

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

До захисту допущено:

Завідувач кафедри

_____ Олександр РОЛІК

«__» _____ 20__ р.

**Дипломний проєкт
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Інформаційні управляючі системи та
технології»
спеціальності 126 «Інформаційні системи та технології»
на тему: « Система прогнозування ризиків обміну криптовалюти на основі
машинного навчання »**

Виконав:

студент IV курсу, групи ІС–92

Белоруський Олексій Андрійович _____

Керівник:

Доцент кафедри ІСТ, к.т.н

Писаренко Андрій Володимирович _____

Рецензент:

Засвідчую, що у цьому дипломному
проєкті немає запозичень з праць інших
авторів без відповідних посилань.

Студент (-ка) _____

Київ – 2023 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інформаційні управляючі системи та технології»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«__» _____ 20__ р.

ЗАВДАННЯ

на дипломний проєкт студенту

Белоруському Олексію Андрійовичу

1. Тема проєкту «Система прогнозування ризиків обміну криптовалютами на основі машинного навчання», керівник проєкту Писаренко Андрій Володимирович, доцент кафедри ІСТ, к.т.н, затверджені наказом по університету від «31» травня 2023 р. №2101-с,

2. Термін подання студентом проєкту: 12 червня 2023 року

3. Вихідні дані до проєкту: трейдингова стратегія, із показником Net Profit за період у 200 днів не менше 10%

4. Зміст пояснювальної записки:

Аналіз актуальної проблеми прогнозування ризиків обміну криптовалютами та існуючих методів вирішення: загальні відомості, огляд ринку криптовалюти і ризиків гри на ньому, огляд існуючих рішень проблеми. Інформаційне забезпечення: вхідні дані, вихідні дані. Математичне забезпечення: постановка задачі, показники оцінки якості прогнозів, опис обраних методів вирішення задачі. Програмне та технічне забезпечення: засоби розробки, вимоги до технічного забезпечення, структура додатку, тестування додатку.

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)

Діаграма станів. Діаграма потоку даних. Схема мережі. Результати тестування.

6. Дата видачі завдання 1 грудня 2022 року

Календарний план

з/п	Назва етапів виконання проєкту	Термін виконання етапів проєкту	Примітка
1)	Затвердження теми роботи	11.02.23	
2)	Ознайомлення з теоретичним матеріалом та літературою по темі	17.04.23 – 23.04.23	
3)	Огляд існуючих методів вирішення проблеми	24.04.23 – 30.04.23	
4)	Проектування програмного забезпечення	01.05.23 – 07.05.23	
5)	Розробка програмного продукту	08.05.23 – 14.05.23	
6)	Оформлення текстової документації	15.05.23 – 06.06.23	

Студент

Олексій БЄЛОРУСЬКИЙ

Керівник

Андрій ПИСАРЕНКО

АНОТАЦІЯ

Пояснювальна записка містить 32 рисунок, 4 таблиці, та 17 джерел.

Метою бакалаврської роботи є покращення існуючих систем прогнозування ризиків обміну на ринку криптовалют.

Задачами бакалаврської роботи є аналіз існуючих систем, розроблення системи з урахуванням покращень, та оцінка результатів її роботи.

У першому розділі описано основні положення, які необхідні для розуміння ситуації на ринку, і побудову якісних прогнозів, також були розглянуті існуючі методи прогнозування, і пояснений принцип їх роботи.

Другий розділ присвячений вибору середовища розробки та встановленню вхідних і вихідних даних інформаційної системи.

У третьому розділі визначені метрики оцінки точності прогнозів моделей, які будуть використовуватися у роботі, проведений порівняльний аналіз між описаними в першому розділі методами передбачення з метою вибору найкращого, описані та обґрунтовані методи оптимізації результатів інформаційної системи.

Четвертий розділ наводить перелік необхідного технічного та програмного забезпечення для роботи програмного додатку, описує структуру програми та алгоритм її роботи, та проводить тестування результатів на актуальних даних.

ABSTRACT

The explanatory note contains 32 figures, 4 tables, and 17 references.

The aim of the bachelor's thesis is to improve existing risk prediction systems in the cryptocurrency exchange market.

The tasks of the bachelor's thesis include analyzing existing systems, developing a system with improvements, and evaluating its performance.

The first chapter describes the fundamental concepts necessary for understanding the market situation and building accurate forecasts. Existing forecasting methods are also discussed, along with an explanation of their principles of operation.

The second chapter is dedicated to establishing the input and output data of the information system.

In the third chapter, evaluation metrics for model forecast accuracy are defined. A comparative analysis is conducted among the prediction methods described in the first chapter to select the best one. Additionally, all optimization methods for improving the information system's results are described and justified.

The fourth chapter lists the necessary technical and software requirements for the final program. It describes the program's structure, its algorithm, and conducts testing of the results using real-time data.

Номер рядка	Формат	Позначення	Найменування	Кільк. аркушів	Номер скзмп.	Примітка
1			<u>Документація загальна</u>			
2						
3			Знову розроблена			
4						
5	A4	IC92.01БАК.004 ПЗ	Пояснювальна записка	60		
6	A3	IC92.01БАК.004 Д1	Система прогнозування	1		
7			ризиків обміну криптовалют			
8			на основі машинного			
9			навчання. Діаграма станів.			
10	A3	IC92.01БАК.004 Д2	Система прогнозування	1		
11			ризиків обміну криптовалют			
12			на основі машинного			
13			навчання. Діаграма потоків			
14			даних.			
15	A3	IC92.01БАК.004 Д3	Система прогнозування	1		
16			ризиків обміну криптовалют			
17			на основі машинного			
18			навчання. Схема нейронної			
19			мережі.			
20	A3	IC92.01БАК.004 Д4	Система прогнозування	1		
21			ризиків обміну криптовалют			
22			на основі машинного			
23			навчання. Результати			
24			тестування.			
25						
26						
27						
28						
Зм.	Аркуш	№ докум.	Підпис	Дата	І С92.010БАК.004 ТП Система прогнозування ризиків обміну криптовалют на основі машинного навчання Відомість проєкту	
Розроб.		Белоруський О. А.				
Керівн.		Писаренко А. В.				
Затв.						
					Літ.	Аркуш
					Т	1
					КПІ ім. Ігоря Сікорського	
					Група IC-92	

**Пояснювальна записка
до дипломного проекту
на тему: «Система прогнозування ризиків обміну
криптовалют на основі машинного навчання»**

Київ – 2023 року

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ АКТУАЛЬНОЇ ПРОБЛЕМИ ПРОГНОЗУВАННЯ РИЗИКІВ ОБМІНУ КРИПТОВАЛЮТ ТА ІСНУЮЧИХ МЕТОДІВ ВИРІШЕННЯ	6
1.1 Загальні відомості.....	6
1.2 Огляд ринку криптовалют та ризиків гри на ньому	8
1.3 Огляд існуючих рішень проблеми.....	11
1.3.1 Фундаментальний аналіз	11
1.3.2 Соціальний аналіз.....	12
1.3.3 Технічний аналіз.....	14
1.3.3.1 Часові ряди і робота з ними	16
1.3.3.2 Рекурентні нейронні мережі	19
1.3.3.3 Довгостроково-короткочасна пам'ять.....	22
Висновки до розділу.....	27
2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ.....	29
2.1 Вхідні дані.....	29
2.2 Вихідні дані	29
2.3 Вибір інструментів розробки	30
2.3.1 Середовище IntelliJ IDEA	30
2.3.2 Середовище Pycharm	31
2.3.3 Середовище Jupyter Notebook	33
3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ	36
3.1 Постановка задачі.....	36
3.2 Показники оцінки якості прогнозів.....	36
3.2.1 Оцінка моделі SARIMA	38
3.2.2 Оцінка моделі RNN	40
3.2.3 Оцінка моделі LSTM	41
3.3 Опис обраних методів вирішення задачі.....	42

					ІС92.001БАК.004 ПЗ			
		№ докум.	Підпис					
Розроб.	Белоруський О. А.				Система прогнозування ризиків обміну криптовалют на основі машинного навчання. Пояснювальна записка	Літ.	Арк.	Аркушів
Керівн.	Писаренко А. В.					Т	2	60
						КПІ ім. Ігоря Сікорського Група ІС-92		
Затв.								

3.3.1 Підготовка вхідних даних	42
3.3.2 Визначення сигналів.....	45
3.3.3 Сгладження прогнозів	49
Висновки до розділу.....	50
4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ	51
4.1 Засоби розробки.....	51
4.2 Вимоги до технічного забезпечення.....	52
4.3 Структура додатку.....	52
4.4 Тестування додатку	55
ВИСНОВКИ ДО РОБОТИ.....	58
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	59

ВСТУП

Криптовалюти стали однією з найгарячіших тем у світі фінансів в останні роки, про що свідчать надзвичайно високі показники капіталізації ринку криптовалют [1]. За цей час вони зуміли здобути значну популярність та відкрити нові можливості для інвестування та споживання. Однією з відрізняючих рис криптовалют від їх фізичних аналогів є використання потужної криптографії для забезпечення безпеки фінансових операцій, контролю за створенням нових одиниць та перевірки передачі активів. Однак, як і в більшості інших фінансових інструментів, курс криптовалют піддається значним коливанням та непередбачуваним змінам, що спричинено здебільшого високим ступенем недоступності та факторами попиту та пропозиції. Такі умови від інвесторів та економістів активної уваги до аналізу та передбачення розвитку крипторинку.

Висока зацікавленість інвесторів, та значні темпи розвитку крипторинку стимулюють проведення наукових досліджень в області оцінки перспектив розвитку даного ринку. Одним з найбільш перспективних напрямків досліджень у галузі є моделювання та прогнозування руху цін на цьому ринку. Це стає особливо актуальною задачею в умовах значної волатильності та непередбачуваності крипторинку, коли кожен дрібний рух ціни може мати серйозні наслідки для інвесторів.

Наявність надійних моделей прогнозування дозволить інвесторам та трейдерам приймати обґрунтовані торговельні рішення та зменшувати ризики, пов'язані з інвестуванням в криптовалюти. Крім того, це може допомогти залучати нових учасників на ринок та збільшувати обсяги торгівлі. Відтак, розробка надійних та ефективних моделей прогнозування руху цін на крипторинку є важливим завданням для інвесторів, трейдерів та економістів. Вона дозволить зробити обґрунтовані торговельні рішення, зменшити ризики та залучити нових учасників на ринок.

Моделювання та прогнозування руху цін на криптовалютному ринку вимагає використання різних методів та підходів. Важливим елементом є використання

					ІС92.010БАК.004 ПЗ	Арк.
						4
Зм.	Лист	№ докум.	Підпис	Дата		

статистичних методів, машинного навчання та штучного інтелекту для аналізу великої кількості даних, що стосуються курсів криптовалют, технічних індикаторів, новин та інших факторів, що можуть впливати на ціни. Крім того, для побудови надійних моделей прогнозування необхідно провести порівняльний аналіз різних методів та підходів, вибрати найкращі з них для конкретних умов ринку. Також важливо постійно оновлювати та перевіряти моделі на актуальних даних, щоб забезпечити їхню точність та роботу в реальному часі.

Дана робота присвячена розробці та оцінці системи, яка включає у себе передбачення ризиків криптовалют за допомогою нейронних мереж.

Метою бакалаврської роботи є покращення існуючих систем прогнозування ризиків обміну на ринку криптовалют. Задачами бакалаврської роботи є аналіз існуючих систем, розробка системи з урахуванням покращень, та оцінка результатів її роботи.

					ІС92.010БАК.004 ПЗ	Арк.
						5
Зм.	Лист	№ докум.	Підпис	Дата		

1 АНАЛІЗ АКТУАЛЬНОЇ ПРОБЛЕМИ ПРОГНОЗУВАННЯ РИЗИКІВ ОБМІНУ КРИПТОВАЛЮТ ТА ІСНУЮЧИХ МЕТОДІВ ВИРІШЕННЯ

1.1 Загальні відомості

Криптовалюти – це віртуальні активи, функціонування яких забезпечується мережею децентралізованих комп'ютерних вузлів [2]. Криптовалюти відрізняються децентралізованістю, це означає, що їхні мережі не контролює жодна центральна влада, така як уряд чи банк. Замість цього, транзакції обробляються тисячами комп'ютерів-вузлів, які підтримують мережу. Кожен вузол має копію бази даних і перевіряє правильність транзакцій, що гарантує безпеку і захищеність мережі від атак та зловживань. Більшість з них побудовані на основі блокчейну – бази даних про транзакції в ланцюжку пов'язаних блоків.

Технологія блокчейн – з англійської «ланцюг блоків», це послідовність ієрархічно пов'язаних вузлів, названих блоками, кожен з яких містить інформацію про проведену у цьому блоці транзакцію, та хеш поточного та попереднього блоку в ланцюгу [2].

Хеш-функція – це математична функція, яка бере будь-який вхідний текст і перетворює його на фіксований хеш-код або хеш-значення. У контексті блокчейну хеш-функції використовуються для забезпечення безпеки та цілісності блоків даних. Кожен блок містить свій власний хеш-код, який формується на основі даних у блоку. Якщо будь-яка інформація в блоку змінюється, хеш-код також змінюється, що робить неможливим зміну даних без виявлення цього. Хеш-функції також використовуються для створення цифрових підписів та ідентифікації користувачів в мережі блокчейн [3].

Майнери – це люди або організації, які забезпечують безперервну роботу мережі криптовалют за допомогою виконання складних математичних обчислень. Ці обчислення потрібні для підтвердження транзакцій і створення нових блоків в блокчейні. Майнери отримують нагороду за свою працю в формі нових монет (Proof of Work або block reward) або комісійних від транзакцій, які вони обробляють [2].

					ІС92.010БАК.004 ПЗ	Арк.
						6
Зм.	Лист	№ докум.	Підпис	Дата		

Proof of Work (PoW) – протокол консенсусу, широко використовуваний у системах блокчейн, включаючи мережі криптовалют. Він передбачає вирішення обчислювально складної задачі для підтвердження та додавання нових блоків до блокчейну. Майнери конкурують, щоб знайти розв'язок, виконуючи велику кількість обчислень, що вимагає значної обчислювальної потужності та енергії. У протоколі Proof of Work майнери повинні надати доказ того, що вони виконали певний обсяг роботи, знаходячи хеш-значення, яке відповідає певним критеріям. Цей процес включає повторне хешування даних блоку з нонсом (випадковим числом) до того моменту, коли отримане хеш-значення задовольняє складність, встановлену мережею. Складність регулюється для забезпечення постійного часу створення блоків. Безпека Proof of Work ґрунтується на його асиметрії. Перевірка правильності розв'язку відносно проста, але його знаходження вимагає значних обчислювальних зусиль. Ця властивість допомагає запобігти зловживанню зловмисниками, оскільки для зміни блокчейну потрібна велика обчислювальна потужність для перезапису всього ланцюжка. Незважаючи на те, що Proof of Work ефективний у забезпеченні безпеки мережі, він має свої недоліки. Він споживає значну кількість енергії, що створює побоювання стосовно його впливу на довкілля. Крім того, зі зростанням конкуренції у галузі майнінгу потрібне спеціалізоване обладнання, що може призвести до централізації майнінгової потужності в руках кількох головних учасників. Незважаючи на ці обмеження, Proof of Work довів свою надійність як механізм консенсусу, особливо в разі Bitcoin, де він успішно використовується з самого початку [3].

Безпека транзакцій в блокчейні – забезпечена тим фактом, що кожен користувач мережі має свою копію бази даних з усіма транзакціями, що були здійснені у ній. Це означає, що, якщо хтось намагається змінити інформацію про будь-яку транзакцію в блокчейні, то це буде видно всім іншим користувачам мережі. Коли майнери створюють новий блок, вони перевіряють вміст усіх транзакцій, що були здійснені на мережі, і переконуються, що вони є правильними. Якщо будь-який запис у блокчейні змінюється, це призведе до порушення послідовності хешів, що стоять за кожною транзакцією, і зміни цих хешів у всіх

					ІС92.010БАК.004 ПЗ	Арк.
						7
Зм.	Лист	№ докум.	Підпис	Дата		

наступних блоках. Це дуже важко зробити, оскільки це вимагає зламування криптографічних алгоритмів, що застосовуються у блокчейні, що займе дуже багато часу і ресурсів. Отже, блокчейн є надзвичайно безпечним і надійним способом зберігання інформації про транзакції [3].

Волатильність – це міра коливання цін активів на ринку криптовалют. Вона вказує на рівень мінливості цін і відображає ступінь ризику, пов'язаного зі змінами цінкових рівнів. Волатильність криптовалют може мати як позитивний, так і негативний вплив на трейдерів та інвесторів. З одного боку, висока волатильність може створити можливості для заробітку, оскільки великі цінові зміни можуть привести до швидкого зростання вартості активів. З іншого боку, висока волатильність також може призводити до значних втрат і ризику нестабільності. Волатильність криптовалют може бути виміряна різними способами, наприклад, за допомогою стандартного відхилення цін на певний період, історичного розмаху цін або за допомогою індикаторів, таких як ATR (Average True Range) або Bollinger Bands.

Bollinger Bands (Лінії Боллінжера) – це популярний індикатор технічного аналізу, який використовується для вимірювання волатильності ринку та ідентифікації можливих точок звороту цін. Вони складаються з трьох ліній, що побудовані на основі середньої ковзної (наприклад, 20-денної простої середньої) та стандартного відхилення цін. Центральна лінія представляє середню ковзну, а верхня та нижня лінії розташовані на певну відстань (зазвичай два стандартних відхилення) від центральної лінії. Bollinger Bands можуть допомогти інвесторам та трейдерам зрозуміти, коли ринок перебуває у періоді високої волатильності або стабільності. Звуження ліній вказує на низьку волатильність, тоді як розширення ліній свідчить про збільшену волатильність.

1.2 Огляд ринку криптовалют та ризиків гри на ньому

Ринок криптовалют є досить нестабільним і змінюється залежно від багатьох факторів, таких як новини про прийняття криптовалют у різних країнах, зміна регуляторної політики, складність добування блоків, геополітичні кризи та інші.

					ІС92.010БАК.004 ПЗ	Арк.
						8
Зм.	Лист	№ докум.	Підпис	Дата		

Зараз найбільшою криптовалютою за капіталізацією є Bitcoin, його ринкова капіталізація перевищує 500 мільярдів доларів США [4].

Окрім Bitcoin, на ринку існує безліч інших криптовалют, таких як Ethereum, Binance Coin, Dogecoin, Cardano, Ripple та інші. Стабільність різних криптовалют може суттєво варіюватись в залежності від їх популярності та капіталізації. Серед них Bitcoin та Ethereum є більш популярними та великими і зазвичай мають більшу стабільність, оскільки вони мають широке використання, активну спільноту користувачів та великі обсяги торгівлі.

Ці валюти є ліквідними на ринку, тобто великі обсяги купівлі-продажу дозволяють забезпечити стабільну цінову динаміку. Вони зазвичай менш вразливі до впливу маніпуляцій та шахрайства, оскільки мають більшу кількість учасників та суворішу регуляторну політику.

З іншого боку, менш популярні криптовалюти можуть бути більш нестабільними. Менша популярність приводить до меншої капіталізації та нижчої ліквідності, що у свою чергу може призвести до значних коливань цін у короткі терміни. Під час аналізу таких валют можуть бути дуже помітний вплив маніпуляцій та недостатня регуляція. Ці зміни можуть показувати себе навіть на відносно коротких проміжках часу. На рисунках 1.1 та 1.2 показані графіки відсоткової зміни цін на валюти BTC(Bitcoin) та XRP(Ripple) у доларах США за останній місяць, щоб проілюструвати думку.

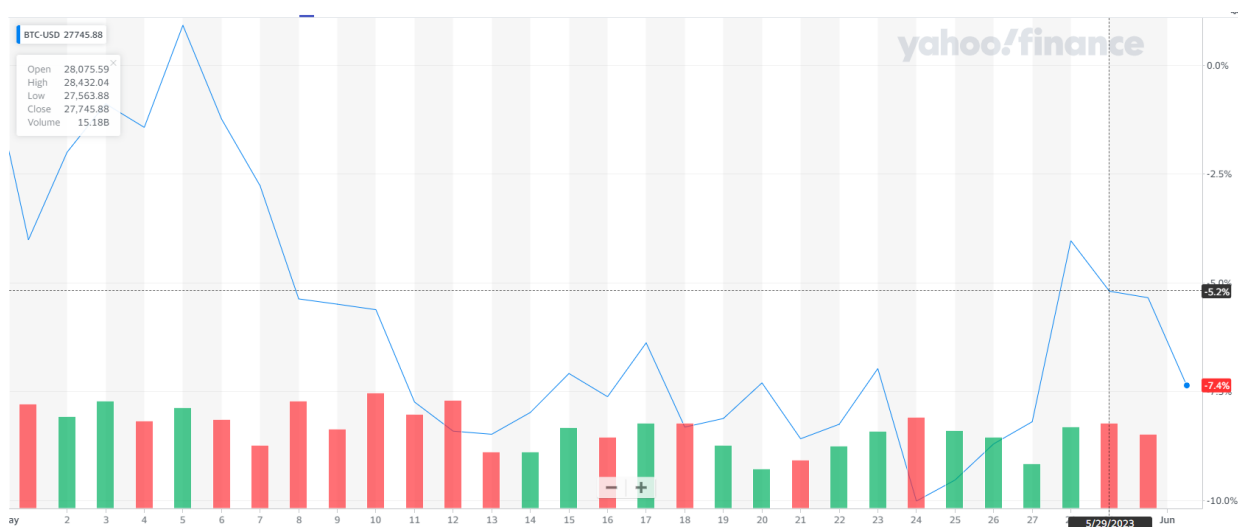


Рисунок 1.1 – Зміна BTC за місяць

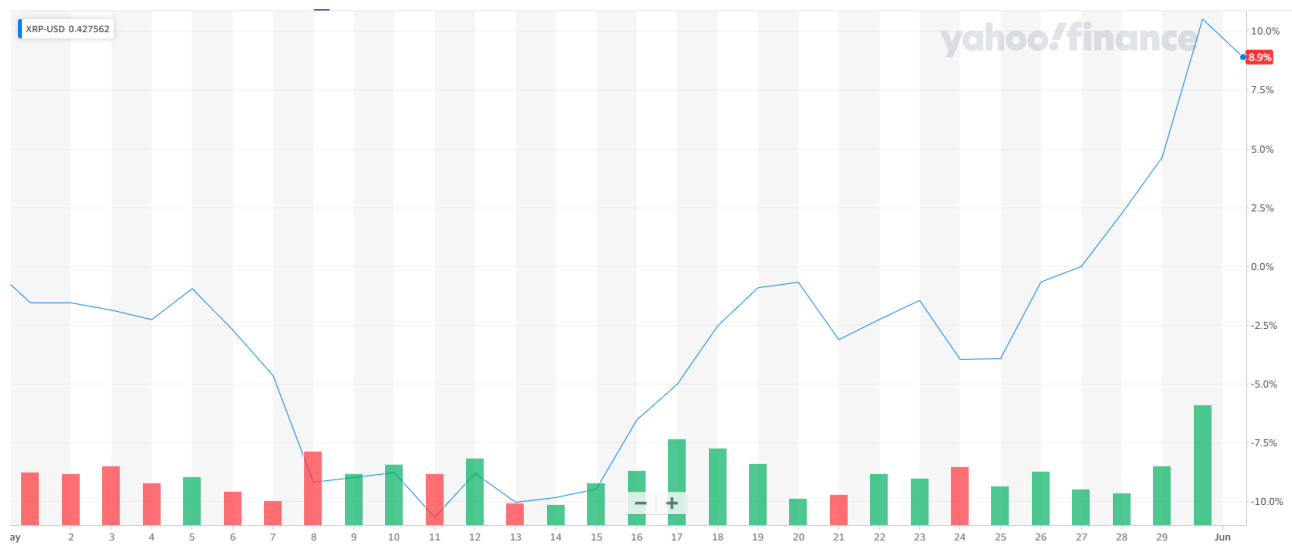


Рисунок 1.2 – Зміна XRP за місяць

Як видно з рисунків, коливання біткоїну за місяць були у межах -10% та 0% від вартості, у той час як Ripple коливається між -10% та $+10\%$.

Однак, варто зазначити, що стабільність криптовалют також може залежати від інших факторів, таких як новини, події на ринку, регуляторна політика тощо. Тому, незалежно від популярності валюти, інвестори повинні бути готові до волатильності ринку та здійснювати власний аналіз для прийняття обґрунтованих рішень. Структура ринку криптовалют може бути порівняна зі структурою акційного ринку: існують лідери, які мають велику капіталізацію та є найбільш ліквідними, та менші криптовалюти, які можуть мати високий потенціал росту, але більш вразливі до ризиків.

Одним зі значних ризиків гри на біржі криптовалют є висока волатильність цін. Це означає, що ціни на криптовалюти можуть різко змінюватись у короткі терміни, що може призвести до значних збитків для інвесторів. Крім того, ринок криптовалют може стати об'єктом маніпулювання та шахрайства, так як його регулювання не настільки строго, як у традиційних фінансових ринках. Отже, інвестори, які бажають грати на біржі криптовалют, повинні бути готові до високих ризиків та проводити власні дослідження та аналіз, щоб зробити інформовані і обґрунтовані рішення.

1.3 Огляд існуючих рішень проблеми

Існує декілька методів передбачення курсу криптовалют, для роботи потрібно розглянути такі з них як фундаментальний аналіз, соціальний аналіз та технічний аналіз.

1.3.1 Фундаментальний аналіз

Фундаментальний аналіз ринку криптовалют – це підхід до аналізу, який спрямований на оцінку інвестиційної привабливості криптовалют на основі їх фундаментальних факторів. Він зосереджений на дослідженні та оцінці основних аспектів, таких як:

- Технологічний потенціал. Оцінка технологічних особливостей криптовалюти, таких як масштабованість, швидкість транзакцій, консенсусний алгоритм тощо. Вивчається як унікальність технології в порівнянні з конкурентами та її можливість вирішення реальних проблем.

- Індустрійна адопція. Аналізується рівень прийняття криптовалюти в різних секторах економіки та бізнес-середовищах. Партнерства з відомими компаніями, використання криптовалюти в реальних сценаріях та рівень масової адопції грають важливу роль.

- Команда розробників. Оцінка досвіду, кваліфікації та репутації команди, яка стоїть за проектом криптовалюти дуже важливі для іміджу валюти та того, як її сприймають у спільноті трейдерів. Розглядаються партнерства з відомими компаніями, співпраця з академічними установами та спеціалістами.

- Регуляторні фактори. Оцінка регуляторного середовища та законодавчого статусу криптовалют в різних країнах та юрисдикціях. Вивчається підхід урядів до криптовалют, рівень регулювання, правовий статус та можливі перешкоди для розвитку та використання криптовалют.

- Споживчий попит та використання. Оцінка реального попиту на

					ІС92.010БАК.004 ПЗ	Арк.
						11
Зм.	Лист	№ докум.	Підпис	Дата		

криптовалюту та її використання в різних сценаріях. Аналізується прийняття криптовалют у різних сферах, таких як електронна комерція, платіжні системи, фінансові послуги та інші. Враховується потенційний обсяг ринку та ріст популярності криптовалют у суспільстві.

1.3.2 Соціальний аналіз

Цей метод передбачення використовує соціальні мережі, форуми та інші онлайн-джерела для вивчення настроїв та думок споживачів щодо певної криптовалюти. Соціальний аналіз ринку криптовалют є важливою складовою вивчення і передбачення курсу криптовалют. Він досліджує вплив соціальних факторів на коливання цін і загальну поведінку ринку. Один із важливих аспектів соціального аналізу - це вивчення суспільного настрою та думок стосовно криптовалют. Соціальні мережі, форуми, новинні портали та інші джерела інформації можуть надати цінні дані про ставлення громадськості до криптовалют, новини та події, які можуть впливати на їх курс [6]. Соціальний аналіз також досліджує роль впливових осіб та експертів у сфері криптовалют. Думки відомих інвесторів, трейдерів, аналітиків та просто впливових осіб можуть мати значний вплив на курс криптовалют. Аналіз їхніх заяв, прогнозів та рекомендацій може допомогти у формуванні уявлення про майбутні тенденції ринку. Один із відомих прикладів події, коли висказування інфлюенсера мала прямий вплив на курс криптовалюти, стосується криптовалюти Dogecoin та відомого підприємця та мільярдера Ілона Маска. На початку 2021 року, Ілон Маск, який є активним користувачем Twitter і має значний вплив у криптогрошовому просторі, почав активно висловлюватись про Dogecoin. У своїх твітах він називав його "любимою криптовалютою" та публікував різні коментарі та меми про неї. На перше квітня того ж року про «відправлення» Dogecoin на місяць, що було сприйнято деякими трейдерами як натяк щодо інвестиції у валюту, і як наслідок за один день її ціна зросла на 20%. Подібні висловлювання викликали значний резонанс серед шанувальників Маска та криптогрошової спільноти. Відразу після того, як Маск

					ІС92.010БАК.004 ПЗ	Арк.
						12
Зм.	Лист	№ докум.	Підпис	Дата		

почав активно згадувати Dogecoin, ціна цієї криптовалюти стрімко зросла. Інвестори та торговці реагували на його висловлювання, купуючи більше Dogecoin та підтримуючи його популярність. У результаті ціна Dogecoin відчувала суттєве зростання, а ринкова капіталізація криптовалюти збільшилась [7]. На рисунку 1.3 показана зміна курсу валюти у USD за квітень 2021го року.

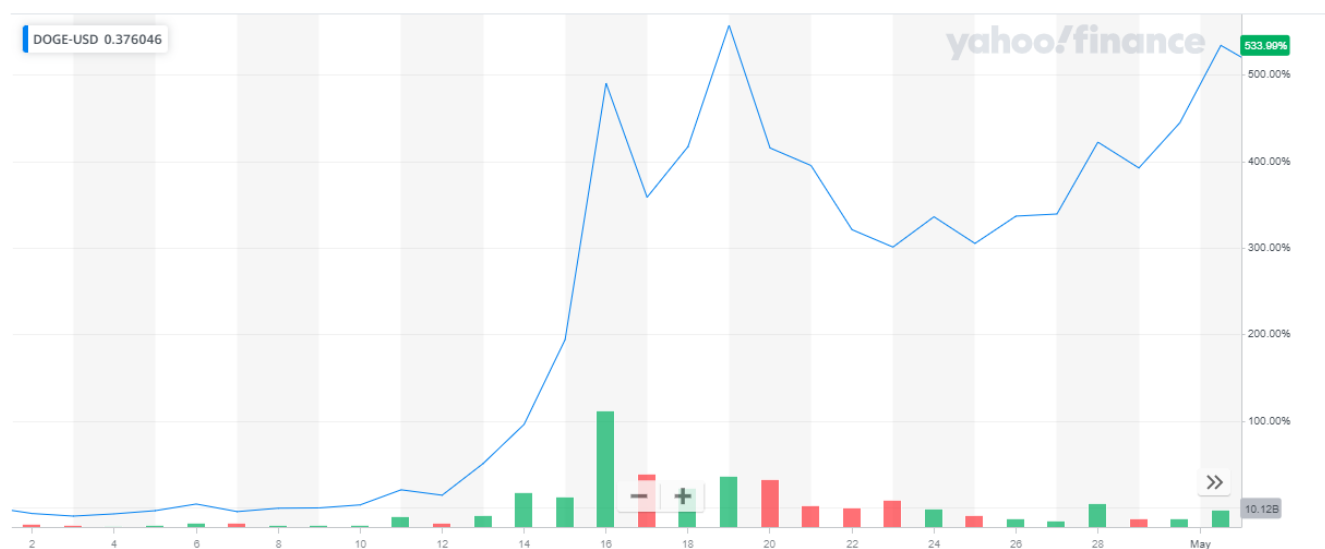


Рисунок 1.3 – Історичні дані DOGE–USD у період 4/1/2021–5/1/2021

До соціального аналізу також відносять аналіз змін в законодавстві, регуляторного впливу, санкцій, скандалів та інші соціальних факторів, які можуть суттєво впливати на курс криптовалют. Вивчення цих взаємозв'язків допомагає прогнозувати можливі ризики та тенденції розвитку ринку. Однак важливо зауважити, що соціальний аналіз має свої обмеження. Суспільні настрої та думки можуть бути дуже мінливими, і їх вплив на ринок може бути складним для передбачення. Крім того, соціальні дані можуть бути піддані маніпуляціям та фейковій інформації, що може призвести до недостовірних результатів. Узагалі, соціальний аналіз ринку криптовалют доповнює інші методи аналізу, такі як технічний та фундаментальний аналіз, і допомагає зрозуміти ширший контекст та взаємозв'язки, що впливають на курс криптовалют.

1.3.3 Технічний аналіз

Технічний аналіз ринку криптовалют – це підхід до аналізу, який базується на вивченні історичних цінових та обсягових даних для прогнозування майбутніх цінових рухів та ідентифікації торговельних можливостей. Він використовує різні технічні інструменти та графіки для аналізу цінових шаблонів, трендів, підтримок та опору, індикаторів та інших параметрів.

Технічний аналіз ринку криптовалют має за основу теорію Доу [5], тобто він заснований на припущенні, що ціна відображає всю інформацію, доступну на ринку, і що історичні цінові дані можуть надати вказівки про майбутні рухи цін. Деякі з популярних технічних інструментів та показників, що застосовуються в технічному аналізі криптовалют, включають:

- Графіки свічок. Свічкові графіки надають інформацію про відкриті, закриті, найвищі та найнижчі ціни активу протягом певного періоду часу. Вони дозволяють аналізувати цінові шаблони, такі як довгі тіла свічок, зворотні формації, зв'язки між тілом та тінями свічок тощо.

- Технічні індикатори. Існує широкий спектр технічних індикаторів, таких як середня ковзна, стохастичний осцилятор, MACD (середня ковзна збіжність/розбіжність) та RSI (індекс відносної сили) [8]. Нижче описаний принцип роботи даних індикаторів, а також їх призначення та особливості.

- а) Середня ковзна обчислюється взяттям суми цін на протязі заданого періоду та поділення на кількість періодів.

- б) Стохастичний осцилятор (Stochastic Oscillator) є популярним індикатором технічного аналізу, який використовується для визначення перекупленості або перепроданості ринку, а також для виявлення можливих змін в ціновому тренді. Цей індикатор працює на основі порівняння поточної ціни активу зі спектром його цін на певний період. Стохастичний осцилятор має дві лінії - %K та %D: %K лінія відображає відносне положення поточної ціни активу у відношенні до діапазону цін за певний період. Ця лінія може коливатися від 0 до 100. %D лінія є згладженою середньою %K лінії і допомагає усунути шум та

					ІС92.010БАК.004 ПЗ	Арк.
						14
Зм.	Лист	№ докум.	Підпис	Дата		

забезпечити більш гладкі сигнали.

в) MACD (Moving Average Convergence Divergence) є популярним індикатором технічного аналізу, який використовується для виявлення змін у цінових трендах та генерації сигналів купівлі-продажу. Він складається з трьох компонентів: MACD-лінії, сигнальної лінії та гістограми. Основна концепція MACD полягає в порівнянні двох експоненційних середніх ковзних (ЕМА) різних періодів. Зазвичай, на MACD використовуються дві ЕМА - швидка ЕМА і повільна ЕМА. Процес розрахунку MACD включає наступні кроки: Обчислення швидкої ЕМА: Швидка ЕМА обчислюється на основі цін закриття активу за короткий період. Обчислення повільної ЕМА: Повільна ЕМА обчислюється на основі цін закриття активу за довгий період.

г) RSI (Relative Strength Index) є індикатором технічного аналізу, який використовується для вимірювання сили та швидкості цінових змін активу та виявлення перекупленості або перепроданості ринку. RSI зазвичай виражається у відсотках і може мати діапазон від 0 до 100. Обчислення приросту ціни: Обчислюється різниця між поточною ціною та попередньою ціною закриття. Розділення приросту ціни на позитивний та негативний: Всі позитивні прирости ціни відображаються окремо, а всі негативні прирости - окремо. Обчислення середнього приросту ціни: Обчислюється середнє значення позитивних та негативних приростів ціни за встановлений період. Обчислення відношення сили: Відношення сили (RS) визначається як відношення середнього позитивного приросту ціни до середнього негативного приросту ціни. Обчислення RSI: RSI обчислюється шляхом використання формули $100 - (100 / (1 + RS))$. Це дозволяє згладити коливання цін та виділити загальний тренд. Ці індикатори допомагають ідентифікувати перекуплені або перепродані ринки, трендові рухи, зміну моментуму та інші торгівельні сигнали

– Підтримка та опір. Аналіз підтримки та опору визначає рівні, на яких ціна має тенденцію зупинитися або змінювати свій рух. Підтримка – це рівень, нижче якого ціна має тенденцію не опускатися, а опір – це рівень, вище якого ціна має тенденцію не підніматися. Аналізуючи історичні дані, можна виявити ці рівні та

					ІС92.010БАК.004 ПЗ	Арк.
						15
Зм.	Лист	№ докум.	Підпис	Дата		

використовувати їх для прийняття торговельних рішень.

– Торговельні моделі. Існує ряд торговельних моделей та стратегій, що використовуються в технічному аналізі криптовалют. Наприклад, "двійковий сигнал" (binary signal) – стратегія, що використовує визначення специфічних цінових шаблонів, які можуть служити сигналом для входу чи виходу з позиції.

– Торговельні індикатори на основі обсягу. Деякі технічні індикатори враховують обсяги торгівлі для отримання додаткової інформації про рух цін. Наприклад, "OBV" (он-балансний обсяг) відображає кумулятивні зміни обсягів, що можуть допомогти виявити потенційні зміни тренду.

– Моделі та теорії ринкового циклу. Деякі аналітики використовують моделі та теорії ринкового циклу, такі як "теорія Елліота" (Elliot Wave Theory), для виявлення циклічності та прогнозування цінових рухів.

У даній роботі розглянутий напрям технічного аналізу як – аналіз часових рядів, і методи, які пов'язані з ними.

1.3.3.1 Часові ряди і робота з ними

Найпопулярнішим напрямом технічного аналізу на сьогодні є аналіз часових рядів. Часовий ряд є послідовністю даних, які вимірюються впродовж різних моментів часу та упорядковуються за часом. Це можуть бути спостереження, зроблені в одному і тому ж місці, але в різні періоди, наприклад, щоденні ціни акцій, місячні продажі або щогодинні температури. Часові ряди можуть мати різну тривалість, від декількох точок до десятиліть чи навіть століть [9].

Аналіз часових рядів – це процес дослідження статистичних властивостей часових рядів з метою виявлення закономірностей, трендів, циклів, сезонності та інших варіацій, що можуть бути присутніми в даних. Основні цілі аналізу часових рядів включають розуміння структури ряду, прогнозування майбутніх значень, виявлення аномалій та розробку моделей, які можуть описати та пояснити поведінку ряду [9].

У роботі з часовими рядами прийнято виділяти три складові: тренд,

					ІС92.010БАК.004 ПЗ	Арк.
						16
Зм.	Лист	№ докум.	Підпис	Дата		

циклічність, похибка. Тренд у часовому ряду – це довгостроковий плавний зсув у значеннях ряду, що відображає загальний напрям змін у часі. Він відображає основну тенденцію росту або спаду ряду протягом тривалого періоду часу, незалежно від короткострокових коливань і сезонності [10]. Циклічність в часовому ряді відноситься до довготривалої зміни значень ряду, яка повторюється протягом певного періоду часу. Цикл може бути нерегулярним та мати змінну амплітуду та тривалість. Часто від циклічності відокремлюють також таку складову часового ряду як сезонність, яка є практично тим самим, але циклічні зміни мають бути прив’язані до таких періодичних факторів як місяці, квартали або дні тижня. Важливо зазначити що сезонні коливання мають фіксовану амплітуду та тривалість [10].

Одною з найпоширеніших моделей аналізу часових рядів є модель ARIMA, а також її розширення SARIMA.

$$ARIMA(p, d, q) = AR(p) + I(d) + MA(q), \quad (1.1)$$

де $AR(p)$ – це авторегресійна частина;

$I(d)$ – інтегрована частина;

$MA(q)$ – це частина з ковзним середнім(moving average).

– Авторегресійна частина – це частина прогнозу, яка враховує залежність поточного значення від деяких p попередніх значень того ж ряду.

– Інтегрована частина – це частина прогнозу, яка враховує похідну попередніх p значень даного ряду, тобто частина яка намагається передбачити не саме значення, а те як зміниться попереднє значення.

– Ковзне середнє – це метод згладжування ряду, який використовує замість поточної точки – середнє з p попередніх значень. Такий ряд буде мати на порядок менше шуму, але відставати від реальних даних на $p/2$ значень.

Основним методом оптимізації рядів ковзних середніх є експоненційне згладжування. Ідея методу – додавання вагів під час обрахунку середніх, надаючи перевагу найбільш новим даним.

					ІС92.010БАК.004 ПЗ	Арк.
						17
Зм.	Лист	№ докум.	Підпис	Дата		

Формула обрахунку коефіцієнтів у загальному випадку виглядає так:

$$EMA_n = \alpha * EMA_{n-1} + (1 - \alpha) * y_n, \quad (1.2)$$

$$EMA_0 = y_0, \quad (1.3)$$

де EMA_n – експоненціальне ковзне середнє (exponential moving average) на кроку n ;

α – параметр експоненціального ковзного середнього, $0 < \alpha < 1$, чим ближче до одиниці, тим менше впливають попередні значення на поточне середнє;

y_n – n -те значення ряду у межах розглянутого вікна.

У моделі $ARIMA(p, d, q)$, де p відповідає порядку авторегресії, $AR(p)$, авторегресія визначається наступною формулою:

$$X_{(t)} = c + \phi_1 X_{(t-1)} + \phi_2 X_{(t-2)} + \dots + \phi_p * X_{(t-p)} + \varepsilon_{(t)}, \quad (1.4)$$

де $X_{(t)}$ – поточне значення часового ряду;

c – константа або зсув;

$\phi_1, \phi_2, \dots, \phi_p$ – коефіцієнти авторегресії;

$X_{(t-1)}, X_{(t-2)}, \dots, X_{(t-p)}$ – попередні значення часового ряду з інтервалом p ;

$\varepsilon_{(t)}$ – шум або помилка в поточному моменті часу.

Коефіцієнти авторегресії ($\phi_1, \phi_2, \dots, \phi_p$) відображають, як сильно кожне попереднє значення впливає на поточне значення. Значення p в моделі $ARIMA$ вказує на кількість попередніх значень, які враховуються для прогнозування поточного значення. Більші значення p означають, що модель більше залежить від попередніх значень, що може виявити складніші залежності в ряду, однак частіше за все p не перевищує 2 [9].

$SARIMA$ (Seasonal Autoregressive Integrated Moving Average) є розширенням моделі $ARIMA$, яке враховує сезонність в часовому ряді. У моделі $SARIMA(p, d, q)(P, D, Q, s)$, де (p, d, q) - порядок несезонного $ARIMA$ компоненту, (P, D, Q) - порядок сезонного $ARIMA$ компоненту та s - період сезонності. Індекс s вказує на

період сезонності, тобто на кількість часових кроків, після яких з'являється патерн, що повторюється. Наприклад, якщо сезонність виявляється кожні 12 кроків, $s = 12$.

1.3.3.2 Рекурентні нейронні мережі

Такі моделі на основі часових рядів, такі як ARIMA або SARIMA, використовують статистичні методи для моделювання та прогнозування часових рядів. Вони засновані на попередніх значеннях ряду та статистичних залежностях. Ці моделі добре підходять для деяких типів часових рядів зі стабільними та прогнозованими патернами. Однак, моделі на основі машинного навчання використовують більш гнучкий підхід, що дає можливість виявити складні залежності, необхідні для аналізу складних та непередбачуваних часових рядів. Отже, розглянемо основні такі моделі.

На відміну від моделей ARIMA та SARIMA, рекурентні нейронні мережі (RNN) мають кілька переваг, які роблять їх особливо ефективними для роботи з послідовними даними, такими як текст, мовні послідовності або часові ряди. Основні переваги RNN включають:

- Здатність моделювати складні нелінійні залежності: RNN є потужними моделями, які можуть моделювати складні нелінійні залежності в послідовних даних. Вони здатні автоматично виявляти і узагальнювати складні взаємозв'язки між даними, що дозволяє їм краще адаптуватися до зміни у розподілі даних та моделювати складні шаблони, які можуть бути складні для ARIMA або SARIMA.

- Здатність працювати зі змінною довжиною послідовності: RNN можуть працювати зі послідовностями різної довжини. Вони не вимагають фіксованої довжини вхідних даних, що робить їх більш гнучкими для моделювання послідовних даних, які можуть мати змінну довжину в часі. У моделях ARIMA та SARIMA потрібно попередньо обробляти та адаптувати дані до фіксованого розміру, що може бути проблематичним у деяких ситуаціях.

Як можна здогадатися з назви мережі, нейрони у ній – рекурентні, тобто на відміну від звичайних неройнів (рисунок 1.4) мають зворотні зв'язки, якими вони

					IC92.010БАК.004 ПЗ	Арк.
						19
Зм.	Лист	№ докум.	Підпис	Дата		

можуть передавати інформацію до самих себе (рисунок 1.5), але на наступному відрізку часу. Саме таким чином досягається «пам'ять» нейронної мережі, якої не мали аналоги [11].

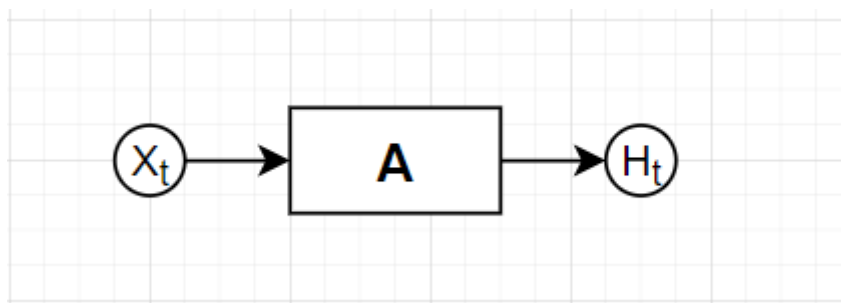


Рисунок 1.4 – Ілюстрація звичайного нейрона

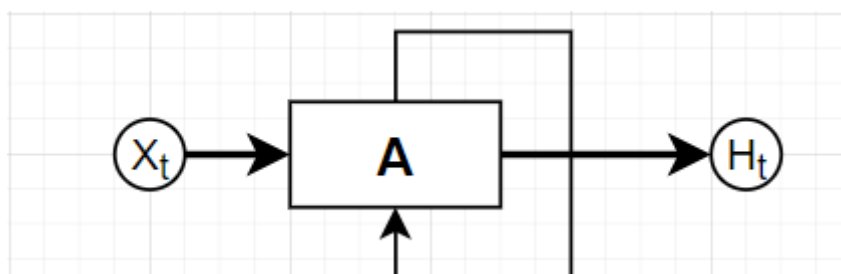


Рисунок 1.5 – Ілюстрація рекурентного нейрона

На рисунку 1.4 та 1.5: X_t – дані що надходять на вхід нейрона, A – функція активації, H_t – прогноз, який виводить нейрон.

Інтерпретувати рисунок 1.5 стає легше якщо розгорнути його по часу, як показано на рисунку 1.6:

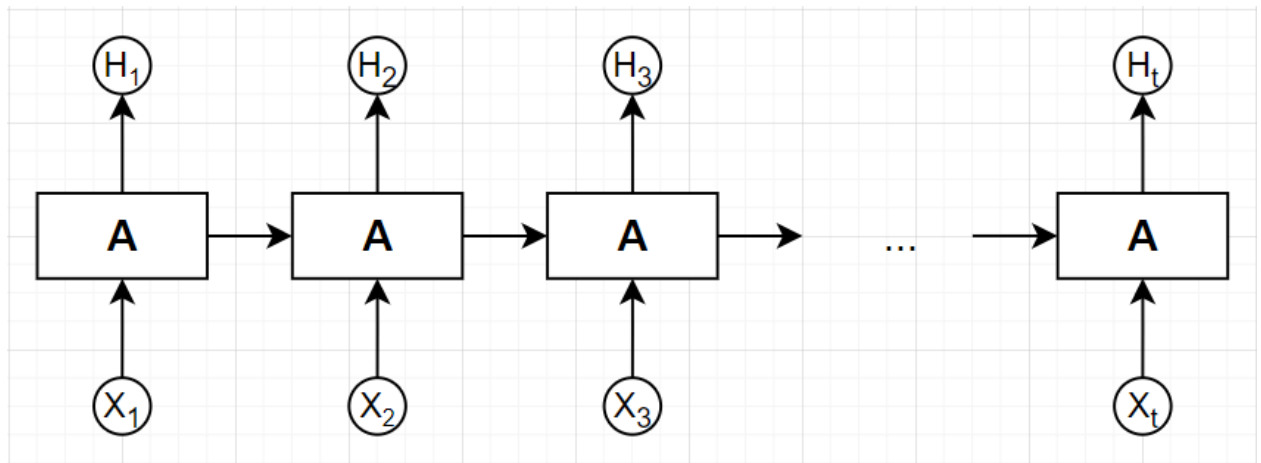


Рисунок 1.6 – Розгортка рекурентного нейрона

У зв'язку з наявністю примітивної форми довгострокової пам'яті, рекурентні нейронні мережі натомість отримали недолік, під назвою вибуховий/зникаючий градієнт (exploding/fading gradient). Ці проблеми виникають під час процесу обчислення градієнту (вагів, за допомогою яких і відбувається зв'язок між нейронами на різних ділянках часу) для оновлення параметрів мережі під час зворотного поширення помилки (Backpropagating), яке лежить у основі навчання нейронної мережі [11].

Проблема вибухового градієнту полягає в тому, що градієнти, які передаються від вихідного шару до вхідного шару, збільшуються з кожним кроком назад. Це може призвести до значної нестабільності в процесі навчання, коли градієнти стають дуже великими і накопичуються на початкових шарах мережі. Це може призвести до вибуху ваг і неконтрольованої зміни параметрів мережі.

Проблема затухаючого градієнту виникає, коли градієнти навпаки зменшуються з кожним кроком назад, що означає, що ваги на початкових шарах мережі вони будуть дуже близькі до нуля, і як наслідок майже не будуть оновлюватися під час навчання. Це може призвести до недостатньої здатності мережі до врахування довгострокових залежностей і погіршення її прогностичних здібностей. Ці проблеми можуть бути особливо помітними в глибоких RNN або в довгих послідовностях.

Одним із рішень цих проблем є метод відсікання(або обмеження) градієнту (Gradient Clipping). Метод відсікання градієнтів полягає у встановленні

максимального порогу для значень градієнтів. Якщо якась компонента градієнту перевищує цей поріг, то її значення обрізається до максимально допустимого значення. Це дозволяє утримувати значення градієнтів в межах прийнятних меж і забезпечує більш стабільний процес навчання [12].

Хоча відсікання градієнтів є корисним методом для зменшення проблем вибухових та затухаючих градієнтів, він також має свої недоліки:

- Зміна орієнтації градієнту: Обмеження градієнтів може змінювати напрямок градієнтного вектора. Наприклад, якщо градієнтний вектор має початкове значення $[0.9, 100.0]$ а обмеження стоїть на одиниці, після обмеження він стане $[0.9, 1.0]$. Що майже обов'язково призведе до втрати інформації про оригінальну структуру градієнту.

- Вплив на здатність збереження інформації: відсікання градієнтів може впливати на здатність мережі зберігати довгострокову інформацію. Значення градієнтів можуть бути обмеженими так, що деякі важливі градієнти, які відповідають за довгострокову залежність, можуть бути погашені або знехтувані.

- Залежність від порогу: Встановлення правильного порогу для обмеження градієнтів є гіперпараметром, який потребує налаштування. Неправильний вибір порогу може призвести до недостатнього обмеження або надто сильного обмеження градієнтів, що може негативно вплинути на процес навчання.

- Вплив на швидкість навчання: Обмеження градієнтів може вплинути на швидкість збіжності алгоритму навчання. Якщо градієнти обмежені занадто сильно, це може зменшити кількість інформації, яку мережа може отримати за одну епоху навчання, і як наслідок модель може потребувати більше ітерацій для досягнення оптимальних параметрів [12].

1.3.3.3 Довгостроково-короткочасна пам'ять

Трохи більш розвинутий варіант вирішення проблеми з затухаючими та вибухаючими градієнтами є мережі з довгостроково-короткочасною пам'яттю (Long-Short Term Memory). У порівнянні з і звичайними рекурентними мережами, LSTM переносить від нейрона до нейрона дві змінні, як можна здогадатися із назви,

					IC92.010БАК.004 ПЗ	Арк.
						22
Зм.	Лист	№ докум.	Підпис	Дата		

одна відповідає за короткочасну пам'ять, а інша – за довгострокову пам'ять. Кожен нейрон LSTM нагадує окрему маленьку нейронну мережу, і має додаткові елементи, такі як вентиль «забути»(forget gate), вхідний вентиль(input gate) та вихідний вентиль(output gate) (рисунок 1.7), які контролюють потік інформації через мережу.

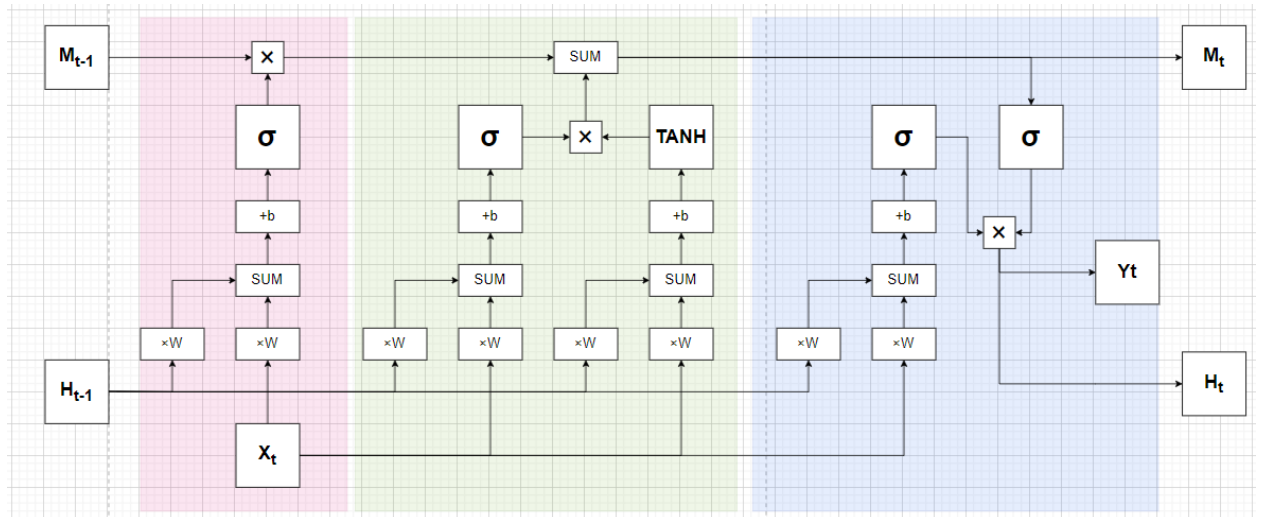


Рисунок 1.7 – Схема нейрону мережі LSTM

– Forget gate – визначає яка частина довгострокової пам'яті має бути забута, в залежності від нових даних, перемножуючи короткочасну пам'ять (прогноз на минулий відрізок часу) на конкатинацію вхідних даних, пропущену через сигмоїдну функцію активації, та перемножаючи це на довгострокову пам'ять.

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) * M_t, \quad (1.5)$$

де f_t – вектор вентилля «забути»;

M_t – вектор довгострокової пам'яті;

σ_g – сигмоїдна функція;

x_t – вектор вхідних даних;

h_{t-1} – вектор даних у короткочастній пам'яті;

W_f , U_f та b_f – матриці вагів та вектор зміщень.

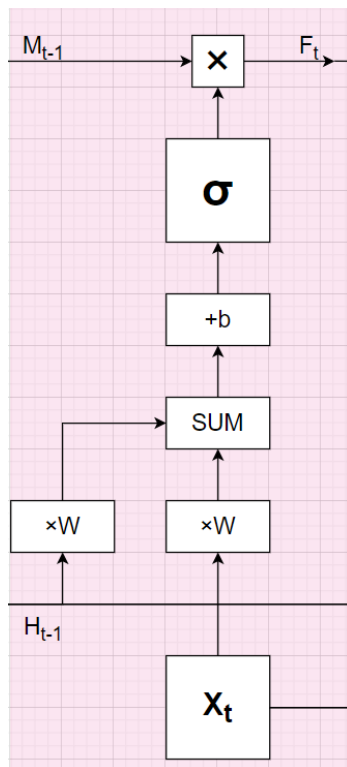


Рисунок 1.8 – Схема Forget Gate

– Input gate – це вентиль, який визначає яка частина даних, поданих на вхід нейрону буде записана у довгострокову пам'ять, перемножуючи конкатинацію вхідних даних та короткочасної пам'яті з попереднього кроку пропущену через функцію активації гіперболічного тангенсу (потенціальна довгострокова пам'ять) і конкатинацію вхідних даних та короткочасної пам'яті з попереднього кроку пропущену через сигмоїдну функцію активації (відсоток потенціальної довгострокової пам'яті що запам'ятовується), і конкатинуючи це все з вектором довгострокової пам'яті, який було обчислено з forget gate на попередньому кроці роботи нейрону (див. 1.5).

$$i_t = (\sigma_g(W_p x_t + U_p h_{t-1} + b_p) \times (\sigma_c W_l x_t + U_l h_{t-1} - 1 + b_l) * W_k) + f_t, \quad (1.6)$$

де i_t – вектор вхідного вентиля;

f_t – вектор вентиля «забути»;

σ_c – функція гіперболічного тангенсу;

σ_g – сигмоїдна функція;

x_t – вектор вхідних даних;

h_{t-1} – вектор даних у короткочастній пам'яті;

W_p, W_l, W_k, U_p, U_l – матриці вагів.

b_p , та b_l – вектори зміщень.

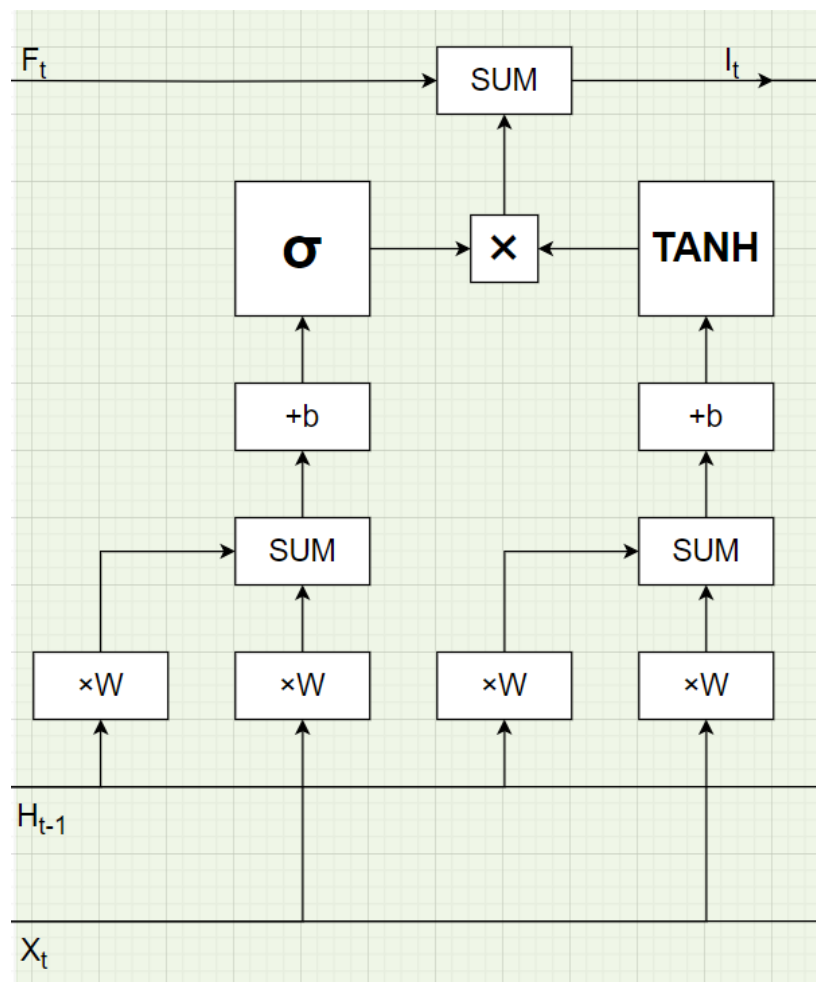


Рисунок 1.9 – Схема Input Gate

– Output gate – або вихідний вентиль, як видно з назви, визначає який прогноз зробить даний нейрон LSTM. Результатом обчислення у Output gate є добуток

сигмоїди від конкатенації вхідних даних та минулого прогнозу на вектор довготривалої пам'яті, який було обчислено на минулому кроці (див. формула 1.6), пропущений через функцію активації гіперболічного тангенса. Таким чином визначається прогноз, який йде одночасно і на вихід до наступного нейрону, і зберігається у короткочасну пам'ять, для використання на наступному кроці.

$$F_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r) * \sigma_c i_t, \quad (1.7)$$

де f_t – вектор вихідного вентилля;

i_t – вектор вхідного вентилля;

σ_c – функція гіперболічного тангенсу;

σ_g – сигмоїдна функція;

x_t – вектор вхідних даних;

h_{t-1} – вектор даних у короткочасній пам'яті;

W_r, U_r – матриці вагів;

b_r – вектори зміщень.

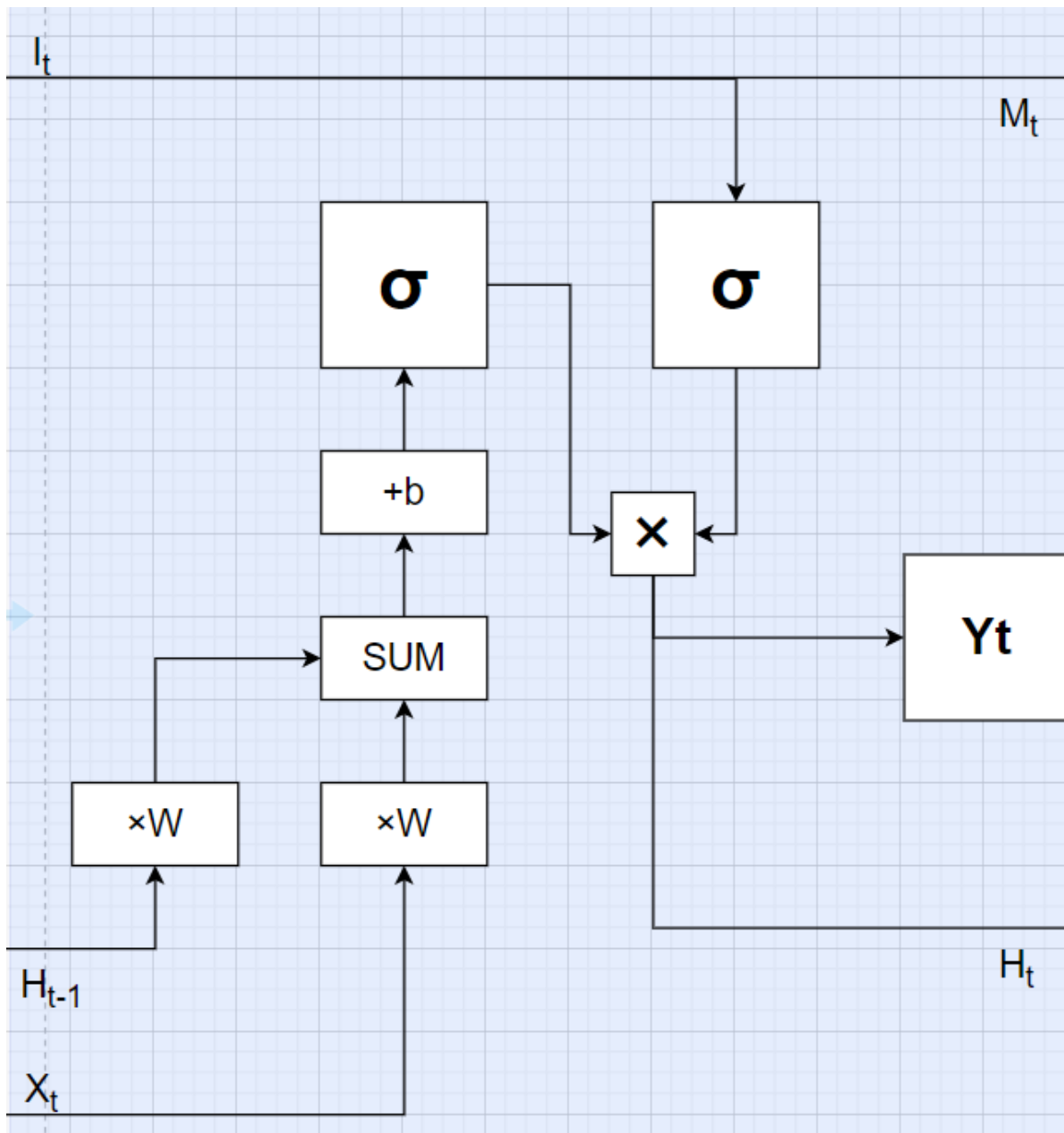


Рисунок 1.10 – Схема Output Gate

Висновки до розділу

У даному розділі розглянуті різноманітні аспекти пов'язані з ринком криптовалют та грою на ньому. Починаючи з загальних положень, надані теоретичні відомості щодо криптовалютного ринку і його особливостей. Описані поняття часових рядів і їх важливість для аналізу ринку. Часовий ряд - це послідовність даних, зібраних у проміжках часу, і він використовується для вивчення та передбачення змін у цінах криптовалют.

Далі висвітлюється питання аналізу часових рядів. Розглядаються різні методи аналізу, зокрема фундаментальний, соціальний та технічний аналіз. Фундаментальний аналіз включає в себе оцінку факторів, які можуть впливати на ціни криптовалют, таких як новини, економічні показники, регуляторні рішення тощо. Соціальний аналіз розглядає вплив суспільної активності, громадської думки та інших соціальних факторів на ринок криптовалют. Технічний аналіз використовує різні індикатори та графіки для виявлення тенденцій та формування прогнозів.

Детально розглядаються популярні технічні індикатори, які широко використовуються при аналізі ринку криптовалют. Ці індикатори допомагають інвесторам та трейдерам знаходити потенційні точки входу та виходу з позицій, визначати перекуплені та перепродані ринки, а також виявляти зміни тренду.

Окрім того, наведений опис та механізм роботи моделей прогнозування, які використовуються для прогнозування цін на ринку криптовалют. Зокрема, розглядається авторегресивна інтегрована модель ковзного середнього (ARIMA) та її сезонна варіація (SARIMA). Ці моделі базуються на статистичних методах і дозволяють прогнозувати майбутні зміни цін на основі попередніх значень. Також розглядаються рекурентні нейронні мережі (RNN) та їх варіації, зокрема мережі з довготривало-короткочасною пам'яттю (LSTM). Ці моделі використовуються для аналізу послідовних даних і володію здатністю розпізнавати складні залежності в часових рядах.

У розділі також детально порівнюються недоліки та переваги цих моделей прогнозування, а також розглядаються їх різні варіації і модифікації. Аналізується точність та ефективність кожної моделі з метою вибору оптимального рішення для подальшого прогнозування цін на ринку криптовалют.

					ІС92.010БАК.004 ПЗ	Арк.
						28
Зм.	Лист	№ докум.	Підпис	Дата		

ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Вхідні дані

Для застосунку який розроблюється існує два типи вхідних даних зі сторони користувача – це криптовалюта, для якої користувач бажає отримати прогноз, та API-ключ користувача для інтернет-ресурсу <https://rapidapi.com/apidojo/api/yahoo-finance1>. Зі сторони програми на вхід подаються історичні дані цін та об'ємів обміну криптовалютами, взяті з вище названого ресурсу, отримані у форматі json (JavaScript Object Notation), на рисунку 2.1.

```
▼ {} 6 keys
  ▼ prices: [] 251 items
    ▼ 0: {} 7 keys
      date: 1686058200
      open: 1.2200000286102295
      high: 1.3099999942779541
      low: 1.2200000286102295
      close: 1.2599999904632568
      volume: 2345200
      adjclose: 1.2599999904632568
    ► 1: {} 7 keys
    ► 2: {} 7 keys
    ► 3: {} 7 keys
    ► 4: {} 7 keys
```

Рисунок 2.1 – Приклад вигляду перших рядків вхідних даних

2.2 Вихідні дані

Вихідними даними застосунку буде сигнал «Чекати», «Купувати» або «Продавати», обраний на основі прогнозів системи та методів, описаних у наступних розділах.

					ІС92.010БАК.004 ПЗ	Арк.
						29
Зм.	Лист	№ докум.	Підпис	Дата		

2.3 Вибір інструментів розробки

Три найпопулярніших середовища для роботи з неймережами – це IntelliJ IDEA, PyCharm та Jupyter Notebook. Кожне з них має свої особливості та переваги, у цьому підпункті буде розглянуто і обрано найкраще з трьох.

2.3.1 Середовище IntelliJ IDEA

IntelliJ IDEA є потужним інтегрованим середовищем розробки (IDE), яке може бути використане для розроблення неймереж та роботи з датасетами. Воно надає широкий спектр інструментів і функціональності, які полегшують роботу з неймережами та обробку даних. Ось деякі розгорнуті аспекти використання IntelliJ IDEA для розроблення неймереж та роботи з датасетами:

- Підтримка мови програмування: IntelliJ IDEA підтримує Python, що робить його ідеальним інструментом для розробки неймереж. Середовище дозволяє використовувати різні фреймворки, бібліотеки та інструменти для неймереж, такі як TensorFlow, Keras, PyTorch, Theano та інші.

- Редагування коду та автодоповнення: IntelliJ IDEA надає потужні засоби для редагування коду, автодоповнення, рефакторингу та аналізу коду. Це допомагає прискорити процес розробки неймереж, забезпечуючи швидку навігацію, виправлення помилок та полегшення написання складного коду.

- Налагодження коду: IntelliJ IDEA має вбудовані інструменти для налагодження коду, які допомагають виправляти помилки, аналізувати значення змінних, відстежувати виклики функцій та контролювати виконання неймереж. Це значно спрощує процес відлагодження та забезпечує більшу точність при розробці неймереж.

- Управління проектами та залежностями: IntelliJ IDEA надає зручний інтерфейс для управління проектами, включаючи створення та організацію неймереж, датасетів та інших компонентів. У середовищі можна легко керувати залежностями, встановлювати пакети та налаштовувати середовище розробки для

					IC92.010БАК.004 ПЗ	Арк.
						30
Зм.	Лист	№ докум.	Підпис	Дата		

оптимальної роботи з нейромережами.

- Робота з датасетами: IntelliJ IDEA підтримує розширення для роботи з датасетами. Є можливість імпортувати, аналізувати та візуалізувати дані в різних форматах. Зручний редактор даних дозволяє виконувати операції з даними, які можуть бути корисні при підготовці даних для нейромереж.

- Інтеграція з іншими інструментами: IntelliJ IDEA може легко інтегруватись з іншими популярними інструментами для розробки нейромереж, такими як Jupyter Notebook, Git, Docker та інші. Це дозволяє використовувати найкращі інструменти для кожної частини розробки нейромережі.

В цілому, IntelliJ IDEA надає розширені можливості для розроблення нейромереж та роботи з датасетами. Воно допомагає розробникам прискорити процес розробки, полегшити написання складного коду та забезпечити зручну роботу з датасетами. Завдяки своїм функціональним можливостям і підтримці Python, IntelliJ IDEA стає популярним вибором для розроблення нейромереж та аналізу даних [13].

2.3.2 Середовище Pycharm

PyCharm – це потужне інтегроване середовище розробки (IDE), спеціально призначене для розроблення проектів з використанням мови програмування Python, включаючи розробку нейромереж та роботу з датасетами. Ось деякі розгорнуті аспекти використання PyCharm для цих цілей:

- Підтримка Python та наукових бібліотек: PyCharm надає повноцінну підтримку відомих бібліотек, таких як TensorFlow, PyTorch, Keras, NumPy, SciPy та багатьох інших. Це дозволяє легко використовувати ці бібліотеки для розроблення нейромереж та аналізу даних.

- Інтелектуальне автодоповнення та перевірка коду: PyCharm надає потужні інструменти автодоповнення коду, аналізу та перевірки синтаксису. Вони допомагають розробникам писати код швидше та надійніше, надаючи підказки, виправлення помилок та організацію коду.

					IC92.010БАК.004 ПЗ	Арк.
						31
Зм.	Лист	№ докум.	Підпис	Дата		

– Візуальне налагодження коду: PyCharm має вбудовані інструменти для візуального налагодження коду. У цьому режимі можливо ставити точки зупинки, крокувати по коду, переглядати значення змінних та виконувати налагодження нейромереж. Це допомагає виявляти та виправляти помилки під час розробки.

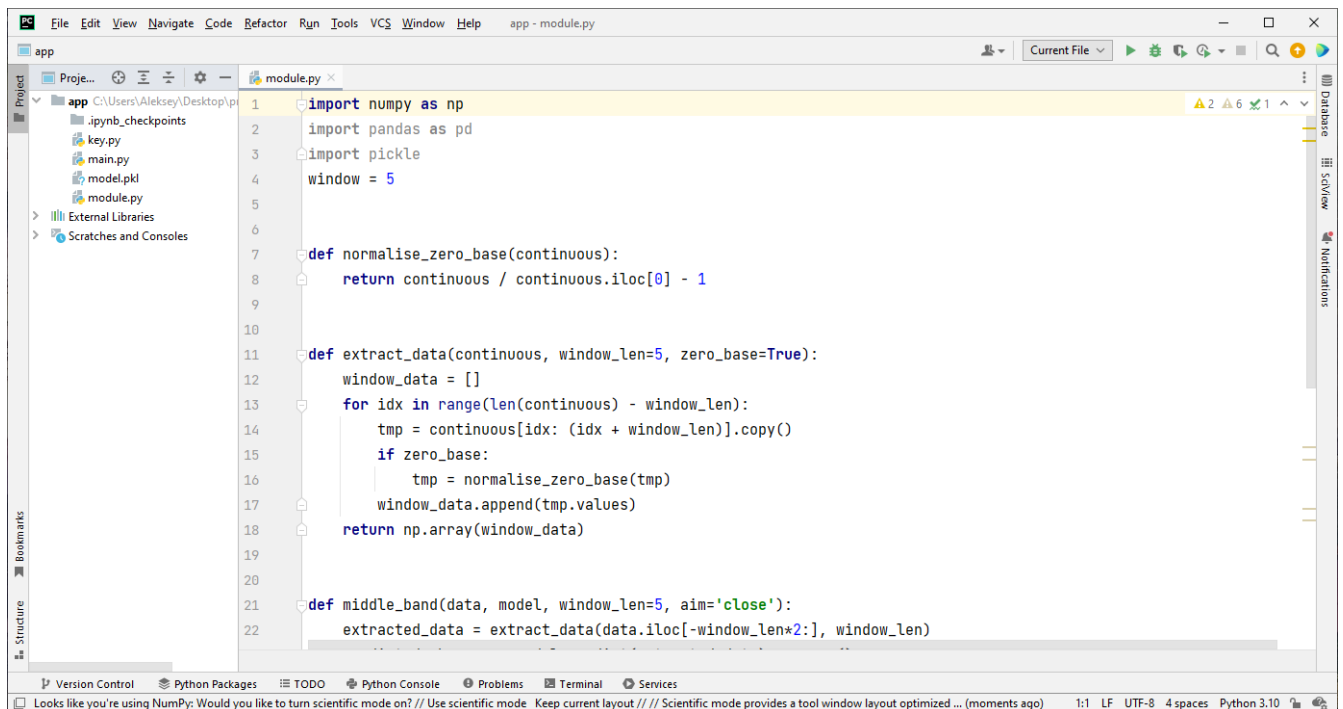


Рисунок 2.2 – Приклад інтерфейсу Pycharm

– Управління проектами та залежностями: PyCharm має зручний інтерфейс для управління проектами, включаючи створення, організацію та налаштування нейромереж та датасетів. Надається можливість керувати залежностями, встановлювати пакети та організовувати структуру проекту для зручної роботи з нейромережами.

– Робота з датасетами: PyCharm надає засоби для роботи з датасетами. Можливо імпортувати, аналізувати та візуалізувати дані в різних форматах. Зручний редактор даних дозволяє виконувати операції з даними, які можуть бути корисні при підготовці даних для нейромереж.

– Інтеграція з іншими інструментами: PyCharm може легко інтегруватись з іншими інструментами, такими як Jupyter Notebook, Git, Docker та інші. Це

					ІС92.010БАК.004 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		32

дозволяє використовувати різноманітні інструменти для розробки неймереж та спрощує спільну роботу з командою.

Загалом, PyCharm є потужним та зручним середовищем для роботи з неймережами та датасетами. Його можливості підтримки Python, інтелектуального автодоповнення, візуального налагодження, управління проектами та роботи з даними допомагають розробникам зосередитися на ефективній розробці неймереж та аналізу даних [14].

2.3.3 Середовище Jupyter Notebook

Jupyter Notebook – це інтерактивне середовище розробки, що дозволяє об'єднувати в одному документі код, текст та візуалізацію результатів. Воно широко використовується для вивчення та аналізу даних, а також наукових досліджень. Ось деякі розгорнуті аспекти використання Jupyter Notebook для розроблення неймереж та роботи з датасетами:

- Інтерактивне програмування: Jupyter Notebook дозволяє виконувати код почастинно, використовуючи комірки. Це дозволяє проводити експерименти з неймережами, змінювати параметри та спостерігати за результатами без необхідності виконувати весь код заново. Це особливо корисно під час розроблення та налагодження неймереж.

- Візуалізація даних: Jupyter Notebook має вбудовану підтримку для візуалізації даних, що дозволяє швидко та зручно відображати графіки, діаграми та інші візуальні елементи. Що дуже важливо при аналізі даних та відображенні результатів роботи неймереж.

```

In [1]: from tensorflow.keras.models import Sequential
        from tensorflow.keras.layers import Activation, Dense, Dropout, LSTM
        import matplotlib.pyplot as plt
        import numpy as np
        import pandas as pd
        from sklearn.metrics import mean_absolute_error

In [2]: original_data = pd.read_csv(filepath_or_buffer='BTC-USD (1).csv', index_col='Date', parse_dates=True)
        data = original_data.loc[:, ['Close']]
        end_date = '2016-01-01'
        test_start_date = '2015-12-1'
        data = data[:end_date]
        data.head()

```

```

Out[2]:

```

Date	Close
2014-09-17	457.334015
2014-09-18	424.440002
2014-09-19	394.795990
2014-09-20	408.903992
2014-09-21	398.821014

Рисунок 2.3 – Приклад інтерактивної візуалізації даних

– Робота з датасетами: Jupyter Notebook надає зручний інтерфейс для завантаження та обробки датасетів. Полегшений процес імпортування даних, їх обробки, візуалізації, та підготовки даних для тренування нейромереж. Комірки Jupyter Notebook дозволяють проводити ітеративні дії з даними та спостерігати за їх впливом на результати.

– Документація та спільна робота: Jupyter Notebook дозволяє створювати документи, що поєднують код, текст та візуалізацію. Можна додавати пояснення, коментарі та описи до кожної комірки коду, що полегшує розуміння та спільну роботу з іншими розробниками чи науковцями. Важливо зазначити що Jupyter Notebook також підтримує мову розмітки HTML, що надає більше опцій для форматування документації.

```

In [2]: from IPython.core.display import display, HTML
        display(HTML('<h1>Hello, world!</h1>'))

```

Hello, world!

Рисунок 2.4 – Приклад вбудовування розмітки HTML у Jupyter Notebook

Для аналізу даних дуже високо цінується їх візуалізація та можливість інтерактивно перезапускати окремі частини коду, зберігаючи увесь контекст, тож для виконання цієї роботи було обрано Jupyter Notebook у якості середовища розробки, як найбільш зручна альтернатива [15].

Висновки до розділу

У даному розділі були розглянуті аспекти інформаційного забезпечення проекту. Вхідні дані для програми включають історичні дані з сайту Yahoo Finance та API-ключ користувача. Історичні дані з сайту Yahoo Finance можуть містити цінову інформацію, обсяги торгів та інші релевантні дані про криптовалютні активи. API-ключ користувача необхідний для отримання доступу до відповідних функціональних можливостей. Вихідні дані програми включають сигнали "утримуватися", "купувати" та "продавати". Ці сигнали є результатом аналізу і обробки вхідних даних та використовуються для прийняття рішень щодо торгівлі криптовалютами. Сигнал "утримуватися" може вказувати на рекомендацію не змінювати поточні позиції, "купувати" означає рекомендацію на покупку активу, а "продавати" - рекомендацію на продаж активу.

У розділі також були розглянуті та порівняні три різні середовища розробки: IntelliJ IDEA, PyCharm та Jupyter Notebook. При виборі середовища розробки для даного проекту було обрано Jupyter Notebook. Головною причиною цього вибору є можливість інтерактивного редагування та перегляду коду. Jupyter Notebook надає зручний інтерфейс для створення та виконання кодових клітинок, які можна використовувати для аналізу даних, візуалізації та отримання результатів. Це дозволяє розробникам ефективно працювати з даними та тестувати різні алгоритми аналізу безпосередньо в середовищі розробки.

Загалом, використання Jupyter Notebook в даному проекті забезпечує зручність та ефективність розробки, що є важливими факторами при роботі з аналізом та прогнозуванням ринку криптовалют.

					ІС92.010БАК.004 ПЗ	Арк.
						35
Зм.	Лист	№ докум.	Підпис	Дата		

3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Постановка задачі

Ця робота спрямована на створення системи передбачення ризиків обміну криптовалютой, поєднуючи звичайні трейдингові стратегії з прогнозами, отриманими за допомогою методів аналізу часових рядів.

Наступний підрозділ фокусується на знаходженні моделі, яка надає найбільш точні прогнози для використання їх у кінцевій системі, для досягнення цього буде проведений порівняльний аналіз обраних методів за допомогою показників, перелічених нижче. Далі буде описано методи оптимізації обраної моделі, які покращать результати як прогнозів, так і роботи системи взагалі.

3.2 Показники оцінки якості прогнозів

Для оцінки якості прогнозу часових рядів існує декілька показників, які допомагають визначити, наскільки точним є прогноз порівняно зі справжніми значеннями часового ряду. Ось деякі з найпоширеніших показників:

– Mean Squared Error (MSE): Це середня квадратична помилка між прогнозованими і справжніми значеннями. Вона вимірює середньоквадратичну величину різниці між прогнозом і фактичним значенням.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2, \quad (3.1)$$

де n – кількість елементів у виборці;

y_i – прогноз i -того елемента;

x_i – реальне значення i -того елемента.

– Root Mean Squared Error (RMSE): Це квадратний корінь з MSE. RMSE дозволяє оцінити помилку прогнозу в тих же одиницях, що і сам часовий ряд.

					ІС92.010БАК.004 ПЗ	Арк.
						36
Зм.	Лист	№ докум.	Підпис	Дата		

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2}, \quad (3.2)$$

де n – кількість елементів у виборці;

y_i – прогноз i -того елемента;

x_i – реальне значення i -того елемента.

– Mean Absolute Percentage Error (MAPE): Це середня абсолютна відсоткова помилка між прогнозованими і справжніми значеннями. Вона вимірює середню відсоткову різницю між прогнозом і фактичним значенням.

$$MAE = \frac{\sum_{i=1}^n \left| \frac{y_i - x_i}{x_i} \right|}{n} * 100\%, \quad (3.3)$$

де n – кількість елементів у виборці;

y_i – прогноз i -того елемента;

x_i – реальне значення i -того елемента.

– R2 Score (Coefficient of Determination): Це статистичний показник, який вимірює пропорцію варіації залежної змінної, що пояснюється моделлю прогнозу. R2 Score може приймати значення від 0 до 1, де 1 вказує на ідеальний прогноз, а 0 - на жодну пояснювальну здатність моделі.

$$R^2 = 1 - \frac{\sum_{i=1}^n (x_i - y_i)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}, \quad (3.4)$$

де n – кількість елементів у виборці;

y_i – прогноз i -того елемента;

$\bar{x}$ – середнє значення по виборці;

x_i – реальне значення i -того елемента.

					ІС92.010БАК.004 ПЗ	Арк.
						37
Зм.	Лист	№ докум.	Підпис	Дата		

3.2.1 Оцінка моделі SARIMA

По перше потрібно визначити параметри моделі, і якщо оптимальні параметри p , d , q , P , D та Q майже завжди знаходяться у невеликих рамках, і з наявними ресурсами можуть бути просто підібрані, то для параметру сезонності треба звернути увагу на графіки автокореляції та часткової автокореляції звичайних та логарифмованих, та диференційованих даних (рисунки 3.1–3.3) [17].

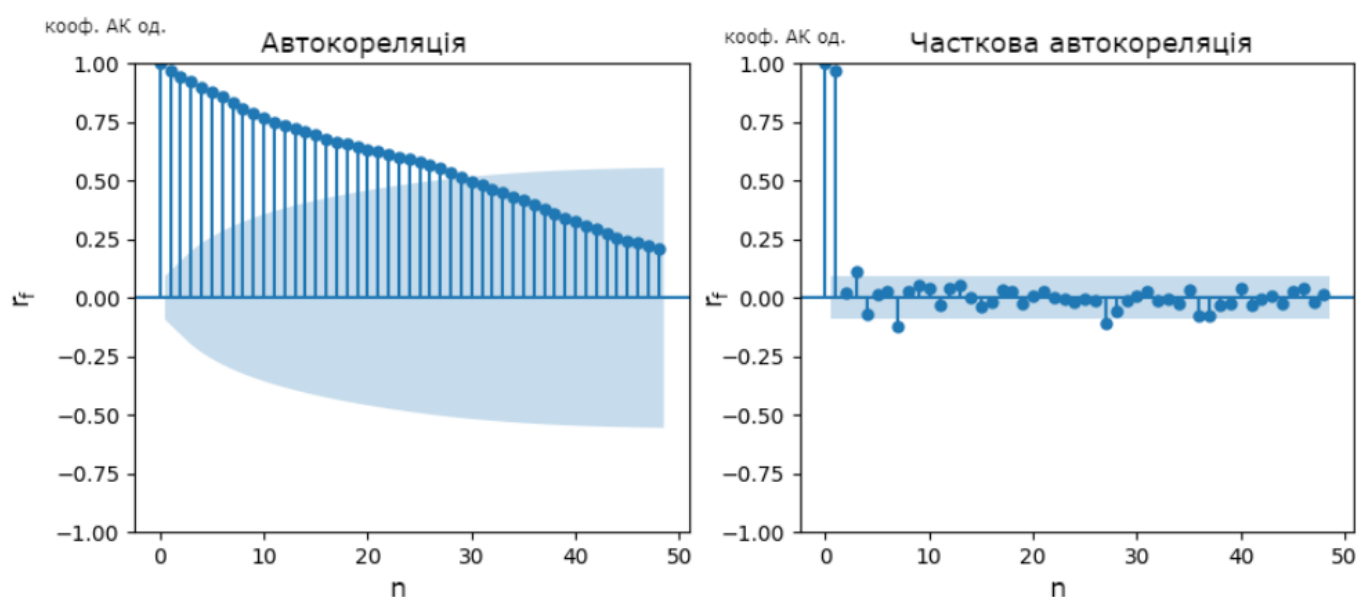


Рисунок 3.1 – Графіки автокореляції необроблених даних

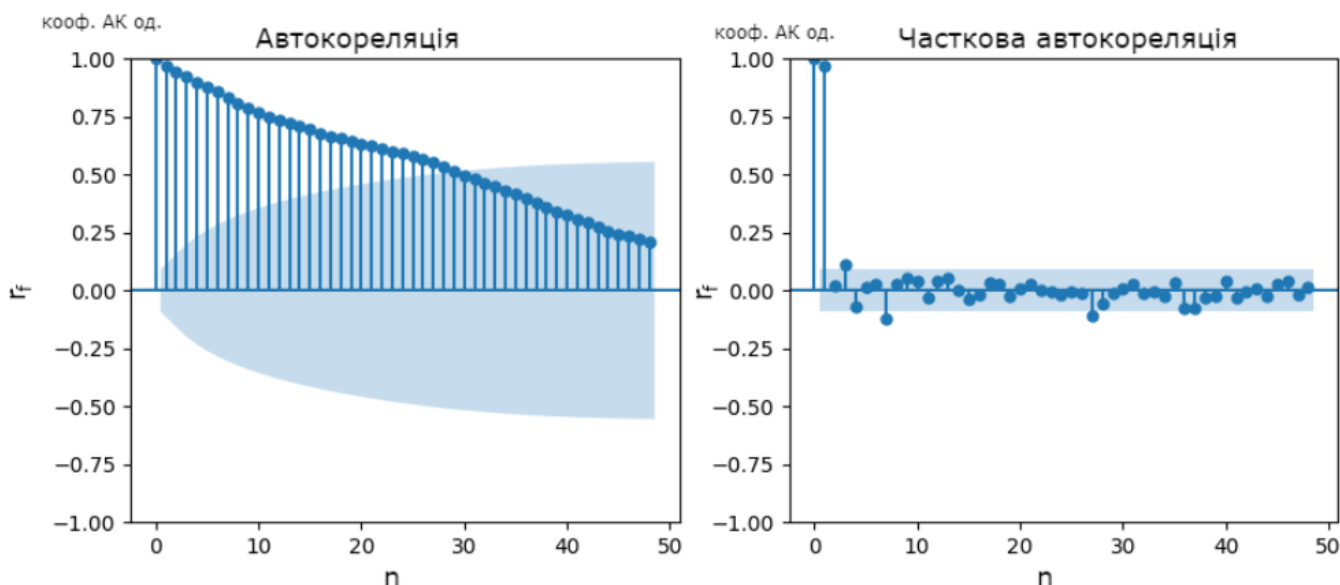


Рисунок 3.2 – Графіки автокореляції логарифмованих даних

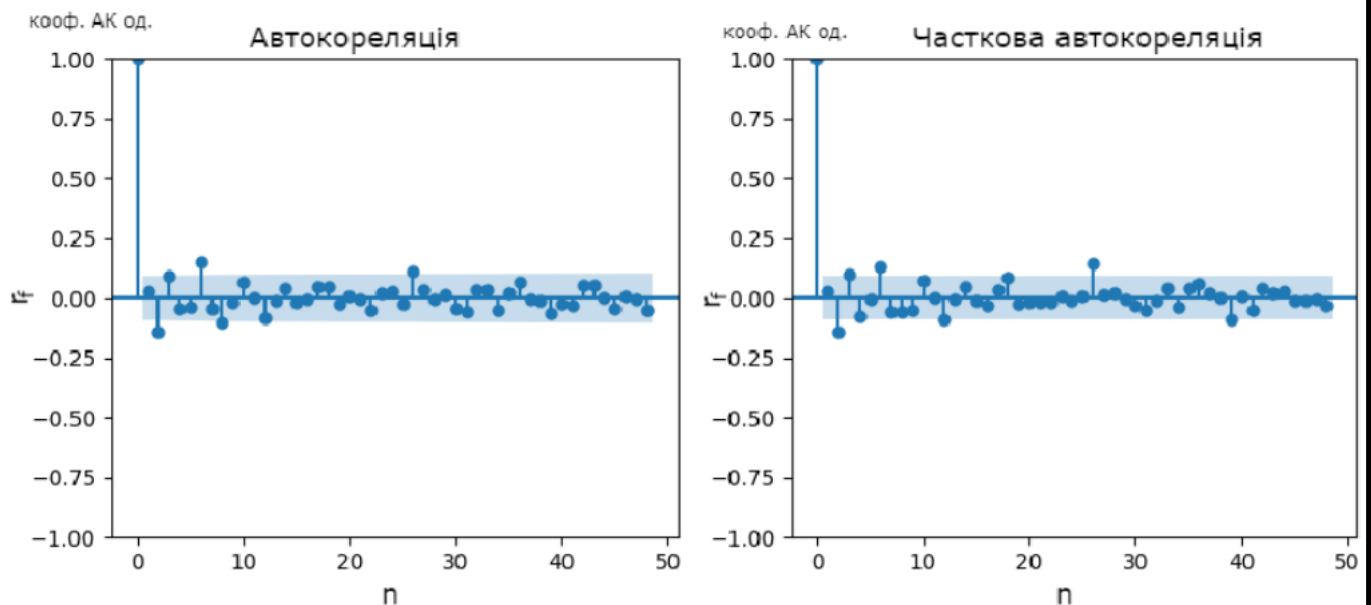


Рисунок 3.3 – Графіки автокореляції диференційованих та логарифмованих даних

Як можна побачити, на рисунку 3.3 диференційованих та логарифмованих даних не можна помітити сильної кореляції.

Іншим способом визначити сезонність є метод `seasonal_decompose`, наданий бібліотекою `statsmodels`. Подивившись на частину графіку сезонного елемента (рисунок 3.4) можна побачити цикл кожні 7 значень, тобто у обраних даних цикл проходить кожен тиждень.

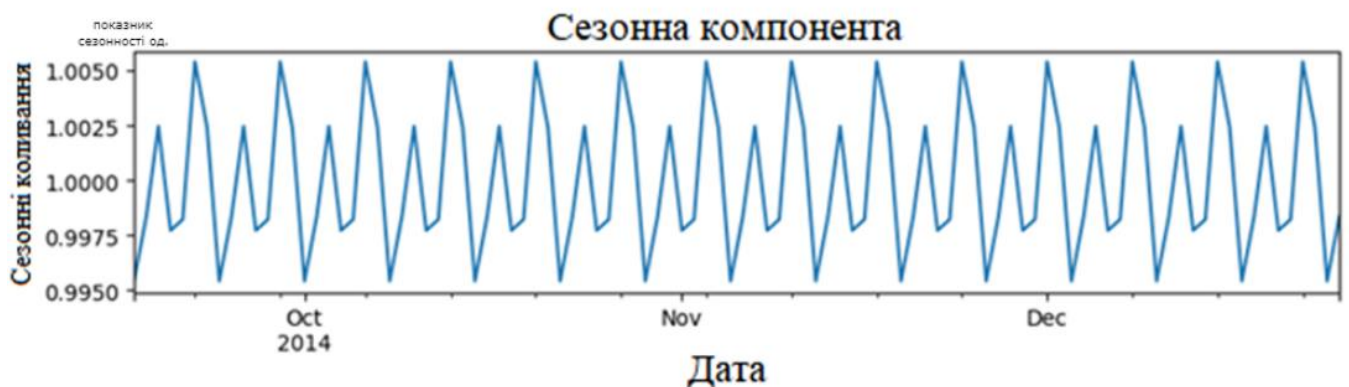


Рисунок 3.4 – Результат сезонної декомпозиції

Це значення параметру сезонності і буде використане для тестування.

Результати тестування можна побачити на рисунку 3.5 та таблиці 3.1.

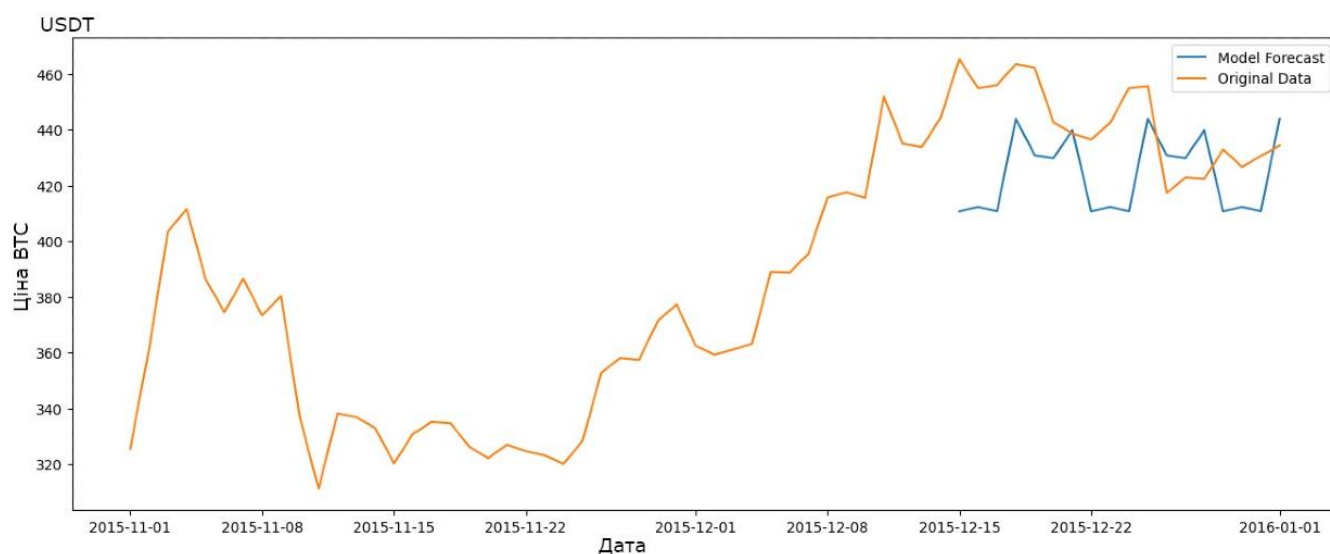


Рисунок 3.5 – Результати прогнозу SARIMA

Таблиця 3.1 – Оцінка результатів SARIMA

Назва метрики	MSE	RMSE	MAPE	R2
Значення	767.2	27.7	5.6%	-3.14

3.2.2 Оцінка моделі RNN

Не зважаючи на очікувану перевагу RNN над моделлю SARIMA, вона показала гірші результати, що може бути наслідком багатьох факторів, найбільш вірогідний з котрих – невдало підібрані гіперпараметри.

На рисунку 3.6 наведені результати роботи RNN, і їх оцінка в таблиці 3.2.

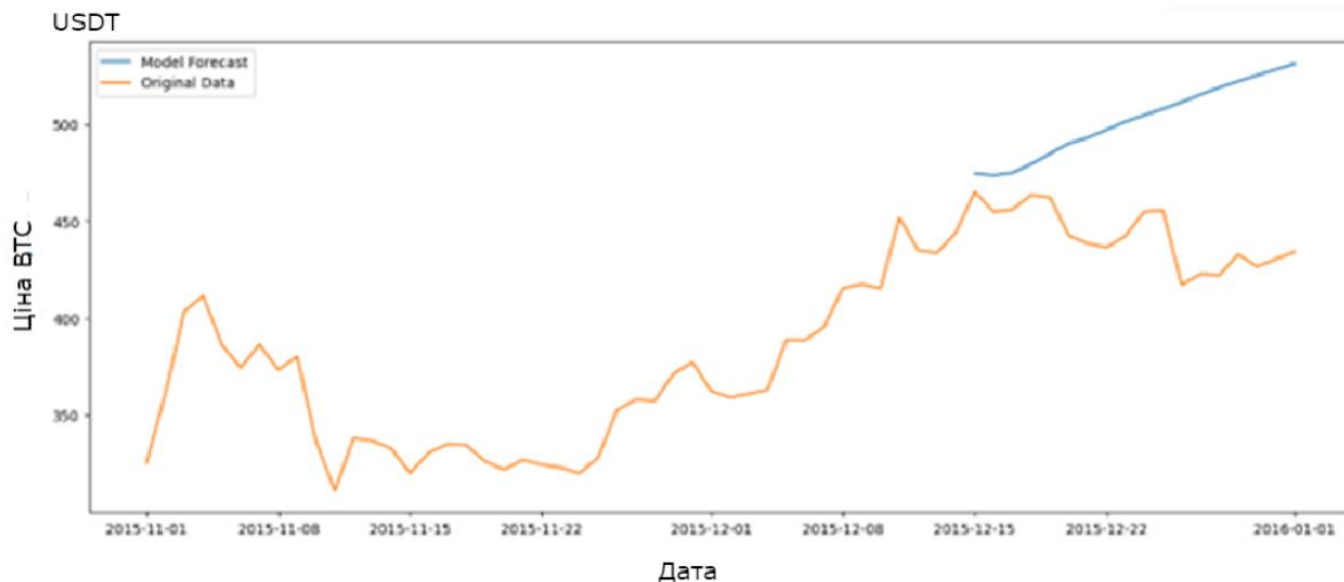


Рисунок 3.6 – Результати прогнозу RNN

Таблиця 3.2 – Оцінка результатів RNN

Назва метрики	MSE	RMSE	MAPE	R2
Значення	4599.9	467.8	11.6%	-11.8

3.2.3 Оцінка моделі LSTM

Серед обраних моделей, LSTM показала найкращий результат, тож для розробки додатку буде використано саме її.

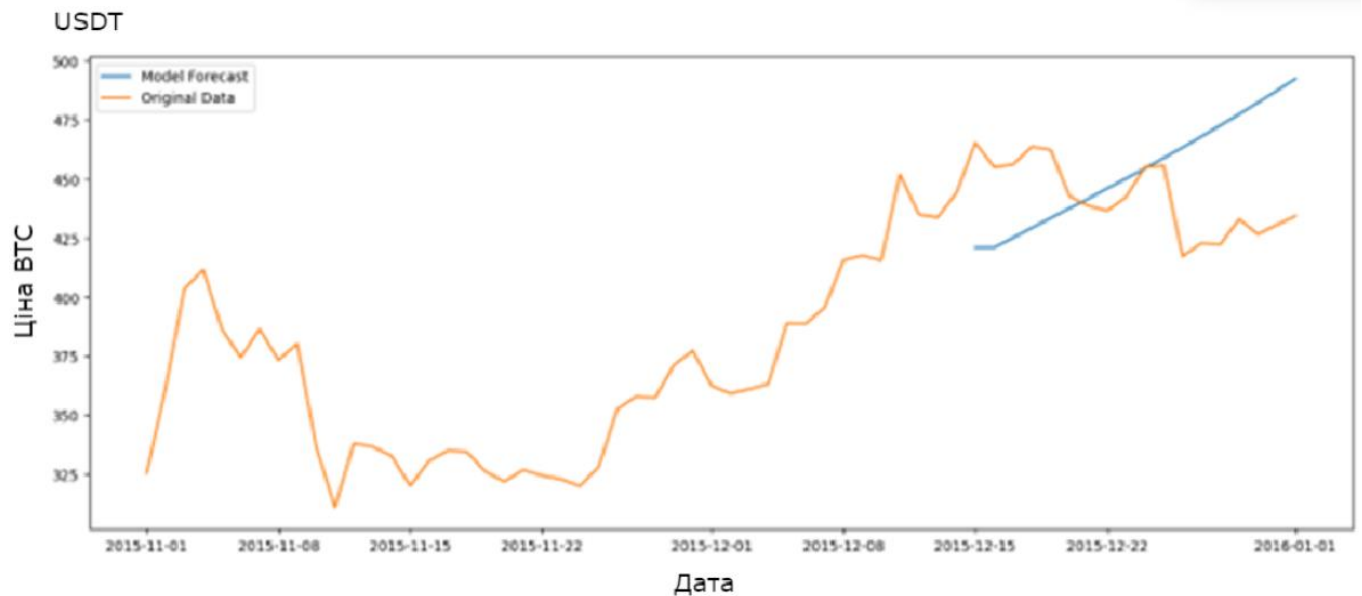


Рисунок 3.7 – Результати прогнозу LSTM

Таблиця 3.3 – Оцінка результатів LSTM

Назва метрики	MSE	RMSE	MAPE	R2
Значення	636.7	25.2	3.0%	-0.6

3.3 Опис обраних методів вирішення задачі

У цьому підрозділі буде описано методи оптимізації системи та покращення прогнозів системи, застосовані у даній роботі, та їх вплив на результати роботи моделі.

3.3.1 Підготовка вхідних даних

У розглянутому вище методі SARIMA доволі часто використовується такий метод стабілізації ряду як диференціація даних, оскільки він у певній мірі усуває тренд та сезонність ряду, а у деяких випадках допомагає виявити до цього непомітні патерни ряду, що значно підвищує точність прогнозів SARIMA. Прогнози нейронних мереж LSTM не настільки залежать від тренду та сезонності ряду, але повне чи часткове його усунення має покращити результат. Окрім того,

ще одним методом покращення результатів є нормалізація даних, яку можна з цим методом поєднати. До робочих даних буде застосований метод `extract_data`, який перетворить кожен елемент датасету на коефіцієнт, на який треба домножити на попередній елемент у тій самій колонці, таким чином досягаючи одразу двох цілей – нормалізація даних та посилення кореляції між поточним та попереднім прогнозом, що також згладить часовий ряд.

Нижче на рисунках 3.9–3.10 та в таблиці 3.4 зображені результати прогнозів моделі до і після нормалізації вхідних даних методом `extract_data`, та різниця у точності. Тести проведені на іншому наборі даних від розділу 3.2, тож показники будуть відрізнятися.

```
print(extracted_data[0])
```

```
[[1.          1.          1.          1.          1.          1.          ]
 [0.98067241 0.97583372 0.91309443 0.92807442 0.92807442 1.63762775]
 [0.91035792 0.91383755 0.84994102 0.86325525 0.86325525 1.80082919]
 [0.847185    0.90414243 0.86176843 0.89410361 0.89410361 1.75067437]
 [0.87597449 0.88092458 0.8690581  0.87205631 0.87205631 1.26230481]]
```

Рисунок 3.8 – Приклад формату даних після обробки

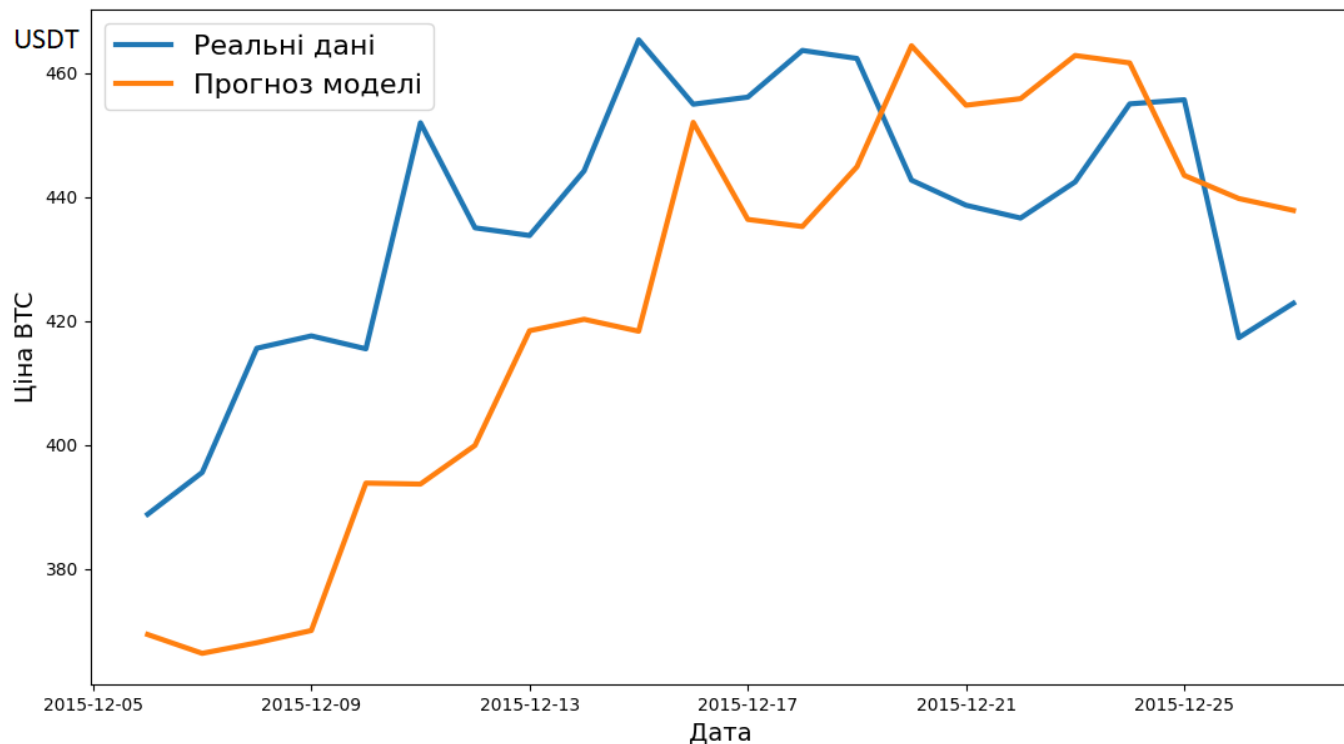


Рисунок 3.9 – Результати прогнозу до обробки

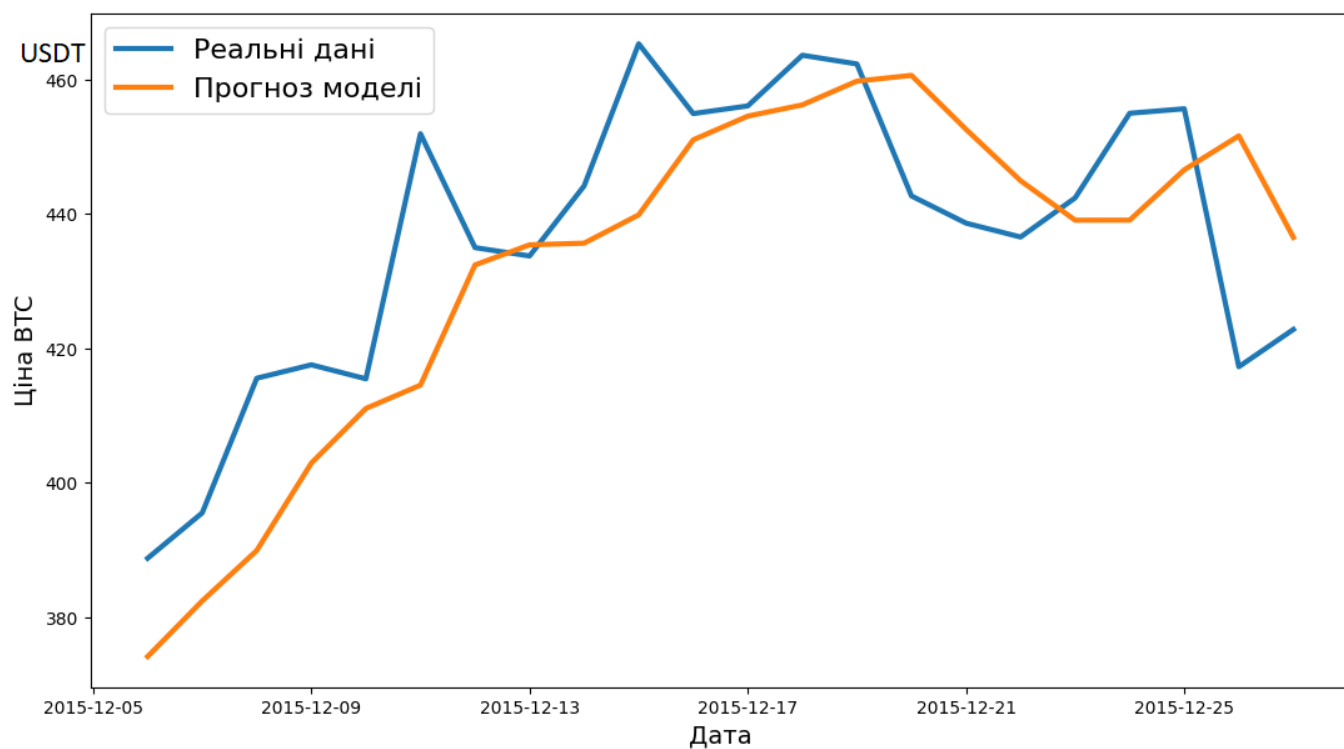


Рисунок 3.10 – Результати прогнозу після обробки

Таблиця 3.4 – Різниця між точністю прогнозів

Назва метрики	MSE	RMSE	MAPE	R2
Результати до обробки	809.5	28.4	6.0%	-0.8
Результати після обробки	262.8	16.2	2.9%	0.4

Як видно з результатів, така обробка даних значно покращує результати прогнозів.

3.3.2 Визначення сигналів

Щоб застосувати отримані з моделі прогнози цін, буде доречно використати їх у поєднанні з уже існуючою трейдинговою стратегією, у цій роботі – розглянуті лінії Боллінжера. Лінії Боллінжера – це популярний і простий у принципі роботи технічний індикатор, який використовується для розпізнавання сигналів покупки або продажу. Індикатор складається з трьох ліній:

- Середня лінія. Це просте рухоме середнє (Simple moving average) за певний період. У цій роботі використане двадцятиденне вікно для обчислення рухомого середнього.

- Верхня лінія. Це лінія яка вище за середню на n (у цій роботі $n=3$) стандартних відхилень за той же період часу, за який рахувалась середня лінія.

- Нижня лінія. Аналогічно, середня лінія – n * стандартне відхилення за 20 днів.

Коли реальна ціна знаходиться між верхньою та нижньою – відправляється сигнал «утриматися», коли нижче нижньої – сигнал «покупати», і коли вище верхньої – «продавати» [16].

Використання Ліній Боллінжера без додаткових інструментів має вагомий недолік – вони покладаються на принцип простого рухомого середнього, який хоч і доволі точно описує тренд часового ряду, але робить це із запізненням у половину вікна, у якому середнє було обраховане. Тому якщо під час обрахунку рухомого середнього використовувати дані «із майбутнього», то цей метод позбудеться

вищезгаданого «запізнення». Важливо помітити, що очевидно зі стовідсотковою точністю дані із наступних кроків отримати неможливо, тож поєднання цих двох методів – це спроба компенсувати похибку одного методу іншим методом із похибкою, що може привести до покращення результатів поки що лише у теорії.

Щоб перевірити цю теорію і оцінити доречність використання цього методу, у цій роботі буде проведений бектестинг на історичних даних цін біткоїну за останні 200 днів. Бектестинг буде проведений таким чином:

- Початковий капітал: 100 USD.
- Якщо капітал на даний момент у USD і поточна ціна BTC нижче за нижню лінію – купувати BTC на усі кошти.
- Якщо капітал на даний момент у BTC і поточна його ціна вище за верхню лінію – продавати увесь біткоїн.
- За припущення що обмін буде відбуватися на біржі binance із використанням BNB, комісія становитиме 0.075% за кожну транзакцію.
- У кінці тесту усі кошти переведуться у USD для оцінки результатів.

Ілюстрації результатів бектестингу (рисунки 3.11 та 3.12) виконані за допомогою бібліотеки plotly. Виконані із допомогою графіків японських свічок, оскільки вони несуть у собі максимум інформації, про рух цін за обраний період. Синіми та жовтими позначками позначені дні, під час яких система надавала сигнал «купувати» та «продавати» відповідно, і згідно з умовами бектестингу переводила усі кошти у відповідну валюту.

					ІС92.010БАК.004 ПЗ	Арк.
						46
Зм.	Лист	№ докум.	Підпис	Дата		

num buys: 7
num sells: 7
ending balance: 100.42817673941761 USD

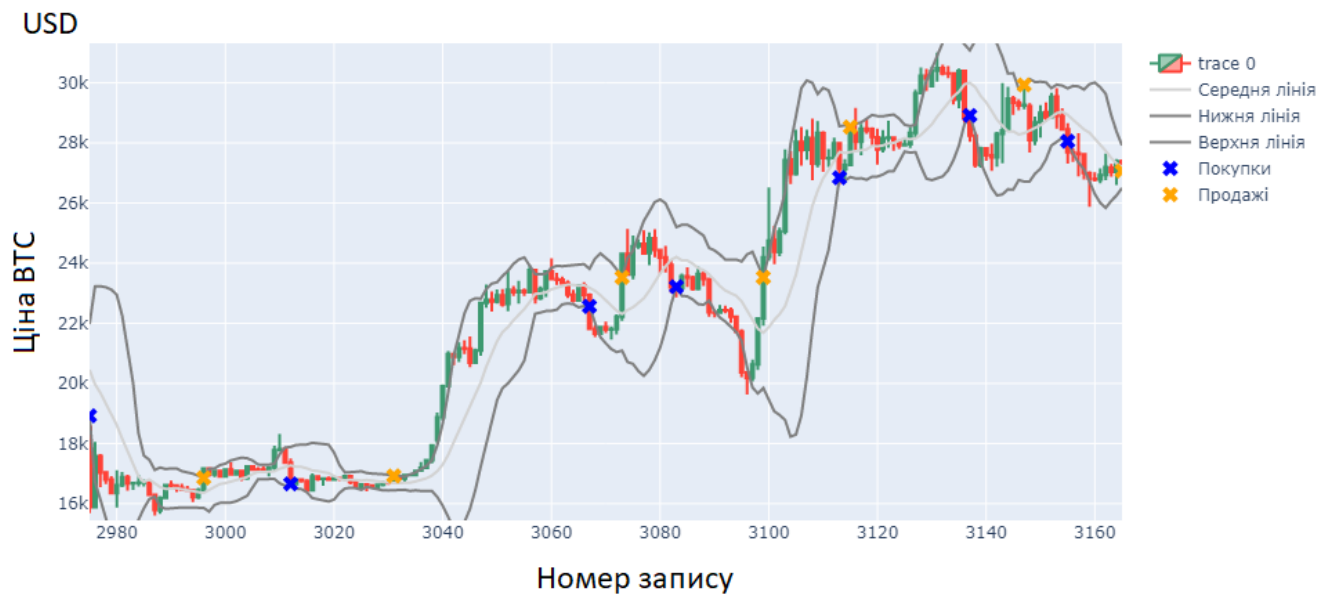


Рисунок 3.11 – Результати бектестінгу ліній Боллінжера

num buys: 16
num sells: 16
ending balance: 166.34632554896527 USD



Рисунок 3.12 – Результати бектестінгу комбінованого методу

Як можна бачити, комбінація методів дійсно показала кращий фінальний показник балансу ніж звичайні лінії Боллінжера. Треба зазначити, що це не є

абсолютною мірою ефективності даного методу. Однак із головною задачею яка була поставлена – а саме компенсація «запізнень» методу простого рухомого середнього ця комбінація впоралась.

На рисунках 3.12 та 3.13, які ілюструють роботу методів у фартмент тестових даних із різким рухом цін, можна побачити, що показник середнього рухомого відстає від реальних даних на декілька днів, у той час як комбінований метод цієї проблеми не має.

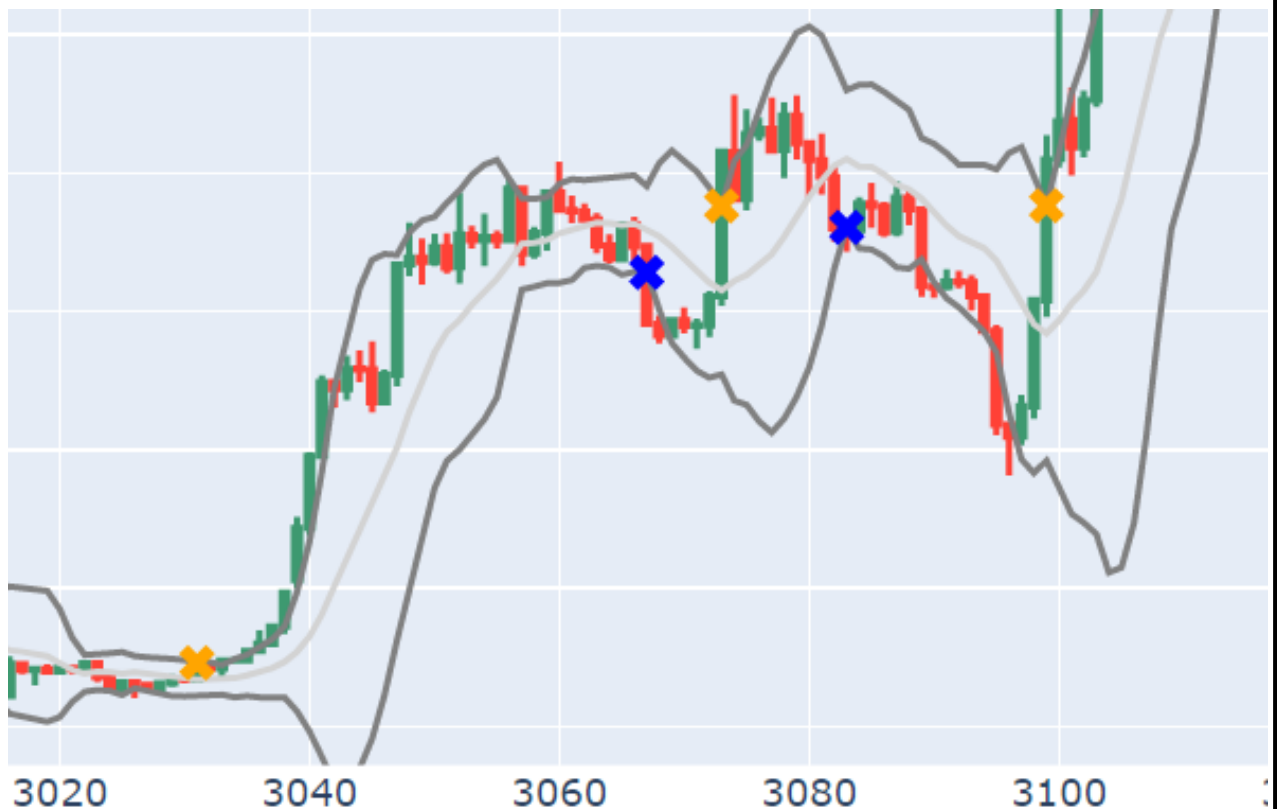


Рисунок 3.13 – Запізнення рухомого середнього відносно реальних даних

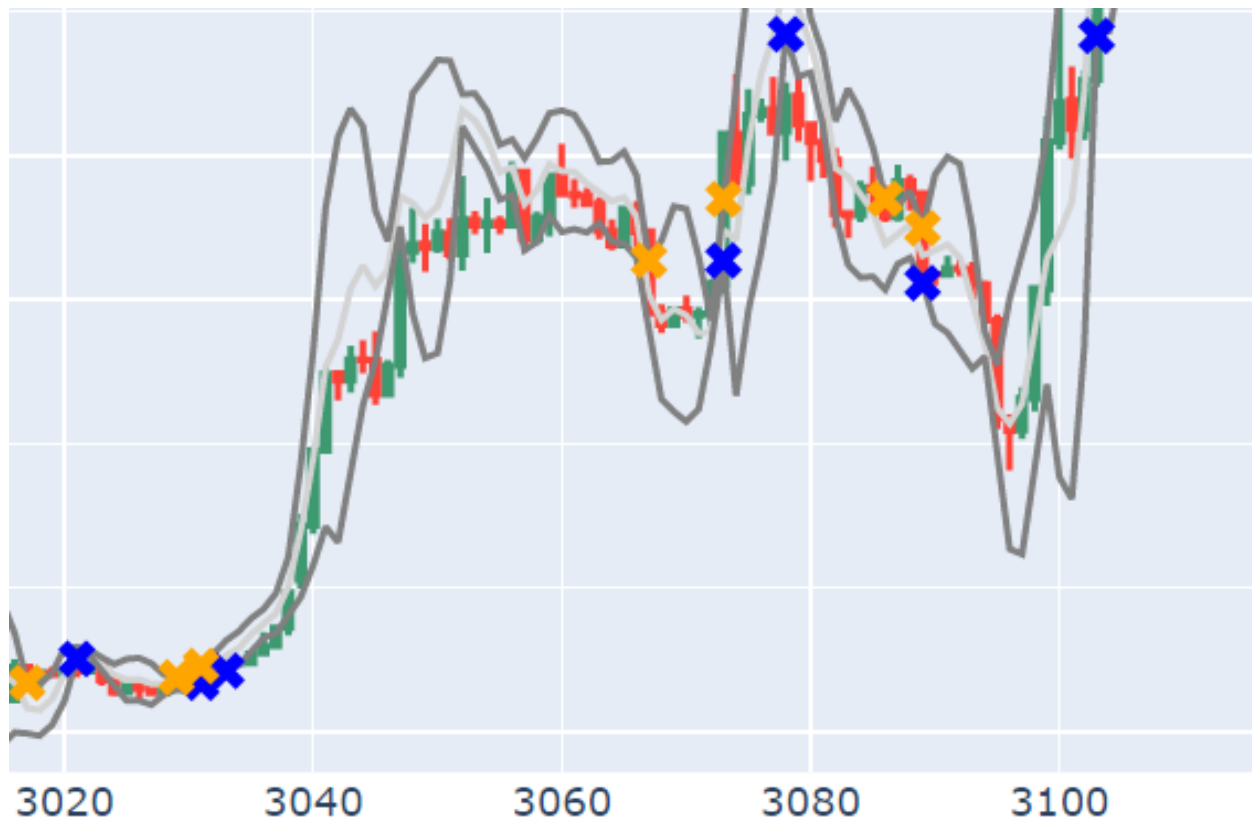


Рисунок 3.14 – Аналогічний фрагмент графіку комбінованого методу

3.3.3 Сгладження прогнозів

Дивлячись на графік бектестингу, можна помітити ділянки, де модель передбачає більш різкий зріст ніж можливо у реалістичних умовах. Впоратись із цією проблемою може метод, схожий на «відсікання градієнтів», який використовувався під час тестування рекурентних нейронних мереж. У даній роботі використано значення у 30% за день у обидві сторони як абсолютний максимум можливої зміни ціни, і будь яке значення що виходить за ці рамки буде вважатися аномалією і «відсічено», тобто обмежено тридцятьма відсотками.

Висновки до розділу

Порівнюючи моделі ефективність і точність моделей прогнозування SARIMA, RNN та LSTM на історичних даних із застосуванням метрик MSE, RMSE, MAPE та R2 score, було виявлено, що LSTM показує найточніші прогнози. За допомогою LSTM вдалося знизити помилки прогнозування і досягнути високої точності. MSE, RMSE, MAPE та R2 score, використані як метрики оцінки якості прогнозу, підтверджують переваги LSTM моделі в порівнянні з SARIMA та RNN. Таким чином, результати аналізу свідчать про те, що LSTM є найбільш підходящою моделлю для прогнозування часових рядів у порівнянні з іншими розглянутими моделями. Далі у цьому розділі було розглянуто методи оптимізації моделі і оцінка їх впливу на результат. Система показала вражаючий результат у +66% за 200 днів, але треба пам'ятати, що з великою вірогідністю такий приріст є аномалією, і що на даний момент жодна модель не може передбачити майбутнє, і отримані результати бажано використовувати як вказівки трейдерам та інвесторам, а не сліпо їм слідувати.

4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

4.1 Засоби розробки

У даній роботі буде використаний наступний стек технологій:

- Мова програмування Python;
- Інтегроване середовище розробки jupyter notebook;
- Бібліотека pandas;
- Бібліотека numpy;
- Бібліотека tensorflow;
- Бібліотека keras;
- Бібліотека sklearn;
- Бібліотека statmodels;
- Бібліотека plotly.

– Мова програмування Python. Python є високорівневою мовою програмування, яка має простий і зрозумілий синтаксис. Вона популярна серед дослідників і аналітиків даних, оскільки має багатий екосистему бібліотек для обробки та аналізу даних.

– Інтегроване середовище розробки Jupyter Notebook. Jupyter Notebook надає можливість запускати код не цілком, фрагментами, зберігаючи весь контекст, що допомагає з візуалізацією даних. Він став популярним серед аналітиків даних, оскільки дозволяє створювати документи, які об'єднують код, результати виконання і пояснення.

– Бібліотека pandas. Pandas – це потужна бібліотека для аналізу та маніпулювання даними. Надає потужні інструменти для роботи з даними, такими як структури даних (DataFrame), операції з фільтрації, сортування, об'єднання, агрегації та інше.

– Бібліотека numpy. NumPy – це основна бібліотека для наукових обчислень в Python. Вона надає нову структуру даних для масивів чисел будь-якої розмірності та інструменти для роботи з ними.

					ІС92.010БАК.004 ПЗ	Арк.
						51
Зм.	Лист	№ докум.	Підпис	Дата		

– Бібліотека tensorflow. TensorFlow – це відкрите програмне забезпечення для побудови та навчання глибоких нейронних мереж. На даний момент є одним з найпопулярніших фреймворків для аналізу даних та прогнозування на основі машинного навчання.

– Бібліотека keras. Keras – це надбудовка над бібліотекою TensorFlow, що надає високорівневий інтерфейс для роботи з нею.

– Бібліотека sklearn. Scikit-learn – це бібліотека машинного навчання для Python, яка у цій роботі виористовувалась здебільшого для оцінки та тестування моделей.

– Бібліотека statmodels. Statsmodels надає інструменти для статистичного моделювання та економетрики. Вона містить реалізації різних моделей, таких як ARIMA, SARIMA, SARIMAX, і дозволяє проводити статистичний аналіз та діагностику моделей часових рядів.

4.2 Вимоги до технічного забезпечення

Рекомендується використовувати потужний процесор з високою частотою такту, такий як Intel Core i7 або вище, для швидкого тренування моделі. Використання графічного процесора з підтримкою CUDA, такого як NVIDIA GeForce або Tesla з багатьма CUDA-ядрами, може прискорити обчислення, зокрема для моделей, які використовують LSTM. Для обробки великого обсягу даних, рекомендується мати щонайменше 16 ГБ оперативної пам'яті, але більше пам'яті поліпшить продуктивність. Бажано використання SSD диска, який забезпечує швидкий доступ до даних, особливо для завантаження і збереження великого обсягу тренувальних даних.

4.3 Структура додатку

Додаток складається з чотирьох файлів.

					IC92.010БАК.004 ПЗ	Арк.
						52
Зм.	Лист	№ докум.	Підпис	Дата		

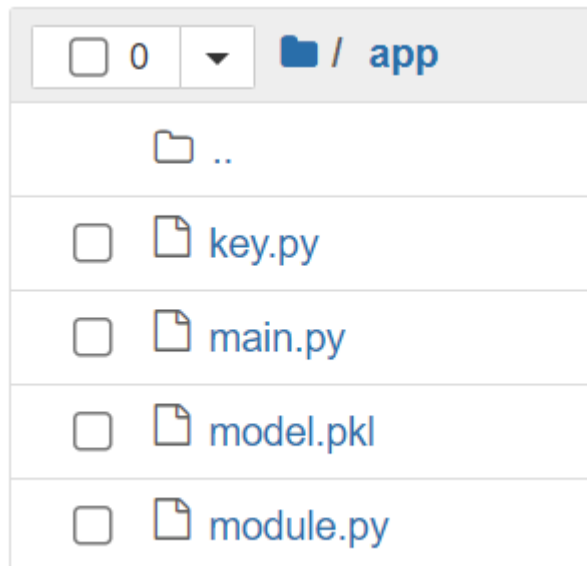


Рисунок 4.1 – Структура додатку

1) `key.py` – це файл python-формату, у якому зберігається API-ключ користувача з сайту <https://rapidapi.com/apidojo/api/yahoo-finance1>, з якого і будуть зібрані історичні дані про обрану криптовалюту.

2) `model.pkl` – це файл pickle-формату, у якому зберігається модель LSTM, натренована для прогнозування обраної криптовалюти. У даному випадку – модель налаштована для передбачення цін USD–BTC, і натренована на даних за останні десять років (мінус останні 200 днів). На рисунку 4.2 зображена схема мережі, на основі якої тренувалась `model.pkl`.

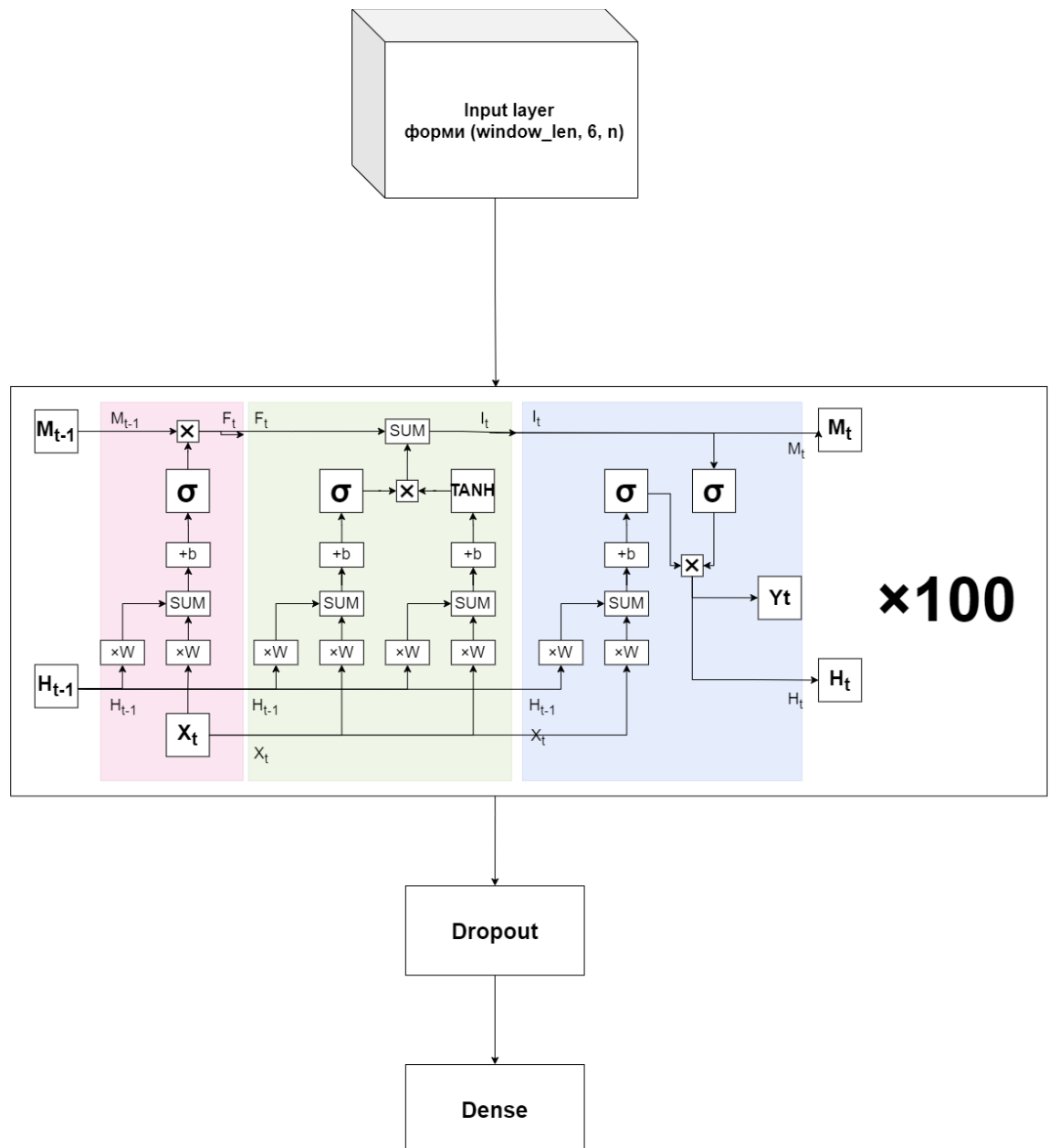


Рисунок 4.2 – Схема нейронної мережі

3) module.py – це файл python-формату, у якому описані функції, необхідні для роботи додатку, такі як:

3.1) normalize_zero_base – функція яка нормалізує масив чисел, представляючи кожен його елемент як коефіцієнт, на який треба домножити перший елемент масиву і додати першого елементу масиву.

3.2) extract_data – функція частково описана у третьому розділі, яка підготовлює данні, представляючи їх як масив коефіцієнтів, на які треба домножити попередні елементи, і яка використовує функцію normalize_zero_base.

Метод застосовувався також і під час тренування моделі.

3.3) `middle_band` – головна функція додатку, вона отримує на вхід данні за останні декілька днів, підготовлює їх функцією `extract_data`, робить прогноз відносно наступних елементів, і з певними вагами використовує значення останнього елементу вікна, рухомого середнього у цьому вікні, та власне прогнозованих значень наступних елементів обраховує значення, яке буде взяте як «середню лінію» для модифікованих ліній Боллінжера.

3.4) `bollinger_band` – функція яка отримує на вхід данні за останні декілька днів, та значення «середньої лінії», обраховує стандартне відхилення у цьому вікні, і обраховує значення нижньої та верхньої ліній, віднімаючи та додаючи стандартне відхилення домножене на коефіцієнт відповідно.

4) `main.py` – це файл `python`-формату, який використовує усі інші у додатку. При запуску йому потрібно передати символ, відповідний до бажаного прогнозу (у даному прикладі – лише `USD-BTC`). Використовуючи ключ з файлу `key.py`, `main` робить запит на отримання історичних даних, перетворює її на датафрейм, підключає модель з `model.pkl`, та викликає функції `middle_band` та `bollinger_band`. На основі отриманих ліній та значення останнього елементу датасету посилає один із трьох сигналів: «Продавати», «Утримуватись», «Покупати».

```
C:\Users\Aleksey\          \диплом\app>python main.py BTC-USD
1/1 [=====] - 0s 119ms/step
Утримуватись
```

Рисунок 4.3 – Приклад результату роботи програми

4.4 Тестування додатку

Проведено бектестинг моделі за останні 200 днів, тобто за період осінь 2022

					ІС92.010БАК.004 ПЗ	Арк.
						55
Зм.	Лист	№ докум.	Підпис	Дата		

— весна 2023, щоб оцінити роботу моделі на найбільш актуальних доступних даних.

num buys: 18
num sells: 18
ending balance: 207.8353532545041 USD

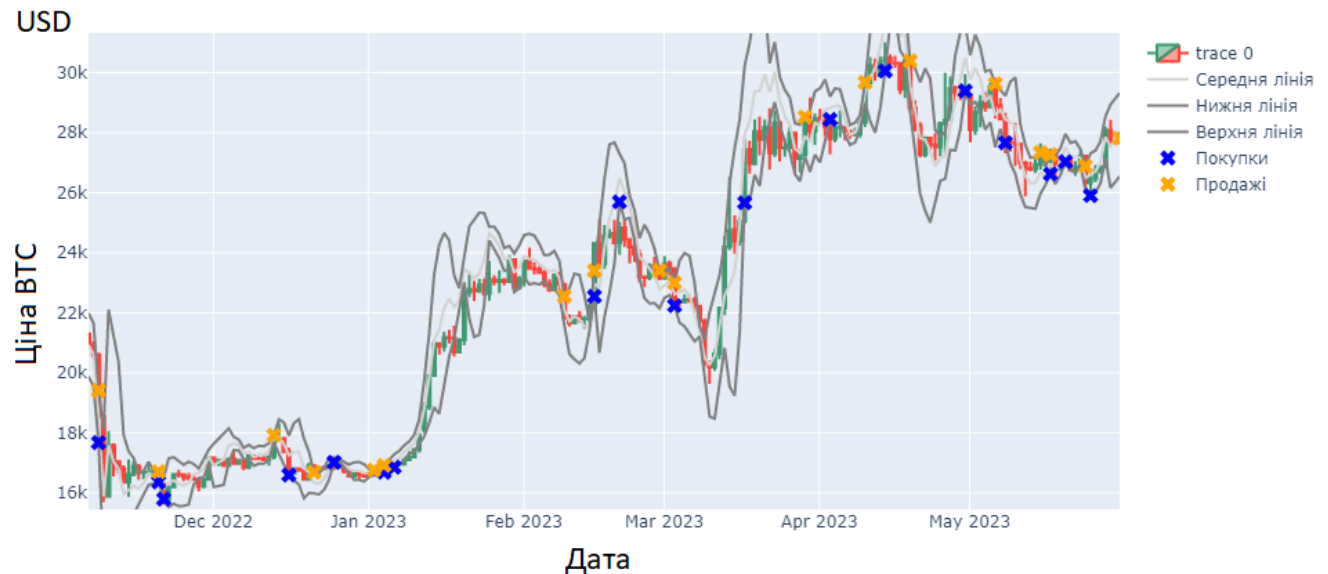


Рисунок 4.4 – Результати бектестінгу за останні сім місяців

Як можна побачити на рисунку 4.4, модель все ще показує задовільні результати, особливо порівнюючи їх з результатами бектестінгу стратегії зі звичайними лініями Боллінжера, наведеними на рисунку 4.5.

					ІС92.010БАК.004 ПЗ	Арк.
						56
Зм.	Лист	№ докум.	Підпис	Дата		

num buys: 8
num sells: 8
ending balance: 121.53750629424846 USD

USD



Рисунок 4.5 – Результати бектестінгу ліній Боллінжера за останні сім місяців

Висновки до розділу

У цьому розділі було наведено технічне та програмне забезпечення, необхідне для коректної роботи програмного застосунку, описана структура фінального виду програми, її принцип роботи, функції і потоки даних, та перевірена робота програми на актуальних даних.

					ІС92.010БАК.004 ПЗ	Арк.
						57
Зм.	Лист	№ докум.	Підпис	Дата		

ВИСНОВКИ

Цей дипломний проект присвячений розробці системи прогнозування ризиків обміну криптовалют, метою створення цієї системи було покращення результатів прогнозу і надання важливих вказівок трейдерам та інвесторам на ринку криптовалют. Розроблена система показала дуже позитивні результати під час тестування, що підтверджує її валідність для прогнозування ризиків обміну на ринку криптовалют, але треба у котре зазначити, що жодна система прогнозування не може мати точність у сто відсотків, тож отримані від неї результати потрібно сприймати в першу чергу як рекомендації для трейдерів–початківців.

У першому розділі описано основні положення, які необхідні для розуміння ситуації на ринку, і побудову якісних прогнозів, також були розглянуті існуючі методи прогнозування, і пояснений принцип їх роботи.

Другий розділ був присвячений встановленню вхідних і вихідних даних інформаційної системи.

У третьому розділі було визначено метрики оцінки точності прогнозів моделей, які будуть використовуватися у роботі, було проведено порівняльний аналіз між описаними в першому розділі методами передбачення з метою вибору найкращого, та були описані та обґрунтовані усі методи оптимізації результатів інформаційної системи.

Четвертий розділ перераховує необхідне для роботи фінальної програми технічне та програмне забезпечення, описує структуру програми та алгоритм її роботи, та проводить тестування результатів на актуальних даних.

Додатки містять графічні матеріали, такі як діаграму станів та схему нейронів, використаних у роботі.

					ІС92.010БАК.004 ПЗ	Арк.
						58
Зм.	Лист	№ докум.	Підпис	Дата		

СПИСОК ЛІТЕРАТУРИ

- 1) Overall cryptocurrency market capitalization per week from July 2010 to April 2023. Statista. – URL: <https://www.statista.com/statistics/730876/cryptocurrency-maket-value/> (дата звернення 01.05.2023)
- 2) Antonopoulos, M. A. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Видавництво O'Reilly Media, 2014
- 3) Drescher, D. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Видавництво Apress, 2017
- 4) Current Cryptocurrency Prices by Market Cap – URL: <https://coinmarketcap.com/> (дата звернення 01.05.2023)
- 5) Schannep, J. Dow Theory for the 21st Century: Technical Indicators for Improving Your Investment Results, Видавництво Wiley, 2008
- 6) Minh-Hao, V. Predicting Cryptocurrency Price Movements Based on Social Media, Конференція 2019 International Conference on Advanced Computing and Applications (ACOMP), 2019
- 7) Albi, N. The doge worth 88 billion dollars: A case study of Dogecoin, Convergence: The International Journal of Research into New Media Technologies, 2022
- 8) Murphy, J. J. Technical Analysis of the Financial Markets, видавництво New York Institute of Finance, 1999
- 9) Shumway, R. H. Time Series Analysis and Its Applications: With R Examples / D. S. Stoffer, Видавництво Springer, 2017
- 10) Brockwell, P.J. Introduction to Time Series and Forecasting / R. A. Davis, Видавництво Springer, 2002
- 11) Goodfellow, I. J. Deep Learning (Adaptive Computation and Machine Learning series) / Y. Bengio, A. Courville, Видавництво The MIT Press, 2016
- 12) Géron, A. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, Видавництво O'Reilly Media, 2019
- 13) IntelliJ IDEA documentation – URL: <https://www.jetbrains.com/help/idea/> (дата звернення 01.05.2023)
- 14) PyCharm IDE documentation – URL:

					IC92.010БАК.004 ПЗ	Арк.
						59
Зм.	Лист	№ докум.	Підпис	Дата		

<https://www.jetbrains.com/help/pycharm/> (дата звернення 01.05.2023)

15) The Jupyter Notebook documentation – URL: <https://jupyter-notebook.readthedocs.io/en/stable/> (дата звернення 01.05.2023)

16) Bollinger, A. J. Bollinger on Bollinger bands , Видавництво McGraw Hill, 2001

17) Hyndman, R. J. Forecasting: Principles and Practice / G. Athanasopoulos, Видавництво OTexts, 2013

					ІС92.010БАК.004 ПЗ	Арк.
						60
Зм.	Лист	№ докум.	Підпис	Дата		