

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Инженерная школа ядерных технологий
Направление подготовки 01.04.02 «Прикладная математика и информатика»
Отделение экспериментальной физики

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Создание и использование chainlet для прогнозирования цен криптовалют

УДК 004.4'414:338.5:336.743:004.738.5:339

Студент

Группа	ФИО	Подпись	Дата
0ВМ12	Карнаухов Владислав Андреевич		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Крицкий О.Л.	к.ф.-м.н., доцент		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Кащук И.В.	к.т.н доцент		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Сечин А.А.	к.т.н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Мерзликин Б.С.	к.ф.-м.н., доцент		

Томск – 2023 г.

Планируемые результаты обучения по ООП

Код компетенции	Наименование компетенции
Профессиональные компетенции	
ПК(У)-1	Способен проводить научные исследования и получать новые научные и прикладные результаты самостоятельно и в составе научного коллектива
ПК(У)-2	Способен проводить поиск и анализ научной и научно-технической литературы по тематике проводимых исследований
ПК(У)-3	Способен разрабатывать и анализировать показатели качества информационных систем, используемых в производственной деятельности
ПК(У)-4	Способен планировать научно-исследовательскую деятельность, анализировать риски, управлять проектами, управлять командой проекта
ПК(У)-5	Способен преподаванию математических дисциплин и информатики в образовательных организациях высшего образования
ПК(У)-6	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий
Общепрофессиональные компетенции	
ОПК(У)-1	Способен решать актуальные задачи фундаментальной и прикладной математики
ОПК(У)-2	Способен совершенствовать и реализовывать новые математические методы решения прикладных задач
ОПК(У)-3	Способен разрабатывать математические модели и проводить их анализ при решении задач в области профессиональной деятельности
ОПК(У)-4	Способен комбинировать и адаптировать существующие информационно-коммуникационные технологии для решения задач в области профессиональной деятельности с учетом требований информационной безопасности
Универсальные компетенции	
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном языке, для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа ядерных технологий
Направление подготовки 01.04.02 «Прикладная математика и информатика»
Отделение экспериментальной физики

УТВЕРЖДАЮ:

Руководитель ООП

_____ Мерзликин Б.С.

(Подпись) (Дата) (Ф.И.О.)

**ЗАДАНИЕ
на выполнение выпускной квалификационной работы**

В форме:

Магистерской диссертации

Студенту:

Группа	ФИО
ОВМ12	Карнаухову Владиславу Андреевичу

Тема работы:

Создание и использование chainlet для прогнозирования цен криптовалют	
Утверждена приказом директора (дата, номер)	№ 142-12/с от 22.05.2023

Срок сдачи студентом выполненной работы:	31.05.2023
--	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	<i>Данные о переводах криптовалюты, которые хранятся в транзакциях блокчейна.</i>
---------------------------------	---

Перечень подлежащих исследованию, проектированию и разработке вопросов	<ol style="list-style-type: none"> 1. Провести анализ литературы по теме диссертации. 2. Разработка модели для прогнозирования с использованием <i>chainlet</i>. 3. Разработка программного модуля для сбора данных из блокчейна и построения <i>chainlet</i>. 4. Реализация программного модуля для прогнозирования цены криптовалюты. 5. Верификация модели на статистических данных.
Перечень графического материала	<ol style="list-style-type: none"> 1. Иллюстрации устройства рекуррентных нейронных сетей. 2. Визуализация исходных данных. 3. Графики сопоставления прогнозов моделей с реальными данными.
Консультанты по разделам выпускной квалификационной работы <i>(если необходимо, с указанием разделов)</i>	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и энергосбережение	Кашук И.В.
Социальная ответственность	Сечин А.А.
Иностранный язык	Смирнова У.А.

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	15.03.2023 г.
---	---------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Мерзликин Борис Сергеевич	к. ф.-м. н., доцент		15.03.2023 г.

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0ВМ12	Карнаухов Владислав Андреевич		15.03.2023 г.

ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ
И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
0BM12	Карнаухов Владислав Андреевич

Школа	ИЯТШ	Отделение Школа	Отделение экспериментальной физики
Уровень образования	магистратура	Направление/специальность	01.04.02 Прикладная математика и информатика

Исходные данные к разделу «Создание и использование chainlet для прогнозирования цен криптовалют»:

<i>1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Стоимость материальных ресурсов и специального оборудования определены в соответствии с рыночными ценами г. Томска Тарифные ставки исполнителей определены штатным расписанием НИ ТПУ
<i>2. Нормы и нормативы расходования ресурсов</i>	Норма амортизационных отчислений на специальное оборудование
<i>3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Отчисления во внебюджетные фонды 30 %

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<i>1. Анализ конкурентных технических решений (НИ)</i>	Расчет конкурентоспособности SWOT-анализ
<i>2. Формирование плана и графика разработки и внедрения (НИ)</i>	Структура работ. Определение трудоемкости. Разработка графика проведения исследования
<i>3. Составление бюджета инженерного проекта (НИ)</i>	Расчет бюджетной стоимости НИ
<i>4. Оценка ресурсной, финансовой, бюджетной эффективности (НИ)</i>	Интегральный финансовый показатель. Интегральный показатель ресурсоэффективности. Интегральный показатель эффективности.

Перечень графического материала

<ol style="list-style-type: none"> 1. Оценка конкурентоспособности ИП 2. Матрица SWOT 3. Диаграмма Ганта 4. Бюджет НИ 5. Основные показатели эффективности НИ 	
--	--

Дата выдачи задания для раздела по линейному графику	01.03.2023
--	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН ШБИП	Кашук Ирина Вадимовна	к.т.н доцент		01.03.2023

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0ВМ12	Карнаухов Владислав Андреевич		01.03.2023

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа		ФИО	
0ВМ12		Карнаухов Владислав Андреевич	
Школа	ИЯТШ	Отделение (НОЦ)	Отделение экспериментальной физики
Уровень образования	магистратура	Направление/специальность	01.04.02 Прикладная математика и информатика

Тема ВКР:

Создание и использование chainlet для прогнозирования цен криптовалют	
Исходные данные к разделу «Социальная ответственность»:	
<p>Введение</p> <ul style="list-style-type: none"> – Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения. – Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации 	<p><i>Объект исследования:</i> разработка алгоритма <i>Область применения:</i> принятие решения студентом <i>Рабочая зона:</i> <u>офис</u> <i>Размеры помещения</i> 15 м x 5 м x 4 м <i>Количество и наименование оборудования рабочей зоны:</i> компьютеры <i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне, разработка алгоритма на компьютере</i></p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>- Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 27.12.2018) - ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования.</p>
<p>2. Производственная безопасность при разработке проектного решения:</p> <ul style="list-style-type: none"> – Анализ выявленных вредных и опасных производственных факторов – Расчет уровня опасного или вредного производственного фактора 	<p>Вредные факторы: - Отклонение показателей микроклимата; - Отсутствие или недостаток необходимого искусственного освещения Опасные факторы: - Повышенный уровень электромагнитных излучений; - Опасные и вредные производственные факторы, связанные с электрическим током</p>
<p>3. Экологическая безопасность при разработке проектного решения</p>	<p>Воздействие на литосферу: образования отходов при написании работы Воздействие на гидросферу: энерго и теплопотребление Воздействие на атмосферу: энерго и теплопотребление</p>
<p>4. Безопасность в чрезвычайных ситуациях при разработке проектного решения</p>	<p>Возможные ЧС аварии, пожары Наиболее типичная ЧС пожар</p>

Дата выдачи задания для раздела по линейному графику	01.03.2023
--	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Сечин Андрей Александрович	к.т.н		01.03.2023

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0ВМ12	Карнаухов Владислав Андреевич		01.03.2023

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Инженерная школа ядерных технологий
Направление подготовки 01.04.02 «Прикладная математика и информатика»
Отделение экспериментальной физики
Период выполнения весенний семестр 2022/2023 учебного года

Форма представления работы:

Магистерская диссертация

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
01.03.2023	Выдача задания	5
14.03.2023	Обсуждение структуры работы	5
01.04.2023	Обзор литературы	10
15.04.2023	Реализация методов извлечения и построения chainlet	15
01.05.2023	Создание модели ARIMA-GARCH	30
10.05.2023	Создание модели LSTM	15
25.05.2023	Анализ полученных результатов	10
31.05.2023	Написание пояснительной записки ВКР	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Крицкий О.Л.	к.ф.-м.н.		

СОГЛАСОВАНО:

Руководитель ООП

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Мерзликин Б.С.	к.ф.-м.н., доцент		

Реферат

Выпускная квалификационная работа содержит 104 страницы, 19 рисунков, 34 таблиц, 30 источников, 39 формулы.

Ключевые слова: блокчейн, chainlet, цепочки графов, биткоин, прогнозирование временных рядов, криптовалюта, стоимость криптовалют, LSTM.

Объектом исследования является проблема прогнозирования стоимости криптовалют.

Целью работы является разработка и реализация методов для построения chainlet и модели прогнозирования с использованием chainlet извлеченных из транзакций блокчейна.

Область применения: прогнозирование временных рядов.

В представленной работе разработаны и реализованы методы по извлечению, построению и анализу chainlet. Так же были разработаны и реализованы модели прогнозирования ARIMA, ARIMA-GARCH, LSTM, позволяющие прогнозировать стоимость криптовалют, с использованием цепочек сетевого графа блокчейна.

Определения, обозначения, сокращения и нормативные ссылки

В данной работе приведены следующие термины с соответствующими определениями:

ARIMA – интегрированная модель авторегрессии скользящего среднего;

GARCH – обобщенная модель авторегрессионной условной гетероскедастичности;

AIC - информационный критерий Акаике;

RNN – рекуррентная нейронная сеть;

LSTM – нейронная сеть с долгой краткосрочной памятью;

Lookback – параметр скользящего окна;

Batchsize – параметр отвечающие за количество входных данных после которого происходит оптимизация весов нейронной сети;

MAPE – средняя абсолютная процентная ошибка;

BTC – биткоин;

Экстремальные цепочки – цепочки в которых количество входов или выходов превышает пороговое значение;

RESTful – архитектурный стиль взаимодействия компонентов распределённого приложения в сети;

API – описание способов взаимодействия одной компьютерной программы с другими.

Оглавление

Реферат	10
Определения, обозначения, сокращения и нормативные ссылки.....	11
Введение.....	15
1 Обзор научно-технической литературы по исследованию.....	17
1.1 Chainlets – цепочки графов	17
1.2 Модель прогнозирования ARIMA.....	21
1.2.1 Модель ARIMA	21
1.2.2 Семейство ARCH моделей.....	22
1.2.3 Модель ARIMA-GARCH.....	23
1.3 Рекуррентные нейронные сети	24
1.3.1 Нейронная сеть с долгой краткосрочной памятью	25
2. Реализация.....	29
2.1 Получение Chainlets.....	29
2.2 Исходные данные.....	30
2.3 Модель прогнозирования ARIMA-GARCH	32
2.4 Модель прогнозирования LSTM	36
Заключение	39
3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	40
Введение.....	40
3.1 Оценка коммерческого потенциала и перспективности проведения исследований с позиции ресурсоэффективности и ресурсосбережения.....	40
3.1.1 Анализ конкурентных технических решений.....	40
3.1.2 SWOT-анализ	42
3.2 Планирование научно-исследовательских работ.....	46
3.2.1 Структура работ в рамках научного исследования.....	46

3.2.2 Определение трудоемкости выполнения работ и разработка графика проведения	47
3.3 Бюджет научно-технического исследования	50
3.3.1 Расчет материальных затрат научно-технического исследования	50
3.3.2 Расчет амортизации специального оборудования.....	51
3.3.3 Основная заработная плата исполнителей темы	52
3.3.4 Отчисления во внебюджетные фонды (страховые отчисления).....	54
3.3.5 Накладные расходы	54
3.3.6 Бюджет НИР	55
3.4 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....	56
Выводы по разделу	59
4. Социальная ответственность	60
4.1 Правовые и организационные вопросы обеспечения безопасности	60
4.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства	60
4.2. Производственная безопасность	63
4.2.1. Производственная безопасность	63
4.2.2 Отклонение параметров микроклимата.....	63
4.2.3 Недостаточная освещенность рабочей зоны.....	64
4.2.4 Повышенный уровень электромагнитных излучений	65
4.2.5 Поражение электрическим током	65
4.3. Экологическая безопасность.....	68
4.3.1 Влияние на атмосферу и гидросферу	68
4.3.2 Влияние на литосферу.....	69
4.4 Безопасность в чрезвычайных ситуациях	69

4.4.1. Анализ вероятных ЧС, которые может инициировать объект исследований и обоснование мероприятий по предотвращению ЧС.....	69
4.4.2. Меры по предупреждению возникновения пожара	70
4.4.3 Действия в случае возникновения пожара	71
Вывод по разделу	73
Список публикаций студента.....	74
Список литературы	75
Приложение А	78
Приложение Б.....	90
Приложение В.....	91
Приложение Г	93
Приложение Д.....	94
Приложение Е.....	98
Приложение Ж.....	101

Введение

Биткойн вызвал огромный интерес и достиг стремительного роста за последние пару лет. Феномен биткойна вызван не только из-за революции в онлайн-платежах, но и из-за большого количества применения лежащей в основе технологии блокчейна.

Особенность биткойна заключается в том, что распределенный реестр поддерживается всеми участниками для проверки подлинности каждой сделки. Существование такого распределенного реестра создает уникальные возможности в отношении анализа графов. Уже разные приложения использовали распределенный реестр и информацию о графе биткойнов для отслеживания различной преступной деятельности [1, 2].

Биткойн-граф можно использовать для других интересных приложений. Например, на большинстве платформ анализа акций рыночный тренд обычно прогнозируется с использованием только исторических цен и других финансово-экономических показателей без учета структуры финансовой сети. Поскольку есть возможность наблюдать полный биткойн-граф, то возникает вопрос, влияет ли структура локального графа на цену актива [3, 4].

В других областях - локальные структуры более высокого порядка сложных сетей, или много узловые подграфы, оказываются незаменимым инструментом для анализ сетевой организации за пределами тривиального масштаба отдельных вершин и ребер. Основная идея заключается в том, что если конкретный подграф встречается более или менее часто чем ожидается, то такой подграф, вероятно, будет играть важную роль в сетевой функциональности.

Целью настоящей магистерской диссертации является разработка и реализация модели для прогнозирования стоимости криптовалют с использованием дополнительной информации из блокчейна о транзакциях, называемой *chainlet*. Для достижения данной цели были поставлены следующие задачи:

- Изучить существующие модели прогнозирования.

- Разработать методы для извлечения и построения chainlet.
- Реализовать модели прогнозирования
- Провести сравнительный анализ и сделать выводы.

Научная и практическая новизна и значимость работы:

В работе рассматривается построение модели прогнозирования, которая помимо стоимости закрытия так же учитывает дополнительные параметры, которые были извлечены из открытого распределенного реестра. Данная работа демонстрирует, что для прогнозирования стоимости криптовалют можно использовать дополнительную информацию, которая не доступна при прогнозировании ценных бумаг.

1 Обзор научно-технической литературы по исследованию

1.1 Chainlets – цепочки графов

Как показано на рисунке 1. биткойн-граф состоит из трех основных компонентов: адрес(кошелек), транзакции и блока. Транзакция может иметь несколько входов(inputs) и несколько выходов(outputs). Например, на рисунке 1 транзакция t_1 получает биткойны от адресов a_1 и a_2 , и перечисляет на адреса a_6 и a_7 . Аналогия из реальной жизни — это человек, использующий несколько банковских счетов, объединяющий средств в одну транзакцию и отправляющий суммы на несколько учетных записей. Существуют сервисы по смешиванию биткойнов, биткойн-миксеры, которые работают в основном по простой схеме: средства клиента дробятся на мелкие части, после чего эти части смешиваются в случайном порядке с частями других клиентов. В результате всех операций к конечному получателю приходит заданное количество биткойнов, но небольшими партиями от разных случайно выбранных участников. Таким образом, адреса, которые появляются на входе и на выходе одни и те же.

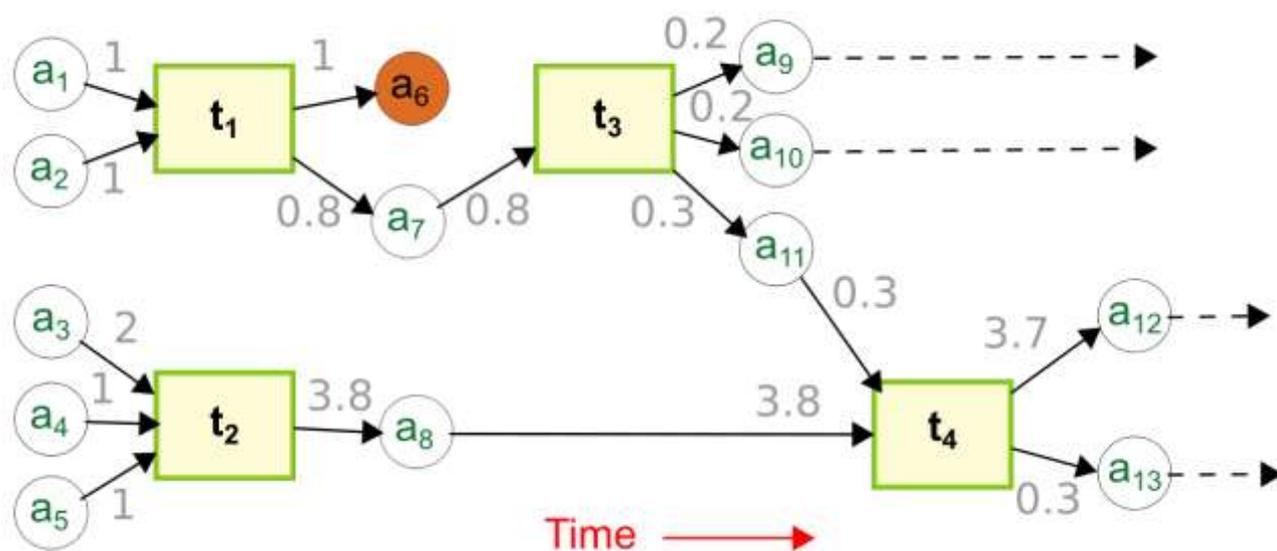


Рисунок 1 - Графическое представление сети Биткойн в виде графа адресов-транзакций (по данным [3])

Входы и выходы могут предоставить важную информацию о целях транзакции. Например, транзакции, в которых участвуют сотни входов и очень мало выходов могут быть большими суммами биткойнов инвестиций.

K -chainlet — это подграф Биткойна, состоящий из $k \geq 1$ транзакций и их соответствующие входные и выходные адреса. В простейшем случае, одна транзакция создает 1-chainlet с одним или несколькими входами и одним выходом. Например, на рисунке 1 транзакция t_2 приводит к передаче биткойнов с адресов a_3, a_4, a_5 на адрес a_8 . Такая транзакция создает 1-chainlet с тремя входами и одним выходом. Обозначим этот подграф как $C_3 \rightarrow 1$, где 3 и 1 - количество адресов входов и выходов, соответственно.

1-chainlet — это наименьший элемент графа Биткойна; входы и выходы цепочки определяются сразу, а транзакция подписана цифровой подписью. Эта подписанная информация не может быть изменена, но несколько цепочек 1-chainlet могут быть объединены для увеличения графа.

Матрица цепочек на рисунке. 2 иллюстрирует, как активность сети может быть представлена, цветовое обозначение указывает на частоту появления цепочки. За определенный период подсчитывается появление каждого $C_i \rightarrow j$. Максимальное количество входов или выходов цепочки может быть большим, иногда превышающие 1000. Цепочки, в которых количество входов и выходов превышает пороговое значение N называются "экстремальными". На основе исторического анализе ежедневных снимков было выбрано $N = 20$, что соответствует 97,5 перцентиллю всей цепочки сети [3].

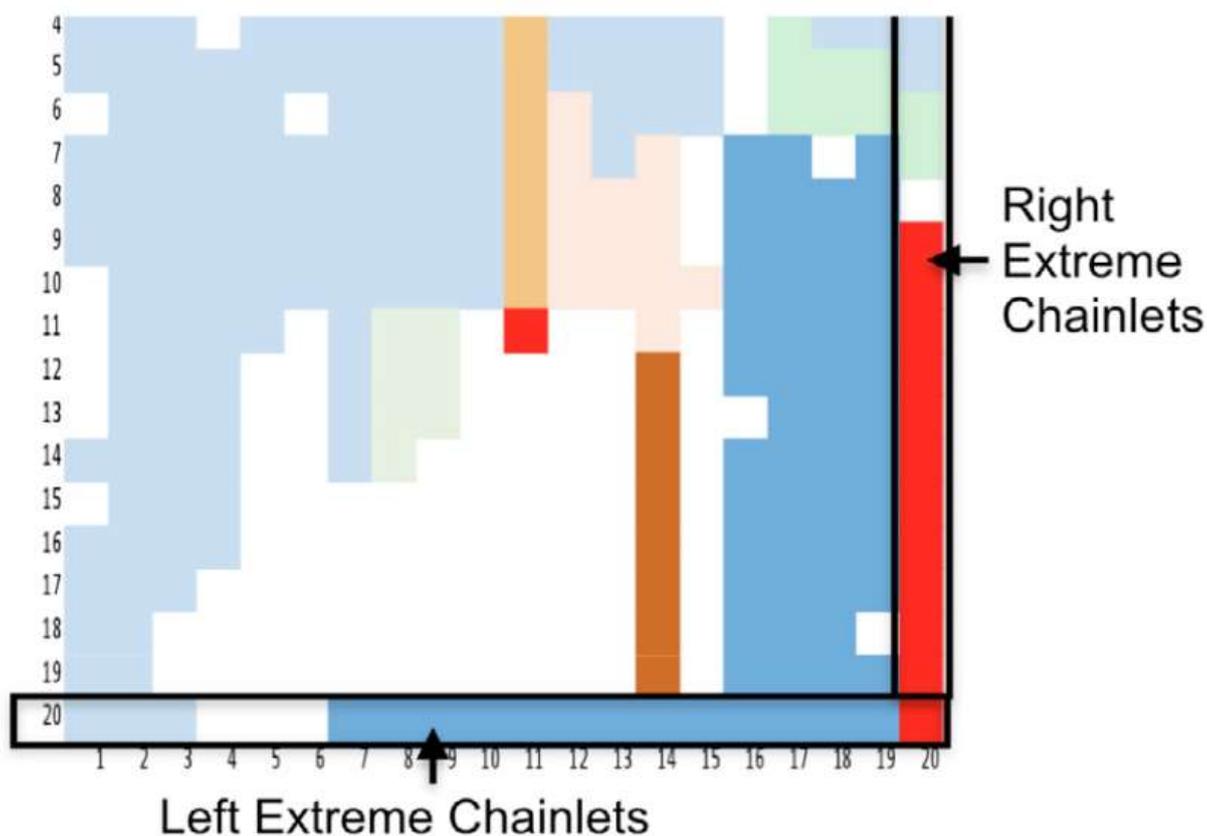


Рисунок 2 - Представление сети в виде матрицы цепочек.

Существуют «левые экстремальные цепочки». и «правые экстремальные цепочки» (по данным [3])

Левые экстремальные цепочки — это подмножество C^l .

$$C^l := \{C_i \rightarrow j \mid i = N, j \in \{1, \dots, N\}\} \quad (1)$$

Они представляют собой транзакции из большого количества адресов на меньшее количество адресов. Как правило, левые экстремальные цепочки указывают на вложения в биткойны.

Правые экстремальные цепочки — это подмножество C^r .

$$C^r := \{C_i \rightarrow j \mid i = \{1, \dots, N - 1\}, j \in N\} \quad (2)$$

Они представляют собой продажу большой суммы биткойнов - продавец по частям распродает свои монеты и отправляет их потенциально на сотни биткойн-адресов.

Обозначим сумму долларов, переведенных в момент t левыми экстремальными и правыми экстремальными цепочками как A_t^l и A_t^r , а общее количество цепочек как O_t^l и O_t^r соответственно.

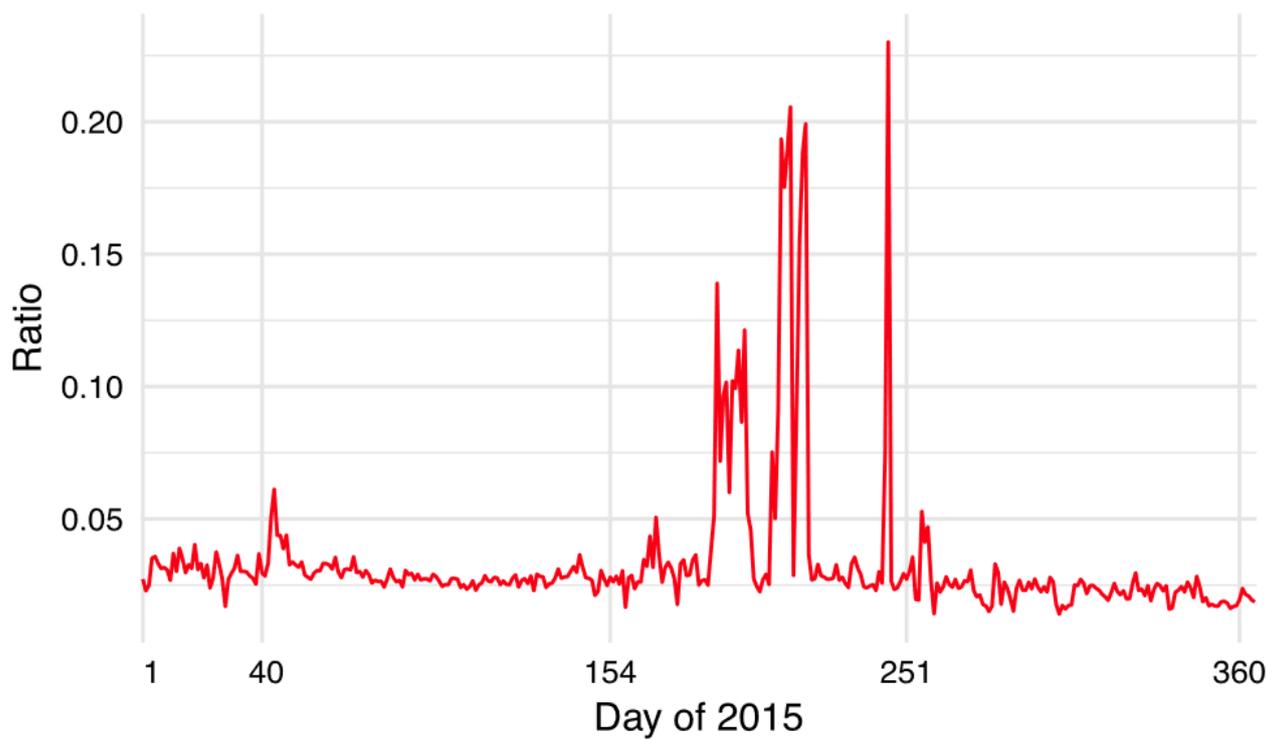


Рисунок 3 - Соотношение экстремальных цепочек к общему количеству (по данным [5])



Рисунок 4 - График стоимости и объемов торгов биткойна за 2015г (по данным [5])

2015 год. 3 июня (день 154), департамент финансовых услуг штата Нью-Йорк выдвинул свод правил BitLicense: набор правил, которые будут регулировать бизнес, связанный с виртуальной валютой. BitLicense вступил в силу 8 сентября (251 день). Вместо того, чтобы соблюдать эти правила, криптовалютные биржи потребовали от своих клиентов покинуть их платформы. Многие клиенты ушли, продав свои биткойны, о чем свидетельствует большое количество экстремальных цепочек.

1.2 Модель прогнозирования ARIMA

1.2.1 Модель ARIMA

В 1976 году Бокс и Дженкинс предложили модель AutoRegressive Integrated Moving Average (ARIMA) [6]. В модели ARIMA есть 3 параметра, которые используются для моделирования основных аспектов временного ряда: сезонность, тренд и шум. Эти параметры обозначены буквами p , d и q .

- p - параметр, связанный с авторегрессивным аспектом модели, который включает прошлые значения. Например, прогнозируя, что, если в последние несколько дней цена на акции росла, вы указываете, что вероятно, цены вырастут и завтра.
- d - параметр, связанный с интегрированной частью модели, который влияет на величину разницы, применяемую к временному ряду. Например, что объем торгов по акции завтра будет аналогичен объему сегодня, если ежедневный объем торгов был аналогичным в течение последних нескольких дней.
- q - параметр, связанный со скользящей средней частью модели.

Модель $ARIMA(p, d, q)$ для нестационарного временного ряда X_t имеет вид:

$$\Delta^d X_t = a_0 + \sum_{i=1}^p a_i \Delta^d X_{t-1} + \sum_{j=1}^q b_j \varepsilon_{t-j} + \varepsilon_t \quad (3)$$

где ε_t – ошибки наблюдений с нормальным распределением;

a_i, b_j – параметры модели;

Δ^d - оператор разности временного ряда порядка d (последовательное взятие d раз разностей первого порядка — сначала от временного ряда, затем от полученных разностей первого порядка, затем от второго порядка и т. д.).

1.2.2 Семейство ARCH моделей

ARCH (Autoregressive Conditional Heteroscedastic model) модели были предложены Робертом Энглом в 1982 году и с тех пор стали популярным инструментом анализа финансовых временных рядов [6]. Данная модель получила название ARCH-модель (Autoregressive Conditional Heteroscedastic model), в которой используется условная, зависящая от времени дисперсия, выражаемая через квадрат значений показателей прошлых периодов:

$$\sigma^2(t) = a_0 + \sum_{i=1}^q a_i \mu_{t-1}^2 \quad (4)$$

где q - коэффициент задержки (лага);

a_i, a_0 – весовые коэффициента;

μ_n - относительные приращения значений временного ряда или логарифмические приращения.

GARCH (Generalized Autoregressive Conditional Heteroscedastic model) модели являются наиболее распространенным вариантом ARCH моделей и используются для анализа финансовых временных рядов с изменчивым риском. Они были предложены Тимом Боллерселом и Янем Тиммерсом в 1986 году и представляют собой обобщение ARCH моделей [6].

GARCH модели могут быть адаптированы к различным инструментам и применены к различным финансовым рядам (валютным курсам, акциям, криптовалютам) [6]. Они позволяют проводить более точный анализ риска, прогнозировать будущее состояние ряда и помогают в принятии правильных инвестиционных решений.

Согласно данной модели (GARCH(p,q)), расчет дисперсии производится по следующей формуле:

$$\sigma^2(t) = \alpha_0 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{j=1}^q \alpha_j \mu_{t-j}^2 \quad (5)$$

где p - количество предыдущих оценок, влияющих на текущее значение;

α , β – весовые коэффициенты, отражающие степень влияния предыдущих оценок на текущее значения;

σ_n – волатильность.

1.2.3 Модель ARIMA-GARCH

Для объединения моделей ARIMA и GARCH необходимо воспользоваться следующей формулой:

$$\hat{y}_t = a_0 + \sum_{i=1}^p a_i y_{t-i} + \sum_{i=1}^q b_i \varepsilon_{t-i} + \varepsilon_t \quad (6)$$

$$\varepsilon_t = \sqrt{\sigma_t} z_t \quad (7)$$

$$\sigma^2 = \omega + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2 \quad (8)$$

$$z_t = iid N(0,1) \quad (9)$$

Окончательный прогноз дается путем объединения выходных данных модели ARIMA и модели GARCH, при этом модель GARCH обучается на остаточных ошибках прогнозов ARIMA модели, которые выражены формулой (7).

1.3 Рекуррентные нейронные сети

Рекуррентные нейронные сети (RNN – Recurrent Neural Network) – класс нейронных сетей, который используется в машинном обучении для обработки последовательностей данных, таких как тексты, временные ряды, звуки и другие. Рекуррентные нейронные сети умеют сохранять контекст в процессе обработки данных и в дальнейшем использовать их для принятия решения [7].

Происходит это за счет того, что на каждом шаге обучения t значение скрытого слоя рекуррентной нейронной сети $h^t \in R^m$ вычисляется следующим образом:

$$h^t = f(W \times x^t + U \times h^{(t-1)} + b^h), \quad (10)$$

где $x^t \in R^n$ – входной вектор в момент времени t (например, стоимость криптовалюты за n дней);

$W \in R^{m \times n}$, $U \in R^{m \times m}$, $b^h \in R^m$ – обучаемые параметры рекуррентной нейронной сети;

f – функция нелинейного преобразования.

Чаще всего в качестве функции нелинейного преобразования выступает одна из следующих функций: сигмоидальная функция (11), гиперболический тангенс (12), «ReLU» (13).

$$f(x) = \sigma(x) = \frac{1}{1 + e^{(-x)}}, \quad (11)$$

$$f(x) = \tanh(x) = \frac{e^{(x)} - e^{(-x)}}{e^{(x)} + e^{(-x)}}, \quad (12)$$

$$f(x) = \max(0, x), \quad (13)$$

В простой рекуррентной нейронной сети (рисунок 5) выходное значение $y^t \in R^l$ на текущем шаге t вычисляется по формуле:

$$y^t = W \times h^t + b, \quad (14)$$

где $W \in R^{l \times m}$ и $b \in R^l$ – обучаемые параметры.

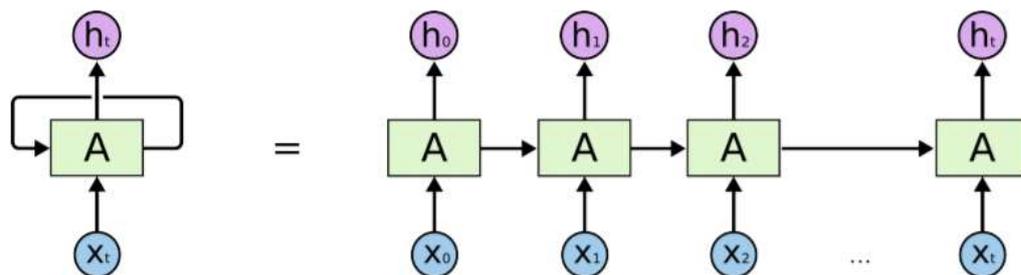


Рисунок 5 – Простейшая система рекуррентной сети (по данным [8])

Применение данных систем используется во многих задачах. RNN показали свою эффективность в анализе временных рядов и прогнозировании будущих значений. Они могут использоваться для анализа технических индикаторов, таких как скользящие средние или стохастический осциллятор, а также для анализа фундаментальных факторов, таких как отчеты об убытках и прибылях компаний, макроэкономические показатели и другие.

Одной из самых популярных реализаций RNN, являются модели LSTM (Long Short Term Memory), которые имеют специальные механизмы для более оптимальной работы с памятью, чем RNN. Более подробное рассмотрение рекуррентных сетей будет приведено в следующем подразделе.

1.3.1 Нейронная сеть с долгой краткосрочной памятью

В 1997 году был представлен новый подход, названный долгой краткосрочная память (LSTM), авторами которого стали Зепп Хохрайтер и Юрген Шмидхубер [9]. Рекуррентные нейронные сети, основанные на этом подходе, имеют более сложный способ вычисления h^t . Для этого используются входные значения, предыдущее состояние сети и специальные фильтры («gates»), которые определяют, как обрабатывать информацию для получения выходных значений на текущем слое y^t и значений скрытого слоя на следующем шаге h^{t+1} .

Все рекуррентные нейронные сети имеют форму цепочки повторяющихся модулей нейронной сети. В стандартных RNN повторяющийся модуль имеет простую структуру, например, он может включать в себя один слой гиперболического тангенса, изображенного на рисунке 6.

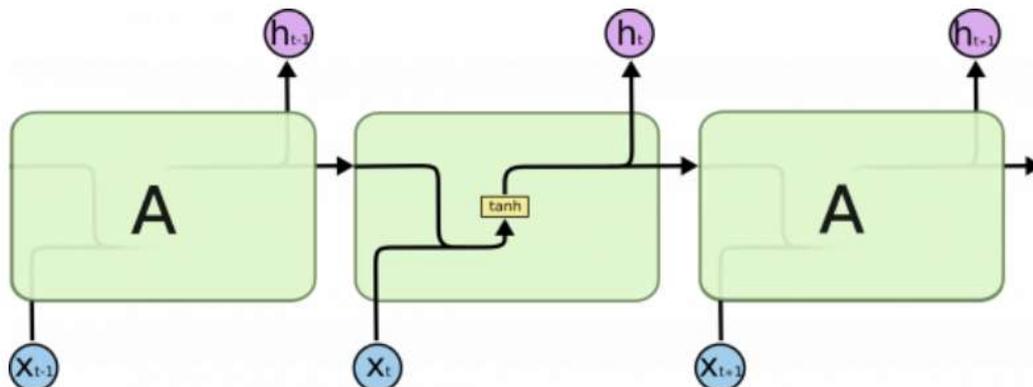


Рисунок 6 – Схематическое изображение повторяющегося модуля стандартной RNN, состоящей из одного слоя (по данным [8])

В свою очередь на рисунке 7 изображена LSTM сеть, которая имеет явные отличия. Для того чтобы разобраться в них, рассмотрим их математические особенности.

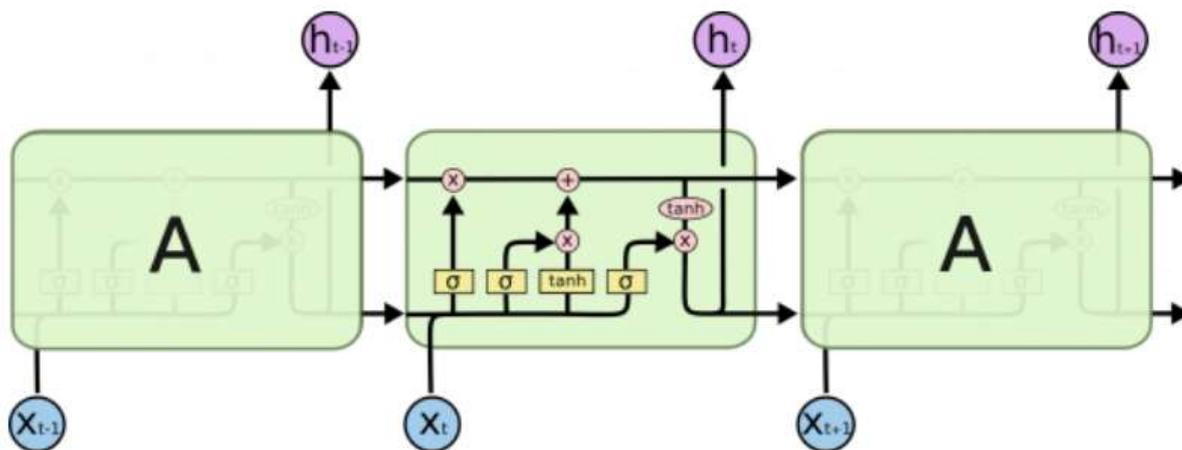


Рисунок 7 – Схематическое изображение повторяющегося модуля LSTM, состоящего из четырех взаимодействующих слоев (по данным [8])

Данная модель разработана специально для устранения проблемы долгосрочной зависимости. Главной особенностью является запоминание

информации в течение длительных периодов времени, что обеспечивает отсутствие необходимости в обучении.

Рассмотрим подробнее структуру LSTM-слоя. Центральным элементом здесь является запоминающий блок («memory cell»), который, наряду с состоянием сети h , вычисляется на каждом шаге, используя текущее входное значение x , t и значение блока на предыдущем шаге c^{t-1} . Входной фильтр («input gate») i^t определяет, насколько значение блока памяти на текущем шаге должно влиять на результат. Значения фильтра варьируются от 0 (полностью игнорировать входные значения) до 1, что обеспечивается областью значений сигмоидальной функции:

$$i^t = \sigma(W^i x^t + U^i h^{t-1}), \quad (15)$$

Фильтр забывания («forget gate») позволяет исключить при вычислениях значения памяти предыдущего шага:

$$f^t = \sigma(W^f x^t + U^f h^{t-1}), \quad (16)$$

На основе всех данных, поступающих в момент времени t , вычисляется состояние блока памяти с t на текущем шаге, используя фильтры (17) и (18):

$$\check{c}^t = \tanh(W^c x^t + U^c h^{t-1}), \quad (17)$$

$$c^t = f^t \cdot c^{t-1} + i^t \cdot \check{c}^t, \quad (18)$$

Выходной фильтр («output gate») аналогичен двум предыдущим и имеет вид:

$$o^t = \sigma(W^o x^t + U^o h^{t-1}), \quad (19)$$

Итоговое значение LSTM-слоя определяется выходным фильтром (19) и нелинейной трансформацией над состоянием блока памяти (20):

$$h^t = o^t \cdot \tanh(c^t), \quad (20)$$

Существует множество вариантов используемых каждым слоем функций активации, возможны некоторые небольшие изменения самой схемы и каких-

либо её параметров. Однако при этом суть функционирования не меняется – сначала фильтруют часть памяти, затем запоминают часть нового сигнала, и уже потом на основе этих данных вычисляется результат. Например, один из популярных вариантов LSTM-сети представлен на рисунке 8. Его предложили в 2000 году Феликс Герс и Юрген Шмидхубер в своей научной работе [9].

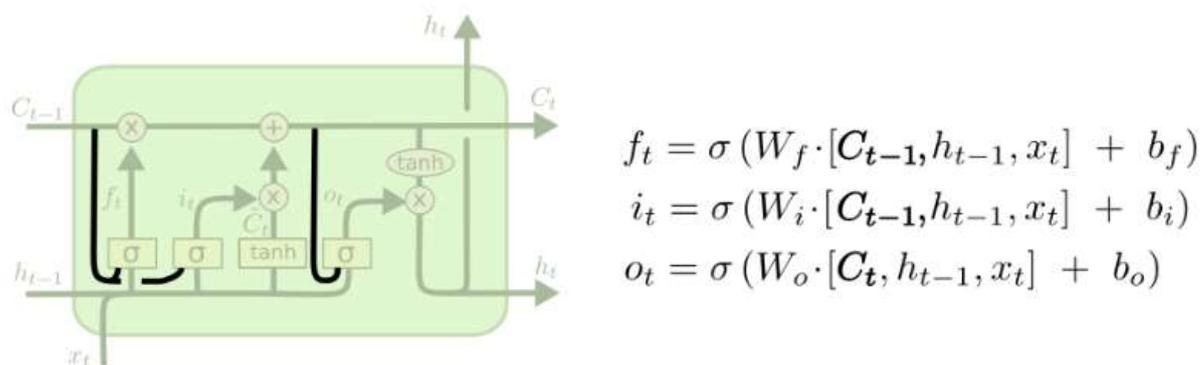


Рисунок 8 – Схематическое изображение LSTM модели Герса и Шмидхубера (по данным [8])

2. Реализация

2.1 Получение Chainlets

Для данной работы был выбран биткойн так как он является основной криптовалютой и все остальные криптовалюты имеют сильную корреляцию с ним. Размер блокчейна биткойна превышает 450 Гб, поэтому вариант извлечения транзакций через установку собственной ноды блокчейна весьма трудозатратен по времени и требователен к наличию свободного места. В качестве альтернативного варианта рассмотрим сервис Google BigQuery.- это бессерверное, масштабируемое облачное хранилище данных с мощной инфраструктурой от Google, которое имеет на борту RESTful веб-сервис. Имеет тесное взаимодействие с другими сервисами от Google. BigQuery содержит огромное количество публичных датасетов.

В BigQuery доступен датасет `bigquery-public-data.crypto_bitcoin.transactions`, который содержит необходимые нам данные: время транзакции, количество входов, количество выходов, сумма входов в сатоши, сумма выходов в сатоши. Особенностью данной таблицы является то, что она типа `Partitioned`, это значит, что одна большая таблица разделена на множестве маленьких с группировкой данных по месяцам. Данная особенность накладывает ряд ограничений на загрузку данных из сервиса: требуется запрашивать, обрабатывать только те данные, что принадлежат определённому временному отрезку, т.к. даже для извлечения одной строки требуется обработка, загрузка данных за весь временный отрезок. В среднем объем данных за 1 месяц занимает ~600 Мб на диске.

Для работы с API BigQuery необходимо создать сервисный аккаунт и скачать ключ доступа, путь до ключа н

а файловой системе необходимо указать в переменной окружения `GOOGLE_APPLICATION_CREDENTIALS`.

Данные были загружены и сохранены с помощью кода представленного в Приложении Б, многопоточная или асинхронная загрузка данных не использовалась из-за требований к оперативной памяти, так как 1 месяц данных требует ~2 Гб оперативной памяти перед сохранением данных.

```
,block_number,block_timestamp,block_timestamp_month,input_count,output_count,input_value,output_value
0,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,1,1026761.000000000,1014761.000000000
1,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,3,1435766763.000000000,1435664892.000000000
2,520650,2018-05-01 00:04:33+00:00,2018-05-01,2,2,1121471.000000000,1116039.000000000
3,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,2,10124954.000000000,10119954.000000000
4,520650,2018-05-01 00:04:33+00:00,2018-05-01,2,2,222687.000000000,218788.000000000
5,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,2,341488255.000000000,341485039.000000000
6,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,2,11120290.000000000,11115124.000000000
7,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,2,1242947.000000000,1241817.000000000
8,520650,2018-05-01 00:04:33+00:00,2018-05-01,1,1,4806890.000000000,4805779.000000000
```

Рисунок 9 - Пример извлеченных данных

Данные обрабатывались с помощью методов спискового включения (List comprehension), так как позволяют быстро выполнять операции без встроенных методов векторизации библиотек pandas, numpy и гораздо проще в использовании чем Cython. С помощью перебора транзакций были сформированы дневные матрицы цепочек, код обработки данных представлен в Приложении В. Цепочки с количеством входов или выходов $N > 20$ считались как $N = 20$.

В Приложении Г представлены функции для анализа цепочек: построение тепловых карт и график соотношения количества экстремальных цепочек к общему количеству цепочек.

2.2 Исходные данные

Для проверки наших дальнейших моделей возьмем актуальные данные за 2022 год, данные за 2022 год разделим на две группы: данные для обучения 80%, данные для прогнозирования 20%. На рисунках 10 и 11 мы можем увидеть отношение экстремальных цепочек, в которых количество входов или выходов

превышает 20 к общему количеству цепочек за день и день, когда это отношение было максимальным в виде тепловой карты цепочек.

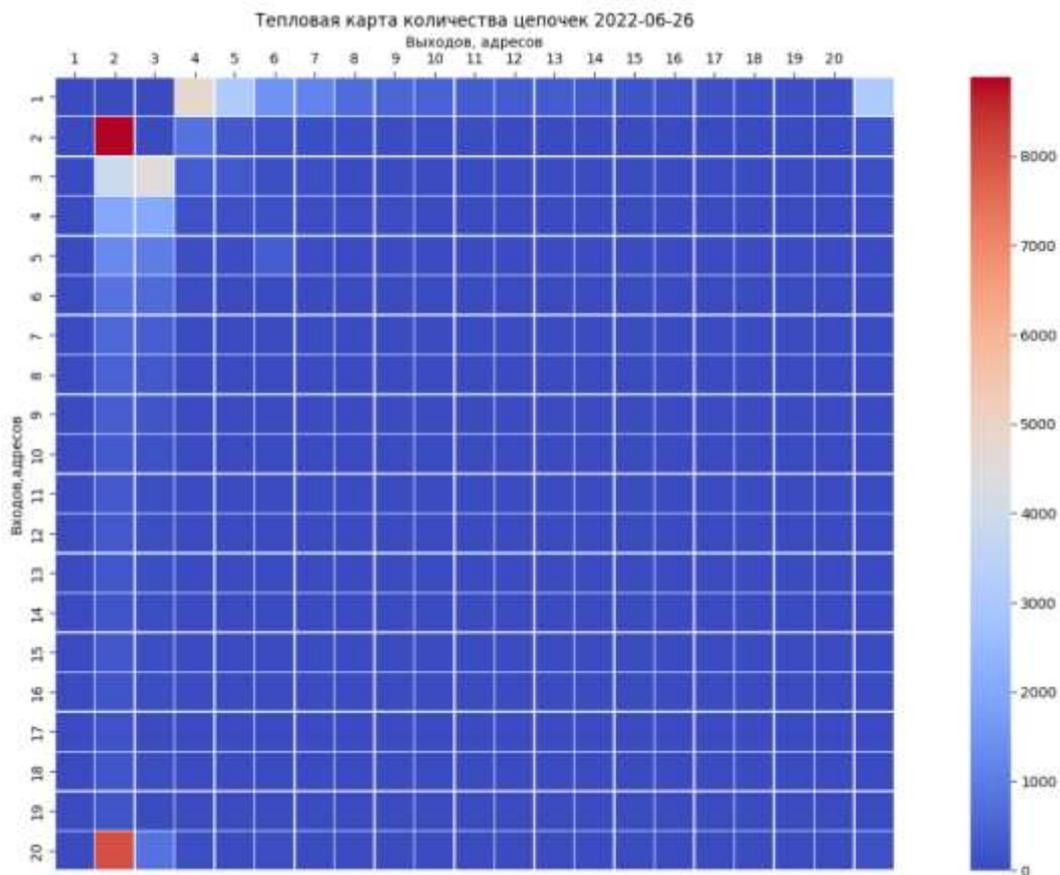


Рисунок 10 - Тепловая карта цепочек за 2022-06-26

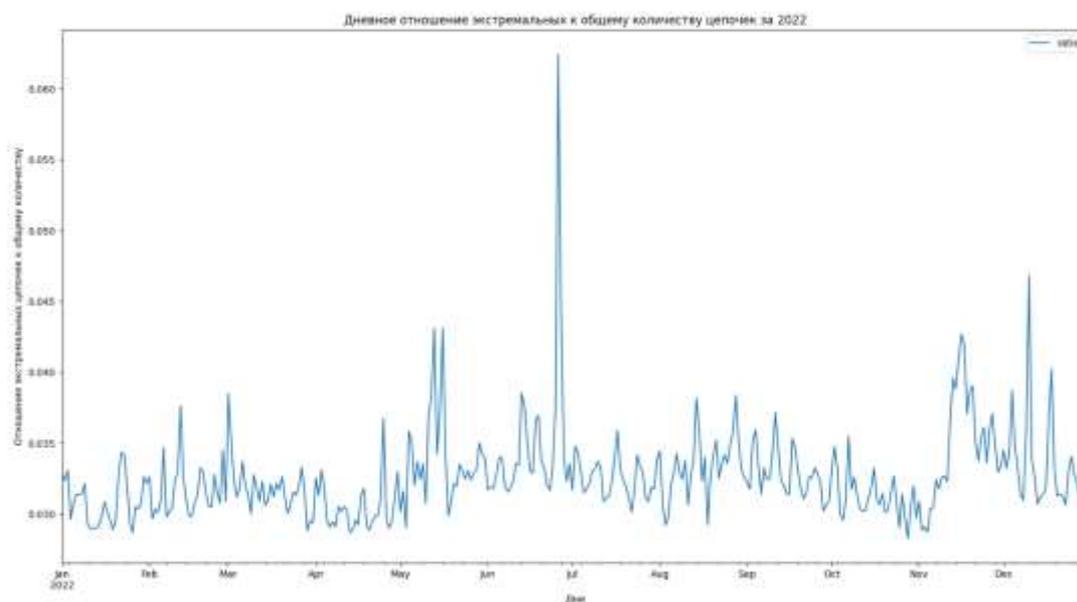


Рисунок 11 - Отношение экстремальных к общему количеству цепочек за 2022

2.3 Модель прогнозирования ARIMA-GARCH

Если в наших данных присутствует сезонная составляющая, то мы используем сезонную модель ARIMA (SARIMA). В этом случае у нас есть другой набор параметров: P , D и Q , которые описывают те же ассоциации, что и p , d и q , но соответствуют сезонным компонентам модели.

Декомпозиция временных рядов - статистическая задача, которая разбивает временной ряд на несколько компонентов, каждый из которых представляет одну из основных категорий шаблонов. С помощью статистических моделей мы сможем увидеть трендовые, сезонные и остаточные компоненты наших данных.

Для декомпозиции есть 2 модели: аддитивная и мультипликативная. Мы можем использовать аддитивную модель, когда кажется, что тренд более линейный, а сезонность и компоненты тренда кажутся постоянными во времени. Мультипликативная модель более уместна, когда мы увеличиваем или убываем с нелинейной скоростью.

Стоимость биткоина уменьшается с более высокой скоростью, чем просто линейная и поэтому нам нужно использовать мультипликативную модель для декомпозиции, в этом мы можем убедиться на рисунке 12 в верхнем графике представлена стоимость биткоина за 2022 год.

Поскольку мы используем дневные котировки, то наш временной ряд имеет период 7, произведем декомпозицию с данным значением периода и построим графики полученных данных.

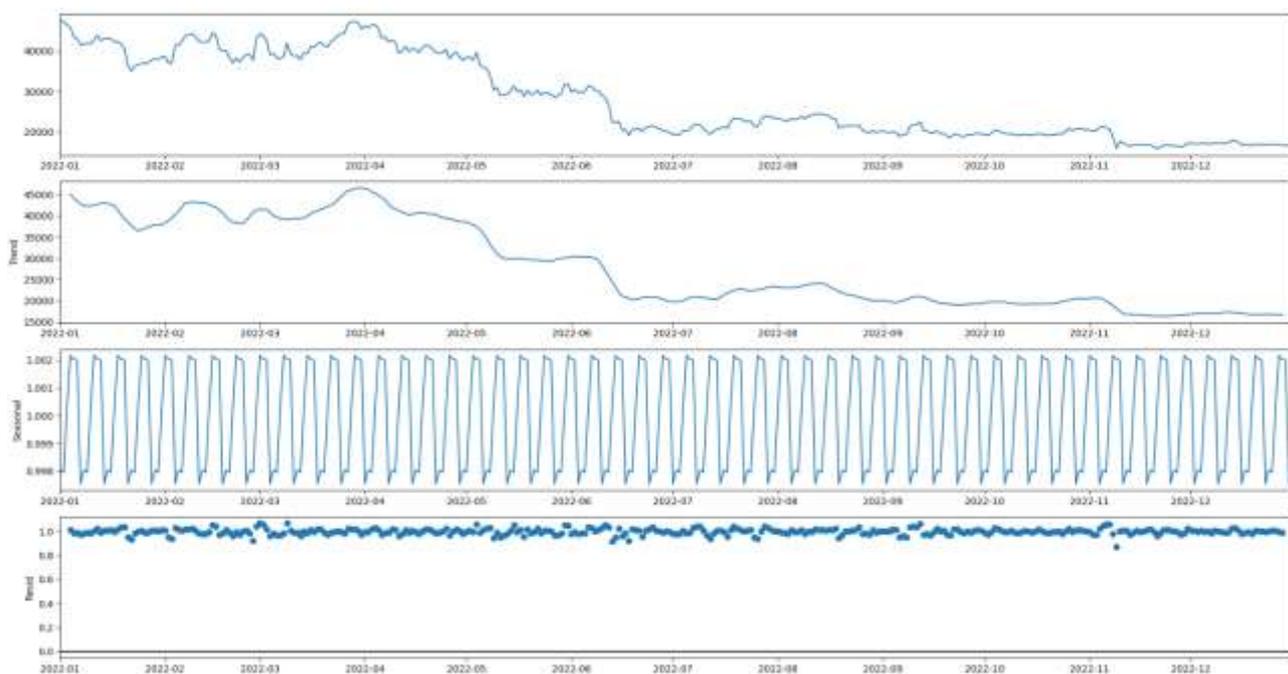


Рисунок 12 – Декомпозиция данных

На рисунке 12 мы видим сезонную составляющую данных (Seasonal), периодичность которой говорит нам о наличии сезонной составляющей.

Теперь, когда мы проанализировали данные, мы знаем, что у нас есть временной ряд с сезонной составляющей, поэтому имеет смысл использовать сезонную модель ARIMA. Для этого нам нужно будет выбрать значения p , d , q для ARIMA и значения P , D , Q для сезонного компонента.

Один из простых подходов - выполнить поиск в сетке по нескольким значениям p , d , q , P , D и Q , используя какие-то критерии производительности. Информационный критерий Акаике (AIC) - это оценка относительного качества статистических моделей для заданного набора данных. Учитывая набор моделей для данных, AIC оценивает качество каждой модели относительно каждой из других моделей.

Значение AIC позволит нам сравнить, насколько хорошо модель соответствует данным и учитывает сложность модели, поэтому модели, которые лучше подходят при использовании меньшего количества функций, получают лучшую (более низкую) оценку AIC, чем аналогичные модели, которые используют больше возможностей.

Для подбора значений p , d , q , P , D и Q по критерию AIC, мы будем использовать библиотеку `rugamid-arima` для Python, которая позволяет нам быстро выполнять поиск по сетке и создать объект модели, который можно подогнать под данные обучения [10]. Эта библиотека содержит функцию `auto_arima`, которая позволяет нам устанавливать диапазон значений p , d , q , P , D и Q , а затем подбирать модели для всех возможных комбинаций. Тогда модель сохранит комбинацию, которая вернула наилучшее значение AIC.

Поскольку модель ARIMA работает с стационарными временными рядами, а данные о стоимости биткоина не являются такими, то нам необходимо сначала их преобразовать, чтобы данные стали стационарными. Вместо стоимости биткоина на конец дня, мы будем использовать относительное изменение - число, показывающее во сколько раз изменилась некоторая величина относительно первоначального её значения.

Таблица 1. Подбор параметров для модели ARIMA

ARIMA(p , d , q)(P , D , Q)	AIC
(1,0,1)(0,1,1)	inf
(0,0,0)(0,1,0)	8685.474
(1,0,0)(1,1,0)	8155.295
(0,0,1)(0,1,1)	inf
(0,0,0)(0,1,0)	8683.475
(1,0,0)(0,1,0)	8686.759
(1,0,0)(2,1,0)	7903.839
(1,0,0)(2,1,1)	7651.144
(0,0,0)(2,1,2)	7595.654

Оптимальной моделью для наших данных является ARIMA(0,0,0)(2,1,2) со значением критерия AIC = 7595.654.

Для обучения GARCH модели возьмем остаточные ошибки нашей модели ARIMA, остаточные ошибки имеют нормальное распределение исходя из рисунка 13.

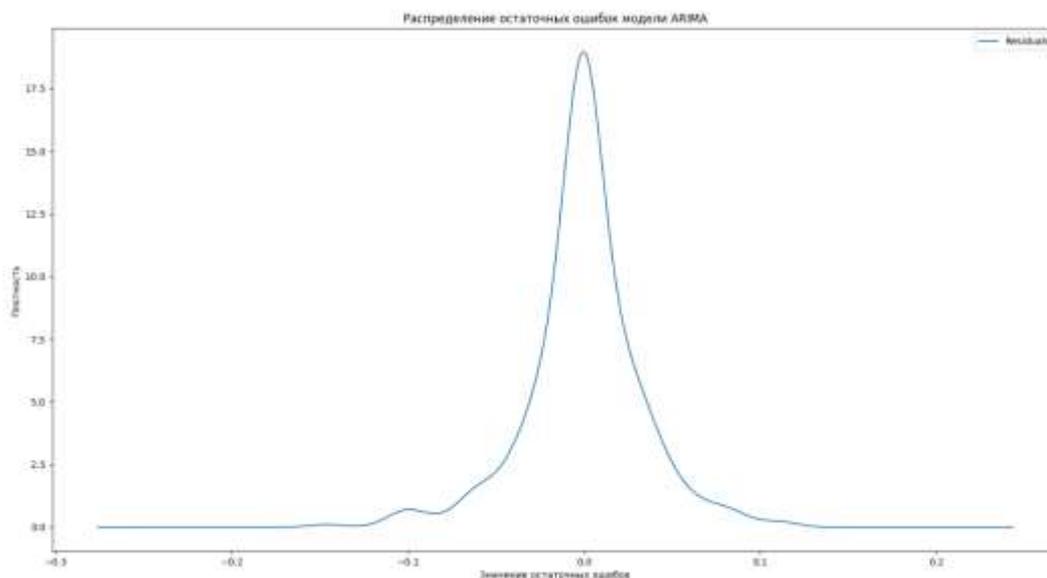


Рисунок 14 – Распределение остаточных ошибок модели ARIMA

Для оценки моделей ARIMA и ARIMA-GARCH посчитаем среднюю абсолютную процентную ошибку (MAPE) (21).

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|X_i - \hat{X}_i|}{X_i} \quad (21)$$

где n - количество прогнозируемых периодов;

X_i , – реальное значение;

\hat{X}_i – прогнозируемое значение, полученное из модели.

Оценки прогнозирования MAPE при прогнозировании стоимости биткоина на 72 дня вперед:

- ARIMA: **2.253 %**
- ARIMA-GARCH: **2.251 %**

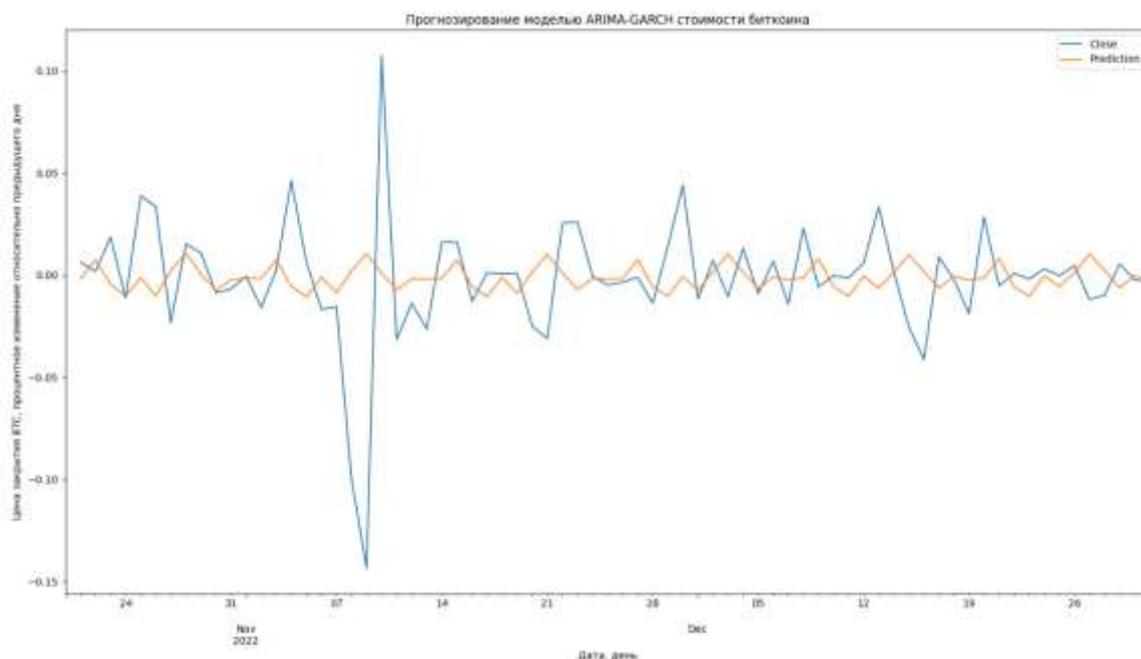


Рисунок 14 – Тестовая выборка и предсказанные значения с помощью ARIMA-GARCH

Модель ARIMA-GARCH показала наилучший результат в $2.251 / 72 * 365 \approx 11,31\%$ при пересчете на годовую среднюю абсолютную процентную ошибку.

В Приложении Д представлен код на Python который использовался для подготовки данных, проверки сезонности, поиска параметра, построение моделей, прогнозирование цен, получения оценок, построения графиков

2.4 Модель прогнозирования LSTM

Модели LSTM требуют подготовки данных и подбор параметров. В качестве подготовки данные были нормализованы и подготовлены входные матрицы, которые представляют из себя текущую цену закрытия на момент t и N цен закрытия за предыдущие дни, N равен параметру Lookback.

Использовалась простая архитектура для модели, параметры модели:

- Lookback = 14
- Batchsize = 5

- Количество эпох = 100
- LSTM слой
- Dropout 20% слой
- Dense(1) слой

Lookback отвечает за количество значений предыдущих цен закрытия, которые модель LSTM использует для прогнозирования на следующем временном шаге. Параметр Batchsize говорит о том, через сколько входных последовательностей будет произведен перерасчет весов. Batchsize позволяет нам разделить данные на более мелкие партии и обновлять веса после каждой партии вместо того, чтобы обрабатывать весь набор данных сразу. Это может сделать процесс обучения более быстрым и эффективным. Однако выбор подходящего размера партии очень важен, так как маленький размер партии может привести к замедлению сходимости, а большой размер партии может привести к переобучению.

Слой Dropout помогает избежать переобучение путем случайного отбрасывания некоторых нейронов во время обучения. Слой Dense с одним выходным элементом и без функции активации позволяет модели предсказывать числовое значение стоимости биткоина на следующий день.

Построим 2 LSTM модели:

- LSTM модель №1 – подаем на вход цену закрытия и предыдущие значения
- LSTM модель №2 – подаем на вход цену закрытия, предыдущие значения цен и матрицу цепочек за день

Оптимальное количество нейронов в LSTM слое подберем методом перебора, оценивать модели будет с помощью MAPE.

Таблица 2. MAPE оценки LSTM моделей

Нейронов \ Модель	LSTM №1, %	LSTM №2, %
8	3.4112	3.411
16	2.5161	2.485
32	2.2692	2.272

Продолжение таблицы 2

64	2.2320	2.231
128	2.8472	3.063
256	2.2685	2.125

Из таблицы 2 можем сказать, что модель LSTM №2 с 256 нейронами показала наилучший результат среди LSTM моделей, так же лучше, чем ARIMA-GARCH модель. Результат модели LSTM №2 с 256 нейронами $2.125 / 72 * 365 \approx 10,77 \%$ при пересчете на годовичную среднюю абсолютную процентную ошибку. На рисунке 15 можно увидеть реальные значения и предсказанные.



Рисунок 15 – Тестовая выборка и предсказанные значения с помощью лучшей LSTM

В Приложении Е. Ж представлен код на Python который использовался для подготовки данных, формирования скользящего окна, построения модели, обучения модели, прогнозирования цен, получения оценок, построения графиков.

Заключение

В результате выполнения выпускной квалификационной работы можно сделать основные выводы:

1. С помощью сервиса Google BigQuery организована загрузка данных помесечных транзаций биткоина (около 600 Мб данных) из блокчейна биткоина (всего около 450 Гб данных транзакций). Для увеличения скорости обработки дневные матрицы цепочек формировались с помощью методов спискового включения. Были построены тепловые карты, обнаружены все экстремальные цепочки;

2. Построены и применены для прогнозирования данных цепочек транзакций (chainlet) эконометрические модели ARIMA, ARIMA-GARCH. Кроме того, обучена, калибрована, верифицирована рекуррентная нейронная сеть с долгосрочно-краткосрочной памятью (LSTM);

3. Сравнительный анализ прогнозирования котировок биткоина различными моделями на 72 дня вперед показал, что модель LSTM по уровню средней абсолютной ошибки в процентах (MAPE) сопоставима по точности с классическими эконометрическими алгоритмами при малом числе нейронов (до 200), но становится более точной, если число нейронов превосходит 256. Повторное применение модели LSTM с 256 нейронами показало наилучший результат среди всех рассмотренных моделей с MAPE = 10.77%.

3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Введение

Основная цель данного раздела – оценить перспективность развития и планировать финансовую и коммерческую ценность конечного продукта, представленного в рамках исследовательской программы. Коммерческая ценность определяется не только наличием более высоких технических характеристик над конкурентными разработками, но и тем, насколько быстро разработчик сможет ответить на следующие вопросы – будет ли продукт востребован на рынке, какова будет его цена, каков бюджет научного исследования, какое время будет необходимо для продвижения разработанного продукта на рынок.

Данный раздел, предусматривает рассмотрение следующих задач:

- Оценка коммерческого потенциала разработки.
- Планирование научно-исследовательской работы;
- Расчет бюджета научно-исследовательской работы;
- Определение ресурсной, финансовой, бюджетной эффективности исследования.

Цель данной ВКР – разработка и проверка работоспособности алгоритма прогнозирования цен криптовалют с использованием chainlets извлеченных из блокчейна криптовалюты.

3.1 Оценка коммерческого потенциала и перспективности проведения исследований с позиции ресурсоэффективности и ресурсосбережения

3.1.1 Анализ конкурентных технических решений

В ходе исследования были рассмотрены две конкурирующие модели прогнозирования:

1) Модель ARIMA;

2) Модель LSTM.

Детальный анализ необходим, т.к. каждая модель прогнозирования имеет свои достоинства и недостатки. В таблице 3 показано сравнение конкурирующих-моделей и модель данной ВКР с точки зрения технических и экономических критериев оценки эффективности.

Таблица 3 – Сравнение конкурентных технических решений (разработок)

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б _ф	Б _{к1}	Б _{к2}	К _ф	К _{к1}	К _{к2}
1	2	3	4	5	6	7	8
Технические критерии оценки ресурсоэффективности							
1. Актуальность исследования	0,1	5	3	4	0,5	0,3	0,4
2. Точность прогнозирования	0,4	5	2	3	2,0	0,8	1,2
3. Возможность корректировки модели	0,2	4	1	2	0,8	0,2	0,4
4. Объем данных для обучения	0,05	3	5	5	0,15	0,25	0,25
5. Скорость работы	0,05	3	5	4	0,15	0,25	0,2
Экономические критерии оценки эффективности							
1. Срок выхода на рынок	0,1	3	5	4	0,3	0,5	0,4
2. Финансирование научной разработки	0,1	3	5	4	0,3	0,5	0,4
Итого	1	32	26	26	4,7	2,8	3,25

Расчет конкурентоспособности, на примере стабильности срабатывания, определяется по формуле:

$$K = \sum B_i \cdot B_i = 4,7$$

где K – конкурентоспособность проекта; B_i – вес показателя (в долях единицы); B_i – балл показателя.

Проведенный анализ конкурентных технических решений показал, что исследование является наиболее актуальным и перспективным, имеет конкурентоспособность.

3.1.2 SWOT-анализ

Для исследования внешней и внутренней среды проекта, в этой работе проведен SWOT-анализ с детальной оценкой сильных и слабых сторон исследовательского проекта, а также его возможностей и угроз.

Первый этап, составляется матрица SWOT, в которую описаны слабые и сильные стороны проекта и выявленные возможности и угрозы для реализации проекта, которые проявились или могут появиться в его внешней среде, приведены в таблице 4.

Таблица 4 – Матрица SWOT-анализа

Сильные стороны	Слабые стороны
С1. Инновационный подход с использованием информации из блокчейна для прогнозирования цен криптовалют.	Сл1. Высокие требования к вычислительным ресурсам.
С2. Использование модели с информацией о цепочках транзакций позволяет включить в анализ большое количество разнообразной информации, что может привести к более точным прогнозам цен на криптовалюты.	Сл2. Высокие требования ко времени обучения модели.
Возможности	Угрозы
В1. Использование LSTM позволяет менять конфигурацию модели и производить тонкую настройку.	У1. Плохая точность прогнозирования разработанной модели.
В2. Проект имеет потенциал привлечения широкой аудитории инвесторов из криптовалютой сферы.	У2. Появление новых моделей с лучшей точностью прогнозирования.

Продолжение таблицы 4

В3. Возможность расширения деятельности на другие криптовалюты и оптимизации под каждую из них.	
---	--

На втором этапе на основании матрицы SWOT строятся интерактивные матрицы возможностей и угроз, позволяющие оценить эффективность проекта, а также надежность его реализации. Соотношения параметров представлены в таблицах 5–8.

Таблица 5 – Интерактивная матрица проекта «Возможности проекта и сильные стороны»

Сильные стороны проекта			
Возможности проекта		С1	С2
	В1	0	+
	В2	+	+
	В3	+	+

Таблица 6 – Интерактивная матрица проекта «Возможности проекта и слабые стороны»

Слабые стороны проекта			
Возможности проекта		Сл1	Сл2
	В1	-	-
	В2	0	0
	В3	-	-

Таблица 7 – Интерактивная матрица проекта «Угрозы проекта и сильные стороны»

Сильные стороны проекта			
Угрозы проекта		С1	С2
	У1	+	0
	У2	+	0

Таблица 8 – Интерактивная матрица проекта «Угрозы проекта и слабые стороны»

Слабые стороны проекта			
Угрозы проекта		Сл1	Сл2
	У1	-	-
	У2	+	+

Результаты анализа представлены в итоговую таблицу 9.

Таблица 9 – Итоговая таблица SWOT-анализа

	<p>Сильные стороны научно-исследовательского проекта</p> <p>С1. Инновационный подход с использованием информации из блокчейна для прогнозирования цен криптовалют.</p> <p>С2. Использование модели с информацией о цепочках транзакций позволяет включить в анализ большое количество разнообразной информации, что может привести к более точным прогнозам цен на криптовалюты</p> <p>.</p>	<p>Слабые стороны научно-исследовательского проекта</p> <p>Сл1. Высокие требования к вычислительным ресурсам</p> <p>Сл2. Высокие требования ко времени обучения модели</p>
<p>Возможности</p> <p>В1. Использование LSTM позволяет менять конфигурацию модели и производить тонкую настройку.</p> <p>В2. Проект имеет потенциал привлечения широкой аудитории инвесторов из криптовалютной сферы.</p> <p>В3. Возможность расширения деятельности на другие рынки, такие как фондовый рынок и рынок форекс.</p>	<p>Направления развития</p> <p>В1В2С2. Позволит получить дополнительное финансирование разработки для более глубокой проработки модели и улучшения прогнозирования.</p> <p>В3С1С2. Позволяет делать прогнозы по любой криптовалюте и получить возможность улучшить показатели модели за счет дополнительных данных из других блокчейнов.</p> <p>.</p>	<p>Сдерживающие факторы</p> <p>В1В3Сл1Сл2. Сложность модели не позволит быстро и качественно построить модель без достаточного финансирования.</p> <p>В2Сл1Сл2. Дополнительное финансирование может помочь убрать слабые стороны, но так же потенциальных инвесторов может пугать сложность разработки.</p>

Продолжение таблицы 9

<p>Угрозы У1. Плохая точность прогнозирования разработанной модели. У2. Появление новых моделей с лучшей точностью прогнозирования.</p>	<p>Угрозы развития У1У2С1. Из-за сложности модели вероятность не удовлетворительной работы модели выше. Появление более производительного алгоритма может лишить разработку актуальности.</p>	<p>Уязвимости: У1Сл1Сл2. Улучшение показателей модели требует очень много ресурсов. У2Сл1Сл2. Конкурентным моделям требуется так же много ресурсов, данный фактор дает нам преимущество</p>
--	---	--

В результате SWOT-анализа показано, что на преимущества разрабатываемой технологии преобладают над ее недостатками. Данные недостатки, которые на данный момент на практике не устранены, но в теории уже есть возможности для их устранения. Результаты анализа учтены в дальнейшей научно-исследовательской разработке.

3.2 Планирование научно-исследовательских работ

3.2.1 Структура работ в рамках научного исследования

Планирование комплекса научно-исследовательских работ осуществляется в порядке:

- определение структуры работ в рамках научного исследования;
- определение количества исполнителей для каждой из работ;
- установление продолжительности работ;
- построение графика проведения научных исследований.

Для оптимизации работ удобно использовать классический метод линейного планирования и управления.

Результатом такого планирования является составление линейного графика выполнения всех работ. Порядок этапов работ и распределение исполнителей для данной научно-исследовательской работы, приведен в таблице 10.

Таблица 10 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Разработка технического задания	1	Составление и утверждение технического задания, утверждение плана-графика	Научный руководитель
	2	Календарное планирование выполнения работ	Инженер, научный руководитель
Выбор способа решения поставленной задачи	3	Обзор научной литературы	Инженер
	4	Сбор данных для обучения	Инженер
Теоретические и экспериментальные исследования	5	Обработка и анализ исходных данных	Инженер, научный руководитель
	6	Проведение эксперимента	Инженер
	7	Оптимизация модели	Инженер
Обобщение и оценка результатов	8	Обработка результатов	Инженер
	9	Оценка правильности полученных результатов	Инженер, Научный руководитель
Оформление отчета по НИР (комплекта документации по ОКР)	10	Составление пояснительной записки	Инженер

3.2.2 Определение трудоемкости выполнения работ и разработка графика проведения

При проведении научных исследований основную часть стоимости разработки составляют трудовые затраты, поэтому определение трудоемкости проводимых работ является важным этапом составления сметы.

Для определения ожидаемого (среднего) значения трудоемкости использована следующая формула:

$$t_{\text{ож}i} = \frac{3t_{\text{min}i} + 2t_{\text{max}i}}{5}, \quad (22)$$

где $t_{\text{ож}i}$ – ожидаемая трудоемкость выполнения i -ой работы, человеко-дни;

$t_{\text{min}i}$ – минимально возможная трудоемкость выполнения заданной i -ой работы, человеко-дни;

$t_{\text{max}i}$ – максимально возможная трудоемкость выполнения заданной i -ой работы, человеко-дни.

Зная величину ожидаемой трудоемкости, можно определить продолжительность каждой i -ой работы в рабочих днях T_{pi} , при этом учитывается параллельность выполнения работ разными исполнителями. Данный расчёт позволяет определить величину заработной платы.

$$T_{pi} = \frac{t_{\text{ож}i}}{Ч_i}, \quad (23)$$

где T_{pi} – продолжительность одной работы, рабочие дни;

$t_{\text{ож}i}$ – ожидаемая трудоемкость выполнения одной работы, человеко-дни;

$Ч_i$ – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

Для перевода длительности каждого этапа из рабочих в календарные дни, необходимо воспользоваться формулой (24):

$$T_{ki.инж} = T_{pi} \cdot k_{кал}, \quad (24)$$

где T_{ki} – продолжительность выполнения i -й работы в календарных днях;
 T_{pi} – продолжительность выполнения i -й работы в рабочих днях;
 $k_{кал}$ – календарный коэффициент.

Календарный коэффициент определяется по формуле:

$$k_{кал.инж} = \frac{T_{кал}}{T_{кал} - T_{вых} - T_{пр}} = \frac{365}{365 - 104 - 14} = 1,48 \quad (25)$$

где $T_{кал}$ – общее количество календарных дней в году; $T_{вых}$ – общее количество выходных дней в году; $T_{пр}$ – общее количество праздничных дней в году (2023 год).

Расчеты временных показателей проведения научного исследования обобщены в таблице 11.

Таблица 11 – Временные показатели проведения научного исследования

Название работы	Трудоёмкость работ						Длительность работ в рабочих днях T_{pi}	Длительность работ в календарных днях T_{ki}
	t_{min} , чел-дни		t_{max} , чел-дни		$t_{ожi}$, чел-дни			
	Исп.1	Исп.2	Исп.1	Исп.2	Исп.1	Исп.2		
1	2	3	4	5	6	7	8	9
Составление и утверждение технического задания, утверждение плана-графика	2	-	4	-	2,8	-	2,8	5
Календарное планирование выполнения работ	1	3	3	4	1,8	3,4	2,6	4
Обзор научной литературы	-	6	-	10	-	7,6	7,6	12
Сбор данных для обучения	-	3	-	5	-	3,8	3,8	6
Обработка и анализ исходных данных	2	6	4	8	2,8	6,8	4,8	8
Проведение эксперимента	-	5	-	7	-	5,8	5,8	9

Продолжение таблицы 11

Оптимизация модели	-	15	-	20	-	17	17	26
Обработка результатов	-	10	-	15	-	12	12	18
Оценка правильности полученных результатов	2	3	4	5	2,8	3,8	3,3	5
Составление пояснительной записки	-	8		10	-	8,8	8,8	14
Итого:	7	59	15	84	10,2	69	68,5	107

Примечание: Исп. 1 – научный руководитель, Исп. 2 – инженер.

На основе таблицы составлен календарный план-график выполнения проекта с использованием диаграммы Ганта (таблица 12).

Таблица 12 – Диаграмма Ганта

№	Вид работ	Исп	T _{кi} кал. дн.	Продолжительность работ													
				февр			март			апр			май				
				1	2	3	1	2	3	1	2	3	1	2	3		
1	Составление и утверждение технического задания, утверждение плана-графика	Исп1	5														
2	Календарное планирование выполнения работ	Исп1 Исп2	4														
3	Обзор научной литературы	Исп2	12														
4	Сбор данных для обучения	Исп2	6														
5	Обработка и анализ исходных данных	Исп1 Исп2	8														
6	Проведение эксперимента	Исп2	9														
7	Оптимизация модели	Исп2	26														
8	Обработка результатов	Исп2	18														
9	Оценка правильности полученных результатов	Исп1 Исп2	5														
10	Составление пояснительной записки	Исп2	14														

Примечание:

– Исп. 1 (научный руководитель), – Исп. 2 (инженер)

3.3 Бюджет научно-технического исследования

При планировании бюджета научно-технического исследования учитывались все виды расходов, связанных с его выполнением. В этой работе использовать следующую группировку затрат по следующим статьям:

- материальные затраты научно-исследовательской работы (НИР);
- затраты на специальное оборудование для экспериментальных работ;
- основная заработная плата исполнителей темы;
- дополнительная заработная плата исполнителей темы;
- отчисления во внебюджетные фонды (страховые отчисления);
- накладные расходы НИР.

3.3.1 Расчет материальных затрат научно-технического исследования

Материальные затраты, необходимые для разработки, представлены в таблице 13.

Таблица 13 – Материальные затраты

Наименование материалов	Цена за ед., руб.	Кол-во, ед.	Сумма, руб.
Комплекс канцелярских принадлежностей	340	4	1 200
Картридж для лазерного принтера	3 490	1	3 490
Итого затраты:			8 290

Цены приняты на основании прайс-листа поставщика материалов:
<https://papyrus-tomsk.ru>, <https://mvideo.ru>.

3.3.2 Расчет амортизации специального оборудования

Расчет сводится к определению амортизационных отчислений, так как оборудование было приобретено до начала выполнения данной работы и эксплуатировалось ранее, поэтому при расчете затрат на оборудовании учитываем только рабочие дни по данной теме.

Расчет амортизации проводится следующим образом:

Норма амортизации: рассчитывается по формуле:

$$H_A = \frac{1}{n}, \quad (26)$$

где n – срок полезного использования в количестве лет.

Амортизация оборудования рассчитывается по формуле:

$$A = \frac{H_A I}{12} \cdot m, \quad (27)$$

где I – итоговая сумма, тыс. руб.; m – время использования, мес.

При выполнении научно-исследовательского проекта использовался ПЭВМ - Asus. Срок полезного использования данного ноутбука по паспорту составляет 3 года.

Таблица 14 – Затраты на оборудование

№	Наименование оборудования	Кол-во, шт.	Срок полезного использования, лет	Цены единицы оборудования, тыс. руб.	Общая стоимость оборудования, тыс. руб.
1	ПЭВМ	1	3	70	70
Итого				70 тыс. руб.	

Расчет амортизации проводится следующим образом:

Норма амортизации определяется по следующей формуле:

$$H_A = \frac{1}{n},$$

где n – срок полезного использования в годах.

Амортизация определяется по следующей формуле:

$$A = \frac{H_A I}{12} \cdot m,$$

где I – итоговая сумма, тыс. руб.;

m – время использования, мес.

Рассчитаем норму амортизации для ноутбука, с учётом того, что срок полезного использования составляет 3 года:

$$H_A = \frac{1}{n} = \frac{1}{3} = 0,33.$$

Общую сумму амортизационных отчислений находим следующим образом:

$$A = \frac{H_A I}{12} \cdot m = \frac{0,33 \cdot 70000}{12} \cdot 4 = 7700 \text{ руб.}$$

3.3.3 Основная заработная плата исполнителей темы

В данном разделе рассчитывается заработная плата инженера и руководителя, помимо этого необходимо рассчитать расходы по заработной плате, определяемые трудоемкостью проекта и действующей системой оклада.

Основная заработная плата $Z_{осн}$ одного работника рассчитывается по следующей формуле:

$$Z_{осн} = Z_{дн} \cdot T_p, \quad (28)$$

где $Z_{дн}$ – среднедневная заработная плата, руб.; T_p – продолжительность работ, выполняемых работником, раб.дн. (таблица 30).

Среднедневная заработная плата рассчитывается по формуле:

Для шестидневной рабочей недели (рабочая неделя руководителя):

$$Z_{дн} = \frac{Z_m \cdot M}{F_0} = \frac{51285 \cdot 10,3}{246} = 2147,3 \text{ руб.}, \quad (29)$$

где z_m – месячный должностной оклад работника, руб.; F_0 – действительный годовой фонд рабочего времени научно-технического персонала, раб. дней; M – количество месяцев работы без отпуска в течение года:

– при отпуске в 28 раб. дня – $M = 11,2$ месяца, 5-дневная рабочая неделя;

– при отпуске в 56 раб. дней – $M = 10,3$ месяца, 6-дневная рабочая неделя.

Для пятидневной рабочей недели (рабочая неделя инженера):

$$z_{он} = \frac{z_m \cdot M}{F_0} = \frac{33150 \cdot 11,2}{213} = 1743,1 \text{ руб.} \quad (30)$$

Должностной оклад работника за месяц:

– для руководителя:

$$z_m = z_{мс} \cdot (1 + k_{np} + k_0) k_p = 26300 \cdot (1 + 0,3 + 0,2) \cdot 1,3 = 51285 \text{ руб.} \quad (31)$$

– для инженера:

$$z_m = z_{мс} \cdot (1 + k_{np} + k_0) k_p = 17000 \cdot (1 + 0,3 + 0,2) \cdot 1,3 = 33150 \text{ руб.}, \quad (32)$$

где $z_{мс}$ – заработная плата, согласно тарифной ставке, руб.; k_{np} – премиальный коэффициент, равен 0,3; k_0 – коэффициент доплат и надбавок, равен 0,2; k_p – районный коэффициент, равен 1,5 (для г. Томска).

Таблица 15 – Баланс рабочего времени исполнителей

Показатели рабочего времени	Руководитель	Инженер
Календарное число дней	365	365
Количество нерабочих дней	52/14	104/14
- выходные дни		
- праздничные дни		
Потери рабочего времени	48/5	24/10
- отпуск		
- невыходы по болезни		
Действительный годовой фонд рабочего времени	246	213

Таблица 16 – Расчет основной заработной платы исполнителей

Исполнители НИ	$Z_{мс}, руб$	$k_{пр}$	$k_{д}$	k_p	$Z_{м}, руб$	$Z_{он}, руб$	$T_p, раб.дн.$	$Z_{осн}, руб$
Руководитель	26300	0,3	0,2	1,3	51285	2147,3	10,2	21902,5
Инженер	17000	0,3	0,2	1,3	33150	1743,1	69	120273,9
Итого:								142176,4

Дополнительная заработная плата определяется по формуле:

– для руководителя:

$$Z_{доп} = \kappa_{доп} \cdot Z_{осн} = 0,15 \cdot 21902,5 = 3285,4 \quad (33)$$

– для инженера:

$$Z_{доп} = \kappa_{доп} \cdot Z_{осн} = 0,15 \cdot 120273,9 = 18041,1 \quad (34)$$

где $\kappa_{доп}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимаем равным 0,15).

3.3.4 Отчисления во внебюджетные фонды (страховые отчисления)

Отчисления во внебюджетные фонды определяется по формуле:

– для руководителя:

$$Z_{внеб} = \kappa_{внеб}(Z_{доп} + Z_{осн}) = 0,3 \cdot (21902,5 + 3285,4) = 7556,4 \text{ руб} \quad (35)$$

– для инженера:

$$Z_{внеб} = \kappa_{внеб}(Z_{доп} + Z_{осн}) = 0,3 \cdot (120273,9 + 18041,1) = 41494,5 \text{ руб} \quad (36)$$

где $\kappa_{внеб}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд ОМС и социальное страхование). Общая ставка взносов составляет в 2023 году – 30% (ст. 425, 426 НК РФ).

3.3.5 Накладные расходы

Накладные расходы включают в себя следующие расходы: печать ксерокопирование материалов исследования, оплата услуг связи и т.д. Сумма 5

статьи затрат, рассчитанных выше, приведена в таблице ниже и используются для расчета накладных расходов.

Таблица 17 – Группировка затрат по статьям

Статьи					
1	2	3	4	5	6
Материальные затраты	Амортизация	Основная заработная плата	Дополнительная заработная плата	Отчисления на социальные нужды	Итого без накладных расходов
8290	7700	142176,4	21326,5	49050,9	228543,7

Величина накладных расходов определяется по формуле (37):

$$Z_{накл} = (\text{сумма статей } 1 \div 5) \cdot k_{пр}, \quad (37)$$

где $k_{пр}$ – коэффициент, учитывающий накладные расходы. Величина коэффициента принимается равной 0,2.

3.3.6 Бюджет НИР

На основании полученных данных по отдельным статьям затрат составляется калькуляция плановой себестоимости НИ «Создание и использование chainlet для прогнозирования цен криптовалют» по форме, приведенной в таблице 18. В таблице также представлено определение бюджета затрат двух конкурирующих научно-исследовательских проектов.

Таблица 18 – Группировка затрат по статьям

№	Наименование статьи	Сумма, руб.			Примечание
		Текущий Проект	Исп.2	Исп.3	
1	Материальные затраты	8290	10626,3	15526,3	Пункт 4.3.1
2	Амортизация	7700	12295,8	14345,0	Пункт 4.3.2
3	Затраты по основной заработной плате	142176,4	198391,0	218391,0	Пункт 4.3.3

Продолжение таблицы 18

4	Затраты по дополнительной заработной плате	21326,5	29758,7	32758,7	Пункт 4.3.3
5	Отчисления во внебюджетные фонды	49050,9	68444,9	75344,9	Пункт 4.3.4
6	Накладные расходы	45708,7	63903,3	71273,2	Пункт 4.3.5
Бюджет затрат НИР		274252,5	384820	427639	Сумма ст. 1- 6

3.4 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Для определения эффективности исследования рассчитан интегральный показатель эффективности научного исследования путем определения интегральных показателей финансовой эффективности и ресурсоэффективности.

Интегральный показатель финансовой эффективности научного исследования получен в процессе оценки бюджета затрат трех вариантов исполнения научного исследования. Для этого наибольший интегральный показатель реализации технической задачи принят за базу расчета (как знаменатель), с которым соотносятся финансовые значения по всем вариантам исполнения.

В качестве аналогов данной НИР рассмотрены:

- 1) Модель ARIMA;
- 2) модель LSTM.

Интегральный финансовый показатель разработки рассчитывается как:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{\text{pi}}}{\Phi_{\text{max}}}, \quad (38)$$

где $I_{\text{финр}}^{\text{исп.}i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{\max} – максимальная стоимость исполнения.

$\Phi_{\text{текущ.проект}} = 274252,5$ руб, $\Phi_{\text{исп.1}} = 384820$ руб, $\Phi_{\text{исп.2}} = 427639$ руб.

$$I_{\text{фин.р}}^{\text{исп1}} = \frac{274252,5}{427639} = 0,6;$$

$$I_{\text{фин.р}}^{\text{исп2}} = \frac{384820}{427639} = 0,9;$$

$$I_{\text{фин.р}}^{\text{исп3}} = \frac{427639}{427639} = 1$$

В результате расчета консолидированных финансовых показателей по трем вариантам разработки вариант 1 (текущий проект) с меньшим перевесом признан считается более приемлемым с точки зрения финансовой эффективности.

Интегральный показатель ресурсоэффективности вариантов выполнения НИР (I_{pi}) определен путем сравнительной оценки их характеристик, распределенных с учетом весового коэффициента каждого параметра (таблица 19).

Таблица 19 – Сравнительная оценка характеристик вариантов НИР

Объект исследования Критерии	Весовой коэффициент параметра	Текущий проект	Исп.2	Исп.3
1. Скорость обучения	0,15	3	5	4
2. Объем дополнительной информации	0,2	5	4	4
3. Оптимизация	0,2	5	3	4
4. Точность прогнозирования	0,3	5	3	4
5. Удобство в эксплуатации	0,15	5	4	4
ИТОГО	1	4,7	3,65	4

Расчет интегрального показателя для разрабатываемого проекта:

$$I_{p-\text{тек.пр.}} = 0,15 \cdot 3 + 0,2 \cdot 5 + 0,2 \cdot 5 + 0,3 \cdot 5 + 0,15 \cdot 5 = 4,7;$$

$$I_{p-\text{исп2}} = 0,15 \cdot 5 + 0,2 \cdot 4 + 0,2 \cdot 3 + 0,3 \cdot 3 + 0,15 \cdot 4 = 3,65;$$

$$I_{p-исп3}=0,15\cdot4+0,2\cdot4+0,2\cdot4+0,3\cdot4+0,15\cdot4=4.$$

Интегральный показатель эффективности вариантов исполнения разработки вычисляется на основании показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{исп.i} = \frac{I_{p-исп.i}}{I_{финр}}. \quad (39)$$

$$I_{исп1} = \frac{I_{p-тек.пр.}}{I_{фин.р}} = \frac{4,7}{0,6} = 7,83$$

$$I_{исп2} = \frac{I_{p-исп2}}{I_{фин.р}} = \frac{3,65}{0,9} = 4,06;$$

$$I_{исп3} = \frac{I_{p-исп3}}{I_{фин.р}} = \frac{4}{1} = 4.$$

Далее интегральные показатели эффективности каждого варианта НИР сравнивались с интегральными показателями эффективности других вариантов с целью определения сравнительной эффективности проекта (таблица 20).

Таблица 20 – Сравнительная эффективность разработки

№ п/п	Показатели	Текущий проект	Исп.2	Исп.3
1	Интегральный финансовый показатель разработки	0,6	0,9	1
2	Интегральный показатель ресурсоэффективности разработки	4,7	3,65	4
3	Интегральный показатель эффективности	7,83	4,06	4
4	Сравнительная эффективность вариантов исполнения	1	0,52	0,51

Сравнение среднего интегрального показателя сопоставляемых вариантов позволило сделать вывод о том, что наиболее финансово- и ресурсоэффективным является вариант 1 (текущий проект). Наш проект является более эффективным по сравнению с конкурентами.

Выводы по разделу

В результате выполнения целей раздела можно сделать следующие выводы:

1. Результатом анализа конкурентных технических решений является выбор одного из вариантов реализации НИР как наиболее подходящего и оптимального по сравнению с другими.

2. В ходе планирования для руководителя и инженера был разработан график реализации этапа работ, который позволяет оценивать и планировать рабочее время исполнителей. Определено следующее: общее количество календарных дней для выполнения работ составляет 107 дней; общее количество дней, в течение которых работал инженер, составляет 69 дней; общее количество дней, в течение которых работал руководитель, составляет 11 дней;

3. Для оценки затрат на реализацию проекта разработан проектный бюджет, который составляет 274252,5 руб;

4. Результат оценки эффективности ИР показывает следующие выводы:

1) значение интегрального финансового показателя ИР составляет 0,6, что является показателем того, что ИР является финансово выгодной по сравнению с аналогами;

2) значение интегрального показателя ресурсоэффективности ИР составляет 4,7, по сравнению с 3,65 и 4;

3) значение интегрального показателя эффективности ИР составляет 7,83, по сравнению с 4,06 и 4, и является наиболее высоким, что означает, что техническое решение, рассматриваемое в ИР, является наиболее эффективным вариантом исполнения.

4. Социальная ответственность

Социальная ответственность и охрана труда являются важнейшими составляющими любой деятельности, в особенности производственной, т.к. непосредственно связаны со здоровьем и жизнью человека.

Социальная ответственность - это сознательное отношение субъекта социальной деятельности к требованиям социальной необходимости, гражданского долга, социальных задач и, норм и ценностей, понимание осуществляемой деятельности для определенных социальных групп и личностей, для социального прогресса общества.

Актуальностью исследования «Создание и использование chainlet для прогнозирования цен криптовалют» является то, что в разрабатываемой модели будет учитываться информация из блокчейна о транзакциях. Криптовалюты позволяют получить гораздо больше информации о рынке, в отличие от, фондового рынка, где нам не доступна информация о передвижении средств.

4.1 Правовые и организационные вопросы обеспечения безопасности

4.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства

Согласно ТК РФ, N 197 -ФЗ работник аудитории 220, 11 корпуса ТПУ

имеет право на:

- рабочее место, соответствующее требованиям охраны труда;
- обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний в соответствии с федеральным законом;
- отказ от выполнения работ в случае возникновения опасности для его жизни и здоровья вследствие нарушения требований охраны труда, за исключением случаев, предусмотренных федеральными законами, до

устранения такой опасности;• обеспечение средствами индивидуальной и коллективной защиты в соответствии с требованиями охраны труда за счет средств работодателя;

- внеочередной медицинский осмотр в соответствии с медицинскими рекомендациями с сохранением за ним места работы (должности) и среднего заработка во время прохождения указанного медицинского осмотра;

Рабочее место в аудитории 220, 1 корпуса ТПУ должно соответствовать требованиям ГОСТ 12.2.032 - 78. Оно должно занимать площадь не менее 4,5 м², высота помещения должна быть не менее 4 м, а объем - не менее 20 м³ на одного человека. Высота над уровнем пола рабочей поверхности, за которой работает оператор, должна составлять 720 мм. Оптимальные размеры поверхности стола 1600 x 1000 кв. мм. Под столом должно иметься пространство для ног с размерами по глубине 650 мм. Рабочий стол должен также иметь подставку для ног, расположенную под углом 15° к поверхности стола. Длина подставки 400 мм, ширина - 350 мм. Удаленность клавиатуры от края стола должна быть не более 300 мм, что обеспечит удобную опору для предплечий. Расстояние между глазами оператора и экраном видеодисплея должно составлять 40 - 80 см. Рабочее место должно быть скомпоновано так, чтобы все операции работника выполнялись в пределах зоны досягаемости моторного поля в вертикальной плоскости (Рисунок 16) и в горизонтальной плоскости (Рисунок 17).

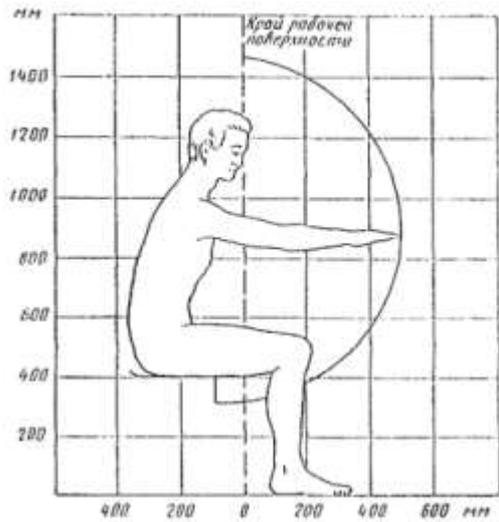


Рисунок 16 – Зона досягаемости моторного поля в вертикальной плоскости

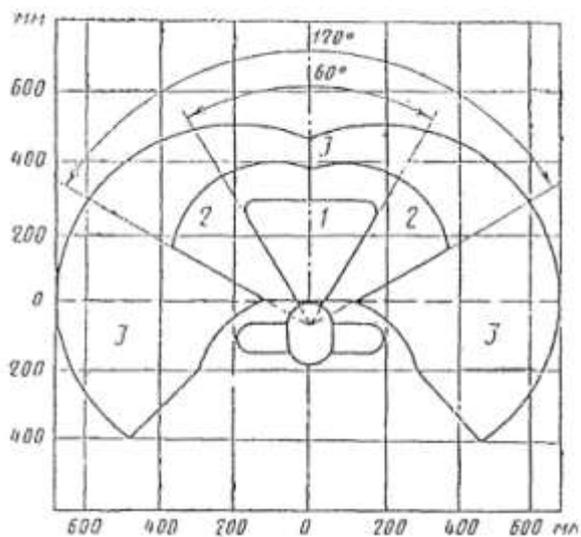


Рисунок 17 – Зона досягаемости моторного поля в горизонтальной плоскости.

- 1 - зона для размещения наиболее часто используемых предметов
- 2 - зона для размещения часто используемых предметов
- 3 - зона для размещения редко используемых предметов

Рабочее место сотрудника аудитории 220, 11 корпуса ТПУ соответствует требованиям ГОСТ 12.2.032 - 78.

4.2. Производственная безопасность

В учебной аудитории, имеются опасные и вредные производственные факторы. В таблице 21 приведены возможные опасные и вредные производственные факторы, влияющие на человека при исследовании.

4.2.1. Производственная безопасность

В таблице 21 приведены возможные опасные и вредные производственные факторы.

Таблица 21 – Опасные и вредные производственные факторы

Факторы	Нормативные документы
1.Отклонение показателей микроклимата	СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания [25]
2.Недостаточная освещенность рабочей зоны	СП 52.13330.2016 Естественное и искусственное освещение. [26]
3. Повышенный уровень электромагнитных излучений	ГОСТ 12.1.006 - 84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. [27]
4. Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	ГОСТ 12.1.038-82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов. [28]

4.2.2 Отклонение параметров микроклимата

Источник возникновения фактора – энергозатраты организма 151 – 200 (ккал/ч) [25], связанные с постоянной ходьбой, перемещением мелких (до 1 кг)

изделий или предметов в положении стоя или сидя и требующие определенного физического напряжения, при работе в аудитории ТПУ.

При неблагоприятном микроклимате нарушается сердечный ритм, снижается концентрация, колеблется температура тела.

Поддержание оптимальных величин температуры и влажности воздуха рабочей зоны обеспечивается приточной и вытяжной вентиляцией, которая рассчитана на поглощение тепловыделений. Вентиляторы включаются автоматически при достижении температуры в помещении 33°C и отключаются при температуре 25 °С.

Допустимые параметры микроклимата приведены в таблице 22.

Таблица 22 Допустимые параметры микроклимата аудитории

Период года	Категория работ	Температура воздуха, °С		Температура поверхностей, t°С	Относительная влажность воздуха, ф%	Скорость движения воздуха, м/с
		Диапазон ниже оптимальных величин t° _{опт}	Диапазон выше оптимальных величин t° _{опт}			
Холодный	Іб	17,0 - 18,9	21,1 - 23,0	20 - 24,0	40 - 60	0,1
Теплый	Іб	16,0 - 18,9	22 - 24,0	21 - 25,0	40 - 60	0,1

В аудитории проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы на ЭВМ.

4.2.3 Недостаточная освещенность рабочей зоны

Аудитория 41 имеет подвальный тип расположения поэтому возможности организации естественного освещения.

Освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300 - 500 лк [26]. Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк [71].

В качестве источников света применяются светодиодные светильники или металлогалогенные лампы (используются в качестве местного освещения) [26].

4.2.4 Повышенный уровень электромагнитных излучений

Источником повышенного уровня электромагнитных излучений являются мониторы наблюдения за технологическим процессом.

При длительном воздействии ЭМИ различных диапазонов длин волн при умеренной интенсивности (выше ПДУ) могут появиться головные боли, повышение или понижение давления, урежение пульса, изменение проводимости в сердечной мышце, нервно-психические расстройства, быстрое развитие утомления.

Нормирование ЭМИ радиочастотного диапазона проводится по ГОСТ 12.1.006 - 84 для производственной среды. Для защиты от влияния ЭМИ рекомендуется отдых и защитные очки [27].

4.2.5 Поражение электрическим током

Для предотвращения поражения электрическим током, где размещаются рабочее место с ЭВМ в аудитории 220, 11 корпуса ТПУ, оборудование оснащено защитным заземлением, занулением в соответствии с техническими требованиями по эксплуатации [29]. Напряжение для питания ЭВМ 220 В, для серверного оборудования 380 В. По опасности поражения электрическим током помещение 220, 11 корпуса ТПУ относится к первому классу – помещения без повышенной опасности.

Основными непосредственными причинами электротравматизма, являются: 1) прикосновение к токоведущим частям электроустановки, находящейся под напряжением в случае пробоя изоляции; 2) прикосновение к металлическим конструкциям электроустановок, находящимся под напряжением; 3) ошибочное включение электроустановки или несогласованных действий обслуживающего персонала; 4) поражение шаговым напряжением.

Основными техническими средствами защиты, согласно ПУЭ [29], являются защитное заземление, автоматическое отключение питания, устройства защитного отключения, изолирующие электрозащитные средства, знаки и плакаты безопасности. Указанные средства защиты обеспечивают защиты от поражения электрическим током в аудитории 41, 4 корпуса ТПУ.

Рассчитано защитное заземление для шкафов релейной защиты и серверного оборудования, которое находится в аудитории 241, 4 корпуса ТПУ.

1. В качестве заземляющего устройства (вертикальные электроды) используем стальные трубы диаметром $d = 55$ мм, в качестве соединяющего элемента – стальная полоса шириной $b = 50$ мм.

2. Сопротивлению грунта в районе размещения установки или устройства.

Таблица 23 - Исходные данные для расчета

Вид заземления	контурное
Длина заземлителя l , м	2,7
Глубина заземлителя в грунте h , м	0,65
Сезонный коэффициент K_c	2,0
Удельное сопротивление земли ρ , Ом·м	70
Диаметр d , мм	55
Ширина соединительной полосы b , мм	50
Допустимое сопротивление системы заземления по ПУЭ $R_{з.у.}$, Ом	4
Уровень напряжения, В	220-380
Коэффициент экранирования	0,58

3. Величина электрического сопротивления растекания тока в грунт с одиночного заземлителя:

$$R_3 = 0,366 \cdot \frac{\rho \cdot K_c}{l} \left(\lg \cdot \frac{2 \cdot l}{d} + 0,5 \lg \cdot \frac{4 \cdot t + 1}{4 \cdot t - 1} \right) =$$

$$0,366 \cdot \frac{70 \cdot 2}{2,7} \left(\lg \cdot \frac{2 \cdot 2,7}{0,055} + 0,5 \lg \cdot \frac{4 \cdot 2 + 2,7}{4 \cdot 2 - 2,7} \right) = 38,51 \text{ Ом},$$

где,

$\rho = 70$ Ом - удельное сопротивление грунта,

$K_c = 2$ - коэффициент сезонности,

$l = 2,7$ м – длина заземлителя,

$d = 0,055$ м – диаметр заземлителя

$t = h + 0,5l = 0,65 + 0,5 \cdot 2,7 = 2$ м

4. Число заземлителей без взаимных помех, получаемых друг от друга, без так называемого явления «экранирования»:

$$n' = \frac{R_3}{R_{3,y}} = \frac{38,51}{4} = 9,62 \approx 10.$$

5. Число заземлителей с коэффициентом экранирования:

$$n = \frac{n'}{\eta_3} = \frac{10}{0,58} = 17,24 \approx 18.$$

Принимаем расстояние между заземлителями $a = l = 2,7$ м.

6. Длина соединительной полосы:

$$l_n = 1,05 \cdot n \cdot a = 1,05 \cdot 18 \cdot 2,7 = 51 \text{ м}.$$

7. Значение сопротивления растекания тока с соединительной полосы:

$$R_3 = 0,366 \cdot \frac{\rho \cdot K_c}{l} \left(\lg \cdot \frac{2 \cdot l_n^2}{b \cdot h} \right) = 0,366 \cdot \frac{70 \cdot 2}{51} \left(\lg \cdot \frac{2 \cdot 51^2}{0,05 \cdot 0,65} \right) = 5,1 \text{ Ом}.$$

8. Полное сопротивление системы заземления:

$$R_{3y} = \frac{R_3 \cdot R_{II}}{R_3 \cdot \eta_{II} + R_3 \cdot \eta_3 \cdot n} = \frac{38,51 \cdot 5,1}{38,51 \cdot 0,51 + 5,1 \cdot 0,58 \cdot 18} = 2,63 \text{ Ом},$$

где,

$\eta_{II} = 0,51$ - коэффициент экранирования полосы.

Таким образом, сопротивление $R_{3y} = 2,63$ Ом не превышает 4 Ом. Следовательно, диаметр заземлителя $d = 55$ мм при числе заземлителей $n = 18$ является достаточным для обеспечения защиты при контурной схеме расположения заземлителей.

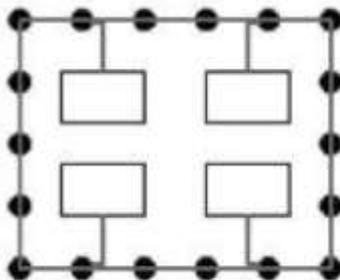


Рисунок 18 – Схема полученного контурного заземления

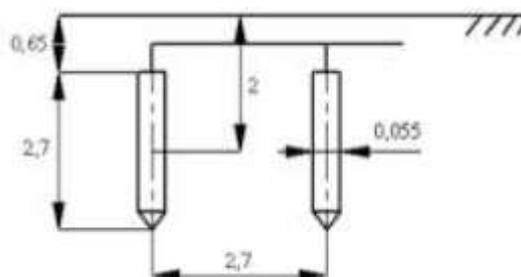


Рисунок 19 – Схема расположения заземлителей

Разработанные мероприятия и расчеты обеспечивают безопасную эксплуатацию электроустановок в аудитории 41, 4 корпуса ТПУ.

4.3. Экологическая безопасность

4.3.1 Влияние на атмосферу и гидросферу

При использовании компьютера для разработки потребляется электроэнергия. Выработка электроэнергии осуществляется на ТЭС и ГРЭС. При выработке электроэнергии за счет сжигания топлива в воздух поступают различные загрязнения.

Для уменьшения выбросов на ТЭС применяют различные фильтры и разные технологии сжигания.

На гидросферу также влияет по большей части выработка тепла и электроэнергии. Гидросфера загрязняется обмывочными водами котлов, замасляными водами, сброс тепловых потоков.

Чтобы минимизировать сброс обмывочных и замасленных вод в гидросферу, чаще всего используют очистку и масляные ловушки.

4.3.2 Влияние на литосферу

Процесс исследования представляет из себя работу с информацией, такой как технологическая литература, статьи, ГОСТы и нормативнотехническая документация, а также разработка математической модели с помощью различных программных комплексов. Таким образом процесс исследования имеет влияние негативных факторов на окружающую среду. Таких как отходы – использованная бумага, использованные шариковые ручки.

Использованная бумага и пластиковые шариковые ручки утилизируется, как вторичное сырье – изготовление картона, пластиковой тары и.т.д. Процесс утилизации негативно влияет на атмосферу, выделяя в нее углекислый газ.

Существующая система фильтрации не может на 100% избавить от выделения вредных веществ в атмосферу. Что касается методов по защите литосферы, то используются следующие методы:

- Энергосбережение;
- Сортировка мусора.

4.4 Безопасность в чрезвычайных ситуациях

4.4.1. Анализ вероятных ЧС, которые может инициировать объект исследований и обоснование мероприятий по предотвращению ЧС

При проектировании и эксплуатации программного продукта возможны следующие чрезвычайные ситуации в рабочей зоне:

- техногенные - производственные аварии, пожары;

Наиболее типичным и вероятным видом чрезвычайной ситуации является пожар в рабочей зоне. Наиболее вероятные причины пожара связаны с неисправностью или ненадлежащей эксплуатацией электроприборов – короткое замыкание или перегрузки по току, нарушение работником правил эксплуатации. В соответствии с СП 12.13130.2009 [30] помещение лаборатории относится к категории В1 зона П-Па. Перед тушением необходимо отключить общее электроснабжение. Для тушения можно использовать огнетушители разного типа, но, если отключить электроснабжение не удастся, необходимо применять только порошковые и углекислотные. Требования к эксплуатации огнетушителей приведены в СП 9.13130.2009 [30].

4.4.2. Меры по предупреждению возникновения пожара

При проведении исследований наиболее вероятной ЧС является возникновение пожара в помещении 220, 11 корпуса ТПУ. Пожарная безопасность должна обеспечиваться системами предотвращения пожара и противопожарной защиты, в том числе организационно-техническими мероприятиями. Основные источники возникновения пожара:

- 1) Неисправное электрооборудование, неисправности в проводке, розетках и выключателях. Для исключения возникновения пожара по этим причинам необходимо вовремя выявлять и устранять неполадки, а также проводить плановый осмотр электрооборудования.
- 2) Электрические приборы с дефектами. Профилактика пожара включает в себя своевременный и качественный ремонт электроприборов.
- 3) Перегрузка в электроэнергетической системе (ЭЭС) и короткое замыкание в электроустановке.

Под пожарной профилактикой понимается обучение пожарной технике безопасности и комплекс мероприятий, направленных на предупреждение

пожаров.

Пожарная безопасность обеспечивается комплексом мероприятий:

- обучение, в т.ч. распространение знаний о пожаробезопасном поведении (о необходимости установки домашних индикаторов задымленности и хранения зажигалок и спичек в местах, недоступных детям);
- пожарный надзор, предусматривающий разработку государственных норм пожарной безопасности и строительных норм, а также проверку их выполнения;
- обеспечение оборудованием и технические разработки (установка переносных огнетушителей и изготовление зажигалок безопасного пользования).

В соответствии с ТР «О требованиях пожарной безопасности» для административного жилого здания требуется устройство внутреннего противопожарного водопровода.

Согласно ФЗ-123, НПБ 104 - 03 «Проектирование систем оповещения людей о пожаре в зданиях и сооружениях» для оповещения о возникновении пожара в каждом помещении должны быть установлены дымовые оптикоэлектронные автономные пожарные извещатели, а оповещение о пожаре должно осуществляться подачей звуковых и световых сигналов во все помещения с постоянным или временным пребыванием людей.

Аудитория 41, 4 корпуса ТПУ оснащена первичными средствами пожаротушения: огнетушителями ОУ-3 1шт., ОП-3, 1шт. (предназначены для тушения любых материалов, предметов и веществ, применяется для тушения ПК и оргтехники, класс пожаров А, Е.).

Согласно НПБ 105-03 помещение, предназначенное для проектирования и использования результатов проекта, относится к типу П - 2а.

4.4.3 Действия в случае возникновения пожара

В случае возникновения пожара необходимо:

- спокойно оценить ситуацию и принять срочные меры по предотвращению распространения огня;
- вызвать пожарных, сообщив точный адрес места возгорания и ФИО, вызывающего;
- отключить общее электроснабжение;
- попытаться потушить пожар при помощи первичных средств пожаротушения;
- если самостоятельно справиться с огнем не удаётся, следует включить сигнал пожарной тревоги и, согласно плану, приступить к эвакуации;
- встретить прибывшую пожарную команду и обеспечить для неё беспрепятственный доступ и пути подъезда к месту пожара.

Вывод по разделу

В результате выполнения задания раздела «Социальная ответственность и ресурсосбережение» ВКР были выявлены и проанализированы вредные факторы при разработке алгоритма.

Были установлены правовые и организационные вопросы обеспечения безопасности, характерные для рабочей зоны.

Проработаны организационные мероприятия при компоновке рабочей зоны.

Разработаны мероприятия по снижению воздействия вредных и опасных факторов.

Был рассмотрен характер воздействия исследуемого решения на окружающую среду. Были выявлены предполагаемые источники загрязнения окружающей среды, возникшие в результате реализации предлагаемых в магистерской диссертации решений.

Можно сделать вывод, что создание магистерская диссертация не является экологически безвредным действием. Так как процесс сопровождается созданием отходов от проектной деятельности.

А также процесс написания магистерской диссертации не является абсолютно безвредным для человека. Так как в процессе написания магистерской диссертации человека сопровождают такие вредные факторы, как электромагнитное излучение от ПК и плохая освещенность рабочей зоны.

Даны общие рекомендации по безопасности, соблюдая которые можно, не только сохранить здоровье, но и увеличить эффективность.

Список публикаций студента

1. Карнаухов В. А. Особенности применения алгоритма цифровой подписи ED25519 в распределенной платформе Solana / В. А. Карнаухов ; науч. рук. О. Л. Крицкий // Молодежь и современные информационные технологии : сборник трудов XV Международной научно-практической конференции студентов, аспирантов и молодых ученых, (Томск, 21–25 марта 2022 г.) / Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2022.— [С. 75-77].

Список литературы

- [1] Portnoff, R.S., Huang, D.Y., Doerfler, P., Afroz, S., McCoy, D.: Backpage and bitcoin: Uncovering human traffickers. In: SIGKDD, ACM (2017) 1595–1604
- [2] Moser, M. and Bohme, R., Breuker, D.: An inquiry into money laundering tools in the bitcoin ecosystem. In: eCRS, IEEE (2013) 1–14
- [3] C. G. Akcora, A. K. Dey, Y. R. Gel, and M. Kantarcioglu, Forecasting bitcoin price with graph chainlets, in PaKDD, 2018, pp. 765–776
- [4] N. Abay, C. Akcora, Y. Gel, U. Islambekov, M. Kantarcioglu, B. Thuraisingham, and Y. Tian, Chainnet: Learning on blockchain graphs with topological features, in IEEE ICDM, 2019, pp. 1–
- [5] Akcora C. G., Dixon M. F., Gel. Y. R., Kantarcioglu M., 2018. Bitcoin risk modeling with blockchain graphs. In: arXiv preprint arXiv:1708.08749.
- [6] Суслов В.И., Ибрагимов Н.М., Талышева Л.П., Цыплаков А.А. Эконометрия. - Новосибирск: СО РАН, 2005. - 744 с.
- [7] Elman J. L. Finding structure in time / Elman J. L. – 1990. – Vol. 14, №2. – P. 179–211.
- [8] Understanding LSTM Networks [Электронный ресурс].- Режим доступа: <http://colah.github.io/posts/2015-08-Understanding-LSTMs>, свободный. – Загл. с экрана. (дата обращения: 10.02.2023)
- [9] Hochreiter S. Long short-term memory. Neural computation / Hochreiter S., Schmidhuber J. – 1997 – Vol. 9, №8. – P. 1735–1780.
- [10] Using Python and Auto ARIMA to Forecast Seasonal Time Series [Электронный ресурс]. – Режим доступа <https://medium.com/@josemarcialportilla/using-python-and-auto-arima-to-forecast-seasonal-time-series-90877adff03c>, свободный – Загл. с экрана. (дата обращения: 15.02.2023)
- [11] ARIMA-GARCH forecasting with Python [Электронный ресурс]. – Режим доступа <https://medium.com/analytics-vidhya/arima-garch-forecasting-with-python-7a3f797de3ff>, свободный – Загл. с экрана. (дата обращения: 15.02.2023)

[12] How to Create an ARIMA Model for Time Series Forecasting in Python [Электронный ресурс]. – Режим доступа <https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/>, свободный – Загл. с экрана. (дата обращения: 15.02.2023)

[13] Azzouni A., Pujolle G. A long short-term memory recurrent neural network framework for network traffic matrix prediction // arXiv preprint arXiv:1705.05690. – 2017.

[14] Nelson D. M. Q., Pereira A. C. M., de Oliveira R. A. Stock market's price movement prediction with LSTM neural networks //2017 International Joint Conference on Neural Networks (IJCNN). – IEEE, 2017. – С. 1419-1426.

[15] Kumar D. N., Raju K. S., Sathish T. River flow forecasting using recurrent neural networks //Water resources management. – 2004. – Т. 18. – №. 2. – С. 143-161.

[16] Chen K., Zhou Y., Dai F. A LSTM-based method for stock returns prediction: A case study of China stock market //2015 IEEE International Conference on Big Data (Big Data). – IEEE, 2015. – С. 2823-2824.

[17] Liu H., Mi X., Li Y. Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum 76 analysis, LSTM network and ELM //Energy Conversion and Management. – 2018. – Т. 159. – С. 54-64.

[18] Massimo Bartoletti, Stefano Lande, Livio Pompianu, and Andrea Bracciali. A general framework for blockchain analytics. In First Workshop on Scalable and Resilient Infra-structures for Distributed Ledgers, 2017.

[19] Akcora C. G., Gel Y. R., Kantarcioglu M., 201. Blockchain: A Graph Primer. In:arXiv preprint arXiv:1708.08749.

[20] How to Model Residual Errors to Correct Time Series Forecasts with Python [Электронный ресурс]. – Режим доступа <https://machinelearningmastery.com/model-residual-errors-correct-time-series-forecasts-python/>, свободный – Загл. с экрана. (дата обращения: 12.04.2023)

[21] Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. От 27.12.2018)

[22] Федеральный закон от 29.11.2010 № 326-ФЗ (ред. От 24.02.2021) «Об обязательном медицинском страховании в Российской Федерации»

[23] ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования.

[24] ГОСТ 12.0.003-2015 ССБТ. Опасные и вредные производственные факторы. Классификация.

[25] СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания

[26] СП 52.13330.2016 Естественное и искусственное освещение.

[27] ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности.

[28] ГОСТ 12.1.038-82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов.

[29] ПУЭ: правила устройства электроустановок. Издание 7.

[30] СП 12.13130.2009 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности

Приложение А
(справочное)

Creating and using a chainlet to predict the price of cryptocurrencies

Студент

Группа	ФИО	Подпись	Дата
0BM12	Карнаухов В.А.		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Крицкий О.Л.	к.ф.-м.н., доцент.		

Консультант-лингвист отделения иностранных языков ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Смирнова У.А			

Chainlets - graph chains

As shown in figure 1, a bitcoin graph consists of three main components: an address (wallet), a transaction and a block. A transaction can have multiple inputs and multiple outputs. For example, in figure 1 the transaction t_1 receives bitcoins from addresses a_1 and a_2 , and transfers to addresses a_6 and a_7 . An analogy from real life is a person using multiple bank accounts, combining funds into one transaction and sending amounts to multiple accounts. There are bitcoin mixing services, bitcoin mixers, which basically work according to a simple scheme: client funds are split into small pieces, then these pieces are mixed randomly with pieces from other clients. As a result of all the operations, a given number of bitcoins arrive at the final recipient, but in small batches from different randomly selected participants. Thus, the addresses that appear at the input and output are the same.

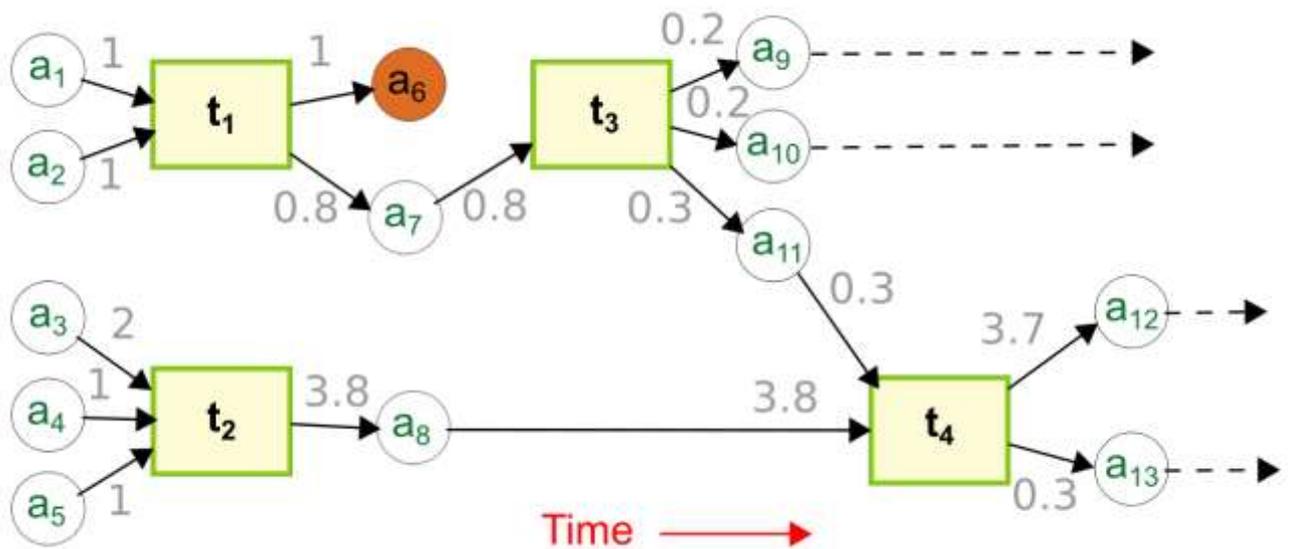


Figure 1 - Graphical representation of the Bitcoin network as an address-transaction graph

Inputs and outputs can provide important information about the purpose of a transaction. For example, transactions involving hundreds of inputs and very few outputs can be large amounts of bitcoin investment.

A K-chainlet is a Bitcoin subgraph consisting of $k \geq 1$ transactions and their respective input and output addresses. In the simplest case, one transaction creates a 1-

chainlet with one or more inputs and one output. For example, in figure 1, transaction t_2 results in the transfer of bitcoins from addresses a_3, a_4, a_5 to address a_8 . Such a transaction creates a 1-chainlet with three inputs and one output. We denote this subgraph as $C_3 \rightarrow 1$, where 3 and 1 are the number of input and output addresses, respectively.

A 1-chainlet is the smallest element in a Bitcoin graph; the chain's inputs and outputs are defined immediately and the transaction is digitally signed. This signed information cannot be changed, but multiple 1-chainlet chains can be combined to enlarge the graph.

Extreme chains. in figure. 2 illustrate how network activity can be represented by a matrix of chains, with the colour coding indicating the frequency of chain occurrence. The occurrence of each $C_i \rightarrow j$ is counted over a certain period. The maximum number of chain inputs or outputs can be large, sometimes exceeding 1000. Chains in which the number of inputs and outputs exceeds the threshold value of N are called "extreme" chains. Based on a historical analysis of daily snapshots, $N = 20$ was chosen, which corresponds to 97.5 percentile of the whole chain.

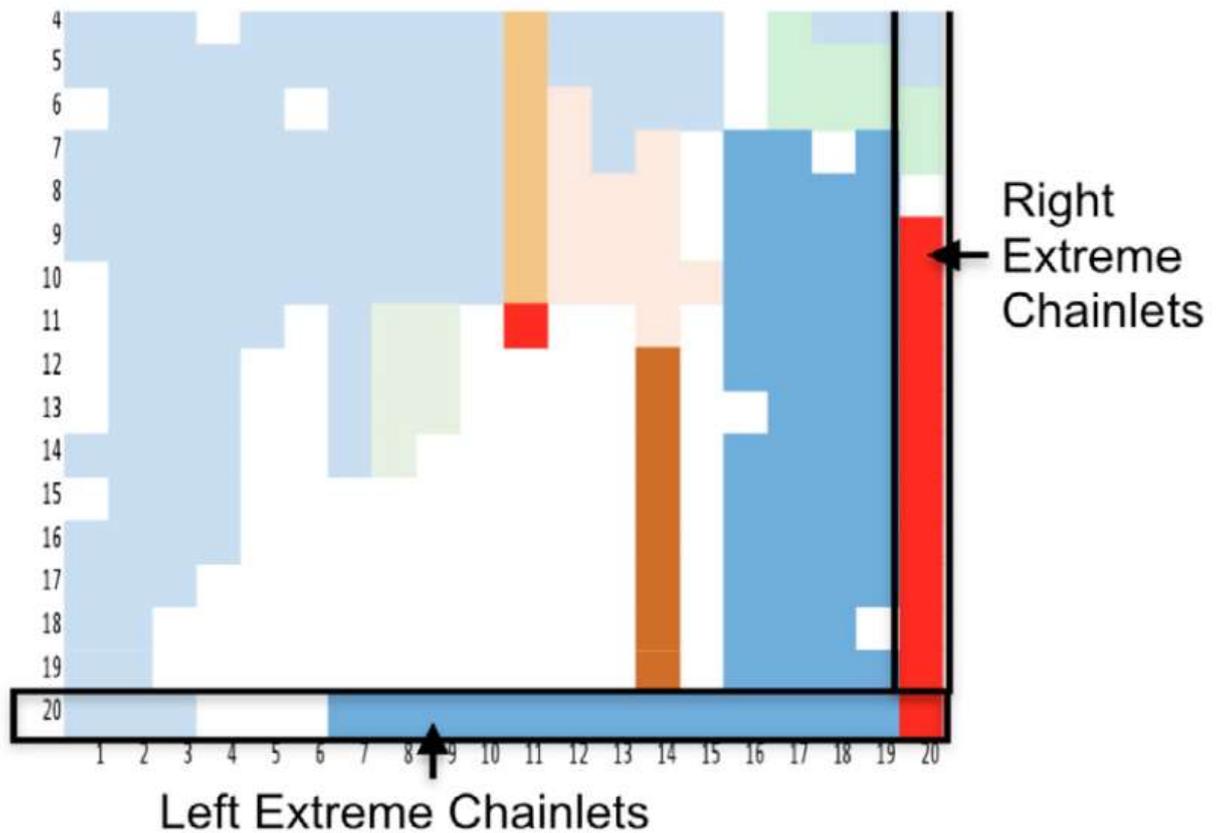


Figure 2 - Representation of the network as a matrix of chains.

There are "left extreme chains". and "right extreme chains".

The left extreme chains are a subset of C .^l

$$C^l := \{C_i \rightarrow j \mid i = N, j \in \{1, \dots, N\}\} \quad (1)$$

They represent transactions from a large number of addresses to a smaller number of addresses. Typically, the left extreme chains indicate bitcoin attachments.

The right extreme chains are a subset of C .^r

$$C^r := \{C_i \rightarrow j \mid i \in \{1, \dots, N - 1\}, j \in N\} \quad (2)$$

They represent the sale of a large amount of bitcoins - the seller sells off his coins piece by piece and sends them potentially to hundreds of bitcoin addresses.

ARIMA forecast model

In 1976 Box and Jenkins proposed the AutoRegressive Integrated Moving Average (ARIMA) model. The ARIMA model has 3 parameters that are used to

model the main aspects of time series: seasonality, trend and noise. These parameters are denoted by p, d and q.

- p is a parameter related to the autoregressive aspect of the model, which includes past values. For example, by predicting that if the price of a stock has risen in the past few days, you are indicating that prices are likely to rise tomorrow as well.
- d is a parameter associated with the integrated part of the model, which affects the amount of variance applied to the time series. For example, that the trading volume of a stock tomorrow will be similar to the volume today if the daily trading volume has been similar over the past few days.
- q is a parameter related to the moving average of the model.

The ARIMA(p, d, q) model for the non-stationary time series X_t has the form:

$$\Delta^d X_t = a_0 + \sum_{i=1}^p a_i \Delta^d X_{t-1} + \sum_{j=1}^q b_j \varepsilon_{t-j} + \varepsilon_t \quad (3)$$

where ε_t - the errors of the observations with a normal distribution;

a_i, b_j - model parameters;

Δ^d - time series difference operator of order d (sequentially taking d first-order differences - first from the time series, then from the resulting first-order differences, then from the second-order differences, etc.).

ARCH model family

ARCH (Autoregressive Conditional Heteroscedastic model) models were proposed by Robert Engle in 1982 and have since become a popular tool for analyzing financial time series. This model is called the ARCH-model (Autoregressive Conditional Heteroscedastic model), which uses conditional, time-dependent dispersion, expressed through the square of the values of indicators of past periods:

$$\sigma^2(t) = a_0 + \sum_{i=1}^q a_i \mu_{t-1}^2 \quad (4)$$

where q - is the delay factor (lag);

a_i, a_0 - weighting factors;

μ_n - relative increments of time series values or logarithmic increments.

GARCH (Generalized Autoregressive Conditional Heteroscedastic model) models are the most common variant of ARCH models and are used for financial time series analysis with volatile risk. They were proposed by Tim Bollersell and Jan Timmers in 1986 and are a generalization of ARCH models.

GARCH models can be adapted to various instruments and applied to different financial series (currency rates, stocks, cryptocurrencies). They allow a more accurate risk analysis, forecast the future state of the series and help to make the right investment decisions.

According to this model (GARCH(p,q)), the variance is calculated according to the following formula:

$$\sigma^2(t) = \alpha_0 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{j=1}^q \alpha_j \mu_{t-j}^2 \quad (5)$$

Where p is the number of previous assessments affecting the current value;
 α, β - weighting coefficients reflecting the degree of influence of previous estimates on the current value;

σ_n - volatility.

ARIMA-GARCH model

To combine the ARIMA and GARCH models, use the following formula:

$$\hat{y}_t = a_0 + \sum_{i=1}^p a_i y_{t-i} + \sum_{i=1}^q b_i \varepsilon_{t-i} + \varepsilon_t \quad (6)$$

$$\varepsilon_t = \sqrt{\sigma_t} z_t \quad (7)$$

$$\sigma^2 = \omega + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2 \quad (8)$$

$$z_t = iid N(0,1) \quad (9)$$

The final forecast is given by combining the output of the ARIMA model and the GARCH model, with the GARCH model trained on the residual forecast errors of the ARIMA model, which are expressed by formula (7).

Recurrent neural networks

Recurrent Neural Networks (RNN - Recurrent Neural Network) is a class of neural networks used in machine learning to process sequences of data such as texts, time series, sounds and others. Recurrent neural networks are able to store context while processing data and then use it to make a decision.

This is due to the fact that at each training step t the value of the hidden layer of the recurrent neural network $h^t \in R^m$ is calculated as follows:

$$h^t = f(W \times x^t + U \times h^{(t-1)} + b^h), \quad (10)$$

where $x^t \in R^n$ is the input vector at time t (e.g. the value of a cryptocurrency for n days);

$W \in R^{m \times n}$, $U \in R^{m \times m}$, $b^h \in R^m$ are the trained parameters of the recurrent neural network;

f is a non-linear transformation function.

The most common non-linear transformation function is one of the following: sigmoidal function (11), hyperbolic tangent (12), "ReLU" (13).

$$f(x) = \sigma(x) = \frac{1}{1 + e^{(-x)}}, \quad (11)$$

$$f(x) = \tanh(x) = \frac{e^{(x)} - e^{(-x)}}{e^{(x)} + e^{(-x)}}, \quad (12)$$

$$f(x) = \max(0, x), \quad (13)$$

In a simple recurrent neural network (Figure 5), the output value $y^t \in R^l$ at the current step t is calculated using the formula:

$$y^t = W \times h^t + b, \quad (14)$$

where $W \in R^{l \times m}$ and $b \in R^l$ are trainable parameters.

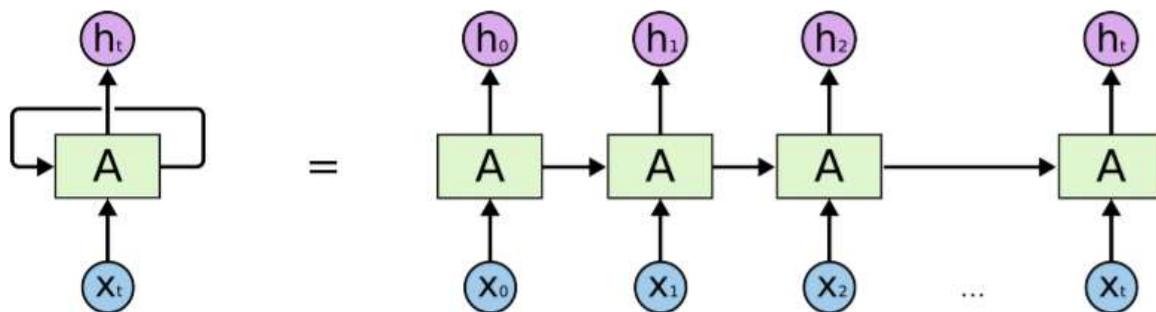


Figure 5 - The simplest recurrent network system

The application of these systems is used in many tasks. RNNs have been shown to be effective in analysing time series and predicting future values. They can be used to analyse technical indicators such as moving averages or stochastic oscillator, as well as fundamental factors such as company profit and loss reports, macroeconomic indicators and others.

One of the most popular implementations of RNN, are LSTM (Long Short Term Memory) models, which have special mechanisms for better memory handling than RNN. A more detailed discussion of recurrent networks will be given in the next subsection.

LSTM

In 1997, a new approach called LSTM (Long Term Memory), authored by Sepp Hochreiter and Jürgen Schmidhuber, was introduced. Recurrent neural networks based on this approach have a more complex way of computing h^t . It uses input values, the previous state of the network and special filters ('gates') that determine how to process

the information to produce output values at the current layer y^l and hidden layer values at the next step h^{t+1} .

All recurrent neural networks take the form of a chain of recurrent neural network modules. In standard RNNs the repeating module has a simple structure, for example, it can include a single layer of hyperbolic tangent, shown in figure 6.

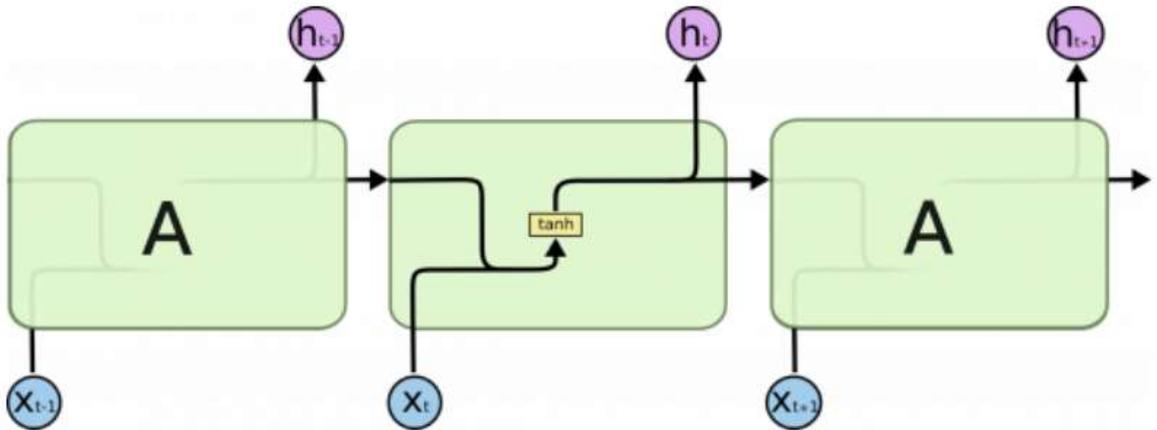


Figure 6 - Schematic representation of a repeating module of a standard RNN consisting of one layer

Figure 7, in turn, shows an LSTM network, which has clear differences. In order to understand them, let us look at their mathematical features.

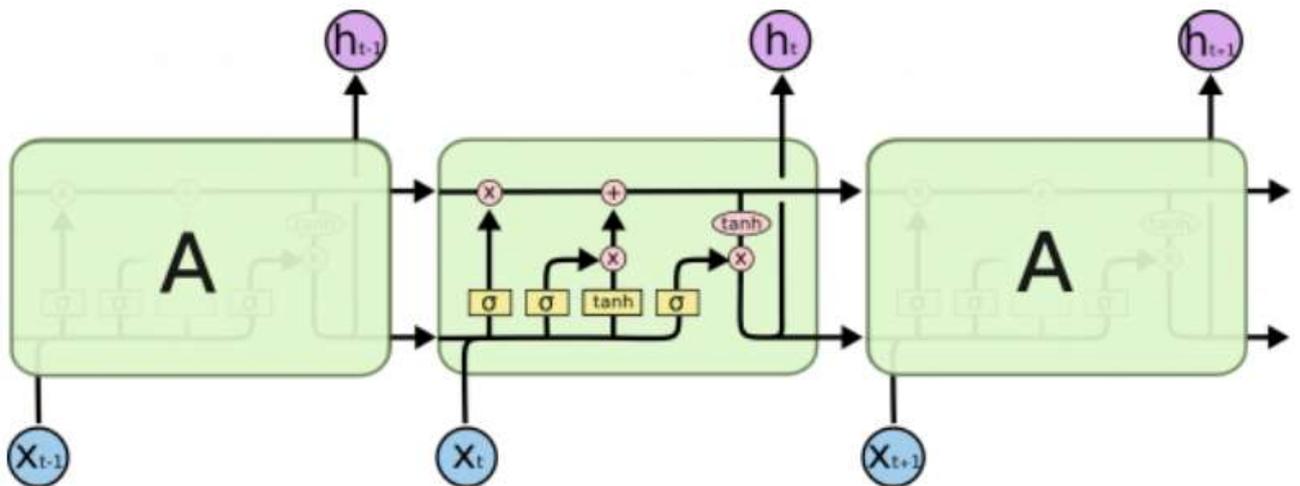


Figure 7 - Schematic representation of a repeating LSTM consisting of four interacting layers

This model is designed specifically to eliminate the problem of long-term addiction. The main feature is that the information is remembered over long periods of time, which ensures that no training is required.

Let's take a closer look at the structure of the LSTM layer. The central element here is the memory block ("memory cell"), which, along with the network state h , is computed at each step using the current input value x , t and the block value at the previous step c^{t-1} . The input filter ("input gate") i^t determines how much the value of the memory block in the current step should affect the result. The filter values range from 0 (ignore input values completely) to 1, which is provided by the value area of the sigmoidal function:

$$i^t = \sigma(W^i x^t + U^i h^{t-1}), \quad (15)$$

The "forget gate" allows you to exclude the memory values of the previous step from the calculations:

$$f^t = \sigma(W^f x^t + U^f h^{t-1}), \quad (16)$$

Based on all the data arriving at time t , the state of the memory block c^t at the current step is calculated using filters (17) and (18):

$$\check{c}^t = \tanh(W^c x^t + U^c h^{t-1}), \quad (17)$$

$$c^t = f^t \cdot c^{t-1} + i^t \cdot \check{c}^t, \quad (18)$$

The "output gate" is similar to the previous two and is of the same form:

$$o^t = \sigma(W^o x^t + U^o h^{t-1}), \quad (19)$$

The final value of the LSTM layer is determined by the output filter (19) and the non-linear transformation over the memory block state (20):

$$h^t = o^t \cdot \tanh(c^t), \quad (20)$$

There are many variations on the activation functions used by each layer, and it is possible to make some minor changes to the circuit itself and any of its parameters. However, the essence of the operation does not change - first a part of the memory is

filtered, then a part of the new signal is stored, and then the result is calculated based on this data. For example, one popular variant of the LSTM network is shown in figure 8. It was proposed in 2000 by Felix Gers and Jürgen Schmidhuber in their research paper.

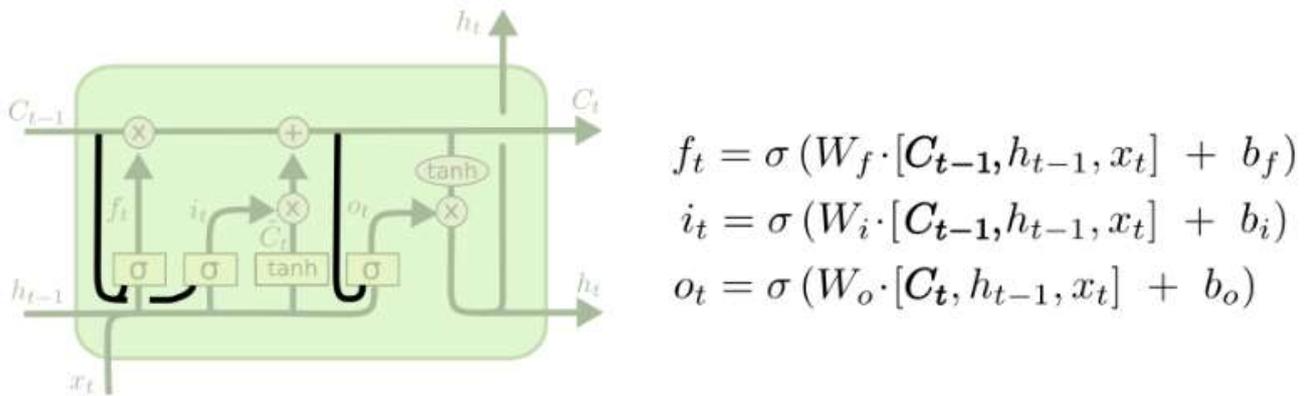


Figure 8 - Schematic representation of the LSTM model by Geers and Schmidhuber

Conclusion

As a result of the final qualifying paper, the main conclusions can be drawn:

1. The Google BigQuery service was used to download monthly bitcoin transaction data (about 600 MB of data) from the bitcoin blockchain (about 450 GB of transaction data in total). To increase processing speed, daily chain matrices were generated using list inclusion techniques. Heatmaps were constructed and all extreme chains were detected;

2. ARIMA, ARIMA-GARCH econometric models have been built and applied to chainlet data forecasting. In addition, a recurrent neural network with long-term short-term memory (LSTM) has been trained, calibrated and verified;

3. The comparative analysis of predicting bitcoin quotes by different models for 72 days ahead showed that LSTM model is comparable to classical econometric algorithms in terms of the percentage average absolute error (MAPE) when the number of neurons is small (up to 200), but becomes more accurate when the number of

neurons exceeds 256. Re-application of the LSTM model with 256 neurons showed the best result among all models considered with MAPE = 10.77%.

Приложение Б

```
from google.cloud import bigquery
import numpy as np
import pandas as pd

SAVE_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\raw\\"

START_YEAR = 2018
END_YEAR = 2022
client = bigquery.Client()

def sql_query_month(datemonth):
    query = """
        SELECT block_number, block_timestamp, block_timestamp_month, input_count,
        output_count, input_value, output_value
        FROM `bigquery-public-data.crypto_bitcoin.transactions`
        WHERE block_timestamp_month = '{datemonth}' AND is_coinbase = FALSE order by
        block_timestamp
        """
    return query.format(datemonth=datemonth)

for year in range(START_YEAR, END_YEAR + 1):
    for month in range(1, 13):
        date = '{}-{}-01'.format(year, month)
        sql = (sql_query_month(date))

client.query(sql).to_dataframe(progress_bar_type="tqdm").to_csv(SAVE_PATH + date
+ ".csv")

print("COMPLETED")
```

Приложение В

```
import pandas as pd
import numpy as np
from datetime import datetime

RAW_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\raw\\"
SAVE_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\"
PRICED_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\BTC-
USD.csv"

START_YEAR = 2018
END_YEAR = 2022

for year in range(START_YEAR, END_YEAR + 1):
    for month in range(1, 13):
        path = RAW_PATH + '{}-{}-01'.format(year, month) + ".csv"
        priced = pd.read_csv(PRICED_PATH, names=['Date', 'Open', 'High', 'Low',
'Close', 'Adj Close', 'Volume'])
        parse_dates = ['block_timestamp']
        df = pd.read_csv(path, parse_dates=parse_dates)
        df['date'] = df['block_timestamp'].dt.date
        days_grouped = df.groupby(["date"])["input_value"].count()
        days = len(days_grouped.index)
        days_range = range(0, days)
        day_occ = [np.zeros((20, 20), dtype=int) for i in days_range]
        day_amo = [np.zeros((20, 20)) for i in days_range]
        del days_grouped

        def handle(timestamp, input_count, output_count, input_value):
            day_index = timestamp.day
            if input_count == 0:
                return
            if input_count > 20:
                input_count = 20
            if output_count > 20:
                output_count = 20
            day_occ[day_index - 1][input_count - 1][output_count - 1] += 1
            day_amo[day_index - 1][input_count - 1][output_count - 1] +=
input_value

            print('Start {}-{} at {}'.format(year, month,
datetime.now().strftime("%Y-%m-%d %H:%M:%S")))
            result = [handle(date, in_c, out_c, in_v) for date, in_c, out_c, in_v in
zip(df['block_timestamp'], df['input_count'],
df['output_count'], df['input_value'])]

            for day in days_range:
                pd.DataFrame(day_occ[day]).to_csv(
                    "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\" +
"occ\\" + '{}-{}-{}'.format(year, month,
day + 1) + ".csv")
                pd.DataFrame(day_amo[day]).to_csv(
                    "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\" +
"amo\\" + '{}-{}-{}'.format(year, month,
day + 1) + ".csv")
```

```
        print('Completed {}-{} at {}'.format(year, month,
datetime.now().strftime("%Y-%m-%d %H:%M:%S")))

print("Finish")
```

Приложение Г

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from datetime import datetime, date

OCC_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\occ\\"
AMO_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\amo\\"
PRICED_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\BTC-
USD.csv"
plt.figure(figsize=(19,10))

def heatmap(year, month, day):
    date = "{}-{}-{}".format(year, month, day)
    arr = pd.read_csv(OCC_PATH + date + ".csv", ).iloc[2:, 3:]
    ax = sns.heatmap(data=arr, yticklabels=range(3, 21), xticklabels=range(3,
21), square=True, linewidth=.5,
                    cmap="coolwarm")
    ax.xaxis.tick_top()
    ax.xaxis.set_label_position('top')
    plt.ylabel("Inputs")
    plt.xlabel("Outputs")
    plt.title("Heatmap of chainlets")
    plt.show()

def ratio(start_date, end_date):
    date_range = pd.date_range(start_date, end_date)
    ratios = []
    for date in date_range:
        occ = pd.read_csv(OCC_PATH + "{}-{}-{}".format(date.year, date.month,
date.day) + ".csv", ).iloc[:, 1:]
        ratios.append((occ.iloc[19].sum() + occ['19'].sum()) / occ.values.sum())
    data = {"date": date_range.values, "ratio": ratios}
    return pd.DataFrame(data)

year = 2022
month = 6
day = 26
heatmap(year, month, day)
df = ratio(date(2022, 1, 1), date(2022, 12, 31))
df.plot(x="date", y="ratio", title="Daily ratio", figsize=(19,10))
plt.show()
```

Приложение Д

```
import pandas as pd
import armagarch as ag
import arch
import pmdarima
import numpy as np
from pmdarima.arima import ndiffs, nsdiffs
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose
from math import sqrt
from sklearn.metrics import mean_squared_error, mean_absolute_percentage_error
import cufflinks as cf
import scipy.stats as st
plt.rcParams['figure.figsize'] = (19, 10)

def arima_forecast_one_step(model):
    fc, conf_int = model.predict(n_periods=1, return_conf_int=True, alpha=0.01)
    return (
        fc.tolist()[0],
        np.asarray(conf_int).tolist()[0])

def arima_forecast(model, n):
    forecasts = []
    confidence_intervals = []
    for i in range(n):
        fc, conf = arima_forecast_one_step(model)
        forecasts.append(fc)
        confidence_intervals.append(conf)
        model.update(fc)
    return (forecasts, confidence_intervals)

def garch_forecast_one_step(model, first_obs):
    res = model.fit(last_obs=first_obs, disp="off")
    temp = res.forecast(horizon=1, reindex=False).mean
    return temp.iloc[0]

def garch_forecast(residuals, starting_date_for_forecase, p, q):
```

```

# fit a GARCH(1,1) model on the residuals of the ARIMA model
garch = arch.arch_model(residuals, p=p, q=q, dist='normal', rescale=False)
end_loc = np.where(residuals.index >= starting_date_for_forecast)[0].min()
forecasts = []
for i in range(len(residuals) - end_loc):
    ret = garch_forecast_one_step(garch, end_loc + i)
    forecasts.append(ret["h.1"])
forecasts = pd.DataFrame(forecasts, index=residuals.index[end_loc:],
columns=["Prediction"])
return forecasts

if __name__ == '__main__':
    # read data
    data =
pd.read_csv("C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\BTC-
USD.csv", parse_dates=["Date"])
    # set dates as the main index
    data = data.drop(columns=['Open', 'High', 'Low', 'Adj Close', 'Volume'])

    # Take data for 2022
    data = data.loc[data["Date"].between("2022-1-1", "2022-12-31")]

    data.set_index('Date', inplace=True)
    data.columns = ["Close"]

    # data = data.asfreq('d')
    data.plot(title="Bitcoin USD Price")
    plt.show()
    # decompose time series
    decompose = seasonal_decompose(data, model="multiplicative", period=7)
    decompose.plot()
    plt.figure(figsize=(19, 10))
    plt.show()

    # prepare train and test datasets
    test_date = data.index[round(len(data.index)*0.8)]
    # test_date = '2021-01-12'
    train = data.loc[:test_date].pct_change().dropna()
    test = data.loc[test_date:].pct_change().dropna()
    to_predict = len(test)

```

```

kpss_diffs = ndiffs(train, alpha=0.05, test='kpss', max_d=6)
adf_diffs = ndiffs(train, alpha=0.05, test='adf', max_d=6)
D = nsdiffs(train,
             m=12,
             test='ocsb')
n_diffs = max(adf_diffs, kpss_diffs)

print(f"Estimated differencing term: {n_diffs}")
print(f"Estimated seasonal differencing term: {D}")

# stepwise arima
stepwise_model = pmdarima.auto_arima(train, start_p=1, start_q=1,
                                     max_p=3, max_q=3, m=12,
                                     start_P=0, seasonal=True,
                                     d=n_diffs, D=1, trace=True,
                                     error_action='ignore',
                                     suppress_warnings=True,
                                     stepwise=True, method='nm')
# stepwise_model = pmdarima.auto_arima(train)
print(stepwise_model.aic())
print(stepwise_model.order)
p = stepwise_model.order[0]
q = stepwise_model.order[2]

(future_forecast, confidence_intervals) = arima_forecast(stepwise_model,
to_predict)
future_forecast = pd.DataFrame(future_forecast, index=test.index,
columns=["Prediction"])

pd.concat([test, future_forecast], axis=1).plot()
plt.title("Bitcoin USD Price Prediction with ARIMA")
plt.show()

rmse = mean_squared_error(test['Close'].to_list(),
future_forecast['Prediction'].to_list(), squared=False)
mse = mean_squared_error(test['Close'].to_list(),
future_forecast['Prediction'].to_list())
mape = mean_absolute_percentage_error(test['Close'].to_list(),
future_forecast['Prediction'].to_list())
print("RMSE of ARIMA model: ", rmse)

```

```

print("MSE of ARIMA model: ", mse)
print("MAPE of ARIMA model: ", mape)

arima_residuals = pd.DataFrame(stepwise_model.arima_res_.resid,
index=data.index[1:], columns=['Residuals'])
arima_residuals.plot(title="Arima residuals")
plt.show()
arima_residuals.plot(title="Arima residuals distribution", kind='kde')
plt.show()

garch_future_forecast = garch_forecast(arima_residuals, test_date, 1, 1)

prediction = future_forecast + garch_future_forecast
pd.concat([test, prediction], axis=1).plot()
plt.title("Bitcoin USD Price Prediction with ARIMA-GARCH")
plt.show()

rmse2 = mean_squared_error(test['Close'].to_list(),
prediction['Prediction'][1:].to_list(), squared=False)
mse2 = mean_squared_error(test['Close'].to_list(),
prediction['Prediction'][1:].to_list())
mape2 = mean_absolute_percentage_error(test['Close'].to_list(),
prediction['Prediction'][1:].to_list())
print("RMSE of ARIMA-GARCH model: ", rmse2)
print("MSE of ARIMA-GARCH model: ", mse2)
print("MAPE of ARIMA-GARCH model: ", mape2)

```

Приложение Е

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import Dense, LSTM, Dropout
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import math

PRICED_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\BTC-
USD.csv"

# Load the stock price data and select the relevant columns
df = pd.read_csv(PRICED_PATH, parse_dates=['Date'])
df = df[['Date', 'Open', 'Close', 'Volume']]

# Take data for 2022
df = df.loc[df["Date"].between("2022-1-1", "2022-12-31")]

# Scale the data using MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
scaled = scaler.fit_transform(df[['Close']])

# Define the lookback sequence length and batch size
lookback = 14
batch_size = 5

# Split the data into training and validation sets
split_index = int(0.8 * len(scaled))
train_data = scaled[:split_index]
val_data = scaled[split_index - lookback:]

# Create the training data sequences
def create_sequences(data):
    X, y = [], []
    for i in range(lookback, len(data)):
```

```

    X.append(data[i - lookback:i, :])
    # y.append(data[i, 1])
    y.append(data[i, 0])
return np.array(X), np.array(y)

train_X, train_y = create_sequences(train_data)
val_X, val_y = create_sequences(val_data)

# Define the model architecture
def buld_model(neurons):
    model = Sequential()
    model.add(LSTM(neurons, input_shape=(lookback, 1)))
    model.add(Dropout(0.2))
    model.add(Dense(1))
    model.compile(optimizer='adam', loss='mean_squared_error')
    return model

# Calculate MAPE
def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

def predict(neurons):
    print("Predictions for LSTM model with { } neurons".format(neurons))
    model = buld_model(neurons)

    # Fit the model on the training data
    history = model.fit(train_X, train_y, epochs=100, batch_size=batch_size,
validation_data=(val_X, val_y), verbose=0)

    # Use the model to predict the stock prices on the validation set
    val_predict = model.predict(val_X)
    val_predict = scaler.inverse_transform(val_predict)
    val_close = scaler.inverse_transform(val_data)

    # Plotting Loss vs Validation loss
    loss = history.history['loss']
    val_loss = history.history['val_loss']

```

```

# Calculate RMSE and MSE
rmse = math.sqrt(mean_squared_error(val_close[lookback:], val_predict))
mse = mean_squared_error(val_close[lookback:], val_predict)
print("RMSE: ", rmse)
print("MSE: ", mse)

mape = mean_absolute_percentage_error(val_close[lookback:], val_predict)
print("MAPE: ", mape)

epochs = range(len(loss))
plt.plot(epochs, loss, 'r', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend(loc=0)
plt.figure()
plt.show()

# Plot the actual and predicted close prices
plt.figure(figsize=(10, 6))
plt.plot(df['Date'][split_index - lookback:], val_close, label='Actual Close Price')
plt.plot(df['Date'][split_index:], val_predict, label='Predicted Close Price')
plt.legend()
plt.xlabel('Date')
plt.ylabel('Close Price')
plt.title('Actual vs. Predicted Close Prices')
plt.show()
print("END")

neurons = [8, 16, 32, 64, 128, 256]

for i in neurons:
    predict(i)
    print("END")

```

Приложение Ж

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import Dense, LSTM, Dropout
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import math

PRICED_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\BTC-
USD.csv"
OCC_PATH = "C:\\Users\\koshe\\PycharmProjects\\BitcoinChainnet\\data\\occ"
start_date = "2022-1-1"
end_date = "2022-12-31"
# Load the stock price data and select the relevant columns
df = pd.read_csv(PRICED_PATH, parse_dates=['Date'])
df = df[['Date', 'Close']]

occ_total = []
date_range = pd.date_range(start_date, end_date)
for date in date_range:
    occ = pd.read_csv(OCC_PATH + "{}-{}-{}".format(date.year, date.month,
date.day) + ".csv", parse_dates=['Date']).iloc[:, 1:]
    occ_total.append(occ)

df = np.concatenate((df, occ_total), axis=1)

# Take data for 2022
df = df.loc[df["Date"].between(start_date, end_date)]

# Scale the data using MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
scaled = scaler.fit_transform(df[['Close']])

# Split the data into training and validation sets
```

```

split_index = int(0.8 * len(scaled))
train_data = scaled[:split_index]
val_data = scaled[split_index:]

# Define the lookback sequence length and batch size
lookback = 14
batch_size = 5

# Create the training data sequences
def create_sequences(data):
    X, y = [], []
    for i in range(lookback, len(data)):
        X.append(data[i - lookback:i, :])
        y.append(data[i, 1])
    return np.array(X), np.array(y)

train_X, train_y = create_sequences(train_data)
val_X, val_y = create_sequences(val_data)

# Define the model architecture
def build_model(neurons):
    model = Sequential()
    model.add(LSTM(neurons))
    model.add(Dropout(0.2))
    model.add(Dense(1))
    model.compile(optimizer='adam', loss='mean_squared_error')
    return model

# Calculate MAPE
def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

def predict(neurons):
    print("Predictions for LSTM model with { } neurons".format(neurons))
    model = build_model(neurons)

# Fit the model on the training data

```

```

history = model.fit(train_X, train_y, epochs=100, batch_size=batch_size,
validation_data=(val_X, val_y), verbose=0)

# Use the model to predict the stock prices on the validation set
val_predict = model.predict(val_X)
val_predict = scaler.inverse_transform(val_predict)
val_close = scaler.inverse_transform(val_data)

# Plotting Loss vs Validation loss
loss = history.history['loss']
val_loss = history.history['val_loss']

# Calculate RMSE and MSE
rmse = math.sqrt(mean_squared_error(val_close[lookback:], val_predict))
mse = mean_squared_error(val_close[lookback:], val_predict)
print("RMSE: ", rmse)
print("MSE: ", mse)

mape = mean_absolute_percentage_error(val_close[lookback:], val_predict)
print("MAPE: ", mape)

epochs = range(len(loss))
plt.plot(epochs, loss, 'r', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend(loc=0)
plt.figure()
plt.show()

# Plot the actual and predicted close prices
plt.figure(figsize=(10, 6))
plt.plot(df['Date'][split_index - lookback:], val_close, label='Actual Close Price')
plt.plot(df['Date'][split_index:], val_predict, label='Predicted Close Price')
plt.legend()
plt.title("Прогнозирование моделью LSTM №2 256 стоимости биткоина")
plt.xlabel("Дата, день")
plt.ylabel("Цена закрытия BTC, доллар США")
plt.show()
print("END")

```

```
neurons = [4, 8, 16, 32, 64, 128, 256]
```

```
for i in neurons:  
    predict(i)  
    print("END")
```