and Mass Transfer, 2015. – С. 1119–1130.

39. Li J., Xue J., Ma Z. Study on the Thermal Distribution Characteristics of High-Speed and Light-Load Rolling Bearing Considering Skidding. // Applied Sciences, 2018. – 1593 c.
40. Gao P., Tang W., Cui Y., Wang Y., Mo G., Yin J. Theoretical and Experimental Investigation on Thermal Characteristics of Railway Double-Row Tapered Roller Bearing. // Energies, 2022. – 4217 c.
41. Truong D.S., Kim B.-S., Ro S.-K. An analysis of a thermally affected high-speed spindle with angular contact ball bearings. // Tribology International, 2021. – 15 c.
42. Wang Y., Cao J., Tong Q., An G., Liu R., Zhang Y., Yan H. Study on the Thermal Performance and Temperature Distribution of Ball Bearings in the Traction Motor of a High-Speed EMU. // Applied Sciences, 2020. – 4373 c.
43. Cui Y., Gao P., Tang W., Mo G., Yin J. Adaptive Thin Film Temperature Sensor for Bearing's Rolling Elements Temperature Measurement. // Sensors, 2022. – 2838 c.
44. Glavatskih S.B. A method of temperature monitoring in fluid film bearings. // Tribology International, 2004. – C. 143–148.
45. Sokolov N.V., Khadiev M.B., Fedotov P.E., Fedotov E.M. Influence of the Lubricant's Supply Temperature on the Operation of a Fluid Film Thrust Bearing. // Russian Engineering Research, 2023. – C. 264–271.
46. Wang Z., Liu C., Yan F. Condition monitoring of wind turbine based on incremental learning and multivariate state estimation technique. // Renewable Energy, 2022. – C. 343–360.
47. Chesterman X., Verstraeten T., Daems P. Overview of normal behavior modeling approaches for SCADA-based wind turbine condition monitoring demonstrated on data from operational wind farms. // Wind Energy Science, 2023. – C. 893–924.
48. Chesterman X., Verstraeten T., Daems P. Condition Monitoring of Wind Turbines and Extraction of Healthy Training Data using an Ensemble of

Advanced Statistical Anomaly Detection Models. // Annual Conference of the PHM Society, 2021. – 12 c.

49. Chesterman X., Verstraeten T., Daems P. The detection of generator bearing failures on wind turbines using machine learning based anomaly detection. // Journal of Physics: Conference Series, 2022. – 11 c.

50. Castellani F., Astolfi D., Natili F. SCADA Data Analysis Methods for Diagnosis of Electrical Faults to Wind Turbine Generators. // Applied Sciences, 2021. – 3307 c.

51. Maron J., Anagnostos D., Brodbeck B. Artificial intelligence-based condition monitoring and predictive maintenance framework for wind turbines. // Journal of Physics: Conference Series, 2022. – 9 c.

52. Beretta M., Julian A., Sepulveda J., Cusidó J., Porro O. An Ensemble Learning Solution for Predictive Maintenance of Wind Turbines Main Bearing. // Sensors, 2021. – 21 c.

53. Black I., Cevasco D., Kolios A. Deep Neural Network Hard Parameter Multi-Task Learning for Condition Monitoring of an Offshore Wind Turbine. // Journal of Physics: Conference Series, 2022. – 10 c.

54. Jamil F., Verstraeten T., Nowé A. A deep boosted transfer learning method for wind turbine gearbox fault detection. // Renewable Energy, 2022. – C. 331–341.

55. Miele E., Bonacina F., Corsini A. Deep anomaly detection in horizontal axis wind turbines using Graph Convolutional Autoencoders for Multivariate Time series. // Energy and AI, 2022. – 13 c.

56. Chen H., Liu H., Chu X. Anomaly detection and critical SCADA parameters identification for wind turbines based on LSTM-AE neural network. // Renewable Energy, 2021. – C. 829–840.

57. Chen Z., Zhu H., Wu J., Fan L. Health indicator construction for degradation assessment by embedded LSTM–CNN autoencoder and growing self-organized map. // Knowledge-Based Systems, 2022. – 13 c.

58. Zhang A., Zhao X., Wang L. CNN and LSTM based Encoder-Decoder for Anomaly Detection in Multivariate Time Series. // In Proceedings of the 2021 IEEE 5th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Xi'an, China, October 15–17, 2021. IEEE, 2021. – С. 571–575.
59. Haputhanthri D, Wijayasiri A. Short-Term Traffic Forecasting using LSTM-based Deep Learning Models. // 2021 Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, 2021. IEEE, 2021. – С. 602–607.
60. Keras: Deep Learning for humans. [Электронный ресурс] URL: <https://keras.io> (дата обращения: 25.04.2024 г.).
61. Why do physicists mention «five sigma» in their results? [Электронный ресурс] URL: <https://home.cern/resources/faqs/five-sigma> (дата обращения: 25.04.2024 г.).
62. Yoon J., Jarrett D., van der Schaar M. Time-series Generative Adversarial Networks. // Advances in Neural Information Processing Systems, 2019. – 11 с.
63. Json – JSON encoder and decoder. [Электронный ресурс] URL: <https://docs.python.org/3/library/json.html> (дата обращения: 25.04.2024 г.).
64. Pandas – Python Data Analysis Library. [Электронный ресурс] URL: <https://pandas.pydata.org> (дата обращения: 25.04.2024 г.).
65. Scikit-learn: machine learning in Python. [Электронный ресурс] URL: <https://scikit-learn.org> (дата обращения: 25.04.2024 г.).
66. SQLAlchemy – The Database Toolkit for Python. [Электронный ресурс] URL: <https://www.sqlalchemy.org> (дата обращения: 25.04.2024 г.).