**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН**

**Некоммерческое акционерное общество**
**«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА»**

**Кафедра** Телекоммуникационных сетей и систем

«Допущен к защите»
**Зав.кафедрой** Темырканова Э.К., доктор PhD, доц.
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

# ДИПЛОМНЫЙ ПРОЕКТ

**На тему:** Методы глубокого обучения в задачах телекоммуникационного оператора
**Специальность** 5В071900 – Радиотехника, электроника и телекоммуникации
**Выполнил** Тұрғамбаев Т.Т **Группа** РЭТ-16-3
**Научный руководитель** Жунусов К.Х д.т.н., профессор
(Ф.И.О., ученая степень, звание)

**Консультанты**
**по технической части:**
Панченко С.В., доц.
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

**по экономической части:**
Ибришев Н.Н., д.э.н., профессор
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

**по безопасности жизнедеятельности:**
Бекбасаров Ш.Ш., д.т.н
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

**по применению вычислительной техники:**
Панченко С.В., доц.
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

**Нормоконтролер:** Гармашова Ю.М., доц.
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

**Рецензент:** _____
(Ф.И.О., ученая степень, звание)
_____«____» _____2020 г.
(подпись)

**Алматы 2020 г.**

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН**

**Некоммерческое акционерное общество**
**«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА»**
**Институт** Космической инженерии и телекоммуникаций (ИКИТК)
**Кафедра** Телекоммуникационных сетей и систем
**Специальность** 5В071900 – Радиотехника, электроника и телекоммуникаций

**ЗАДАНИЕ**
на выполнение дипломного проекта

Студенту Тұрғамбаев Т.Т.
Тема проекта Методы глубокого обучения в задачах телекоммуникационного оператора

Утверждена приказом ректора № 147 от «11» _____ноября___ 2019 г.

Срок сдачи законченного проекта «25» _____мая____ 2020 г.

Исходные данные к проекту требуемые параметры результатов проектирования (исследования) и исходные данные объекта: Технологии глубокого обучения, пакет прикладных программ «Python». Технические данные города Милан. Функциональные дополнения Python. Функциональные дополнения ARIMA.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта:
Методы глубокого обучения
Прогнозирование с помощью алгоритма ARIMA
Использование методов глубокого обучения в разных отраслях, так же в телекоммуникационной сети

Перечень графического материала (с точным указанием обязательных чертежей):

Цели и задачи дипломного проекта
Актуальность
Методы глубокого обучения
Прогнозирование с помощью алгоритма ARIMA на Python
Листинг программы «Python»
Результаты моделирования
Анализ результатов прогнозирования
Выводы
Экономическая эффективность проекта
Безопасность проекта для жизнедеятельности

Основная рекомендуемая литература:

1 A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A survey on enabling technologies, protocols, and applications,” IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.

2 F. Chollet, Deep learning with Python. Manning Publications, 2018

3 N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, “A deep learning approach to network intrusion detection,” Forest (RF), vol. 2, 2017.

4 Боканова Г.Ш., Бекишева А.И. Методические указания к выполнению расчетно-графических работ для студентов специальности 5В071900 – «Радиотехника, электроника и телекоммуникации». – Алматы: АУЭС, 2015.

5 Баклашова Н.И. Охрана труда на предприятиях связи и охрана окружающей среды: Учебник. – М: Радио.и связь, 1999. – 288 с.

Консультанты по проекту с указанием относящихся к ним разделов проекта

| Раздел | Консультант | Сроки | Подпись |
|---|---|---|---|
| Теоретическая часть | Панченко С.В. | 25.05.2020 г. | |
| Экономическая часть | Ибришев Н.Н | 25.05.2020 г. | |
| Безопасность жизнедеятельности | Бекбасаров Ш. Ш | 25.05.2020 г. | |
| Применение вычислительной техники | Панченко С.В. | 25.05.2020 г. | |
| Нормоконтроль | Гармашова Ю.М. | 17.06-18.06.2020 г. | |

График
подготовки дипломного проекта

| № п/п | Наименование разделов, перечень разрабатываемых вопросов | Сроки представления научному руководителю | Примечание |
|---|---|---|---|
| 1 | Обзор характеристик и требований к методам глубокого обучения | 13.01-01.02. 2020 г. | Выполнено |
| 2 | Особенности данных телекоммуникационных операторах | 03-30.02. 2020 г. | Выполнено |
| 3 | Описание Python | 01-12.03. 2020 г. | Выполнено |
| 4 | Состояние и направления развития глубокого обучения | 14-20.03. 2020 г. | Выполнено |
| 5 | Функциональные дополнения к прогнозированию с помощью Python | 22.03-12.04. 2020 г. | Выполнено |
| 6 | Моделирование данных | 12-25.04. 2020 г. | Выполнено |
| 7 | Технические требования, предъявляемые методам глубокого обучения | 25-30.04. 2020 г. | Выполнено |
| 8 | Мировая практика применения Глубокого обучения | 16-18.05. 2020 г. | Выполнено |
| 9 | Стратегия внедрения и применения глубокого обучения | 19-22.05. 2020 г. | Выполнено |
| 10 | Экономическая часть | 22-23.05. 2020 г. | Выполнено |
| 11 | Безопасность жизнедеятельности | 24-25.05. 2020 г. | Выполнено |

Дата выдачи задания «__17__»_____февраля_____2020 г.

Заведующий кафедрой _____ (Темырканова Э.К.)
(подпись) (Ф.И.О.)

Научный руководитель проекта _____ (Жунусов К.Х.)
(подпись) (Ф.И.О.)

Задание принял к исполнению студент _____ (Тұрғамбаев Т.Т.)
(подпись) (Ф.И.О.)

**Аңдатпа**

Бұл дипломдық жобада телекоммуникациялық желіде терең оқыту технологиясына байланысты мәселелер қарастырылды және ARIMA моделі көмегімен болжам жасалды. Python бағдарламалау тілінде код әзірлеу бойынша жұмыс жүргізілді.

Өміртіршілік қауіпсіздігі бөлімінде микроклимат мәселелері қарастырылды және бизнес жоспар әзірленді.

**Аннотация**

В данном дипломном проекте были рассмотрены и изучены вопросы, связанные с технологией глубокого обучения в телекоммуникационной сети и разработана модель прогнозирования трафика ARIMA. Проделана работа по разработке кода на языке программирования Python.

Рассмотрены проблемы с микроклиматом из раздела безопасности жизнедеятельности и разработан бизнес-план.

**Annotation**

In this diploma project, issues related to the deep learning technology in the telecommunications network were considered and studied, and forecasting was carried out using ARIMA.Work has been done to develop code in the Python programming language.

Problems with the microclimate from the section of life safety are considered and a business plan is developed.

## Содержание

Введение ..... 7
1. Методы машинного обучения в задачах телекоммуникационного оператора ..... 9
1.1 Машинное обучения ..... 9
1.2 Характеристика Big Data ..... 13
2 Архитектура глубокого обучения ..... 15
2.1 Введение в глубокое обучение ..... 15
2.2 Модели и архитектуры глубокого обучения ..... 16
2.3 Сверточные нейронные сети ..... 17
2.4 Рекуррентные нейронные сети ..... 20
2.5 Генеративные состязательные сети ..... 23
2.6 Приложения глубокого обучения ..... 31
2.7 Будущие направления деятельности глубокого обучения ..... 34
3 Техническая часть ..... 36
3.1 Язык программирования - Python ..... 36
3.2 Библиотеки Python ..... 37
3.3 Прогнозирование временных рядов ..... 38
3.4 Компоненты временных рядов ..... 41
3.5 Автокорреляция временного ряда ..... 41
3.6 Стационарность временного ряда ..... 42
3.7 ARIMA ..... 43
3.8 Прогнозирование ARIMA ..... **Ошибка! Закладка не определена.**
4 Безопасность жизнедеятельности ..... 48
4.1 Расчет системы кондиционирования воздуха ..... 53
5 Экономическая часть ..... 57
5.1 Резюме ..... 57
5.2 Трудоемкость разработки ПП ..... 57
5.3 Материальные затраты ..... 58
5.4 Затраты на электроэнергию ..... 59
5.5 Затраты на оплату труда ..... 60
5.6 Социальный налог ..... 60
5.7 Заключение по экономической части ..... 61
Заключение ..... 62
Список сокращений ..... 63
Список литературы ..... 64
Приложение А ARIMA ..... 67
Приложение Б Справка антиплагиата
Приложение В Электронная версия ДП и демонстрационные материалы (CD-R)
Приложение Г Раздаточные материалы (формат А4 – листов 16)

## Введение

Почти каждый телекоммуникационный оператор использует искусственный интеллект и глубокое обучение для улучшения обслуживания своих клиентов, прежде всего с помощью виртуальных помощников и чат-ботов. Телекоммуникационные компании получают огромное количество запросов на поддержку для установки, устранения неполадок и обслуживания. Виртуальные помощники автоматизируют и масштабируют ответы на эти запросы поддержки, что значительно сокращает бизнес-расходы и повышает удовлетворенность клиентов. В одном примере, Vodafone увидел 68-процентное улучшение удовлетворенности клиентов после представления своего чат-бота TOBi [1 ].

Одним из наиболее важных способов дать клиентам то, что они хотят, является телекоммуникационная связь для предотвращения перебоев в работе. Интеллектуальное обслуживание, обеспечиваемое глубоким обучением, является важным, хотя и более закулисным, использованием ИИ и машинного обучения, что также повышает удовлетворенность клиентов. Данные, основанные на данных, помогают компаниям контролировать оборудование, извлекать уроки из исторической информации, предвидеть отказы оборудования и активно устранять его.

Другим важным аспектом, который помогает глубокое обучение, является оптимизация сети. Самоорганизующаяся сеть (SON) [2], работающая на искусственном интеллекте, может помочь сетям постоянно адаптироваться и реконфигурироваться в зависимости от текущих потребностей. Это также полезно при проектировании новых сетей.

Телекоммуникационные компании обладают огромным количеством данных от клиентов. С помощью ИИ и глубокого обучения телекоммуникационные компании могут извлечь из этих данных значимые бизнес-идеи, чтобы они могли принимать более быстрые и качественные бизнес-решения.

В этом дипломном проекте будут представлены алгоритмы глубокого обучения и модели прогнозирования ARIMA временных рядов.

Модели ARIMA предоставляют другой подход к прогнозированию временных рядов [3]. Экспоненциальное сглаживание и модели ARIMA являются двумя наиболее широко используемыми подходами к прогнозированию временных рядов и обеспечивают дополнительные подходы к проблеме. В то время как модели экспоненциального сглаживания основаны на описании тренда и сезонности в данных, модели ARIMA стремятся описать автокорреляции в данных.

Прежде чем мы представим модели ARIMA, сначала сделаем концепцию стационарности и технику дифференцирования временных рядов. Целю дипломного проекта является построение прогноза с помощью модели ARIMA с параметрами $(p, P, q, Q)$ [3 ].

В конечном счете, искусственный интеллект и глубокое обучение позволили телекоммуникационной отрасли извлекать информацию из своих обширных наборов данных, облегчали решение проблем, более эффективно управляли повседневными делами и повышали уровень обслуживания и удовлетворенности клиентов [4 ].

Индустрия дает нам отличный пример того, как внедрение искусственного интеллекта и глубокого обучения не только выгодно для бизнеса, это было важно для выживания каждой компании и ее способности конкурировать с конкурентами.

## 1. Методы машинного обучения в задачах телекоммуникационного оператора

### 1.1 Машинное обучения

Машинное обучение (Machine Learning) — это метод анализа данных, который автоматизирует построение аналитической модели. Это отрасль искусственного интеллекта, основанная на идее, что системы могут обучаться на основе данных, выявлять закономерности и принимать решения с минимальным вмешательством человека.[9 ]

Чтобы заставить их работать, приходится изобретать дополнительные эвристики, компенсирующие несоответствие сделанных в теории предположений условиям реальных задач.

Практически ни одно исследование в машинном обучении не обходится без эксперимента на модельных или реальных данных, подтверждающего практическую работоспособность метода.

Рисунок 1.1 – Генерация данных интернета вещей на разных уровнях и модели глубокого обучения для решения проблемы абстракции знаний

Машинное обучение будет оказывать влияние на рабочие места и рабочую силу, поскольку многие рабочие места могут быть “пригодны для применения машинного обучения” . Это приведет к увеличению спроса на некоторые продукты машинного обучения и производного спроса на задачи, платформы и экспертов, необходимых для производства таких продуктов. Экономический эффект машинного обучения в докладе McKinsey определяется в разделе автоматизация работы с знаниями «использование компьютеров для выполнения задач, которые основаны на сложном анализе, тонких суждениях и творческом решении проблем»[41 В докладе упоминается, что достижения в области технологий машинного обучения, таких как глубокое обучение и нейронные сети, являются основными факторами автоматизации работы с знаниями. Естественные пользовательские интерфейсы, такие как распознавание речи и жестов, являются другими средствами, которые в значительной степени выигрывают от технологий машинного обучения. По оценкам, потенциальный экономический эффект от автоматизации

интеллектуальных работ может составить от 5,2 до 6,7 триллионов долларов в год к 2025 году. На рисунке показана разбивка этой оценки по различным профессиям. По сравнению с экономическим воздействием интернет вещей, эта оценка предполагает большее внимание к извлечению ценности из данных и потенциальному влиянию машинного обучения на экономическое положение индивидов и обществ.

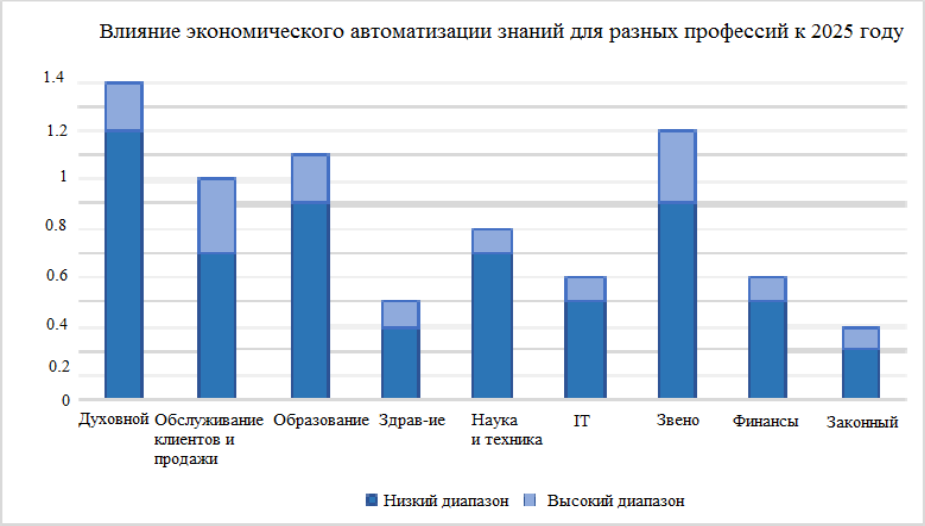



Рисунок 1.2 – Разбивка оценочных экономических последствий на 5,2 трлн долл. $6,7 трлн в год на машинное обучение в 2025 году

В последние годы появилось много приложений в различных вертикальных областях, таких как здравоохранение, транспорт, умный дом, Умный город, сельское хозяйство, образование и т. д. Основным элементом большинства из этих приложений является интеллектуальный механизм обучения для прогнозирования (т. е. регрессии, классификации и кластеризации), интеллектуального анализа данных и распознавания образов или анализа данных в целом. Среди многих подходов машинного обучения, глубокое обучение активно используется во многих приложениях Интернета вещей в последние годы. Эти две технологии (например, глубокое обучение и интернет вещей) входят в тройку лучших стратегических технологических трендов на 2017 год, которые были объявлены на симпозиуме Gartner/ITxpo 2016 [42 . Причина такой интенсивной рекламы для глубокого обучения связана с тем, что традиционные подходы машинного обучения не учитывают возникающие аналитические потребности систем интернета вещей. Вместо этого, системы интернета вещей нуждаются в различных современных подходах к анализу данных и методах искусственного интеллекта (ИИ) в соответствии с иерархией генерации и управления данными интернета вещей.

Растущий интерес к интернету вещей (ИВ) и производным от него большим данным требует от заинтересованных сторон четкого понимания их определения, структурных блоков, потенциалов и проблем. ИВ и большие

данные имеют двустороннюю связь. С одной стороны, ИВ является основным производителем больших данных, а с другой-важной целью для аналитики больших данных является улучшение процессов и сервисов ИВ. Более того, анализ больших данных ИВ доказал свою ценность для общества. Например, сообщается, что, обнаружив поврежденные трубы и починив их департамент управления парками в Майами сэкономил около миллиона долларов на своих счетах за воду.[22 ]

Данные интернета вещей отличаются от общих больших данных. Чтобы лучше понять требования к анализу данных интернета вещей, нам необходимо изучить свойства данных интернета вещей и то, чем они отличаются от свойств обычных больших данных. Данные ИВ обладают следующими характеристиками:

- крупномасштабные потоковые данные: множество устройств сбора данных распределяются и развертываются для приложений интернета вещей, а также непрерывно генерируют потоки данных. Это приводит к огромному объему непрерывных данных;
- неоднородность: различные устройства сбора данных ИВ собирают различную информацию, что приводит к разнородности данных;
- корреляция времени и пространства: в большинстве приложений Интернета вещей, сенсорные устройства прикреплены к определенному местоположению и, таким образом, имеют местоположение и отметку времени для каждого из элементов данных;
- данные с высоким уровнем шума: из-за крошечных фрагментов данных в приложениях ИВ многие из таких данных могут подвергаться ошибкам и шуму во время сбора и передачи.

Хотя получение скрытых знаний и информации из больших данных обещает улучшить качество нашей жизни, это непростая и простая задача. Для решения такой сложной и сложной задачи, выходящей за пределы возможностей традиционных подходов к умозаключению и обучению, необходимы новые технологии, алгоритмы и инфраструктуры. К счастью, последние достижения в области быстрых вычислений и передовых методов машинного обучения открывают двери для анализа больших данных и извлечения знаний, которые подходят для приложений интернета вещей.

Помимо анализа больших данных, данные интернета вещей требуют еще одного нового класса аналитики, а именно быстрой и потоковой аналитики данных, для поддержки приложений с высокоскоростными потоками данных и требующими действий, зависящих от времени (например, в режиме реального времени или почти в режиме реального времени). Действительно, такие приложения, как автономное вождение, противопожарная подготовка, поза человека (и, следовательно, состояние сознания или здоровья), требуют быстрой обработки поступающих данных и быстрых действий для достижения своей цели. Некоторые исследователи предложили подходы и структуры для быстрой потоковой аналитики данных, которые используют возможности облачных инфраструктур и сервисов.[30 ] Однако для вышеупомянутых ИВ -

приложений, в частности, нам нужна быстрая аналитика на небольших платформах (например, на краю системы) или даже на самих ИВ - устройствах. Например, автономные автомобили должны принимать быстрые решения о действиях вождения, таких как изменение полосы движения или скорости. Действительно, такого рода решения должны подкрепляться быстрой аналитикой возможных мультимодальных потоков данных из нескольких источников, включая многочисленные датчики автомобиля (например, камеры, радары, спидометр, левый/правый сигналы и т. д.).), сообщения от других транспортных средств и объектов дорожного движения (например, светофор, дорожные знаки). В этом случае передача данных на облачный сервер для анализа и возврат ответа обратно происходит с задержкой, которая может привести к нарушениям правил дорожного движения или авариям. Более критическим сценарием было бы обнаружение пешеходов такими транспортными средствами. Точное распознавание должно выполняться строго в режиме реального времени, чтобы предотвратить несчастные случаи со смертельным исходом. Эти сценарии подразумевают, что быстрая аналитика данных для ИВ должна быть рядом с источником данных или у него, чтобы устранить ненужные и запретительные задержки связи.

Модели глубокого обучения в целом приносят два важных улучшения над традиционными подходами машинного обучения в двух этапы обучения и прогнозирования. Во-первых, они уменьшают потребность для ручной работы и инженерных наборов функций, которые будут использоваться для обучения. Следовательно, некоторые функции, которые могут не быть очевидным для человеческого взгляда может быть легко извлечен из модели глубокого обучения. Кроме того, модели глубокого обучения улучшают точность.

Данные интернета вещей могут передаваться непрерывно или накапливаться как источник больших данных. Потоковые данные относятся к данным генерируется или захватывается в течение крошечных интервалов времени и потребности быть быстро проанализированным для извлечения немедленных решений или принимать быстрые решения. Большие данные относятся к огромным наборам данных, которые обычно используемые аппаратные и программные платформы не способны хранить, управлять, обрабатывать и анализировать. Эти два подхода следует относить по-разному, так как их требования к аналитические реакции-это не одно и то же. Информацию из больших данных аналитика может быть доставлена после нескольких дней генерации данных, но понимание от анализа потоковых данных должно быть готово в диапазон от нескольких сотен миллисекунд до нескольких секунд.

Многие исследовательские попытки предполагали анализ потоковых данных это может быть в основном развернуто на высокопроизводительных вычислениях системы или облачные платформы. Анализ потоковых данных основаны на параллелизме данных и инкрементальной обработки. При параллелизме данных большой набор данных секционируется в несколько меньших наборов данных, на которых параллельная аналитика выполняется

одновременно. Инкрементная обработка ссылается для получения небольшой партии данных, которые будут быстро обработаны в конвейер вычислительных задач. Хотя эти приемы уменьшают время задержки для возврата ответа из потоковой передачи данных аналитической структуры, они не являются лучшим возможным решением для жестких по времени приложений интернета вещей. Однако, потоковые данные анализа ближе к источнику данных (например, устройства интернета вещей или пограничные устройства) необходимость параллелизма данных и инкрементальная обработка менее чувствительна, так как размер данных в источник позволяет его быстро обрабатывать. Однако, быстрая аналитика на устройствах интернет вещей создает свои собственные проблемы, ограничение вычислительных, запоминающих и энергетических ресурсов у источника данных.[3]

### 1.2 Характеристика Big Data

Хорошо известно, что интернет вещи является одним из основных источников больших данных, так как она основана на подключении огромного количества умных устройств в интернете, чтобы сообщить о своем часто состоянии их окружения. Распознавание и извлечение значимого паттерны из огромных необработанных входных данных - это основная аналитика больших данных, поскольку она приводит к более высоким уровням понимания для принятия решений и прогнозирование тенденций. Поэтому извлечение этих идеи и знания из больших данных имеют чрезвычайное значение важность для многих предприятий, так как это позволяет им получить конкурентное преимущество. В социальных науках Гильберт сравнивает влияние аналитики больших данных с влиянием изобретение телескопа и микроскопа для астрономии и биологии.[4 ]

В нескольких работах были описаны общие черты больших данных из разных аспектов по объему, скорости и разнообразии. Однако мы принимаем общее определение больших данных для характеристики больших данных ИВ через следующие функции"6V":

- объем: объем данных является определяющим фактором для рассмотрения набора данных как большие данные или традиционные данные. Количество генерируемых данных с помощью устройств Интернета вещей составляет гораздо больше, чем раньше, и четко соответствует этой функции;
- скорость: скорость производства и обработки больших данных интернета вещей достаточно высока, чтобы поддерживать доступность больших данные в режиме реального времени. Это оправдывает потребности в передовых инструменты и технологии для эффективной работы аналитики учитывая такой высокий темп производства данных;
- разнообразие: как правило, большие данные поступают в различных формах и типы. Он может состоять из структурированных, полу структурированных и неструктурированных данных. Большое разнообразие типов данных может быть производится с помощью интернета вещей, таких как текст, аудио, видео, сенсорные данные и так далее;

- правдивость: правдивость относится к качеству, последовательности и достоверность полученных данных, что в свою очередь приводит к точной аналитике. Это свойство нуждается в особом внимании, чтобы удержать для приложений интернета вещей, особенно с толпой зондирования данных;

- вариабельность: это свойство относится к различным темпам роста потока данных. В зависимости от характера интернет вещей - приложений, различные компоненты, генерирующие данные, могут быть несовместимы поток данных. Кроме того, это возможно для источника данных иметь различные скорости загрузки данных на основе конкретных попыток. Например, приложение службы парковки, которое использует датчики имеют пиковую нагрузку данных;

- ценность: ценность-это преобразование больших данных в полезные. информации и идеи, которые приносят конкурентное преимущество в организации. Значение данных сильно зависит как от базовых процессов службы и способ передачи обработанных данных. Например, определенное применение (например, медицинское мониторинг жизненно важных признаков), возможно, потребуется захватить все данные датчиков, в то время как служба прогноза погоды может понадобиться просто случайно образцы данных с его датчиков.

Помимо вышеупомянутых свойств, исследователи определили другие характеристики, такие как:

- большие данные могут быть побочным продуктом или следом цифровой активность или взаимодействие интернет вещей. Использование Google наиболее общие условия поиска для прогнозирования сезонного гриппа - это хороший пример такого цифрового побочного продукта;

- системы больших данных должны быть горизонтально масштабируемыми, большие источники данных должны быть в состоянии расширенными до несколько наборов данных. Этот атрибут также приводит к сложности больших данных, который в свою очередь накладывает другие такие проблемы, как передача и очистка данных.[4

Выполнение аналитики над непрерывными потоками данных, как правило, называется потоковой обработкой или иногда сложной обработкой событий. Предложили платформу анализа больших данных для ИВ для поддержки атрибуты объема и скорости анализа данных интернета вещей. То интеграция интернета вещей, больших данных и аналитики потоковой передачи данных открытый вопрос, который нуждается в дополнительном исследовании, также был изучен как часть этой работы. Однако предлагаемая ими структура такова: предназначен для развертывания на облачных инфраструктурах.

Кроме того, их внимание сосредоточено на аспекте управления данными фреймворка и не используют продвинутые модели машинного обучения, такие как глубокое обучение. Другие готовые продукты, такие как Apache Storm также доступно для анализа в реальном времени в облаке. Большой разрыв в этой области отсутствуют структуры и алгоритмы, которые могут быть

развернутым на тумане (т. е. на краю системы) или даже на устройствах интернет вещей.[25

## 2 Методы глубокого обучения

### 2.1 Архитектура глубокого обучения

Глубокое обучение состоит из контролируемых или неконтролируемых методов обучения на основе многих слоев искусственных нейронных сетей, которые способны изучать иерархические представления в глубинных архитектурах. Архитектуры глубокого обучения состоят из нескольких обработка слоев. Каждый слой способен производить нелинейные ответы основаны на данных из его входного слоя. Функциональность глубокого обучения имитируется из механизмов человека и нейроны для обработки сигналов.[23

Архитектуры глубокого обучения получили больше внимания в последние годы по сравнению с другими традиционными подходами машинного обучения. Такие подходы рассматриваются как неглубоко структурированные изучение архитектур глубокого обучения. Поисковик Google показывает тенденцию поиска пяти популярных машин изучение алгоритмов в Google тренд, в которых глубокое обучение становится более популярный среди других.

Методы глубокого обучения были развиты поверх традиционных нейронных сетях обратной связи. Небольшой размер из тренировочных данных был еще один фактор, который приводит к переобучению модели.. Эти вычислительные ограничения были устранены в последнее время из-за аппаратных достижений в целом и развития графических процессоров и аппаратного обеспечения в частности, ускорители. Помимо структурных аспектов и значение глубины архитектуры глубокого обучения, а также аппаратного обеспечения:

- использование выпрямленных линейных единиц в качестве функции активации;
- введение методов отсева;
- случайная инициализация для весов сети;
- решение проблемы снижения точности обучения по остаточному принципу; обучение сети.[2

Одно из преимуществ архитектур глубокого обучения, по сравнению с традиционными искусственными нейронными сетями, это то, что методы глубокого обучения могут изучать скрытые функции из исходных данных. Каждый слой тренируется на наборе объектов на основе результатов предыдущего слоя. Самые внутренние слои может распознавать более сложные признаки, так как они агрегируют и комбинируют объекты из предыдущих слоев. Это называется иерархия признаков. Например, в случае лица модель опознавания, необработанные данные изображения портретов как вектор пиксели подаются на модель в ее входном слое. Каждый скрытый слой затем

можно узнать больше абстрактных объектов из предыдущего слоя. выходы, например, первый скрытый слой идентифицирует линии и края, второй слой идентифицирует части лица, такие как нос, глаза, и т.д., а третий слой объединяет в себе все предыдущие функции, чтобы создать лицо.

Модели глубокого обучения основаны на эмпирических оценках и до сих пор не имеют конкретных аналитических основ. Более того, здесь нет четкой границы между глубокими и мелкими сетями на основе количество скрытых слоев. Как правило, нейронные сети с двумя или более скрытых слоев, которые включают в себя последние передовые обучающие алгоритмы рассматриваются как глубинные модели. Также, рассматриваются рекуррентные нейронные сети с одним скрытым слоем, так как у них есть цикл на единицах скрытого слоя, которые могут быть развернуты в эквивалентную глубокую сеть.

## 2.2 Модели и архитектуры глубокого обучения

В этом разделе представляется краткий обзор нескольких распространенных моделей глубокого обучения, а также самые передовые архитектуры.

Глубокая нейронная сеть состоит из входного слоя, нескольких скрытых слоев, и выходной слой. Каждый слой включает в себя несколько блоков, называемых нейроны.[20

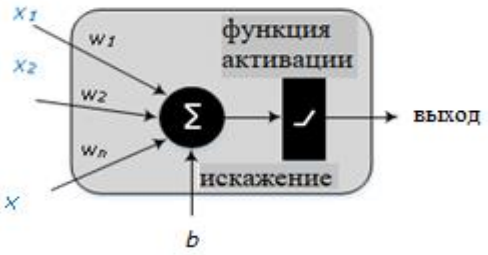



Рисунок 2.1 – Структура нейрона

Нейрон получает несколько входов, выполняет взвешенное суммирование по его входам, затем полученная сумма проходит через функцию активации для получения выходного сигнала. Каждый нейрон имеет вектор весов, связанных с его входным размером, а также смещение это должно быть оптимизировано во время тренировочного процесса. Рисунок 2.1 изображает структуру нейрона.

В процессе обучения входной слой присваивает (обычно произвольно) взвешивает входные обучающие данные и передает их к следующему слою.

Каждый последующий слой также присваивает веса к их входу и производит их выход, который служит как ввод для следующего слоя. На последнем слое конечный результат представление модели производится прогнозирование. Функция потерь определяет правильность этого

предсказания путем вычислений частота ошибок между прогнозируемыми и истинными значениями. Один алгоритм оптимизации, такой как стохастический градиентный спуск используется для регулировки веса нейронов путем вычисления градиента функции потерь. Частота ошибок распространяется обратно по сети на входной слой (известный как алгоритм обратного распространения). Тогда сеть повторяет этот тренировочный цикл, на каждый нейрон в каждом цикле, пока частота ошибок не упадет ниже желаемого порога. В этот момент глубокая нейронная сеть обучена и готова к выводу. На рисунке 2.2 показан механизм обучения высокого уровня для моделей глубокого обучения.

В широкой классификации модели глубокого обучения делятся на три категории: а именно генеративные, дискриминативные и гибридные модели. Хотя и не являясь твердой границей, различительной модели обычно обеспечивают контролируемые подходы к обучению, в то время как генеративные модели используются для бесконтрольного обучения. Гибридный модели включают в себя преимущества как дискриминационных, так и генеративные модели.[20

### 2.3 Сверточные нейронные сети

Для задач, основанных на зрении, глубокая нейронная сеть с плотной связью между слои трудно поддаются дрессировке и плохо масштабируются. Один важной причиной является свойство инвариантности перевода таких модели. Таким образом, они не изучают особенности, которые могут трансформироваться в изображении (например, вращение руки при обнаружении позы). Сверточная нейронная сеть решила эту проблему, поддерживая эквивалентность вычисления. Сверточная нейронная сеть получает 2 - D вход (например, изображение или речевой сигнал) и извлекает высокоуровневые характеристики через ряд скрытых слоев. Слои хиддона состоят из слоев свертки, а также полностью соединенных слоев в конце. Слой свертки находится в ядре сверточной нейронной сети и состоит из набора обучаемых параметров, называемых фильтрами, которые имеют одинаковые значения. форма как форма входного сигнала, но с меньшими размерами. В тренировочный процесс, фильтр каждого сверточного слоя идет через весь входной объем (например, в случае изображения, оно идет по ширине и длине изображения) и вычисляет внутреннее произведение входного сигнала и фильтра. Это вычисление по всему входу выводится карта объектов фильтра.[5

Таблица 2.1 - Резюме моделей глубокого обучения

| Модель | Категория | Модель обучения | Типичные входные данные | Характеристики |
|---|---|---|---|---|
| AE | Генеративный | Неконтролируемый | Различный | -Подходит для извлечения признаков, уменьшения размерности<br>-Одинаковое количество входных и выходных блоков<br>-Выход реконструирует входные данные<br>-Работает с немаркированными данными |
| RNN | Дискриминационный | Контролируемый | Серийный, временные ряды | -Обрабатывает последовательности данных через внутреннюю память<br>-Полезно в приложениях Интернета вещей с зависящими от времени данными |
| RBM | Генеративный | Неконтролируемый Контролируемый | Различный | -Подходит для извлечение, уменьшение размерности и классификация<br>-Дорогостоящая процедура обучения |
| DBN | Генеративный | Неконтролируемый Контролируемый | Различный | -Подходящая для иерархической структуры обнаружение особенностей<br>-Жадное обучение самого себя сеть слой за слоем |
| LSTM | Дискриминационный | Контролируемый | Серийный, временные ряды, данные зивасящие от длительного времени | -Хорошая производительность с данными на продолжительный период времени<br>-Доступ к ячейке памяти есть защищен воротами |
| CNN | Дискриминационный | Контролируемый | 2-D (изображение, звук и т.д.) | -Сверточные слои берут самая большая часть вычислений<br>-Меньше соединения по сравнению в Дннс<br>-Нужна большая тренировка набор данных для визуальных задач |
| VAE | Генеративный | Полу-контролируемый | Различный | -Класс автоматических энкодеров<br>-Пригодных для дефицита помеченные данные |
| RNN | Дискриминационный | Контролируемый | Серийный, временные ряды | -Обрабатывает последовательности данных через внутреннюю память<br>-Полезно в приложениях Интернета вещей с зависящими от времени данными |

Еще одним строительным блоком сверточной нейронной сети является объединение слоев, которые работают на картах объектов. Цель иметь объединение слоев заключается в уменьшении пространственного размера представления, для того чтобы сократить количество параметров, так и время вычислений и уменьшить вероятность переоснащения. Максимальное объединение в пул - это распространенный подход, который разделяет входное пространство в неперекрывающиеся области и выбирает максимальное значение для каждого региона.

Последним важным компонентом в сверточной нейронной сети является выпрямитель, который состоит из нейронов с активационной функцией в виде f (x) = max (0, x). Введение этой функции активации в сверточной нейронной сети результаты в более быстром времени тренировки без влияния обобщение

сети в осмысленном негативном ключе. На рисунок 2.3 показана структура канала сверточной нейронной сети.

Основное различие между сверточной нейронной сети и полностью подключенными сети заключается в том, что каждый нейрон в сверточной нейронной сети связан только с небольшим подмножество входных данных. Это уменьшает общее число параметры в сети и увеличивает временную сложность учебный процесс. Это свойство называется локальным подключением.

Многие устройства интернета вещей, такие как дроны, смартфоны и умные подключенные автомобили, оснащены камерами. Канал СНН архитектура и ее вариации были исследованы в течение длительного времени, разнообразие сценариев применения, в которых задействованы эти устройства. Некоторые типичные приложения включают в себя прогноз наводнений или оползней через изображения дронов, обнаружение болезней растений с помощью изображений растений на смартфонах и обнаружение дорожных знаков с помощью камеры транспортных средств.[1]

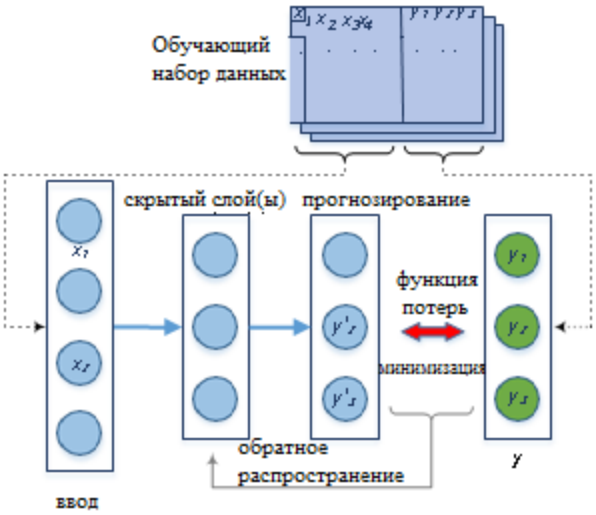
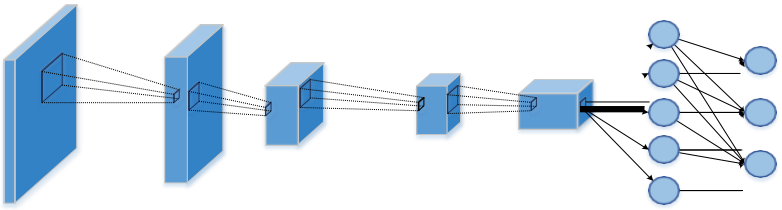



Рисунок 2.2 – Общий механизм обучения модели



Входной слой Сверточный +ReLu Объединяющий слой1 Сверточный +ReLu Объединяющий слой2 выход подключен

Рисунок 2.3 – Архитектура сверточной нейронной сети

## 2.4 Рекуррентные нейронные сети

Во многих задачах прогнозирование зависит от нескольких предыдущих образцов, также нужно проанализировать последовательности входных данных. В таких приложениях, обратная нейронная сеть неприменима поскольку он не предполагает никакой зависимости между входом и выходом слоев. Рекурентные нейронные сети (РНН) были разработаны для решения этой проблемы в последовательные (например, речь или текст) или проблемы временных рядов с различной длиной. Обнаружение поведения водителей в интеллектуальных транспортных средствах идентификация индивидуальных моделей движения, а оценка энергопотребления домохозяйства-это несколько примеры, где могут быть применены рекурентные нейронные сети.[2]

Вход в РНН состоит как из текущей выборки, так и из предыдущей наблюдаемой образец. Другими словами, выход РНН на временном шаге t-1 влияет на выход на временном шаге t. каждый нейрон оснащен петля обратной связи, которая возвращает текущий выходной сигнал в качестве входного для следующий шаг. Эта структура может быть выражена так, что каждый нейрон в РНН имеет внутреннюю память, которая сохраняет информация о вычислениях из предыдущего ввода.

Чтобы обучить сеть, расширение обратного распространения алгоритм, используется обратным перемещением во времени. Из-за существования циклов на нейронах, здесь нельзя использовать оригинальное обратное распространение, так как оно работает, основываясь на выводе ошибок относительно веса в их верхнем слое, в то время как у нас нет модели сложенного слоя в РНН. На рисунок 2.4 показана структура РНН и развернутая концепция.

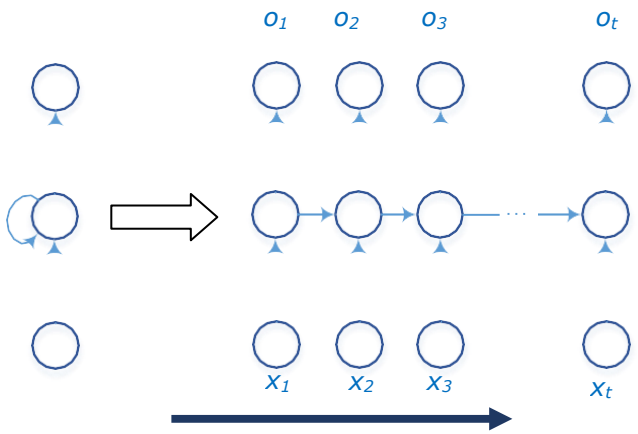



Рисунок 2.4 – Структура рекуррентной нейронной сети

Традиционные РНН можно рассматривать как глубокие модели, так как их можно рассматривать как несколько нелинейных слоев нейронов между входным и выходным слоями, когда они находятся развернуто во времени. Однако, учитывая архитектуру и функциональность РНН, скрытые слои в РНН предполагается, что они обеспечивают память вместо иерархической

представление признаков. Существует несколько подходов чтобы сделать РНН глубже, включая добавление большего количества слоев между входные и скрытые слои, укладка дополнительных скрытых слоев и добавление дополнительных слоев между скрытыми слоями и выводом слой.

Долгая краткосрочная память (ДКП) - это расширение РНН. Различные вариации ДКП были предложены, хотя большинство из них последовали та же конструкция исходной сети. ДКП использует концепция ворот для своих блоков, каждый вычисляет значение между 0 и 1 на основе их входных данных. В дополнение к контуру обратной связи для хранения информации каждый нейрон в ДКП (также называемый а ячейка памяти) имеет мультипликативный элемент забывания, элемент чтения и напишите ворота. Эти ворота вводятся для контроля доступа к ячейкам памяти и предотвратить их от возмущения путем нерелевантные входные данные. Когда ворота забвения активны, нейрон записывает свои данные в себя. Когда ворота забвения выключены посылая 0, нейрон забывает свое последнее содержание. Когда строб записи установлен в 1, другие подключенные нейроны могут писать в него нейрон. Если элемент чтения установлен в 1, то подключенные нейроны могут прочитать содержимое нейрона. На рисунке 2.5 изображена эта структура.[3]

Важным отличием ДКП по сравнению с РНН является что блоки ДКП активно контролируют клеточные состояния и гарантировать, что они не деградируют. Затворы могут использовать сигмовидную или как их активирующую функцию. На самом деле, эти активационные функции вызывают проблему исчезающего градиента во время обратного распространения в фазе обучения других моделей использующий их. Изучая, какие данные следует запоминать в ДКП, сохраненные вычисления в ячейках памяти не искажаются. время. ВРТТ является распространенным методом обучения сети, чтобы минимизируйте ошибку.

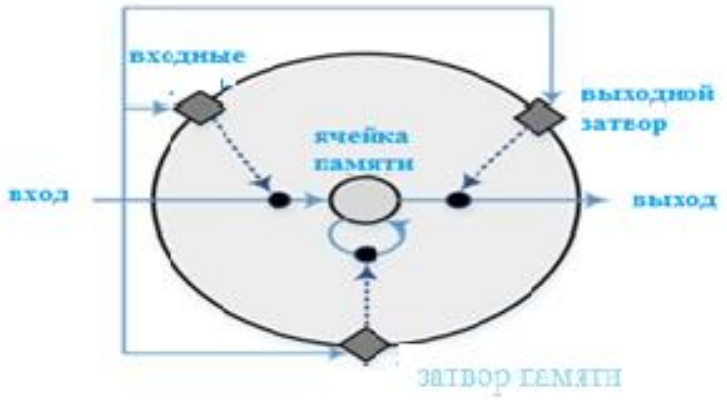



Рисунок 2.5 – Структура ячейки памяти ДКП. Сплошные линии стрелок показывают поток данных, а пунктирные линии стрелок- сигналы, поступающие от Гейтса

Когда данные характеризуются длительной зависимостью во времени, Модели ДКП работают лучше, чем модели РНН. Это длительное запаздывание зависимости можно наблюдать в приложениях интернета вещей, например, распознавание человеческой деятельности, прогнозирование образовательных производительность в онлайновых программах и прогнозирование катастроф на основе по мониторингу окружающей среды, чтобы назвать несколько.

Автоэнкодеры состоят из входного слоя и выходного слоя, которые являются соединены через один или несколько скрытых слоев. Но есть одинаковое количество входных и выходных блоков. Эта сеть направлена на реконструированный входящим путем преобразования входов в выходы с помощью самый простой из возможных способов, такой, чтобы он не искал входные данные. Этот вид нейронных сетей был использован главным образом для решения неконтролируемых задач обучения, а также для перевода обучение. Из-за их поведения построения входных данных в выходном слое, который в основном используется для диагностики и задачи обнаружения неисправностей. Это представляет большой интерес для промышленных предприятий интернет вещей для обслуживания многих приложений, таких как диагностика неисправностей в аппаратные устройства и машины, а также обнаружение аномалий в производительность сборочных линий. Автоэнкодеры имеют два основных компонента: шифратор и декодер. Кодировщик получает входной сигнал и преобразует его в новое представление, которое обычно называют кодом или латентным переменная. Декодер получает сгенерированный код по адресу: шифратор, и преобразует его в реконструкцию оригинала вход. Процедура обучения в автоэнкодере включает в себя минимизацию ошибок реконструкции, то есть отображение выходных и входных данных минимальное различие. Рисунок 2.9 иллюстрирует структуру автоэнкодера.[3]

Вариационные автоэнкодеры, введенные в 2013 году, являются популярной генеративной моделью, чьи предположения о структуре данных не сильны, при этом имеет быстрый процесс обучения через обратное распространение. Кроме того, эта модель имеет был использован для полу-контролируемого обучения. Поэтому он подходит для работающих с разнообразными данными и скудость помеченных данных. Такие приложения включают в себя обнаружение отказов в датчике или срабатывании уровней и вторжений обнаружение в системах безопасности. Для каждой точки данных x существует является вектором соответствующих латентных переменных, обозначаемых $z$. Обучающая архитектура вариационного автоэнкодера состоит из кодера и декодера с параметрами $\varphi$ и $\theta$ соответственно. Фиксированный распределение формы $q\varphi(z|x)$ помогает кодировщику в оценке заднее распределение $p\theta\ (z|x)$. Модель состоит из двух частей: сети: одна генерирует образцы, а другая выполняет приблизительный вывод.

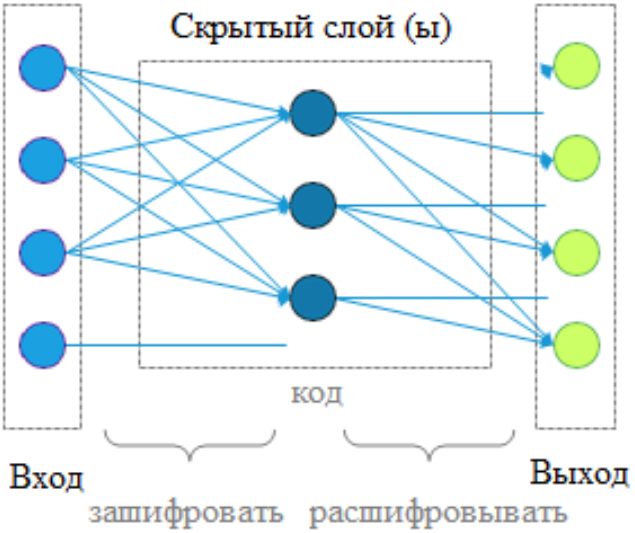



Рисунок 2.6 – Структура сети автоэнкодер

## 2.5 Генеративные состязательные сети

Генеративные состязательные сети состоят из двух нейронных сетей, а именно генеративные и дискриминативные сети, которые работают вместе, чтобы произвести синтетическое и высокое качество данные. Прежняя сеть (она же генератор) – это отвечает за генерацию новых данных после того, как изучит распределение данных из обучающего набора данных. Последняя сеть (дискриминатор) выполняет различение между реальными данными (исходящие из обучающих данных) и поддельные входные данные (исходящие из генератора). Генеративная сеть оптимизирована для производства входные данные, которые являются обманчивыми для дискриминатора (т. е. дискриминатор не может легко отличить, является ли это подделкой или настоящий).[5 Рисунок 2.7 изображена концепция Генеративной состязательной сети.

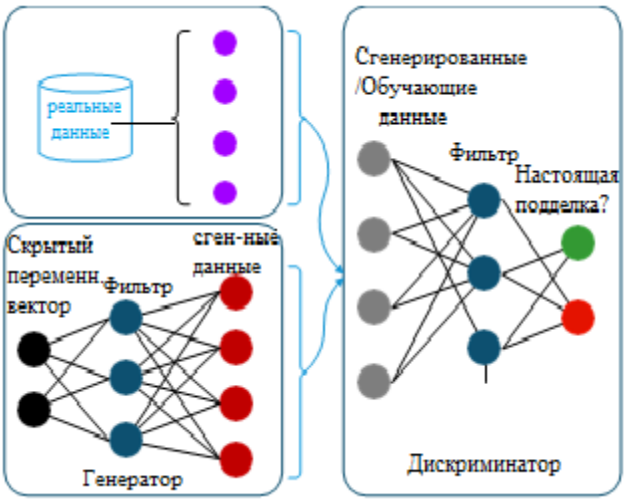



Рисунок 2.7 – Концепция генеративной состязательной сети.

Целевая функция в ГСС основана на минимаксе игры, такие, что одна сеть пытается максимизировать значение функция и другая сеть хочет свести ее к минимуму. В каждом из них в этой воображаемой игры, генератор, желающий обмануть дискриминатор, воспроизводящий путем получения выборочных данных из случайных шумов. С другой стороны, дискриминатор получает несколько примеры реальных данных из обучающего набора вместе с выборками от генератора. Его задача состоит в том, чтобы различать реальное и поддельное данные. Считается, что дискриминатор работает удовлетворительно если его классификации верны.

Генератор также выполняет хорошо, если его примеры обманули дискриминатора. Оба затем Параметры дискриминатора и генератора обновляются до будьте готовы к следующему раунду игры. Дискриминатора по выход помогает генератору оптимизировать свои сгенерированные данные для следующий раунд.

В приложениях интернета вещей ГСС можно применять для сценариев, которые требуется создание чего-то нового из имеющегося данные. Это может включать приложения в локализации и способ нахождения, где генераторная сеть в ГСС производит потенциальные пути между двумя точками, в то время как дискриминатор определяет, какие пути выглядят жизнеспособными. ГСС также очень полезны для развития услуг для слабовидящих людей. Создания текста из данного изображения и другой модели для выполнения преобразования текста в речь. В изображении обработка исследований с использованием ГСС, большое количество реальных снимков знаменитостей были проанализированы, чтобы создать новую подделку образов таких, что человек не может определить, реальны ли образы или нет.[1]

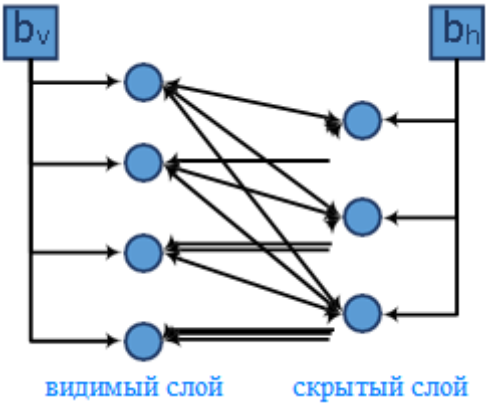



Рисунок 2.8 – Структура ограниченной машины Больцмана. Видимое и скрытое слои имеют отдельный уклон

Ограниченная машина Больцмана (ОМБ) – это стохастическая модель, состоящая из двух слоев: видимый слой, содержащий известные нам входные данные, и скрытый слой, содержащий скрытые переменные. Ограничение применяется к связности сравниваемых нейронов к машине Больцмана. ОМБ

должен построить двудольный граф, так что каждый видимый нейрон должен быть соединен со всеми скрытыми нейронов и наоборот, но никакой связи между ними нет. две единицы в одном слое. Кроме того, блок смещения подключен ко всем видимым и скрытым нейронам. ОМБ можно штабелировать для формирования. Они также являются строительным блоком глубокой веры сети.[8

Обучающие данные присваиваются видимым единицам. Обучение процедура может использовать алгоритмы обратного распространения и градиентного спуска для оптимизации веса сети. Цель обучение ОМБ заключается в максимизации произведения всех вероятностей видимых единиц измерения. Функциональность ОМБ аналогична автоэнкодеру, поскольку он использует прямое питание для вычисления латентных переменных, которые, в свою очередь, используются для реконструкции входных данных с использованием обратной связи кормление.

ОМБ может выполнять извлечение объектов из входных данных. Этот происходит путем моделирования распределения вероятностей по а набор входных данных, который представлен в виде набора скрытых единиц измерения. Например, имея набор любимых фильмов отдельных лиц, Ан Модель ОМБ может иметь видимый слой, состоящий из нескольких нейронов как по количеству доступных фильмов, так и по скрытому слой, состоящий из трех нейронов, представляющих три различных такие жанры, как драма, боевик и комедия. Итак, основываясь на применение, скрытый слой можно рассматривать как выходной слой. Или его можно дополнить дополнительным классификатором слой для выполнения классификации на основе извлеченных объектов.

От типов потенциальных применений, где может использоваться, мы называем крытую локализацию, потребление энергии прогнозирование, прогнозирование заторов движения, анализ положения, и как правило, любое приложение, которое извлекает выгоду из извлечения наиболее важные функции из доступных.

Сеть глубокого убеждения (СГУ) - это тип генеративных РНН, которые состоят из видимый слой (соответствующий входным данным) и несколько скрытых слоев (соответствующих скрытым переменным). Они могут извлечь иерархическое представление обучающих данных, а также реконструируйте их входные данные. Путем добавления слоя классификатора, он может быть использован для задач прогнозирования.[3]

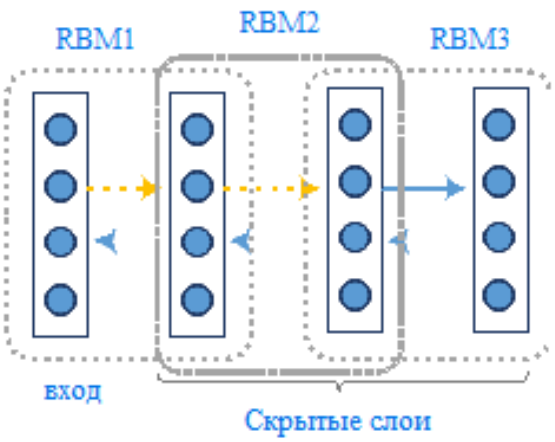



Рисунок 2.9 – Структура сети глубоких убеждений. Пунктирные стрелки показывают путь извлечения объекта, а сплошные стрелки- генеративный путь

Обучение СГУ выполняется слой за слоем, так что каждый слой рассматривается как ОМБ, обученный поверх предыдущего обученного слоя. Этот механизм делает СГУ эффективным и быстрым алгоритмом в глубоком обучении. Для данного скрытого слоя в СГУ скрытый слой предыдущего ОМБ выступает в качестве входного слоя. На рисунке 2.10 показана структура типичного СГУ. Несколько приложений могут извлечь выгоду из структуры базы данных, например, классификация обнаружения неисправностей в промышленных средах, идентификация угроз в системах оповещения о безопасности и извлечение эмоциональных характеристик из изображений. Лестничные сети были предложены в 2015 году для поддержки бесконтрольного обучения

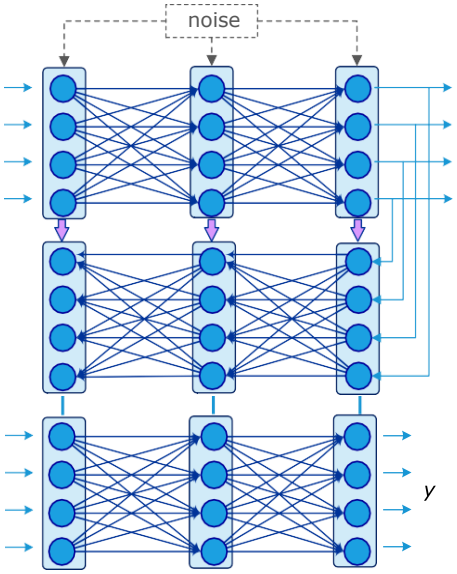



Рисунок 2.10 – Лестничная сетевая структура с двумя слоями

Быстрые и в реальном масштабе времени архитектуры глубокого обучения, исследование работает для быстрого и реального времени аналитики с использованием модели глубокого обучения над потоком данных все еще находятся в зачаточном состоянии. Использовании этой архитектуры для локализации в помещении, алгоритм основан на Wi-Fi дактилоскопии, в котором связь моделей может хорошо переносить динамические изменения окружающей среды, в то время как по-прежнему показывают хорошую точность.[10

Для сверточных сетей архитектура, предложенная Renom и др., называемый быстрее РНН, нацелен на обнаружение объектов на изображениях в режиме реального времени. Обнаружение объектов на изображениях требует дополнительных вычислений и, следовательно, потребляет больше энергии по сравнению с классификацией изображений, так как система обладает большим количеством потенциальных предложения объектов, которые должны быть оценены [43]. Предложенное архитектура основана на применении алгоритмов предложения области в полных СНН, которые выполняют предсказание границ объекта и вычисление баллов в каждой позиции одновременно. Их оценка предложенной архитектуры обнаружения объектов указывает, что время работы системы находится между 5-17 кадров в секунду при условии, что исходные входные кадры масштабируются таким образом, что самая короткая сторона изображения будет 600 пикселей. Мао и др. также использовал быстрый P-CHH для встроенные платформы, сообщающие о времени выполнения 1.85 кадры, масштабированные до 600 пикселей в самой короткой стороне встроенного CPU + GPU Платформа, которые, как было показано, являются энергоэффективными с близким к реальному времени представлением [43 . Однако, для задачи обработки изображений, мы можем рассматривать подход, чтобы быть действительно в реальном времени, когда он может обрабатывать и анализировать 30 кадров в секунду или лучше. Redmon и другие, которые достигли производительности 45 кадров в секунду для ввода, изображения увеличены до 448 × 448, и даже уменьшена его версия, достигающий 155 кадров в секунду, которые подходят для smart камеры.

Архитектуры глубокого обучения также были использованы совместно в других подходах машинного обучения, чтобы сделать их более эффективными. Нелинейная аппроксимация функций глубокого обучения моделей, которые может поддерживать тысячи или даже миллиарды параметров. сильная мотивация использовать этот метод в других машинах подходы к обучению нуждаются в таких функциях. Более того, автоматическое извлечение признаков в глубоких моделях-это другое. мотивируя причину эксплуатировать эти модели совместно с другой подход. В следующих подразделах приводится краткое изложение предусмотрены такие подходы, которые подходят для сценариев интернета вещей.[1]

Глубокое обучение представляет собой комбинацию обучения с помощью ГНС. Она направлена на создание программные агенты, которые могут самостоятельно научиться устанавливать успешная политика для

получения максимального долгосрочного вознаграждения. При таком подходе обучение с подкреплением находит наилучшую политику действий над набор состояний в среде из модели ГНС. Потребность в ГНС в модели обучении с подкреплением становится очевидной, когда базовая среда может быть представлена в виде большое количество государств. В такой ситуации традиционное обучение с подкреплением является недостаточно эффективно. Вместо этого модель глубокого обучения может быть использована для приближения значения действия, чтобы оценить качество действия в данном состоянии. Системы, использующие ГОП в своих системах контекст, находятся в зачаточном состоянии, но уже показали очень обнадеживающие результаты. В области интернета вещей работа, использует ГОП в полу управляемой установке для локализации в умных условиях кампуса. Функция вознаграждения является обратной по отношению к расстоянию ошибка к заранее определенной цели, такая что обучающий агент получает больше наград, когда приближается к своей цели цель и наоборот.

Передача обучения с помощью глубоких моделей: перенос обучения, который попадает в область предметной адаптации и многозадачное обучение, предполагает адаптацию и улучшение обучения в новой области путем переноса представление знаний, которое было получено из данных родственной области. Переносимое обучение - это интересный процесс, потенциальное решение для многих приложений Интернета вещей, где сбор обучение данным-задача не из легких. Например, рассматривая обучение системы локализации с помощью BLE или Wi-Fi дактилоскопии используя смартфоны, значения в то же время и расположение для различных платформ (например, iOS и Android) различаются. Если у нас есть обученная модель для одной платформы, то модель может быть перенесен на другую платформу без повторного сбора еще один набор обучающих данных для новой платформы.[1]

Модели ГО хорошо подходят для передачи обучения из-за их способности изучать как низкоуровневые, так и абстрактные представления из входных данных. В частности, шумящие автоэнкодеры и другие варианты автоэнкодеров, как было показано, выполняют очень хорошо в этой области. Передача обучения с помощью ГНС все еще является постоянная и активная исследовательская область в сообществе ИИ, и мы не видели сообщений о реальных приложениях в ИВ.[1]

Как идет поток данных, генерируемых из ИВ-приложений через облачные платформы для анализа, роль онлайн алгоритмы машинного обучения становятся более выделенными, так как модель обучения нуждается в обновлении за счет инкрементного объема данных. Это противоречит тому, что современные технологии поддержка, которая основана на методах пакетного обучения, где весь набор обучающих данных должен быть доступен для обучения. и, следовательно, обученная модель не может развиваться по новой данные. Несколько исследовательских работ сообщают о применении онлайн-обучения методики на различные модели ГО, включая наборный восстановление автоэнкодеров и ОМБ.

Быстрый рост интереса к использованию архитектур ГО в различные домены были поддержаны путем введения нескольких ГО структур в последние годы. Каждый фреймворк имеет свой собственный прочность на основе поддерживаемых архитектур ГО, оптимизация алгоритмы, а также простота разработки и развертывания. Некоторые из этих фреймворков широко использовались в исследованиях для эффективного обучения РНН. В этом разделе мы рассмотрим некоторые из этих рамок.

H2O: H2O-это платформа машинного обучения, которая обеспечивает интерфейсы для R, Python, Scala, Java, JSON и CoffeeScript / JavaScript . H2O может работать в различных режимах в том числе в автономном режиме, на Hadoop, или в кластере. В дополнение к обычным алгоритмам машинного обучения, H2O включает в себя реализацию алгоритма ГО, который основан о нейронных сетях обратной связи, которые могут быть обучены с обратным распространением. ГО-автоэнкодеры-H2O основан на стандарте глубокой (многослойная) архитектуры нейронных сетей, где вся сеть изучается вместе, а не укладывается слоями.[26

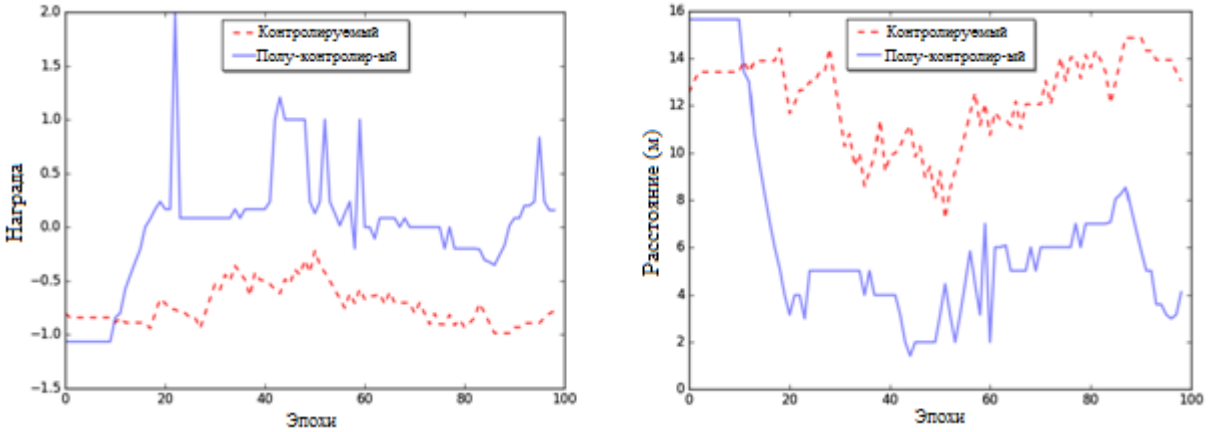



Рисунок 2.11 – Глубокое обучение (под наблюдением): получение вознаграждений (слева) и их соответствующее измерение точности (справа).

Tensorflow: изначально разработан для проекта Google Brain, Tensorflow - это библиотека с открытым исходным кодом для машинного обучения системы, использующие различные виды ГНС. Он используется многие продукты Google, включая Google Search, Google Maps и Street View, Google Translate, YouTube и некоторые другие продукты. Tensorflow использует графовые представления для построения нейронных сетевая модель. Разработчики также могут воспользоваться преимуществами. Тензорная доска, представляющая собой пакет для визуализации нейронной сети, моделируйте и наблюдайте процесс обучения, включая обновление параметры. Керас2 также обеспечивает высокий уровень программирования абстракция для тензорного потока.[26

Torch: Torch - это фреймворк с открытым исходным кодом для машинного обучения, содержащее широкий спектр алгоритмов ГО для легкого разработка моделей DNN. Он был разработан на основе Язык программирования Lua должен быть легким и быстрым для обучения алгоритмов ГО. Он используется несколькими компаниями и исследовательские лаборатории, такие как Google, Facebook и Twitter. Он поддерживает разработку моделей машинного обучения для ЦП и графических процессоров, и предоставляет мощные пакеты распараллеливания для обучения DNS.

Theano: Theano - это фреймворк с открытым исходным кодом на основе Python для эффективных алгоритмов машинного обучения, которые поддерживают компиляция для процессоров и графических процессоров. Он использует CUDA библиотека в оптимизации сложных кодов, которые должны быть запуск на графических процессорах. Это также позволяет параллелизм на процессорах. Theano использует графические представления для символьных математических выражений. С помощью этого представления, символическое дифференцирование математических выражений поддерживается в Theano. Несколько обертки, включая Pylearn2, Keras и Lasagne, обеспечивают более легкий опыт программирования на вершине Theano.[26

Cafe является открытым исходным кодом рамки ГО алгоритмы и набор эталонных моделей. Он основан на C++, поддерживает CUDA для вычислений GPU и предоставляет интерфейсы для Python и Matlab. Cafe разделяет модель представление от его осуществления. Это было сделано возможно путем определения моделей по конфигурациям без жесткого кодирования их в исходном коде нет. Переключение между платформами (например, CPU к GPU или мобильным устройствам) легко, только изменив флаг. Сообщается, что его скорость составляет 1 мс, изображение для предсказания 4 мс, изображение для тренировки.[26

Neon: Neon3 другое открытым исходным кодом на основе ГО на Python с высокой производительностью для современных ГНС, таких как AlexNet и GoogleNet. Он поддерживает разработку нескольких обычно используемые модели, такие как СНС, РНН, ДКП и автоэнкодоры, на обоих процессоры и графические процессоры. Список расширяется по мере их реализации для частичного обучения с использованием моделей ГО. Он также поддерживает легкую смену аппаратной платформы.[26

Тут было рассмотрено несколько распространенных архитектур ГО что может служить в компоненте анализа различных вещей приложения. Большинство из этих архитектур работают с различными типами входных данных, генерируемых приложениями ИВ. Однако, чтобы получить лучшую производительность для последовательных или временных рядов данных, РНН и их варианты рекомендуются. В частности, на длительный срок зависимости между точками данных, ДКП является более благоприятным из-за к его концепции ворот для ячеек памяти. Для случаев, когда входные данные более чем одномерны, вариации СНС работать лучше. ОМБ, СГУ и вариации

автоэнкодеры работают хорошо в обработке высокомерного уменьшения размерности и иерархической извлечение признаков. В сочетании с классификационным слоем они может использоваться для различных сценариев обнаружения и прогнозирования. Более поздние архитектуры, включая вариационные автоэнкодеры, ГСС и ожидается, что лестничные сети окажут большое влияние на ИВ приложения, так как они охватывают полу-контролируемое обучение. Те более благоприятны для ИВ приложений, где огромное количество данных генерируется в то время как небольшая часть этого может быть аннотировано для машинного обучения. Роль этих архитектур можно подчеркнуть, зная, что только около 3% всех данных о вселенной к 2012 году были аннотированы и поэтому полезны для контролируемого машинного обучения.[19

Развивающееся машинное обучение архитектуры и методы, которые оба извлекают выгоду из ГО и также были рассмотрены конкретные требования к ИВ - приложениям. выделенный. Действительно, глубокое обучение с усилением может поддерживать автономность ИВ приложения, передача обучения может заполнить пробел отсутствия обучающие наборы данных, и онлайн-обучение соответствует потребности в потоковый анализ данных интернета вещей.

Также было рассмотрено несколько общих и мощных фреймворков для разработки моделей ГО. Для приложений интернета вещей, время обучения, время выполнения и динамическое обновление обученные модели являются определяющими факторами для надежного и эффективный аналитический модуль. Первый не допускает динамических обновлений модели, в то время как последняя поддерживает такие модификации. Среда глубокого обучения с открытым исходным кодом - это структура, которая следует последнему шаблону и может обрабатывать динамические изменения модели.

### 2.6 Приложения глубокого обучения

Методы ГО были показаны многообещающими с современным состоянием результаты в нескольких областях, таких как обработка сигналов, естественные обработка языка и распознавание изображений. Тенденция такова поднимаясь по вертикалям Интернета вещей. Некоторые нейросетевые модели работают лучше в специальных областях. Например, сверточные сети обеспечение лучшей производительности в приложениях, хотя автоэнкодеры очень хорошо справляются с обнаружением аномалий, данные уменьшение шума и размерности для визуализации данных. Важно установить эту связь между разновидностью нейронной сетевая модель, которая наилучшим образом соответствует каждому из различных приложений домены.[42

В этом разделе мы рассмотрим успешные применения ГО в доменах интернета вещей. Основываясь на наших наблюдениях, многие связанные с ГО приложения используют видение и классификацию изображений, как их базовая интеллектуальная служба. Есть другие услуги, такие как обнаружение позы человека, которые используются для умных домашних приложений или

интеллектуальной помощи автомобилю. Выделяем несколько видов этих услуг как основополагающие сервисы, на основе которых можно создавать другие приложения интернета вещей. То общим свойством этих услуг является то, что они должны быть обрабатываются в быстром аналитическом режиме вместо того, чтобы складывать свои данные для последующей аналитики. Действительно, каждый домен может иметь свои особенности. услуги, выходящие за рамки этих базовых услуг. Рисунок 2.15 показывает базовые сервисы и приложения Интернета вещей на вершине их.

В следующих подразделах мы сначала рассмотрим основополагающие сервисы для ИВ, которые используют глубокое обучение как их интеллект, затем выделите приложения интернета вещей и домены, где а сочетание фундаментальных и специфических услуг может быть использован.

Распознавание изображений:

Большая часть приложений интернета вещей обращается к сценариям, в которых входные данные для ГО представлены в виде видео или изображений. Вездесущие мобильные устройства, оснащенные высоким разрешением камеры, облегчают генерацию изображений и видео всеми желающими, везде. Кроме того, интеллектуальные видеокамеры используются в многие места, такие как умные дома, кампусы и производители для различных приложений. Распознавание образов, классификация и обнаружение объектов является одним из основных видов использования таких устройства. Одна проблема с системами, связанными с ИВ, которые были рассмотрены распознавание изображений - это использование определенных исходных наборов данных для оценки их эффективности. Например, входные данные для задачи транспортного средства обнаружение в смарт-подключенных автомобилях не всегда будет четкое изображение, и есть случаи, когда входное изображение находится на ночь, или в дождливую или туманную погоду. Эти случаи не являются обрабатывается с помощью доступных наборов данных и, следовательно, моделей обучение на основе этих наборов данных не является достаточно полным.

Распознавание речи, голоса:

С массовым распространением умных мобильных устройств и носимых устройств, автоматическое распознавание речи становится все более естественный и удобный способ для людей взаимодействовать с их приборы. Кроме того, небольшой размер мобильных устройств и носимые устройства в настоящее время снижают возможность прикосновения экраны и клавиатуры как средства ввода и взаимодействия с этим устройством. Однако главная забота о предоставлении функциональность распознавания речи / голоса при ограничении ресурсов приборами является его энергоемкость, особенно когда данные обрабатывается через нейронные сети. В обычной речи распознавание нейросетевой модели, голосовые данные представлены в виде необработанный вход в сеть. Данные обрабатываются через скрытые слои и вероятность передачи голосовых данных в конкретный речевой звук представлен на выходном слое.[2

Предоставление услуг с учетом местоположения, таких как навигация в помещении и маркетинг с учетом местоположения в розничных сетях становится все более распространенным в закрытых помещениях. Внутренняя локализация также может иметь приложения в других секторах Интернета вещей, например, в умных домах, умные кампусы или больницы. Входные данные, полученные из такого приложения обычно приходят из разных технологий, как зрение, связь видимого света, инфракрасный, ультразвук, WiFi, сверх широкий диапазон и Bluetooth.[26

Для подходов, основанные на Wi-Fi или Bluetooth, большинство из них литература использовала мобильные телефоны для приема сигналов от фиксированные передатчики (т. е., точки доступа или iBeacons), которые называются отпечатки пальцев. Среди этих дактилоскопических подходов, несколько попыток сообщили об использовании моделей ГО для предсказания расположение.

ГО успешно используется для определения местоположения внутренних позиций с высокой точностью. В системе, называемой DeepFi, ГО способ поверх дактилоскопии информации о состоянии канала Wi-Fi данные были использованы для идентификации позиций пользователей.[6 Эта система состоит из этапов офлайн-обучения и онлайн-локализации. На этапе автономного обучения ГО используется для обучения всех веса, основанные на ранее сохраненном состоянии канала информационные отпечатки пальцев. Другие работы, отчет с использованием вариации моделей ГО в сочетании с другими методами обучения методы извлечения объектов и оценки позиций. Эти эксперименты утверждают, что количество скрытых слоев и единиц измерения в моделях ГО имеет прямое влияние на точность локализации. В работе СНС используется для локализации внутри помещений путем слияния данные магнитного и визуального зондирования. Кроме того, СНС имеет был обучен в, чтобы определить внутренние позиции пользователей анализируя изображение из окружающей их сцены.

Лу и др. также использовали сети ДКП для локализации футбольных роботов[44 . В этом приложении данные, собранные из несколько датчиков, а именно инерционная навигационная система и зрение восприятия, анализируются, чтобы предсказать положение роботов. Авторы сообщили о повышении точности и эффективности по сравнению с двумя базовыми методами, а именно стандартными расширенями фильтрации Калмана (ЭКФ) и статической фильтрации частиц.

Рисунок 2.12– Приложения интернета вещей и базовые сервисы.

## 2.7 Будущие направления деятельности глубокого обучения

Одна примечательная часть данных интернета вещей поступает с мобильных устройств. Исследование эффективных способов использования мобильных больших данных в объединение с подходами ГО-это способ придумать лучшие сервисы для доменов интернета вещей, особенно в умных домах. В возможности моделей ГО в мобильных устройствах анализ больших данных был исследован с использованием распределенной обучающей платформы, которая выполняет итерационный MapReduce задание на нескольких параллельных искровых рабочих.[8

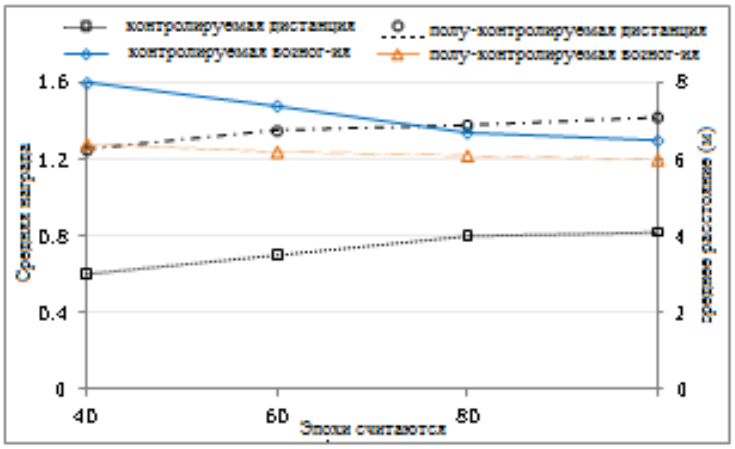



Рисунок 2.13 - Глубокое обучение с использованием только помеченных данных (под наблюдением) по сравнению с маркированными и немаркированными данными (управляемые). В каждую эпоху полу-контролируемая модель превосходит контролируемую модель как с точки зрения общего получали задачи и близость к цели

Данные интернета вещей должны быть объединены с другими источниками данных, а именно контекстной информацией, которая дополняют понимание окружающей среды. Это интеграция также может помочь быстрому анализу данных и быстрому рассуждение из-за ограниченного пространства поиска для рассуждения. Например, умная камера с возможностью лица

распознавание поз может выполнять свою работу в различных контекстах, таких как как ворота безопасности в умных домах или правительственных зданиях, или в умных автомобилях для помощи в вождении. Во всех этих ситуациях, дополнительная контекстуальная информация (например, время в пределах день, ежедневные привычки и т. д.) помогает системе рассуждать о лучшее действие, которое может быть сделано на основе обнаруженной позы человек.[1]

Поскольку мы больше полагаемся на кибер-физические системы (КФС) и ИВ в больших масштабах, то потребность в механизмах обеспечения безопасности системы от вредоносных атак, а также сбоев имеет большее решающее значение. Подходы ГО могут быть применены в следующих случаях: направления путём анализа огромного количества КФС систем Интернета вещей, чтобы выявить и предсказать слабые места системы, в которой могут произойти. Это поможет системе предотвратить или восстановить неисправностей и, следовательно, повысить уровень надежности системы КФС и ИВ.

С огромным количеством устройств интернета вещей, конфигурация и поддержание их основного физического M2M коммуникации и создание сетей становятся все сложнее. Хотя большое количество узлов сети и их связь – это вызов для традиционных подходов машинного обучения, ИТ открывает возможность для архитектур ГО доказать свою компетентность в этой области путем предоставления комплекса самостоятельных услуг такие как самонастройка, самооптимизация, самовосстановление, и балансировка собственной нагрузки. Предоставили обзор традиционных подходов машинного обучения для самоорганизующейся сотовой сети.

Беспилотные летательные аппараты: использование беспилотных летательных аппаратов транспортные средства (БПЛА) является перспективным приложением, которое может улучшить предоставление услуг в труднодоступных регионах или в критических ситуации. БПЛА также использовались для многих изображений анализ в реальном масштабе времени как задачи наблюдения, поисково-спасательные работы операции и инспекция инфраструктуры. Эти устройства сталкиваются с рядом проблем для их принятия, включая маршрутизацию, энергосбережение, избегание частных регионов, и избегание препятствий. ГО может иметь большое влияние в этой области для прогнозирования и принятия решений задачи, чтобы получить лучшее от БПЛА. Более того, БПЛА могут быть рассматривается как аналитические платформы "на лету", которые потенциально могут предоставлять временно туманные вычислительные аналитические услуги, а также как распределенная аналитика.[26

## 3 Техническая часть

### 3.1 Язык программирования - Python

Для начала немножко об истории языка. В конце 80-х годов прошлого века сотрудник голландского центра математики и информатики Гвидо ван Россум решил создать свой собственный язык. Основной целью он ставил создать язык простой и выразительный, на котором было бы просто писать код. В 1991 году он опубликовал исходники языка, который получил название Python. Не в честь известного всем вида пресмыкающихся, а в честь популярного телешоу 70-х годов прошлого века Летающий цирк Монти Python. Однако это не помешало змее стать маскотом, символом языка, а также присутствовать на логотипах языка и связанных с ним проектах. Сейчас, спустя много лет, мы видим, что у Гвидо получилось. У него получилось создать интерпретируемый язык с динамической типизацией и автоматической сборкой мусора, на котором действительно приятно писать код. Читая код Python, мы практически читаем книгу на английском языке. Конечно это преувеличение, однако Python действительно выделяется в этом отношении. Также за годы существования языка вокруг него сложилось огромное сообщество, и была написана масса готовых библиотек на все случаи жизни. Теперь еще один нюанс, который важно понимать. Так получилось, что разработчики языка, которых уже много, а не только Гвидо, выпустили третью версию, обратно несовместимую со второй. Это было сделано специально для того, чтобы решить некоторые архитектурные недостатки второй версии языка. Это получилось сделать, однако обратная несовместимость привела к тому, что до сих пор многие продакшн-системы используют Python версии 2. Но есть четко установленный дедлайн — 2020 год, когда официальная поддержка Python 2 прекратится, поэтому новые проекты стоит начинать именно на Python'е 3. Что также стоит отметить, Python — это название спецификации языка, основная его реализация написана на языке C, называется CPython. Есть и другие реализации спецификации языка Python такие, как IronPython для.NET, либо PyPy, который добавляет JIT-компиляцию коду.[18 Однако в дипломной работе, именно реализация на – Jupyter Python.

Первое на что посмотрим — это интерактивный интерпретатор Python. Чтобы запустить интерактивный интерпретатор, нужно в терминале набрать команду Jupyter notebook[13 . Интерактивный интерпретатор, по большому счету, — это программа, которая считывает ввод пользователя, интерпретирует его и выдает на экран результат. Интерпретатор — это то, что делает Python таким доступным. В любой момент мы можем открыть интерактивную оболочку и проверить какую-то свою гипотезу, поэкспериментировать или просто воспользоваться ей как калькулятором. В реальных проектах код на Python пишут конечно же не в интерактивном интерпретаторе, а в файлах. А в больших проектах — в большом количестве файлов. В Python имена переменных могут содержать буквы, цифры и символ нижнего подчеркивания.

При этом начинаться переменная должна либо с буквы, либо с символа нижнего подчеркивания.[12

### 3.2 Библиотеки Python

Pandas — программная библиотека на языке Python для обработки и анализа данных. Работа pandas с данными строится поверх библиотеки NumPy, являющейся инструментом более низкого уровня. Предоставляет специальные структуры данных и операции для манипулирования числовыми таблицами и временными рядами. Название библиотеки происходит от эконометрического термина «панельные данные», используемого для описания многомерных структурированных наборов информации.[16

Основная область применения - обеспечение работы в рамках среды Python не только для сбора и очистки данных, но для задач анализа и моделирования данных.

Основные возможности библиотеки:

- объект DataFrame для манипулирования индексированными массивами двумерных данных;
- инструменты для обмена данными между структурами в памяти и файлами различных форматов;
- встроенные средства совмещения данных и способы обработки отсутствующей информации;
- переформатирование наборов данных, в том числе создание сводных таблиц;
- срез данных по значениям индекса, расширенные возможности индексирования, выборка из больших наборов данных;
- вставка и удаление столбцов данных;
- возможности группировки позволяют выполнять трёхэтапные операции типа «разделение, изменение, объединение» (англ. split-apply-combine);
- слияние и объединение наборов данных;
- иерархическое индексирование позволяет работать с данными высокой размерности в структурах меньшей размерности;
- работа с временными рядами: формирование временных периодов и изменение интервалов и так далее;

NumPy - библиотека с открытым исходным кодом для языка программирования Python. Осуществляет поддержки многомерных массивов (включая матрицы), высокоуровневых математических функций, предназначенных для работы с многомерными массивами.[15

Математические алгоритмы, реализованные на интерпретируемых языках (например, Python), часто работают гораздо медленнее тех же алгоритмов, реализованных на компилируемых языках (например, Фортран, Си, Java). Библиотека NumPy предоставляет реализации вычислительных алгоритмов (в виде функций и операторов), оптимизированные для работы с многомерными массивами. В результате любой алгоритм, который может быть выражен в виде последовательности операций над массивами (матрицами) и

реализованный с использованием NumPy, работает так же быстро, как эквивалентный код, выполняемый в MATLAB.

Matplotlib — библиотека на языке программирования Python для визуализации данных двумерной (2D) графикой (3D графика также поддерживается). Получаемые изображения могут быть использованы в качестве иллюстраций в публикациях.[28

Matplotlib является гибким, легко конфигурируемым пакетом, который вместе с NumPy, SciPy и IPython предоставляет возможности, подобные MATLAB. В настоящее время пакет работает с несколькими графическими библиотеками, включая wxWindows и PyGTK.

Пакет поддерживает многие виды графиков и диаграмм:

- графики (line plot);
- диаграммы разброса (scatter plot);
- столбчатые диаграммы (bar chart) и гистограммы (histogram);
- круговые диаграммы (pie chart);
- ствол-лист диаграммы (stem plot);
- контурные графики (contour plot);
- поля градиентов (quiver);
- спектральные диаграммы (spectrogram).

Перечисленные библиотеки были применены в данной дипломной работе.

### 3.3 Прогнозирование временных рядов

Временным рядом называется последовательность $y1, \ldots, yT, \ldots, \in R$ (потенциально, впрочем, бесконечное) значений признака y, измеряемого через постоянные временные интервалы. Признак y мы будем считать вещественным. Будем работать исключительно с вещественными временными рядами. Важно в этом определении – требование постоянности временного интервала, в которое происходит измерение. Ваш признак вы, может быть, измеряете каждый день, или каждый месяц, или каждый год. Важно, что у нас есть данные именно за последовательные точки с одинаковым временным расстоянием между ними. Временными являются, например, ряды с средними дневными ценами акций какой-нибудь компании, или среднемесячным уровнем безработицы, измеренным за несколько лет, или среднегодовым объемом производства автомобилей за несколько десятилетий. Как правило, временные ряды интересно прогнозировать, то есть зная значение вашего признака в прошлом, предсказывать, какими они будут в будущем. Формально, это задача ставится, как задача поиска функции $fT$ такой, что значение ряда в момент времени $T + d$ приближается значением этой функции $f$ от известной на момент времени $T$ истории и параметра $d$. Параметр $d$ определяет, насколько далеко вперед вы хотите предсказывать значение вашего ряда. Как правило, меняется от 1 до некоторой величины $D$, которая называется горизонтом прогнозирования, $d$, в свою очередь, отсрочкой прогноза. Практически всегда на протяжении всей нашей специализации до этого, когда мы анализировали данные, мы считали, что имеем дело с простыми выборками. То есть с

независимыми одинаково распределенными наблюдениями. Разные наблюдения друг от друга никак не зависели. Задача анализа временных рядов — всё ровно наоборот. Мы надеемся, что данные по прошлому, которые у нас есть, как-то связаны со значениями признаков в будущем. И чем сильнее они связаны, тем больше у нас есть информации о том, как будет себя вести наш признак в будущем, тем лучше мы можем его в будущем предсказывать. Посмотрим на данные о трафики в городе Милан. Уже на этом графике, на котором по горизонтальной оси отложено время, а по вертикальной — значение нашего признака, видно, что это ни в коем случае не простая выборка. Измерения здесь не являются независимыми и одинаково распределенными. То, что мы наблюдаем здесь, это разворачивающийся во времени сложно структурированный процесс, и выявив эту структуру, мы сможем учесть её в нашей прогнозирующей модели, и таким образом построить действительно хороший и точный прогноз. Лучше всего мы умеем решать задачу обучения с учителем, поэтому, конечно, хочется сразу задачу прогнозирования временного ряда к ней свести, поскольку этот процесс, разворачивающийся во времени, логично сгенерировать какие-то признаки, связанные со временем, и попробовать сделать какую-то, например, обычную регрессию нашего у на вот такие временные признаки. Ну можно взять, например, линейную функцию во времени, или квадратичную, и каким-то образом подобрать её параметры так, чтобы она лучше всего описывала данные. Это решение слишком простое, чтобы быть хорошим. Действительно, если мы посмотрим на остатки такой регрессии, мы увидим, что эти остатки далеко не похожи на случайный шум. В них остается еще большое количество структуры, которую мы в нашей регрессионной модели не учли. Чем больше особенностей данных, чем больше структуры в нашей модели учитывается, тем лучше предсказания, которые такая модель дает. Поэтому, когда мы смотрим на эти остатки, у нас возникает надежда, что мы можем построить какую-то более сложную модель, которая будет лучше описывать имеющиеся у нас данные и заодно давать более точные прогнозы будущего. Для начала введем неформально несколько компонент, которыми очень часто можно описать поведение временных рядов. Первый из них называется трендом. Тренд — это плавное долгосрочное изменение уровня ряда, то есть это какое-то изменение среднего уровня, которое мы можем охарактеризовать, если мы достаточно долго ряд наблюдаем. Еще одна компонента — это сезонность. Сезонностью называется циклические изменения уровня ряда с постоянным фиксированным периодом. Еще во временных рядах бывают циклы. Циклом называется изменение уровня ряда с переменным периодом. Значение ряда может определяться как этими тремя компонентами, так и большим количеством других признаков. Некоторые из этих признаков можно явно учесть, другие могут быть слишком слабыми или их может быть совершенно невозможно измерить, и в таком случае, просто удобно считать, что данные содержат какой-то случайный шум. Вот эту последнюю, случайную компоненту ряда, которую спрогнозировать

невозможно, мы будем называть ошибкой связанные с циклическими изменениями среднего уровня экономической активности.[18

Интерес представляет задача прогнозирования временных рядов. Подразумевается, что зная значение признака в прошлом, можно предсказать его в будущем. Формально задача ставится как поиск функции $f$T:

$$yT + d _ fT\ (yT, \dots y1, d) _ \hat{}yT + djT, \tag{3.1}$$

где d 2 {f1,…, dj} — отсрочка прогноза.

D — горизонт прогнозирования.

До этого, на протяжении практически всей специализации, считалось, что анализируемые данные — это простые выборки, то есть независимые одинаково распределённые наблюдения. В задаче анализа временных рядов всё с точностью наоборот: предполагается, что данные в прошлом каким-то образом связаны с данными в будущем. Чем сильнее они связаны, тем больше имеется информации о поведении временного ряда в будущем и тем точнее можно сделать прогноз.

Таблица 3.1 - Данные с датасета Милана

| | ts | internet |
|---|---|---|
| 0 | 2013-12-26 23:00:00 | 0.000000 |
| 1 | 2013-12-26 23:00:00 | 10.251357 |
| 2 | 2013-12-26 23:00:00 | 0.027300 |
| 3 | 2013-12-26 23:00:00 | 0.000000 |
| 4 | 2013-12-26 23:00:00 | 10.414706 |

| | square_id | ts | country_code | sms_in_activity | sms_out_activity | call_in_activity | call_out_activity | internet_trafic_activity |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1383260400000 | 0 | 0.081363 | NaN | NaN | NaN | NaN |
| 1 | 1 | 1383260400000 | 39 | 0.141864 | 0.156787 | 0.160938 | 0.052275 | 11.028366 |
| 2 | 1 | 1383261000000 | 0 | 0.136588 | NaN | NaN | 0.027300 | NaN |
| 3 | 1 | 1383261000000 | 33 | NaN | NaN | NaN | NaN | 0.026137 |
| 4 | 1 | 1383261000000 | 39 | 0.278452 | 0.119926 | 0.188777 | 0.133637 | 11.100963 |

| | square_id | ts | country_code | sms_in_activity | sms_out_activity | call_in_activity | call_out_activity | internet_trafic_activity |
|---|---|---|---|---|---|---|---|---|
| count | 1000000.000000 | 1.000000e+06 | 1000000.000000 | 634239.000000 | 370347.000000 | 330285.000000 | 469486.000000 | 511348.000000 |
| mean | 1926.700761 | 1.383307e+12 | 246.255329 | 0.594881 | 0.621558 | 0.654855 | 0.570285 | 14.163467 |
| std | 830.127613 | 2.295022e+07 | 3762.798490 | 1.025349 | 1.048773 | 1.106221 | 1.116809 | 22.102522 |
| min | 1.000000 | 1.383260e+12 | 0.000000 | 0.000004 | 0.000004 | 0.000004 | 0.000004 | 0.000004 |
| 25% | 1393.000000 | 1.383290e+12 | 0.000000 | 0.081152 | 0.086037 | 0.078920 | 0.040789 | 0.096252 |
| 50% | 1991.000000 | 1.383308e+12 | 39.000000 | 0.241488 | 0.263960 | 0.262351 | 0.162682 | 5.970990 |
| 75% | 2599.000000 | 1.383326e+12 | 41.000000 | 0.669338 | 0.713366 | 0.733681 | 0.583188 | 18.728128 |
| max | 10000.000000 | 1.383346e+12 | 88239.000000 | 57.070936 | 45.021426 | 22.679426 | 24.217149 | 573.286023 |

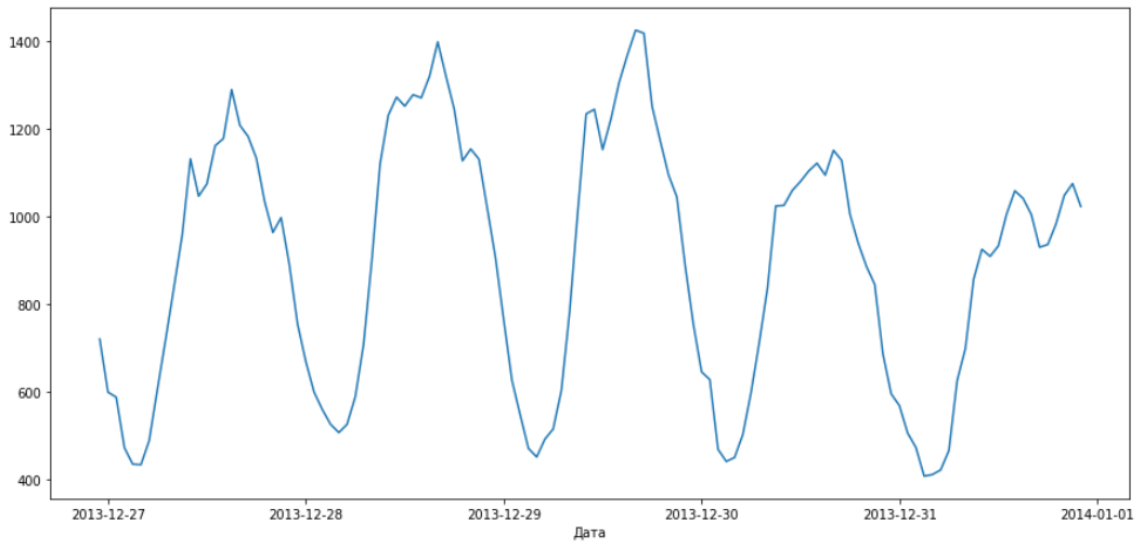



Рисунок 3.1 – Среднегодовой трафик в городе Милан

Видно, что на графике изображена не простая выборка (измерения не являются независимыми и одинаково распределёнными), а сложный, структурированный процесс.

Выявив структуру этого процесса, можно учесть её в прогнозирующей модели и построить действительно точный прогноз.

### 3.4 Компоненты временных рядов

Тренд - плавное долгосрочное изменение уровня ряда. Эту характеристику можно получить, наблюдая ряд в течение достаточно долгого времени.

Сезонность - циклические изменения уровня ряда с постоянным периодом.

Цикл - изменение уровня ряда с переменным периодом.

Ошибка - непрогнозируемая случайная компонента ряда. Сюда включены все те характеристики временного ряда, которые сложно измерить (например, слишком слабые).[9

### 3.5 Автокорреляция временного ряда

3.5.1 Пример автокорреляции трафика города Милан. Одной из важнейших характеристик временного ряда является автокорреляция. Далее суть этой характеристики будет демонстрироваться на примере данных трафика Милана.

Количественной характеристикой сходства между значениями ряда в соседних точках является автокорреляционная функция (или просто автокорреляция), которая задаётся следующим соотношением [18

$$r\tau = E((yt - Ey)(yt + \tau - Ey)), Dy. \quad (3.2)$$

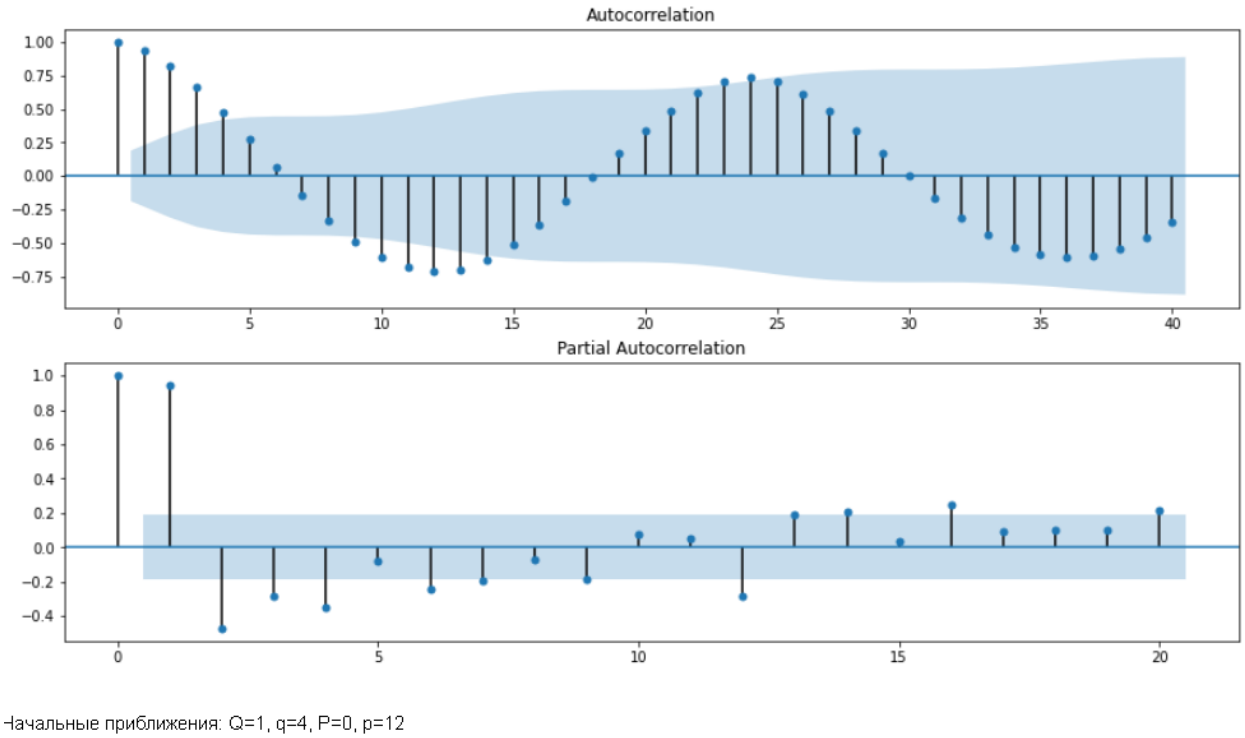



Рисунок 3.2 – Автокорреляционная функция ряда

### 3.6 Стационарность временного ряда

3.6.1 Понятие стационарности временного ряда. Ещё одно важное свойство временных рядов — это стационарность. Временной ряд $y1, \ldots, yT$ называется стационарным, если s (ширина окна) распределение $yt, \ldots, yt + s$ не зависит от $t$, т.е. его свойства не зависят от времени.[18

Из этого определения следует, что ряды, в которых присутствует тренд, являются нестационарными: в зависимости от расположения окна изменяется средний уровень ряда. Кроме того, нестационарные ряды с сезонностью: если ширина окна меньше сезонного периода, то распределение ряда будет разным, в зависимости от положения окна. При этом интересно, что ряды, в которых есть непериодические циклы, не обязательно являются нестационарными, поскольку нельзя заранее предсказать положение максимумов и минимумов этого ряда.

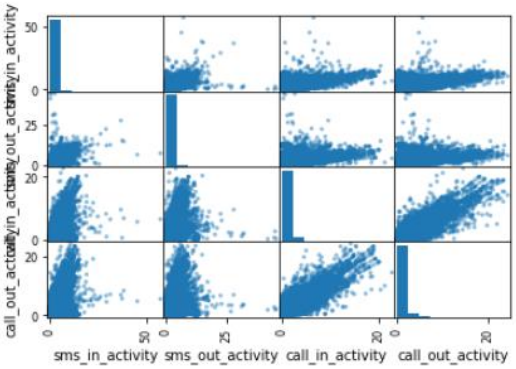
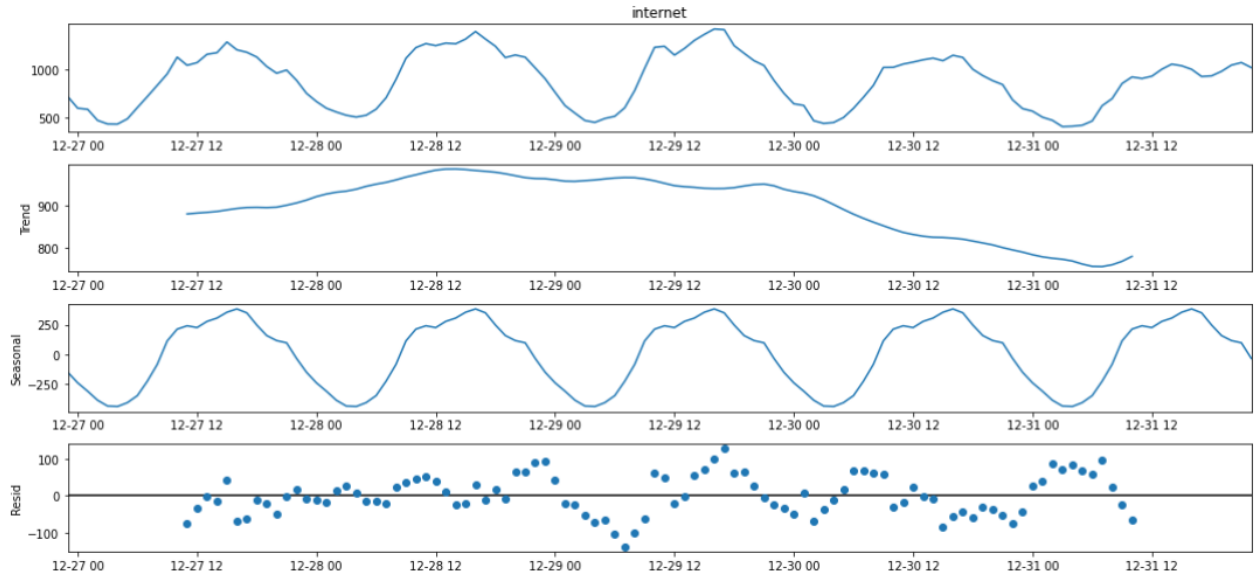



Рисунок 3.3 Матрица рассеяния




Критерий Дики-Фуллера: p=0.030796

Рисунок 3.4 – Компоненты временного ряда

3.6.2 Критерий Дики-Фуллера. Формально гипотезу о стационарности можно проверить с помощью критерия Дики-Фуллера. Статистика данного критерия выглядит достаточно сложно и пока что рассматриваться не будет.

Вообще говоря, существует большое количество критериев для проверки гипотезы о стационарности, на практике можно использовать любой из них. Однако в этой дипломной работе внимание акцентируется на критерии Дики-Фуллера, потому что для него существует реализация в языке Python.[2]

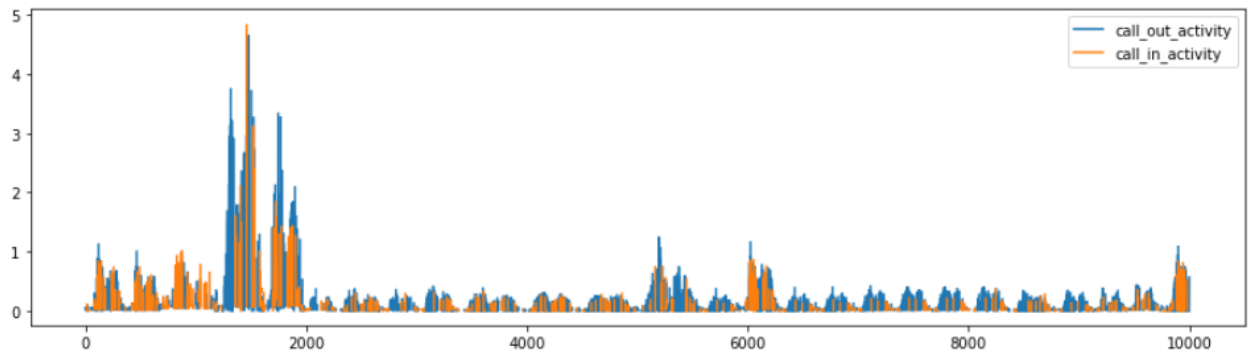



Рисунок 3.5 – Динамический график входящих и исходящих звонков

### 3.7 ARIMA

ARIMA (англ. Autoregressive integrated moving average) — интегрирована модель авторегрессии — скользящего среднего — модель и методология анализа временных рядов. Является расширением моделей ARMA для нестационарных временных рядов, которые можно сделать стационарными взятием разностей некоторого порядка от исходного

временного ряда (так называемые интегрированные или разно стационарные временные ряды).[17

Модели типа ARIMA — это обобщение модели класса ARMA.

На данный момент известны два факта:

- теорема вольда: любой стационарный ряд может быть описан моделью arma $(p, q)$ с любой наперёд заданной точностью;

- при помощи дифференцирования нестационарный ряд можно сделать стационарным.

Эти две идеи и лежат в основе моделей класса ARIMA. Модель $\mathrm{ARIMA}(p, d, q)$ — это модель $\mathrm{ARMA}(p, q)$ для $d$ раз продифференцированного ряда.

На рисунке 3.3 было показано, что ряд не стационарен (и это видно невооружённым глазом), но зато стационарен ряд его первых разностей. Из этого следует, что для ряда разностей можно подобрать достаточно хорошую модель в классе ARMA. Если сделать это, а затем произвести операцию, обратную дифференцированию, то в результате будет получена модель ARIMA для исходного ряда.

В этой модели происходит одно дифференцирование и не используется ни одной компоненты авторегрессии и скользящего среднего, и это немного странно. Ряд разностей моделируется константой, но после проведения операции, обратной дифференцированию, полученный результат не является константой. В этой модели много странностей, но результат в любом случае лучше, чем то, что можно было бы получить с помощью регрессии ряда на временные признаки.

Итоговый алгоритм подбора модели в классе ARIMA состоит в следующем

В первую очередь необходимо построить график ряда и посмотреть на него. Уже из визуального анализа можно сделать определённые выводы: есть ли в данных сезонность, какой сезонный период, есть ли в ряде пропуски и выбросы, необходимо ли стабилизировать дисперсию, стоит ли исключить из рассмотрения начало ряда, потому что значения в начале совсем не похожи на значения в конце.

Следующий шаг — это стабилизация дисперсии при необходимости. Стабилизация производится с помощью метода Бокса-Кокса или логарифмированием, что является частным случаем того же метода.

Если исследуемый ряд не стационарен, необходимо подобрать порядок дифференцирования, при котором он становится стационарным. Таким образом фиксируются параметры d, D модели ARIMA.

Далее необходимо построить графики автокорреляционной функции (ACF) и частичной автокорреляционной функции (PACF) и из этих графиков определить примерные значения параметров p, q, P, Q. Фактически эти значения — начальные приближения, с которых начинается перебор разных моделей.

Полученные модели необходимо обучить, сравнить их по информационному критерию Акаике и выбрать ту, которая его минимизирует.

Необходимо посмотреть на остатки получившейся модели, чтобы понять, насколько хорошей она получилась, можно ли, теоретически, её улучшить, нет ли в ней каких-то видимых недостатков. Подробнее об анализе остатков будет рассказано позднее [17 .

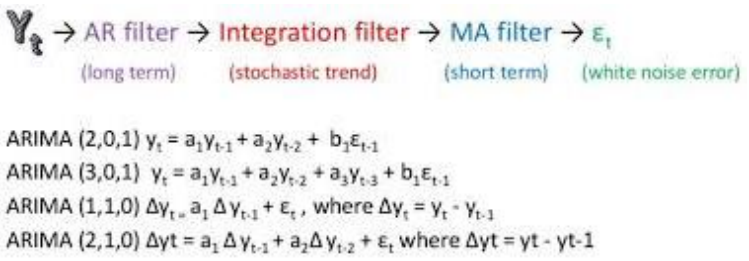



Рисунок 3.6 – Параметры ARIMA

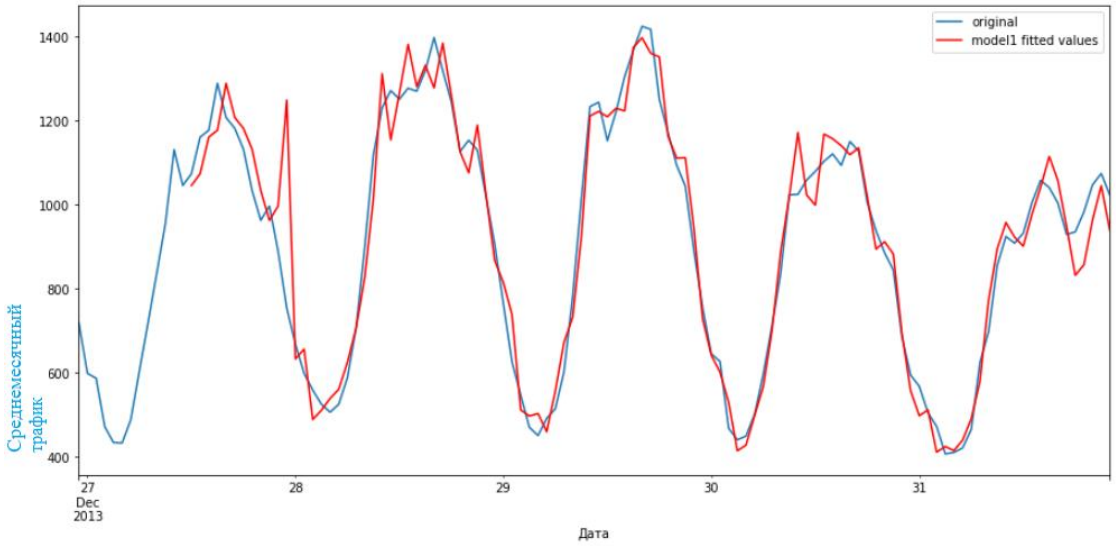



Рисунок 3.7 – Модель ARIMA применённая к ряду значений индекса Доу-Джонса

### 3.8 Прогнозирование ARIMA

Теперь необходимо разобраться, как на основании настроенной модели ARIMA правильно строить прогноз. Пусть модель построена, определены значения всех неизвестных параметров, получены их оценки $\hat{\alpha}, \hat{\varphi}, \hat{\theta}$, которые записаны в этом уравнении

$$Y^t = \alpha + \varphi^{1yt} - 1 + \ldots + \varphi^{pyt} - p + \varepsilon t + \theta^{1\varepsilon t} - 1 + \ldots + \theta^{q\varepsilon t} - q. \qquad (3.8)$$

Чтобы построить прогноз на момент времени T + 1, нужно в этом уравнении заменить все индексы t на T + 1

$$y^T + \frac{1}{T} = \alpha + \varphi^{1yT} + \ldots + \varphi^{pyT} + 1\text{-}p + \mathcal{E}T + 1 + \theta^{1\varepsilon t} + \theta^{q\varepsilon t} + 1\text{-}q. \quad (3.9)$$

В этом уравнении присутствует значение ошибки из будущего $\varepsilon_{T+1}$. Неизвестно, какой будет наблюдаться шум в будущем, однако можно предполагать, что в среднем он будет равен 0. Поэтому значения будущих ошибок можно безболезненно заменить на 0. Фактически из уравнения просто удаляются все члены, которые связаны с ошибками из будущего

$$y^T + \frac{1}{T} = \alpha + \varphi^{1yT} + \ldots + \varphi^{pyT} + 1\text{-}p + \mathcal{E}T + 1 + \theta^{1\varepsilon t} + \theta^{q\varepsilon t} + 1\text{-}q. \quad (3.10)$$

В уравнении также присутствуют ошибки из прошлого. Их необходимо заменить на остатки модели в этих точках, потому они являются самыми лучшими оценками ошибок из имеющихся

$$y^T + \frac{1}{T} = \alpha + \varphi^{1yT} + \ldots + \varphi^{pyT} + 1\text{-}p + \mathcal{E}T + 1 + \theta^{1\varepsilon t} + \theta^{q\varepsilon t} + 1\text{-}q. \quad (3.11)$$

Если прогноз необходимо построить не на одну точку вперёд, а, например, на две, то в формуле появляется значение ряда из будущего $yT + 1$

$$y^T + \frac{2}{T} = \alpha + \varphi^{1yT} + 1 + \ldots + \varphi^{pyT} + 2\text{-}p + \theta^{1\mathcal{E}T} + 1 + \theta^{1\varepsilon t} + \theta^{q\varepsilon t} + 2\text{-}q. \quad (3.12)$$

Его необходимо заменить на прогноз $\hat{y}T + 1|T$.

Использовалась модель ARIMA и преобразование Бокса-Кокса. Полученный прогноз не выглядит тривиальным: он не является константой, не ведёт себя странно, не падает до нуля. Визуально этот прогноз можно одобрить.[2]

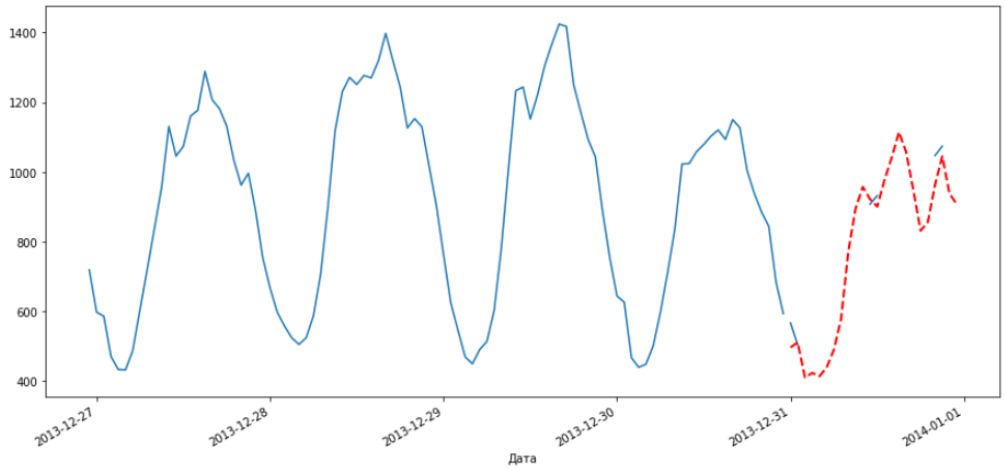



Рисунок 3.8 – Прогнозирование временного ряда

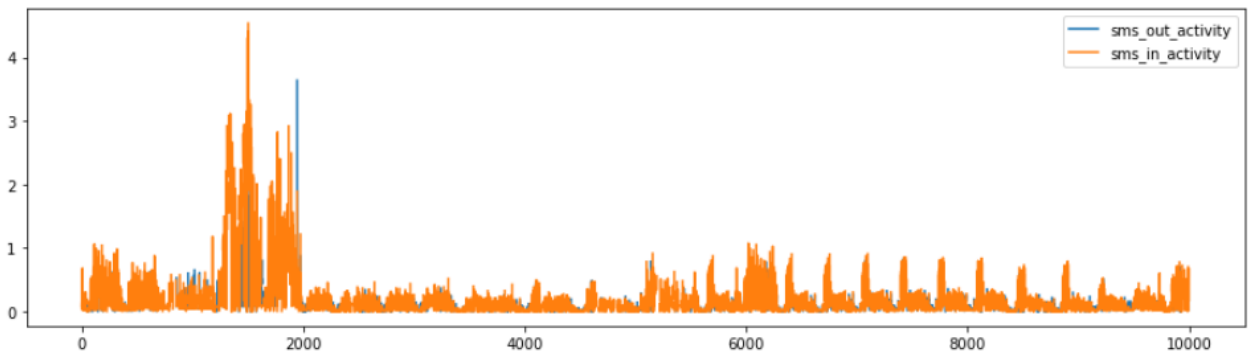



Рисунок 3.9 Динамический график входящих и исходящих сообщений

Почти повсеместное внедрение мобильных телефонов и экспоненциальный рост использования интернет-услуг порождают огромное количество данных, которые можно использовать для предоставления новых фундаментальных и количественных данных о социально-технических системах. Подробные записи о вызовах (CDR) 6,8 миллиардов абонентов мобильных телефонов по всему миру, потенциально представляют самый бесценный прокси для общения людей и их мобильности в глобальном масштабе. Доступность этих данных действительно определяет новую область исследований, которая использует CDR для извлечения моделей мобильности человека и социальных взаимодействий. Оценка плотности населения, модели городов структуры, предсказывает социально-экономические показатели и результаты территорий, и модель распространения болезней. Кроме того, появление новых гео-локализованных информационных и коммуникационных технологий (ИКТ), таких как Twitter и Foursquare, предоставляет дополнительные возможности для количественного анализа различных аспектов человеческого поведения, таких как социальное благополучие отдельных лиц и сообществ социально-экономический статус географических регионов и мобильность людей. Еще более перспективным является исследование наборов данных, объединяющих данные социальных сетей с CDR и другими экономическими и демографическими показателями.

Существует много типов CDR, и Telecom Italia зарегистрировала следующие виды деятельности:

- полученное SMS CDR генерируется каждый раз, когда пользователь получает SMS;
- отправленные SMS CDR генерируется каждый раз, когда пользователь отправляет SMS;
- входящий вызов CDR генерируется каждый раз, когда пользователь получает вызов;
- исходящий вызов CDR генерируется каждый раз, когда пользователь выполняет вызов;
- интернет CDR генерируется каждый раз, когда пользователь запускает подключение к Интернету или завершает подключение к Интернету. Во время

того же соединения создается CDR, если соединение длится более 15 минут или пользователь перевел более 5 МБ.

Этот набор данных служит мерой уровня взаимодействия между пользователями и сетью мобильной связи.

Идентификатор квадрата: идентификационная строка данного квадрата Милана / Trentino GRID;

Интервал времени: начальный интервал времени, выраженный в миллисекундах. Время окончания интервала может быть получено путем добавления 600 000 миллисекунд (10 мин) к этому значению;

SMS-активность: активность, пропорциональная количеству полученных SMS-сообщений внутри данного идентификатора квадрата и в течение определенного интервала времени. SMS-сообщения отправляются из страны, указанной кодом страны;

Активность SMS-выхода: активность, пропорциональная количеству отправленных SMS-сообщений внутри данного идентификатора квадрата в течение заданного интервала времени. SMS-сообщения принимаются в стране, указанной кодом страны;

Активность вызова: активность, пропорциональная количеству принятых вызовов внутри идентификатора Square в течение заданного интервала времени. Звонки поступают из страны, указанной кодом страны;

Активность вызова: активность, пропорциональная количеству исходящих вызовов внутри данного идентификатора Square в течение заданного интервала времени. Звонки принимаются в стране, указанной кодом стран ;

Активность интернет-трафика: количество CDR, созданных внутри заданного идентификатора Square в течение заданного интервала времени. Интернет-трафик инициируется из страны, обозначенной кодом страны;

Код страны: телефонный код страны.

## 4 Безопасность жизнедеятельности

Данный раздел дипломного проекта посвящен рассмотрению следующих вопросов: анализ условий труда для комнаты программиста, описать эргономику рабочего места и расчет кондиционирования:

а) Анализ условий труда для комнаты программиста. Научно-технический прогресс внес серьезные изменения в условия производственной деятельности работников умственного труда. Их труд стал более интенсивным, напряженным, требующим значительных затрат умственной, эмоциональной и физической энергии. Это потребовало комплексного решения проблем эргономики, гигиены и организации труда, регламентации режимов труда и отдыха.

В настоящее время компьютерная техника широко применяется во всех областях деятельности человека. При работе с компьютером человек подвергается воздействию ряда опасных и вредных производственных

факторов: электромагнитных полей (диапазон радиочастот: ВЧ, УВЧ и СВЧ), инфракрасного и ионизирующего излучений, шума и вибрации, статического электричества и др.

Работа с компьютером характеризуется значительным умственным напряжением и нервно-эмоциональной нагрузкой операторов, высокой напряженностью зрительной работы и достаточно большой нагрузкой на мышцы рук при работе с клавиатурой ЭВМ. Большое значение имеет рациональная конструкция и расположение элементов рабочего места, что важно для поддержания оптимальной рабочей позы человека-оператора.

В процессе работы с компьютером необходимо соблюдать правильный режим труда и отдыха. В противном случае у персонала отмечаются значительное напряжение зрительного аппарата с появлением жалоб на неудовлетворенность работой, головные боли, раздражительность, нарушение сна, усталость и болезненные ощущения в глазах, в пояснице, в области шеи и руках.

б) Эргономические требования к рабочему месту. Проектирование рабочих мест, снабженных видеотерминалами, относится к числу важных проблем эргономического проектирования в области вычислительной техники.

Рабочее место и взаимное расположение всех его элементов должно соответствовать антропометрическим, физическим и психологическим требованиям. Большое значение имеет также характер работы. В частности, при организации рабочего места программиста должны быть соблюдены следующие основные условия: оптимальное размещение оборудования, входящего в состав рабочего места и достаточное рабочее пространство, позволяющее осуществлять все необходимые движения и перемещения.

Эргономическими аспектами проектирования видеотерминальных рабочих мест, в частности, являются: высота рабочей поверхности, размеры пространства для ног, требования к расположению документов на рабочем месте (наличие и размеры подставки для документов, возможность различного размещения документов, расстояние от глаз пользователя до экрана, документа, клавиатуры и т.д.), характеристики рабочего кресла, требования к поверхности рабочего стола, регулируемость элементов рабочего места.

Главными элементами рабочего места программиста являются стол и кресло. Основным рабочим положением является положение сидя.

Рабочая поза сидя вызывает минимальное утомление программиста. Рациональная планировка рабочего места предусматривает четкий порядок и постоянство размещения предметов, средств труда и документации. То, что требуется для выполнения работ чаще, расположено в зоне легкой досягаемости рабочего пространства.

Моторное поле - пространство рабочего места, в котором могут осуществляться двигательные действия человека.

Максимальная зона досягаемости рук - это часть моторного поля рабочего места, ограниченного дугами, описываемыми максимально вытянутыми руками при движении их в плечевом суставе.

Оптимальная зона - часть моторного поля рабочего места, ограниченного дугами, описываемыми предплечьями при движении в локтевых суставах с опорой в точке локтя и с относительно неподвижным плечом.

Для комфортной работы стол должен удовлетворять следующим условиям [5]:

- высота стола должна быть выбрана с учетом возможности сидеть свободно, в удобной позе, при необходимости опираясь на подлокотники;
- нижняя часть стола должна быть сконструирована так, чтобы программист мог удобно сидеть, не был вынужден поджимать ноги;
- поверхность стола должна обладать свойствами, исключающими появление бликов в поле зрения программиста;
- конструкция стола должна предусматривать наличие выдвижных ящиков (не менее 3 для хранения документации, листингов, канцелярских принадлежностей);
- высота рабочей поверхности рекомендуется в пределах 680-760мм. Высота поверхности, на которую устанавливается клавиатура, должна быть около 650мм.

Большое значение придается характеристикам рабочего кресла. Так, рекомендуемая высота сиденья над уровнем пола находится в пределах 420-550мм. Поверхность сиденья мягкая, передний край закругленный, а угол наклона спинки - регулируемый.

Необходимо предусматривать при проектировании возможность различного размещения документов: сбоку от видеотерминала, между монитором и клавиатурой и т.п. Кроме того, в случаях, когда видеотерминал имеет низкое качество изображения, например заметны мелькания, расстояние от глаз до экрана делают больше (около 700мм), чем расстояние от глаза до документа (300-450мм). Вообще при высоком качестве изображения на видеотерминале расстояние от глаз пользователя до экрана, документа и клавиатуры может быть равным.

Положение экрана определяется:

расстоянием считывания (0,6…0,7м);

- углом считывания, направлением взгляда на 20° ниже горизонтали к центру экрана, причем экран перпендикулярен этому направлению.

Должна также предусматриваться возможность регулирования экрана:

- по высоте +3 см;
- по наклону от -10° до +20° относительно вертикали в левом и правом направлениях.

Большое значение также придается правильной рабочей позе пользователя. При неудобной рабочей позе могут появиться боли в мышцах, суставах и сухожилиях. Требования к рабочей позе пользователя видеотерминала следующие:

- голова не должна быть наклонена более чем на 20°,
- плечи должны быть расслаблены,
- локти - под углом 80°…100°,

- предплечья и кисти рук - в горизонтальном положении.

Причина неправильной позы пользователей обусловлена следующими факторами: нет хорошей подставки для документов, клавиатура находится слишком высоко, а документы - низко, некуда положить руки и кисти, недостаточно пространство для ног.

В целях преодоления указанных недостатков даются общие рекомендации: лучше передвижная клавиатура; должны быть предусмотрены специальные приспособления для регулирования высоты стола, клавиатуры и экрана, а также подставка для рук.

Существенное значение для производительной и качественной работы на компьютере имеют размеры знаков, плотность их размещения, контраст и соотношение яркостей символов и фона экрана. Если расстояние от глаз оператора до экрана дисплея составляет 60…80 см, то высота знака должна быть не менее 3мм, оптимальное соотношение ширины и высоты знака составляет 3:4, а расстояние между знаками – 15…20% их высоты. Соотношение яркости фона экрана и символов - от 1:2 до 1:15.

Во время пользования компьютером медики советуют устанавливать монитор на расстоянии 50-60 см от глаз. Специалисты также считают, что верхняя часть видеодисплея должна быть на уровне глаз или чуть ниже. Когда человек смотрит прямо перед собой, его глаза открываются шире, чем когда он смотрит вниз. За счет этого площадь обзора значительно увеличивается, вызывая обезвоживание глаз. К тому же если экран установлен высоко, а глаза широко открыты, нарушается функция моргания. Это значит, что глаза не закрываются полностью, не омываются слезной жидкостью, не получают достаточного увлажнения, что приводит к их быстрой утомляемости.

Создание благоприятных условий труда и правильное эстетическое оформление рабочих мест на производстве имеет большое значение как для облегчения труда, так и для повышения его привлекательности, положительно влияющей на производительность труда.

Здесь мы рассматриваем помещение программиста. В помещении имеется 1 рабочее место, которое оснащено компьютером. Площадь всего помещения 18 м$^2$, высота потолков Н=2,5 м. В помещении имеется 2 окна. Схема помещения приведена на рисунке 4.1

В помещении находится 1 рабочее места, оснащенное одним ЭВМ. В соответствие с СанПиН 2.2.2. 542-96 для работы с ЭВМ, площадь помещения на одно рабочее места с компьютером должна составлять не менее 6 м$^2$, объем - не менее 19,5 м$^3$ с учетом максимального числа одновременно работающих [34

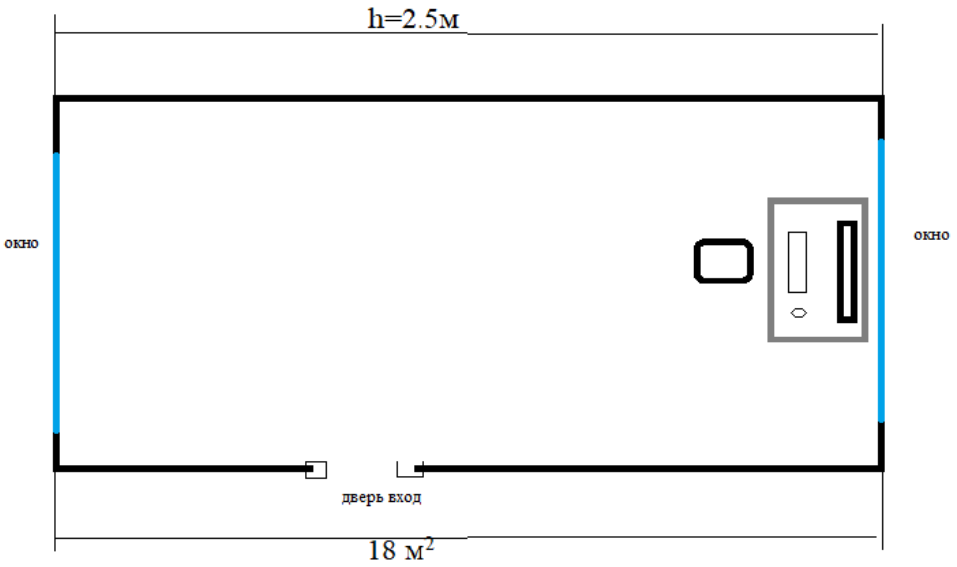



Рисунок 4.1 – Схема помещения

Таким образом, площадь S помещения составляет 18 м$^2$, а объем составляет 45 м$^3$. Что соответствует всем нормам.

Основной целью систем кондиционирования и вентиляции в помещении является создание оптимального микроклимата для обеспечения правильной работы компьютеров и положительного условия пребывания там людей.

Система вентиляции и кондиционирования помещений такого типа должна выполнять следующие задачи:

- поддерживать определенную температуру, влажность и пыль;
- быстрый и точный контроль микроклимата;
- стабильность и непрерывность соблюдения определенных микроклиматических параметров круглосуточно;
- точный контроль контроля и блокировки в случае пожара;
- формирование необходимого резерва для холодопроизводительности.

Для организации систем вентиляции и кондиционирования на высоком уровне требуется знания и строгое соблюдение правил строительных норм.

Самые популярные системы кондиционирования воздуха является сплит-системы для подвесных стен и потолков. У каждого из них имеются как преимущества, так и недостатки.

Сплит-системы являются недорогими, но к тому же эффективными и требующие мало места. Недостаток сплит-систем заключается в том, что это оборудование имеет очень низкую охлаждаемость и ограниченную длину.

В помещении программиста используются, как и сплит-системы так и система вентиляции. Вентиляция представляет собой организованный и регулируемый воздухообмен, обеспечивающий удаление из помещений воздуха, загрязненного вредными газами, пылью, а также улучшающий микроклиматические условия в помещении.

Вентиляция в помещении работает по принципу вытяжки – удаление воздуха. Вытяжная вентиляция удаляет воздух из верхней зоны.

Сплит-система под потолком очень эффективны в том плане, что обеспечивают значительную охлаждающую способность, равномерное распределение воздуха в компьютерном клубе через поток воздуха вдоль потолка. Одно из главных преимуществ таких сплит-систем в том, что расстояние внутреннего и внешнего блока может достигать пятидесяти метров.

Преимущества прецизионных моделей:

- поддержание заданных температурных параметров с высокой точностью;
- совместима с устройствами удаленного доступа для контроля над управлением микроклимата;
- безопасность, надежность и длительный срок службы при использовании по назначению;
- способность работать даже при больших колебаниях температуры.

Данная сплит-система зарекомендовала в помещениях с микроэлектронными устройствами.

### 4.1 Расчет системы кондиционирования воздуха

В помещении имеется 2 окна, общей площадью 3,92 м$^2$ (шириной 1,4 м, высотой 1,4 м), на расстоянии от пола 1 метр.

Для поддержания оптимальных условий микроклимата в помещении, оборудуем его системой кондиционирования. Произведем расчет необходимой системы кондиционирования воздуха в помещении. Для расчета воспользуемся методическими указаниями.

Теплопоступление в результате разности температур рассчитывается по формуле

$$Q_{огрЗима}=V_{пом}\cdot X_o\cdot(t_{Врасч}-t_{Нрасч})\ (Вт), \qquad (4.1)$$

где $V_{пом}$ – объем помещения, $V_{пом}$ = 18*2.5= 45(м$^3$);

$X_o$ – удельная тепловая характеристика, $X_o$ =0,42 Вт/(м$^3$ · ° С) ;

$t_{нрасч}$ – расчетная наружная температура, равная средней температуре июля в 14 часов, для Алматинской области примем $t_{нрасч}$ = 34 ° С;

$t_{врасч}$ = 23 ° С – расчетная внутренняя (оптимальная) температура.

Получим

$$Q_{огрЛетом} = 45 \cdot 0{,}42 \cdot (34 - 23) = 207{,}9\ (Вт).$$

в декабре, для Алматинской области примем $t_{нрасч}$ = 6 ° С;

$t_{врасч}$ = 23 ° С – внутренняя температура при отоплении. Получим:

$$Q_{огрЗимой} = 45 \cdot 0{,}42 \cdot (-4-23) = -510{,}3\ (Вт)$$

Так как окна в помещении программиста расположены с северной и с южной стороны, то на окна в течение суток не будут падать прямые солнечные лучи. Поэтому теплопоступление от солнечного излучения (радиации) через остекление рассчитывается по формуле:

$$Q_p = q^{II} \cdot F_0 \cdot \beta_{сз} \cdot К_1^т \cdot К_{2,} \quad (4.2)$$

где qII – тепловой поток от рассеянной солнечной радиации qII = 1972 Вт/м$^2$;

$\beta_{сз}$= 0.15 – коэффициент теплопропусканя;

$F_0$=n$S_0$ = 16*1,96=31,36 м$^2$ – площадь светового проема;

$К_1$=0,72 – коэффициент затемнения остекления переплетами;

$К_2$=0,9 – коэффициент загрязнения остекления.

$Q_р$=45 · 3, 92·0, 15·0, 72·0,9=17, 14608 (Вт).

Теплопоступление от людей зависит от интенсивности выполняемой работы и параметров окружающего воздуха. Тепло, выделяемое человеком, складывается из ощутимого (явного), то есть передаваемого в воздух помещения путем конвекции и лучеиспусканий, и скрытого тепла, затрачиваемого на испарение влаги с поверхности кожи и из легких.

Летом при 25$^0$С один мужчина выделяет явного тепла 61 Вт, а общего 102 Вт. Женщина выделяет 85% от нормы тепловыделений взрослого мужчины.

Тогда выделение явного тепла в помещении составит

$Q_л$= 61·1+61·1·0.85=61+51, 85=112, 85 (Вт).

А выделений общего тепла

$Q_л$= 102·1+102·1·0,85=102+0,85=188,7 (Вт).

Зимой при -5$^0$С один мужчина выделяет явного тепла 82 Вт, а общего – 103 Вт. Женщина выделяет 85% от нормы тепловыделений взрослого мужчины. Тогда выделение явного тепла в помещении составит

$Q_л$= 82·1+82·1·0,85=82+=69,7 (Вт).

А выделение общего тепла

$Q_л$= 103·1+103·1·0,85=103+=190,55 (Вт).

Теплопоступление от осветительных приборов, оргтехники и оборудования рассчитывается следующим образом. Теплопоступление от ламп определяется по формуле

$$Q_{осв} = \mu \cdot N_{осв} \cdot F, \quad (4.3)$$

где $\mu$ - коэффициент перехода электрической энергии в тепловую (для люминесцентных ламп $\mu$ = 0.5-0.6);

$N_{осв}$ – установленная мощность ламп ($N_{осв}$=75 Вт/м$^2$);

F – Площадь пола.

$$Q_{осв} = 0{,}5 \cdot 75 \cdot 18 = 675 \text{ (Вт)}.$$

Тепло, выделяемое производственным оборудованием, определяется по формуле

$$Q_{об} = N_{уст} \cdot K, \quad (4.4)$$

$$Q_{об} = 1{,}8 \cdot 1 \cdot 0{,}95 = 1710 \text{ (Вт)}.$$

Тепло, возникающие за счет находящейся оргтехники, - это 30% мощности оборудования

$$Q_{орг} = 1{,}8 \cdot 1 \cdot 0{,}3 = 0{,}54 \text{ (кВт)}.$$

Общее теплопоступление для помещения определяется как сумма внешних и внутренних теплопоступлений летом

$$Q = Q_{огр} + Q_р + Q_л + Q_{осв} + Q_{об} + Q_{орг}, \quad (4.5)$$

$$Q = 207{,}9 + 17{,}14608 + 188{,}7 + 675 + 1710 + 540 = 3338{,}75 \text{ (Дж)}.$$

Зимой

$$Q = -510{,}3 + 17{,}14608 + 190{,}55 + 675 + 1710 + 540 = 2622{,}40 \text{ (Дж)}.$$

Учитывая то, что тепловой баланс летом больше зимнего, рассчитываем теплонаряженность по формуле

$$Q_н = Q_{изблетом} \cdot 860 / V_{пом} = 3338{,}75 \cdot 860 / 18 = 159518 \text{ (ккал/м}^3\text{)}.$$

При $Q_н$ >20 ккал/м$^3$ , $\Delta$ t=8$^0$С.

Определение количества воздуха, необходимое для поступления в помещение:

$L = Q_{изб} \cdot 860 / C \cdot t \cdot y = 3{,}33875 \cdot 860 / 0{,}24 \cdot 8 \cdot 1{,}206 = 1240{,}035$ (м$^3$/час),

где С=0,24 ккал/(кг$^0$С) – теплоемкость воздуха;

у =1,206 кг/м$^3$ – удельная масса приточного воздуха.

Определение кратности воздухообмена

$$n=L/V_{пом}=1240,035/18=68,891\ (час^{-1}).$$

Исходя из полученных данных, выберем сплит-систему настенного типа.

Таким образом, для кондиционирования воздуха в помещении следует выбирать кондиционер с холодопроизводительностью не менее 6 кВт и циркуляцией не менее 68,891 м$^3$/час.

Для обеспечения круглогодичной работы кондиционера выберем оборудование фирмы Panasonic (Япония), модель CS-W18NKD/CU-W18NKD технические характеристики приведем в таблице 4.1.

Таблица 4.1 – технические характеристики кондиционера Panasonic CS-W18NKD/CU-W18NKD

| Технические характеристики | Значение |
|---|---|
| Уровень охлаждения, кВт | 10,5 |
| Уровень обогрева, кВт | 10,84 |
| Уровень шума внутреннего блока, дБ | 40 |
| Потребляемая мощность охлаждения, кВт | 3,67 |
| Потребляемая мощность обогрева, кВт | 3,44 |
| Циркуляция воздуха, м$^3$/ч | 158,4 |
| Электропитание | 220 В, 50 Гц |
| Габариты внутреннего блока, мм (ДхШхВ) | 350x1460x285 |
| Габариты внешнего блока, мм (ДхШхВ) | 805x900x360 |

После модернизации настенной сплит-системы, на схеме указано нахождение кондиционеров. Схема помещения после установления кондиционеров на рисунке 4.2.

Вывод по разделу безопасности жизнедеятельности. В данном разделе дипломной работы были изложены требования к рабочему месту инженера - программиста. Созданные условия должны обеспечивать комфортную работу. На основании изученной литературы по данной проблеме, были указаны оптимальные размеры рабочего стола и кресла, рабочей поверхности, а также проведен выбор системы и расчет оптимального кондиционирования производственного помещения. Соблюдение условий, определяющих оптимальную организацию рабочего места инженера - программиста, позволит сохранить хорошую работоспособность в течение всего рабочего дня, повысит как в количественном, так и в качественном отношениях

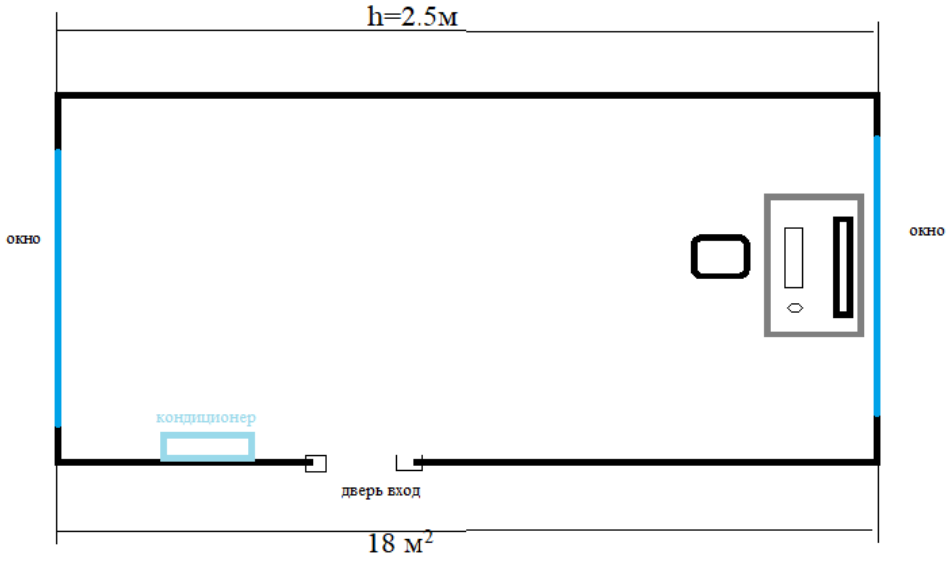



Рисунок 4.2 – Схема помещения с установленным кондиционером

производительность труда программиста, что в свою очередь будет способствовать быстрейшей разработке и отладке программного продукта. Установлен 1 кондиционер для обеспечения необходимых микроклиматических условий в соответствии с требованиями «Санитарных норм, микроклимата производственных помещений» ГОСТ 12.1.005–88 и нормального функционирования, выбран Panasonic CS-W18NKD/CU-W18NKD.

## 5 Экономическая часть

Расплата финансовой производительности плана – это обязательная часть проектирования.

### 5.1 Резюме

Задача данной заключительной работы - изучить построение проекта для прогнозирования с помощью анализа ARIMA на программе Python.

Дается расчет технико-экономических характеристик. Для возведения станет применено надлежащее оснащение: персональный компьютер, монитор, мышь, клавиатура [41].

Еще для реализации плана станут привлечены надлежащие знатоки: инженер-программист. Создание проекта займет 3-4 дня и обойдется в 1 218 200 тенге.

### 5.2 Трудоемкость разработки ПП

Ведущими задачами планирования дел считаются:

– определение размера грядущих дел;

– обоюдная увязка работы и установление здравой очередности грядущих дел;

– установление сроков выполнения работ.

Работы по планированию сводятся к составлению списка дел, определению их трудозатратности расчету продолжительности цикла дел, обоснования сметы расходов на проведение дел.

Абсолютный список дел с делением их по шагам выполнения плана следует оформить, в виде таблицы фрагмент, которой показан в таблице 5.1.

### 5.3 Материальные затраты

На основные и вспомогательные материалы относятся к материальным затратам.[42] Затрат на материальные ресурсы и стоимость оборудования производятся по форме, приведенной в таблице 5.2.

Таблица 5.1 - Распределение работ по этапам и оценка их трудоемкости

| Этапы разработки ПП | Вид работы на данном этапе | Трудоемкость разработки ПП | |
|---|---|---|---|
| | | Чел. х час | Час х день |
| Разработка | Подбор важных составляющих и библиотек Изучение возможностей, взаимосвязей и совместимостей. | 1 х 2 | 2 х 2 |
| Реализация | Сборка и настройка программы Построение кластерной части проекта Разработка и интеграция алгоритмов для анализа данных Реализация системы по сбору входных данных Реализация системы по отображению результатов | 1 х 3 | 3 х 2 |
| Тестирование | Запуск рабочей бета-версии и тестирование на "живых" данных Исправление ошибок и доработки | 1 х 2 | 2 х 1 |
| Итого трудоемкость выполнение дипломного проекта | | 1 х 4 | 1 х 4 |

Таблица 5.2 - Стоимость оборудования и ПО

| Наименование | Цена за единицу, $ | Цена за единицу, тг | Сумма, тг |
|---|---|---|---|
| Intel i9 9900K | 666 | 275300 | 275300 |
| ASUS ROG Z390 F-Gaming | 263 | 108810 | 108810 |
| ASUS ROG RTX 2080ti 11GB | 1363 | 563000 | 563000 |
| 32GB DDR4 | 157 | 65000 | 65000 |
| 6TB HDD | 187 | 77620 | 77620 |
| 500GB SSD | 107 | 44590 | 44590 |
| МЫШЬ ПРОВОДНАЯ ZOWIE S2 | 104 | 42990 | 42990 |
| КЛАВИАТУРА HYPERX ALLOY | 99 | 40890 | 40890 |

Посчитано, согласно курсу 1$=413тг[29

**5.4 Затраты на электроэнергию**

В предоставленной главе рассчитываются издержки на технологические дела, приведены в таблице 5.4. сумма затрат рассчитывается по формуле (5.1).

$$Зэ = \sum i = 1\ n\ Mi \cdot Ti \cdot Ц\ . \qquad (5.1)$$

Первого января 2020 года цена электроэнергию по тарифу ТОО «АлматыЭнергоСбыт» составляет 17,12 тенге за 1 кВтч без НДС. Цена на электроэнергию с учетом НДС составит 19,17 тенге за 1 кВтч [32

Таблица 5.3 – Затраты на электроэнергию

| Наименование оборудования | Кондиционер | Освещение |
|---|---|---|
| Паспортная мощность, кВт | 0,7 | 0,0095 |
| Коэффициент использования мощности | 0,8 | 0,7 |
| Время работы оборудования для разработки ПП, ч | 36 | 36 |
| Цена электроэнергии тг/кВт×ч | 19,17 | 19,17 |
| Сумма, тг | 386,467 | 4,589 |
| Итого затраты на электроэнергию | 391,056 | |

Арендная плата за 1 кв. м. площади определяется по формуле (5.2)

$$Ап = C \cdot S \cdot K1 \cdot K\ 2 \cdot K\ 3 \cdot K\ 4 \cdot K\ 5 \cdot K\ 6, \qquad (5.2)$$

где Ап – ставка арендной платы за пользование помещением;

С – базовая ставка арендной платы за имущественный наем 1 кв. м. - 1,5 МРП;
S – арендная площадь, квадратный метр; 1 МРП= 2778 тг.

$$K1 = 1.0, K2 = 1.0, K3 = 1\ .0, K4 = 1.0, K5 = 1.\ 0, K6 = 1\ .0.$$

Тогда

Ап=1,5 · 18 · 13· 1 ·1 · 1 · 1 ·1 · 1 = 146261 (тенге).

Таблица 5.4 – Затраты на технологические нужды

| Наименование оборудования | Паспо рт ная мощ - ность, кВт | Коэффициент использования мощности | Время работы оборудова ния для разработки ПП, ч | Цена электроэнергии тг/кВт*ч | Сумма, тг |
|---|---|---|---|---|---|
| Персональный компьютер | 0,8 | 0,7 | 36 | 19,17 | 386 |
| Итого затраты на электроэнергию | | | | | 386 |

### 5.5 Затраты на оплату труда

Эта выплаты по заработной плате выполненную работу, исчисленные на основании тарифных планов и окладов в соответствии с принятой в организации - разработчике системой оплаты труда. Затраты на оплату труда рассчитываются по форме, приведенной в таблице 5.5. Общая сумма затрат на оплату труда рассчитывается по формуле (5.3).

$$Зэ = \sum i = 1\ n\ ЧС i \cdot Ti. \qquad (5.3)$$

На реализацию проекта потребуется 3-4 дня реализации, тестированием и запуском проектом будет заниматься один инженер-программист.

Таблица 5.5 – Затраты на оплату труда

| Категория работника | Трудоемкость разработки, чел. Х месяц | Месячная оплата тг/ мес | Сумма, тг |
|---|---|---|---|
| Инженер-программист | 1 х 12 | 900000 | 10800000 |
| ИТОГО затрат на оплату труда в год | | | 10800000 |

### 5.6 Социальный налог

Отчисления на социальные нужды учитывает 13 % от затрат на оплату труда всех работников + 1,5% мед, страховка, однако пенсионные отчисления

(10%) не облагаются социальным налогом.

Обязательные пенсионные отчисления составят

$$ОПВ=10800000\cdot10\%=1080000 \text{ (тенге)}.$$

Отсюда, сумма социального налога составляет

$$СН=(10800000-1080000)\cdot14{,}5\%=1409400 \text{ (тенге)}.$$

### 5.7 Заключение по экономической части

Таким образом, не обращая внимания на конкретные трудности и трудности становления, технологии глубокого обучения делаются одним из наиглавнейших направлений формирования свежих сервисов, увеличения конкурентоспособности сервисных компаний, сотворения инноваторских рекламных инструментов продвижения в экономике.

Еще мы обсудили вопрос, который касается вентиляции в помещении, где станет трудиться наша система, со всеми работниками и оборудованием. Избрали оптимальную как раз нам систему кондиционирования, исходя из проделанных в ходе главы по безопасности жизнедеятельности расчетов. Нами был избран повешенный на стену кондиционеров серии Panasonic (Япония), модель CS-W18NKD/CU-W18NKD. Проанализировав нужные нам характеристики и технические свойства кондиционеров, мы пришли к выводу, собственно что этот повешенный на стену кондиционер подходит для наших целей.

## Заключение

В этом дипломном проекте мы представили методы глубокого обучения и прогнозировали трафик в виде временного ряда.

В этом контексте мы разработали модель ARIMA для моделирования прогнозирования спроса на трафик в телекоммуникационном производстве с использованием подхода временных рядов. Исторические данные о трафике использовались для разработки нескольких моделей и были выбраны критериями качества. С помощью этих критерий качеств, мы выявили наилучшую модель. Модель, которую мы выбрали, которая минимизирует четыре предыдущих критерия - это ARIMA с параметрами (1, 4,0, 12).

Прогноз, сделанный в этом исследовании, не учитывает некоторые другие факторы, которые могут повлиять на потребление. Например, в последние годы интенсивность трафика была очень высокой, что также положительно скажется на потреблении. Другим фактором, который может оказать влияние, является изменение поведения в праздничные дни.

Полученные результаты свидетельствуют о том, что эта модель может быть использована для моделирования и прогнозирования будущего трафика в телекоммуникации, эти результаты обеспечат руководителям надежные ориентиры при принятии решений. В будущем мы разработаем другие модели, используя комбинацию качественных и количественных методов для получения надежных прогнозов и повышения точности прогноза.

Телекоммуникационные компании в настоящее время применяют их в широком спектре приложений. Помимо этих кибер-физических взаимодействий и контрольных циклов оптимизации, представленных в этом документе, операторы ищут новые и интеллектуальные способы усовершенствовать свои услуги и усилить их роль в экосистеме поставщиков услуг.

**Список сокращений**

5G 5-ое поколение (Мобильной связи)
АЭ Авто-энкодер
БНЭ Bluetooth с низким энергопотреблением
БПЛА Беспилотный летательный аппарат
ВАЭ Вариационный авто-энкодер
ВСК Визуальная световая коммуникация
ВРТТ Обратное Распространение Во Времени
ГСД Глубокая сеть доверия
ГО Глубокое обучение
ГНС Глубокая нейронная сеть
ГОС Глубокое обучение с усилением
ГСС Генеративная состязательная сеть
гDNS Групповой DNS
ДКП Долгая краткосрочная память
ИТ Искусственный интеллект
РПОС Расширенный Протокол Очереди Сообщений
ИНС Искусственная нейронная сеть
СА Сократительный автокодер
СНС Сверточная нейронная сеть
ПЗВ Подробная Запись о Вызове
ПОП Протокол ограниченного применения
КФС Кибер-физические системы
УОМБ Условно ограниченная машина Больцмана
DNS-ОС Обнаружение службы DNS
ЭОМ Экстремальная обучающая машина
ИЛД Инъекция ложных данных
НСПР Нейронная сеть прямого распространения
ОЛМ Обобщенная линейная модель
ГП Графический процессор
ИВ Интернет вещи
ИЛП Интеллектуальный личный помощник
ИТС Интеллектуальная транспортная система
МО Машинное обучение
МП Многослойный персептрон
ОМБ Ограниченная машина Больцмана
ReLU Выпрямитель (нейронные сети)
ОП Обучение с подкреплением
РНН Рекуррентная нейронная сеть
МОВ Метод опорных векторов

## Список литературы


1 C. Engle, A. Lupher, R. Xin, M. Zaharia, M. J. Franklin, S. Shenker, and I. Stoica, “Shark: fast data analysis using coarse-grained distributed memory,” in Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data. ACM, 2012, pp. 689–692.

2 Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” nature, vol. 521, no. 7553, p. 436, 2015.

3 Большие данные // Wikipedia.org // URL: https://ru.wikipedia.org/wiki/ Большие_данные (дата обращения 20.04.19)

4 Что такое Big Data: характеристики, классификация, примеры / neurohive.io // URL: https://neurohive.io/ru/osnovy-data-science/big-data (дата обращения 21.03.19) 66

5 E. Brynjolfsson and T. Mitchell, “What can machine learning do? workforce implications,” Science, vol. 358, no. 6370, pp. 1530–1534, 2017.

6 Работа с Big Data: основные области и возможности // marketing.spb.ru // URL: https://www.marketing.spb.ru/lib-around/stat/Big_Data.htm (дата обращения 20.03.19)

7 Что такое Big data: собрали всё самое важное о больших данных // rb.ru // URL: https://rb.ru/howto/chto-takoe-big-data (дата обращения 25.03.19)

8 G. Chen, R. Xu, and S. N. Srihari, “Sequential labeling with online deep learning: Exploring model initialization,” in Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, 2016, pp. 772–788.

9 Машинное обучение //www.machinelearning.ru / / URL: http://www.machinelearning.ru/wiki/index.php

10 N. D. Lane, S. Bhattacharya, P. Georgiev, C. Forlivesi, and F. Kawsar, “An early resource characterization of deep learning on wearables, smartphones and internet-of-things devices,” in Proceedings of the 2015 International Workshop on Internet of Things towards Applications. ACM, 2015, pp. 7–12.

A. Luckow, M. Cook, N. Ashcraft, E. Weill, E. Djerekarov, and B. Vorster, “Deep learning in the automotive industry: Applications and tools,” in Big Data (Big Data), 2016 IEEE International Conference on. IEEE, 2016, pp. 3759–3768.

11 Использование больших данных сотовыми операторами на примере построения маршрутов абонентов // cyberleninka.ru // URL: https://cyberleninka.ru/article/v/ispolzovanie-bolshih-dannyh-sotovymioperatorami-na-primere-postroeniya-marshrutov-abonentov (дата обращения 15.04.19)

12 IPython // wikipedia.org // URL: https://ru.wikipedia.org/wiki/IPython (дата обращения 01.04.20)

13 Jupyter // wikipedia.org // URL: https://jupyter.org (дата обращения 03.04.20)

14 Импортирование модулей в Python // python-scripts.com // URL: https://python-scripts.com/import-modules-python (дата обращения 15.04.19)

15 NumPy // wikipedia.org // URL: https://ru.wikipedia.org/wiki/NumPy (дата обращения 07.04.20)

16 Pandas // wikipedia.org // URL: https://ru.wikipedia.org/wiki/Pandas (дата обращения 07.04.20)

A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A survey on enabling technologies, protocols, and applications,” IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015.

17 How to Create an ARIMA Model for Time Series Forecasting in Python https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/

18 Специализация Машинное обучение и анализ данных https://www.coursera.org/learn/mathematics-and-python?specialization=machine-learning-data-analysis#syllabus

19 F. Chollet, Deep learning with Python. Manning Publications, 2018

20 N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, “A deep learning approach to network intrusion detection,” Forest (RF), vol. 2, 2017

21 L. Xiao and X. Yichao, “Exact clothing retrieval approach based on deep neural network,” in Information Technology, Networking, Electronic and Automation Control Conference, IEEE. IEEE, 2016, pp. 396–400.

22 C. R. Pereira, D. R. Pereira, J. P. Papa, G. H. Rosa, and X.-S. Yang, “Convolutional neural networks applied for parkinson’s disease identification,” in Machine Learning for Health Informatics. Springer, 2016, pp. 377–390

23 P. Feng, M. Yu, S. M. Naqvi, and J. A. Chambers, “Deep learning for posture analysis in fall detection,” in Digital Signal Processing (DSP), 2014 19th International Conference on. IEEE, 2014, pp. 12–17.

24 M. Courbariaux and Y. Bengio, “Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1,” arXiv preprint arXiv:1602.02830v3 [cs.LG], 2016.

25 L. Tai and M. Liu, “Deep-learning in mobile robotics-from perception to control systems: A survey on why and why not,” arXiv preprint arXiv:1612.07139v3 [cs.RO], 2016.

26 N. Hemsoth, “GPU Platforms Set to Lengthen Deep Learning Reach,” The Next Platform, 2015. [Online] , VOL. X, NO. X, XXXXX 201X 39 Available: http://www.nextplatform.com/2015/12/07/gpu-platformsemerge-for-longer-deep-learning-reach/

27 С.Е. Мананбаева, А.С.Бегимбетов Безопасность жизнедеятельности Часть I «Производственная санитария» Учебное пособие – Алматы 2017

28 Matplotlib // Wikipedia.org // URL: https://ru.wikipedia.org/wiki/Matplotlib (дата обращения 07.04.20)

29 Официальные курсы валют на сегодня // prodengi.kz // URL: https://prodengi.kz/currency/ (дата обращения 12.05.20)

30 K. A. Steen, P. Christiansen, H. Karstoft, and R. N. Jørgensen, “Using deep learning to challenge safety standard for highly autonomous machines in agriculture,” Journal of Imaging, vol. 2, no. 1, p. 6, 2016.

31 Боканова Г.Ш., Бекишева А.И. Методические указания к выполнению расчетно-графических работ для студентов специальности

5В071900 – «Радиотехника, электроника и телекоммуникации». – Алматы: АУЭС, 2015. – 37 с. 68

32 Базылов К.Б., Алибаева С.А., Бабич А.А. Методические указания по выполнению экономического раздела выпускной работы бакалавров для студентов всех форм обучения специальности 050719 – Радиотехника, электроника и телекоммуникации – Алматы: АИЭС, - 2008. -19 с.

33 F. Chollet, Deep learning with Python. Manning Publications, 2018.Расчет по формулам https://www.kazedu.kz/referat/195775

34 СНиП 2.04.05 – 91 – отопление, вентиляция, кондиционирование ( с изменениями по И-1-94 ).

35 ГОСТ 12.1.005-88-ССБТ. Общие санитарно – гигиенические требования к воздуху рабочей зоны.

36 Приходько Н.Г. Основы безопасности жизнедеятельности. Конспект лекций для студентов всех специальностей. – Алматы: АУЭС, 2011.

37 R. E. Sutanto, L. Pribadi, and S. Lee, “3d integral imaging based augmented reality with deep learning implemented by faster r-cnn,” in International Conference on Mobile and Wireless Technology. Springer, 2017, pp. 241–247.

38 H. Maeda, Y. Sekimoto, and T. Seto, “Lightweight road manager: smartphone-based automatic determination of road damage status by deep neural network,” in Proceedings of the 5th ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems. ACM, 2016, pp. 37–45.

39 Q. Feng, Y. Zhang, C. Li, Z. Dou, and J. Wang, “Anomaly detection of spectrum in wireless communication via deep auto-encoders,” The Journal of Supercomputing, vol. 73, no. 7, pp. 3161–3178, 2017

40 R. Li, Z. Zhao, X. Zhou, G. Ding, Y. Chen, Z. Wang, and H. Zhang, “Intelligent 5g: When cellular networks meet artificial intelligence,” IEEE Wireless Communications, vol. 24, no. 5, pp. 175–183, 2017

41 J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, and A. Marrs, Disruptive technologies: Advances that will transform life, business, and the global economy. McKinsey Global Institute San Francisco, CA, 2013, vol. 180.

42 K. Panetta. (2016) Gartner’s top 10 strategic technology trends for 2017. [Online]. Available: http://www.gartner.com/smarterwithgartner/ gartners-top-10-technology-trends-2017/

43 S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards realtime object detection with region proposal networks,” IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 6, pp. 1137–1149, 2017.

44 H. Zou, H. Jiang, X. Lu, and L. Xie, “An online sequential extreme learning machine approach to Wi-Fi based indoor positioning,” in Internet of Things (WF-IoT), 2014 IEEE World Forum on. IEEE, 2014, pp. 111–116

## Приложение А

```
1. #Импортируем библиотеку:
2. import pandas as pd
3. import numpy as np
4. import statsmodels.api as sm
```

```
5. from itertools import product
6. import warnings
7. from tqdm import tqdm
8. from matplotlib import pyplot as plt
9.
10.     %pylab inline
11.
12.     ?pd.read_csv
13.
14.     nrows = 10000 #or None for all file
15.     columns = {
16.     #     'cellid': 0,
17.           'ts': 1,
18.     #      'cc': 2,
19.     #      'smsin': 3,
20.     #      'smsout': 4,
21.     #      'callin': 5,
22.     #      'callout': 6,
23.           'internet': 7
24.     }
25.     #column_ids = [0, 1, 2, 3,]
26.     dates = [27, 28, 29, 30, 31]
27.
28.     cdrs = pd.DataFrame({})
29.
30.     for date in dates:
31.           df          =          pd.read_csv('sms-call-internet-mi-2013-12-
   {}.txt'.format(date), sep='\t',
32.                                 header=None,          names=columns.keys(),
   usecols=columns.values(), nrows=nrows)
33.           cdrs = cdrs.append(df)
34.     #обнуление пустых значений
35.     cdrs.fillna(value=0, inplace=True)
36.     #учет времени с точностью до 1 часа
37.     cdrs.ts    =    pd.to_datetime(cdrs.ts,    unit='ms').apply(lambda    x:
   x.replace(minute=0, second=0))
38.
39.
40.     #print(cdrs.shape)
41.     cdrs.head()
42.     #cdrs.cellid.value_counts()
43.
44.     grouped_cdrs       =       df_cdrs_internet       =       cdrs[['ts',
   'internet']].groupby(['ts'], as_index=True).sum()
45.     grouped_cdrs
46.
47.     pylab.figure(figsize(15,7))
48.     pylab.plot(grouped_cdrs.internet)
49.     #pylab.title('')
50.     pylab.xlabel('Дата')
51.     #pylab.ylabel('Средняя ЗП, руб.')
52.     pylab.show()
53.
```

```
54.     pylab.figure()
55.     sm.tsa.seasonal_decompose(grouped_cdrs.internet, period=24).plot()
56.     pylab.show()
57.     print("Критерий Дики-Фуллера: p=%f" % sm.tsa.stattools.adfuller(grouped_cdrs.internet)[1])
58.
59.     ## Определение порядка дифференцирования
60.
61.
62.     grouped_cdrs['internet_diff1'] = grouped_cdrs.internet - grouped_cdrs.internet.shift(1)
63.     print("Критерий Дики-Фуллера: p=%e" % sm.tsa.stattools.adfuller(grouped_cdrs.internet_diff1[1:])[1])
64.
65.     grouped_cdrs['internet_diff24'] = grouped_cdrs.internet_diff1 - grouped_cdrs.internet.shift(12)
66.     print("Критерий Дики-Фуллера: p=%e" % sm.tsa.stattools.adfuller(grouped_cdrs.internet_diff24[12:])[1])
67.
68.     pylab.figure(figsize(15,8))
69.     ax = pylab.subplot(211)
70.     sm.graphics.tsa.plot_acf(grouped_cdrs.internet_diff24[12:].values.squeeze(), lags=40, ax=ax)
71.     ax.xtitle('lag')
72.     ax.ytitle('R')
73.     ax = pylab.subplot(212)
74.     sm.graphics.tsa.plot_pacf(grouped_cdrs.internet_diff24[12:].values.squeeze(), lags=20, ax=ax)
75.     ax.xtitle('lag')
76.     ax.ytitle('R')
77.     pylab.show()
78.
79.     Начальные приближения: Q=1, q=4, P=0, p=12
80.
81.
82.
83.     ps = range(0, 13)
84.     d=1
85.     qs = range(0, 5)
86.     Ps = range(0, 1)
87.     D=1
88.     Qs = range(0,2)
89.
90.     parameters = product(ps, qs, Ps, Qs)
91.     parameters_list = list(parameters)
92.     len(parameters_list)
93.
94.     %%time
95.     results = []
96.     best_aic = float("inf")
97.     warnings.filterwarnings('ignore')
98.
99.     for param in tqdm(parameters_list):
```

```
100.        #try except нужен, потому что на некоторых наборах параметров
   модель не обучается
101.        try:
102.            model=sm.tsa.statespace.SARIMAX(grouped_cdrs.internet,
   order=(param[0], d, param[1]),
103.                                                seasonal_order=(par
   am[2], D, param[3], 24)).fit(disp=-1)
104.        #выводим параметры, на которых модель не обучается и переходим
   к следующему набору
105.        except ValueError:
106.            print('wrong parameters:', param)
107.            continue
108.        aic = model.aic
109.        #сохраняем лучшую модель, aic, параметры
110.        if aic < best_aic:
111.            best_model = model
112.            best_aic = aic
113.            best_param = param
114.        results.append([param, model.aic, model.bic])
115.
116.    warnings.filterwarnings('default')
117.
118.    result_table = pd.DataFrame(results)
119.    result_table.columns = ['parameters', 'aic', 'bic']
120.    #Параметры лучшей модели по метрике AIC
121.    print(result_table.sort_values(by = 'aic', ascending=True).head())
122.
123.    print(result_table.sort_values(by = 'bic', ascending=True).head())
124.
125.    print(best_model.summary())
126.
127.    grouped_cdrs['model'] = best_model.fittedvalues
128.    pylab.figure(figsize(15,7))
129.    grouped_cdrs.internet.plot(label='original')
130.    grouped_cdrs.model[13:].plot(color='r',     label='model     fitted
   values')
131.    pylab.legend()
132.    pylab.xlabel('Дата')
133.
134.    pylab.show()
135.
136.    nforecast = 24
137.    internet2 = grouped_cdrs[['internet']]
138.
139.    date_list = [datetime.datetime.strptime("2013-12-31", "%Y-%m-%d") +
   relativedelta(hours=x) for x in range(0,nforecast)]
140.
141.    future = pd.DataFrame(index=date_list, columns= internet2.columns)
142.
143.    internet2 = pd.concat([internet2, future])
144.
```

```
145.    internet2['forecast']                                                =
   best_model.get_prediction(end=best_model.nobs                            +
   nforecast).predicted_mean
146.    pylab.figure(figsize(15,7))
147.    internet2.internet.plot()
148.    internet2.forecast[-nforecast:].plot(color='r',      linestyle='--',
   linewidth=2)
149.
150.
151.    pylab.xlabel('Дата')
152.    pylab.show()
```