

ВВЕДЕНИЕ

Нейронные сети представляю собой мощный инструмент для решения разного рода задач. Ещё в конце пятидесятих годов было доказано, что нейронные сети, потенциально, способны решить любую задачу. С развитием технологий и вычислительных мощностей стал полноценно раскрываться потенциал нейронных сетей, как мощного инструмента решения задач разного плана. Обученные нейронные сети способны классифицировать продукты производства и определять их дальнейшую судьбу, способны предсказывать поведение некоторой системы для дальнейшего планирования производства, искать аномалии в данных технологического процесса и многое другое.

Данная работа как раз и посвящена прогнозированию данных технологического процесса производственного оборудования, а именно пастеризационной установки. Для выполнения данной задачи будут использоваться нейронные сети, поскольку решение задачи прогнозирования именно нейросетевым способом на сегодняшний день является одним из самых актуальных, эффективных, точных, изученных и разнообразных способов решения данной задачи. Мало того, что такой способ позволяет гибко адаптироваться к процессам и научиться достаточно точно их прогнозировать, так ещё и такой способ предусматривает возможность, так скажем, подстраивания, благодаря чему, можно улавливать новые особенности процесса. К тому же, существует большое количество различной литературы про прогнозирование с помощью нейронных сетей, что даёт уверенность в том, что такой способ обеспечивает необходимыми результатами.

По большей части, данная работа является смесью задачи прогнозирования и машинного обучения, поскольку в её рамках будет рассматриваться изучение предметной области и определение задач и целей прогнозирования, сбор, анализ и обработка данных технологического процесса пастеризационной установки, их подготовка к прогнозированию и обучению нейронной сети, после проектирование и реализация самой нейронной сети, её обучение, тестирование и оценка, а в заключительном этапе готовую модель необходимо развернуть для взаимодействия с пастеризационной установкой.

Дойдя до этапа создания модели, будут выбраны основные архитектуры нейронных сетей, подробно описаны их способы функционирования и обучения, а также детально будут рассмотрены результаты работы моделей и их сравнение между собой.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТАНОВКА ЗАДАЧИ

1.1 Основные понятия предметной области

Объектом исследования и внедрения нейронной сети является пастеризационная установка. Для понимания её технологического процесса, необходимо для начала понять с чем она работает и что она производит, а также какими функциями обладает пастеризационная установка. Поэтому в данной главе мы будем рассматривать пастеризационную установку как некий «серый ящик»: опишем объекты, которые попадают на вход и выходят из пастеризационной установки, а также опишем самые основные процессы, которые происходят с подаваемым на вход объектом.

И так, основной объект обработки пастеризационной установки является молоко. Поэтому в дальнейшем молоко будет рассматриваться как объект технической обработки. Рассматривая его таким образом, мы понимаем, что оно должно обладать некоторыми показателями, например, состав молока, степень чистоты, кислотность, наличие токсичных и нейтрализующих веществ. При этом молоко обладает ещё и различными свойствами: органолептическими, физико-механическими и биохимическими. И так, разберём лишь те параметры и свойства, которые будут нам в дальнейшем интересны.

Молоко можно разделить на две составляющих: вода и распределённые в этой воде пищевые вещества. К таким веществам относят жиры, белки, углеводы, ферменты, различные минеральные вещества и газы. Помимо этого, в молоке могут находиться различные микроорганизмы. И как известно, некоторые из этих микроорганизмов, содержащиеся в молоке, являются опасными или вредными, например, бруцеллеза, ящура, возбудитель кишечной палочки и другие. Но как избавиться от вредных и опасных микроорганизмов? Для этого используется процесс пастеризации – уничтожение различных форм вредных и опасных микроорганизмов в молоке. Но при этом молоко должно сохранить свою биологическую и питательную ценность, а также и своё качество.

Однако перед тем, как перейти к пастеризации, необходимо определить, какое молоко можно пастеризовать, каким требованиям оно должно соответствовать. Для определения подходящего перед пастеризацией молока существует множество различных показателей и требований. Так, например, пригодное для пастеризации молоко должно быть кислотностью не более 22 °Т, а бактериальная обсеменённость молока должна быть один миллион клеток на сантиметр кубический. При этом, молоко не должно быть вспененным. Перед пастеризацией, молоко должно быть также предварительно очищено на фильтрах или на сепараторах-молокоочистителях. Что ж, основные требования перечислены.

А теперь к самой пастеризации. Основные параметры пастеризации есть температура пастеризации, а также время выдержки, т.е. время нахождения молока в данном процессе. Относительно данных параметров существует формула пастеризации, выведенная Г. А. Куком и называемая критерием Пастера:

$$P = \frac{t}{\tau} \quad (1.1)$$

где t – время действия температуры пастеризации, с;

τ – время бактерицидного действия температуры пастеризации, с.

Что такое бактерицидное действие температуры пастеризации? Это, как раз и есть

					ДП.АС59.200043-05 81 00	Лист 5
Изм.	Лист.	№ докум.	Подп.	Дата		

эффект, в результате которого происходит уничтожение вредных и опасных микроорганизмов.

Также известна и ещё одна немаловажная зависимость: продолжительность выдержки зависит от температуры пастеризации. Это можно увидеть в следующем выражении:

$$\ln t = 36.84 - 0.48T \quad (1.2)$$

где T – температура пастеризации, °C.

Завершение процесса пастеризации характеризуется полным уничтожением содержащихся в молоке микроорганизмов. Это можно будет определить благодаря уже известному критерию Пастера, значение которого должно быть не меньше единицы, для того чтобы считать, что процесс пастеризации завершён.

Перед тем, как перейти к описанию пастеризационной установки, следует коротко разобрать ещё одно понятие, с которым мы будем сталкиваться в дальнейшем, а именно гомогенизация. Она представляет собой процесс дробления или уничтожения жировых шариков, образовавшихся в ходе хранения молока. Под воздействием внешних сил можно достичь значительного уменьшения объёма жировых шариков. Процесс гомогенизации позволяет предотвратить самопроизвольное отстаивание жира в молоке на производстве или при его хранении. При этом, гомогенизация даёт возможность сохранить однородную консистенцию молока. Далее рассматривать гомогенизацию так подробно, как процесс пастеризации, не имеет смысла, поскольку процесс гомогенизации не является центральным понятием предметной области.

1.2 Результаты обследования пастеризационной установки

Пастеризационная установка, она же пастеризационное-охладительная установка, предназначена непосредственно для самой пастеризации, т.е. для уничтожения вредных и опасных микроорганизмов в молоке. Она состоит из множества различных элементов, изучая которые мы также сможем изучить принцип работы пастеризационной установки.

Строго говоря, пастеризационная установка состоит из:

1. Балансного танка
2. Подающего насоса
3. Регулятора потока
4. Секции регенеративного предварительного подогрева
5. Центробежного очистителя
6. Секции нагрева
7. Трубы выдержки
8. Вспомогательного насоса
9. Системы нагрева горячей воды
10. Секции регенеративного охлаждения
11. Секции охлаждения
12. Возвратного клапана
13. Панели управления

И так, балансный танк является по сути резервуаром с молоком, оборудованным поплавковым входным клапаном, который регулирует расход молока и поддерживает его постоянный уровень в резервуаре. В балансном танке также имеется электрод минимального уровня. Он срабатывает, в том случае, когда уровень молока достигает минимальной точки, тем самым включая клапан распределения потока. Молоко заменяется водой и пастеризатор отключается.

Подающий насос позволяет обеспечить молоком сам пастеризатор, выкачивая его из балансного танка.

Для обеспечения устойчивого контроля температуры и постоянного времени выдержки пастеризационной установки используется регулятор потока. Он также необходим для поддержки расхода через пастеризатор на должно уровне.

Молоко, попав в секцию предварительного регенеративного подогрева, должно приобрести некоторую начальную температуру. Это осуществляется с помощью регенерированного тепла уже пастеризованного молока, которое в этот момент должно охлаждаться в секции регенеративного охлаждения.

Секцию регенеративного нагрева можно разделить на три секции: секция начального регенеративного подогрева, секция предварительного очищения и секция конечного регенеративного подогрева. Такое разделение используется, поскольку существует необходимость в предварительной обработке молока между входной и выходной температурой молока в секции регенеративного подогрева. Секция начального регенеративного подогрева обычно придаёт молоку температуру около 55-75 °С. После молоко отправляется в секцию предварительного очищения, представленную центробежным очистителем. После молоко поступает в секцию конечного регенеративного подогрева, где молоко добирает максимальную температуру.

Эффективность такого нагревания молока регенеративным способом достигает 90-96%, что отлично сказывается на энергосбережении. Остальную температуру входное молоко добирает за счёт горячей воды, температура которой на 2 или 3 °С больше температуры пастеризации. Все перечисленные процессы происходят в секции подогрева. Горячая вода имеется благодаря системе нагрева воды.

Нагретое молоко попадает во внешнюю трубу выдержки, где температура проверяется датчиком. Этот датчик непрерывно контактирует с регулятором температуры на панели управления, а также воздействует на регистрирующий прибор, который сохраняет температуру пастеризации. Тем самым происходит контроль процесса пастеризации.

Как только температура падает ниже минимума, активизируется возвратный клапан, позволяющий пастеризованному молоку попасть в секцию регенеративного охлаждения.

Секцию регенеративного охлаждения также можно разделить на две секции: первая секция регенеративного охлаждения и вторая секция регенеративного охлаждения. В первой секции регенеративного охлаждения горячее пастеризованное молоко передаёт свою температуру холодному, предварительно очищенному молоку, находящегося в секции регенеративного конечного подогрева молока. А во второй секции регенеративного охлаждения пастеризованное молоко отдаёт температуру необработанному молоку, находящемуся в секции регенеративного начального подогрева.

После охлаждённое молоко попадает в секцию охлаждения, которую тоже можно разделить на две секции: первая секция охлаждения и вторая секция охлаждения. В первой секции пастеризованное молоко охлаждается холодной водой, а во второй секции молоко уже охлаждается ледяной водой.

1.3 Введение в прогнозирование, нейронные сети и машинное обучение

А сейчас опишем средства, которые будут использоваться для составления прогнозов, а также опишем само понятие прогноза.

Одним из ключевых понятий данной работы является понятие временных рядов. Что же это? Временные ряды – это, по сути, некоторая последовательность, каждый элемент из которой состоит из двух или более параметров, один из которых обязательно должен обозначать время. Причём, все эти элементы в последовательности расположены в хронологическом порядке, т.е. в порядке возрастания параметра времени.

Параметр времени может быть представлен в разных форматах. Выбор формата времени зависит от задачи, удобства использования, длительности, в пределах которой будут собираться данные, а также от требуемой точности. Например, в случае если данные фиксируются раз в день, то хорошо подойдёт отсчёт времени по дням с указанием месяца и года. Если же данные фиксируются в определённые моменты дня, то к вышеописанному стоит прибавить указание часа и минуты фиксации. При необходимости можно указывать и секунды, и миллисекунды. Но что, если нам не особо-то и важно знать, в какой год, месяц или день это происходило, когда нам важно знать, сколько прошло времени от начала того или иного процесса? Тогда, нам скорее подойдёт формат дискретного времени. С помощью этого формата можно узнать длительность процесса в единственной выбранной нами единице измерения времени. Например, если мы сохраняем время в секундах, то 1000-ча секунд сохранит свой формат 1000-чи секунд, время не будет переведено в 16-ать минут и 40-ок секунд. Всё это нам позволяет не привязываться к датам, которые не особо-то и влияют на технологический процесс промышленного оборудования.

Остальные параметры могут уже характеризовать или описывать какой-либо процесс или процессы, причём даже не обязательно одного элемента, а даже целой системы элементов. Так, например, когда наш элемент последовательности состоит из двух признаков, а один из которых, как мы уже знаем, время, то второй, конечно же, уже будет обозначать характеристику или состояние изучаемого нами элемента. Но как только у нас появляется три или более признаков, тогда мы уже можем говорить о фиксации характеристик или состояний разных элементов изучаемой системы в один и тот же момент времени или же о фиксации характеристики или состояния одного из множества элементов системы, но с указанием этого элемента, например, с помощью идентификационного номера элемента. Как можно убедиться, временные ряды дают весьма гибкую возможность описания процессов или систем относительно времени.

Технологический процесс пастеризационной установки как раз и представляет собой временной ряд, в котором имеется информация о показаниях различных датчиков в определённые моменты времени. Поэтому, говоря о данных технологического процесса пастеризационной установки, мы будем понимать, что они имеют форму временных рядов.

А что из себя представляет работа с временными рядами? В основном работа делится на две части. Первая часть – это понимание структуры временного ряда, его закономерностей, таких как цикличность, тренд, сезонность и так далее, обработка данных временного ряда, визуализация данных, в общем, это всесторонний анализ временного ряда. Если опускать различные математические, статистические и тому подобные подробности, то анализ также может дать нам возможность понять, как начинался процесс, как шёл и

развивался и на чём он закончился или остановился на данном моменте времени. Строго говоря, нейронным сетям, как и исследователям временных рядов, тоже необходимо это понять, чтобы выполнить вторую часть работы, а именно, составление прогноза, что, зачастую, и является основной задачей работы с временным рядом. Да, анализ данных временного ряда технологического процесса делается с целью понять, что будет происходить с этим процессом дальше. Но чем нам так полезна информация о будущем, зачем она нужна? Для этого необходимо понять, что есть прогноз.

Сам прогноз – это некоторая случайная величина, характеризующая вероятность того, что график в будущем пройдёт через определённую точку или некоторую область. Тогда прогнозирование – это получение максимально точных прогнозов, или, если говорить в отрыве от понятия прогноза, то это точное предсказание будущего, учитывающее исторические данные об объекте прогнозирования, а также знания о любых будущих событиях, которые могут повлиять на прогнозы.

Как понять, что прогнозы действительные? Прогнозы являются таковыми, если они отражают подлинные закономерности и взаимосвязи, которые есть в исторических данных, при этом не повторяя прошлые события, которые более не актуальны или уже не повторяются.

Что необходимо, чтобы составить хороший прогноз на основе временных рядов? Для этого обычно требуется выполнить следующие шаги:

1. Определить задачу.
2. Собрать информацию.
3. Произвести предварительный анализ.
4. Выбрать и создать модель прогнозирования.
5. Использование и оценивание модели прогнозирования.

Вкратце разберём каждый пункт. При определении задачи необходимо понять, что вообще будет прогнозироваться, как будут использоваться прогнозы, благодаря чему будут получены прогнозы, кому эти прогнозы нужны или для чего. Второй пункт подразумевает непосредственный сбор или получение данных, а также оценка накопленного опыта людей, которые собирают данные и используют прогнозы. Для выполнения третьего пункта необходимо визуализировать данные, составить инфографику, если она необходима, определить взаимосвязь с признаками, закономерности временных рядов, качество данных и так далее, другими словами, провести анализ данных. На четвёртом пункте необходимо либо приобрести и адаптировать готовую модель, либо создать её самостоятельно. Но модель должна учитывать исторические данные, силы взаимосвязи прогнозируемым признаком и любым другим признаком, а также способы использования прогнозов. В заключении проводится тестирование, оценивание, развёртывание и сопровождение модели прогнозирования.

В дальнейшем на эти пять шагов прогнозирования мы будем ссылаться как на пять этапов составления прогноза.

Теперь поговорим о нейронных сетях и машинном обучении, а также о том, какое место занимает задача прогнозирования в рамках машинного обучения.

О нейронных сетях мы будем говорить, как о некоторой математической модели и её программной реализации, которая имитирует работу и взаимосвязи настоящих нейронов. Рассматривая нейронную сеть как «чёрный ящик», изображённую на рисунке 1.1, мы понимаем, что есть некоторое n входных данных и m преобразованных выходных данных. В зависимости от назначения сети входные данные могут обозначать исторические данные о процессе, необработанные картинки, зашумлённые чёрно-белые фотографии, текст

с ошибками и так далее, а выходными данными могут быть прогнозы, картинки с размеченными на них объектами, восстановленные цветные фотографии, исправленные тексты и так далее. Как можно понять, нейронные сети весьма универсальное средство, которое может решать большой спектр задач.

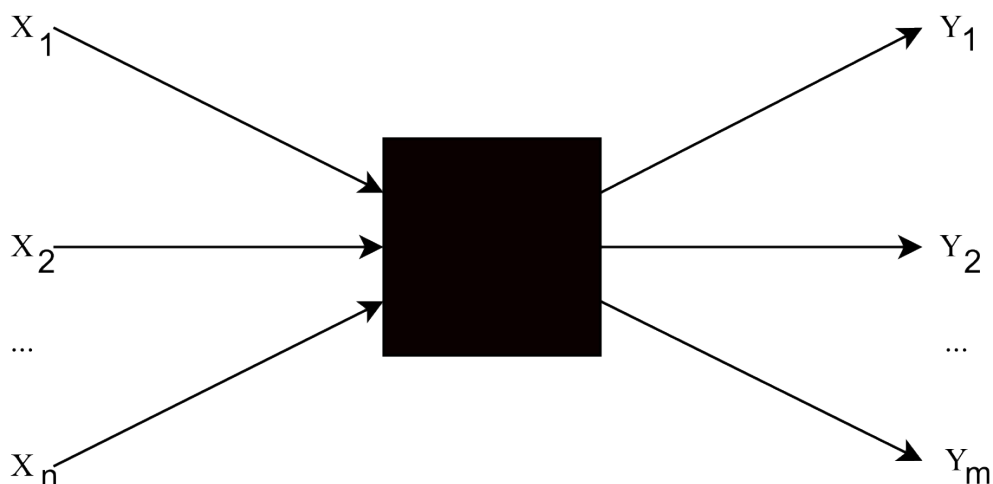


Рисунок 1.1 – Представление нейронной сети как ”чёрный ящик”

Нейронная сеть, если её рассматривать уже как «белый ящик», состоит из нейронов. Эти нейроны могут связываться между собой различными способами, например, один нейрон к одному нейрону, один ко многим, иметь обратные связи и так далее. По большей части, то, какие связи имеются между нейронами в сети, определяет, так называемую, архитектуру нейронной сети. Далее нейроны собираются в упорядоченные нейронные слои. Таких слоёв может быть несколько. Тот слой, первый из которых работает с входными данными, называется входным слоем. Слой, который производит расчёт выходных значений, называется выходным слоем. Остальные слои называются скрытыми. Каждый из этих слоёв имеет собственную размерность. На рисунке 1.2 нейронные слои изображены как вертикальные последовательности из нейронов, изображённые кружочками.

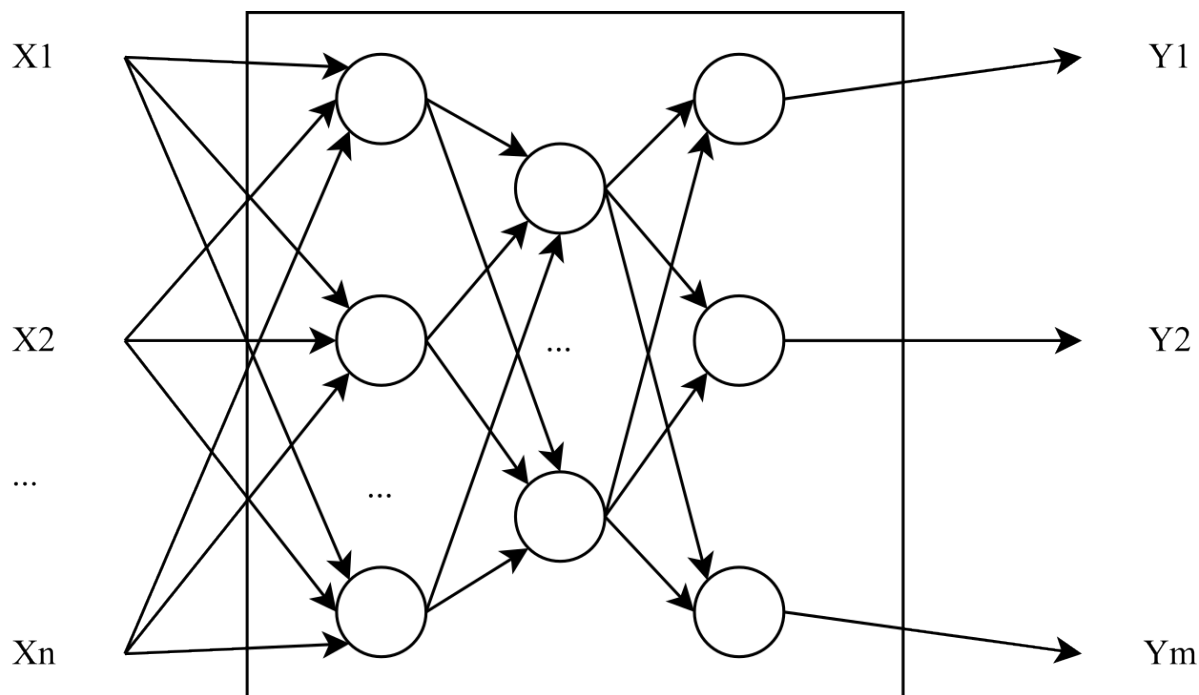


Рисунок 1.2 – Представление нейронной сети как "белый ящик"

Слои, как уже было сказано, состоят из нейронов. Каждый нейрон, в свою очередь, состоит из сумматора и активирующей функции. В первом элементе нейрона может происходить комбинирование операции умножения, сложение, умножения Адамара и так далее. Как правило, вначале производится умножение между входными данными нейрона и весами, каждый из которых соответствует своему входу. Веса могут обозначаться следующими символами: W , w , V , v , U , u и т.п. Веса могут быть представлены в виде отдельных значений, либо векторов, но чаще всего, в виде матриц. Результат каждого умножения в дальнейшем складываются между собой, а после выполняется операция сложения со смещением, которое обычно обозначается символами B или b , или операция вычитания порога, который обычно обозначается символами T или t . Как правило, смещения и пороги представлены в виде одного значения, либо некоторого вектора, размерность которого должна равняться размерности выхода нейрона. Результат всех этих операций в конечном итоге преобразуется с помощью активирующей функции, в качестве которой могут выступать такие функции как: линейная, тождественная, логистическая (сигмоидальная), биполярная логистическая, функции гиперболического тангенса, ReLU и так далее.

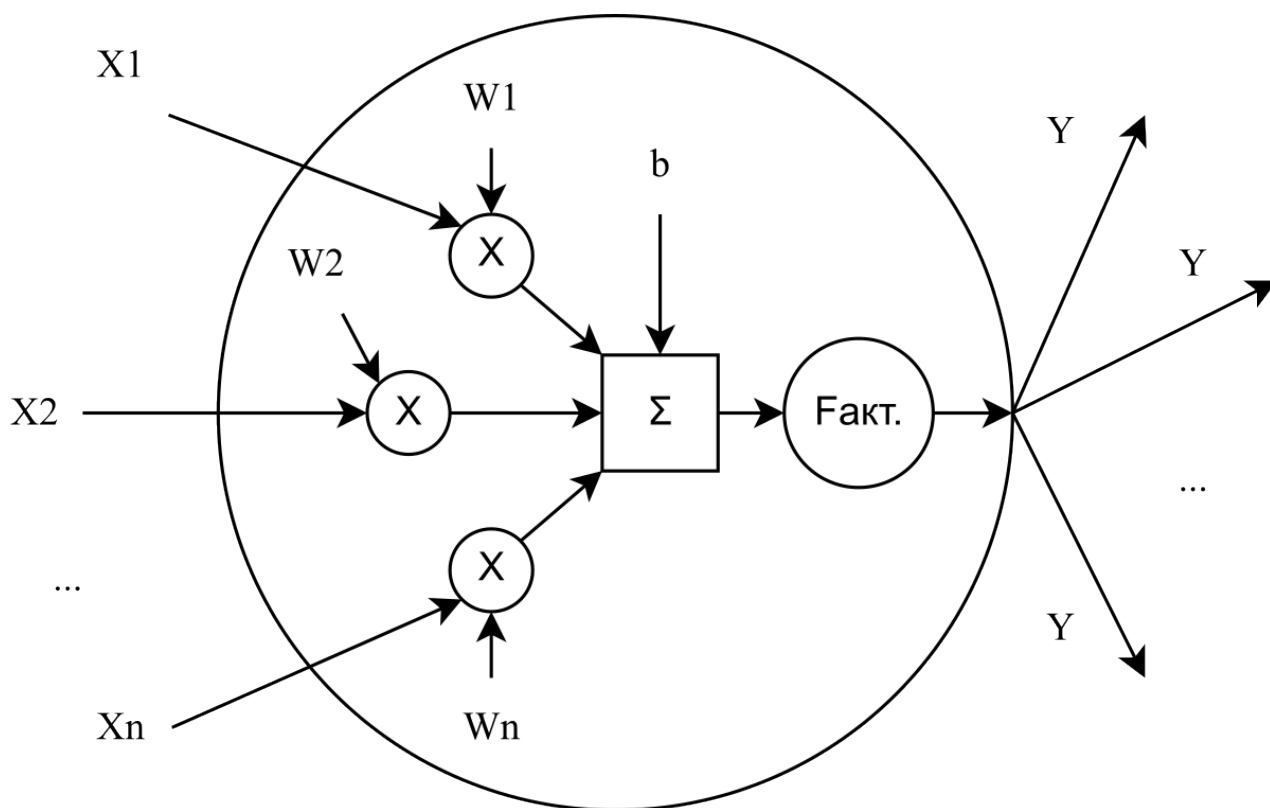


Рисунок 1.3 – Представление нейронной сети как ”белый ящик”

Пример расчёта нейрона:

$$y = F_{act} (w_1x_1 + w_2x_2 + \dots + w_nx_n + b) \quad (1.3)$$

И так, теперь перейдём к машинному обучению – набор алгоритмов, благодаря которым осуществляется обучение нейронной сети. А её обучение можно ещё определить, как подбор или поиск таких значений весов и смещений или порогов каждого нейрона, чтобы нейронная сеть выдавала необходимые результаты.

Теперь давайте определим, как связана задача прогнозирования и машинное обучение, чтобы обучить нейронную сеть предсказывать будущее. Чтобы это понять, нужно иметь ввиду, что машинное обучение делится на четыре направления: обучение с учителем, обучение без учителя, обучение с подкреплением и обучение с частичным привлечением учителя. Разберём только обучение с учителем и обучение без учителя, поскольку другие виды обучения не будут затронуты вообще.

Однако, для начала необходимо разъяснить несколько понятий, которые используются в лексике машинного обучения. Первое из таких понятий это понятие признака – некоторое измерение или параметр, определяющий характеристику или состояние чего-либо. Следующее понятие есть метка – по сути то, чему должен соответствовать признак. И в конце концов, пример – это то, что подаётся на вход нейронной сети. Существуют помеченные и непомеченные примеры. Помеченные примеры — это такие примеры, которые состоят из пар признаков и меток, т.е. $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, или такие примеры, в которых каждому вектору или каждой последовательности признаков соответствует вектор или последовательность меток $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$. Непомеченные примеры, это по факту просто признаки, т.е. $\{x_1, x_2, \dots, x_n\}$.

И так, говоря об отличии между обучением с учителем и обучением без учителя, можно сказать, что обучение с учителем использует помеченные примеры, а обучение без

учителя использует непомеченные примеры.

Рассмотрим задачу прогнозирования и определим, какими примерами мы оперируем, чтобы выбрать подходящую модель обучения модели нейронной сети. И так, для прогнозирования используются исторические данные, а данных из будущего у нас нет. В таком случае, как бы мы не рассчитывали будущее с помощью нейронных сетей на основе исторических данных без проверки с тем, что должно получиться, мы не достигнем никаких результатов. Нам необходимы некоторые эталоны, с которыми мы могли сравнивать полученные результаты сети. Откуда их брать? Чтобы их получить мы разделим исторические данные на примеры и соответствующие им эталоны, т.е. метки. Таким образом для обучения сети мы получим помеченные данные, в которых каждому вектору признаков, взятого из исторических данных, будет соответствовать вектор меток, взятых из тех же исторических данных, взятых сразу после последнего значения вектора признаков из общей последовательности исторических данных. Тем самым, мы говорим, что наша задача прогнозирования может быть решена с помощью нейронных сетей и машинного обучения.

Раз мы определились, что мы будем использовать машинное обучение для решения задачи прогнозирования, стоит пару слов сказать о инженерии машинного обучения, которая поможет достичь нам необходимых результатов. Рассмотрим этапы инженерии машинного обучения, которые определяют последовательность дальнейшей работы:

1. Определение задачи и цели.
2. Сбор и подготовка данных.
3. Конструирование признаков.
4. Обучение модели.
5. Оценка модели.
6. Развёртывание модели.
7. Обслуживание запросов к модели.
8. Мониторинг модели.
9. Сопровождение модели.

Первый, второй и третий этапы инженерии машинного обучения схожи с первыми тремя этапами составления прогноза. На этапе обучения модели производится непосредственно само обучение сети, по правилам, в соответствии с типом обучения. На пятом этапе проводится всесторонняя оценка модели, а также настройка некоторых её элементов. Именно на этом этапе делается вывод о готовности модели нейронной сети выполнять поставленные задачи. После наступает этап развёртывания нейронной сети, в ходе которого модель нейронной сети попадает в целевую среду для выполнения поставленных перед ней задач. На этапе обслуживания запросов необходимо убедиться в том, что на входы нейронной сети будут попадать данные такого формата, который будет понятен модели, и что сеть выдаёт необходимые результаты. Другими словами, это вопросы снабжения модели сети всем необходимым для её функционирования. Далее идёт этап мониторинга нейронной сети и её сопровождения. По факту это этапы использования сети по назначению.

Изучив этапы инженерии машинного обучения, становится понятно, что необходимо сделать, чтобы была создана модель нейронной сети, способная давать приемлемые результаты. Поэтому дальнейшая работа будет развиваться относительно этапов инженерии машинного обучения.

И так, порядок работы определён, поэтому в следующей главе будет разобрано, какие требования предъявляются к самой модели для прогнозирования, после будут определены цели и задачи прогнозирования, выполнив тем самым первые этапы инженерии машинного обучения и построения прогноза.

1.4 Требования к модулю прогнозирования и постановка задачи

Модуль прогнозирования должен выполнять прогноз технологического процесса пастеризационной установки. Для этого ему необходимо получать данные технологического процесса пастеризационной установки с помощью программируемого микроконтроллера, использующего платформу PLCnext Technology. По этой причине, программа должна быть написана на низкоуровневом языке программирования, а именно на C++, который поддерживается данным микроконтроллером.

Поскольку процесс пастеризации происходит непрерывно, то программа должна работать в режиме реального времени и своевременно снабжать необходимой информацией о прогнозах поведения пастеризационной установки.

Программа должна быть устойчива к различным сбоям и авариям, а также не создавать их сама.

Должна быть предоставлена возможность вносить некоторые настройки в модуль прогнозирования для корректировки работы модели нейронной сети.

Для постановки задачи, необходимо понимать, что сама программа, которая будет в дальнейшем использоваться для прогнозирования данных временных рядов технологического процесса пастеризационной установки, будет использовать нейросетевое решение задачи прогнозирования. Для обучения модели нейронной сети применяются алгоритмы машинного обучения, от которых зависит качество модели прогнозирования. Оттого данная работа в основном и будет развиваться по этапам инженерии машинного обучения, и потому данная работа будет являться проектом машинного обучения.

Задачей для проекта машинного обучения будет являться разработка модуля прогнозирования данных временных рядов пастеризационной установки с целью прогноза поведения пастеризационной установки, планирования производства, а также выявления аномального поведения, возможных сбоев и выбросов пастеризационной установки.

Для выполнения поставленной задачи необходимо изучать данные технологического процесса, а также провести их анализ и обработку. Далее необходимо выбрать подходящую архитектуру нейронной сети, построить и обучить модель сети, после чего организовать эффективное обучение сети, используя алгоритмы машинного обучения. Написать код нейронной сети необходимо на низкоуровневом языке программирования для дальнейшего развёртывания на программируемом контроллере, взаимодействующим с пастеризационной установкой.

Также необходимо произвести тестирование и оценку качества полученной модели. После чего непосредственно выполнить процесс развёртывания нейронной сети.

2. АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

В этой главе будут проанализированы методы, которые были использованы при решении схожей задачи в организации Kaspersky.

Перед организацией ставилась цель разработать систему, которая позволяла бы предотвращать отказы, аварии и незапланированные простои промышленного оборудования, выявляя признаки проблемы задолго до того, как проблема повлияет на работу предприятия. Другими словами, необходимо произвести прогнозирование данных технологического процесса промышленного оборудования для выявления признаков, по которым можно говорить о будущих аномалиях. Под аномалией для Kaspersky понимается значительное отклонение реальных данных технологического процесса от прогноза.

Причинами, по которым Kaspersky занялся данной задачей, стали устаревшие системы мониторинга, не способные эффективно искать причины возникновения аварий, а также огромный объём телеметрии, с которыми не могли справиться даже опытные операторы.

Для решения таких проблем была создана система Kaspersky MLAD. Это система для наблюдения за большим количеством пользователей телеметрии, а также для выявления отклонений, которые могут нанести повреждения производству.

Данная система способна предотвращать неполадки различного рода, представленные дефектом или сбоем оборудования, или ошибкой персонала, с целью дальнейшего предотвращения развития опасной ситуации. Также система способна определять отклонения в действиях сотрудников, позволяя раскрыть бойкот, диверсию, срыв производства и так далее. В добавок система Kaspersky MLAD призвана выявлять атаки злоумышленников. Место и роль данной системы на производстве можно увидеть на рисунке 2.1.

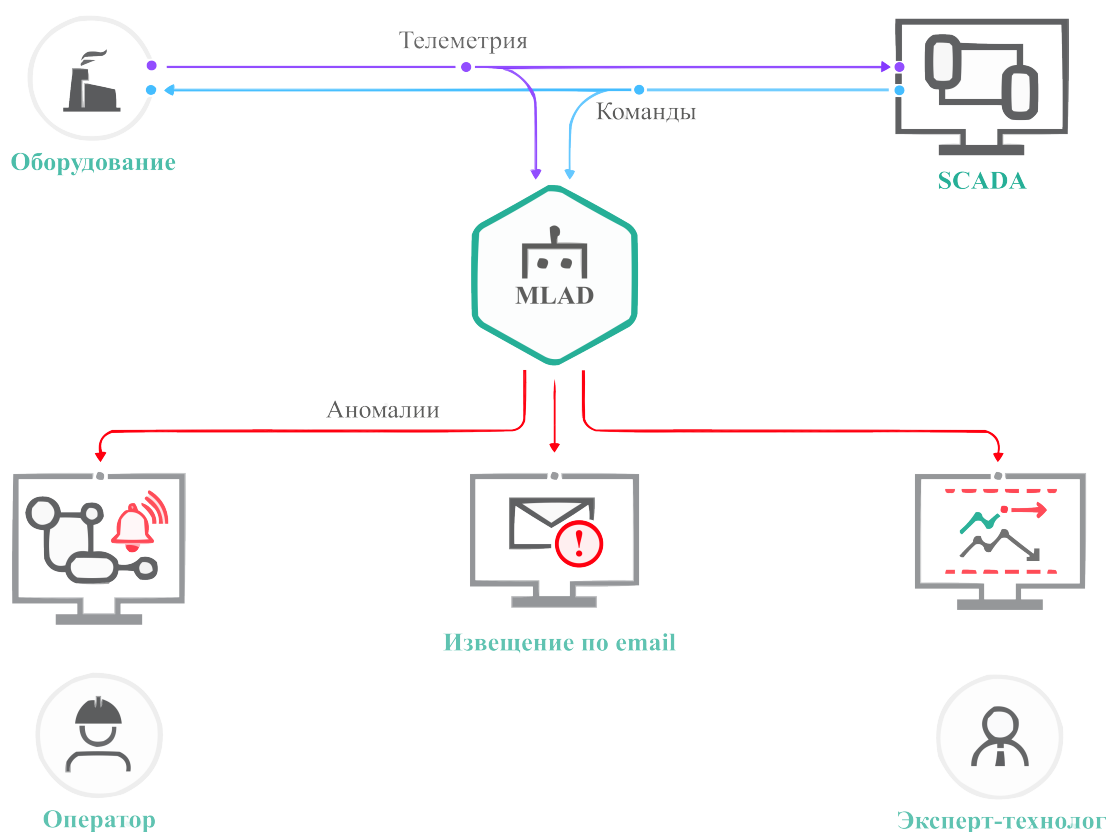


Рисунок 2.1 – Место системы Kaspersky MLAD на производстве

Изм.	Лист.	№ докум.	Подп.	Дата

ДП.AC59.200043-05 81 00

Лист

15

Для работы системы Kaspersky MLAD не требуется изменять как-либо технологической процесс промышленного оборудования. Эта система никак не воздействует на него. К тому же она не вмешивается в передачу данных или в средства управления промышленным оборудованием. Данная система оповещает оператора о возможном сбое, и оставляет задачу принятия решения на операторе. Данным свойством будет обладать и модели, описанные в данной работе.

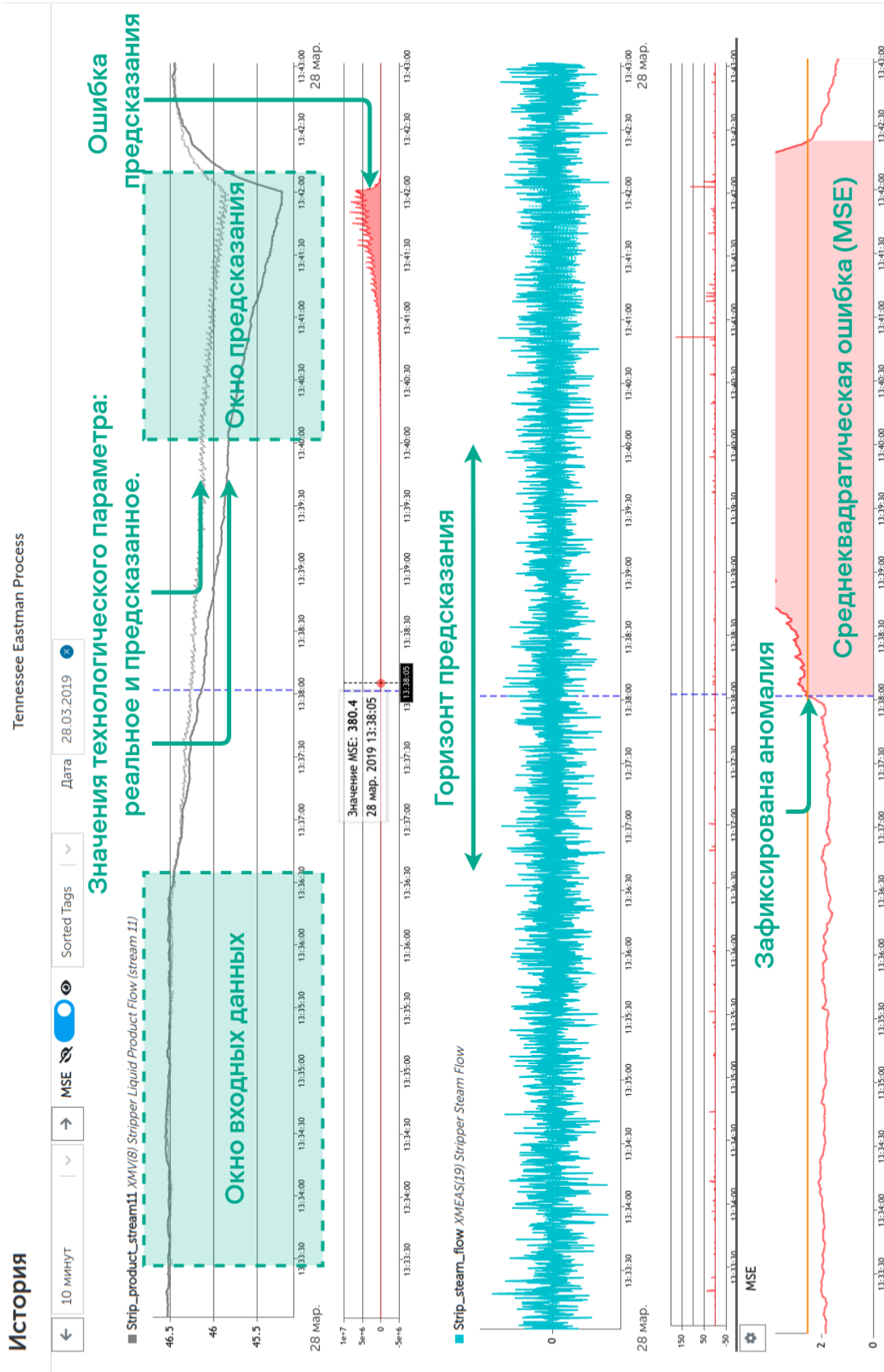
Если рассматривать Kaspersky MLAD с технической стороны, то это программное обеспечение, которое использует нейронные сети, способные анализировать существующий поток телеметрии. Поскольку параметры телеметрии связаны друг с другом и влияют друг на друга, то система Kaspersky MLAD должна это учитывать, и она с этим справляется. Она способна выявлять взаимосвязи между параметрами и использовать их для построения прогноза. Этот прогноз в дальнейшем обрабатывается различными нейросетевыми средствами, выявляя тем самым аномалии.

Система Kaspersky MLAD использует различные архитектуры нейронных сетей, такие как: CNN, DenseNet, TCN и RNN. Первые три сети используются для различного рода обработки данных, их анализа и выявления аномалий. А сети RNN используются для прогнозирования данных технологического процесса промышленного оборудования. Данная работа последует примеру Kaspersky MLAD и также будет использовать RNN сети для прогнозирования.

Таким образом, Kaspersky MLAD состоит из различных модулей нейронных сетей, взаимодействующих друг с другом. Рассматривая модель прогнозирования данной работы под призмой Kaspersky MLAD, можно понять, что задача данной работы состоит в том, чтобы создать один из этих модулей, выполняющий задачу прогнозирования. В дальнейшем планируется развитие данной работы и создание остальных модулей по примеру системы Kaspersky MLAD.

Но как именно система Kaspersky MLAD определяет аномалии. Для этого у неё есть пять видов детекторов аномалий. Но интересует нас лишь, так называемый, предиктивный детектор, который строится на основе нейронных сетей. Суть его работы проста: сначала выполняется прогноз данных, затем этот прогноз сравнивается с фактическим наблюдаемым поведением и с помощью машинного обучения определяются аномалии. Данный модуль можно разделить на две модели: модель прогнозирования и модель обработки прогнозов. Тем самым, данная работа, если рассматривать её под призмой системы Kaspersky MLAD, ставит своей целью создание первого модуля предиктивного детектора. Также нас интересует ещё один из видов детекторов аномалий, а именно потоковый процессор. Его задача состоит в том, чтобы приводить данные телеметрии к равноинтервальному временному виду, что подразумевает собой равным промежутки между точками относительно оси времени. Такой вид данных необходим для работы остальных видов детекторов аномалий, в том числе и для предиктивного детектора. Аномалии в потоковом процессоре вычисляются в процессе приведения данных к равноинтервальному временному виду.

Раз была определена схожесть данной работы с предиктивным детектором аномалий системы Kaspersky MLAD, тогда необходимо более подробно рассмотреть этот модуль и то, как он функционирует. Пример его работы можно увидеть на рисунке 2.2. Предиктивный детектор необходим для анализа данных технологического процесса, в основном представленных в виде временных рядов, содержащих разную информацию о состоянии объекта, различные физические величины, взятые с датчиков, и так далее. Предиктивный детектор работает по определённому алгоритму. Сначала формируется окно входных данных из поступивших на предиктивный детектор данных технологического процесса. На основе окна



входных данных выполняется прогноз, формируя окно прогноза или, как оно обозначено на рисунке 2.2, окно предсказания. Это окно показывает поведение объекта в определённом недалёком будущем. После вычисляется ошибка между прогнозом и реально наблюдаемыми данными для каждого предсказанного значения. Далее по совокупности этих ошибок производится вычисление среднеквадратичной ошибки (MSE). При вычислении этой ошибки значение индивидуальных ошибок прогноза умножаются на определённые веса. Таким образом осуществляется переход ко второй модели нейронной сети предиктивного детектора. Этот модуль определяется по значению среднеквадратичной ошибки аномалию с помощью некоторого порога, который формируется при обучении модели нейронной сети. Данный порог изображён на рисунке 2.2 в секции отображения среднеквадратической ошибки как оранжевая линия.

Предиктивный детектор позволяет снизить количество ошибочных оповещений об аномалиях технологического процесса, выявлять аномалии на ранних этапах развития проблемы с целью её предотвращения, а также обнаружить слабо проявляющиеся аномалии. Данная работа, хоть и не ставит своей целью именно поиск аномалий, однако известно, что её прогнозы будут в дальнейшем использоваться именно с этой целью. Этими словами мы однозначно закрываем первые этапы инженерии машинного обучения и построения прогноза.

3. АНАЛИЗ И ОБРАБОТКА ДАННЫХ ПАСТЕРИЗАЦИОННОЙ УСТАНОВКИ

3.1 Анализ и визуализация исходных данных технологического процесса

Данные, которые мы будем использовать для обучения, имеют форму временных рядов, состоящих из трёх параметров. Первый параметр – это идентификатор датчика или исполняющего устройства, который в дальнейшем мы будем называть просто сид. Вторым параметром – это дискретный момент времени, измеряемый в секундах. И, наконец, третий параметр – значение датчика.

Имеющиеся данные, описывающие технологический процесс пастеризационной установки, содержат информацию от шести различных датчиков, каждый из которых изображён на рисунке рисунке 3.1. Первый сид определяет значение циркуляционного насоса, измеряемое в процентах. Второй сид говорит нам о расходе молока, измеряемого в метрах кубических за час. Третий сид показывает температуру гомогенизации, изменяемую в градусах Цельсия. Четвёртый сид определяет степень открытия парового клапана в секции подогрева молока, измеряемое в процентах. Пятый сид показывает температуру пастеризации, изменяемую в градусах Цельсия. А шестой сид определяет степень открытия парового клапана в секции пастеризации. Шестой датчик измеряется в процентах.

Давайте взглянем на данные с точки зрения инженерии машинного обучения и определим некоторые характеристики данных.

Для начала определим, достаточно ли данных для обучения. Имеется 7 файлов формата .csv, которые хранят информацию о технологическом процессе пастеризационной установки в течении 70995465 секунд или около 835.5 дней. За такой промежуток времени мы имеем 20123304 данных различных сидов. Поэтому можно с уверенностью сказать, что данных для проекта машинного обучения более чем достаточно, а потому способны обеспечить достаточную обобщающую способность модели.

Данные описывают состояние пастеризационной установки круглые сутки в течении почти двух с половиной лет. Это нам позволяет говорить о большом покрытии данных. Это значит, что данные отражают все состояния объекта. Также мы можем говорить об информативности данных, поскольку она отражает реальный технологический процесс.

Для прояснения дальнейших моментов, хочется в очередной раз обратить внимание на то, что данные были собраны с помощью датчиков, поэтому данные являются надёжными и не подверженными к большинству видов смещений – несогласованность данных с явлением, которые эти данные описывают. Однако, эти данные всё-таки могут быть подвержены систематическому смещению, возникающее при измерениях или наблюдениях с помощью некоторого устройства, поскольку качество и характеристики устройства влияют на качество самих данных. Однако, мы можем игнорировать данное смещение, поскольку дальнейшая работа будет построена с теми же датчиками, а потому мы можем говорить и о ещё одной черте качественных данных, а именно о том, что данные отражают реальный вход. Это значит, что будущая сеть будет обучаться на данных, представленных в таком виде, в котором они будут в последствии приходить на входы сети. В будущем мы ещё упомянем это свойство и расскажем о некоторых подводных камнях, связанных с этим.

Также хочется отметить, что данные не являются результатом обратной связи, т.е.

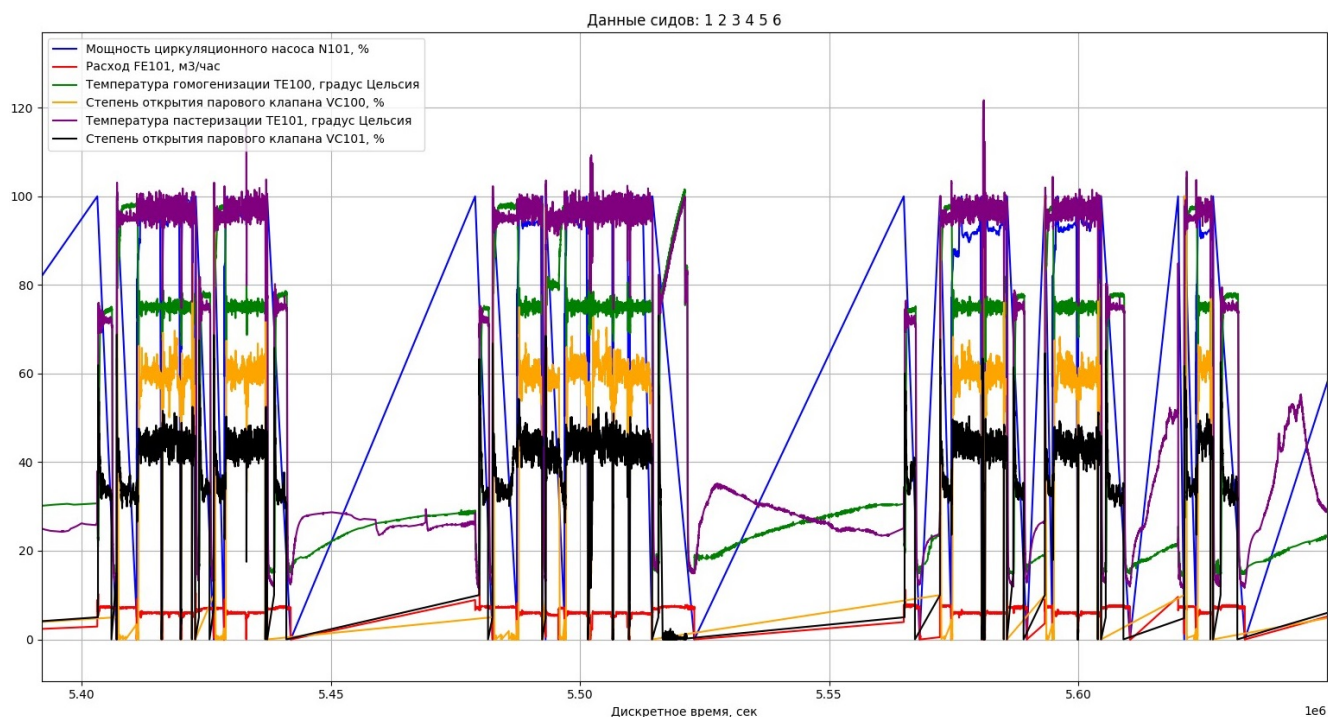


Рисунок 3.2 – Визуализация данных 1-ого файла.

не являются результатом самой модели, поскольку при сборе данных модели ещё не существовало.

Для дальнейшего описания данных необходимо их визуализировать.

Сравнивая данные на рисунках 3.2, 3.3 и 3.4, хотелось бы пояснить, что данные остальных файлов очень похожи с данными на представленных рисунках. Из этого делаем вывод, что данные могут содержать аномалии, поскольку схожее поведение пастеризационной установки средствами визуализации в других файлах выявлено не было. Это значит, что мы имеем большинство данных без аномалий, но также имеем и примеры данных с аномалиями.

Продолжим изучение данных.

ИНФОГРАФИКА

Теперь пришло время определить связи между параметрами пастеризационной установки. Беря во внимание всю собранную о данных и самой пастеризационной установке информацию, а также общение с персоналом, ответственные за работу установки, удалось более подробно сформулировать задачу данной работы. Необходимо прогнозировать технологический процесс пастеризационной установки, но это подразумевает прогнозирование трёх величин: расход, температура гомогенизации и температура пастеризации. В случае же с параметрами, описывающими степень открытия паровых клапанов и мощность циркуляционного насоса, то их прогнозировать нет необходимости, поскольку они изменяются в зависимости от сигналов регуляторов.

Таким образом, параметр мощности напрямую влияет на параметр расхода. Чем больше мощность насоса, тем больше расход молока. Однако, может быть такой момент, когда насос работает на полную мощность, а расход падает или равен нулю. Ответом на эту ситуацию служит тот факт, что в танке, из которого насос подкачивает молоко, собственно говоря, молока нет. Однако данной информацией о том, если ли молоко в танке, исходные данные не располагают. В таком случае, если мощность насоса не будет влиять на расход молока, то результаты прогноза и реальные данные будут значительно расходиться и это

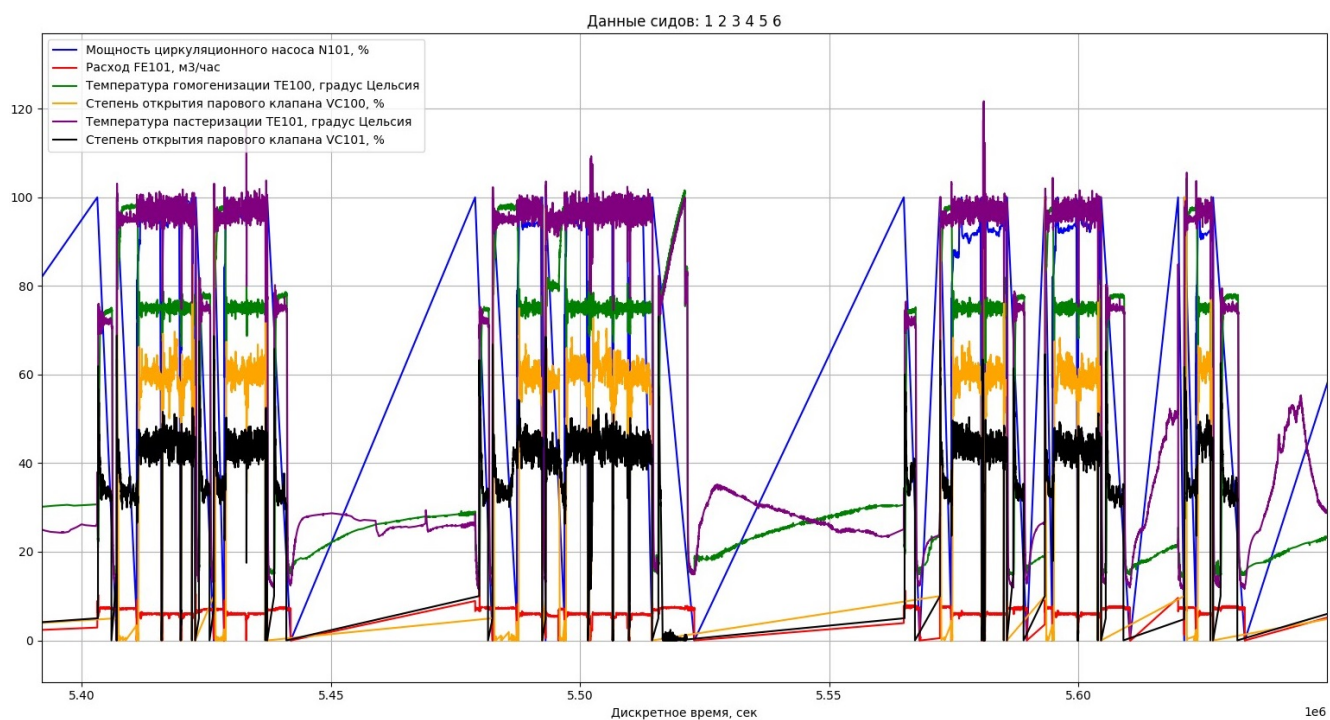


Рисунок 3.3 – Визуализация данных 2-ого файла с растяжением масштаба по оси абсцисс.

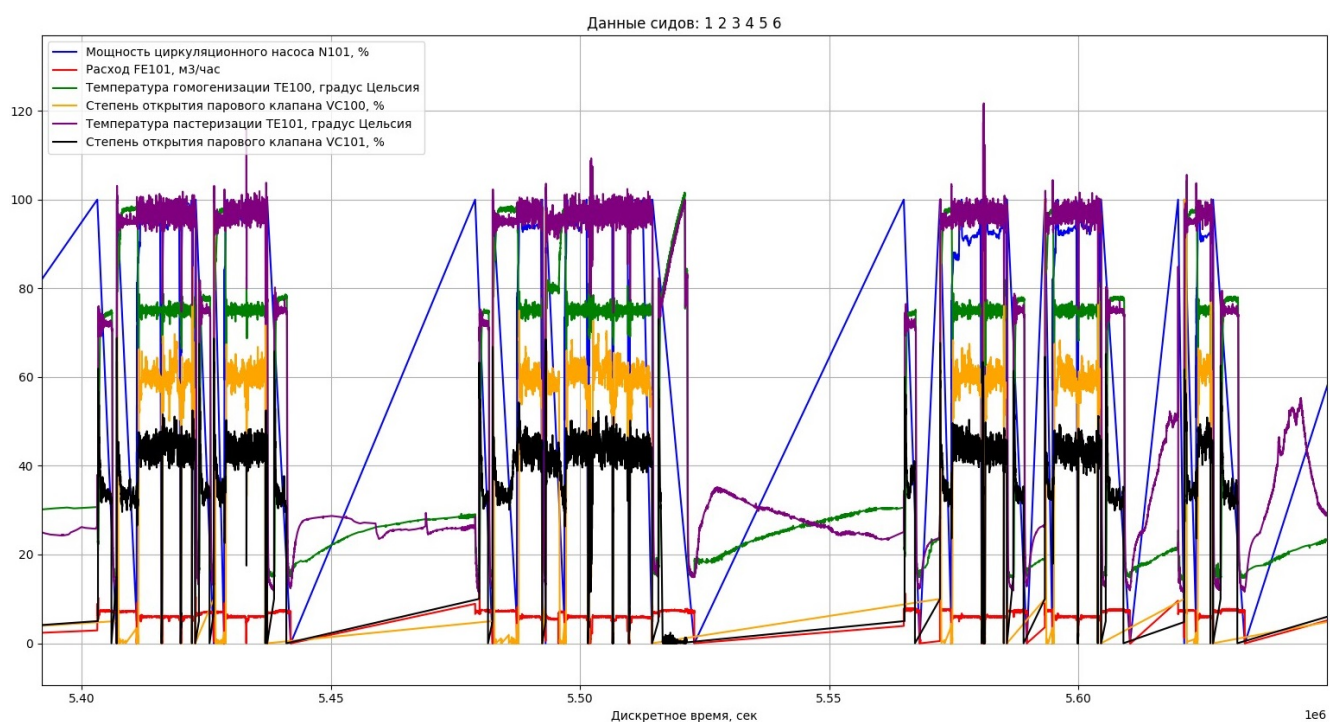


Рисунок 3.4 – Визуализация данных 3-ого файла с суженным масштаба по оси абсцисс.

Изм.	Лист.	№ докум.	Подп.	Дата

ДП.АС59.200043-05 81 00

Лист

22

должно будет быть определено как аномалия.

Параметр температуры гомогенизации напрямую зависит от степени открытия парового клапана. Пар нагревает молоко, теряя свою температуру, поэтому пар должен непрерывно циркулировать в системе при нагревании молока. В этом нам помогает паровой клапан, который способен препятствовать потоку пара, либо наоборот, дать ему возможность циркулировать в системе. Такая же история и с температурой пастеризации. Тоже существует секция с паром, влияющая на температуру молока при пастеризации.

Также можно создать дополнительные связи: параметр температуры гомогенизации зависит от параметра расхода и параметр температуры пастеризации зависит от температуры гомогенизации. Но параметр расход и параметр температуры гомогенизации также влияет и на степени открытия паровых клапанов, поэтому данные связи будут опущены, поскольку они учтены в других сидах, имеющих более тесные связи с прогнозируемыми величинами.

3.2 Обработка данных и их подготовка к прогнозированию

И так, пришло время поговорить об обработке данных. Она нужна в основном для того, чтобы сделать данные понятными для модели, а также для заполнения «дыр» в данных или для различного рода аппроксимаций. Под «дырами» понимаются некоторые пропущенные значения, например, определено время и значение датчика, а сид пропущен, вместо него пустое поле. Также под «дырами» понимаются магические числа, вроде -1, 99999 и другие, говорящие о том, что значение, заменённое магическим числом, неизвестно, но пустого поля быть не может. В данных формата oscdf такой проблемы не наблюдается, однако не стоит об этом забывать. В будущем мы ещё с этим столкнёмся.

Поговорим о хранении данных. Начальные данные хранятся в файлах формата .csv. Это весьма распространённый формат файлов для хранения данных, представленных в форме временных рядов. В них данные хранятся в следующем виде:

<сид>;<время>;<значение>.

В дальнейшем такой вид данных мы будем обозначать как oscdf-формат данных. Преимуществом хранения данных в таких файлах является их доступность для любых программ или языков. Однако такие файлы занимают большой объём, а также при работе с такими файлами чтение или запись информации занимает большое количество времени.

Для решения основных проблем при работе с файлами формата .csv, было принято решение использовать бинарные файлы формата .bin для хранения данных. Это дало возможность уменьшить занимаемый объём файлов, но самое главное, это позволило значительно сократить время для записи или чтения информации. Для примера: тут будут примеры. Разница чтения данных составляет , а разница записи данных составляет .

В ходе работы с моделью возможно не будет необходимости работать со всеми сидами, а только с определёнными. Для того, чтобы не задействовать оперативную память зря, необходимо разделить данные по сидам.

Для формирования различных наборов данных необходима возможность обрезать данные до необходимого нам диапазона или количества. Для более полного понимания смысла обрезки данных в очередной раз обратимся к инженерии машинного обучения: весь набор данных делится на обучающий, контрольный и тестовый наборы. Обучающий набор используется для непосредственного обучения модели. На контрольном наборе проверяет-

ся качество модели и подбираются её параметры, другими словами, настраивается модель. На тестовом наборе проверяется работа сети и делается вердикт о готовности модели.

Для формирования различных наборов данных как раз и нужны различные способы обрезки данных, позволяющие отбросить данные как справа, так и слева.

Также необходима возможность добавлять новые данные, чтобы при обучении сеть, которая учитывает зависимости между различными датчиками, была обеспечена данными необходимых сидов.

Учитывая особенности вида данных, которые поуступают на предиктивный детектор системы Kaspersky MLAD, а также ввиду того, что данная модель будет работать в режиме реального времени и данные будут поступать через некоторый промежуток времени, то необходимо сделать так, чтобы начальные данные также были равноудалены друг от друга относительно оси времени. Однако, взаимное удаление точек по оси времени в начальных данные не постоянно. Эти диапазоны необходимо выровнять, чтобы повысить схожесть реальных входов и входов при обучении модели. Другими словами, необходимо преобразовать начальный данные в такие данные, в которых каждое значение времени представляло собой арифметическую прогрессию:

$$t_i = t_0 + i \times r \quad (1)$$

где:

t_0 – начальное значение времени;

r – необходимое расстояние между точками, представляющее собой некоторую константу, которую может задать пользователь;

i – номер точки, итерации, элемента прогрессии; t_i – i -ый элемент прогрессии.

Теперь осталось определить значение ординаты для каждого рассчитанного значения времени. Для этого было принято решение воспользоваться аналитической геометрией и взять за основу уравнение прямой, проходящей через две точки.

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} \quad (3.2)$$

где точка $(,)$ – это точка, которую необходимо найти между точками $(1, 1)$ и $(2, 2)$, взятых из реальных данных. При этом, известно, что $1 \leq x \leq x_2$.

Для определение ординаты для i -того элемента прогрессии необходимо взять две ближайших точки из реальных данных относительно оси абсцисс к искомой точке. Тогда получим выражение для определение ординаты:

$$y = \left(\frac{(x - x_1) \times (y_2 - y_1)}{x_2 - x_1} \right) + y_1 \quad (3.3)$$

Таким образом можно получить аппроксимированные данные, где взаимное удаление между двумя соседними точками будет везде постоянным.

Благодаря возможности выравнивания диапазонов по оси абсцисс, появляется возможность создать, так называемые, аккуратные данные – данные представленные в виде упорядоченной таблицы без каких-либо «дыр». Однако, если попытаться перевести имеющиеся данные в табличный формат, который в последствии мы будем называть tdf форматом, то мы обнаружим, что начальные данные содержат информацию о значениях датчика при неповторяющихся уникальных значениях времени. Это значит, что если для некоторого датчика определено его значение в некоторый момент времени в начальных данных, то значение другие датчиков в этот же момент времени не известно. Для решения данной проблемы нам помогут операции парсинга и выравнивания диапазонов по оси абсцисс.

Чтобы преобразовать данные формата ocdf в данные формата tdf необходимо выполнить несколько последовательных действий.

Для начала необходимо убедиться, что данные формата ocdf содержат в себе данные всех сидов, а после распарсить данные по сидам. Затем обрезать все последовательности распарсенных данных с их начала и до максимального значения времени первых элементов последовательности распарсенных данных, также с минимального значения времени последних элементов последователи распарсенных данных до конца всех последовательностей распарсенных данных. Если в обрезаемой последовательности нет точки с граничным значением времени, тогда необходимо найти эту точку средствами уравнения прямой через две точки, описанной выше. Далее выровнять диапазоны по оси абсцисс для всех распарсенных данных. В заключении требуется соединить полученные данные в одну последовательность следующего вида. Формат tdf выглядит следующим образом:

<время>;<ЗД1>;<ЗД2>;<ЗД3>;<ЗД4>;<ЗД5>;<ЗД6>

где, ЗДН – значение датчика под номером Н.

Выполнив алгоритм, мы получаем аккуратные данные в табличном виде. Хранить их предпочтительно в бинарном виде, поскольку в формате .csv такие данные будут занимать очень много места, да и время на их чтение или запись будет довольно-таки немаленьким.

Учитывая некоторые особенности активирующих функций, некоторые модели будут содержать в себе встроенные модули преобразования данных из одного диапазона по оси ординат в эквивалентные данные из другого диапазона по оси ординат и наоборот, сохраняя значения по оси абсцисс. Такую операцию мы будем в дальнейшем называть скейлингом.

Данная задача выглядит следующим образом: пусть есть $y \in [a; b]$, а необходимо преобразовать y в эквивалентное значение $z \in [c; d]$. Тогда для операции скейлинга имеем следующее выражение:

$$z = \frac{(y - a) \times (b - a)}{d - c} + c \quad (3.4)$$

А для того, чтобы восстановить эти значения, т.е. провести операцию обратного скейлинга, необходимо использовать следующее выражение:

$$y = \frac{(z - c) \times (d - c)}{b - a} + a \quad (3.5)$$

Операции скейлинга предполагают те модели, выходные элементы которых используют активирующие функции, имеющие горизонтальные асимптоты. Другими словами, активирующие функции, ограниченные по оси ординат. В таком случае, перед обучением или прогнозированием данных, будет проведена операция скейлинга, после основная операция, а в завершении операция обратного скейлинга. Т.к. данные уже изучены, можно с полной уверенностью сказать, что операцию скейлинга для данных по оси ординат мы можем провести.

4. РАЗРАБОТКА И ОБУЧЕНИЕ МОДЕЛИ НЕЙРОННОЙ СЕТИ

4.1 Выбор архитектуры нейронной сети для прогнозирования

Для выбора архитектуры нейронной сети необходимо для начала понять, какие архитектуры используются для прогнозирования данных временных рядов. Зачастую для выполнения данной задачи используются однослойные персептроны, многослойные персептроны и рекуррентные нейронные сети. Обращая внимание на свойства данных, становится понятно, что однослойный персептрон не сможет обеспечить необходимую обобщающую способность. Тогда остаётся выбор между многослойным персептроном и рекуррентной нейронной сетью.

Относительно многослойных персептронов существует теорема, согласно которой Персептрон с одним скрытым слоем является универсальным аппроксиматором, т. е. он способен с любой степенью точности аппроксимировать любую непрерывную функцию, если в качестве функции активации нейронных элементов скрытого слоя используется непрерывная, монотонно возрастающая, ограниченная функция. При этом точность аппроксимации функции зависит от количества нейронов в скрытом слое. Чем больше количество нейронов, тем больше точность аппроксимации. Однако при слишком большой размерности скрытого слоя может наступить явление, которое называется перетренировкой сети, когда сеть имеет плохую обобщающую способность.

Что касается RNN, также существует теорема, говорящая нам о том, что любая нелинейная динамическая система при использовании достаточного количества сигмоидальных нейронных элементов в скрытом слое с любой точностью может быть аппроксимирована рекуррентной нейронной сетью.

Выходит, что для прогнозирования можно использовать как многослойный персептрон, так и RNN. Однако, RNN обладает очень интересной чертой: она обладает, так называемой, памятью, способной «запоминать» особенности данных временных рядов. Такой памятью многослойный персептрон не обладает. Поскольку данные имеют в себе отличающиеся друг от друга, но повторяющиеся в будущем участки, то наличие такой памяти было бы очень кстати. Также, как можно будет узнать дальше, большинство, если не все, RNN имеют чётко определённую структуру, в которую довольно сложно что-то добавить, да и нет в этом необходимости. В случае с многослойными персептронами, их структура не постоянна: необходимо подбирать количество скрытых слоёв. Поэтому в некотором смысле, реализация RNN не только будет выгоднее, но и проще. При этом, если обратиться к Kaspersky MLAD, то можно вспомнить, что для прогнозирования данных телеметрии используются именно RNN. Учитывая всё вышесказанное, выбор архитектуры падает на RNN.

Теперь необходимо выбрать модель RNN. Существует большое количество различных рекуррентных нейронных сетей: SRN Джордана, SRN Элмана, мультирекуррентная нейронная сеть, LSTM, GRU и так далее. Однако, одной из самых распространённых и эффективных RNN считают LSTM сеть. Эта сеть встречается куда чаще, чем её модификации, такие как GRU, MGU, LSTM «с глазками» и другие. Основной задачей ставилось упрощение сети без вреда её точности, результатом чего появилась ранее упомянутая GRU, точность предсказаний которой не уступает LSTM, но при этом сеть менее громозд-

кая. Упростить удалось и GRU, получив тем самым MGU, однако точность этой сети всё также сравнивалась с LSTM. Поэтому для данной работы за основу будет взята классическая LSTM.

К слову, LSTM была выбрана и Kaspersky при разработке системы MLAD. Это в очередной раз говорит о популярности и надёжности сети LSTM, что укрепляет уверенность в выборе.

4.2 Обзор LSTM как средство прогнозирования

Сеть LSTM обладает двумя видами памяти, которые способны сохранять различные зависимости в данных. Есть долгосрочная память, которая обозначается как C_t , и краткосрочная память, которая обозначается как H_t . Как можно догадаться, долгосрочная память сети хранит в себе особенности данных на протяжении всей истории, а краткосрочная память хранит зависимости ближайших промежутков и сильнее влияет на результат прогноза. Зачастую краткосрочную память сети обозначают как её выход. При этом входом сети является некоторый вектор данных, на основании которых будет делаться прогноз. Вектор входных данных будет обозначаться как X_t . Как можно заметить, у всех параметров сети присутствует временная компонента t . Это говорит о том, что расчёт прогноза происходит относительно времени. Каждая новая итерация сети происходит с изменением времени. Память сети изменяется на одной из итерации и передаётся на следующую, тем самым сеть и сохраняется особенности данных временного ряда, который был подан на вход сети. Такая передача памяти возможно благодаря, так называемых в нейронных сетях, обратных связях, наличие которые и определяет сеть как рекуррентную. Вышеописанное можно увидеть на рисунке 4.1.

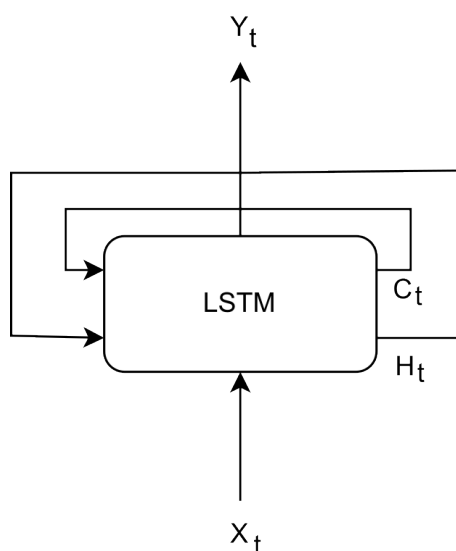


Рисунок 4.1 – Блок LSTM сеть с изображением обратных связей.

Каждый такой блок сети LSTM, изображённый на рисунке выше, зачастую называют ячейкой LSTM. Связь этих ячеек между собой можно гораздо лучше увидеть на

развёрнутой схеме LSTM, изображённой на рисунке 4.2.

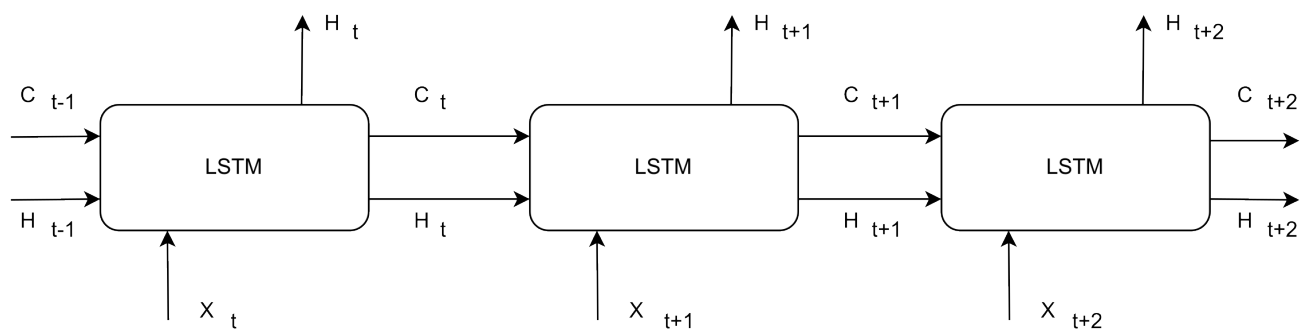


Рисунок 4.2 – Взаимосвязь ячеек LSTM в развёрнутом виде.

СПИСОК СОКРАЩЕНИЙ

НС – нейронная сеть

ТП – технологический процесс

ПУ – пастеризационная установка

ВР – временные ряды

Kaspersky MLAD – (Kaspersky Machine Learning for Anomaly Detection) методы машинного обучения компании Kaspersky для выявления аномалий

SCADA – (Supervisory for Control And Data Acquisition) система, обеспечивающая диспетчерское управление и сбор данных, относящаяся к классу программного обеспечения для создания автоматизированных систем управления технологическими процессами

CNN – (Convolutional Neural Network) свёрточная нейронная сеть

DenseNet – (Densely Connected Networks) тип сверточной нейронной сети, которая использует плотные связи между слоями через плотные блоки, где мы соединяем все слои напрямую друг с другом

RNN – (Recurrent Neural Network) рекуррентная нейронная сеть

TCN – (Temporal Convolutional Networks) темпоральная CNN

MSE – (Mean Squared Error) среднеквадратичная ошибка

SRN – (Simple Recurrent Network) простая рекуррентная сеть

LSTM – (Long Short-Term Memory) сеть долгой краткосрочной памяти

GRU – (Gated Recurrent Units) управляемые рекуррентные нейроны

MGU – (Minimal Recurrent Units) минимальные управляемые нейроны