

ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ  
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**КВАЛІФІКАЦІЙНА РОБОТА**

на тему: «Розробка методу автоматизованого аналізу  
динаміки руху цін на біржах віртуальних активів»

на здобуття освітнього ступеня магістра  
зі спеціальності 121 Інженерія програмного забезпечення  
(код, найменування спеціальності)  
освітньо-професійної програми Інженерія програмного забезпечення  
(назва)

*Кваліфікаційна робота містить результати власних досліджень.  
Використання ідей, результатів і текстів інших авторів мають посилання  
на відповідне джерело*

\_\_\_\_\_  
(підпис)                      Леонід ЧЕРНОВ

Виконав: здобувач вищої освіти групи ПДМ-62  
Леонід ЧЕРНОВ

Керівник: Вікторія КОРЕЦЬКА  
к.пед.н., доцент

Рецензент: Юрій ПЕПА  
к.т.н., доцент

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**  
**Навчально-науковий інститут інформаційних технологій**

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти Магістр

Спеціальність 121 Інженерія програмного забезпечення

Освітньо-професійна програма «Інженерія програмного забезпечення»

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

Інженерії програмного забезпечення

\_\_\_\_\_ Ірина ЗАМРІЙ

« \_\_\_\_\_ » \_\_\_\_\_ 2023 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

\_\_\_\_\_ Чернову Леоніду Олександровичу \_\_\_\_\_

1. Тема кваліфікаційної роботи: «Розробка методу автоматизованого аналізу динаміки руху цін на біржах віртуальних активів»

керівник кваліфікаційної роботи Вікторія КОРЕЦЬКА к.пед.н., доцент,

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «19» жовтня 2023р. №145.

2. Строк подання кваліфікаційної роботи «29» грудня 2023р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, нейронна мережа на основі довготривалої короткочасної пам'яті (LSTM), алгоритм для прогнозування цінового тренду на ринку віртуальних активів, модель прогнозування цінового тренду з використанням LSTM, набір даних з біржі віртуальних активів для тренування та тестування роботи мережі.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Аналіз предметної галузі та існуючих методів та підходів.

2. Постановка задачі, підготовка набору даних.

3. Розробка алгоритму автоматизованого аналізу цін на біржах.
4. Розробка нейронної мережі на основі довготривалої короткочасної пам'яті та її навчання.
5. Експериментальна оцінка ефективності методів.

5. Перелік графічного матеріалу: *презентація*

1. Порівняння методів аналізу руху цін на біржах віртуальних активів.
2. Діаграма послідовностей обробки кількісних даних.
3. Діаграма послідовностей обробки повідомлень та новин.
4. Діаграма послідовностей процесу прогнозування.
5. Алгоритм роботи системи.

6. Дата видачі завдання «19» жовтня 2023 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	19.10-05.11.23	
2	Огляд предметної галузі та аналіз існуючих методів	06.11-12.11.23	
3	Дослідження та порівняння основних концепцій, які пояснюють зміни цін на ринку віртуальних активів	13.11-19.11.23	
4	Вивчення підходів та моделей машинного навчання	20.11-26.11.23	
5	Дослідження технологій машинного навчання	27.11-03.12.23	
6	Розробка нового підходу до прогнозування на основі використання елементів штучного інтелекту	04.12-10.12.23	
7	Оформлення роботи: вступ, висновки, реферат	11.12-20.12.23	
8	Розробка демонстраційних матеріалів	21.12-29.12.23	

Здобувач вищої освіти

\_\_\_\_\_ (підпис)

Леонід ЧЕРНОВ

Керівник кваліфікаційної роботи

\_\_\_\_\_ (підпис)

Вікторія КОРЕЦЬКА





## РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 71 стор., 10 табл., 20 рис., 40 джерел.

*Мета роботи* – підвищення якості прогнозування, за допомогою автоматизованого аналізу руху цін на біржах віртуальних активів.

*Об'єкт дослідження* – автоматизований аналіз руху ціни на біржах віртуальних активів.

*Предмет дослідження* – методи аналізу ринкових даних та рекурентні нейронні мережі для передбачення часових рядів.

*Короткий зміст роботи:* У роботі проведено дослідження існуючих методів аналізу руху цін на біржах. Проаналізовано основні індикатори технічного аналізу, та роботу алгоритмів машинного навчання на даних технічних індикаторів.

У ході роботи було визначено, що індикатори технічного аналізу, довели свою неефективність для ринку віртуальних активів, на більшості часових проміжках. Варто зауважити, що проведення фінансового аналізу за допомогою технічних індикаторів як вхідних даних для навчання моделей машинного навчання є більш ефективним, ніж використання самих індикаторів.

В результаті роботи розроблено новий метод автоматизованого аналізу динаміки руху цін на ринку віртуальних активів, який може підвищити точність прогнозів та покращити результативність інвесторів. Варто зазначити, що такого результату вдалось досягти, лише після додавання показників тональності як вхідних даних для навчання LSTM моделі, разом із даними технічного індикатору.

**КЛЮЧОВІ СЛОВА:** ВІРТУАЛЬНІ АКТИВИ, ІНДИКАТОРИ ТЕХНІЧНОГО АНАЛІЗУ, РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ, МАШИННЕ НАВЧАННЯ.

## **ABSTRACT**

Text part of the master's qualification work: 71 pages, 10 tables, 20 pictures, 40 sources.

The purpose of the work is to improve the quality of forecasting, using automated analysis of price movements on virtual asset exchanges.

Object of research – is an automated analysis of price movements on virtual asset exchanges.

Subject of research – is methods of market data analysis and recurrent neural networks for predicting time series.

Summary of the work: The work examines the existing methods of analyzing price movements on stock exchanges. The main indicators of technical analysis and the work of machine learning algorithms on the data of technical indicators were analyzed.

In the course of the work, it was determined that technical analysis indicators proved ineffective for the virtual asset market on most time frames. It is worth noting that conducting financial analysis using technical indicators as input data for training machine learning models is more effective than using the indicators themselves.

As a result of the work, a new method of automated analysis of the dynamics of price movements on the virtual assets market was developed, which can increase the accuracy of forecasts and improve the performance of investors. It is worth noting that such a result was achieved only after adding tonality indicators as input data for LSTM model training, along with technical indicator data.

**KEYWORDS: VIRTUAL ASSETS, TECHNICAL ANALYSIS INDICATORS, RECURRENT NEURAL NETWORKS, MACHINE LEARNING.**

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ .....	9
ВСТУП.....	10
РОЗДІЛ 1 ОГЛЯД ІСНУЮЧИХ МЕТОДІВ АНАЛІЗУ ДИНАМІКИ РУХУ ЦІН НА БІРЖАХ.....	12
1.1 Становлення технології блокчейн та ринку віртуальних активів.....	12
1.2 Технічний аналіз та гіпотеза ефективного ринку.....	16
1.3 Аналіз останніх наукових досліджень.....	20
РОЗДІЛ 2 АНАЛІЗ МАТЕМАТИЧНИХ МОДЕЛЕЙ ТА ВИЗНАЧЕННЯ ОПТИМАЛЬНИХ ПАРАМЕТРІВ СИСТЕМИ ПРОГНОЗУВАННЯ .....	27
2.1 Аналіз нейромережових технологій .....	27
2.2 Класичні алгоритми машинного навчання .....	32
2.3 Огляд бібліотек та фреймворків використаних у роботі .....	37
2.4 Підготовка даних .....	38
2.5 Побудова та налаштування моделі .....	46
РОЗДІЛ 3 РЕЗУЛЬТАТИ РОБОТИ МЕТОДУ АВТОМАТИЗОВАНОГО АНАЛІЗУ ДИНАМІКИ РУХУ ЦІН НА БІРЖАХ ВІРТУАЛЬНИХ АКТИВІВ .....	50
3.1 Результати перевірки технічних індикаторів.....	50
3.2 Результати роботи алгоритмів машинного навчання.....	61
ВИСНОВКИ.....	70
ПЕРЕЛІК ПОСИЛАНЬ .....	72
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація).....	76



## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ANN – Artificial Neural Network – штучна нейронна мережа;

BB – Bollinger Bands – Смуги Боллінджера;

EMA – Exponential Moving Average – експоненційна ковзна середня;

EMH – Efficient Market Hypothesis – Гіпотеза ефективного ринку;

LSTM – Long Short-Term Memory – довга короткочасна пам'ять;

MACD – Moving Average Convergence/Divergence – розбіжність/збіжність ковзних середніх;

MLP – Many Layer Perceptron – багат шаровий перцептрон;

MSE – Mean squared error – середньоквадратична помилка.

PCA – Principal components analysis – аналіз головних компонент;

SMA – Simple Moving Average – проста ковзна середня;

RSI – Relative strength index – індекс відносної сили;

RNN – Recurrent neural network – рекурентні нейронні мережі.

## ВСТУП

Інтерес громадськості до віртуальних активів в останні роки зростає в геометричній прогресії. Багато інвесторів проводять фінансовий аналіз, щоб отримати інформацію про актив, оскільки ринок досить нестабільний. Інвестори використовували багато методів, щоб отримати більше знань про майбутні тенденції віртуальних активів. Одним із таких прийнятих способів є прогнозування цін за допомогою алгоритмів машинного навчання.

*Актуальність дослідження* зумовлена зростаючим інтересом до інвестування у ринок віртуальних активів. Прийняття швидких та точних рішень на ринку віртуальних активів, стає все важчими для інвесторів, у зв'язку зі зростанням обсягів даних, постійними змінами на ринку та конкуренцією. Розуміння цих змін через автоматизований аналіз дозволяє розробляти кращі фінансові стратегії, прогнозувати ризики та швидше реагувати на нові умови, що сприяє покращенню результативності інвестування. Варто зазначити, що відчутний вплив на ціни віртуальних активів здійснюють публікації відомих осіб у соціальних мережах, а також настрої користувачів, які мають особистий інтерес у цьому процесі.

Тому, розробка нового методу автоматизованого аналізу динаміки руху цін на ринку віртуальних активів, може підвищити точність прогнозів та покращити результативність інвесторів.

*Метою роботи* є підвищення якості прогнозування, за допомогою автоматизованого аналізу руху цін на біржах віртуальних активів.

Для досягнення мети були поставлені наступні завдання:

- Огляд предметної галузі та аналіз існуючих методів;
- Дослідження та порівняння основних концепцій, які пояснюють зміни цін на фінансових ринках, зокрема на ринку віртуальних активів;
- Вивчення підходів та моделей машинного навчання;
- Розробка нового підходу до прогнозування на основі використання елементів штучного інтелекту;

- Створення методу навчання рекурентних нейронних мереж, що дозволяють підвищити якість прогнозування;
- Дослідження ефективності отриманих даних на основі створення власних прогнозів.

*Об'єктом дослідження є автоматизований аналіз руху ціни на біржах віртуальних активів.*

*Предметом дослідження є методи аналізу ринкових даних та рекурентні нейронні мережі для передбачення часових рядів.*

*Методами дослідження є літературний огляд та аналіз наукових публікацій, методи аналізу динаміки руху цін на віртуальні активи та методи прогнозування, а також статистичний аналіз результатів тестування роботи алгоритму.*

*Структура роботи.* Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та демонстраційних матеріалів.

# 1 ОГЛЯД ІСНУЮЧИХ МЕТОДІВ АНАЛІЗУ ДИНАМІКИ РУХУ ЦІН НА БІРЖАХ

## 1.1 Становлення технології блокчейн та ринку віртуальних активів.

Концепція блокчейну була представлена в 1991 році Стюартом Хабером та Скоттом Сторнеттою. Дослідження представило метод безпечного запису часу та дати, коли цифровий документ був створений, змінений або доступ до нього. [1]

Особа або група на ім'я Сатоші Накамото використовувала ті самі принципи та випустила технічну документацію «Bitcoin: однорангова електронна грошова система» у 2008 році. Документ представив концепцію децентралізованої цифрової валюти, яка працюватиме без банків чи платіжних процесорів. Замість офіційного інституту, який контролює фінансові операції, розгалужена мережа забезпечує дійсність усіх транзакцій. Поки ніхто не володіє більш ніж 50% обчислювальної потужності мережі або в коді немає помилок, усі транзакції захищені. Кожна транзакція додається до пулу транзакцій. Мережа додає транзакції з пулу до блоків. Блок містить всю необхідну інформацію про нові транзакції та хеш попереднього блоку. Тепер, якщо хтось спробує змінити попередній блок, хеш блоку зміниться, і всі наступні блоки повинні бути обчислені знов. Якщо обчислювальна потужність становить менше 50% потужності мережі, то особа або група, яка змінила інформацію, ніколи не зробить необхідні обчислення вчасно. У якийсь момент мережа відмовиться від зміненої гілки блоків, оскільки в мережі є новіша гілка. [2]

Для базової версії блокчейну не потрібно багато елементів. Блокчейн має блоки, пов'язані між собою хронологічно. Кожен блок, крім першого, має посилання на попередній блок. Блоки містять один або кілька даних передачі всередині них. Ці дані мають мітку часу, інформацію про передачу та nonce. Nonce – це короткий текст, який додається до блоку, щоб отримати правильний хеш. Коли в блоці будуть всі необхідні дані, він буде хешований. Коли блок хешується в блокчейні типу підтвердження роботи, функція намагатиметься знайти хеш, у

якому, наприклад, перші шість чисел дорівнюють одиниці. Якщо на початку хешу немає потрібної кількості одиниць, тоді попереду буде змінено, і блок знову буде хешовано. Цей цикл триватиме, доки хеш не матиме належної кількості одиниць на початку. Знаходження номеру, який дасть хеш та який схвалить мережа, вимагає дуже багато ресурсів. Скільки перших чисел має бути одиницею, залежить від обчислювальної потужності мережі. Потрібні числа постійно змінюються, щоб обчислення були достатньо складними. [3]

Хешування гарантує, що ланцюжок блоків не можна змінити пізніше. Якщо хтось змінить щось із минулого блоку, то хеш цього блоку буде неправильним. Наступний блок містить інформацію про хеш попереднього блоку, якщо він зміниться, цей блок потрібно буде обчислити знову, це пройшло б весь список блоків. Це означає, що якщо хтось хоче змінити будь-яку інформацію, необхідно прорахувати кожен блок, який йде після зміненого блоку.

Якщо особа, яка намагається внести зміни, має менше ніж 50% обчислювальної потужності всієї мережі, вона не зможе знову обчислити блоки досить швидко. Система відправляє повідомлення мережі про те, що є гілка, вміст якої відрізняється від іншої фактичної гілки. Потім мережа перевіряє, яке відділення більш розширене. У той час обидві можуть бути однаковими, але якщо особа, яка внесла зміни, має меншу обчислювальну здатність, ніж усі інші, тоді з часом фактична гілка буде більшою, і врешті-решт вона буде довшою, ніж змінена гілка. У цей момент мережа більше не буде спільно використовувати змінену гілку, оскільки коротшу гілку буде знищено. Якщо видалена гілка мала транзакції, які не належать до розгалуженої гілки, ці транзакції повернуться до пулу транзакцій. [4]

Bitcoin є прикладом блокчейну, призначеного для захищених транзакцій, його блокчейн зберігає лише дані, необхідні для транзакцій, і не може зберігати нічого іншого. Ethereum був першим блокчейном, який представив децентралізовані програми. Блокчейн Ethereum можна використовувати для смарт-контрактів і збереження даних.

Наприклад, це дозволяє художникам зберігати свої роботи як цифрове мистецтво. Кожен може володіти копією цифрового мистецтва, але блокчейн

доводить, кому належить оригінальна копія мистецтва. Так само, як будь-хто може мати плакат відомої картини, але лише один може мати оригінальну картину.

Змінити дані, збережені в блокчейні, важко. Дані можна змінити, якщо є згода більшості валідаторів блокчейн-мережі. Отримати такий консенсус щодо неавторизованих змін практично неможливо, якщо блокчейн достатньо великий. Зміна минулих транзакцій вимагатиме більше обчислювальної потужності, ніж будь-хто може отримати. Все ж теоретично це можливо, але це буде важко, оскільки кожен учасник мережі має копію всього блокчейну, що ускладнює його зміну. Структура децентралізації робить блокчейн прозорим, тому будь-хто може відстежувати та перевіряти кожну транзакцію. Транзакції мають мітки часу, і вони пов'язані з попередніми транзакціями, що створює слід і зменшує можливості шахрайства, маніпуляцій або помилок у ланцюжку.

Варто зазначити, що блокчейн має деякі недоліки. Децентралізована та безпечна система потребує більше ресурсів, ніж централізована. Блокчейн-мережі споживають величезну кількість простору, пропускної здатності мережі та обчислювальної потужності.

Криптовалюта – це актив із обліковою книгою та криптографічними можливостями. Тому теоретично його можна використовувати як валюту. Люди можуть передавати власні криптовалюти іншим користувачам, але для оплати товарів широко використовується лише декілька з них. Навіть з Bitcoin компанії проводять більше випробувань щодо того, як будуть працювати платежі за допомогою його мережі, замість того, щоб використовувати його як фактичний варіант оплати. Схоже, що криптовалюти знайдуть застосування для певних видів транзакцій, таких як міжнародні грошові перекази та купівля ліків, але не для повсякденних транзакцій, принаймні протягом десятиліття чи двох. [5]

Криптовалюти відрізняються від традиційних активів тим, що вони не мають фізичної копії. Усі реальні валюти мають фізичну версію, хоча більшість емісії грошей є лише цифровими. У деяких випадках криптовалюти мають публічну книгу, і будь-хто може завантажити всі транзакції, щоб перевірити, чи вони дійсні.

Bitcoin вважається першою криптовалютою. Після нього з'явилося багато різних криптовалют, і постійно з'являється більше, на сьогодні їх кількість перевищує 20000. Більшість нових криптовалют намагаються робити деякі речі краще, а саме енергоефективність, швидкість транзакцій і ціна транзакцій.

Якщо визначення криптовалюти не є тривіальним, поміщення різних криптовалют у різні категорії ще складніше. Не існує конкретних категорій для криптовалют, але валюти можна впорядкувати в категорії за основними властивостями валюти.

Однією з можливостей для категорій є оплата, конфіденційність, децентралізовані фінанси, ігри та транскордонні платежі. Оскільки криптовалюти, в більшості, є копіями інших криптовалют, ці категорії не є досконалими, та більшість з них можуть належати до кількох категорій. [6]

На початку криптовалюти були винайдені для децентралізованих переказів без центрального органу, який би контролював вартість валюти. Тому не було потреби довіряти системі, як це відбувається з традиційною валютою. За допомогою традиційної валюти центральні банки можуть контролювати вартість валюти, друкуючи більше грошей або знищуючи гроші. За допомогою криптовалюти ніхто не може безпосередньо контролювати вартість. Інші переваги криптовалют – нижчі комісії за транзакції, більша доступність і підвищена безпека.

Недоліками криптовалют є висока волатильність, обмежене прийняття та шахрайство. Немає центрального органу, який контролює вартість криптовалют. Це робить ціну дуже мінливою. Вартість може зрости або впасти на значні суми за короткий час. Деякі люди інвестують у криптовалюти, і коли центральні органи влади приймають рішення, які впливають на вартість грошей, це вплине на вартість криптовалют через інвесторів. Інвесторам може знадобитися продати свої активи, і коли багато інвесторів продають свої активи, вартість активів впаде. Криптовалюти не є валютою, як стандартні гроші. З офіційною валютою люди знають, що їхні гроші приймаються в одній або кількох країнах. З криптовалютами люди повинні один за одним перевіряти кожен магазин, чи приймають вони платежі за допомогою віртуальних активів.

Оскільки криптовалюти гаманці можуть бути анонімними, використовувати їх для шахрайства легше, ніж використовувати банківські рахунки. Що стосується банківських рахунків, шахраї повинні переконатися, що перекази не повертаються до них за допомогою посередника чи інших систем, які приховують, що гроші йдуть шахраям. З криптовалютами шахраї можуть мати свій гаманець, який не має жодних доказів того, хто є особою чи групою, що стоїть за гаманцем. Повернути гроші від шахрая може бути складніше, ніж звичайні банківські перекази. Шахраї також скористалися жадібністю людей і браком знань про криптовалюту. Було багато шахрайств, коли люди інвестували в криптовалюту без відома, сподіваючись на великі прибутки, але в такому випадку гроші потрапляють безпосередньо до шахраїв. [7]

## **1.2 Технічний аналіз та гіпотеза ефективного ринку**

За допомогою технічного аналізу інвестори можуть передбачити майбутнє ринків. Технічний аналіз використовується, поки існують фондові ринки, він аналізує ринкові дані та визначає закономірності. Основне припущення технічного аналізу полягає в тому, що тенденції повторюють свої дії. Технічний аналіз не ідеальний для передбачення майбутнього. Це скоріше інструмент для прийняття кращих інвестиційних рішень. Інвестори повинні завжди знати поточну ситуацію та не стежити бездумно за технічними індикаторами. Минулі ринкові дані не є доказом майбутніх ринкових тенденцій. [8]

Технічний аналіз можна розділити на такі методи, як: діаграми, технічні індикатори та свічковий аналіз.

Патерни діаграм — це візуальні презентації вартості активів і ринкових рухів. Шаблон діаграми використовується для візуалізації трендів.

Технічні індикатори — це математичні розрахунки, які використовуються для визначення тенденцій. Технічні індикатори можна використовувати з торговими ботами, оскільки вони не є візуальними. Натомість це числа, які комп'ютери можуть порівнювати. У цьому дослідженні використовуються технічні індикатори.



Свічковий аналіз — ще один метод візуального аналізу. Замість однієї точки ціни, як на діаграмі, свічник показує найнижчу, найвищу, початкову та кінцеву ціну періоду.

Гіпотеза ефективного ринку — це гіпотеза, яка стверджує, що ціна активів відображає всю доступну інформацію. Ця теорема означає, що неможливо постійно передбачувати ринок активів, оскільки вся доступна інформація вже відображена у ціні активу. [9]

Ідея ЕМН полягає в тому, що ринок є ефективним. Розрізняють три форми ефективного ринку: слабку, напівсильну та сильну.

Слабка форма стверджує, що вся минула інформація міститься в поточних цінах. Тому аналіз минулих цін не дає жодних переваг інвесторам.

Напівсильна вказує, що вся загальнодоступна інформація в поточних цінах.

Сильна форма стверджує, що вся інформація, включаючи інсайдерську, наведена в поточних цінах, і жодна інформація не може принести користь інвестору.

ЕНМ заявляє, що технічний аналіз або будь-який інший аналіз із використанням минулих цін або іншої загальнодоступної інформації буде марним, оскільки вся інформація стосуватиметься поточної ціни. [10]

Серед дослідників точаться дискусії, чи ефективно працює ринок. Навіть якби всю інформацію з активів можна було розрахувати за цінами, неможливо розрахувати кожен подію у світі за цінами. Війни, пандемії, нові винаходи та результати досліджень впливатимуть на поточні ринки, і принаймні деякі події не так легко передбачити. [11]

Якщо технічний аналіз працює ідеально, то будь-хто, хто його використовує, буде купувати активи, доки ціна не стане такою ж, як у майбутньому. Після цього всі будуть продавати актив, доки ціна не стане настільки низькою, наскільки вона може бути. На жаль, це не так; тому технічний аналіз не завжди передбачає майбутнє.

Ринок віртуальних активів не закритий, і реальні події впливатимуть на нього. Непередбачувані події можуть сильно вплинути на ціну будь-яких інвестицій. Багато інвесторів приймають рішення лише на відчуттях. Технічний аналіз працює

на ринку, оскільки багато інвесторів не використовують технічний чи інший аналіз перед тим, як приймати рішення про інвестування.

Ще один спосіб передбачити майбутню вартість будь-якого активу — це використовувати поведінкові фінанси. У поведінкових фінансах психологія та інші соціальні науки використовуються для кращого розуміння поведінки інвесторів і того, як ця поведінка впливає на ринки.

Інвестори не завжди раціональні та не завжди приймають найкращі інвестиційні рішення. Існує багато важливих упереджень, які пояснюють таку поведінку, включаючи надмірну самовпевненість, несприйняття втрат і стадну поведінку.

Згідно з теорією ЕМН, технічний аналіз не працює, тому що все, що він показує про майбутні ціни, вже є в поточній ціні. З іншого боку, згідно з теорією поведінкових фінансів, технічний аналіз ідеально працює на ринку. Це пов'язано з тим, що коли інвестори користуються відчуттями під час прийняття рішень, можна розрахувати, куди вони інвестуватимуть і як зміниться вартість активу. [12]

Гіпотеза ефективного ринку та поведінкові фінанси забезпечують істотний теоретичний погляд на роботу ринку. Необхідно розуміти, як працює ринок, щоб робити відповідні інвестиції. Чим більше інформації, тим більша ймовірність прийняти правильні інвестиційні рішення. Розуміння поведінки ринку та того, чому інші інвестори приймають свої інвестиційні рішення, є важливим. Саме цьому метод у цьому дослідженні стосується не лише технічного аналізу, але також спирається на знання про поведінку ринку.

Утримання віртуальних активів під час їх злетів і падінь може бути нервовим, оскільки це нестабільні активи із багатьма просадками в минулому. На такому ринку легко втратити контроль через великі прибутки чи втрати за короткий проміжок часу, що ускладнює процес прийняття рішень щодо купівлі та продажу. З точки зору інвестицій, віртуальні активи є дуже спекулятивними, і можна втратити всі інвестовані кошти.

Існують інші менш ризиковані способи орієнтуватися на ринку віртуальних активів і отримувати прибуток від зростання його ціни. Можна визначити чіткий

набір правил — торгову стратегію, щоб торгувати на ринку абсолютно без емоцій. Алгоритм, визначений цими правилами, можна потім запрограмувати та автоматично генерувати сигнали «купівлі» та «продажу». [13]

Метод пошуку низьких ризикових сигналів входу та раціонального виходу на ринку називається технічним аналізом. У науковій літературі технічний аналіз визначається як «вивчення ринкових дій, головним чином за допомогою діаграм, з метою прогнозування майбутніх тенденцій цін».

Іншим підходом до торгівлі на ринку є фундаментальний аналіз, який намагається оцінити внутрішню вартість активу. Технічні аналітики сказали б, що технічний підхід включає основи. Це тому, що можна стверджувати, що все, що впливає на ціну, як фундаментальні показники, політика чи психологія, насправді вже оцінено ринком.

Віртуальні активи є відносно новим і мінливим класом активів, сприйнятливим до будь-яких новин, що робить ціну легко маніпулятивною. Технічний підхід здається розумним способом торгівлі на ринку криптовалют, оскільки він блокує багато шуму, зосереджуючись лише на графіку та деяких допоміжних технічних індикаторах. Такі індикатори є математичними конструкціями, отриманими з ціни. [14]

Нижче наведено деякі з найбільш використовуваних технічних індикаторів. Це також були індикатори, які були використані для розробки методу автоматичного аналізу в цій роботі.

Просте ковзне середнє – це середнє арифметичне ціни закриття активу за певний період. Це один з найбільш використовуваних технічних індикаторів. Чим більше період, тим повільніше він реагує на зміни ціни. [15]

Експоненціальне ковзне середнє – це особливий вид ковзного середнього, який надає більшої ваги останнім цінам закриття.

Індикатор розбіжності ковзної середньої був розроблений у 1979 році Джеральдом Аппелем. Він розраховується шляхом віднімання короткого періоду ЕМА з більш тривалого періоду ЕМА. Поверх лінії MACD може бути нанесена додаткова дев'ятиденна ЕМА, яка називається «сигнальною лінією». [16]

Смуги Боллінджера було розроблено Джоном Боллінджером у 1980-х роках. Три лінії утворюють смугу Боллінджера, а саме: SMA зі стандартним періодом 20 цін закриття, верхня та нижня смуги. Верхня та нижня смуги зазвичай у два рази перевищують стандартні відхилення за 20 періодів вище та нижче середньої смуги. Оскільки стандартне відхилення є мірою ходу ціни, відстань між нижньою та верхньою смугами стає ширшою, зі збільшенням ходу ціни активу. [17]

Також, Джон Боллінджер визначив індикатор Ширини смуги Боллінджера як ширину між верхньою та нижньою смугами Боллінджера, нормалізовану середньою смугою, як у. Якщо рух ціни невеликий, індикатор Ширини смуги Боллінджера стискається, і можна очікувати поштовху вгору або вниз.

Об'єм визначається як кількість акцій — або, у нашому випадку, кількість криптовалют, — якими торгували протягом певного часу. Об'єм сильно корелює з рухом цін та із шириною смуги Боллінджера. Обидва індикатори, об'єм та індикатор Ширини смуги Боллінджера, можна використовувати як спосіб підтвердити силу руху ціни. Об'єм не слід плутати з ліквідністю, яка є показником того, наскільки легко можна змінити ринкову ціну активу. [18]

Найвищий максимум і найнижчий мінімум – це два технічні індикатори, які відображають найвищу та найнижчу ціну протягом певного періоду всередині свічкового графіка, формуючи ціновий канал.

### **1.3 Аналіз останніх наукових досліджень**

Технічний аналіз часто використовується для прогнозування цін усіх видів фінансових інструментів. З цієї причини для багатьох ринків розроблено різні методи технічного аналізу. Однак відомо, що не всі ці методи поширені. Крім того, поведінкові фінанси можуть пояснити логіку деяких методів, але деякі методи не можуть виявити жодної логіки. З цих причин у цьому дослідженні використовувалися методи, які найчастіше використовувалися в попередніх академічних дослідженнях і яким найбільше віддають перевагу інвестори на ринках.

Agrawal, M., Khan, A. U., Shukla, P. K. провели дослідження з використанням різних технічних індикаторів, включаючи ковзні середні, експоненціальні ковзні середні, індекс відносної сили, розбіжність конвергенції ковзного середнього, стохастичний осцилятор і регресійний аналіз. Науковці провели дослідження, щоб спрогнозувати ціни на акції. Дослідження проводилося на Національній фондовій біржі Індії. В результаті дослідження спостерігалася точність понад 60%. [15]

Bustos, O., Pomares, A., Gonzales, E. досліджували ефективність простих ковзних середніх, зважених ковзних середніх, MACD, RSI, індексу товарного каналу та регресійного аналізу. Дослідження присвячено колумбійській фондовій біржі. У цьому дослідженні, в якому використовувалося машинне навчання, згадані технічні показники використовувалися як вхідні дані. Рухи вгору і вниз використовуються як вихід. Їхнє дослідження виявило, що продуктивність опорних векторних машин краща, ніж штучні нейронні мережі. [16]

У дослідженні Wang, Q., Xu, W., Zheng, H. розглядали SMA, EMA, MACD, RSI та регресійний аналіз. У їх дослідженні, яке вивчає взаємозв'язок між суспільними настроями та рухом цін на акції та демонструє вплив громадської думки на інвестиційні рішення, було розроблено нову основу для прогнозування фінансового ринку. Цей фреймворк поєднує в собі мудрість натовпу та технічний аналіз. На основі зібраних реальних даних експериментальні результати показали, що запропонований метод перевершує базові моделі щонайменше на 14,2% за значенням AUC, що вказує на ефективність DRSE як життєздатного механізму для прогнозування фінансового ринку. У цьому дослідженні підкреслюється важливість ефективної інтеграції джерел інформації для прогнозування фінансових ринків і використання методів глибокого навчання та ансамблевого навчання. [17]

Дослідження Labiad, B., Berrado, A., Benabou, L. зосереджено на SMA, EMA, RSI та індикаторі швидкості зміни. У центрі уваги цього дослідження було точне передбачення дуже короткострокових коливань, до 10 хвилин вперед, на марокканському фондовому ринку з використанням трьох різних методів машинного навчання. У цьому дослідженні керований набір технічних індикаторів слугував вхідними змінними з метою підвищення точності прогнозування та

скорочення часу навчання за допомогою процесу вибору функцій і вибірки. Експериментальний набір даних охоплював восьмирічний проміжок часу внутрішньоденних даних від акцій. Експериментальні результати однозначно продемонстрували кращу продуктивність RF і GBT над SVM. Крім того, обчислювальна простота та скорочений час навчання RF та GBT виявилися особливо придатними для короткострокового прогнозування. Це дослідження має на меті зробити внесок у поточні зусилля з прогнозування нелінійних і нестационарних цінових рухів на фінансових ринках за допомогою методів машинного навчання. [18]

Ghanavati, M., Wong, R., Chen, F., Wang, Y., Fong, S. проаналізували SMA, EMA, RSI, STO та регресійний аналіз. У цьому дослідженні дослідники вийшли за межі запропонованих підходів до прогнозування, запропонувавши структуру, яка дає змогу користувачам брати участь у прогнозуванні фондового ринку за допомогою більш широкого спектру інструментів. [19]

Patel, M. M., Tanwar, S., Gupta, R., Kumar, N. досліджували SMA, EMA, MACD, RSI, STO та регресійний аналіз. Це дослідження стосується проблеми прогнозування напрямку індексів цін на акції та цін на акції на фондових ринках Індії. Використовуючи два різні методи введення даних, було розглянуто чотири моделі прогнозування. Одна зі стратегій зосереджена на зображенні цих технічних факторів як даних, що визначають тренд, тоді як інший підхід введення даних використовує дані торгівлі акціями для розрахунку 10 технічних параметрів. Для кожної з двох процедур введення даних була оцінена точність кожної моделі прогнозування. Для проведення оцінки були використані дані історії акцій за 10 останніх років, а також індекси цін на акції. Результати експерименту показують, що Random Forest перевершує інші три моделі прогнозування з точки зору загальної продуктивності для першого методу введення даних, де десять технічних параметрів представлені як безперервні значення. Експериментальні результати також показують, що коли ці технічні фактори представлені як дані, що визначають тенденцію, ефективність усіх моделей прогнозування підвищується. [20]

Shynkevich, Y., McGinnity, T., Coleman, S., Li, Y., Belatreche, A. у своїх дослідженнях використовували SMA, WMA, MOM, MACD, RSI, CCI, STO та регресійний аналіз (R). У цьому дослідженні розглядався зв'язок між точністю прогнозування на один крок вперед (різні кроки) та розміром вікна, що використовується для розрахунку технічних показників. За допомогою алгоритмів машинного навчання та технічного аналізу було спрогнозовано майбутні рухи цін у їх загальних напрямках. Для підходу опорних векторних машин результати показали кореляцію між розміром вікна та розміром кроку прогнозування, але інші підходи не показали такої кореляції. На завершення дослідження досліджує використання технічного аналізу, машинного навчання та оптимізацію параметрів технічних індикаторів у фінансовому прогнозуванні. [21]

У своїй статті «Автоматизована торгівля біткойнами за допомогою алгоритмів машинного навчання» Madan A. та інші дослідники намагалися передбачити рух ціни віртуальних активів за допомогою методів машинного навчання. Вони використовували опорні векторні машини (SVM), випадкові ліси та узагальнений біном лінійна модель (GLM). Автори використовували кілька різних наборів даних, щоденні дані про ціну та інформацію про блокчейн і мережу Bitcoin, а також 10-хвилинні та 10-секундні дані про ціну. Продуктивність різних методів машинного навчання вимірювалася щодо цих різних наборів даних. Вони використовували розподіл навчання та перевірки 70% до 30%. Мета авторів полягала в тому, щоб передбачити знак зміни ціни, а не саму ціну, таким чином зводячи проблему до проблеми бінарної класифікації, а не до регресії. GLM дуже добре показав щоденні дані з точністю 98,79%, досягнувши лише 8,5% для 10-секундних і 53,9% точності для 10-хвилинних даних. Машина опорних векторів використовувалася для щоденних даних і показала гірші результати з точністю 27,16%, тоді як випадкові ліси показали точність 94,98% для щоденних і 57,4% для 10-хвилинних даних. Варто зазначити, що така висока точність, швидше за все, пов'язана зі значним переобладнанням, оскільки кількість навчальних вибірок для щоденних даних досить низька. Загалом автори приходять до висновку, що

оцінювані методи підходять для різної деталізації даних, оскільки щоденні дані дещо пригнічують високу волатильність віртуальних активів. [22]

McNally та інші дослідники зробили подібну спробу у своїй статті «Predicting the Price of Bitcoin Using Machine Learning». Їхньою метою також було передбачити ознаки розвитку ціни. Використані дані включали дані про ціну, а також складність майнінгу та хешрейт. Для вибору функцій був обраний алгоритм під назвою Boguta на основі випадкових лісів. Для прогнозування вони використали рекурентну нейронну мережу і мережу LSTM і порівняли їх продуктивність з моделлю ARIMA. Автори використали від 80% до 20% навчання та валідації та використали відсівання, щоб запобігти переобладнанню. Хоча як рекурентна нейронна мережа, так і мережа LSTM змогли перевершити модель ARIMA, модель LSTM мала найвищу точність 52,78% і найменшу середньоквадратичну помилку 8%. Повторювана нейронна мережа мала точність 50,25% порівняно з 50,05% ARIMA. Автори також досліджували різницю в продуктивності різних моделей на центральному та графічному процесорах, де графічний процесор значно швидший для навчання, а навчання мережі LSTM займає більше часу. Автори приходять до висновку, що LSTM трохи кращий для пошуку довгострокових залежностей, але загалом не перевершує суттєво використану рекурентну нейронну мережу. [23]

У статті «Прогнозування обмінного курсу біткойна до американського долара за допомогою методів штучної нейронної мережі» Radityo A. та інші дослідники спробували передбачити ціну Bitcoin на закриття наступного дня на основі історичних щоденних даних про ціни в поєднанні з технічними індикаторами. Це призвело до набору даних із 1278 рядків після розрахунку технічних показників і нормалізації. Набір даних було розділено на навчальний та перевірочний набір від 80% до 20%. Автори використовували нейронні мережі, які навчалися по-різному: одна із зворотним поширенням (BPNN), друга з генетичним алгоритмом (GANN), одна з гібридом цих двох методів (GABPNN) і остання розвивалася за допомогою нейроеволюції розширених топологій (NEAT). . Продуктивність вимірювалася за допомогою середньої абсолютної відсоткової помилки (MAPE). GABPNN мав найкращі показники з MAPE 1,883 ( $\pm 0,066$ )%, BPNN був близьким до 1,998



( $\pm 0,038$ )%, тоді як NEAT становив 2,175 ( $\pm 0,096$ )%, а GANN 4,461 ( $\pm 0,49$ )%. Автор також виміряв час навчання, де BPNN навчався швидше та потребував лише близько 22,5% часу GABPNN. Вони дійшли висновку, що для їхнього експерименту BPNN був найкращим вибором. [24]

У статті «Прогнозування цін на криптовалюту за допомогою машинного навчання» Алессандретті В. Було проаналізовано 1681 різну криптовалюту. Автор зібрав щоденну середню ціну, обсяг і ринкову капіталізацію за період з 11 листопада 2015 року по 24 квітня 2018 року. Потім ці ціни були виражені в кількості Bitcoin замість доларів США, щоб відкинути розвиток ринку криптовалют. Було протестовано три різні моделі, дві з використанням дерева рішень і одна з використанням мережі LSTM. Моделі використовувалися для прогнозування валют з найкращою прибутковістю, а потім розділити початковий капітал, щоб придбати портфель із найкращих  $n$  криптовалют, де  $n$  та інші параметри (наприклад, часове вікно, кількість епох) були обрані на основі коефіцієнта Шарпа та оптимізаторів середнього геометричного. Збільшення вартості портфеля за перевірений період часу вимірюється та порівнюється з простою моделлю ковзного середнього. Усі три моделі здатні перевершити модель простого ковзного середнього. Модель LSTM мала найкращу продуктивність і могла розпізнавати залежності протягом більших вікон часу. [25]

Chai С. опублікував статтю під назвою «Передбачення ціни біткойна за допомогою ансамблів нейронних мереж». Автор використовував ансамбль багаторівневих нейронних мереж прямого зв'язку між різними функціями Bitcoin та зміною ціни Bitcoin наступного дня. Використані дані включали 190 різних характеристик, включаючи ціну, обсяг і ринкову капіталізацію кількох різних бірж криптовалют, а також багато фундаментальних характеристик мережі Bitcoin, таких як хешрейт, розмір блоку тощо. Набір даних складався з 780 зразків. Було створено п'ять різних нейронних мереж, які загинули лише через кількість вузлів у двох прихованих шарах. Ці мережі тренувалися протягом 30 епох. Після навчання важливість кожної нейронної мережі була визначена відповідно до методу вибіркового ансамблю нейронних мереж, заснованого на генетичному алгоритмі.

Остаточним виходом є зважений і усереднений результат кожної нейронної мережі в ансамблі. Цей метод був перевірений протягом 50 днів, які не використовувалися для навчання, і мав точність прогнозу 60% за цей час. Зростання вартості перевершило стратегію слідування тренду попереднього дня, використану для порівняння, і збільшило його значення приблизно на 85%, порівняно з 38% слідування тренду. [26]

Вивчення попередніх досліджень показує, що останніми роками дослідження на цю тему регулярно включають різноманітні алгоритми та методи штучного інтелекту та машинного навчання. Машини опорних векторів, глибоке навчання, штучні нейронні мережі, нечітка логіка, генетичні алгоритми та дерева рішень є кількома прикладами цих методів. З розвитком комп'ютерних технологій зросла здатність робити висновки щодо фінансових питань. Це дає дослідникам шанс використовувати величезні дані для створення більш точних спостережень. Особливо при застосуванні до історичних даних алгоритми класифікації дають точні результати. Зокрема, алгоритми класифікації забезпечують чіткі висновки на основі минулих даних. З цих причин, щоб виміряти валідність технічного аналізу для ринку віртуальних активів, за допомогою розробленого програмного забезпечення були протестовані такі індикатори, як: SMA, EMA, MACD, BB та RSI.

## 2 АНАЛІЗ МАТЕМАТИЧНИХ МОДЕЛЕЙ ТА ВИЗНАЧЕННЯ ОПТИМАЛЬНИХ ПАРАМЕТРІВ СИСТЕМИ ПРОГНОЗУВАННЯ

### 2.1 Аналіз нейромережових технологій

Рекурентна нейронна мережа підходить для даного дослідження, оскільки її модель, зазвичай, призначена для перетворення вхідної послідовності, у нашому випадку часового ряду, однієї предметної області, у вихідну послідовність, цієї предметній області. RNN створювалася з можливістю обробки довгих послідовних даних та вирішення завдань із розподілом контексту в часі. [27]

Модель обробляє один елемент у послідовності на одному часовому кроці. Після обчислення оновлений стан передається на наступний крок часу, щоб полегшити обчислення наступного елемента.

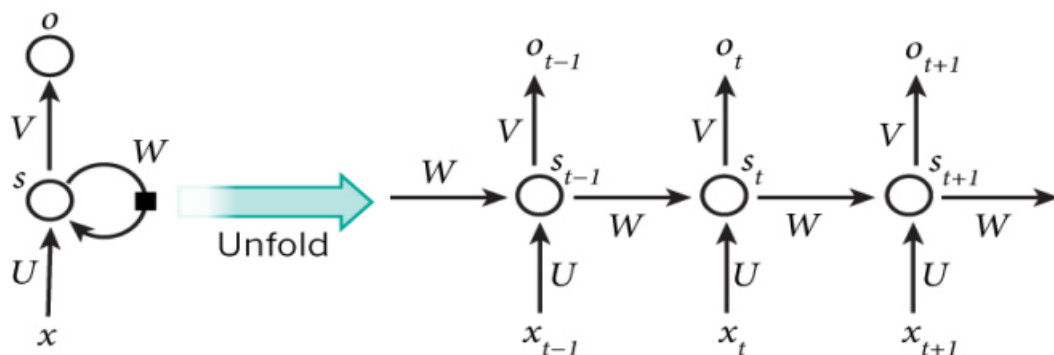


Рис.2.1 Рекурентна нейронна мережа з одним прихованим елементом та її версією, що розкривається з часом

На рис.2.1 можна побачити, що відбувається з часом:  $s_{(t-1)}$ ,  $s_t$  та  $s_{(t+1)}$  - це одиниця з різними станами на різних тимчасових кроках  $t-1$ ,  $t$  і  $t+1$ .

Але є ймовірність того, що прості перцептронні мережі, що лінійно поєднують поточний елемент введення та останній елемент виведення, можуть з легкістю втратити довгострокові залежності.

Для вирішення проблем забування, був створений нейрон з більш складною внутрішньою структурою для запам'ятовування довгострокового контексту, що називається осередком LSTM. Він досить розумний, щоб дізнатися, як довго має запам'ятовувати стару інформацію, коли використовувати нові дані та як поєднати стару пам'ять із новим входом. На рис.2.2 зображена структура LSTM, ланцюг містить 4 шари та взаємодія відбувається між ними особливим способом. [28]

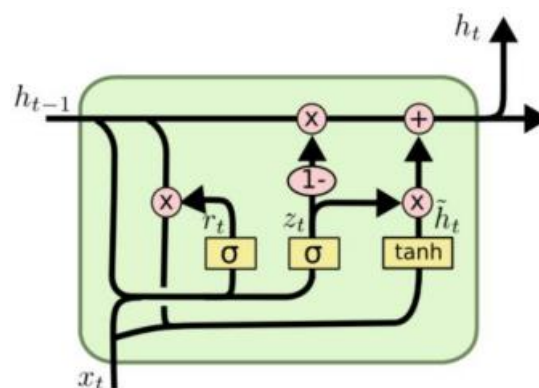


Рис.2.2 Структура LSTM нейрона

Така архітектура рекурентної нейронної мережі не використовує функцію активації, тим самим значення про помилки не розмиваються з часом, і перебувають у пам'яті мережі.

Дана архітектура обчислює, наскільки подане значення використовується у вихідному значенні, забуваючи старе значення, якщо з'явиться нове значення, тим самим реалізуючи більш інтелектуальне використання своєї пам'яті.

Найголовніший компонент LSTM – це стан осередку, а саме горизонтальна лінія на рис.2.3, що проходить по верхній частині схеми, що проходить через весь ланцюжок і бере участь лише кількох лінійних перетвореннях. Інформація про помилку може легко проходити нею, не піддаючись змінам. [28]

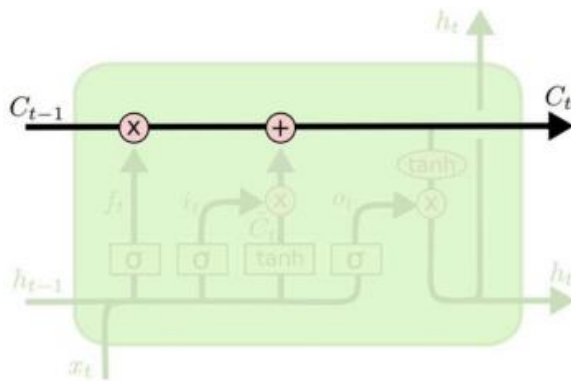


Рис.2.3 Горизонтальна лінія LSTM

Але тим часом, LSTM може видаляти інформацію з осередку, цей процес регулюється структурами, які називаються фільтрами. Фільтри можуть дозволяти пропускати інформацію на підставі певних умов. На рис.2.4 можна побачити, що вони складаються з шару сигмоїдальної нейронної мережі та операції поточкового множення. [29]

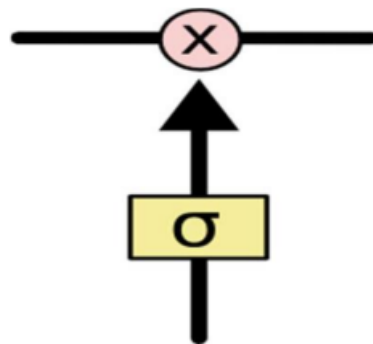


Рис.2.4 Фільтри в LSTM

Сигмоїдальний шар повертає числа від 0 до 1, що позначають, яку частку кожного блоку інформації слід пропустити далі по мережі, 0 означає "не пропускати нічого", 1 - "пропустити все". У LSTM міститься три такі фільтри, що дозволяють захищати та контролювати стан комірки.

Така архітектура нейронної мережі, як багат шаровий перцептрон використовується зараз найчастіше. Вона була запропонована Румельхартом в 1986 році і є окремим випадком перцептронів Розенблатта, в якому один алгоритм зворотного поширення помилки навчає всі верстви. [30]

Кожен елемент мережі будує виважену суму своїх входів з поправкою у вигляді доданку і потім пропускає цю величину активації через передану функцію, і цим є вихідне значення початкового елемента. Елементи організовані в пошарову топологію із прямою передачею сигналу. Таку мережу легко можна інтерпретувати як модель вхід-вихід, в якій ваги та порогові значення є вільними параметрами моделі.

Така мережа може моделювати функцію практично будь-якого ступеня складності, кількість шарів та кількість елементів у кожному шарі визначають складність функції. Визначення числа проміжних шарів та числа елементів є важливим питанням при конструюванні MLP. [30]

Кількість вхідних та вихідних елементів визначається умовами завдання. Складність може полягати в тому, якими вхідними значеннями користуватися, а якими не варто. Тепер відзначимо, що це вхідні змінні обрані інтуїтивно і є значущими. Дилема в тому, скільки використовувати проміжні шари та елементи в них, абсолютно невідома. Як початкове наближення можна взяти один проміжний шар, а число елементів у ньому зробити рівним половині числа суми вхідних і вихідних елементів.

Після того, як виявлено кількість шарів та елементів у кожному з них, потрібно шукати значення для ваг та порогів мережі, що мінімізує помилку прогнозу, отриманого мережею, для цього можемо використовувати алгоритми навчання. Із застосуванням зібраних історичних даних ваги та порогові значення автоматично коригуються з метою мінімізації цієї помилки.

У результаті такий процес представимо у вигляді моделі, що реалізується мережею, до наявних навчальних даних. Помилка для конкретної конфігурації мережі формується шляхом прогону через мережу всіх наявних спостережень і порівняння вихідних значень, що реально видаються, з цільовими значеннями. Усі такі різниці підсумовуються у функцію помилок, значенням якої є помилка мережі.

Як функцію помилок часто застосовують суму квадратів помилок, а саме всі помилки вихідних елементів всіх спостережень зводяться в квадрат і після

складаються. У традиційному лінійному моделюванні можна алгоритмічно вказати конфігурацію моделі, що видає абсолютний мінімум для зазначеної помилки. [31]

Ціна, яку необхідно заплатити за більш розширені, нелінійні можливості моделювання за допомогою нейронних мереж, полягає в тому, що коригуючи мережу з метою мінімізувати помилку, ніколи не зможемо бути впевнені в тому, що неможливо досягти ще меншої помилки.

У таких дослідженнях, також корисним буде визначення поверхні помилок. Кожному з ваг і порогів мережі, тобто вільних параметрів моделі, їх загальне число позначимо через  $N$ , відповідатиме один вимір у багатовимірному просторі. В такому разі  $N+1$  вимір дорівнює помилці мережі. Для будь-яких можливих поєднань ваги відповідну помилку мережі можна зобразити точкою в  $N+1$ -мірному просторі, і всі такі точки утворюють там деяку поверхню - поверхню помилок. Метою навчання нейронної мережі буде задача знаходження на цій багатовимірній поверхні найнижчої точки. [30]

У разі лінійної моделі із сумою квадратів як функція помилок така поверхня помилок буде у вигляді параболоїда - гладкої поверхні, схожої на частину поверхні сфери, з єдиним мінімумом.

У випадку нейронної мережі поверхня помилок має більш складну будову і має ряд негативних властивостей, наприклад, мати локальні мінімуми, плоскі ділянки, сідлові точки і довгі вузькі яри. [31]

Аналітичними засобами неможливо визначити положення глобального мінімуму, що лежить на поверхні помилок, через це навчання нейронної мережі зводиться до дослідження поверхні помилок.

Відштовхуючись від початкової конфігурації ваги і порогів, тобто випадково взятої точки на поверхні помилок, алгоритм навчання поступово знаходить глобальний мінімум. Для цього обчислюється градієнт поверхні помилок у цій точці, і ця інформація використовується для просування вниз по схилу. За підсумком алгоритм робить зупинку в нижній точці, яка може бути лише локальним мінімумом.

## 2.2 Класичні алгоритми машинного навчання

Регуляризація вважається ефективним методом усунення перенавчання. Адже справжній розподіл даних невідомий і емпіричний ризик, взятий на основі емпіричного розподілу, матиме схильність до перенавчання. Тому, найкращою стратегією буде ситуація, де є можливість добре підібрати дані навчання, а потім застосувати метод регуляризації для узагальнення моделі. Одним із неявних припущень методів регуляризації, таких як регуляризація параметрів L1 та L2, полягає в тому, що величина параметрів повинна дорівнювати нулю і прагнути стиснути всі параметри до нуля. Це говорить, що необхідно уникати дуже хорошого слідування навчальним даним, що може змушувати алгоритм навчання виділяти певний шум, некорисний при використанні до невидимих даних. Застосовувати регуляризацію треба лише тоді, коли модель страждає від перенавчання. [32]

В основі L1 лежить додавання штрафу до початкової функції витрат, в L1-регуляризації застосовується спеціальне L1 нормування. Називається ця регуляризація ласо-регресії.

L2-регуляризація називається ще регресією Ріджа. Суть у тому, що проводиться зміна початкової функції, додаванням «штрафу» великі вагові коефіцієнти. [33]

Відомо, що L1-регуляризація сприяє розрідженості функції, коли лише деякі чинники не дорівнюють нулю. L2-регуляризація сприяє появі малих вагових коефіцієнтів моделі, але з сприяє їх точної рівності нулю. Тому що потрібно звернути увагу на те, що обидва методи допомагають покращити узагальнення та помилки тесту, оскільки не допускають перенавчання моделі через шум у даних.

L1-регуляризація реалізує це шляхом відбору найбільш важливих факторів, що найсильніше впливають на результат. Можна вважати, що фактори з малою величиною впливу на кінцевий результат фактично «допомагають» вам передбачати лише шум набору навчальних даних.

L2-регуляризація перешкоджає перенавченості моделі шляхом заборони непропорційно величезні вагові коефіцієнти.



Контурні графіки негативних логарифмів на рис.2.5 показують різницю між кожним із типів регуляризації.

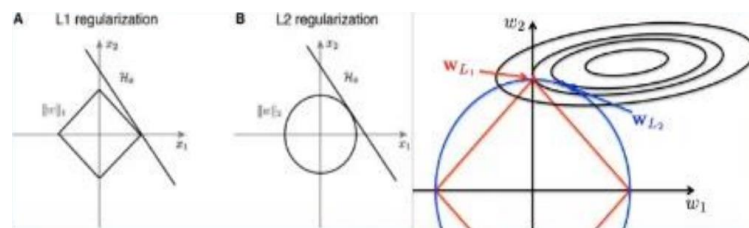


Рис.2.5 Відмінності L1 та L2 регуляризації

При L2-регуляризації додатковий член буде квадратичною функцією, при L1-регуляризації - модулем. Тут похідна функції є ключовою, адже градієнтний спуск в основному рухається в напрямку похідної. При квадратичному члені чим ближче перебування до нуля, тим меншою буде похідна, поки також не наблизиться до нуля. Тому при L2-регуляризації, коли величина  $w$  вже мала, подальший градієнтний спуск вже її не змінить. У випадку похідна модуля буде константою з абсолютною величиною, що дорівнює одиниці. Формально в нулі вона не визначена, але вважаємо її також рівною нулю. [33]

Тому при L1-регуляризації градієнтний спуск буде наближатись до нуля з постійною швидкістю, а досягнувши його, там і залишиться. Через це L2-регуляризація сприяє малій величині вагових коефіцієнтів, а L1-регуляризація сприяє їх рівності нулю, тим самим провокуючи розрідженість.

Лінійна регресія — це регресійна модель, що використовується в статистиці, залежності однієї, залежної змінної від іншої незалежної. Лінійна регресія відноситься до контрольованого навчання. Є ряд значень даних  $x$  та  $y$ . Завдавши їх на графік, можна розглядати кожне значення  $x$  як вхідну змінну, в результаті якої на виході отримувати деяке вихідне значення  $y$ . Контрольоване навчання, своєю чергою, також поділяється на великі категорії - класифікацію і регресію. [34]

Класифікація намагається визначити категорію вхідних даних, наявність чи відсутність їх якоїсь особливості – наприклад, намагається розпізнати написану цифру чи визначити, чи міститься на зображенні предмет. Регресія ж обчислює

певну кількість чи вектор – наприклад, завтрашню температуру або, в нашому разі, ціни на криптовалюти.

Розглянемо тепер власне лінійну регресію. Тут є залежність між  $x$  і  $y$ , причому вихідне значення є числом. Багато науковців називають регресію «лінією найкращої відповідності», оскільки якщо нанести на графік, на рис.2.6, точки даних  $x$  і  $y$  і провести відповідним чином пряму, то вийде лінія найкращої відповідності. [35]

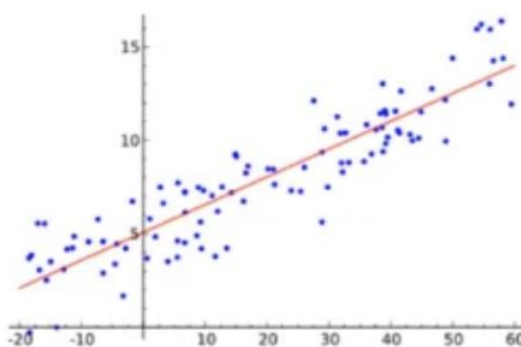


Рис.2.6 Графік лінійної регресії

Цільова функція даної лінійної регресійної моделі, має такий вигляд:

$$f(x, b) = b_0 + b_1x_1 + \dots + b_kx_k \quad (2.1)$$

де  $b_i$  - параметри (коефіцієнти) регресії,  $x_i$  - регресори (фактори моделі),  $k$  - кількість факторів моделі.

Лінійна регресія використовує простий функціонал якості – середньоквадратичну помилку. У ході завдання нам доведеться працювати з вибіркою, що містить 6 ознак. Для налаштування параметрів (ваги) моделі вирішується наступне завдання:

$$\frac{1}{l} \sum_{i=1}^l ((w_0 + w_1x_{i1} + w_2x_{i2} + \dots + w_7x_{i6}) - y_i)^2 \rightarrow \min_{w_1, w_2, \dots, w_6} \quad (2.2)$$

де  $x_{i1}, x_{i2}, \dots, x_{i6}$  - значення ознак  $x_i$  об'єкта,  $y_i$  – значення цільової ознаки  $i$ -го об'єкта,  $l$  - кількість об'єктів у навчальній вибірці. [35]

Багатовимірна лінійна регресія, називається також множинною лінійною регресією.

Також, варто відмітити, що композиції вирішальних дерев часто показують високу якість роботи у багатьох завданнях. Популярним і ефективним методом є випадковий ліс. Побудова кожного вирішального дерева може брати до уваги і категоріальні ознаки. Для цього в кожному вузлі для вибору розбиття відбувається перебір розбиття значень ознак на дві множини, тобто на лівого та правого спадкоємця у дереві.

Оскільки перебір різноманітних значень має велику обчислювальну складність, то на практиці частіше використовується варіант з перебором деякої довільної частини розбиття.

Випадковий ліс) представимо у вигляді ансамблю дерев рішень, які навчаються незалежно. Кожен із вирішальних дерев, зазвичай, роблять простим. Для ухвалення остаточного рішення обирається значення, за яке проголосувала більшість дерев. Кожне дерево окремо дає низьку якість, але за рахунок їх великої кількості результат покращується. На рис.2.7 показано приклад випадкового лісу з двох дерев. [35]

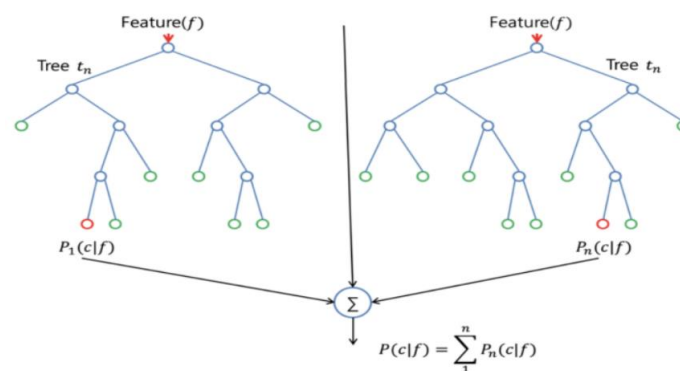


Рис.2.7 Випадковий ліс із двох дерев

У побудові дерев на даних є свої особливості, тож опишемо традиційний підхід у побудові дерева. Нехай  $N$  – кількість прикладів на навчання, тоді:

- обирається частина навчальної вибірки розміру  $N$ ;

- будується дерево, при пошуку розбиття під час побудови кожного вузла розглядаються не всі ознаки, а лише частина з них;
- дерево будується до вичерпання підвибірки, або на основі будь-яких інших критеріїв.

Кожне з дерев намагаються зробити якомога простіше. При ухваленні рішення обирається клас, за який проголосувало більшість дерев рішень.

Випадковий ліс дає порівняно погані результати при поділі вибірки у метричному просторі. У тому випадку, коли очікується певний «правильний» вид роздільної гіперплощини. Тим не менш, у випадкового лісу є велика кількість переваг, які можуть використовуватися в задачі визначення шкідливості файлу. Алгоритм не оперує метриками, що дозволяє вільно працювати з ознаками різної природи. Можна обробляти дані з пропущеними значеннями ознак. [35]

Також як порівнюване рішення було обрано метод k найближчих сусідів. Даний алгоритм вимагає способу обчислення відстані між об'єктами, тобто основою методу є гіпотеза компактності: якщо метрика відстані між прикладами введена дуже вдало, то схожі приклади набагато частіше знаходяться в одному класі, ніж у різних. Є популярним методом класифікації, що часто використовується в задачах регресії. У науковій літературі цей метод вважається, одним із найзрозуміліших підходів до класифікації. [36]

У багатьох навчальних мережах помилка розраховується як різниця між фактичним вихідним значенням і прогнозованим вихідним значенням. Функція, застосовувана обчислення цієї помилки - функція втрат чи функція витрат, помилки.

Середня квадратична помилка – є однією з найбільш поширених функцією втрат. Функція втрат MSE масово застосовується в лінійній регресії як показник ефективності. Щоб розрахувати MSE, потрібно взяти різницю між передбаченими значеннями та істинними, звести її у квадрат і усереднити по всьому набору даних:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.3)$$

де  $y^{(i)}$  – фактичний очікуваний результат, а  $\hat{y}^{(i)}$  – прогноз моделі. [37]

Також нам потрібен якийсь кількісний показник того, наскільки гарна наша модель. Зазвичай для цього в лінійній регресії, і не тільки в лінійній, а й у інших, використовується коефіцієнт детермінації  $R^2$ .

Коефіцієнт детермінації визначається як:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (2.4)$$

де  $SS_{res}$  сума квадратів залишків регресії,  $SS_{tot}$  – загальна сума квадратів помилки. [38]

### 2.3 Огляд бібліотек та фреймворків використаних у роботі

Більшість представлених фреймворків представляють модель як обчислювальний граф, статичного чи динамічного.

Фреймворки, що оголошують статичні графи обчислень, такі як TensorFlow, дозволяють спочатку визначити архітектуру графа, а потім виконати його багато разів, прогнавши через нього деякий обсяг даних. Це дозволяє легко розподіляти завдання виконання на різних машинах. Крім цього, фреймворки можуть оптимізувати граф для нас заздалегідь. Вони також дозволяють користувачам серіалізувати графік після його створення та виконувати його, не вимагаючи коду, що його побудував.

Для реалізації обраних моделей було обрано фреймворк TensorFlow. Популярність даного фреймворку цілком заслужена: є підтримка великої кількості розробників, широкий вибір документації, відкритий вихідний код, розподілене виконання наших моделей, різні архітектурні оптимізації, візуалізацію під час виконання (що спрощує налагодження моделі) та портативність.

Як фреймворк для імплементації візьмемо Keras - він дуже простий, інтуїтивно зрозумілий і є можливість просто реалізовувати досить складні обчислювальні графи. Також Keras дозволяє досить гнучко контролювати процес навчання.

Можливості фреймворку Jupyter Notebook дозволяють взаємодіяти із середовищем Python через інтерактивний веб-інтерфейс шляхом введення команд у поля веб ноутбука. Пакет Jupyter Notebook використовується разом з великою кількістю бібліотек і доповнень. Одним із найважливіших з них є пакет Pandas. Ця бібліотека підтримує роботу як з вбудованими типами Python, так і завантаження даних з диска в форматах TXT, TSV, CSV і Microsoft Excel. Також пакет надає базові можливості щодо передоброби та візуалізації даних.

Наступним пакетом, без якого неможливо будь-яке дослідження у цьому середовищі є пакет NumPy. Основною її перевагою перед бібліотекою Pandas є широкий набір функціоналу роботи з багатовимірними даними, без яких неможливо уявити жодну комплексну модель машинного навчання.

Також необхідно відзначити пакет Scikit-learn, що містить набір ряду алгоритмів машинного навчання, що використовуються для створення моделей. Цей пакет містить у собі великий набір методів для передоброби, аналізу та візуалізації даних. Ця бібліотека включає такі методи класифікації, як DecisionTreeClassifier, RandomForestClassifier, GradientBoostingClassifier та інші.

## **2.4 Підготовка даних**

У цьому дослідженні для однакових наборів даних використовуються однакові методи технічного аналізу з різними налаштуваннями. Тому немає точної відповіді на те, який метод технічного аналізу або які налаштування на якій криптовалюті будуть найкращими. На будь-якому ринку активів не може бути простої відповіді, яка працює найкраще, оскільки ринкові ситуації надто змінюються, а різні методи та налаштування по-різному працюють у різних ринкових ситуаціях. Наприклад, припустимо, що технічний аналіз використовується для реальних інвестицій. У цьому випадку ринок повинен постійно досліджувати, щоб визначити, який метод або налаштування буде найкращим у поточній ситуації.

У цьому дослідженні використовувалися публічні дані для розрахунку ефективності різних методів технічного аналізу, та машинного навчання. Набір даних було зібрано з криптовалютних біржі Coinbase. У цьому дослідженні використовуються дані з торговими парами USD, щоб отримати сумісні результати. В даній роботі використовувалися дані торгів криптовалюти Litecoin з 01.01.2015 р. до 01.01.2023 р., оскільки деякі методи технічного аналізу потребують більше часу та історичних даних, ніж інші.

У цьому дослідженні використовувалися найпоширеніші базові індикатори технічного аналізу, а саме: смуги Боллінджера, експоненціальне ковзне середнє, розбіжність конвергенції ковзного середнього, індекс відносної сили і просте ковзне середнє. Усі методи були перевірені на різних налаштуваннях.

Смуги Боллінджера мають три лінії: верхню, середню та нижню. Відстань між верхньою та нижньою смугами показує волатильність. Коли смуги розташовані далеко, волатильність вища, ніж коли смуги близькі. Коли середня смуга торкається або перетинає верхню смугу, валюта вважається перепроданою та вказує на покупку. Коли середня смуга торкається або перетинає нижню смугу, вважається, що валюта перекуплена, і вказує на сигнал про продаж. Смуги Боллінджера використовувалися з таким налаштуванням:

- значення періодів були 20, 21 та 22;
- значення стандартного відхилення були 2, 2,5 та 3;
- тип ковзного середнього був простим ковзним середнім, експоненціальним ковзним середнім та зваженим ковзним середнім.

Просте ковзне середнє використовується для визначення тенденцій у ціні. SMA розраховується шляхом розрахунку середньої ціни за період. Коли поточна ціна вище середньої, це вважається сигналом до покупки; коли він нижче середнього, це вважається сигналом до продажу. [39]

SMA – це метод, який використовується для розрахунку середньої ціни активу в певний час. У цьому дослідженні SMA використовується з такими налаштуваннями:

- «10, 0, 20, 0»;

- «50, 0, 100, 0»;
- «50, 0, 200, 0»;
- «20, 0, 50, 0» ;
- «10, 0, 50, 0»;

Перше число - кількість періодів. Друге число - зсув. Третє число використовується для розрахунку верхньої смуги індикатора ВВ. Четверте число використовується для розрахунку нижньої смуги індикатора смуг Боллінджера.

Експоненціальне ковзне середнє схоже на SMA. Різниця полягає в тому, що в ЕМА останні ціни важать більше, ніж попередні. Це робить ЕМА більш відповідальним за зміни, ніж SMA. Досі незрозуміло, який з них кращий для прогнозування ринку. ЕМА отримуватиме зміни швидше, але повільніша реакція на сигнал також може означати, що реакція є точнішою. Коли SMA реагує на сигнал, сигнал є більш важливим. [40]

Для індикатору ЕМА є чотири налаштування, а саме кількість періодів, згрупованих для розрахунку. ЕМА не можна обчислити, якщо часова шкала має менше періодів, ніж у цьому параметрі. Друге число представляє зсув. Зміщення використовується для зсуву ЕМА вліво або вправо. Зсув використовується, коли інвестор хоче знати, як індикатор буде виглядати в майбутньому чи минулому. У цьому дослідженні офсет не використовувався. Третім параметром є коефіцієнт згладжування. Він представляє кількість періодів, використаних для згладжування ЕМА. Фактори згладжування зменшують шум і короткочасні зміни. Четвертий параметр зсув для коефіцієнта згладжування. Він не використовувався в цьому дослідженні, тому завжди буде приймати значення 0.

Параметри, які використовуються в цьому дослідженні для індикатору ЕМА:

- «10, 0, 20, 0»;
- «50, 0, 100, 0»;
- «50, 0, 200, 0»;
- «20, 0, 50, 0»;
- «10, 0, 50, 0».



Розбіжність конвергенції ковзного середнього обчислюється шляхом віднімання довшої ЕМА з коротшої ЕМА, а потім порівняння з найкоротшою ЕМА, яка використовується для сигнальної лінії. Наприклад, типовим параметром для MACD є віднімання 26-періодної ЕМА від 12-періодної ЕМА та порівняння її з 9-періодною ЕМА. Коли розрахована лінія MACD перетинає сигнальну лінію і знаходиться над нею, це сигнал до покупки. Коли він падає нижче лінії, це вказує на сигнал про продаж. [36]

У ковзному середньому є три налаштування. Перше число представляє періоди для обчислення швидкої експоненціальної ковзної середньої. Друге число вказує на кількість періодів, які використовуються для розрахунку повільніше ЕМА. Третє число представляє кількість періодів, які використовуються для обчислення сигнальної лінії. MACD розраховується шляхом віднімання повільної ЕМА від швидкої ЕМА.

Параметри, які використовуються для індикатора MACD:

- «12, 26, 9»;
- «5, 35, 5»;
- «8, 17, 9»;
- «7, 14, 7»;
- «21, 55, 8»;
- «10, 20, 5»;
- «10, 30, 5»;
- «15, 30, 5»;
- «18, 35, 6»;
- «6, 13, 5».

Індекс відносної сили відрізняється від попередніх індикаторів тим, що це лише одне число, а не лінія. Індикатор RSI – це число від 0 до 100. RSI розраховується як відношення середнього прибутку до середнього збитку за кількість обраних періодів. Потім результат масштабується до 0–100. У звичайних випадках ринок вважається перепроданим, коли значення перевищує 70, і

перекупленим, коли значення нижче 30. Якщо значення перевищує 50, це вказує на висхідний тренд, коли він нижче 50, це вказує на спадний тренд. [34]

Усі індикатори перевірені таким чином, що після не менш ніж 200 днів аналізу даних програма буде інвестувати, якщо поточний метод бачить, що настав сприятливий час для інвестування, або продавати, якщо індикатор передбачає, що ринкова ціна буде нижчою. Для отримання більш реалістичних результатів програма вкладатиме лише 1000 доларів США за один раз, і після торгівлі не здійснюватиме іншу торгівлю протягом наступної хвилини. Комісія за торгівлю – це невеликий платіж, який надходить на ринок криптовалют. У цьому дослідженні комісія за торгівлю була визначена як 0,075%, для кожної угоди та кожного методу технічного аналізу застосовувалася однакова торгова комісія. Початковий капітал визначений в програмі 100000 доларів США, якщо початкові суми менші за суму найбільшої можливої торгівлі, то програма завжди може торгувати всіма грошима. Це може бути реалістичною ситуацією, коли доступні гроші для інвестування невеликі, але нереалістичною, якщо вони достатньо великі, оскільки під час торгівлі значними сумами грошей недостатньо доступних продавців або покупців, і трейдер впливатиме на ціну криптовалюти. Торгівля завжди впливатиме на ціну, але в цьому дослідженні підраховано, що 1000 доларів США за хвилину не суттєво впливають на ціну на ринку, і результати досить реалістичні.

Для алгоритмів машинного навчання, у роботі як набір даних будемо використовувати OHLC, а також інші важливі ознаки, такі як тональність повідомлень та аналіз новин. Прогноз алгоритму на певний день буде здійснюватись за рахунок попередніх 10 днів торгів протягом одного обраного нами місяця. В результаті роботи вийде прогноз на 30 днів.

Висвітливо основні закономірності використання індикатора volume, при односпрямованому зростанні обсягу та цін слід ситуація, що ринок підтримує тренд. У цьому випадку при висхідному русі цін радять купувати, при низхідному продавати. Якщо ж обсяг знижується у разі зростання цін, це означає відсутність підтримки тренда з боку ринку. Така ситуація сигналізує про можливий розворот. Volume часто визначає зміни цін. Коли ціна зростає разом з обсягами, а потім

починає зниження, деякий час ціна ще зростатиме за інерцією. Головним недоліком індикатора є його схильність до сильних стрибків. Його значення можуть спотворювати реальну ситуацію, особливо у разі використання на валютному ринку. Тому потрібно заздалегідь тестувати його можливості. [35]

OHLC – це скорочене позначення котирувань, які вказуються для елементарної діаграми будь-якого цінового графіка. OHLC бар є важливим етапом у торгівлі без індикаторів. У цій аббревіатурі: О означає Open – ціну відкриття інтервалу, Н означає High – максимум ціни інтервалу, L означає Low – мінімум ціни інтервалу, С означає Close – ціну закриття інтервалу. [28]

Кожен графік типу OHLC, є класичний стовпчиковий графік, бар. У ньому кожен інтервал часу, наприклад, 5 хвилин, представлений OHLC цінами прямо всередині цього інтервалу. Іншими словами, це основна одиниця торгівлі на біржі, що містить ціни відкриття та закриття, максимальне та мінімальне значення ціни за період часу, що міститься в барі.

Ціна відкриття - Open, означає знаходження ціни в момент початку даного бару, тобто в самий початковий момент часу певного часового відрізка, описаного в барі. Цей параметр вказує, з якої ціни почався певний часовий відрізок на ринку акцій. Ціна закриття - Close, значенням, яке набуває ціна за певний проміжок часу, що міститься в конкретному барі.

Варто зазначити, що ціна закриття може не співпадати з ціною відкриття наступного бару, особливо тоді, коли ціни змінюються дуже швидко. Між ціною закриття старого бару та відкриття нового минає досить багато часу, якого вистачає для зміни ціни. Максимальне та мінімальне значення ціни – High та Low, говорять про найбільше чи найменше значення, яке набувалося ціною за час формування одного конкретного бару.

Рекурентна нейронна мережа є тип штучної нейронної мережі з рекурентними переходами в прихованих шарах, що дозволяє їй використовувати попередній стан прихованих нейронів для отримання нового результату з урахуванням нового введення. RNN може обробляти послідовні дані. [34]

Довга короткострокова пам'ять - спеціально розроблений осередок, що допомагає RNN краще запам'ятовувати довгостроковий контекст.

Модель RNN, яку ми будемо, має LSTM-осередки як основні приховані елементи. Ми використовуємо значення від початку навчання в першому ковзному вікні  $W_0$  до вікна  $W_t$  в момент часу  $t$ , як показано на рис. 2.8.

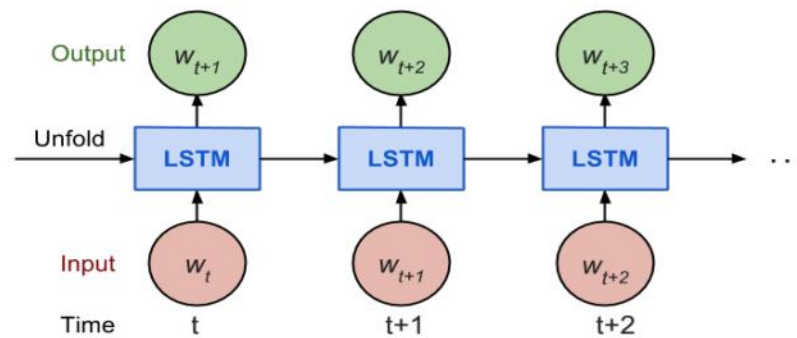


Рис.2.8 Розгорнута версія RNN

Розглядаючи, як працює зворотне поширення в часі, ми зазвичай навчаємо RNN у розгорнутій версії, так що немає сенсу проводити зворотне поширення дуже далеко за часом, збільшуючи складність навчання.

Вихід рекурентної нейронної мережі залежить від довільно віддалених входів. На жаль, це ускладнює обчислення зворотного розповсюдження. Щоб зробити процес навчання прийнятним, звичайною практикою є створення «розгорнутої» версії мережі, що містить фіксоване число, `num_steps`, входів та виходів LSTM. Потім модель навчається у цьому кінцевому наближенні RNN.

Це може бути реалізовано шляхом подачі вхідних даних завдовжки `num_steps` за один раз та виконання зворотного проходу після кожного такого вхідного блоку. Послідовність цін спочатку поділяється на невеликі вікна, що не перекривають один одного. Кожне з них містить числа `input_size`, кожне з яких розглядається як один незалежний елемент введення. Потім будь-які послідовні елементи введення `num_steps` групуються в одне введення для навчання, утворюючи «нерозкручену» версію RNN для навчання TensorFlow. Відповідна мітка є елементом введення відразу після них.

У рекурентній нейронній мережі на одному часовому кроці  $t$ , вхідний вектор містить `input_size`, позначений як  $W$ , а також добові значення ціни  $i$ -го активу, а саме « $p_{i,tw}, p_{i,tw+1}, \dots, p_{i,(t+1)W-1}$ ». Актив відображається на векторі довжини `embedding_size`, позначений як  $k$ , та містить « $e_{i0}, e_{i1}, \dots, e_{ik}$ ». Як показано на рис. 2.9 вектор ціни конкатенується з вектором вкладення, а потім подається в комірку LSTM. [30]

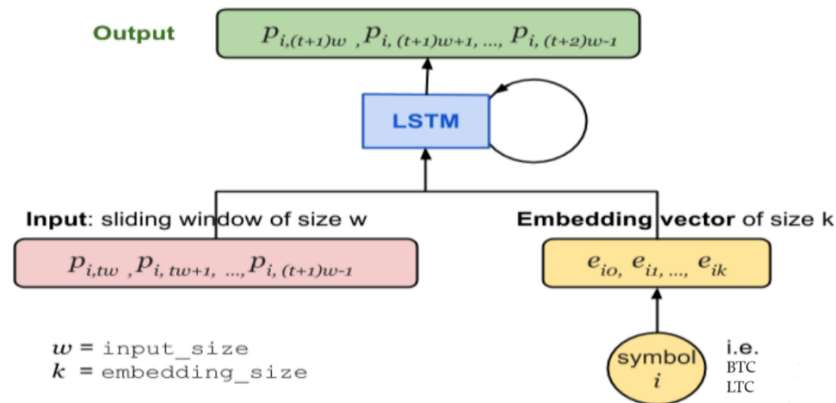


Рис.2.9 Архітектура моделі RNN для прогнозування цін на криптовалюти

Іншою альтернативою є конкатенація векторів вкладення з останнім станом осередку LSTM та отриманням нових ваг  $W$  і зміщення  $b$  у вихідному шарі. Але, таким чином, LSTM-елемент не може відрізнити ціни однієї акції від іншої, і його потужність значною мірою стримуватиметься.

Існує кілька різних методів нормалізації даних, в даній роботі буде використовуватися такий метод, як нормалізація `min-max`, який встановлює найменшим значенням, що спостерігається - 0, а найбільшим значенням - 1.

Нормалізація стандартного відхилення обчислюється за формулою:

$$z = \frac{x - \mu}{\sigma} \quad (2.5)$$

де,  $\mu$  – середнє,  $\sigma$  – стандартне відхилення. [36]

Аналіз основних компонентів – це статистична процедура для перетворення набору спостережень за можливими корельованими змінними на набір значень лінійно некорельованих змінних.

Кожен з основних компонентів вибирається таким чином, щоб він описував більшу частину все ще доступної дисперсії, і всі ці основні компоненти ортогональні один одному. У всіх основних компонентах перший головний компонент має максимальну дисперсію.

У роботі ми будемо використовувати метод основних компонентів для пошуку взаємозв'язку між змінними у даних, а також для інтерпретації та візуалізації даних. Оскільки кількість змінних зменшується, це спрощує подальший аналіз.

## 2.5 Побудова та налаштування моделі

Для початку підключимо та імпортуємо потрібні для роботи бібліотеки та скрипти у Python. Визначення, які будуть використовуватися в моделі: `start` та `end` - період, на якому буде проходити навчання; `start_test` та `end_test` - період, на якому проходитиме передбачення; `seq_len` – кількість попередніх денних цін (ознак), на основі яких робиться прогноз ціни дня.

Отримує історичні дані за денними цінами криптовалюти Litecoin між датами `param ticker` - компанія або компанії, чий дані мають бути вилучені; `type ticker` - string або list of strings; `param start_date` – початкова дата отримання цін на акції; `type start_date` - string of date "YYYY-mm-dd"; `param end_date` – кінцева дата отримання цін на акції; `type end_date` - string of date "YYYY-mm-dd"; `param file` - ім'я файлу, що повертається з даними; `return` - файл формату CSV.

Перетворюємо дані, розбиваючи на ознаки та відповіді, для цього нам потрібні такі параметри: `data` - вихідний масив даних; `list_x` - список, до якого додаються ознаки; `list_y` - список, до якого додаються відповіді.

Отримаємо три списки даних: тренувальний, для передбачення та для налаштування скалера. Після цього розіб'ємо отримані дані на ознаки та відповіді, перетворимо дані на формат, що використовується в нейронній мережі та зробимо скалінг даних.

Для ознак OHLC проведемо додаткову нормалізацію даних за допомогою аналізу головних компонентів. Розіб'ємо тренувальні дані на навчання та перевірку в відношенні 9 до 1 та перемішаємо їх.

Модель Sequential є лінійним стеком шарів. Створимо модель Sequential, передавши до конструктора список екземплярів шару. Для перших двох шарів використовуємо LSTM – рекурентну нейронну мережу "Довга короткострокова пам'ять".

У цій моделі ми укладаємо два шари LSTM один на одного, що робить модель здатною до навчання темпоральних уявлень вищого рівня.

Перша LSTM повертає свою повну вихідну послідовність, але друга повертає тільки останній крок у своїй вихідній послідовності, тим самим відкидаючи тимчасовий вимір, тобто перетворюючи вхідну послідовність в один вектор.

Модель має знати, яку вхідну форму вона повинна чекати. Тому перший шар у послідовній моделі, і тільки перший, тому що наступні шари можуть робити автоматичний висновок форми, повинен отримувати інформацію про свою вхідну форму - `input_shape`. Вкажемо фіксований розмір пакета для наших вхідних даних, 20 зразків. Зразки партії обробляються незалежно, паралельно. Під час навчання пакет призводить лише до одного оновлення моделі.

Пакет зазвичай апроксимує розподіл вхідних даних краще ніж один вхід. Чим більший пакет, тим краща апроксимація; однак вірно і те, що обробка пакета займе більше часу і все одно призведе лише до одного оновлення. Для виведення рекомендується вибрати розмір пакету, який настільки великий, наскільки можна собі дозволити, не виходячи з пам'яті.

Передаємо шару `batch_size=20` та `input_shape = (10, 1)`, і шар буде очікувати, що кожен пакет вхідних даних буде мати форму пакета (20, 10, 1) Вихідний шар складатиметься з 1 нейрона з функцією активації 'relu'. Функція активації визначає вихідне значення нейрона. Вихідний нейрон залишимо без нелінійності, щоб мати можливість прогнозувати будь-яке значення.

Далі для всіх 5 ознак по черзі і нормалізованих PCA передбачимо ціни за допомогою навченої моделі і зробимо зворотний скалінг.

Запишемо дані у файли, для подальшого порівняння та побудови графіків в іншій частині програми.

Для MLP всі дії аналогічні крім самої настройки стека шарів. Модель Sequential є лінійним стеком шарів. Створимо модель Sequential, передавши до конструктора список екземплярів шару. Для перших двох шарів застосуємо MLP.

Цілковито пов'язані шари визначаються за допомогою класу `Dense`. Ми можемо вказати кількість нейронів або вузлів у шарі як перший аргумент, а також вказати функцію активації, використовуючи аргумент активації. Ми будемо використовувати функцію активації `ReLU`. Вихідний шар також складатиметься з одного нейрона з функцією активації `ReLU`.

Функція активації визначає вихідне значення нейрона. Вихідний нейрон також залишимо без нелінійності, щоб мати можливість прогнозувати будь-яке значення. Раніше вважалося, що `Sigmoid` та `Tanh` активаційні функції були кращими для всіх шарів. У наші дні більш висока продуктивність досягається за допомогою функції активації `ReLU`.

Далі проведемо спільні дії для двох алгоритмів. Перед навчанням моделі необхідно налаштувати процес навчання, що робиться за допомогою методу `compile`. Він отримує два аргументи: оптимізатор та функція втрат.

В якості оптимізатора використовуємо `Adam`, алгоритм градієнтної оптимізації стохастичних цільових функцій першого порядку, що ґрунтується на адаптивних оцінках моментів нижчого порядку. Цей метод простий у реалізації, обчислювально ефективний, має невеликі вимоги до пам'яті, інваріантний до діагонального масштабування градієнтів і добре підходить для завдань, які є великими з точки зору даних та параметрів.

Цей метод також підходить для нестационарних завдань та завдань з дуже галасливими та/або розрідженими градієнтами. Емпіричні результати показують, що `Adam` добре працює на практиці та вигідно відрізняється від інших методів стохастичної оптимізації.



Функція втрат – це та мета, яку модель намагатиметься звести до мінімуму. Використовуємо середньоквадратичну помилку. Фізичного сенсу MSE не має, але чим ближче до нуля, тим модель краще.

А ось MLP-мережа при 100 епохах показує набагато якісніше результати, ніж за значення дорівнює 50. Помилка в даному випадку стає мінімальною і надалі вже практично не знижується.

Налаштування алгоритмів машинного навчання, а саме додаткове налаштування `RandomForestRegressor`. Отже, скористаємося оптимізацією гіперпараметрів моделі для того, щоб покращити якість передбачень, що видаються RF-моделлю.

Гіперпараметри можна розглядати як "налаштування" моделі. Але налаштування, які відмінно підходять для одного набору даних, для іншого не актуальні - тому варто займатися їх оптимізацією.

Застосуємо алгоритм `RandomizedSearchCV`, що дозволяє досліджувати широкі діапазони значень. У ході роботи генеруємо словник `random_grid`, що містить для кожного гіперпараметра діапазон значень, які потрібно випробувати. Далі ми ініціалізуємо об'єкт `random_search` за допомогою функції `RandomizedSearchCV`, передаючи їй RF модель, `param_dist`, число ітерацій та кількість крос-валідацій, які потрібно виконати.

Гіперпараметр `verbose` дозволяє керувати обсягом інформації, що виводиться моделлю під час її роботи. Гіперпараметр `n_jobs` дозволяє вказувати, скільки процесорних ядер потрібно використовуватиме забезпечення роботи моделі. Встановлення `n_jobs` у значення `-1` призведе до швидшої роботи моделі, оскільки при цьому будуть використовуватися всі ядра процесора.

Займемося підбором наступних гіперпараметрів: `n_estimators` – число «дерев» у «випадковому лісі». `max_features` – число ознак для вибору розщеплення. `max_depth` – максимальна глибина дерев. `min_samples_split` - мінімальна кількість об'єктів, необхідне для того, щоб вузол дерева міг би розщепитися. `min_samples_leaf` - мінімальна кількість об'єктів у листі.

### 3 РЕЗУЛЬТАТИ РОБОТИ МЕТОДУ АВТОМАТИЗОВАНОГО АНАЛІЗУ ДИНАМІКИ РУХУ ЦІН НА БІРЖАХ ВІРТУАЛЬНИХ АКТИВІВ

#### 3.1 Результати перевірки технічних індикаторів

Середня продуктивність усіх використовуваних індикаторів технічного аналізу, з усіма протестованими налаштуваннями, склала -44 898,71. Медіана продуктивності становила -61 754,83.

Таблиця 3.1

Продуктивність використання технічних індикаторів

середнє значення	-44898,71
медіана	-61754,83
максимум	99861,71
мінімум	-97292,69

Як видно з таблиці 3.1, загальна продуктивність була поганою для обраного активу, обраних методів технічного аналізу та обраної шкали часу. Медіана продуктивності всіх методів і налаштувань технічного аналізу склала -61%.

У випадку реальної торгівлі – це величезна втрата, оскільки часовий графік, коли здійснювалися угоди, становив лише близько 100 днів.

Варто зазначити, що найкраща продуктивність склала +99%, що є чудовим результатом.

Ці результати можна розглядати як приклад того, наскільки нестійкими є віртуальні активи, але їх також можна розглядати як приклад поганої роботи технічного аналізу. Середнє значення та медіана продуктивності показують, що всі обрані технічні індикатори не працюють належним чином, для постійного ефективного аналізу.

Таблиця 3.2

## Загальні результати продуктивності використання кожного індикатора

Назва індикатору	Тип значення	Значення
BB	середнє	-45029,74
	медіана	-61800,73
	максимум	99761,71
	мінімум	-94421,30
EMA	середнє	-43778,84
	медіана	-48420,48
	максимум	72250,53
	мінімум	-97392,69
MACD	середнє	-50720,78
	медіана	-53633,82
	максимум	14637,00
	мінімум	-86455,09
RSI	середнє	-28748,77
	медіана	-41569,70
	максимум	43398,95
	мінімум	-90205,99
SMA	середнє число	-43674,79
	медіана	-44557,39
	максимум	37450,05
	мінімум	-91519,17

Як видно з таблиці 3.2, усі обрані методи були здебільшого збитковими. Медіана втрати склала 49%. Через комісію за торгівлю цей результат показує, що використання технічного аналізу та сплата комісії не є прибутковими на поточному ринку. Ці результати можуть відрізнятися в інших ринкових ситуаціях, але для визначення того, як ці методи технічного аналізу працюватимуть у різних ринкових

ситуаціях, потрібне інше дослідження. Медіана ефективності всіх стратегій була збитковою. Згідно з результатами, володіння деякими знаннями з технічного аналізу є важливим. Без знань швидше втратити гроші, ніж отримати прибуток. Навіть найкращий результат, RSI, був збитковим. Смуги Боллінджера дають найкращий максимальний прибуток, але середній результат склав -62%. Це означає, що смуги Боллінджера не працюють для кожної валюти, ситуації чи налаштування. Максимальний результат може бути хорошим, якщо людина, яка використовує технічний аналіз, знає, як працює технічний аналіз, і знає, які найкращі налаштування. Не знаючи, чому максимальна продуктивність така хороша, як вона є, марно знати найкращу продуктивність. Краще бачити середню ефективність, ніж максимальну або середню продуктивність без знання. Якщо використовується лише одна стратегія з одним параметром, половина випадків є кращою за середню продуктивність, а половина випадків не така хороша, як середня продуктивність. Середнє значення не показує, який прибуток чи збиток очікується, якщо використовується лише одне з налаштувань.

Таблиця 3.3

## Ефективність смуг Боллінджера за різних параметрів

Період	Стандартне відхилення	Тип	Середнє число	Медіана	Максимум	Мінімум
20	2	0	-46246,78	-61974,44	84389,18	-95421,30
20	2	1	-46215,61	-61798,45	86624,49	-94521,30
20	2	2	-45781,72	-61779,74	87353,85	-94521,30
20	2,5	0	-44668,71	-61329,67	87347,59	-93521,30
20	2,5	1	-45486,34	-61594,08	88220,30	-94521,30
20	2,5	2	-45120,74	-61825,90	88860,56	-94521,30
20	3	0	-43794,31	-62126,17	92375,29	-94521,30
20	3	1	-44312,57	-61942,55	89510,69	-93521,30
20	3	2	-44250,40	-61547,09	90618,98	-97521,30
21	2	0	-46068,44	-61783,34	85051,69	-96520,58

Продовження таблиці 3.3

Період	Стандартне відхилення	Тип	Середнє число	Медіана	Максимум	Мінімум
21	2	1	-46125,20	-61648,22	86826,44	-93520,58
21	2	2	-45928,03	-61910,48	86944,60	-94520,58
21	2,5	0	-44849,63	-61747,64	87916,93	-93520,58
21	2,5	1	-45171,87	-62868,69	88784,65	-94520,58
21	2,5	2	-45077,91	-61631,32	88663,85	-94520,58
21	3	0	-44759,78	-62303,37	96763,97	-95420,58
21	3	1	-44202,78	-62107,30	86983,30	-95420,58
21	3	2	-44738,40	-61936,87	91300,00	-95420,58
22	2	0	-46109,39	-61747,30	85574,43	-95415,30
22	2	1	-46784,95	-61621,61	86538,59	-95415,30
22	2	2	-45520,81	-61675,17	87640,89	-95715,30
22	2,5	0	-44695,00	-61787,44	88411,22	-95615,30
22	2,5	1	-45045,67	-61839,30	88641,06	-95315,30
22	2,5	2	-44981,96	-61827,80	88781,06	-95315,30
22	3	0	-43447,87	-62142,57	99761,71	-95715,30
22	3	1	-44150,50	-62012,20	88241,40	-95315,30
22	3	2	-43927,63	-61948,48	91828,48	-95215,30

Максимальна продуктивність смуг Боллінджера була найкращою в усіх налаштуваннях. Однак результати середньої результативності не були хорошими, як максимальні. Більш детальне дослідження показує, що смуги Боллінджера принесли прибуток лише на деяких часових проміжках, інші були збитковими. Причина, чому тільки ці три валюти були прибутковими, невідома. Ретельніше дослідження прибуткових валют не виявило вагомих причин, чому смуги Боллінджера здійснювали з ними прибуткові операції. Далі в цьому дослідженні ці валюти аналізуються більш точно. Загальною рисою цих валют було те, що їх

вартість то зростала, то знижувалася на досліджуваній шкалі часу. Значення більшості валют були набагато вищими на початку шкали часу, ніж у кінці.

Цікаво, що діапазон Боллінджера отримав найкращий прибуток серед усіх перевірених методів технічного аналізу на цих трьох валютах. У налаштуваннях діапазонів Боллінджера тип ковзного середнього має три параметри: 0 для простого ковзного середнього, 1 для експоненціального ковзного середнього і 2 для зваженого ковзного середнього. Існують невеликі відмінності між типами ковзних середніх, але відмінності не є суттєвими.

Таблиця 3.4

## Показники ЕМА з різними періодами

Період	Середнє число	Медіана	Максимум	Мінімум
5	-42346,57	-47597,84	72340,53	-95657,66
10	-41578,43	-44598,41	53756,11	-94043,05
20	-43510,12	-49639,36	59157,08	-94457,17
50	-44534,52	-56744,43	30358,12	-96559,21
100	-47554,45	-49359,79	67450,54	-97152,69
200	-46508,95	-41506,20	66653,00	-95696,14

ЕМА була прибутковою з XMR, DRC і частково на LTC. Що стосується налаштувань ЕМА, не було суттєвих відмінностей у різних налаштуваннях. У середньому період 50 спрацював набагато гірше, ніж інші, але середній прибуток був таким же поганим, як і в будь-якому іншому випадку. Найгірші показники були відносно однаковими в усіх налаштуваннях.

Виходячи з цього дослідження, різниці між кількістю періодів немає.

П'ять-двадцять періодів, здається, працюють трохи краще, ніж сто чи двісті, але різниця незначна.

У цьому дослідженні MACD запускався, вказуючи три числа: швидкий період, повільний період і період сигналу. Швидкий період – це кількість періодів, які використовуються для розрахунку швидкої ЕМА. Для більш чутливих ліній MACD швидкі періоди повинні бути короткими. Повільний період розраховує

довгострокові тенденції цін. У цьому дослідженні було б краще досліджувати лише короткострокові зміни цін і сигнали. Це ще більше розширило б масштаб дослідження, і в цьому масштабі обидва досліджувалися одночасно. Це погано для продуктивності, але загальна продуктивність не сильно відрізнялася від інших методів технічного аналізу.

Таблиця 3.5

## Продуктивність MACD при роботі на різних налаштуваннях

Швидкий період	Повільний період	Сигнал	Середнє значення	Медіана	Максимум	Мінімум
5	35	5	-53521,18	-54164,84	-8310,45	-84451,58
6	13	5	-48456,90	-49119,32	-13529,13	-86520,32
7	14	7	-53565,93	-55747,25	-17687,60	-81581,79
8	17	9	-55356,96	-53558,11	-15808,13	-86595,32
10	20	5	-53743,55	-54185,65	-11447,54	-86545,09
10	30	5	-53154,96	-53613,74	-5174,02	-83525,42
12	26	9	-53452,29	-52752,59	-22722,23	-87436,14
15	30	5	-51471,75	-54503,39	-15189,56	-81402,58
18	35	6	-41652,83	-51964,07	14237,00	-82780,32
21	55	8	-39351,48	-46750,87	5127,14	-84477,51

На MACD є значна різниця в продуктивності за різних налаштувань: середній прибуток становить від -40% до -56%, а середній прибуток становить від -43% до -60%. MACD був прибутковим лише на двох налаштуваннях: 21,55,8 і 18,35,6. Логіки в результатах немає. Неможливо сказати, яке налаштування найкраще. Відповідно до цього результату, випадковість впливає на результат більше, ніж саме налаштування. Не має значення значення швидкого, повільного або сигнального налаштування. Це дослідження не показує, яке налаштування MACD є найкращим.

Таблиця 3.6

## Продуктивність RSI в різні періоди

Період	Середнє число	Медіана	Максимум	Мінімум
7	-39248,99	-44452,93	27727,69	-91605,99
14	-34563,09	-37650,86	32614,22	-84743,27
21	-14613,23	-17454,58	44458,95	-76478,38

З усіх досліджуваних методів RSI показав найкращі результати. Це був найвигідніший спосіб. Однак між різною кількістю періодів була значна різниця. RSI розраховувався з використанням 7, 14 і 21 періодів. Двадцять один період був найкращим лише -18%, а сім - найгіршим -39%. Для наступних досліджень слід провести дослідження того, як працює RSI, якщо кількість періодів ще більша. Це дослідження розраховало RSI лише з використанням трьох різних періодів. Виходячи з цього, неможливо сказати, чи продуктивність краща з більшою кількістю періодів, чи вона випадкова в порядку кількості періодів. У цьому дослідженні RSI мав значення 30 для нижньої межі та 70 для верхньої межі. При більш ретельному розгляді ефективності RSI на окремих валютах здається, що це просто збіг, що в цій таблиці показники такі, що сім є найгіршим, а 21 – найкращим. Це тому, що існують різні типи замовлень на одну валюту.

Таблиця 3.7

## Продуктивність SMA за різних налаштувань

Мінімальний період SMA	Максимальний період SMA	Середнє число	Медіана	Максимум	Мінімум
10	20	-44570,72	-54522,03	16452,61	-90132,88
10	50	-40584,99	-44839,84	14839,35	-87747,29
20	50	-38154,35	-43128,74	31853,01	-83544,44
50	100	-34529,82	-37452,14	37850,05	-88454,41
50	200	-52584,08	-36492,14	21759,90	-97519,17



Середня продуктивність SMA становить від -36% до -52%, а середня продуктивність становить від -38% до -61%. Максимальна сума прибутку незначна, але збиток у гіршому випадку становить 92%. Згідно з цим дослідженням, не можна сказати, що ефективність SMA буде передбачуваною. Різниця між найгіршою та найкращою продуктивністю досить значна.

У попередньому етапі дослідження було показано, що технічний аналіз не можна використовувати для прибуткової торгівлі у кожній ситуації, тепер технічний аналіз порівнюється з випадковими інвестиційними рішеннями.

Таблиця 3.9

## Методи технічного аналізу у порівнянні з випадковістю

Назва індикатору	Тип значення	Значення	Значення у порівнянні з випадковістю
BB	середнє	-45029,74	8178,72
	медіана	-61800,73	7929,47
	максимум	99761,71	99722,77
	мінімум	-94421,30	-2510,93
EMA	середнє	-43778,84	9329,62
	медіана	-48420,48	21399,72
	максимум	72250,53	72411,59
	мінімум	-97392,69	-4172,32
MACD	середнє	-50720,78	2417,68
	медіана	-53633,82	15186,38
	максимум	14637,00	14208,06
	мінімум	-86455,09	5465,18
RSI	середнє	-28748,77	23779,59
	медіана	-41569,70	29240,50
	максимум	43398,95	43170,01
	мінімум	-90205,99	2714,38

Продовження таблиці 3.9

Назва індикатору	Тип значення	Значення	Значення у порівнянні з випадковістю
SMA	середнє	-43674,79	10353,67
	медіана	-44557,39	25662,81
	максимум	37450,05	35631,11
	мінімум	-91519,17	402,20

У таблиці 3.9 можна побачити, що кожен метод технічного аналізу кращий за випадковість. Тільки мінімальні результати іноді гірші, при використанні деяких методів технічного аналізу.

Таблиця 3.10

Смуги Боллінджера на різних налаштуваннях порівняно з випадковістю

Період	Стандартне відхилення	Тип	Середнє число	Медіана	Максимум	Мінімум
20	2	0	7011,68	7825,76	84160,24	-2500,72
20	2	1	6922,85	7811,75	86055,55	-2500,72
20	2	2	7236,74	7930,46	87334,91	-2500,72
20	2,5	0	8349,75	7880,53	87148,65	-2500,72
20	2,5	1	7932,12	7916,12	87081,36	-2500,72
20	2,5	2	8087,72	7974,30	83541,62	-2500,72
20	3	0	9224,15	7354,03	92146,35	-2500,72
20	3	1	8925,89	7757,65	83521,75	-2500,72
20	3	2	8958,06	7863,11	90370,04	-2500,72
21	2	0	7140,02	7426,86	84823,75	-2500,11
21	2	1	7013,26	8351,98	86357,50	-2500,11
21	2	2	7320,43	7459,72	84515,66	-2500,11
21	2,5	0	8428,83	8012,56	87667,99	-2500,11
21	2,5	1	8046,59	7351,51	88665,71	-2500,11

## Продовження таблиці 3.10

Період	Стандартне відхилення	Тип	Середнє число	Медіана	Максимум	Мінімум
21	2,5	2	8280,55	7658,88	88354,91	-2500,11
21	3	0	9568,68	7756,83	96455,03	-2500,11
21	3	1	8385,68	7752,90	86854,36	-2500,11
21	3	2	9700,06	7363,33	90871,06	-2500,11
22	2	0	7319,07	8452,90	85365,49	-2493,97
22	2	1	7053,51	8768,59	83519,65	-2493,97
22	2	2	7307,65	8355,03	87521,95	-2493,97
22	2,5	0	8533,46	8752,76	88342,28	-2493,97
22	2,5	1	8262,79	7350,90	87542,12	-2493,97
22	2,5	2	8226,50	8352,40	86262,12	-2493,97
22	3	0	9840,59	7757,63	99452,77	-2493,97
22	3	1	9057,96	7658,00	87312,46	-2493,97
22	3	2	9280,83	7751,72	92589,54	-2493,97

Смути Боллінджера працюють краще, ніж випадковість. Відмінності були майже однакові з кожним налаштуванням. Середня продуктивність була на 7-9 процентів кращою, а середня продуктивність була на вісім процентів кращою, ніж випадковість.

Таблиця 3.11

## Параметри ЕМА порівняно з випадковістю

Період	Середнє число	Медіана	Максимум	Мінімум
5	10681,89	22342,36	71330,53	-2527,29
20	11238,34	23350,84	71330,53	-1376,80
50	7383,94	13575,77	71330,53	-2548,84
100	6984,01	20961,41	71330,53	-4182,32
200	7259,51	19874,00	71330,53	-3455,77

На ЕМА було більше розповсюджених результатів. Середні результати були кращими на 7–12 відсотків із технічним аналізом, а середні результати були кращими на 13–23 відсотків.

Таблиця 3.12

## Параметри MACD порівняно з випадковістю

Швидкий період	Повільний період	Сигнал	Середнє число	Медіана	Максимум	Мінімум
5	35	5	1617,28	14705,36	-8359,39	8428,79
6	13	5	4301,56	22600,88	-10148,07	13100,05
7	14	7	-827,47	12432,95	-19726,54	7028,58
8	17	9	-2278,50	11162,09	-14457,07	5515,05
10	20	5	1425,09	10433,55	-11246,48	5355,28
10	30	5	133,50	13506,46	-5332,96	8454,95
12	26	9	-12,83	14867,61	-21551,17	12584,23
15	30	5	1706,71	15626,81	-15358,50	10537,79
18	35	6	7265,63	18426,13	15378,06	11190,05
21	55	8	13345,98	25769,33	5094,20	8842,86

MACD відрізняється від попередніх методів тим, що в деяких випадках середня ефективність була гіршою, ніж продуктивність випадковості. Можливо, причиною цього є ситуація на ринку, але незрозуміло, в чому причина. Проблема цього дослідження полягає в тому, що досліджується лише одна ринкова ситуація, і ситуація може відрізнитися в залежності від іншої ситуації на ринку.

Таблиця 3.13

## RSI з різними параметрами порівняно з випадковістю

Період	Середнє число	Медіана	Максимум	Мінімум
7	14168,47	24027,27	28716,68	2824,38
14	21355,37	31468,34	28716,68	6467,10
21	35405,23	52465,62	28716,68	13441,99

RSI був збитковим у цій ринковій ситуації, але показав набагато краще, ніж випадковість. Середній показник був кращим на 25–52 відсотків.

Таблиця 3.14

## SMA порівняно з випадковістю

		Середнє число	Медіана	Максимум	Мінімум
10	20	6147,74	17648,17	16243,67	2827,49
10	50	11813,47	24250,36	13620,41	5313,08
20	50	13864,11	26351,46	30744,07	7065,93
50	100	17478,64	32558,06	37721,11	4435,96
50	200	524,37	8483,33	21370,96	412,20

SMA мала велику середню різницю у випадковості, але середня продуктивність із налаштуванням 50 200 була лише трохи кращою, ніж випадковість. Середня продуктивність важлива, якщо трейдер торгує кількома валютами одночасно. Середня продуктивність набагато важливіша, якщо трейдер торгує лише однією валютою за раз.

Порівняно з випадковими покупками та продажами кожен метод технічного аналізу показав кращі результати. Лише пара налаштувань виявилася гіршою. Виходячи з ситуації на ринку на момент проведення цього дослідження, RSI є найкращим інструментом для допомоги в інвестуванні. Це було набагато краще, ніж випадковість. ЕМА та SMA також показали хороші результати, але дисперсія на SMA була набагато вищою, ніж на ЕМА. Для SMA необхідно ретельно вибирати найкращі налаштування. У найгірших налаштуваннях SMA лише трохи краще, ніж випадковість.

### 3.2 Результати роботи алгоритмів машинного навчання

Навчимо 3 різні стандартні моделі алгоритмів машинного навчання та оцінимо точність для кожної за середньоквадратичною помилкою та коефіцієнтом

детермінації. За допомогою лінійної регресії проведемо прогноз застосувавши регуляризацію L1. Як бачимо з рис.3.1, результат досить поганий, адже значення MSE більше 2.8, а коефіцієнт детермінації показує значення нижче середнього.

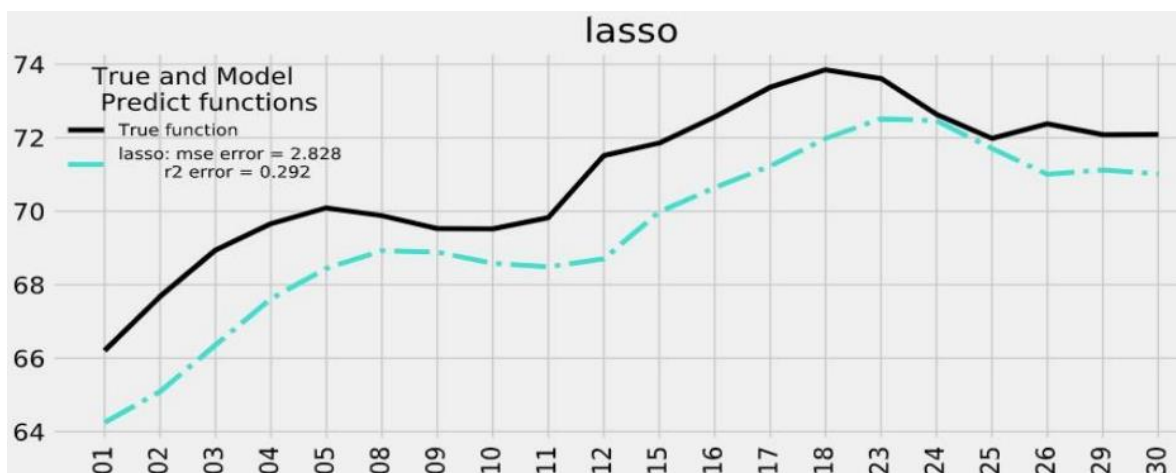


Рис.3.1 Застосування регуляризації L1 (lasso) на періоді передбачення

На рис. 3.2 зображена вже робота лінійної регресії із застосуванням регуляризації L2. Варто зазначити, що результат доволі непоганий, порівняно з регуляризацією L1, адже значення MSE більше 2.8, а коефіцієнт детермінації показує значення вище середнього.

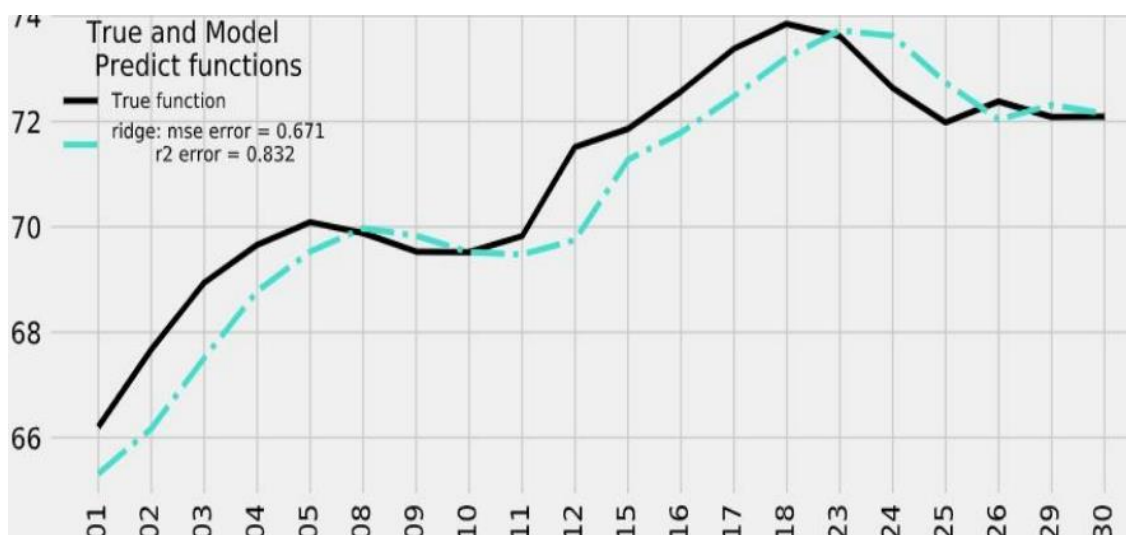


Рис.3.2 Застосування регуляризації L2 на періоді передбачення

На рис.3.3 зображено вже роботу лінійної регресії із застосуванням алгоритму підсумовування регуляризації L1 і L2, під назвою ElasticNet:

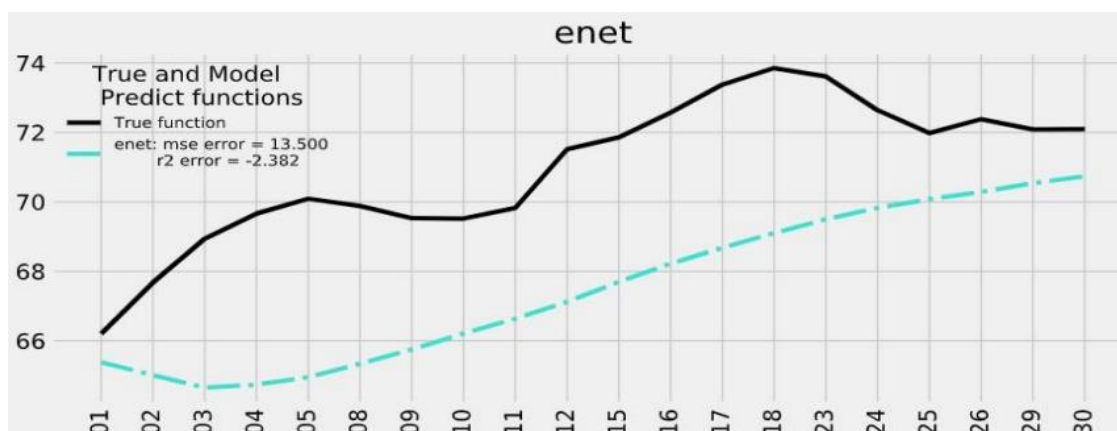


Рис.3.3 Застосування регуляризації ElasticNet на періоді передбачення

ElasticNet зводить нанівець всю роботу регуляризації L2, так як об'єднання L1 та L2 тільки заплутуватимуть модель.

Як бачимо за графіками роботи алгоритму лінійної регресії із застосуванням різних регуляризацій разом, результат прогнозу лише погіршуватиметься, оскільки дані які ми використовуємо для навчання – вже оптимізовані, а тому немає того шуму даних із якого регуляризацію варто було б застосувати. Модель не страждає від перенавчання.

Тепер можемо протестувати роботу алгоритму KNN. Результат відображено на рис.3.4.

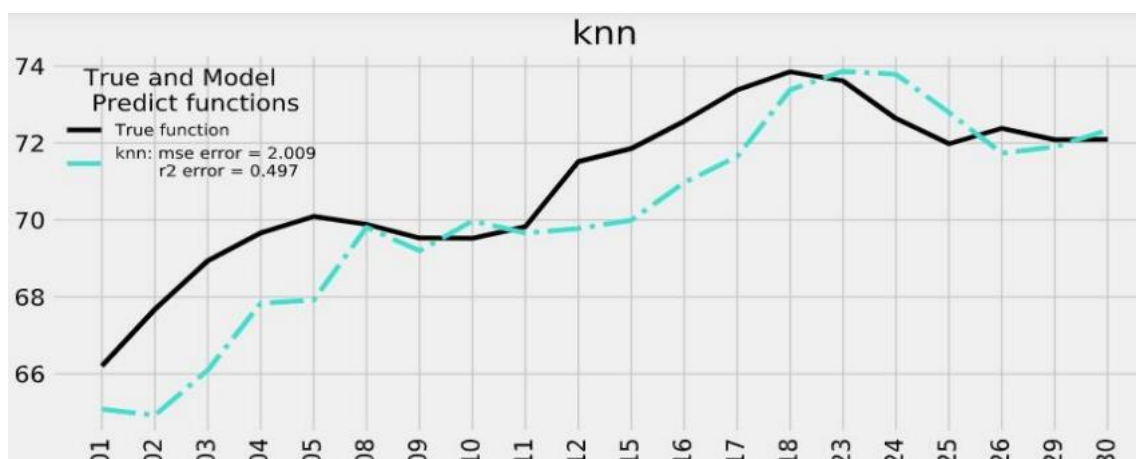


Рис.3.4. Графік роботи алгоритму KNN на періоді передбачення

Як бачимо алгоритм KNN не підходить для завдання із запропонованим набором даних. Виявлено великий відсоток помилки, передбачення вийшло дуже приблизним.

Тепер можемо вивести порівняльний графік результатів роботи всіх алгоритмів машинного навчання, де як вхідні дані застосовувалися лише «Ціна закриття», «Ціна відкриття», «Максимальна ціна», «Мінімальна ціна».



Рис.3.5 Графік роботи всіх алгоритмів з використанням вхідних даних без ознаки тональності

На рис. 3.5 бачимо, що серед обраних алгоритмів нейронних мереж, які більшою мірою підходили під дане завдання, результати є кращими ніж у індикаторів технічного аналізу, але досить поганими для реальної інвестиційної діяльності.

Перевіримо якість роботи Лінійної регресії із застосуванням регуляризатора Ridge та без нього, після навчання на ознаках OHLC та volume, які в теорії мають впливати на кінцеву вартість активів на біржі.



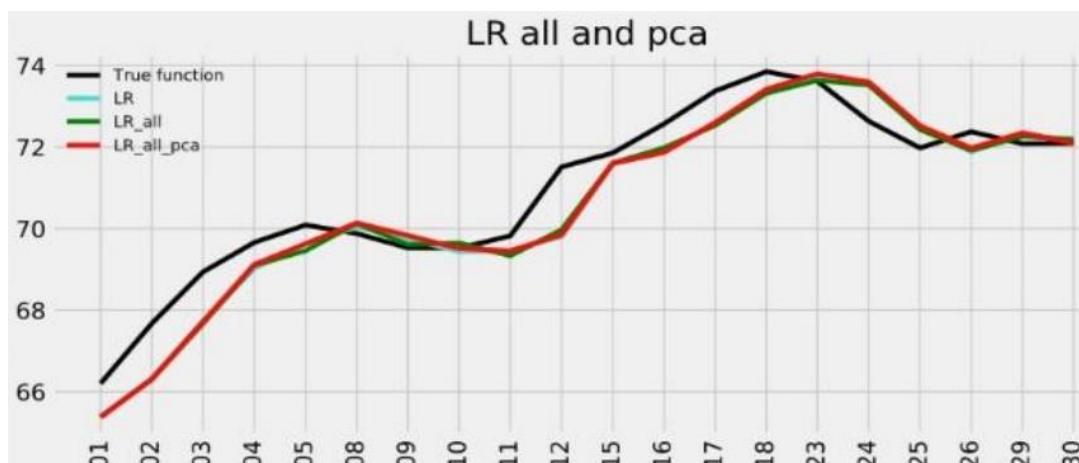


Рис.3.6 Результат роботи моделі LR після застосування PCA

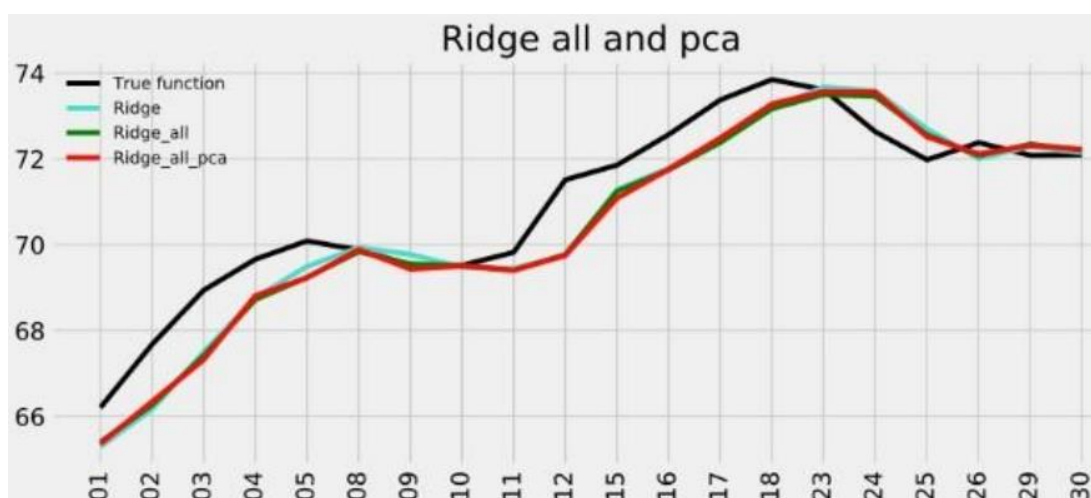


Рис.3.7 Результат роботи моделі Ridge після застосування PCA

З результатів відображених на рис.3.6 та рис.3.7, ми бачимо, що нормалізація введених ознак методом PCA в модель, не принесла бажаного результату. Модель лінійної регресії лише погіршила свій результат, і навіть застосування регуляризації його не покращило, отже, можемо зробити висновок про те, що регуляризацію даних наборів даних використовувати не варто, у нашому випадку ознаки, додані до моделі, будуть лише завдавати шкоди якості прогнозу.

Як додаткове дослідження можливостей нейронних мереж і якості їх прогнозу додаємо в нейромережу не всі з раніше введених даних, виконаємо перетворення та побудуємо графіки.

На рис.3.8 зображено графіки для MLP мережі, де:

- True function: Справжня крива графіка цін на актив;

- MLP: Крива на графіку лише з ознакою «ціна закриття»;
- MLP\_all: Крива, що відображає роботу нейромережі з усіма ознаками;
- MLP\_1: Крива, що відображає роботу нейромережі без ознаки тональності;
- MLP\_2: Без ознаки – High;
- MLP\_3: Без ознаки – Low;
- MLP\_4: Без ознак – Open;
- MLP\_5: Без ознаки – Close.

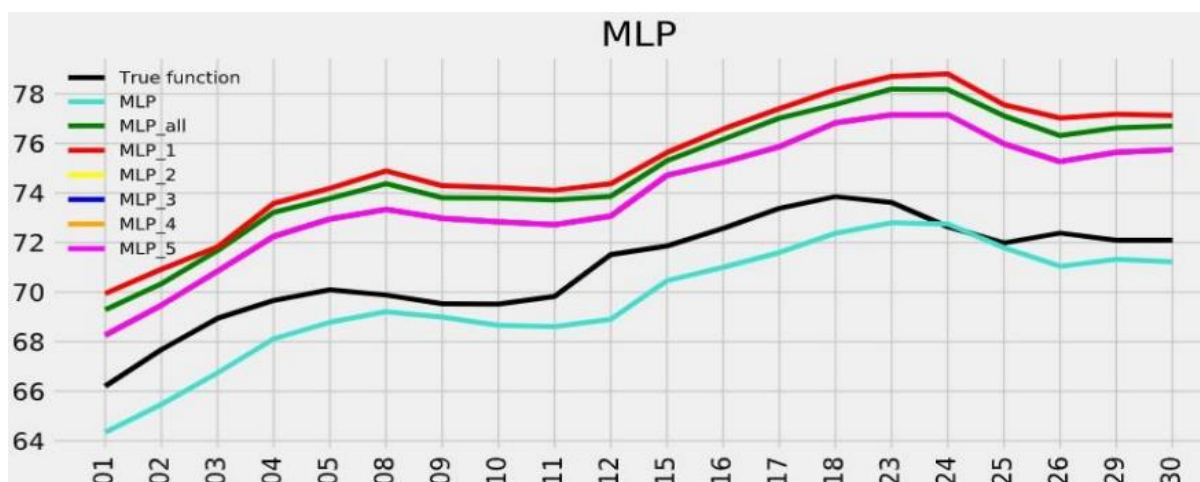


Рис.3.8 Графік роботи MLP з ознаками

Далі на рис.3.9 бачимо графік LSTM мережі, де:

- True function: Справжня крива графіка цін на актив;
- LSTM: Крива на графіку лише з ознакою «ціна закриття»;
- LSTM\_all: Крива, що відображає роботу нейромережі з усіма ознаками;
- LSTM\_2: Без ознак – High;
- LSTM\_3: Без ознаки – Low;
- LSTM\_4: Без ознаки – Open;
- LSTM\_5: Без ознак тональності.

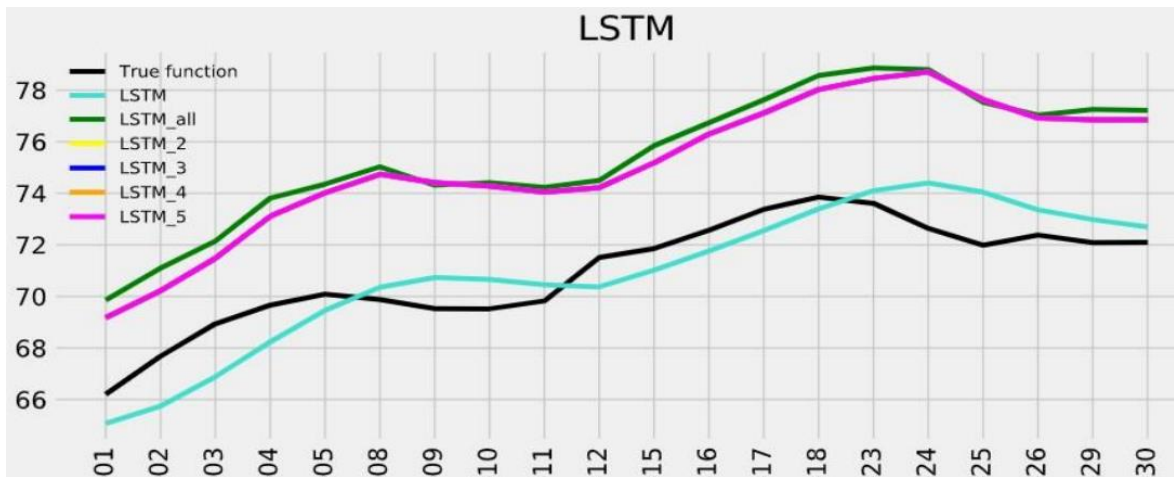


Рис.3.9 Графік роботи LSTM з ознаками

З графіків робіт нейронних мереж випливає, що додавання додаткових ознак, які у теорії мали поліпшити прогнози моделі, у деяких випадках для технічного аналізу не ефективні, наприклад ознаки OHLC лінійно залежні одна від іншої, щодо ознаки тональності, то вона не завжди корелює з ними, але схильний до стрибків у різні часові періоди, іноді практично без зв'язку з самою ціною.

Щоб позбавитися сильної кореляції ознак OHLC між собою, проведемо додаткову нормалізацію даних. Для цього застосуємо метод основних компонентів. Він відмінно підходить для корелюючих даних в ознаках моделі та здатний поліпшити якість прогнозування. За допомогою методу PCA піддамо перетворення ознак OHLC в одну.

На рис.3.10 бачимо графік, що відображає роботу MLP мережі, а саме:

- True function: Справжня крива графіка цін на актив;
- MLP: Крива на графіку лише з ознакою close;
- MLP\_all: Крива з усіма ознаками без методу PCA;
- MLP\_6: Враховуються лише close та ознаки тональності;
- MLP\_pca: Враховуються лише ознаки open та close з PCA;
- MLP\_all\_pca: Ознаки OHLC з PCA.

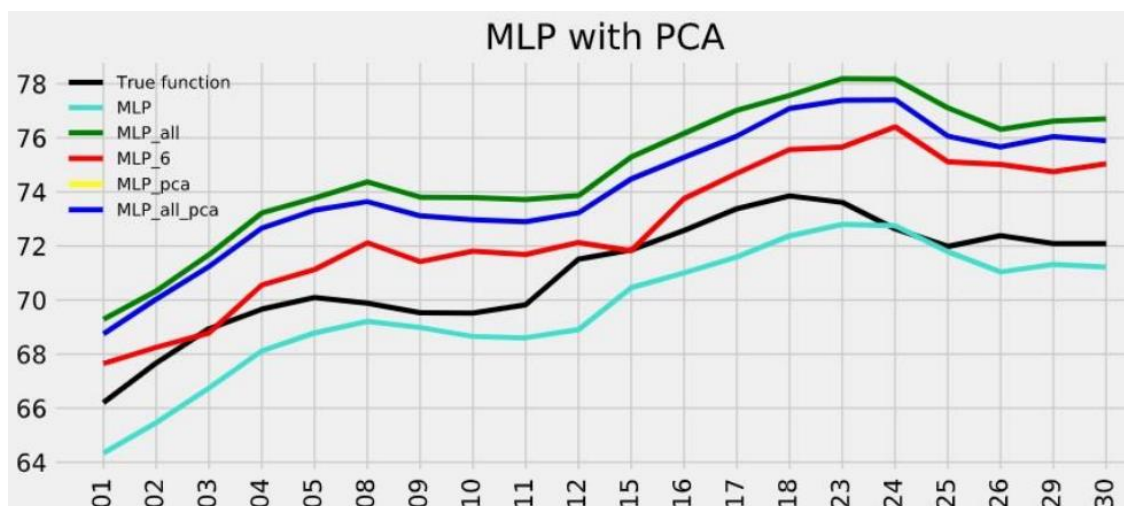


Рис.3.10 Графік роботи MLP з ознаками, з PCA

На рис.3.11 бачимо графік, що відображає роботу LSTM мережі, а саме:

- True function: Справжня крива графіка цін на актив;
- LSTM: Крива на графіку лише з ознакою close та ознакою тональності;
- LSTM\_all: Крива з усіма ознаками без PCA;
- LSTM\_6: Враховуються лише ознака close;
- LSTM\_pca: Враховуються лише ознаки OHLC з PCA;
- LSTM\_all\_pca: Ознаки OHLC з PCA та ознаками тональності.

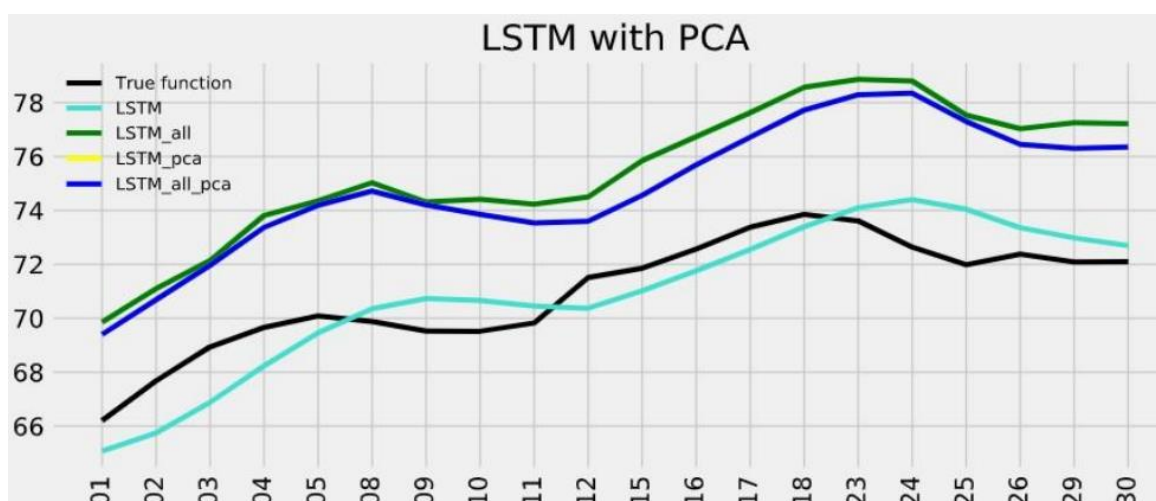


Рис.3.11 Графік роботи LSTM з ознаками, з PCA

Із графіків представлених на рис.3.10 та рис.3.11 можна зробити висновки, що після додаткової нормалізації ознак методом PCA є несуттєве збільшення якості

деяких результату прогнозу, але загалом додавання цих ознак у модель, які у теорії мали поліпшити прогноз, в деяких випадках, погіршило його, навіть після додаткових перетворень. Тому, можна дійти висновку, що важливо аналізувати та використовувати як вхідні дані для машинного навчання, не тільки ознаки технічних індикаторів, а саме, OHLC – відомі усім інвесторам, а й ознаки тональності, оскільки вони суттєво поліпшують результати прогнозування.

Переконаємось у цьому поглянувши на результат роботи LSTM мережі, яка зараз використовує ознаки технічних індикаторів разом з ознаками тональності.

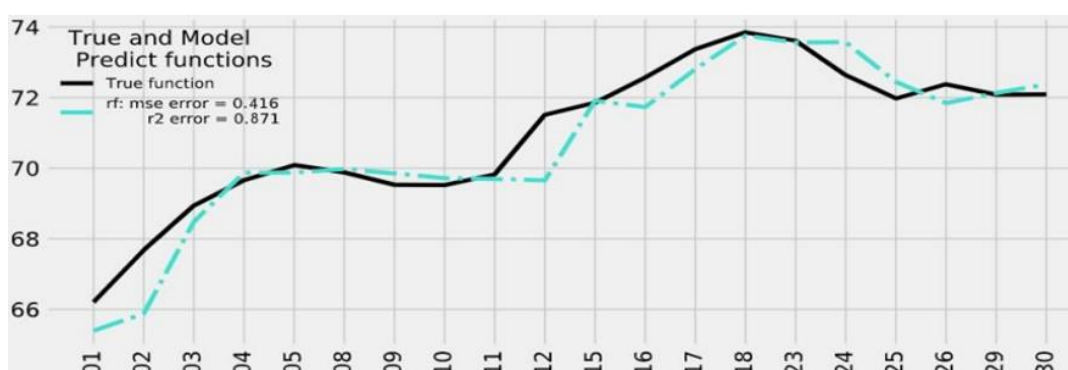


Рис.3.11 Графік роботи LSTM з ознаками тональності

На рис.3.12 наведено графіку роботи LSTM мережі, із застосуванням ознак тональності разом з ознаками, що використовують технічні індикатори. Виходячи з того, що значення MSE нижче середнього, а саме 0,416, а коефіцієнт детермінації показує значення вище середнього, а саме 0.871, то можемо зробити висновок, що використовуючи такі ознаки як OHLC та ознаки тональності, нам вдалося покращити результати прогнозування.

## ВИСНОВКИ

У даній роботі, було розглянуто різні методи аналізу цін на ринку віртуальних активів, які використовуються реальними інвесторами.

Технічний підхід ґрунтується на аналізі та спостереженні минулих показників. Основна концепція полягає в тому, що ціна вже відображає всі необхідні фактори, тож аналіз змін ціни на певних проміжках часу має відповіді на її подальший рух. Цей підхід популярний завдяки своїй простоті у використанні.

Фундаментальний підхід ґрунтується на аналізі факторів, що впливають на формування ціни віртуальних активів, наприклад, стан світової економіки. Оскільки це охоплює значну кількість даних, то для звичайних інвесторів такий підхід може здаватись доволі складним, тому цей підхід, в більшості, має популярність серед великих компаній та інвестиційних фондів.

Сентиментальний, або емоційний, підхід ґрунтується на дослідженні активності користувачів соціальних мереж та відомих особистостей. Хоча цей підхід може здатися ненадійним, аналіз трендів на основі поведінки вказує на непогані результати, оскільки думка впливових та зацікавлених осіб має значний вплив на масову аудиторію.

В результаті роботи було визначено, що індикатори технічного аналізу, такі як: SMA, EMA, MACD, RSI та смуги Боллінджера довели свою неефективність, на більшості часових проміжках. Варто зауважити, що проведення фінансового аналізу за допомогою технічних індикаторів як вхідних даних для навчання моделей машинного навчання є більш ефективним, ніж використання самих індикаторів.

З метою вибору алгоритму для автоматизованого аналізу цін та покращення якості прогнозування, було проведено огляд таких алгоритмів: ML, LR, LSTM, RF. Оптимальні параметри для кожної моделі були знайдені шляхом виконання налаштування параметрів за допомогою набору даних перевірки. Потім моделі навчили на технічних індикаторах, розрахованих на основі історичних даних про

ціну Litecoin в доларах США. У процесі роботи, навчені моделі перевіряли та оцінювали за допомогою різних показників оцінювання, щоб отримати уявлення про продуктивність кожного алгоритму.

Після порівняння метричних оцінок кожної моделі, було виявлено, що LSTM з оцінкою MSE (0,416) та оцінкою  $R^2$  (0.871) показала найкращі результати у прогнозуванні цін на віртуальні активи, у порівнянні з іншими досліджуваними моделями. Варто зазначити, що такого результату вдалось досягти, лише після додавання показників тональності як вхідних даних для навчання LSTM моделі, разом із даними технічного індикатору.

Результати, отримані під час виконання цієї роботи, стосуються лише набору даних і технічних індикаторів, які використовуються для навчання моделей. Тому, у майбутніх дослідженнях можна змінити набір даних, наприклад, взявши більший набір даних, враховуючи інші технічні індикатори, або тестуючи на іншому наборі даних та враховуючи глобальні економічні та соціальні фактори.



## ПЕРЕЛІК ПОСИЛАНЬ

1. Dasgupta D. “Designing Application-Specific Neural Networks using the Structured Genetic Algorithm”. – 1992.
2. Nakamoto, S. “Bitcoin: A peer-to-peer electronic cash system”. – 2008.
3. Akyildirim, E., Goncu, A., & Sensoy, A. “Prediction of Cryptocurrency Returns Using Machine Learning”. – 2018 [Електронний ресурс]. Режим доступу: <https://www.researchgate.net/publication/329322600>.
4. Albuquerque, Y., de Sá, J., Padula, A., & Montenegro, M. “The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with support vector regression” – 2019 [Електронний ресурс]. Режим доступу: <https://doi.org/10.1016/j.eswa.2017.12.004>.
5. Alessandretti, L., ElBahrawy, A., Aiello, L. “Anticipating cryptocurrency prices using machine learning. Hindawi Complexity”. – 2018. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1155/2018/8983590>.
6. Amjad, M., & Shah, D. “Trading Bitcoin and online time series prediction. NIPS 2016 Time Series Workshop”. – 2016 [Електронний ресурс]. Режим доступу: <http://proceedings.mlr.press/v55/amjad16.pdf>.
7. Belinskyi, A., Soloviev, V., Semerikov, S., & Solovieva, V. “Detecting stock crashes using Levy distribution. In CEUR Workshop Proceedings”. – 2019 [Електронний ресурс]. Режим доступу: [http://ceur-ws.org/Vol-2422/paper\\_34.pdf](http://ceur-ws.org/Vol-2422/paper_34.pdf).
8. Y. K. Pardeshi and P. P. Kale, “Technical analysis indicators in stock market using machine learning”, 2021 [Електронний ресурс]. Режим доступу: <https://ieeexplore.ieee.org/document/9580172>
9. J.-Z. Huang, W. Huang, and J. Ni, “Predicting bitcoin returns using high-dimensional technical indicators,”, Sep. 2019 [Електронний ресурс]. Режим доступу: <https://www.sciencedirect.com/science/article/pii/S2405918818300928>



10. Scarpino M. TensorFlow. Hoboken, New Jersey : John Wiley & Sons, Inc., 2018 – С. 335-336.
11. K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “Lstm: A search space odyssey”. – 2017. [Электронный ресурс]. Режим доступа: <https://ieeexplore.ieee.org/abstract/document/7508408>
12. S. Hiregoudar, “Ways to Evaluate Regression Models”. – 2022. [Электронный ресурс]. Режим доступа: <https://towardsdatascience.com/ways-to-evaluate-regression-models-77a3ff45ba70>
13. M. Iqbal, M. Iqbal, F. Jaskani, K. Iqbal, and A. Hassan, “Time-Series Prediction of Cryptocurrency Market using Machine Learning Techniques”. – 2021. [Электронный ресурс]. Режим доступа: <http://eudl.eu/doi/10.4108/eai.7-7-2021.170286>
14. J. Kim, S. Kim, H. Wimmer, and H. Liu, “A cryptocurrency prediction model using lstm and gru algorithms,”. – 2021. [Электронный ресурс]. Режим доступа: <https://ieeexplore.ieee.org/document/9581397>
15. Agrawal, M., Khan, A. U., Shukla, P. K. “Stock price prediction using technical indicators: a predictive model using optimal deep learning.”. – 2019. International Journal of Recent Technology and Engineering.
16. Bustos, O., Pomares, A., Gonzales, E. “A Comparison between SVM and Multilayer Perceptron in Predicting an Emerging Financial Market: Colombian Stock Market. Congreso Internacional de Innovaciony Tendencias en Ingenieria”. – 2017. Institute of Electrical and Electronics Engineers.
17. Wang, Q., Xu, W., Zheng, H. “Combining the wisdom of crowds and technical analysis for financial market prediction using deep random subspace ensembles.”. – 2018.
18. Labiad, B., Berrado, A., Benabbou, L. “Machine Learning Techniques for Short Term Stock Movements Classification for Moroccan Stock Exchange”. – 2016. 11th International Conference on Intelligent Systems: Theories and Applications, Morocco.

19. Ghanavati, M., Wong, R., Chen, F., Wang, Y., Fong, S. “A Generic Service Framework for Stock Market Prediction.”. – 2016. International conference on services computing, San Francisco.
20. Patel, M. M., Tanwar, S., Gupta, R., Kumar, N. “A deep learning-based cryptocurrency price prediction scheme for financial institutions”. – 2020. Journal of Information Security and Applications.
21. Shynkevich, Y., McGinnity, T., Coleman, S., Li, Y., Belatreche, A. “Forecasting Stock Price Directional Movements Using Technical Indicators: Investigating Window Size Effects on One-Step-Ahead Forecasting”. – 2014. Conference on computational intelligence for financial engineering, proceedings, London.
22. Y. Liu, Z. Li, R. Nekhili, and J. Sultan, “Forecasting cryptocurrency returns with machine learning”. – 2023. [Електронний ресурс]. Режим доступу: <https://www.sciencedirect.com/science/article/pii/S0275531923000314>
23. Савчук О.В., Кривенко К.С. Інтелектуальний аналіз діагностичної інформації електро- радіокомпонентів в умовах невизначеності // Інтелектуальний аналіз інформації ІАІ- 2013 / Зб. праць. – К.: Просвіта. – 2013. – С.211-217
24. H. Lyu, “Cryptocurrency price forecasting: A comparative study of machine learning model in short-term trading,”. – 2022 [Електронний ресурс]. Режим доступу: <https://ieeexplore.ieee.org/document/9852573>
25. B. Mahesh, “Machine Learning Algorithms”. – 2019.
26. Y. K. Pardeshi and P. P. Kale, “Technical analysis indicators in stock market using machine learning: A comparative analysis”. – 2021. [Електронний ресурс]. Режим доступу: <https://ieeexplore.ieee.org/document/9580172>
27. T. Phaladisailoed and T. Numnonda, “Machine learning models comparison for bitcoin price prediction,”. – 2018, pp. 506–511. [Електронний ресурс]. Режим доступу: <https://ieeexplore.ieee.org/document/8534911>
28. E. Pintelas, I. Livieris, S. Stavroyiannis, T. Kotsilieris, and P. Pintelas, “Investigating the Problem of Cryptocurrency Price Prediction: A Deep Learning Approach”. – 2020.

29. R. M. Schmidt, “Recurrent Neural Networks (RNNs): A gentle Introduction and Overview”. – 2019.
30. P. P. Shinde and S. Shah, “A review of machine learning and deep learning applications,”. – 2018. [Электронный ресурс]. Режим доступа: <https://ieeexplore.ieee.org/document/8697857>
31. C. Zhang and P. Woodland, “High Order Recurrent Neural Networks for Acoustic Modelling”. – 2018.
32. Hitam, N. A., Ismail, A. R. “Comparative Performance of Machine Learning Algorithms for Cryptocurrency Forecasting.”. – 2018. [Электронный ресурс]. Режим доступа: <https://www.researchgate.net/publication/327415267>
33. Persio, L., Honchar, O. “Multitask machine learning for financial forecasting”. – 2018.
34. Rebane, J., Karlsson I. “RNNs and ARIMA models for cryptocurrency prediction: A comparative study”. – 2018. [Электронный ресурс]. Режим доступа: [https://doi.org/10.475/123\\_4](https://doi.org/10.475/123_4)
35. Yao, Y., Yi, J., “Predictive analysis of cryptocurrency price using deep learning”. – 2018.
36. J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling”. – 2014.
37. G. Biau, E. Scornet, “A random forest guided tour”. – 2016. [Электронный ресурс]. Режим доступа: <https://doi.org/10.1007/s11749-016-0481-7>
38. L. Deng, D. Yu, “Deep Learning: Methods and Applications”. – 2014.
39. V. Derbentsev, N. Datsenko, V. Babenko, “Forecasting cryptocurrency prices using ensembles-based machine learning approach”. – 2020. [Электронный ресурс]. Режим доступа: <https://ieeexplore.ieee.org/document/9468090>
40. G. Louppe, “Understanding Random Forests: From Theory to Practice”. – 2015.

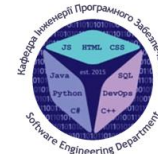
## ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-  
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ  
ТЕХНОЛОГІЙ

КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



### Магістерська робота

**«РОЗРОБКА МЕТОДУ АВТОМАТИЗОВАНОГО АНАЛІЗУ  
ДИНАМІКИ РУХУ ЦІН НА БІРЖАХ ВІРТУАЛЬНИХ АКТИВІВ»**

Виконав: студент групи ПДМ-64 Чернов Леонід Олександрович

Керівник: к.п.н., доц., зав. кафедри ІТ Корецька Вікторія Олександрівна

Київ - 2024

### МЕТА, ОБ'ЄКТА ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

**Мета роботи:** підвищення якості прогнозування за допомогою автоматизованого аналізу руху цін на біржах віртуальних активів

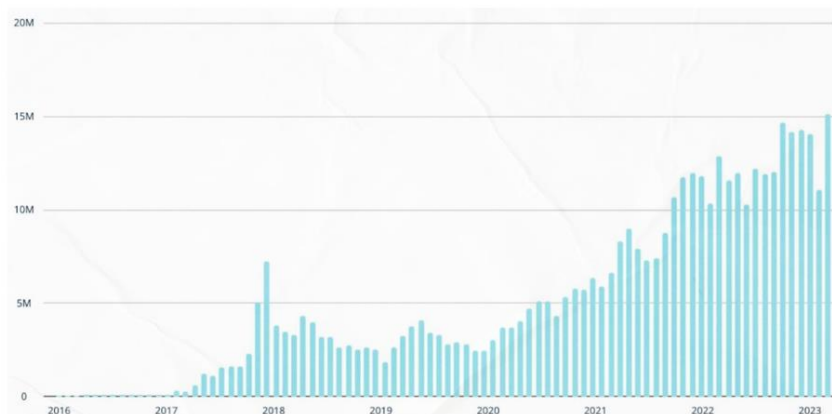
**Об'єкт дослідження:** аналіз руху ціни віртуальних активів.

**Предмет дослідження:** методи аналізу ринкових даних та рекурентні нейронні мережі для передбачення часових рядів.

## АКТУАЛЬНІСТЬ РОБОТИ

Інтерес громадськості до віртуальних активів в останні роки зростає в геометричній прогресії. Актуальність дослідження зумовлена зростаючим інтересом до інвестування у ринок віртуальних активів.

Тому, розробка нового методу автоматизованого аналізу динаміки руху цін на ринку віртуальних активів, може підвищити точність прогнозів та покращити результативність інвесторів.



Графік зростання кількості активних користувачів

3

## ПОРІВНЯННЯ МЕТОДІВ АНАЛІЗУ РУХУ ЦІН НА БІРЖАХ ВІРТУАЛЬНИХ АКТИВІВ

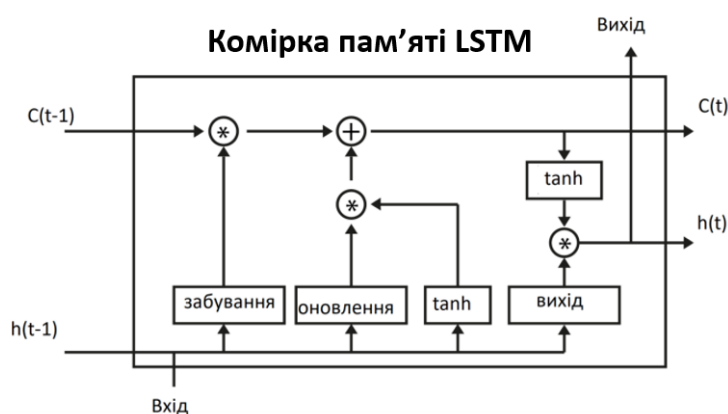
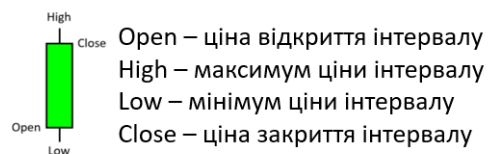
Назва	Переваги	Недоліки
Технічний аналіз	- визначає момент здійснення угоди для окремого активу	- не враховує поточні настрої учасників ринку, а лише спирається на дані та поведінку в минулому
Фундаментальний	- визначає ринковий тренд - враховує поточні настрої учасників ринку	- не вказує конкретного моменту здійснення угоди
Макроекономічний	- визначає багаторічний тренд для більшості світових ринків	- не враховує показники окремих активів, а враховує лише загальні показники окремого ринку
<b>Розроблений метод</b>	<b>- визначає момент здійснення угоди для окремого активу - враховує поточні настрої учасників ринку</b>	<b>- потребує обробки та зберігання великого об'єму даних</b>

4

## МАТЕМАТИЧНА МОДЕЛЬ МЕТОДУ АВТОМАТИЗОВАНОГО АНАЛІЗУ РУХУ ЦІН НА БІРЖАХ ВІРТУАЛЬНИХ АКТИВІВ

$$RSI = 100 - \frac{100}{1 - RS}, \text{ де}$$

$$RS = \frac{\text{середнє значення підвищення цін відкриття за N днів}}{\text{середнє значення зменшення цін закриття за N днів}}$$



$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i + \hat{y}_i)^2, \text{ де}$$

$y_i$  – фактичний очікуваний результат,  
 $\hat{y}_i$  – прогноз моделі

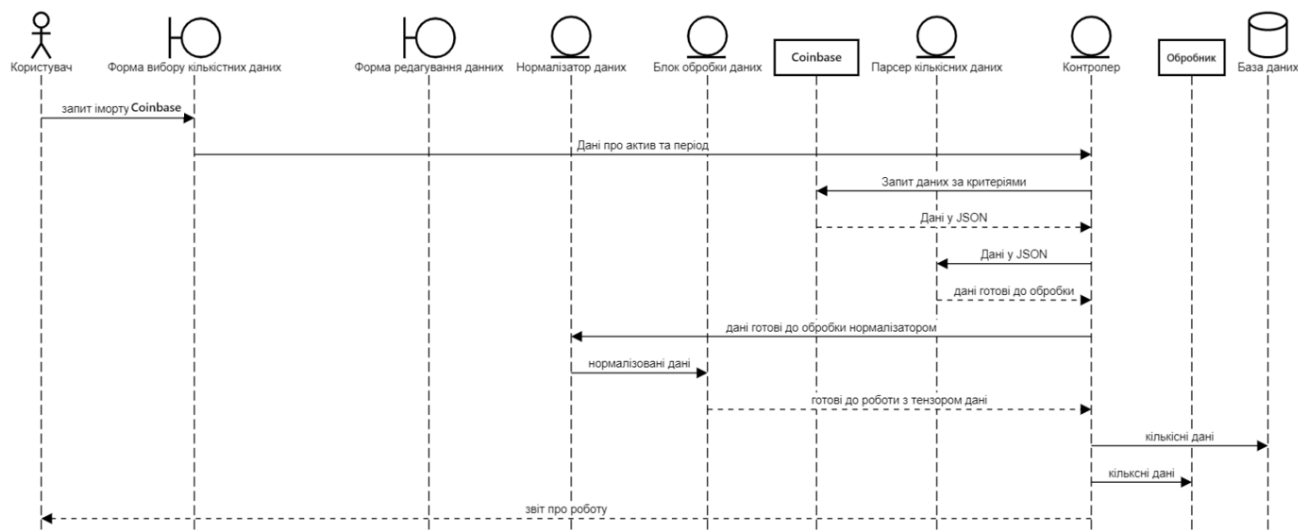
$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}, \text{ де}$$

$SS_{res}$  – сума квадратів різниць між фактичними значеннями та передбаченими

$SS_{tot}$  – це сума квадратів відхилень фактичних значень змінної від їх середнього значення

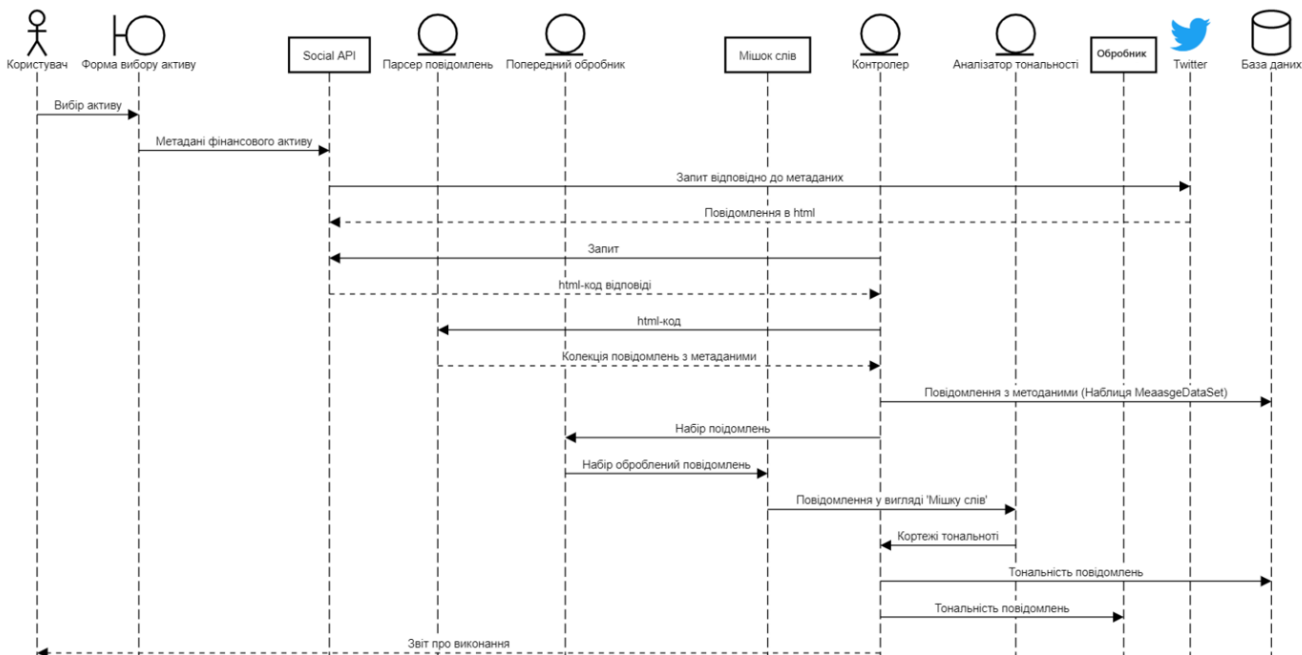
5

## ДІАГРАМА ПОСЛІДОВНОСТЕЙ ОБРОБКИ КІЛЬКІСНИХ ДАНИХ



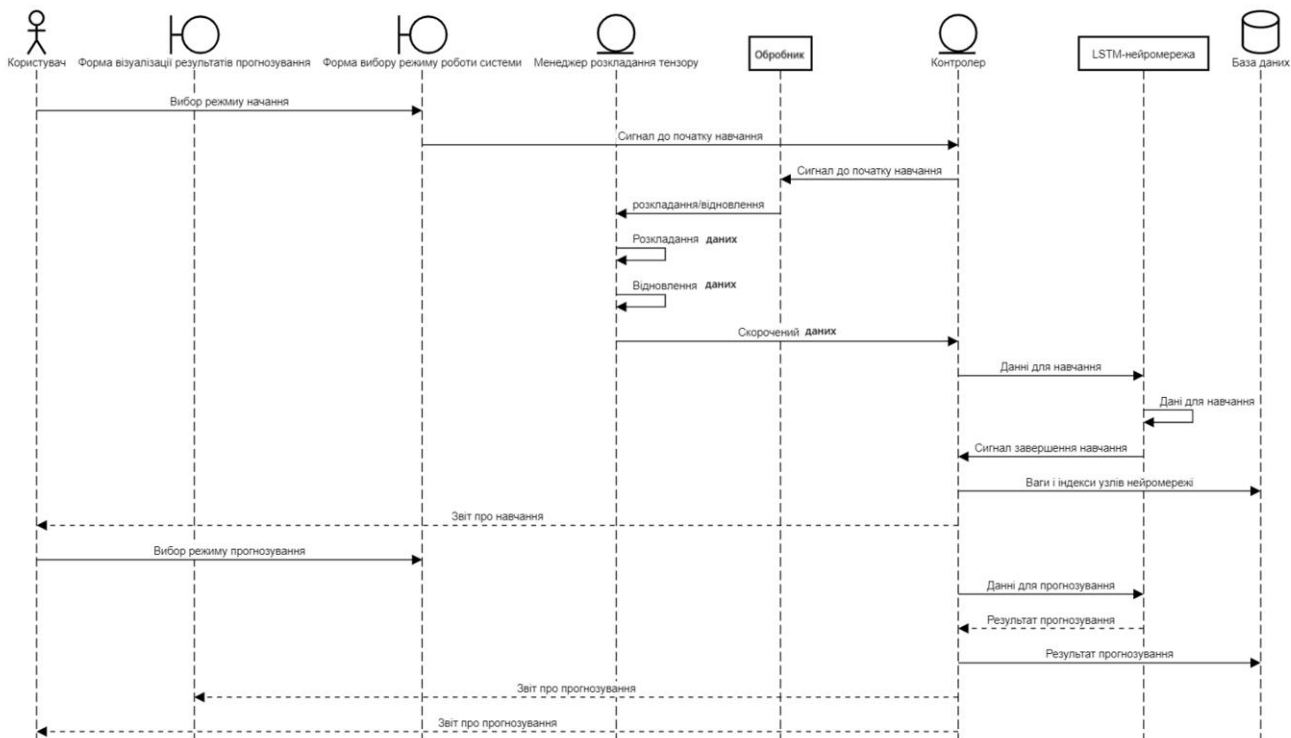
6

## ДІАГРАМА ПОСЛІДОВНОСТЕЙ ОБРОБКИ ПОВІДОМЛЕНЬ ТА НОВИН



7

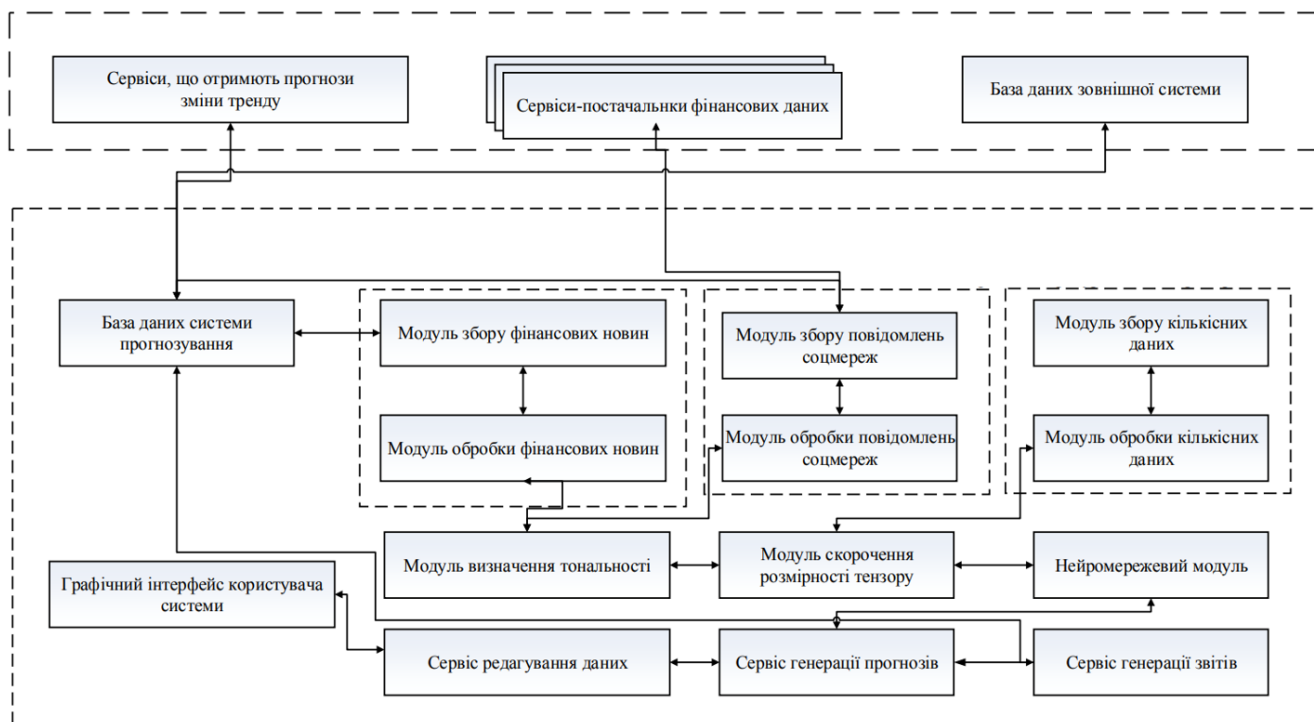
## ДІАГРАМА ПОСЛІДОВНОСТЕЙ ПРОЦЕСУ ПРОГНОЗУВАННЯ



8



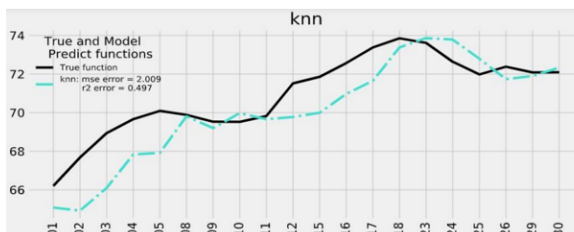
## АЛГОРИТМ РОБОТИ СИСТЕМИ



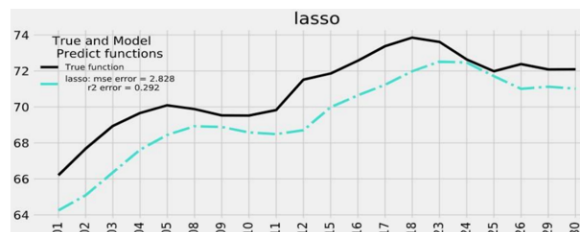
9

## ГРАФІЧНЕ ВІДОБРАЖЕННЯ РЕЗУЛЬТАТІВ

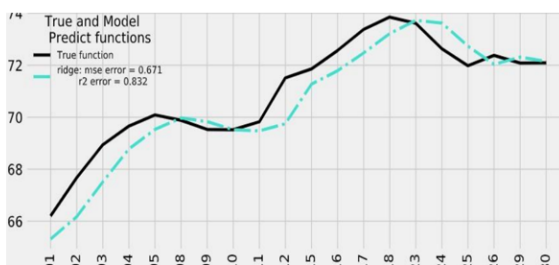
Роботи алгоритму KNN



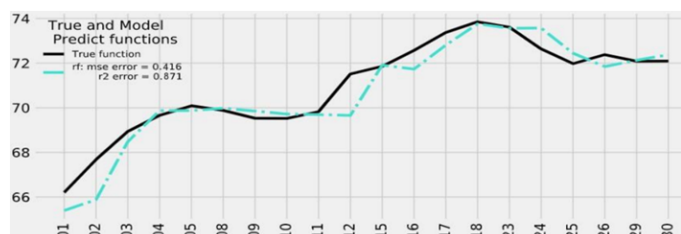
Робота лінійної регресії із застосуванням регуляризації L1 (lasso)



Роботи LSTM без показників тональності



Робота LSTM з урахуванням показників тональності та OHLC



10



## ВИСНОВКИ

В результаті роботи було визначено, що індикатори технічного аналізу, такі як: SMA, EMA, MACD, RSI та смуги Боллінджера довели свою неефективність, на більшості часових проміжках.

З метою вибору алгоритму для автоматизованого аналізу цін та покращення якості прогнозування, було проведено огляд таких алгоритмів: ML, LR, LSTM, RF. Оптимальні параметри для кожної моделі були знайдені шляхом виконання налаштування параметрів за допомогою набору даних перевірки.

Після порівняння метричних оцінок кожної моделі, було виявлено, що LSTM з оцінкою MSE (0,416) та оцінкою R2 (0.871) показала найкращі результати у прогнозуванні цін на віртуальні активи, у порівнянні з іншими досліджуваними моделями. Варто зазначити, що такого результату вдалось досягти, лише після додавання показників тональності як вхідних даних для навчання LSTM моделі, разом із даними технічного індикатору.

11

## ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

### Тези доповідей:

1. Чернов Л.О., Корецька В.О. Огляд існуючих підходів технічного аналізу для прогнозування руху цін на ринку цифрових активів. // Науково-практична конференція «Проблеми комп'ютерної інженерії» – Київ: ДУТ, 2023.
2. Чернов Л.О., Корецька В.О. Використання нейронних мереж для прогнозування цін на ринку віртуальних активів. // VI Міжнародної наукової конференції «Науковий простір: актуальні питання, досягнення та інновації» – Київ, 2023.

**ДЯКУЮ ЗА УВАГУ!**